Sparse Classification - Methods & Applications

Einarsson, Gudmundur

Publication date:
2018

Document Version
Publisher's PDF, also known as Version of record

Ph.D. Thesis
Doctor of Philosophy

DTU Compute
Department of Applied Mathematics and Computer Science

Sparse Classification

Methods & Applications

Gudmundur Einarsson

Kongens Lyngby 2018

DTU Compute
Department of Applied Mathematics and Computer Science
Technical University of Denmark

Matematiktorvet
Building 303B
2800 Kongens Lyngby, Denmark
Phone +45 4525 3031
compute@compute.dtu.dk
www.compute.dtu.dk
PHD-2018-471
ISSN: 0909-3192

Summary

With increasing number of more sophisticated tools to acquire data, we are faced with the important question of what matters in the sea of information at hand. This challenge is becoming more prevalent across virtually all scientific disciplines. Improvements over state of the art methods for analysing such data carry the potential to revolutionize tasks such as medical diagnostics where often decisions need to be based on only a few high-dimensional observations. This explosion in data dimensionality has sparked the development of novel statistical methods. In contrast, classical statistics build upon the assumption that we have more samples than variables, and the main asymptotic results, such as the central limit theorem, reflect that. As the assumption of having many samples does not hold for modern datasets, we need new tools and methods to find the signal within the dataset which is predictive of the relevant response variable. The focus in this thesis is on sparse methods where sparse implies that the method selects only a few variables.

Different types of data call for different methods. In this thesis the sparse methods we study concern settings where the response variable is ordinal. Such ordinal labeling is common in many fields, for example, medical doctors often summarize their observations into a single class of disease severity, which is known as a medical rating score. Automation offers the potential to improve both the reliability and objectivity of such tasks.

To demonstrate the effectiveness of the sparse methods developed in this thesis, they were applied to both challenging and diverse real-world problems: Predicting the severity of motion disorders from Parkinson's patients, generating short summaries of content from hundreds of online user reviews and detecting foreign objects from Multispectral X-ray scans. It may be noted, that to achieve these results, novel optimization approaches and open-source software were implemented.

Resumé

Med et øget antal af sofistikerede værktøjer til indsampling af data, rejses det vigtige spørgsmål om, hvilken information der har betydning i dette hav af data. Disse værktøjer bliver mere udbredt i alle videnskabelige discipliner. Forbedringer af de nyeste metoder til analyse af sådanne data har potentialet til at revolutionere opgaver som medicinsk diagnostik, hvor beslutninger ofte skal baseres på kun få højdimensionelle observationer. Denne eksplosion i datadimensionalitet har motiveret udviklingen af nye statistiske metoder. I modsætning hertil bygger klassisk statistik på antagelsen om, at vi har flere målinger end variabler, og det afspejles i de vigtigste asymptotiske resultater, som f.eks. Central Limit Theorem. Da antagelsen om at have mange målinger ikke holder til moderne datasæt, har vi brug for nye værktøjer og metoder til at finde signalet i datasættet, som kan forudsige den relevante responsvariabel. Fokus i denne afhandling er på sparse metoder, hvor sparse betyder, at metoden kun vælger få variabler.

Forskellige typer af data har brug for forskellige metoder. I denne afhandling sætter vi fokus på situationen hvor responsvariablen er ordinal. Ordinale labels er generalt brugt i mange fagområder, for eksempel opsummerer læger ofte deres observationer i et enkelt tal, der beskriver sygdomsgraden, som er kendt som en klinisk vurdering. Automatisering giver mulighed for at forbedre både pålideligheden og objektiviteten af sådanne opgaver.

For at demonstrere effektiviteten af de sparse metoder, der blev udviklet i denne afhandling, blev de anvendt til både udfordrende og forskellige virkelige problemer: 1) Forudsigelse af sværhedsgraden af bevægelsesforstyrrelser fra Parkinsons patienter. 2) Genererer korte resuméer af indhold fra hundredvis af onlinebruger anmeldelser. 3) Detektion af fremmedlegemer i multispectrale røntgenscanninger. Det kan nævnes, at for at opnå disse resultater blev nye optimeringsmetoder og open source software implementeret.

Preface

This PhD thesis was prepared at the Department of Applied Mathematics and Computer Science at the Technical Univeristy of Denmark (DTU) in fulfillment of the requirements for acquiring a Doctor of Philosophy degree (PhD) in applied mathematics with emphasis on Statistical Learning and Image Analysis.

The project is a cross-institutional collaboration between DTU Compute and Region Hovedstadens psykiatriske hospital. The Lundbeck Foundation and DTU funded the project.

The work conducted is a step towards improving and extending the tools available for classifying high dimensional data, where the number of variables vastly outnumbers the number of samples, as is often the case in medical studies, where the sample size is small, but the number of measurements per participant is high. The extensions are regarding enhanced optimization methods, the inclusion of unlabelled data and ordinal labels. The work involves applications concerning numerous data sets including multispectral X-ray images, online user reviews, and motion tracking data.

The thesis contains a general introduction to the project, an overview of sparse discriminant analysis and examples of applications. Finally, the appendix includes eight publications, which are either published or under submission.

The project was supervised by Associate Professor Rasmus R. Paulsen and co-supervised by Associate Professor Line Clemmensen, Professor Anders Fink-Jensen, and Associate Professor Anne Katrine Pagsberg. The research has been conducted at DTU in the Section of Image Analysis and Computer Graphics. I did external research at deCODE genetics in Iceland.

Kongens Lyngby, April 25, 2018

Gudmundur Einarsson

Dedicated to my family,
in particular to my father.

Acknowledgements

I would like to thank my principal supervisor Associate Professor Rasmus R. Paulsen for guidance, patience, expert advice, support and being empathetic. I would also like to thank my co-supervisor Associate Professor Line K. H. Clemmensen for guidance, support, specialist advice and the many great and fun brainstorming meetings we had. You two have made my PhD an enjoyable experience, where I have had a lot of freedom and opportunities to grow as a young researcher. I want to give special thanks to the Lundbeck Foundation, which have generously funded this project.

I would further like to thank my medical co-supervisors Professor Anders Fink-Jensen and Associate Professor Anne Katrine Pagsberg. This PhD had not been possible without your collaboration and expert knowledge. Further, I would like to thank my medical collaborator Ditte Rudå, who collected a lot of the data I worked on and discussed with me the details of motion disorders. I would like to give a special thanks to Jannik Boll Nielsen, who coded the Motor-game before me starting the PhD, and has provided me with great advice through my PhD.

Much of the work presented here is the results of collaborations with others, of which I am very grateful and thank everyone that has worked with me. In particular, I would like to thank Professor Bjarne Ersbøll and Associate Professor Anders Dahl for inviting me to work on the NEXIM project. I have learnt through the course of my PhD studies, that finding an interesting problem to solve is often a lot harder than finding a solution, Anders you are full of interesting problems! I hope I can work on some more with you in the future. I want to give special thanks to Associate Professor Brendan P. Ames at the University of Alabama and Summer Atkins for the collaboration on optimisation for sparse discriminant analysis. I have immensely enjoyed the collaboration, and working with someone in another country is a great learning experience.

I did my external stay at deCODE genetics in Iceland. There I was supervised by Hreinn Stefánsson who leads the CNS department, and Magnús Úlfarsson further guided me. In the four months I stayed at deCODE I was introduced to the world of genetics. I will never forget this experience, and I am forever grateful for the opportunity to stay there. Hreinn and Magnús are experts in their fields, and it was an honour to work under their guidance at deCODE.

I am forever grateful for the companionship from my fellow PhD students and faculty. To single anyone out would be unfair, I regard all of you as an extended family. I have thoroughly enjoyed all of our social events, which I hope you will continue to organise in the future! The faculty in the Image group consists of exceptionally bright and talented scholars; it has been a privilege to be with you for the past three years!

I would like to thank my friends and family, in particular, I would like to thank my brother Hafsteinn, who has shown great interest in my work and helped me both in problem solving and dissemination, I admire his dedication to science and attention to detail. I would also like to thank my friend and scholar Ólafur Birgir for discussions concerning my work. I want to give special thanks to my father who passed away the 11th of March 2017. He always regretted not seeking higher education, but he did not have the opportunity, since he lost his father at an early age, and was orphaned by his mother, who left for America with a soldier after the Second World War. He was fascinated by science and physics. He stimulated the minds of my brother and I, with riddles and stories. I am very fond of his story about the infinity machine, which he claimed to have built when he was a young boy. I had many questions about this machine, which was supposed to be able to generate endless energy. How big was it? Can it genuinely run forever? My father said that he could build it such that it fitted in a matchbox. Later I started asking why he had not sold the idea; he asked me what implications I thought that would have? Would he still be alive if people knew about this? These mental exercises made me more skeptical about the world, not to take everything for granted and ultimately led me to study mathematics at the University of Iceland. I will forever be grateful for the inspiration my father provided me with, and I will forever miss him.

The greatest thanks of all I give to my wife Hildur, who has shown me patience and support through the course of the PhD. *Þú og Matthias gerið líf mitt betra!*

List of Contributions

Included Thesis Contributions

The contributions of this thesis are listed in the order of appearance. The contributions are further grouped according to the main topics of the manuscripts, *methods & algorithms, applications and software*.

Methods & Algorithms

[A] Atkins, S., **Einarsson, G.**, Ames, B., & Clemmensen, L. (2017). *Proximal Methods for Sparse Optimal Scoring and Discriminant Analysis*. arXiv preprint arXiv:1705.07194 **In submission**. [Atk+17]

[B] **Einarsson, G.**, Paulsen R. R, Ames, B. (2018). *Semi-Supervised Sparse Discriminant Analysis*. **In submission**.

[C] Sjöstrand, K., Clemmensen, L. H., **Einarson, G.**, Larsen, R., & Ersbøll, B. (2018). *SpaSM: A Matlab toolbox for sparse statistical modeling*. Journal of Statistical Software. **Accepted for publication**. [Sjö+18]

[D] **Einarsson, G.**, Paulsen R. R, Ames, B., Clemmensen, L. (2018). *Sparse Interpretations of Online Reviews* **In submission**.

Applications

[E] **Einarsson, G.**, Jensen, J. N., Paulsen, R. R., Einarsdottir, H., Ersbøll, B. K., Dahl, A. B., & Christensen, L. B. (2017, June). *Foreign Object Detection in Multispectral X-ray Images of Food Items Using Sparse Discriminant Analysis*. In Scandinavian Conference on Image Analysis (pp. 350-361). Springer, Cham. [Ein+17]

[F] **Einarsson, G.**, Clemmensen, L. K. H., Rudå, D., Fink-Jensen, A., Nielsen J. B., Pagsberg, A. K., Winge, K., & Paulsen, R. R. (2018) *Computer Aided Identification of Movements Related to Parkinson's Disease* **In submission**.

[G] Rudå, D., **Einarsson, G.**, Andersen, A. S. S., Nielsen, J. B., Correll, C., Winge K., Clemmensen L. K. H., Paulsen R. R., Pagsberg A. K., Fink-Jensen A. (2018). *Exploring Movement Impairments in Patients with Parkinson's disease using the Microsoft Kinect Sensor* **In submission**.

[**H**] Rudå, D., **Einarsson, G.**, Nielsen, J. B., Correll, C., Jensen, K. G., Klauber, D. G., Jeppesen, J. R., Fagerlund, B., Winge, K., Clemmensen L. K. H., Paulsen, R. R., Pagsberg, A. K., Fink-Jensen, A. (2018) *Exploring movements in adolescents with psychosis and healthy controls using the Microsoft Kinect sensor – a new tool for assessing drug-induced parkinsonism?* **In submission**.

Software

[**I**] **Einarsson, G.**, Atkins, S., Ames, B., & Clemmensen, L. (2017). *Accelerated Sparse Discriminant Analysis*. R package version 1 github.com/gumeo/accSDA [Ein+18]

Other Contributions

[**J**] Dahl, V. A., **Einarsson, G.**, Darvann, T. A., Hermann, N. V., Hove, H. B., Kakimoto, N., Kreiborg, S., & Dahl, A. B. (2016, April). Automatic measurement of orbital volume in unilateral coronal synostosis. In Biomedical Imaging (ISBI), 2016 IEEE 13th International Symposium on (pp. 889-893). IEEE. [Dah+16]

[**K**] Dal Corso, A., Olsen, M., Steenstrup, K. H., Wilm, J., Jensen, S., Paulsen, R. R., Eiríksson, E., Nielsen, J., Frisvad, J. R., **Einarsson, G.** & Kjer, H. M. (2015, November). *VirtualTable: a projection augmented reality game*. In SIGGRAPH Asia 2015 Posters (p. 40). ACM. [Dal+15]

[**L**] **Einarsson, G.**, Runarsson, T. P., & Stefansson, G. (2014, December). A competitive coevolution scheme inspired by Differential Evolution. In Differential Evolution (SDE), 2014 IEEE Symposium on (pp. 1-8). IEEE. [ERS14]

Abbreviations

ADMM Alternating Direction Method of Multipliers. 35

BLL Beer-Lambert Law. 61

BLUE Best Linear Unbiased Estimator. 24

CCA Canonical Correlation Analysis. 10–13, 28

CRAN Comprehensive R Archive Network. 71

DMRI Danish Meat Research Institute. 4, 61, 71

DTM Document Term Matrix. 68

DTU Technical Univeristy of Denmark. v, 4

EM Expectation-Maximization. 43

FDR False Discovery Rate. 25, 26

GWAS Genome Wide Association Study. 26

i. i. d. Independent and Identically Distributed. 20, 27

LDA Linear Discriminant Analysis. xvii, 7–16, 28, 45

MAP Maximum A Posteriori. 20

MDS-UPDRS *Movement Disorder Society Unified Parkinson's Disease Rating Scale.* 54–56

MLSS Machine Learning Summer School. 25

OLS Ordinary Least Squares. 23, 24

PCA Principal Component Analysis. 9, 28

PD Parkinson's Disease. 2, 3, 54

PLS Partial Least Squares. 13–15

PSD Positive Semi-Definite. 14

QDA Quadratic Discriminant Analysis. 16, 17

SAS Simpson Angus Scale. 54, 55

SDA Sparse Discriminant Analysis. xix, 7, 18, 22, 28, 35, 42, 44, 45, 62, 65, 67, 68, 72, 179

SOS Sparse Optimal Scoring. 18, 22

SVD Singular Value Decomposition. 62

Nomenclature

K Number of classes. 10

$\mathbf{\Omega}$ Elastic net coefficient matrix. 18

$\boldsymbol{\beta}$ Discriminant vector. 10

$\boldsymbol{\theta}$ Scoring vector. 11

$\mathbf{X}$ Data matrix. 9–11

$\mathbf{Y}$ Class indicator matrix. 10, 11

n Number of samples. 9

p Number of variables. 9, 16, 18

$||\bullet||_1$ 1-norm. 18

$||\bullet||_2$ 2-norm. 11

$||\bullet||_p$ p-norm. 18

Contents

CHAPTER 1

Introduction

Current advances in statistics and statistical learning center on problems where we have more features (p) than observations (n), these are known as $p \gg n$ problems. Classical statistics cannot handle more variables than samples, most of the estimation procedures require more samples than variables to work [Don+00]. As the amount of sensors, monitoring devices, and ways of tracking us is rapidly increasing there is a growing incentive to find a good solution to these problems.

When I attended the Machine Learning Summer School back in 2015 Emmanuel Candés described his interpretation of Big-Data. He saw Big-Data as a paradigm for statistical analysis. In the past we followed a stringent recipe for doing statistics:

1. Formulate a hypothesis
2. Collect data
3. Reject the hypothesis or not

Candés highlighted another approach, which is more common today: Collect data first, ask questions later. That is,

1. Collect data
2. Generate the hypotheses
3. Minimise the number of false discoveries

With this approach we aim to identify variables which genuinely predict the response. Finding these variables is relevant for many applications, such as genome wide association studies. In such studies the aim is to determine the combination of single nucleotide polymorphisms which predict a hereditary disease. Detecting these features can aid the development of novel, preventative and more effective medical treatments. Another clear example of a relevant application concerns motion disorders. The key question is whether we can identify from a motion-tracking time-series which types of motion best predict or correlate with a given score for severity of the disease?

I asked the online statistics community for a description of Big-Data[1]. Given the diverse responses we can conclude that there is no consensus on the matter. Ultimately I agree with Candés' description, which opens the door to an emerging type of research holding great potential in their applications.

[1] https://stats.stackexchange.com/questions/173060/what-exactly-is-big-data

Inspired by these new approaches, we develop sparse methods to solve $p \gg n$ classification problems, which we apply to real world applications. In the context of this thesis sparse relates to the paradigm mentioned by Candés. That is, we aim to select a sparse set of variables. This allows us to generate multiple hypotheses where the parameters corresponding to some variables are non-zero and most parameters are zero.

1.1 Motivation

The main motivation of this thesis is to characterise motion disorders based on multi-sensor input. We explore the use of range sensors for diagnostics of patients with either a Parkinson's Disease (PD) diagnosis or patients suffering from severe schizophrenia. In both cases the number of patients is limited and the amount of measured data is large (i.e. many variables). Motion disorders have various causes, they can be side-effects of drugs, or symptoms of a disease, which highlights the challenge of determining from motion data the underlying reason for the symptoms. Patients are commonly assessed and diagnosed with clinical rating scales. This is a manual inspection performed by a clinician, which introduces both score variation across clinicians and a potential operator bias. The score is usually an integer ranging from zero, (the subject is not affected), to four or five, (where the subject is severely affected). The highest score can mean that the subject is incapable of finishing the task under inspection.

PD is an example of a challenging case with no diagnostically conclusive test available. The current way of diagnosis sometimes results in misdiagnoses or does not detect the disease since many of the symptoms such as depression, dementia, tiredness, smell problems, blood pressure problems etc. are common to other conditions [Dun+14]. With a conclusive diagnosis the disease has often progressed to an advanced stage of motor symptoms and neurophysiological damage (See Fig. 1.1). It is thus of great importance to develop tools that aid an unbiased diagnosis for PD in an earlier stage of the condition as a preventative measure.

Managing to diagnose patients earlier is thus ambitious, especially if we only apply standard diagnostics tools once. It would be easier, more accurate, and less prone to bias, to make a computerised diagnostic test a part of the regular screening process. Using acquired data from such a test would allow us to model individual abnormalities more accurately, and make personalised and accurate objective predictions of disease status and progression, by comparing to earlier screenings. Another way to detect a condition would be to have access to a proxy variable, which the patient can decide to be analysed. The proxy variable can be data from a personal health monitor, movement data from a GPS tracker, mobile phone data, or data acquired by playing a video game.

In the presented work, we employ the Microsoft Kinect sensor [Sho+11] in a game-like environment to collect movement data on individuals suffering from motion-disorders, in particular, patients with PD and patients with severe Schizophrenia.

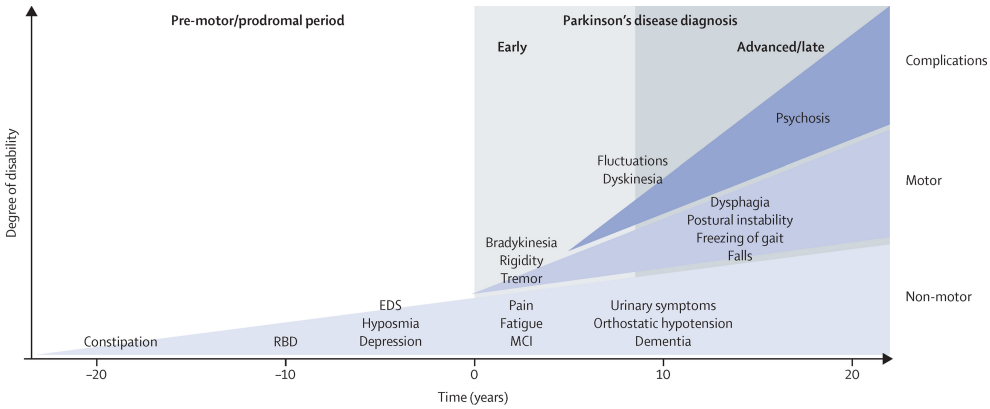


Figure 1.1: Diagram showing the progression of PD. The diagram covers the pre-motor symptom period, and indicates that diagnosis usually happens some time after the onset of motor symptoms. Taken from [KL15a].

These devices generate tracking data 30 times per second; so we can quickly obtain many variables on a single subject.

Compared to more traditional classification problems the main difference in this setting is the *ordinal* label (the clinical score by the doctor). Having ordinal labels means that there is some natural order or relationship in the response. The methods developed in this thesis are tailored for ordinal, $p \gg n$ data. We highlight the generality of the methods by applying them to other problem domains, such as the summarisation of online reviews (See Chapter 7) and foreign object detection in Multispectral X-ray images.

A considerable amount of the work in this thesis is not entirely reflected in the results. Significant effort went into making the classification methods faster by using novel algorithms for optimisation. We further contribute back to the community which made all of this work possible by developing the tools in an open-source manner.

1.2 Objectives

The main objectives, forming the basis of the included contributions are:

1. Acquire and analyse motion data from patients (PD and Schizophrenia) and controls.

2. Develop and evaluate classification methods that can handle ordinal clinical labels.

3. Evaluate the generalisability of the methods by applying and evaluating them to data from other domains.

These objectives were accomplished in collaborations with researchers at DTU, Danish Meat Research Institute (DMRI), Region Hovedstaden and the University of Alabama.

1.3 Outline & Reading Guide

The rest of the thesis is split into three parts. This first one covers the relevant methodology, algorithms, background and related technical work. The first part also summarises the technical part of contributions A, B, C and D.

Part two is about the applications and contains three chapters, the first on motion tracking data, the second on foreign object detection in multispectral X-ray images and the third on natural language processing. There is substantial related work in these chapters, but we refer the reader to the papers for a full treatment of it.

Finally, part three contains all the contributions. The first two parts are summaries of the papers and tie the contributions together. We recommend the reader first to read the material presented in the thesis before digging into the manuscripts in the appendix. It is best to start by finishing part one and two, and then seek answers to the questions that arise in the relevant papers in part three.

Part I

Methods

CHAPTER 2
Background & Related Work

In this chapter, we summarise the background that leads to the development of SDA. We do not need to go far back in time to the infancy of linear classifiers; it started with the influential work of the statistician and biologist Sir Ronald Fisher.

2.1 Historical Perspective

It is helpful to look at the historical developments of linear classification to put the development of the current state of SDA into perspective. The earliest example of LDA is the one developed by Sir Ronald Fisher in 1936 [Fis36]. The method is sometimes referred to as simply Fisher's linear discriminant. Fisher's paper is also recognised for introducing the *Iris* dataset, which is one of the most famous classification datasets available for study and often used to introduce the concept of classification to students.

Fisher presents the method for classifying between two classes. He does not make any particular distributional assumptions, and describes a formula, (or rather *arithmetical procedure*), for calculating the discriminant vector $\boldsymbol{\beta}$. We are given two classes of observations, with means $\boldsymbol{\mu_1}$ and $\boldsymbol{\mu_2}$ and covariances $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_2$. Fisher proposes to maximise the entity:

$$S(\boldsymbol{\beta}) := \frac{(\boldsymbol{\beta}(\boldsymbol{\mu_1} - \boldsymbol{\mu_2}))^2}{\boldsymbol{\beta}^T(\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2)\boldsymbol{\beta}} =: \frac{\sigma^2_{\text{between}}}{\sigma^2_{\text{within}}}. \tag{2.1}$$

Fisher's original notation does not portray the numerator as the variation between classes, instead only as a linear combination of the difference between the means. He describes a method to solve for $\boldsymbol{\beta}$, which is mostly maximum likelihood estimation. One important thing to note regarding Fisher's approach is that he starts off by stating that he is looking for a *linear function*, not a discriminant vector. Using linear functions is essential concerning how functional analysis is used as a tool for Statistical and Machine Learning theory. Fisher approached the problem from a general perspective, but he missed the opportunity to generalise the method to more classes.

Fisher supervised Calyampudi Radhakrishna Rao (C. R. Rao) in his PhD. In 1948 his paper *The utilisation of multiple measurements in problems of biological classification* was published [Rao48; Rao47]. He had generalised Fisher's work in the section *The Problem of Three and More Groups* of his work. Rao starts off with a

general theory for multi-class classification and finally draws up multi-class LDA in the section *Application to multivariate normal populations with common dispersion matrices*. Not only did Rao generalise Fisher's work to handle multiple classes, but he also formalised the assumptions needed to arrive at this generalisation. So now the generalised quantity corresponding to S, (for K classes), in Eq. 2.1 becomes:

$$S(\boldsymbol{\beta}) := \frac{\boldsymbol{\beta}^T \boldsymbol{\Sigma}_b \boldsymbol{\beta}}{\boldsymbol{\beta}^T \boldsymbol{\Sigma}_w \boldsymbol{\beta}}, \tag{2.2}$$

where $\boldsymbol{\Sigma}_b$ and $\boldsymbol{\Sigma}_w$ are defined as:

$$\boldsymbol{\Sigma}_b := \frac{1}{K} \sum_{i=1}^{K} (\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T, \quad \boldsymbol{\Sigma}_w := \frac{1}{N} \sum_{i=1}^{K} \sum_{j=1}^{N_i} (\mathbf{x}_{i,j} - \boldsymbol{\mu}_i)(\mathbf{x}_{i,j} - \boldsymbol{\mu}_i)^T. \tag{2.3}$$

Here class i includes N_i elements, where $\mathbf{x}_{i,j}$ refers to measurement j from class i and $\boldsymbol{\mu}_i$ is the mean of class i. $\boldsymbol{\Sigma}_b$ is referred to as the *mean scatter matrix*. Maximising S now amounts to find the $\boldsymbol{\beta}$ vectors that make the numerator as large as possible, and the denominator as small as possible. If we write out the numerator in Eq. 2.2 explicitly, we get:

$$\boldsymbol{\beta}^T \boldsymbol{\Sigma}_b \boldsymbol{\beta} = \boldsymbol{\beta}^T \left(\frac{1}{K} \sum_{i=1}^{K} (\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T \right) \boldsymbol{\beta} = \frac{1}{K} \sum_{i=1}^{K} (\boldsymbol{\beta}^T \boldsymbol{\mu}_i - \boldsymbol{\beta}^T \boldsymbol{\mu})^2. \tag{2.4}$$

Using this formulation clarifies further what is meant by scattering, i.e., we seek a vector $\boldsymbol{\beta}$, such that the projected means $\boldsymbol{\beta}^T \boldsymbol{\mu}_i$ are as far away from the projected general mean $\boldsymbol{\beta}^T \boldsymbol{\mu}$ as possible. Of course we need the denominator, otherwise, for a given solution we could always increase it in magnitude to get a better one.

We have only discussed the first discriminant vector, but we are only limited by the rank of $\boldsymbol{\Sigma}_b$, which is $K - 1$. Commonly the full LDA problem is presented as follows, for finding the k-th discriminant vector:

$$\boldsymbol{\beta}_k := \arg\min_{\boldsymbol{\beta}_k} \frac{\boldsymbol{\beta}_k^T \boldsymbol{\Sigma}_b \boldsymbol{\beta}_k}{\boldsymbol{\beta}_k^T \boldsymbol{\Sigma}_w \boldsymbol{\beta}_k} \tag{2.5}$$

$$\text{subject to} \quad \boldsymbol{\beta}_k^T \boldsymbol{\Sigma}_w \boldsymbol{\beta}_k = 1, \quad \text{and} \quad \boldsymbol{\beta}_k^T \boldsymbol{\Sigma}_w \boldsymbol{\beta}_l = 0, \ \forall l < k. \tag{2.6}$$

Equation 2.2 requires a norm-constraint, similar to the one above for a unique solution.

2.2 Dimensionality Reduction & Inference

Note that minimising S with respect to $\boldsymbol{\beta}$ has not yielded any classification boundaries directly, only vectors $\boldsymbol{\beta}$ for projecting the data to a lower dimensional representation.

The projected data satisfies the property that the means are as scattered as possible, and the internal variation for each group is as small as possible, (assuming dispersion matrices are all the same and Gaussian data). The method proposed by Rao is in some sense a supervised version of Principal Component Analysis (PCA) [Pea01; FHT01]. Using the Gaussian assumption further, we can additionally demonstrate that the region of equal probability between two multivariate Gaussian densities, with the same dispersion matrix, is, in fact, a hyperplane, which makes the separation linear, or the L in LDA. The classification is commonly carried out by assigning a new point to the class which has the closest mean calculated using the Mahalanobis distance [Mah36] imposed by the common dispersion matrix.

The quotient in Eq. 2.2 is a Rayleigh quotient and can be solved as a generalised eigenvalue problem. The rank of $\mathbf{\Sigma}_b$ is maximum $K-1$, which is an upper bound on the number of different discriminant vectors $\boldsymbol{\beta}$ we can find, and an upper bound on the dimensionality of the lower dimensional embedding. We can see that the problem in Eq. 2.2 is a generalised eigenvalue problem by calculating the gradient of S with respect to $\boldsymbol{\beta}$:

$$\nabla_{\boldsymbol{\beta}} S := \frac{2\mathbf{\Sigma}_b\boldsymbol{\beta}(\boldsymbol{\beta}^T\mathbf{\Sigma}_w\boldsymbol{\beta}) - 2(\boldsymbol{\beta}^T\mathbf{\Sigma}_b\boldsymbol{\beta})\mathbf{\Sigma}_w\boldsymbol{\beta}}{(\boldsymbol{\beta}^T\mathbf{\Sigma}_w\boldsymbol{\beta})^2} \tag{2.7}$$

$$= \frac{2\mathbf{\Sigma}_b\boldsymbol{\beta} - 2\frac{\boldsymbol{\beta}^T\mathbf{\Sigma}_b\boldsymbol{\beta}}{\boldsymbol{\beta}^T\mathbf{\Sigma}_w\boldsymbol{\beta}}\mathbf{\Sigma}_w\boldsymbol{\beta}}{\boldsymbol{\beta}^T\mathbf{\Sigma}_w\boldsymbol{\beta}} \tag{2.8}$$

$$= \frac{2\mathbf{\Sigma}_b\boldsymbol{\beta} - 2S(\boldsymbol{\beta})\mathbf{\Sigma}_w\boldsymbol{\beta}}{\boldsymbol{\beta}^T\mathbf{\Sigma}_w\boldsymbol{\beta}} \tag{2.9}$$

Now if we solve for $\nabla_{\boldsymbol{\beta}} S = \mathbf{0}$ we get:

$$\mathbf{\Sigma}_b\boldsymbol{\beta} = S(\boldsymbol{\beta})\mathbf{\Sigma}_w\boldsymbol{\beta}, \tag{2.10}$$

which is better recognisable generalised eigenproblem. The vectors $\boldsymbol{\beta}$ that we now seek are eigenvectors of the matrix $\mathbf{\Sigma}_w^{-1}\mathbf{\Sigma}_b$, assuming of course that $\mathbf{\Sigma}_w$ is positive definite. In the case that we have fewer observations than variables, $p \gg n$ problems, the matrix $\mathbf{\Sigma}_w$ is singular, and this method essentially breaks down. Some generalisations exist to deal with this particular case, one is indeed zero-variance discriminant analysis [AH16], where we begin by assuming that the solution lies in the null space of $\mathbf{\Sigma}_w$, along with other assumptions.

2.3 Other Approaches for LDA

There are different ways to approach discriminant analysis with linear boundaries, here we describe a couple of alternative approaches and how they provide different generalisations. Note that when we refer to LDA in the text, we mean the version made by Rao. Generally in this section $\mathbf{X}$ refers to the $n \times p$ data matrix with n

samples and p variables, $\mathbf{Y}$ is an $n \times K$ indicator/dummy/one-hot-encoding matrix defined in Eq. 2.11. Other entities are defined as they appear, but otherwise, we refer the reader to the terminology.

2.3.1 Canonical Correlation Analysis

The same year that Fisher published his work on the Iris dataset, Harold Hotelling's theoretical work *Relations Between Two Sets of Variates* was published [Hot36]. Hotelling presents Canonical Correlation Analysis (CCA) in the paper; he begins with an interesting, motivating example:

> Concepts of correlation and regression may be applied not only to ordinary one-dimensional variates but also to variates of two or more dimensions. Marksmen side by side firing simultaneous shots at targets, so that the deviations are in part due to independent individual errors and in part common causes such as wind , provide a familiar introduction to the theory of correlation; but only the correlation of the horizontal component is ordinarily discussed, whereas the complex consisting of horizontal and vertical deviations may be even more interesting.
>
> — Harold Hotelling

CCA is the problem of finding linear functions (vectors) that maximise the correlation between *two data matrices*. The data projected by these functions are called *canonical variables*. The formulation is so general that it contains most of the common parametric statistical tests [Kna78]. One special case of CCA is in fact LDA. If one of the matrices is a data matrix $\mathbf{X}$, and the other one is an indicator matrix of class belongings $\mathbf{Y}$, where:

$$Y_{ij} = \begin{cases} 1, & \text{sample } i \text{ in class } j \\ 0, & \text{otherwise,} \end{cases} \tag{2.11}$$

then the first linear function corresponding to the data matrix is in fact the same as the one corresponding to the first discriminant vector $\boldsymbol{\beta}$ for LDA, up to a small difference in constraints. The following minimisation problem provides a solution to CCA:

$$(\boldsymbol{\Theta}, \mathbf{B}) := \underset{\boldsymbol{\Theta}, \mathbf{B}}{\arg\min} \, ||\mathbf{Y}\boldsymbol{\Theta} - \mathbf{X}\mathbf{B}||_2^2 \tag{2.12}$$

$$\text{subject to} \quad \mathbf{B}^T\mathbf{X}^T\mathbf{X}\mathbf{B} = \mathbf{I}, \quad \text{and} \quad \boldsymbol{\Theta}^T\mathbf{Y}^T\mathbf{Y}\boldsymbol{\Theta} = \mathbf{I}. \tag{2.13}$$

We will mostly use the formulation in Eq. 2.12 to contrast the other related problems.

Recent work related to CCA, *Canonical Information Analysis*, where correlation is replaced with mutual information, can be found in [VN15; Yin04]. Replacing correlation with mutual information makes the optimisation harder, but the authors provide compelling empirical evidence for the usage.

2.3.2 Optimal Scoring

Trevor Hastie, Robert Tibshirani, and Andreas Buja present Optimal Scoring in their 1994 paper *Flexible Discriminant Analysis by Optimal Scoring* [HTB94]. They present the method as an alternative to CCA. Regarding the matrices $\mathbf{X}$ and $\mathbf{Y}$ presented in the last section, and the $K \times 1$ scoring vector $\boldsymbol{\theta}$, we can define the optimal scoring problem as:

$$(\boldsymbol{\theta}_k, \boldsymbol{\beta}_k) := \underset{\boldsymbol{\theta}_k, \boldsymbol{\beta}_k}{\arg\min} ||\mathbf{Y}\boldsymbol{\theta}_k - \mathbf{X}\boldsymbol{\beta}_k||_2^2 \tag{2.14}$$

$$\text{subject to} \quad \boldsymbol{\theta}_k^T \mathbf{Y}^T \mathbf{Y} \boldsymbol{\theta}_k = 1, \quad \text{and} \quad \boldsymbol{\theta}_k^T \mathbf{Y}^T \mathbf{Y} \boldsymbol{\theta}_l = 0, \ \forall l < k. \tag{2.15}$$

The major difference from the previous approaches is the usage of the 2-norm $|| \bullet ||_2$. The problem has been cast as a regression problem, where we regress $\mathbf{X}$ onto a transformed version of the dummy matrix $\mathbf{Y}$. We refer to the transformed values $\mathbf{Y}\boldsymbol{\theta}$ as *scores*. Hastie, Tibshirani, and Buja summarize the benefits of LDA nicely in their paper:

1. All relevant distance information is contained in the $K-1$ dimensional subspace spanned by the class centroids.
2. The decision boundaries are linear.
3. One can plot the data in a reduced space, giving a graphical representation of the group separation.
4. One does not need to use all the discriminant vector to represent the data in the lower dimensional embedding, in the case of little data with many classes, fewer dimensions can yield more stable and accurate classification.

They further describe how this formulation lends itself nicely to generalisation. Casting the problem into a regression framework allows us to adapt the tools in the regression toolbox for classification. The article further describes how they can adjust the classification as non-parametric regression, the primary point of the article is how to deal with the problem of underfitting, i.e., augmenting the data to another representation.

The authors published an accompanying paper in 1995, *Penalized Discriminant Analysis* [HBT95], there the authors show how they can deal with overfitting using l_2 regularization, or essentially a Tikhonov regulariser [Tik43]. In both cases, the authors approach the problem of Optimal Scoring. The authors prove in the appendix, the equivalence of penalised discriminant analysis and penalised optimal scoring. For the sake of completeness, let's prove that the classical LDA can be derived from Optimal Scoring. We will provide a proof which is a bit more modern concerning notation.

Theorem 1. *The first discriminant vector $\boldsymbol{\beta}$ of the Optimal Scoring problem defined in Eq. 2.14 is an optimal solution to the LDA problem defined in Eq. 2.5, with a norm constraint on the scatter matrix.*

Proof. Note that we can represent the first $\boldsymbol{\theta}$ in terms of $\boldsymbol{\beta}$ as the classical least squares solution:

$$\boldsymbol{\theta} = (\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{X}\boldsymbol{\beta} \tag{2.16}$$

Now let's derive the the LDA solution from the Optimal Scoring one. We add the norm-constraint to the cost function with a Lagrange multiplier.

$$\begin{aligned}\arg\min_{\boldsymbol{\theta},\boldsymbol{\beta}} ||\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}||_2^2 &= \arg\min_{\boldsymbol{\theta},\boldsymbol{\beta}} \langle \mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}, \mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}\rangle, \quad \text{s.t.} \quad ||\mathbf{Y}\boldsymbol{\theta}||_2^2 = 1 && (2.17)\\ &= \arg\min_{\boldsymbol{\theta},\boldsymbol{\beta},\lambda} \langle \mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}, \mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}\rangle + \lambda(\langle \mathbf{Y}\boldsymbol{\theta}, \mathbf{Y}\boldsymbol{\theta}\rangle - 1). && (2.18)\end{aligned}$$

Using the bilinearity of the inner product we now get:

$$= \arg\min_{\boldsymbol{\theta},\boldsymbol{\beta},\lambda} \boldsymbol{\theta}^T\mathbf{Y}^T\mathbf{Y}\boldsymbol{\theta} - 2\boldsymbol{\theta}^T\mathbf{Y}^T\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\beta}^T\mathbf{X}^T\mathbf{X}\boldsymbol{\beta} + \lambda(\boldsymbol{\theta}^T\mathbf{Y}^T\mathbf{Y}\boldsymbol{\theta} - 1). \tag{2.19}$$

Now when substitute Eq. 2.16 for $\boldsymbol{\theta}$ some of the matrices above simplify as follows:

$$\begin{aligned}\boldsymbol{\theta}^T\mathbf{Y}^T\mathbf{Y}\boldsymbol{\theta} &= \boldsymbol{\beta}^T\mathbf{X}^T\mathbf{Y}((\mathbf{Y}^T\mathbf{Y})^{-1})^T\mathbf{Y}^T\mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{X}\boldsymbol{\beta} && (2.20)\\ &= \boldsymbol{\beta}^T\mathbf{X}^T\mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{X}\boldsymbol{\beta} && (2.21)\\ &= \boldsymbol{\beta}^T\boldsymbol{\Sigma}_b\boldsymbol{\beta} && (2.22)\end{aligned}$$

In a similar fashion we get:

$$\begin{aligned}-2\boldsymbol{\theta}^T\mathbf{Y}^T\mathbf{X}\boldsymbol{\beta} &= -2\boldsymbol{\beta}^T\boldsymbol{\Sigma}_b\boldsymbol{\beta} && (2.23)\\ \boldsymbol{\beta}^T\mathbf{X}^T\mathbf{X}\boldsymbol{\beta} &= \boldsymbol{\beta}^T(\boldsymbol{\Sigma}_b + \boldsymbol{\Sigma}_w)\boldsymbol{\beta} && (2.24)\end{aligned}$$

Now we can substitute these results in the minimisation problem, which simplifies to:

$$\arg\min_{\boldsymbol{\beta},\lambda} \boldsymbol{\beta}^T\boldsymbol{\Sigma}_w\boldsymbol{\beta} + \lambda(\boldsymbol{\beta}^T\boldsymbol{\Sigma}_b\boldsymbol{\beta} - 1). \tag{2.25}$$

We recognise this as the Lagrange version of Eq. 2.5, where we minimise the inverse instead and the constraint is on the scatter matrix. □

One can in a similar manner show the equivalence of LDA and CCA. There is only a minor deviation between the three approaches regarding solutions, the differences are more concerning algorithms we can use to solve these problems, and how we can extend them and adapt to other situations. We have to remember that when Fisher's work was published in 1936, all the calculations were done by hand, that might create some bias regarding techniques to use. In the following section, we will inspect in more detail the subtle differences between Optimal Scoring, LDA and CCA.

2.3.3 Deviations of Linear Classifiers

The three methods, LDA, CCA and Optimal Scoring provide solutions which span the row-space (orthogonal complement to the nullspace) of $\mathbf{\Sigma}_b$. The only way the problems differ is precisely how the constraints are formulated. Optimal Scoring differs from LDA as demonstrated in the theorem above, i.e., the constraint is on the scatter matrix instead of the common covariance matrix. These constraints are also for preserving orthogonality. CCA is similarly related, where the constraint is with respect to the matrix $\mathbf{\Sigma}_b + \mathbf{\Sigma}_w$.

These are not major differences, but some of them lend themselves better to generalisations. Of all the formulations, Optimal Scoring lends itself best to generalisation, because it is so similar to traditional regression. The method was constructed precisely with that in mind. A good summary of eigenproblems in pattern recognition can be found in [DCR05].

2.3.4 Classification by Regression

One might ask himself/herself now, why? Why go through all this hassle instead of doing just regular multiple regression? Hastie, Tibshirani, and Buja discuss this matter in the flexible discriminant analysis article [HTB94]. They compare LDA via optimal scoring to *softmax* regression. This is essentially like Optimal Scoring, *without* the constraints containing $\mathbf{Y}$. The optimisation can be implemented as backpropagation for neural network architecture, with no hidden layers, and softmax activation on the output [Bri90]. Ignoring the constraint with $\mathbf{Y}$ causes bad decision boundaries when the centers of the classes are colinear, and the softmax solution does not take the covariance of the data into account.

2.3.4.1 Partial Least Squares

One regression approach, which is very similar to CCA, is Partial Least Squares (PLS) [Wol+84]. These methods look surprisingly similar, but there is a subtle difference between them. PLS can essentially be formulated as maximising covariance, instead of correlation. To contrast Eq. 2.12 we present the corresponding minimisation problem for PLS.

$$(\mathbf{\Theta}, \mathbf{B}) := \underset{\mathbf{\Theta},\mathbf{B}}{\arg\min} ||\mathbf{Y\Theta} - \mathbf{XB}||_2^2 \tag{2.26}$$

$$\text{subject to} \quad \mathbf{B}^T\mathbf{B} = \mathbf{I}, \quad \text{and} \quad \mathbf{\Theta}^T\mathbf{\Theta} = \mathbf{I}. \tag{2.27}$$

In Eq. 2.26, the only difference from 2.12 are the constraints. This means that if we are doing classification, the covariance of the individual classes is not modelled, or assumed to be spherical. There do also exist sparse approaches for PLS [CK10].

2.3.4.2 Visualising the Difference

The effects described above are best explained with recreations of the figures from the original paper [HTB94], see Fig. 2.1 and Fig. 2.2, we use PLS instead of softmax, it has the same effect. From a modelling point of view, the parameters in the regression models are estimated based on a conditional likelihood, conditioned on the label, while in LDA, we estimate the parameters based on the joint likelihood.

2.4 Altering Assumptions

If we focus on the original formulation of LDA in Eq. 2.5, there are a few key points in where this method can break down, and the assumptions can be invalid. The major ones are:

1. There are more variables than observation, so $\mathbf{\Sigma}_w$ is Positive Semi-Definite (PSD). It is singular, thus we cannot invert the matrix for a solution.

2. The classes do not share a common covariance matrix.

3. The classes are not normally distributed.

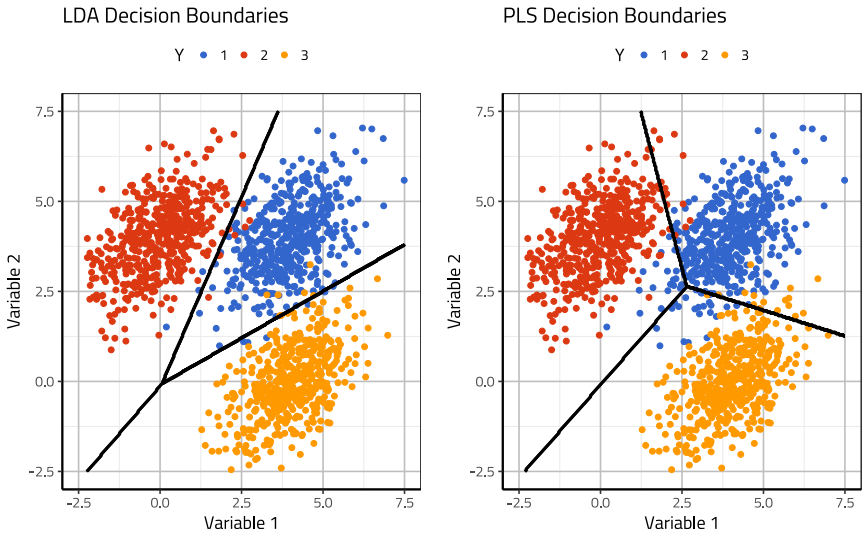


Figure 2.1: PLS uses the wrong metric. This data is sampled from bivariate Gaussians with the same covariance. Recreated from [HTB94].

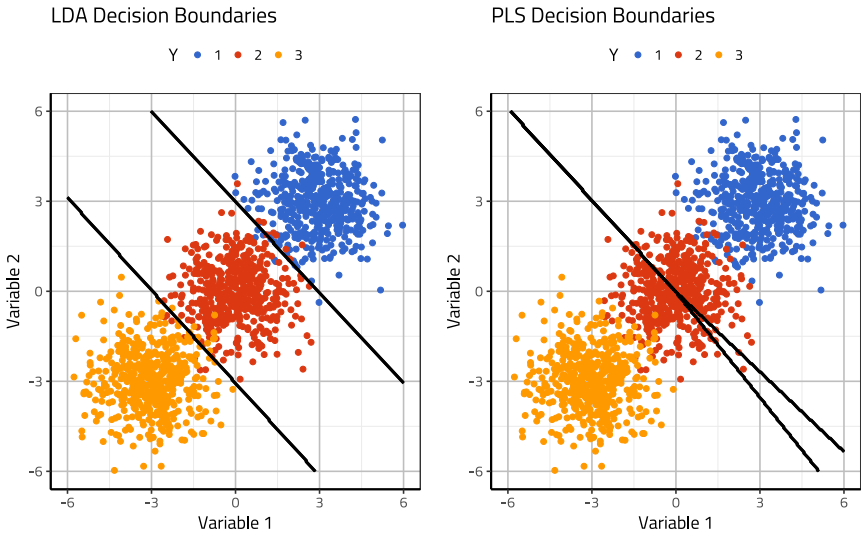


Figure 2.2: PLS cannot handle colinearity of centroids. Recreated from [HTB94].

There are numerous other issues, that can come up, e.g., missing data, label errors, etc. The focus of this thesis is mostly on how to deal with the first item, but let's investigate briefly what dropping the other two assumptions can lead to.

Rao derived LDA from a generic *multiclass-classification* point of view [Rao48]. Rao, as we mentioned before, derived LDA as a special case from his original assumptions. Friedman [Fri89] starts by deriving a general loss-based framework, which is a more modern approach to that of Rao. Friedman starts off with a generic loss function, where he only makes the assumption, that an item can only belong to one class:

$$L(k, \hat{k}), \quad 1 \leq k, \hat{k} \leq K, \tag{2.28}$$

k os the true label, and $\hat{k}$ is the predicted one. One can now model the probability of assigning a measurement to class k using a class conditional density $f_k(\mathbf{X}_i)$. We can interpret this as, *given this data, what is the probability it belongs to class k*? Friedman presents a general risk-minimisation problem as:

$$R(\hat{k}|\mathbf{X}_i) := \frac{\sum_{k=1}^{K} L(k, \hat{k}) f_k(\mathbf{X}_i)\pi_k}{\sum_{k=1}^{K} f_k(\mathbf{X}_i)\pi_k}, \tag{2.29}$$

where π_k is the prior for observing a member from class k. We of course identify the risk function as a generalisation of the Bayes-rule, where the minimisation leads precisely to the Bayes-rule. The loss that leads to LDA is then the 0-1 loss defined

as:

$$L(k, \hat{k}) = 1 - \delta(k, \hat{k}), \tag{2.30}$$

where δ is the Kronecker-delta. The 0-1 loss simplifies Eq. 2.29 to the classification rule:

$$f_{\hat{k}}(\mathbf{X}_i)\pi_{\hat{k}} := \max_{1 \leq k \leq K} f_k(\mathbf{X}_i)\pi_k. \tag{2.31}$$

Now the only part left is to choose a class-conditional density, so we can plug in for f_k the multivariate normal density function:

$$f_k(\mathbf{X}_i) := \frac{1}{\sqrt{(2\pi)^p \det(\mathbf{\Sigma}_k)}} \exp\left(\frac{-1}{2}(\mathbf{X}_i - \boldsymbol{\mu}_k)^T \mathbf{\Sigma}_k^{-1}(\mathbf{X}_i - \boldsymbol{\mu}_k)\right), \tag{2.32}$$

where $\boldsymbol{\mu}_k$ is the $p \times 1$ vector representing the mean of class k, where p is the number of variables. Plugging this into Eq. 2.31 and using the negative log-likelihood instead gives us the decision rule:

$$d_k(\mathbf{X}_i) := (\mathbf{X}_i - \boldsymbol{\mu}_k)^T \mathbf{\Sigma}_k^{-1}(\mathbf{X}_i - \boldsymbol{\mu}_k) + \ln(\det(\mathbf{\Sigma}_k)) - 2\ln(\pi_k). \tag{2.33}$$

So this amounts to minimising the Mahalanobis distance. Note that if we assume that the $\mathbf{\Sigma}_k$ are the same for each class, then this formulation implicitly defines the discriminant vectors. On the other hand, if we do not impose such a strict condition as the classes sharing a covariance matrix, we end up with Quadratic Discriminant Analysis (QDA). QDA derives its name from the fact that the decision boundaries are quadratic, instead of linear, (See Fig. 2.3 for an example of how these boundaries look). If we look at the level curves of quadratic form, like the one in Eq. 2.33, for $p = 2$, then we get conic-sections. Dropping the condition that the covariance matrices are identical may sound minuscule, but it has an entirely significant effect. The quadratic boundaries are more flexible, but they are also harder to interpret or understand. Using the linear discriminant functions makes it easier to understand which variables contribute the most to the discrimination.

Another potential issue with QDA is sample size. It is a critical factor for choosing between QDA and LDA [WK77]. If p is moderately large and we have multiple classes, then we have quite a large number of parameters to estimate. The sample size is one of the key points that Friedman addresses [Fri89]. Friedman proposes several models for the covariance matrices, making the total loss in degrees of freedom less severe as if all the matrices are estimated fully.

The issue of non-normality for LDA and QDA has of course been addressed before [LSR73]. The choice of normal distribution essentially boils down to tractability, simplicity, the fact that the normal distribution is often represented in nature, the central limit theorem and computational advantages.

From a modelling or application point of view, I would personally say that before I would consider trying to use another distribution than the normal, there would have

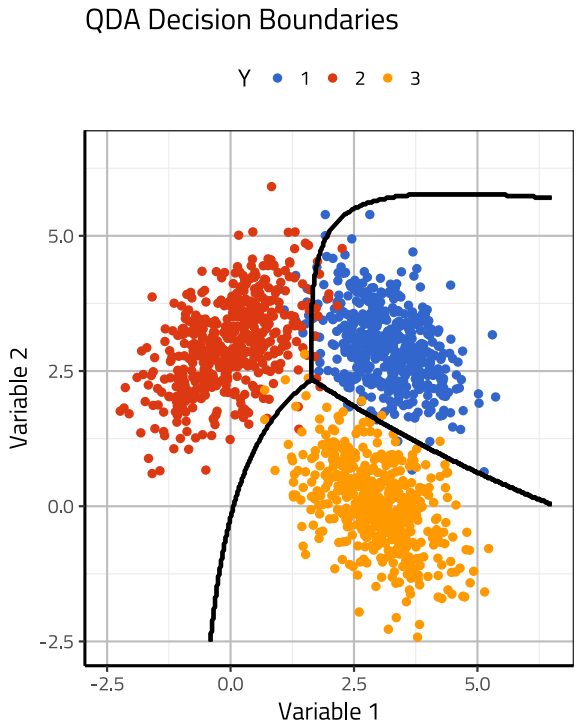


Figure 2.3: Example of QDA decision boundary. Data sampled from Gaussians with different covariance structure..

to be substantial evidence for the fact that the underlying physical process, generating the data, is not Gaussian. Non-normality could easily be the case for extreme situations, like extreme precipitation, or temperatures, where one often models these phenomena using an extreme value distribution, such as, e.g., the Gumbel distribution. Another flaw of the normal distribution is its inability to model outliers, and its tendency to be sensitive to them. One could therefore instead consider the student-t distribution or even the Cauchy distribution. In any case, the choice of distribution should be a well-grounded decision.

One thing to consider in the choice of distribution is of course, who is the audience of the results. If the primary objective is to create a robust classifier, which can be put into production, then the modelling choice should be most strongly supported by empirical evidence, such as test-error, which is a measure on how well the method generalizes to data, which was completely excluded from training. If we do not have enough data for this, then our primary concern should be to get more data, such that we have substantial empirical evidence for good performance. If however, we need to interpret the results, we might want to choose a simple model, which makes this

interpretation easier. The model selection of course also depends on the scientific question that we are trying to answer.

2.4.1 Validating Assumptions

Tests to validate that the covariance, representing the different classes are equal are commonly used. A classical test in that regard is Box's M-test [Box49]. That test is susceptible to non-normality, and if the covariances are unequal, but it cannot cover the $p \gg n$ case. Modern approaches to test for equality of covariance matrices exist, where one is presented in work by Ishii et al. [IYA16]. There the focus is on the eigenvalues and principal components, i.e., the tests are on low dimensional representations.

2.4.2 Sparse Discriminant Analysis

SDA was first presented in the paper by Clemmensen et al. [Cle+11] in 2011. The method is approached via Optimal Scoring, or rather a sparse version called Sparse Optimal Scoring (SOS):

$$(\boldsymbol{\theta}_k, \boldsymbol{\beta}_k) := \underset{\boldsymbol{\theta}_k, \boldsymbol{\beta}_k}{\arg\min} ||\mathbf{Y}\boldsymbol{\theta}_k - \mathbf{X}\boldsymbol{\beta}_k||_2^2 + \gamma \boldsymbol{\beta}_k^T \boldsymbol{\Omega} \boldsymbol{\beta}_k + \lambda ||\boldsymbol{\beta}_k||_1 \tag{2.34}$$

$$\text{subject to} \quad \boldsymbol{\theta}_k^T \mathbf{Y}^T \mathbf{Y} \boldsymbol{\theta}_k = 1, \quad \text{and} \quad \boldsymbol{\theta}_k^T \mathbf{Y}^T \mathbf{Y} \boldsymbol{\theta}_l = 0, \ \forall l < k. \tag{2.35}$$

γ and λ are regularisation parameters, commonly found via cross-validation. These parameters control the influence of the regularisers.

The only difference from Eq. 2.14 is the regularisation terms in the objective function, where $\boldsymbol{\Omega}$ is a $p \times p$ elastic net coefficient matrix and $|| \bullet ||_1$, is the l_1-norm. Generally the p-norm $|| \bullet ||_p$-norm is defined as:

$$||\boldsymbol{\beta}||_p := \left(\sum_{i=1}^{n} |\beta_i|^p \right)^{\frac{1}{p}} \tag{2.36}$$

see the level curves for different values of p in Fig. 2.4.

Hastie, Tibshirani and Buja had already introduced the *penalised* version of Optimal Scoring in [HBT95]. The word *penalised* is traditionally used by the statistics community to refer to regularisation, (more so before 2000). So the major new addition here is the inclusion of the l_1-norm. The l_1-norm is also what makes the problem significantly harder to solve than before, i.e., the l_1-norm is not differentiable at zero, making basic gradient-based approaches fail and *no closed form solution*, except for some special cases. The optimisation becomes, in general, more challenging.

2.4.2.1 The Lasso & the Elastic Net

SOS is sparse classification cast into a regression framework. This, of course, means that we have access to the tools available for regression to approach this problem. As

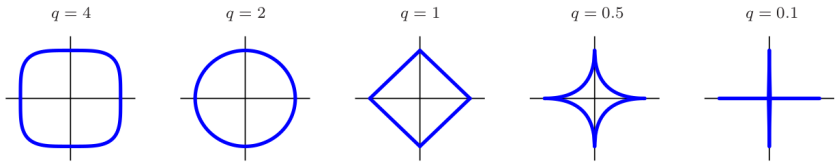


Figure 2.4: Level curves for different choices of p in the L_p-norm, where $||\boldsymbol{\beta}||_p = 1$. Taken from [HTW15].

mentioned earlier, the problem becomes significantly harder to solve when we add the l_1-norm, so there must be some benefits from using it. The advantage stems from the shape of the unit ball imposed by the l_1-norm; it is a diamond. Of all the L_p-norms, $p = 1$ is the smallest p where we still have a well-defined norm, and the level curves of the norm define convex sets.

The problem defined in Eq. 2.34 is in Lagrange form. We can also write the regularization term as the constraint:

$$||\boldsymbol{\beta}_k|| \leq t, \tag{2.37}$$

where there is a one to one correspondence between t and λ in Eq. 2.34, dependent on the data. The optimal solution will be at the intersection of the level curves defined by the l_1-norm and the *sum of squares term*. This idea is presented in Fig. 2.5. The level curves defined by the sum of squares term are an ellipsoid, so they are most likely to intersect with the diamond in one of the corners, or higher dimensional edges. The corners and edges are places where some of the parameters are zero, imposing a sparse solution. Having solutions at these corners means that we can **perform feature selection in the optimisation procedure**. Sparsity is a great property to have for the method. At the time that Lasso paper was published, the amount of available data was increasing with computing power, so there was, and still is, a strong incentive to perform excellent and efficient variable selection.

The Lasso is the central theme of the work presented by Tibshirani, who coins the term Lasso [Tib96]. He remarks that the case where $\mathbf{X}^T\mathbf{X} = \mathbf{I}$ had already been solved by Donaho et al couple of years earlier, in their work on wavelet basis functions [DJ94; DJ95]. Tibshirani further reflects on how the Lasso regulariser can be interpreted in a Bayesian setting as a prior, and he shows that it is equivalent to putting a *double exponential* prior on the parameters. Tibshirani did provide an algorithm to find a solution, but there are no guarantees on the runtime, and worst case it needs to try all combinations of parameters, which is not very efficient.

In theory, we can find the best solution by trying all combinations of variables, which are total 2^p, we refer to this problem as *the best subset problem*. That is not an effective strategy and explodes fast with the number of variables. Finding the best subset of variables essentially amounts to having the "zero-norm" as a regulariser.

Using the l_1-norm is in some sense the best convex relaxation to the best-subset problem, allowing us to use techniques from convex optimisation to solve it [BV04].

The most significant advancements for the Lasso came in 2004 when Efron et al presented the *least angle regression* algorithm to solve the Lasso [Efr+04], they call the algorithm LARS. They prove that the algorithm provides the correct solution and provide estimates for the degrees of freedom in the models among other statistics for model comparison.

$$\boldsymbol{\theta} := \arg\min_{\boldsymbol{\beta}} ||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||_2^2 + \gamma \boldsymbol{\beta}^T \boldsymbol{\Omega} \boldsymbol{\beta} + \lambda ||\boldsymbol{\beta}||_1 \tag{2.38}$$

The next logical step was, of course, to add the Tikhonov regularizer, which Zou and Hastie did in 2005 [ZH05], see Eq. 2.38. The commonly use an identity matrix for the regularizer, and then the regularizer is often referred to as the *Ridge Penalty*, [HK70], since we are adding a ridge to the diagonal of the covariance matrix. They show that the method outperforms the Lasso, specifically in the $p \gg n$ case. They also show how the elastic net encourages a grouping effect, where correlated variables are either selected or omitted together in groups.

2.4.3 Bayesian View on Regularisation

Independent of whether we approach the world from a Bayesian or Frequentist view, it is always nice and healthy to *put on the Bayesian goggles* to get a better understanding of regularisers.

Bayesian statisticians impose their beliefs and ideas about models and their parameters in the form of priors. The prior is a distribution of the parameters. Using priors sometimes helps when models are not identifiable, or we want to derive some more detailed information about the parameters. When a model is not identifiable, the Frequentist can resort to regularisers.

In a Bayesian setting, we sometimes have to provide point estimates of parameters, whereas traditionally Bayesian statisticians inspect properties of posterior distributions. These estimates are commonly Maximum A Posteriori (MAP) estimates. In a Bayesian regression model with normal errors, the conjugate prior for the regression parameters is the normal distribution. If we impose a normal prior on the regression parameters, the MAP estimate becomes equivalent to the maximum likelihood ridge regression solution [HK70]. So there is a natural correspondence between the two approaches. Now it is curious to see if there is a logical relation for the Lasso and the elastic net.

Park and Casella presented *the Bayesian Lasso* in 2008 [PC08]. They use the same prior as presented by Tibshirani in the original Lasso paper, Independent and Identically Distributed (i. i. d.) Laplace, (also called double exponential). They derive tractable full conditional distributions, allowing them to sample from the posterior in a Gibbs sampler. They run into similar issues with variable selection as with the Bayesian elastic net.

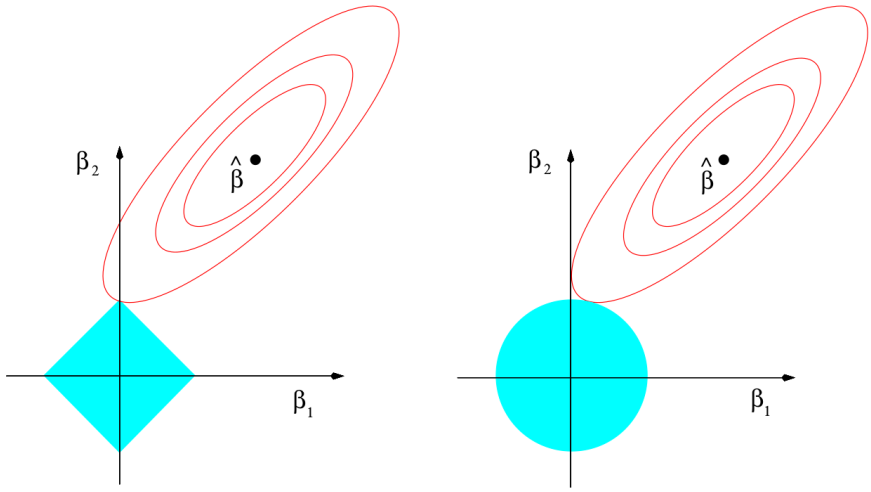


Figure 2.5: Feature selection effect of the Lasso. We can see on the left that the l_1-sphere, which looks like a diamond, intersects the level curves of the cost-function along the coordinates axis, meaning the value of the β_1 parameter is zero. This is a special property of the l_1-norm, whereas the sphere of the l_2-ball, seen on the right, does not portray a preference for an intersection along the coordinate axes, thus not imposing sparsity. Taken from [HTW15].

Li and Lin presented the Bayesian Elastic net in 2010 [L+10]. Li and Lin present the elastic net prior as:

$$\pi(\boldsymbol{\beta}) \propto \exp\left\{-\lambda||\boldsymbol{\beta}||_1 - \gamma||\boldsymbol{\beta}||_2^2\right\}. \tag{2.39}$$

They state that this is a compromise between Gaussian and Laplacian priors. Then they show how the conditional posterior of $\boldsymbol{\beta}$ is similar to the elastic net:

$$f(\boldsymbol{\beta}|\sigma^2, \mathbf{y}) \propto \exp\left\{\frac{-1}{2\sigma^2}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) - \lambda||\boldsymbol{\beta}||_1 - \gamma||\boldsymbol{\beta}||_2^2\right\}. \tag{2.40}$$

They further describe how this provides a hierarchical model, and how to derive a posterior distribution for a prior, which is conditional on σ^2, i.e., a little different from Eq. 2.39. This distribution is not a closed for of any other traditional distribution, but rather a mix of a multivariate normal and a truncated gamma distribution. The distribution is derived based on a noninformative prior for σ^2.

They mention how the regularisation parameters can be chosen via empirical Bayes [Cas01] and how the variable selection does not come as naturally as from the traditional approach. They view the variable selection as a hypothesis testing

problem and remove variables ad hoc by checking if zero is contained in a credible interval for the parameters, (this is similar for the Bayesian Lasso). They provide guiding heuristics for making this selection, but the user essentially has a parameter, the width of the interval, to select. The Bayesian community continuously seeks approaches to do a variable selection, where the most notable recent advances are the spike and slab prior [I+05; Lat+16], and penalising complexity priors [Sim+17].

It is interesting to see that the elastic net regulariser is not a well known prior, it is *a combination* of well-known distributions. It is simpler to look at the terms individually like Tibshirani did in the Lasso paper, but it is still possible to do something with the priors, although they do not have well defined closed form.

2.4.4 Sparse Discriminant Analysis Algorithm

The original algorithm used by Clemmensen et al. is a modification of least angle regression [Cle+11]. The algorithm is block-coordinate descent algorithm (See Alg. 1), where the algorithm alternates between updating $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ until convergence, or the maximum number of iterations is achieved.

There is no proof of convergence, but Witten and Tibshirani show that the critical points of SDA are the same as the ones in the Sparse Fisher Discriminant Analysis [WT11]. Witten and Tibshirani use minorisation approach to solve their formulation.

One of the significant criticisms of SDA is the lack of convergence results. One thing to note about the SOS problem formulated in Eq. 2.34, is, of course, the fact that it is non-convex. For a given $\boldsymbol{\beta}$ vector we can find $\boldsymbol{\theta}$ in polynomial time with the formula:

$$\boldsymbol{\theta}_{\text{New}} := \frac{(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{X}\boldsymbol{\beta}}{\sqrt{\boldsymbol{\beta}^T\mathbf{X}^T\mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{X}\boldsymbol{\beta}}}, \tag{2.41}$$

This is essentially the same $\boldsymbol{\theta}$ as the one we find in Eq. 2.16, except we scale it to ensure unit length.

Now, for a given $\boldsymbol{\theta}$ we mostly have an elastic net problem, so we can employ any algorithm that solves the elastic net to solve it. This block coordinate descent generic approach to the problem is depicted in Alg. 1. We will explore in Chapter 3 how we can apply alternative methods to the optimisation and prove convergence to stationary points.

2.5 Over- & Underfitting

The idea of regularisation is *parsimony* or simplicity. We prefer a simpler model, with fewer *effective parameters*, to a complex one. This ideology is ancient and essentially derives from a problem-solving principle called *Occam's razor*. One form of the principle that I am particularly fond of is by Thomas Aquinas, from his Summa Theologica:

Algorithm 1 Block Coordinate Descent for SDA Eq. 2.34

Start with initial iterate $\boldsymbol{\theta}^0$.
for $t = 0, 1, 2 \ldots$ until converged **do**
Update $\boldsymbol{\beta}^t$ as the solution of Eq. 2.38 with $\boldsymbol{\theta} = \boldsymbol{\theta}^t$ defining $\mathbf{y} := \mathbf{Y}\boldsymbol{\theta}^t$.
Update $\boldsymbol{\theta}^{t+1}$ by

$$\mathbf{w} = (\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{X}\boldsymbol{\beta}, \qquad \boldsymbol{\theta}^{t+1} = \frac{\mathbf{w}}{\sqrt{\mathbf{w}^T\mathbf{Y}^T\mathbf{Y}\mathbf{w}}}$$

end for

> It is superfluous to suppose that what can be accounted for by a few principles has been produced by many.

The razor part in the name refers to *shaving away* unnecessary assumptions.

When we create statistical models, classifiers, or algorithms that learn from data and make predictions, we should choose the simplest model that does the job. This choice may sound a bit arbitrary but think of the simple problem of predicting weight from height in humans. A natural hypothesis we might have about weight and height is that weight increases with height. To validate this hypothesis statistically, we project the real world measurements to the simple model of a line and make a test of whether the slope is positive or not.

This simple model might have been sufficient to provide an answer to our hypothesis, but if we want to predict weight from the height, then this simple model is maybe not enough, therefore we need tools to control the complexity sensibly.

We can see in Fig. 2.6 the effect of making a model too *complicated*. In Fig. 2.6 we draw 20 sample from the simple model:

$$y \sim x + \sin(x) + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, 0.5). \tag{2.42}$$

We add more and more columns to the data matrix $\mathbf{X}$, where we are increasing the polynomial basis we fit to the data. We can also just add random noise to the columns, and get a better fit, but then we can't visually demonstrate the effect of overfitting. We find the best polynomial parameters via Ordinary Least Squares (OLS), to do that we use the standard formula:

$$\mathbf{y} \sim \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \quad \hat{\boldsymbol{\beta}}_{\text{OLS}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}. \tag{2.43}$$

Note the matrix we invert to get the OLS solution. This matrix is of size $p \times p$, where p is the number of variables/columns in $\mathbf{X}$. This matrix is only invertible if the rank of $\mathbf{X}$ is p or greater. By construction, the matrix is positive semi-definite, if p is not high enough, so the smallest eigenvalue is zero. The matrix $\mathbf{A} := \mathbf{X}^T\mathbf{X}$ is a self-adjoint operator, this means that if we plug $\mathbf{A}$ into the polynomial $f(x) = x + c$, which in matrix terms would be $f(\mathbf{A}) = \mathbf{A} + c\mathbf{I}$, we can calculate the transformed eigenvalues by applying this same function. This result derives from the spectral theorem for

self-adjoint operators. Add the *ridge* means that we are essentially shifting all the eigenvalues up by a constant c, and can, therefore, invert this matrix and get a solution. The ridge solution, in correspondence to the OLS solution in Eq. 2.43 is presented below:

$$\hat{\boldsymbol{\beta}}_{\text{Ridge}} = (\mathbf{X}^T\mathbf{X} + \gamma\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}. \tag{2.44}$$

We can see the effect of tuning the parameter γ in Eq. 2.44 in Fig. 2.7. We start off with a polynomial of degree 14, and regularise it until we have $\gamma = 1$. The effective number of parameters p_{Eff} in the last example is 9.15, we can calculate it as the trace of the hat matrix:

$$p_{\text{Eff}} := \text{trace}\left(\mathbf{X}^T(\mathbf{X}^T\mathbf{X} + \gamma\mathbf{I})^{-1}\mathbf{X}^T\right) \tag{2.45}$$

2.5.1 Bias-Variance Tradeoff

One of the theoretical grounds for using regularisation, stems from the bias-variance tradeoff [Dom00; FHT01]. One of the properties that the OLS solution has, is that it is the Best Linear Unbiased Estimator (BLUE). This is results from the *Gauss-Markov* theorem [Pla50], where the assumptions are that the errors from the model satisfy the following properties:

1. Errors have expectation zero.
2. Errors are uncorrelated.
3. Errors are homoscedastic.

Now, if the OLS solution is BLUE, and the assumptions are met, why should we consider anything else? This boils down to the bias-variance tradeoff. So when we train a model with data, we are indirectly trying to optimise the generalisation error, but this error can be decomposed into three components:

$$\text{E}\left[(y - \hat{f}(x))^2\right] = \text{Bias}\left[\hat{f}(x)\right] + \text{Var}\left[\hat{f}(x)\right] + \sigma^2. \tag{2.46}$$

In the OLS case, the first component is zero, but the variance might actually be very big, the magnitude of the variance is related to the inverse of the smallest eigenvalue of $\mathbf{A} := \mathbf{X}^T\mathbf{X}$. So we might want to trade some of that variance for bias, to get better generalisation. Choosing the best bias amounts to selecting the best regularisation parameter, since that parameter controls the bias. This can be achieved via cross-validation.

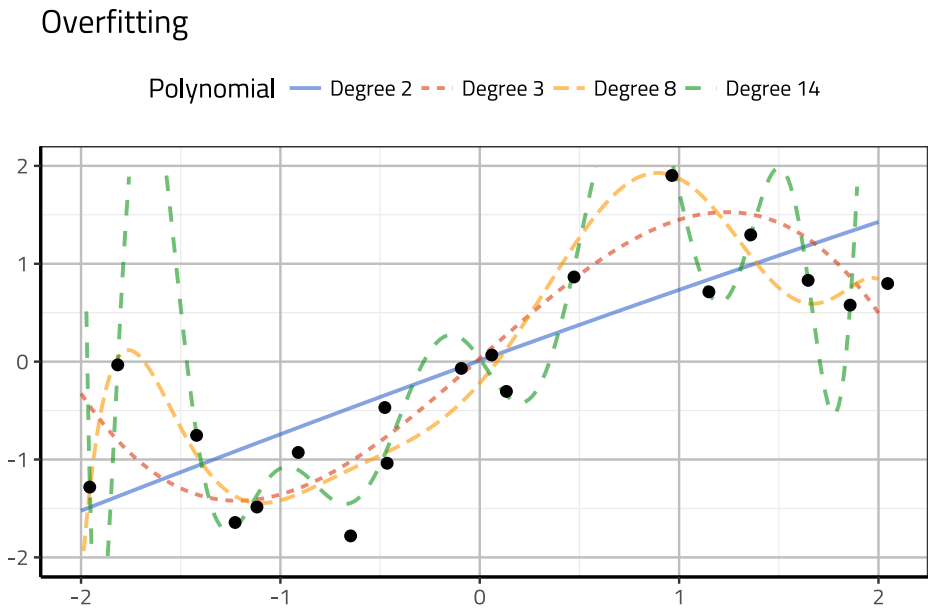


Figure 2.6: Example of overfitting. We have 20 data points (*one is outside the plotting region*) and fit polynomials of various degrees. The underlying model is $y = \sin(x)+x$, with Gaussian noise, the third degree polynomial should provide the most reasonable fit.

2.5.1.1 Other Generalisation Criterions

The notion of indirectly optimising a generalisation criterion is not restricted to predictions. Candés et al. presented the knockoff filter in 2015, where they prove that they can provide bounds on the False Discovery Rate (FDR) in variable selection for the knockoff filter [B+15; WBC17; BCS18]. The basic idea behind the knockoff filter is that we create a rotated version of our variables, then we add these *knockoff* variables to our data set and perform variable selection. If the knockoff variable is selected before the actual variable, in the selection process, then that is substantial evidence for the original to be a false discovery.

I was fortunate enough to attend the Machine Learning Summer School (MLSS) in Kyoto Japan. Candés gave a presentation about the knockoff filter, and he also described *Big-Data* as a new paradigm of doing research. Traditionally in statistics, one performs the following steps:

1. Find hypothesis.

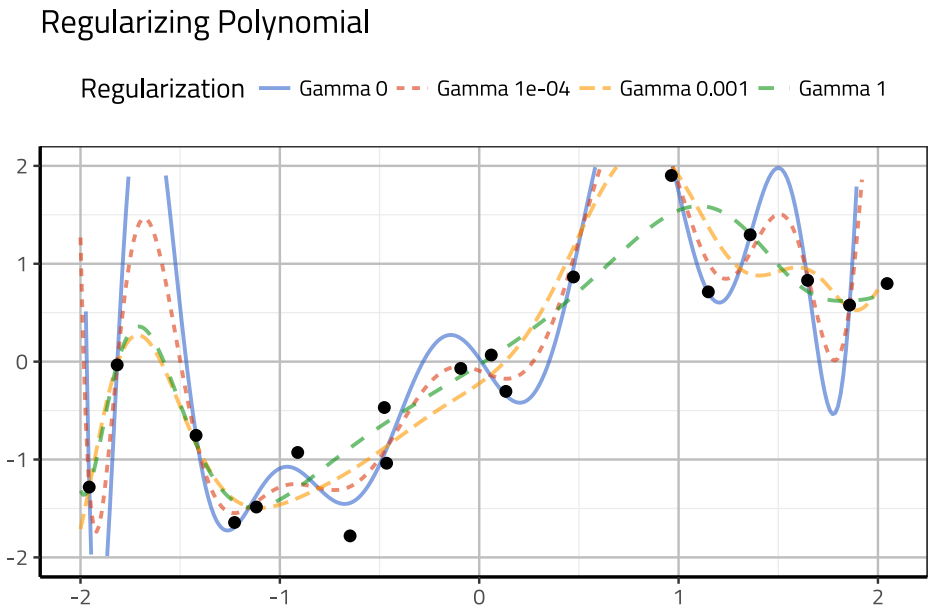


Figure 2.7: Example of ridge regularisation. We have 20 data points (*one is outside the plotting region*) and a polynomial of degree 14, we try various penalty parameters for the ridge penalty, showing how we can effectively provide a simpler model. The underlying model is $y = \sin(x) + x$, with Gaussian noise.

2. Design an experiment to reject or validate the hypothesis.
3. Run the analysis and report findings.

Candés suggests an alternative:

1. Receive an abundance of data.
2. Generate correct hypotheses, and include as few false discoveries as possible.

This approach limits p-hacking/fishing, which is a severe problem in academia. We need tools to control the FDR to do proper research. There are some fantastic applications for this work, one, in particular, is for doing a multivariate Genome Wide Association Study (GWAS). Barber, Foygel, and Candés have shown promise for such applications [BC16].

The following quote is from Fisher, which reflects strong views on the traditional paradigm for statistical analysis:

> *To call in the statistician after the experiment is done may be no more than asking him to perform a post-mortem examination: he may be able to say what the experiment died of.*

2.5.2 Curses & Blessings

In the year 2000 Donoho gave a speech called *High-dimensional data analysis: The curses and blessings of dimensionality* [Don+00]. The talk was held exactly 100 years after Hilbert gave the talk on *Mathematical Problems* at the International Congress of Mathematicians in Paris. The points Donoho makes in this talk regarding the main trends in how statistics and data analysis are changing.

The statistics of the 20th century were built on the assumption that the number of samples exceeded the number of observations, $p < n$, and further asymptotic results were derived in the limit when $n \to \infty$. Most of the results based on these assumptions fail in the case $p \gg n$. Donoho emphasises that the $p \gg n$ case may instead be the typical case and draws examples from genomics and speech analysis.

2.5.2.1 The Curses

Donoho refers to Bellman as having coined the term *curse of dimensionality* [Bel15]. Donoho mentions three classic cases:

1. If we must approximately optimise a function of p variables and the only information we have is that it is Lipschitz continuous, then we need approximately $(1/\epsilon)^p$ evaluations on a grid, to find variables that give us error less than ϵ.
2. If we want to approximate a function of p variables, only knowing it is Lipschitz. Then we need $(1/\epsilon)^p$ evaluations on a grid to obtain uniform approximation error less than ϵ.
3. If we want to integrate a Lipschitz function numerically, we need $(1/\epsilon)^p$ evaluations on a grid to achieve error less than ϵ.

These results are apparent but quite commonly used as general assumptions. Only knowing that the function we are dealing with is Lipschitz, is very generic, and demonstrates the difficulty in relaxing the conditions or assumptions we commonly use. For statistical purposes, Donoho presents the estimation problem:

$$y = f(x_1, x_2, ..., x_p) + \varepsilon. \tag{2.47}$$

If we only assume that f is Lipschitz, and that the errors are i. i. d. Gaussian, how many observations do we need to estimate f? If $\mathcal{F}$ is the class of all functions which are Lipschitz on $[0, 1]^p$, then we have from the asymptotic theory of statistical estimation [IH13]:

$$\sup_{f \in \mathcal{F}} \mathrm{E}(\hat{f} - f(x))^2 \geq C \cdot N^{-2/(2+p)}, \quad n \to \infty, \tag{2.48}$$

where C is a constant. So for a given error, we allow in our estimated function $\hat{f}$, we can get bounds on the number of observations needed, which unfortunately scales exponentially with the number of variables.

2.5.2.2 The Blessings

The first blessing Donoho mentions is the *concentration of measure phenomenon*. Assume that we have a Lipschitz function f on a p-dimensional sphere. If we have P as a uniform measure on the sphere and X is a P-distributed random variable, then:

$$P\{|f(x) - \mathrm{E}f(x)| > t\} \leq C_1 \exp(-C_2 t^2). \tag{2.49}$$

The C constants are independent of f and the dimension p. This means that the Lipshitz functions are nearly constant, and the tails of this distribution behave like the tails of a Gaussian random variable.

The second blessing relates to the existence of asymptotic results, where we let the number of predictors go to infinity. These results have, e.g., been used to derive the extreme value distributions.

The third blessing is concerning why the data is high dimensional. We can have spatial or temporal relationships. The underlying phenomena that we are sampling may be of much lower dimension and are realised in high dimension, because of the way the signals are sampled.

2.6 Other Methods Inspired by the Lasso

There is a plethora of other methods inspired by the Lasso regulariser, or methods that directly incorporate the Lasso. The book *Statistical Learning with Sparsity, the Lasso and Generalisations* from 2015 gives the most recent extensive overview [HTW15].

The most notable similar methods to SDA are the sparse analogs derived from CCA [PTB09; WTH09] and Fisher's discriminant analysis [TJ07]. Another important related method is sparse PCA [JTU03; WTH09]. We can consider LDA as a supervised version of PCA, similarly sparse PCA is the unsupervised analog of SDA. LDA provides us with the discriminant vectors, that we can use to project the data to a lower dimensional representation, where this representation contains all the information necessary for discriminant analysis, under the Gaussian assumptions for the data. The discriminant vectors are also ranked according to the amount of discrimination they provide, similar to the loadings of PCA are ranked for the amount of variation they explain.

Many of these methods are implemented in the Matlab toolbox associated with contribution D. The toolbox includes implementations of elastic net, forward selection, least angle regression, lasso, SDA and sparse-PCA. Contribution D also includes an extended related work section on the currently related available tools.

2.7 What is Statistical Learning?

Why do we need a new name for the field? Why can't we call it Machine Learning, or Statistics? One of the significant differences from traditional statistics is a stronger focus on making prediction models and classifiers. Another critical deviation is the importance of being able to compute the solutions efficiently, and the development of algorithms to do so.

Compared to machine learning, statistical learning usually has a statistical method as a starting point, whereas that is not needed for machine learning in general. This method usually needs to be generalised to a specific problem or case, e.g., like ridge regression, to handle more predictors than observations. The way these classical methods are generalised does not necessarily have theoretical underpinnings in the same way as traditional statistics. An excellent example of that is the elastic net. We can cast the elastic net into the Bayesian framework, to try to understand better the kind of distributional assumptions that can lead to a model like that, but we end up with a prior distribution which is a combination of two known closed form distributions. It is easier to understand these methods from the perspective of the bias-variance tradeoff.

More conservative practitioners of statistics have heavily criticised the development of statistical learning methods and modern machine learning. The epitome of such discussions is the paper *Statistical modelling: The two cultures* by Leo Breiman [Bre+01]. Leo describes the two cultures as the data modelling culture and the algorithm modelling culture. One of the main arguments against the data modelling culture is the way they handle model validation. Model validation is commonly achieved with the goodness of fit tests and residual examination. The algorithm culture, on the other hand, does validation by measuring predictive accuracy.

The discussion following the paper shows how these different opinions are reflected. Brad Efron comments that the 20th century may be described as 100 years of unbiasedness.

In the case where we have an abundance of data, i.e., a lot of observations, I agree with Breiman. Algorithmic methods, which are mainly very good and fast function approximations, are perfect for this task, such as Deep Learning, Random Forest and Xgboost, [Goo+16; Fri01; Bre01]. On the other hand, when we have fewer observations, expert knowledge as priors, which can be imposed into the modelling frameworks, are the best option.

CHAPTER 3

Optimisation

3.1 General Theory

Optimisation is a versatile tool; the dictionary definition is *the action of making the best or most efficient use of a situation or resource*. The dictionary definition is very related to the mathematical definition. From a mathematical point of view, optimisation concerns with problems of finding the best element, from a set, that satisfies our criterion.

In most cases, this is translated to the problem of minimising or maximising a function over some domain, where the domain might be specified with constraints. In general, a mathematical optimisation problem can be defined as:

$$\arg\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) \tag{3.1}$$

$$\text{subject to } g_i(\mathbf{x}) \leq b_i, \quad \forall i \in \{1, 2, ..., m\}, \tag{3.2}$$

where f and g_i are all functions $f : \mathbb{R}^n \to \mathbb{R}$. This general formulation incorporates all optimisation problems[1]. For a new problem, usually, the difficult part is casting it into the correct optimisation problem to solve.

3.1.1 Linear Programming

Throughout the 20th century, people worked on developing techniques for solving these types of problems under certain assumptions on the function f and the constraints g_i. The most common of these is probably *linear programming*. Linear programming makes rigid assumptions, meaning that the function f and the constraints g_i are linear. This means:

$$f(\alpha\mathbf{x}_1 + \beta\mathbf{x}_2) = \alpha f(\mathbf{x}_1) + \beta f(\mathbf{x}_2) \tag{3.3}$$

Under these assumptions, Eq. 3.1 translates to:

$$\arg\min_{\mathbf{x}\in\mathbb{R}^n} \mathbf{w}^T\mathbf{x} \tag{3.4}$$

$$\text{subject to } \mathbf{A}\mathbf{x} \leq \mathbf{b}, \text{ and } \mathbf{x} \geq 0 \tag{3.5}$$

[1] We could of course also have the domain as $\mathbb{C}^n$, but for the practical purposes presented here, we are concerned with reals.

The most commonly used algorithm to solve the problem in Eq. 3.4 is the simplex method by Dantzig [D+55]. The idea is that if the best solution is not in the region defined by the constraints, then it must lie on the corner points of a surface determined by the constraints, which is a polytope.

The algorithm works well, but there exist examples showing that the worst-case time-complexity is exponential. There exist interior point methods [Wri97] which work in polynomial time, but in practice, the simplex method is the fastest.

Generalisations of linear programming include integer programming and mixed integer programming. There we restrict the solution vector, or part of the solution vector to contain integers. This restriction does not simplify the problem; it further complicates it somewhat. Common algorithms to solve these are the branch and bound based methods. The bounds are calculated as linear relaxations, allowing us to reduce the size of the search space.

3.1.2 Quadratic Programming

Another class of problems is the quadratic programming problems. They are similar to Eq. 3.4, except we get an additional quadratic term:

$$\arg\min_{\mathbf{x}\in\mathbb{R}^n} \frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{w}^T\mathbf{x} \tag{3.6}$$

$$\text{subject to } \mathbf{A}\mathbf{x} \leq \mathbf{b}, \text{ and } \mathbf{x} \geq 0, \tag{3.7}$$

where $\mathbf{Q}$ is real symmetric. Quadratics programs can also be solved with interior point methods.

Researchers struggled to generalise optimisation or to categorise it in different ways. Mostly the problems were either linear or nonlinear, where, unfortunately, no general polynomial time approach exists. Later in the 20th century, a generalisation of linear programming appeared, it was convex optimisation.

3.2 Convex Optimisation

Convex optimisation is about solving problems similar to that of a linear program. Instead of imposing the rigid assumptions that the functions are linear, we relax the condition, requiring that the functions are convex:

$$f(\alpha\mathbf{x}_1 + \beta\mathbf{x}_2) \leq \alpha f(\mathbf{x}_1) + \beta f(\mathbf{x}_2), \quad \alpha + \beta = 1,\ \alpha, \beta \geq 0, \forall \mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^n. \tag{3.8}$$

We can see that linear programming is a special case.

It was not until 1983 that a formal argument was presented [NYD83] for showing that convex optimisation problems were easier to solve than general nonlinear optimisation problems. Further in 1993 Rockafeller said in his review paper [Roc93]:

> In fact, the great watershed in optimisation isn't between linearity and nonlinearity, but convexity and non-convexity.

These developments lead to the creation of the field of convex optimisation. Often we can also relax certain problems to be convex, allowing us to find an approximate solution efficiently. Stephen Boyd, who wrote the book *Convex Optimization* [BV04] has the following to say:

> The challenge, and art, in using convex optimisation is in recognising and formulating the problem. Once this formulation is done, solving the problem is, like least-squares or linear programming, (almost) technology.

So the hardest part of the problem is representing it properly as a convex optimisation problem. Convex optimisation includes multiple specialised algorithms, way too many to cover in this thesis. We will focus on approaches which are related to the algorithms developed in contribution A.

Remember that the main problem that we are solving is in Eq. 2.34. For a given score vector $\boldsymbol{\theta}$ we have a convex function which we are minimising. This is easy to see since a linear combination of convex functions is also convex. The algorithms we are focusing on correspond to the *update* $\boldsymbol{\beta}$ part in Algorithm 1. We will consider a couple of algorithms for this purpose.

3.3 Proximal Methods

For extended work on proximal methods, we refer the reader to [P+14]. The work presented here is a summary of the work presented in contribution A, that work is very influenced by the ISTA and FISTA methods by Beck and Teboulle [BT09].

The word proximal relates to the word proximity. Proximal gradient-based methods are based on the proximal operator, which revolves around balancing the objectives of minimising a function and remaining in the proximity of a given point. So for a given function $f : \mathbb{R}^p \to \mathbb{R}$, we can define the *proximal operator* $\text{prox}_f : \mathbb{R}^p \to \mathbb{R}^p$ as:

$$\underset{f}{\text{prox}}(\mathbf{y}) = \underset{\mathbf{x}\in\mathbb{R}^p}{\arg\min} \left\{ f(\mathbf{x}) + \frac{1}{2}\|\mathbf{x} - \mathbf{y}\|^2 \right\}, \tag{3.9}$$

The proximal operator is handy for minimising functions which are partly differentiable. We can then minimise the non-smooth part while retaining close to the gradient of the smooth portion. Using the proximal operator allows us to take an optimisation problem, and solve the *easy* part independently, and cast it into a new optimisation problem, where the primary focus is on the non-differentiable function. Now let's assume that we have a problem that we can represent as:

$$\min_{x\in\mathbb{R}^p} f(\mathbf{x}) + g(\mathbf{x}) \tag{3.10}$$

where f is smooth everywhere, and g is potentially not differentiable. For a proximal gradient-based approach we take a step in the direction of $-\nabla f$ and evaluate the

proximal operator of g. If we have iterate $\mathbf{x}^t$, then the next iterate is produced as follows:

$$\boldsymbol{x}^{t+1} = \operatorname*{prox}_{\alpha_t g}(\boldsymbol{x}^t - \alpha_t \nabla f(\boldsymbol{x}^t)) \tag{3.11}$$

$$= \operatorname*{arg\,min}_{\boldsymbol{x}\in\mathbb{R}^p} \left\{ \alpha_t g(\boldsymbol{x}) + \frac{1}{2}\|\boldsymbol{x} - (\boldsymbol{x}^t - \alpha_t \nabla f(\boldsymbol{x}^t))\|^2 \right\}, \tag{3.12}$$

where α_t controls the step length, or the learning rate. Now if we consider the problem in Eq. 2.34, with a fixed $\boldsymbol{\theta}^t$, we can split it up into the two functions f and g, where

$$f(\boldsymbol{\beta}) = ||\mathbf{Y}\boldsymbol{\theta}^t - \mathbf{X}\boldsymbol{\beta}||_2^2 + \gamma \boldsymbol{\beta}^T \Omega \boldsymbol{\beta} \tag{3.13}$$

$$g(\boldsymbol{\beta}) = \lambda ||\boldsymbol{\beta}||_1 \tag{3.14}$$

g is differentiable everywhere, except at zero. Now we can represent f a little more conveniently as:

$$f(\boldsymbol{\beta}) = \frac{1}{2}\boldsymbol{\beta}^T \mathbf{A}\boldsymbol{\beta} + \mathbf{d}^T\boldsymbol{\beta}, \tag{3.15}$$

where $\mathbf{A} = 2(\mathbf{X}^T\mathbf{X} + \gamma\Omega)$ and $\mathbf{d} = -2\mathbf{X}^T\mathbf{Y}\boldsymbol{\theta}^{t+1}$. Now the gradient of f is:

$$\nabla f(\boldsymbol{\beta}) = \mathbf{A}\boldsymbol{\beta} + \mathbf{d}. \tag{3.16}$$

The proximal operator of g, i.e. of the scaled l_1-norm is a solved optimisation problem, so we have:

$$\operatorname*{prox}_{\lambda||\bullet||_1}(\mathbf{y}) = \operatorname{sign}(\mathbf{y}) \max\{|\mathbf{y}| - \lambda\mathbf{e}, \mathbf{0}\} =: S_\lambda(\mathbf{y}); \tag{3.17}$$

where S_λ is commonly referred to as the *soft-thresholding operator*. The sign and max functions above are defined element-wise and $\mathbf{e}$ is a vector of all ones.

Now we can create a sequence of iterates for $\boldsymbol{\beta}$ as

$$\boldsymbol{\beta}^{t+1} = \operatorname{sign}(\mathbf{p}^t) \max\{|\mathbf{p}^t| - \lambda\alpha_t\mathbf{e}, \mathbf{0}\}, \tag{3.18}$$

$$\text{where} \quad \mathbf{p}^t = \boldsymbol{\beta}^t - \alpha_t \nabla f(\beta^t) = \boldsymbol{\beta}^t - \alpha_t(\mathbf{A}\boldsymbol{\beta}^t + \mathbf{d}). \tag{3.19}$$

The rate of convergence is $\mathcal{O}(1/t)$. We refer the reader to contribution A for a more extended treatment and guidelines for selection of the step-size parameter.

3.3.1 Acceleration

We can extend the results presented in Eq. 3.18 with acceleration, which was pioneered by the work of Nesterov [Nes83; Nes05; Nes13]. Detailed arguments for why such acceleration works can be found in [AO14; BLS15; FB15; LRP16; OC15; SBC14;

Tse08]. The main idea is to use two consecutive iterates to extrapolate the next solution. The new sequence of iterates now becomes:

$$\mathbf{y}^{t+1} = \mathbf{x}^t + \omega_t(\mathbf{x}^t - \mathbf{x}^{t-1}) \tag{3.20}$$

$$\mathbf{x}^{t+1} = \underset{\alpha g}{\text{prox}}(\mathbf{y}^{t+1} - \alpha \nabla f(\mathbf{y}^{t+1}). \tag{3.21}$$

$\omega_t \in [0, 1)$ is the extrapolation parameter, where we set it as $t/(t+3)$. With acceleration we have rate of convergence in function value as $\mathcal{O}(1/t^2)$. Empirical evidence in contribution A also suggests that this is the fastest approach for solving SDA. Further convergence results and more extended work can be found in contribution A.

3.4 Alternating Direction Method of Multipliers

Alternating Direction Method of Multipliers (ADMM) solves problems of the form:

$$\min_{\mathbf{x}\in\mathbb{R}^p, \mathbf{y}\in\mathbb{R}^m} \{f(\mathbf{x}) + g(\mathbf{y}) : \boldsymbol{A}\mathbf{x} + \boldsymbol{B}\mathbf{y} = \boldsymbol{c}\}, \tag{3.22}$$

via an approximate dual gradient accent [Boy+11]. We can use the same decomposition as for the proximal gradient based method, so g represents the Lasso term in SDA. Then we can use an equality constraint for $\mathbf{x}$ and $\mathbf{y}$. The problem then becomes in ADMM form:

$$\min_{\mathbf{x},\mathbf{y}\in\mathbb{R}^p} \left\{ \frac{1}{2}\mathbf{x}^T \boldsymbol{A}\mathbf{x} - \mathbf{x}^T \boldsymbol{d} + \lambda\|\mathbf{y}\|_1 : \mathbf{x} - \mathbf{y} = \mathbf{0} \right\}. \tag{3.23}$$

To generate the iterates, we perform dual gradient ascent steps, for that we need the augmented Lagrangian:

$$L_\mu(\mathbf{X}, \mathbf{y}, \mathbf{z}) = \frac{1}{2}\mathbf{x}^T \boldsymbol{A}\mathbf{x} - \mathbf{x}^T \boldsymbol{d} + \lambda\|\mathbf{y}\|_1 + \mathbf{z}^T(\mathbf{x} - \mathbf{y}) + \frac{\mu}{2}||\mathbf{x} - \mathbf{y}||_2^2 \tag{3.24}$$

$\mu > 0$ is a penalty parameter controlling the emphasis on enforcing the feasibility of the primal iterates $\mathbf{x}$ and $\mathbf{y}$, i.e. enforcing that the constraint is satisified. We minimise the augmented Lagrangian by alternating between minimising with regards to $\mathbf{x}$ and $\mathbf{y}$. We then use the approximate gradient to update the dual variable $\mathbf{z}$ by a dual ascent step.

To update $\mathbf{x}$, given $(\mathbf{x}^t, \mathbf{y}^t, \mathbf{z}^t)$, we need to solve the following optimization problem:

$$\mathbf{x}^{t+1} = \underset{\mathbf{x}\in\mathbb{R}^p}{\arg\min}\, L_\mu(\mathbf{x}, \mathbf{y}^t, \mathbf{z}^t) = \underset{\mathbf{x}\in\mathbb{R}^p}{\arg\min}\, \frac{1}{2}\mathbf{x}^T(\mu\boldsymbol{I} + \boldsymbol{A})\mathbf{x} - \mathbf{x}^T(\mathbf{d} + \mu\mathbf{y}^t - \mathbf{z}^t). \tag{3.25}$$

We can minimise this by solving for when the gradient is zero, which gives us the following equation

$$(\mu\mathbf{I} + \mathbf{A})\mathbf{x}^{t+1} = \mathbf{d} + \mu\mathbf{y}^t - \mathbf{z}^t. \tag{3.26}$$

Since the matrix on the left hand side stays fixed throughout the iterations, we can exploit that to reduce computation. We refer the reader to the paper (contribution A) for more details on reducing the amount of computation needed.

Next we need to update $\mathbf{y}$. Then we need to solve the following optimisation problem:

$$\mathbf{y}^{t+1} = \underset{\mathbf{y}\in\mathbb{R}^p}{\arg\min}\, L_\mu(\mathbf{x}^{t+1}, \mathbf{y}, \mathbf{z}^t) = \underset{\mathbf{y}}{\arg\min}\, \lambda\|\mathbf{y}\|_1 + \frac{\mu}{2}\|\mathbf{y} - \mathbf{x}^{t+1} - \mathbf{z}^t/\mu\|^2. \tag{3.27}$$

This is precisely the proximal operator of the scaled l_1-norm applied to the vector $\mathbf{x}^{t+1} + \mathbf{z}^t/\mu$. So we can use the soft-thresholding operator to update $\mathbf{y}$:

$$\mathbf{y}^{t+1} = S_\lambda(\mathbf{x}^{t+1} + \mathbf{z}^t/\mu). \tag{3.28}$$

Finally, the dual variable is updated using the approximate dual ascent step:

$$\mathbf{z}^{t+1} = \mathbf{z}^t + \mu(\mathbf{x}^{t+1} - \mathbf{y}^{t+1}). \tag{3.29}$$

3.5 Summary

In this chapter, we have presented the essential step needed for deriving the algorithms developed in contribution A. Much more extended results can be found in that manuscript, including run-time analysis, empirical comparisons, and proofs of convergence.

These algorithms are implemented in the R-package `accSDA`, along with other variants, such as sparse zero-variance discriminant analysis [AH16].

CHAPTER 4
Ordinal Labels and Unlabeled Data

In this chapter we summarise the methods and algorithms from contributions B, D and F. These advances regard the cases when the labels have an inherent ordinal relation, compared to a nominal one, or if some of the data is unlabeled.

4.1 Ordinal Labels for $p \gg n$ Problems

Ordinal labels are all around us; we may not be entirely aware of it. Every time some grading is involved, the label is ordinal. Most notorious of such modern cases are probably online user reviews. In this section, we shall inspect further what ordinal labels are, and how we can tackle them in the $p \gg n$ setting. We rely heavily on other people's opinion and online reviews for making decisions [GY08]. In chapter 7 we will inspect how ordinal approaches to sparse classification can help us manage the increasing amount of information contained in user reviews.

4.1.1 What are ordinal labels?

Ordinal labels are used as a way to rank observations, subjectively, or according to a predefined protocol. We assign these ranks because the direct response that we would wish to observe is just not available. The ordinal label thus becomes a proxy for a phenomenon we would like to measure. We can compare observations based on ordinal labels, e.g., is one observation better than the other? But we cannot quantify the extent of this difference. The ordinal labels have the discreteness property from regular nominal labels, and the ordinal property from numeric responses, making them fall between the two groups.

A classical example of ordinal labels are grades. We can think of grades as a proxy for estimating the amount of effort a student has put into a subject. Of course, other factors play a part here, e.g., general level of intelligence, but let's assume that the latent variable we are trying to estimate, is, in fact, the number of hours the student has put in. There is no way of knowing that the relationship is linear, it might just as well be exponential, meaning that for each increase in grade we need exponentially more effort. This is commonly reflected in grades, where the distribution is approximately Gaussian, sometimes the grades are intentionally distributed on a bell curve. This type of data is depicted in Fig. 4.1.

Imagine that if we have data on the students, and we want to predict their effort from this data, but the only available response is the grades, which is a proxy for the

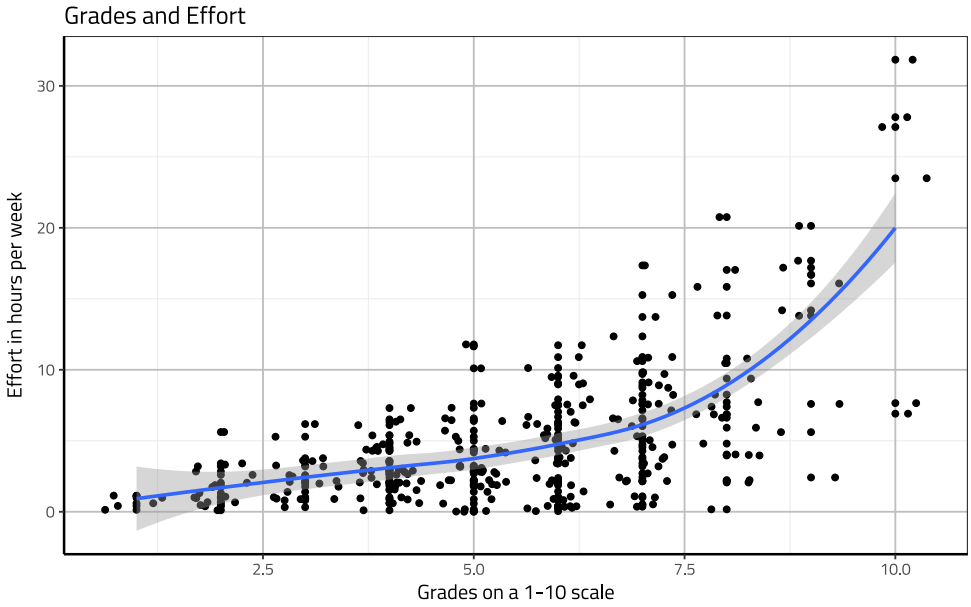


Figure 4.1: This plot shows synthetic grades (Noise was added to the grade for the visualisation) plotted against effort in hours per week. This data shows example of nonlinear relationship between the ordinal label proxy variable, and the true signal that we are after. Note that variation also increases with higher grades..

same signal. The effort can have various problems, e.g. there is a nonlinear relationship between the grades and effort, and there might be overdispersion, meaning that the variation in effort increases with higher grades.

This means that if we approach this by doing traditional regression on the labels, we cannot conclude which variables are truly most affecting the effort. The assumptions for standard linear regression are essentially broken. This means that we need to resort to other approaches, specifically for ordinal labels, a good summary can be found in [AK97].

4.1.2 From a Classification Point of View

Another common approach to ordinal labels is to ignore the ordinality aspect and use a simple classifier. For motivating the effort of doing something else, other than ignoring the ordinality, we want the ordinal classifier to have some properties, which make it superior. Ideally:

1. The ordinal classifier should be simple, i.e., need fewer parameters.
2. It should also be easier to interpret, specifically in alignment with the target problem.
3. It should a have a lower *off-by-one* error, meaning that if we classify wrong, it is more likely to be the label one above or one below. The classifier should capture the ordinal aspect of the data.

There is a generic method to achieve these goals. Let us inspect it further.

4.1.2.1 The data replication method

The data replication is a general tool, to adapt any simple classifier to ordinal labels [CC07; CCC05]. The trick is hidden in the name. We create replicas of the data into the dataset. These replicas are created with additional binary variables, which code which class they belong to, we do not naively create replicas, there is more to it.

The motivation for the data replication methods is that the classification boundary, should be identical between classes (See Fig. 4.2). Identical boundaries between classes can be extended to non-linear boundaries, but here we are concerned with linear boundaries. To achieve these same boundaries means that for each boundary we define a binary classification problem, choosing the number of adjacent classes to use on each side. We do this for all the boundaries, and then solve all of the binary classification problems together, to find the jointly best separating hyperplane. First, we describe the data replication method, and then we demonstrate how it can be adapted to a sparse setting. The approach is somewhat notationally cumbersome, so we refer the reader to table 4.1.2.1 for lookup. Our way of creating the replicated data matrix is more constructive than that of Cardoso et al.

Table 4.1: Notation used for developing the data replication method.

Symbol	Description	Symbol	Description
$\boldsymbol{\beta}_k$	Discriminant vector k	$\hat{\boldsymbol{\Omega}}$	Ordinal regularisation matrix
$\boldsymbol{\theta}_k$	Scoring vector k	$\mathbf{e}_i$	$1 \times (K-1)$ i-th unit vector
$\mathbf{Y}$	Label indicator matrix	$\mathbf{E}_{k,i}$	n_k rows of $\mathbf{e}_i$
$\mathbf{X}$	Feature matrix	b_i	Bias i, where $i \in \{1, ..., K-1\}$
K	Number of classes	s	Width of classif. problem
n	Number of samples	$\mathbf{X}_{\text{Ord}}$	Replicated data matrix
p	Number of variables	$\mathbf{Y}_{\text{Ord}}$	Replicated label vector
$\boldsymbol{\Omega}$	Regularization matrix	$\mathbf{X}_{\text{Ord},i}$	Data matrix for boundary i
$\|\| \cdot \|\|_1$	1-norm	$\mathbf{Y}_{\text{Ord},i}$	Labels for boundary i
λ_i	Regularisation parameters	$\mathbf{x}^{(k)}$	$n_k \times p$ class k data matrix
n_k	Samples in class k	$\mathbf{1}_k$	$n_k \times 1$ vector with only ones
$\boldsymbol{\beta}_{\text{Ord}}$	Ordinal discr. vector	$\mathbf{2}_k$	$n_k \times 1$ vector with only twos

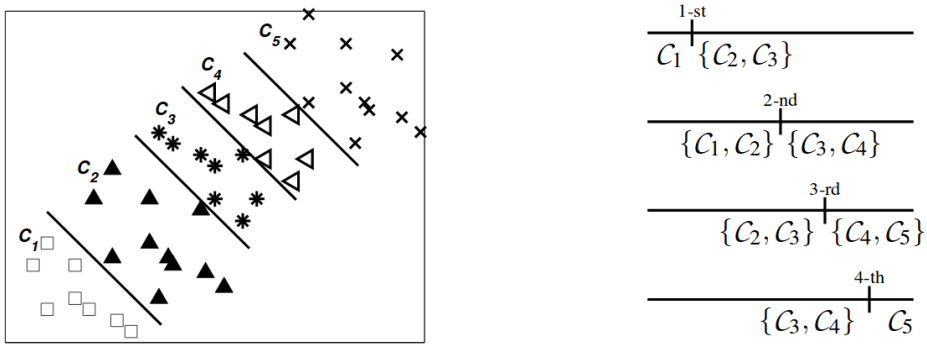


Figure 4.2: Visualization showing the idea behind the data replication method. We seek linear boundaries between classes, where the separating hyperplane is always the same up to the bias. We achieve this by creating a binary classification problem for each hyperplane, (see on the right). We can control the number of adjacent classes we wish to include on each side for the binary problems. Fewer means that we need to replicate less data. This visualisation is taken from [CC07].

For each of the $K-1$ classification boundaries, we define a binary classification problem. If the boundary is the one between classes i and $i+1$, then the binary labels correspond to labels i and lower, and $i+1$ and higher. The maximum number of classes we use on each side of the boundary is called s, which is an integer specified by the user. So for the boundary between classes i and $i+1$ we label classes $i-s+1, i-s+2, ..., i$ as belonging to class 1 and the classes $i+1, i+2, ..., i+s$ labeled as belonging to class 2. Then we append the vector $\mathbf{e}_i$, which is of length $K-1$, to the samples, where $\mathbf{e}_i$ is a vector of all zeroes, except it takes the value 1 in position i. All these new samples are added to the new data matrix. Class k has n_k samples and we define $\mathbf{x}^k$ as the $n_k \times p$ data matrix, only containing samples from class k. We also define the $n_k \times (K-1)$ matrix $\mathbf{E}_{k,i}$, which is a matrix of all zeroes, except the i-th column is a vector of all ones.

$$\mathbf{X}_{\text{Ord},i} := \begin{bmatrix} \mathbf{x}^{(i-s+1)} & \mathbf{E}_{(i-s+1),i} \\ \vdots & \vdots \\ \mathbf{x}^{(i)} & \mathbf{E}_{i,i} \\ \mathbf{x}^{(i+1)} & \mathbf{E}_{i+1,i} \\ \vdots & \vdots \\ \mathbf{x}^{(i+s)} & \mathbf{E}_{i+s,i} \end{bmatrix} \tag{4.1}$$

$$\mathbf{Y}_{\text{Ord},i} := \begin{bmatrix} \mathbf{1}_{1-s+1} \\ \vdots \\ \mathbf{1}_i \\ \mathbf{2}_{i+1} \\ \vdots \\ \mathbf{2}_{i+s} \end{bmatrix} \tag{4.2}$$

The data matrix $\mathbf{X}_{\text{Ord},i}$, corresponding only to the data needed for the boundary between class i and $i+1$ can be seen in Eq. 4.1 and the corresponding label vector in Equation 4.2.

$$\mathbf{X}_{\text{Ord}} := \begin{bmatrix} \mathbf{X}_{\text{Ord},1} \\ \mathbf{X}_{\text{Ord},2} \\ \vdots \\ \mathbf{X}_{\text{Ord},K-1} \end{bmatrix} \mathbf{Y}_{\text{Ord}} := \begin{bmatrix} \mathbf{Y}_{\text{Ord},1} \\ \mathbf{Y}_{\text{Ord},2} \\ \vdots \\ \mathbf{Y}_{\text{Ord},K-1} \end{bmatrix} \tag{4.3}$$

The final data matrix is constructed from the matrices corresponding to the binary classification problems as shown in Eq. 4.3.

4.1.3 Adapting the Data Replication Method to SDA

Now we would wish to plug the matrix $\mathbf{X}_{\text{Ord}}$ and the labels $\mathbf{Y}_{\text{Ord}}$ into the SOS problem in Eq. 2.34. We call the new discriminant vector that comes from this problem $\boldsymbol{\beta}_{\text{Ord}}$:

$$\boldsymbol{\beta}_{\text{Ord}} := \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \\ b_1 \\ b_2 \\ \vdots \\ b_{K-1} \end{bmatrix}, \tag{4.4}$$

which is composed of a traditional discriminant vector, corresponding to the first p elements, and then $K-1$ biases, denoted b_i, for $i \in \{1, 2, ..., K-1\}$. We want to regularise the first p parameters like in nominal SDA, but the parameters corresponding to the biases, should not be regularised. In Equation 2.34 we use instead a $(p+K-1) \times (p+K-1)$ regularisation matrix $\hat{\boldsymbol{\Omega}}$, where the top left-most block corresponds to the $p \times p$ $\boldsymbol{\Omega}$ regularisation matrix for the original variables and the rest is zero, such that there is no regularisation for the extra parameters corresponding to the biases:

$$\hat{\boldsymbol{\Omega}} := \begin{bmatrix} \boldsymbol{\Omega} & 0 \\ 0 & 0 \end{bmatrix} \tag{4.5}$$

In a similar manner for the l_1-norm term, we only calculate the l_1-norm of the first p parameters in $\boldsymbol{\beta}_{\text{Ord}}$, not regularising the parameters corresponding to biases. These details are masked from the user in the implementation, thus the user can optionally specify a $p \times p$ $\boldsymbol{\Omega}$ regularisation matrix. This makes the usage almost identical to that of nominal SDA. The main difference from the usage of nominal SDA is that the user cannot specify the number of discriminant vectors in the output, ordinal SDA only generates one.

After doing the data transformation, and redefining the $\boldsymbol{\Omega}$ regularisation matrix, the only change we need to do in the algorithms from contribution A, is to only do the soft-thresholding update for the first p parameters in $\boldsymbol{\beta}_{\text{Ord}}$. This amounts to only using the first p parameters of $\boldsymbol{\beta}_{\text{Ord}}$ to calculate the l_1-norm in Equation 2.34.

$$\begin{aligned} \underset{\boldsymbol{\theta}\in\mathbb{R}^2,\, \boldsymbol{\beta}_{\text{Ord}}\in\mathbb{R}^{p+K-1}}{\arg\min} \quad & \|\mathbf{Y}_{\text{Ord}}\boldsymbol{\theta} - \mathbf{X}_{\text{Ord}}\boldsymbol{\beta}_{\text{Ord}}\|_2^2 \\ & +\lambda_2 \boldsymbol{\beta}_{\text{Ord}}^T \hat{\boldsymbol{\Omega}} \boldsymbol{\beta}_{\text{Ord}} + \lambda_1 \textstyle\sum_{i=1}^{p} |\beta_i| \end{aligned}$$

$$\text{s.t. } \frac{1}{n}\boldsymbol{\theta}^T \mathbf{Y}_{\text{Ord}}^T \mathbf{Y}_{\text{Ord}} \boldsymbol{\theta} = 1. \tag{4.6}$$

The original SOS problem 2.34 is reformulated with regards to the new notation and change in the problem in Eq. 4.6. Note that we no longer need the orthogonality constraint, since we only have one discriminant vector.

4.1.4 Predictions

The biases in the ordinal discriminant vector are ordered, so they define intervals on the real line. For doing predictions on an unseen sample x, assuming it is a $1 \times p$ row vector, we first normalise it appropriately, and call the normalised sample x_{n}. Afterwards we calculate the scalar value $\tilde{x} = x_{\text{n}} \cdot \hat{\boldsymbol{\beta}}_p$, where $\hat{\boldsymbol{\beta}}_p$ is the $p \times 1$ vector corresponding to the first p values from the ordinal discriminant vector. Then we find the lowest bias b_i, such that $\tilde{x} < b_i$, that means that we predict that the new observation belongs in class i.

4.1.5 Synthetic Data

Figure 4.3 shows how the classifier performs on a synthetic two-dimensional dataset with 15 classes. The data is generated along a sine curve, where most of the variation for each class is along the vertical axis. The method correctly learns to classify the data, and one can visually examine that the decision boundaries are parallel to each other.

In Fig. 4.4 we use the traditional SDA method, but only request a single discriminant vector. The discriminant vector does not align properly with the class boundaries and fails. When we add the second discriminant vector, as can be seen in Fig.4.5, the SDA method succeeds.

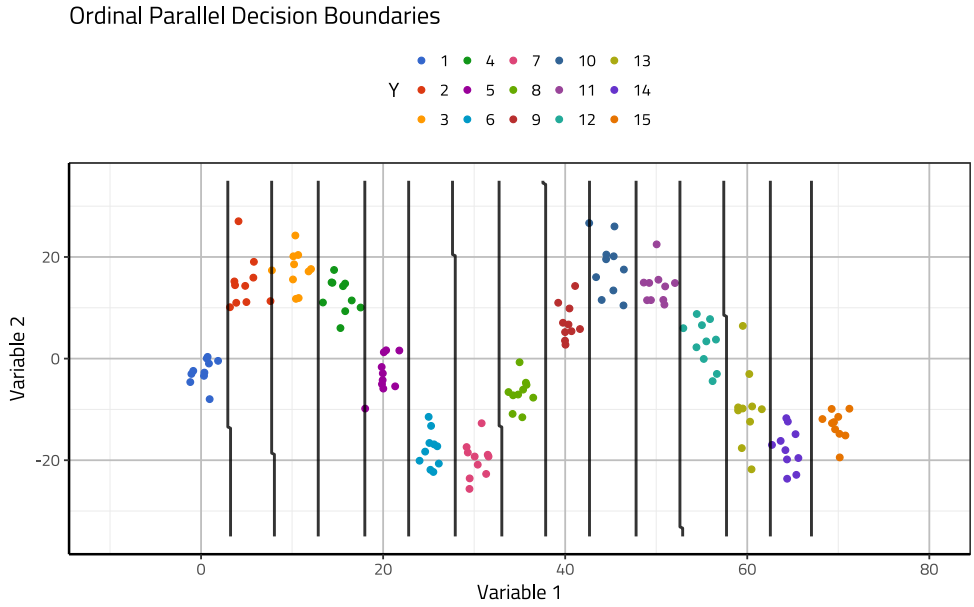


Figure 4.3: Synthetic data to show the discriminative potential of the ordinal classifier and how the decision boundaries are parallel.

4.2 Semi-Supervised Learning

Semi-supervised learning is the concept of learning with both labeled and unlabeled data. This scenario commonly occurs, there are two main practical reasons:

1. It is expensive to create labels, e.g., having Medical doctors evaluate patients or Radiologists labelling images.

2. Model has been deployed and running for some time, and we want to retrain with new data, but omit a lengthy labelling process.

The latter case commonly includes a phenomena called *concept drift* [Tsy04]. Concept drift means, that with time, the mean of the distribution representing the different classes shifts. Concept drift commonly occurs with spam e-mail, where the people generating the spam make it based on new trends, and to pass through the current spam-filters [WIP13].

There are ways to take a classifier and make it semi-supervised generically. One such classical approach is the Yarowsky algorithm [Yar95; Abn04]. Other approaches are based on the Expectation-Maximization (EM)-algorithm [Nig+00]. The EM approach consists of training the algorithm on labeled data, then predicting on the

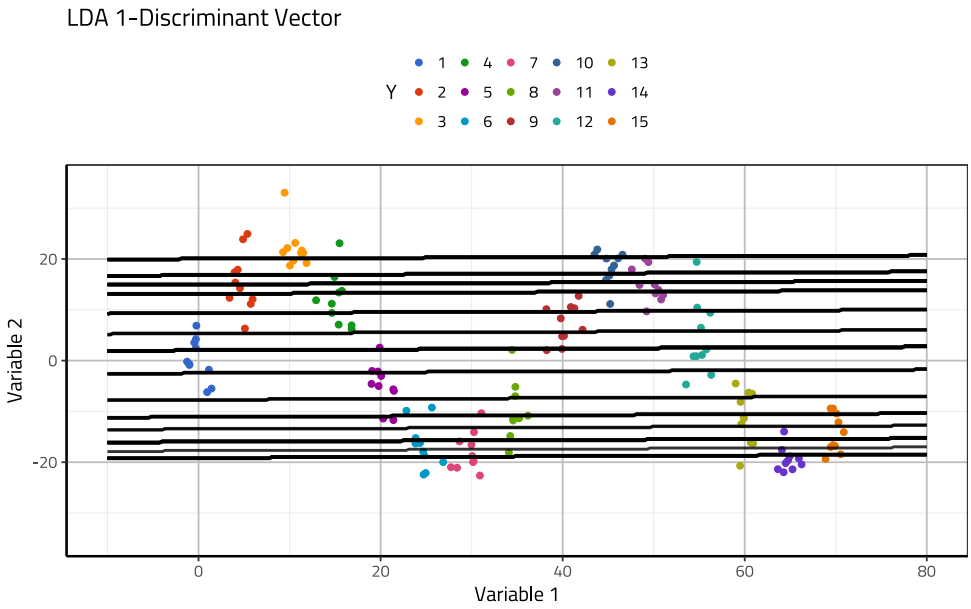


Figure 4.4: Synthetic data, here we see how the SDA method fails if we only use a single discriminant vector.

unlabeled data. The assigned labels are used as soft or hard labels to retrain the classifier and make predictions again. This procedure is continued until convergence. An implementation of this approach can be found in the R-package RSSL [KL15b; Kri16]. The approach described above is a kind of self-learning approach, let the algorithm teach itself how to use the unlabeled data, this approach has been reinvented on numerous occasions.

Usually, the incorporation of unlabeled data is based on some assumptions. Common assumptions are that low-density regions separate the classes, a manifold assumption, assuming that labels have the same label as the closest neighbours or merely a clustering assumption, where the classes are assumed to form clusters. Under certain assumptions on the data and models, we can prove to what extent unlabeled data helps [BLP08; SNZ09].

We have to be careful. Sometimes the labels do not align with underlying cluster structure, see Fig. 4.6. From a Bayesian perspective, we can often view our model as a posterior distribution over the parameters

$$P(\boldsymbol{\theta}|\mathbf{x}) \propto P(\boldsymbol{\theta})P(\mathbf{x}|\boldsymbol{\theta}) \tag{4.7}$$

where the first factor on the right-hand side is the prior distribution. Usually, we can think of the unsupervised part of the model as a prior. Now to find the parameters for

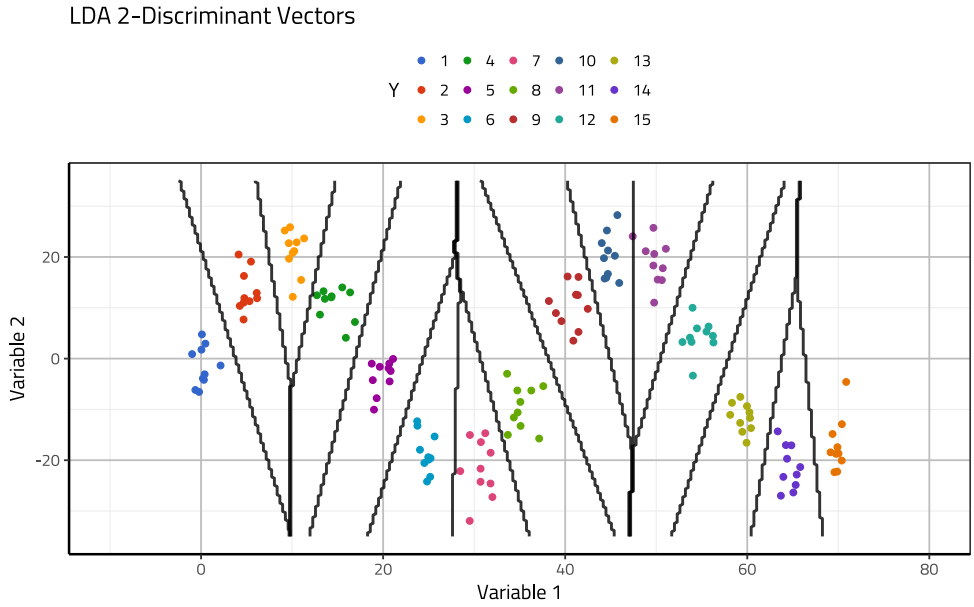


Figure 4.5: Synthetic data, here we see how the SDA method succeeds with two discriminant vectors.

solving the problem, we need to balance the two objectives, adhere to the prior, or the labeled data. In the peculiar case presented in Fig. 4.6, the unlabeled data will drive the decision boundary to separate the two clusters, and hinder performance. Too much unlabeled data is bad in that case, and we are most likely better off without it. There are ways to incorporate unlabeled data, which ensures that the decision boundary does not get worse than when we only have the labeled data [KL15b; KL14].

4.2.1 Semi-supervised regulariser for SDA

The following summarises the technicalities in contribution B. A natural way to incorporate prior information, into a frequentist model, is through regularisation. We demonstrate how we can integrate a local consistency based assumption into a regulariser for SDA.

We propose a regularisation term, similar to one that had prior been proposed for LDA, [CHH07]. The main contribution of Cai et al. [CHH07] is the construction of a Tikhonov regularisation term which borrows ideas from spectral dimensionality reduction and spectral clustering [BN01; HN03; N+01], where they construct a k-nearest neighbor graph on labeled and unlabeled data in feature space. Assuming

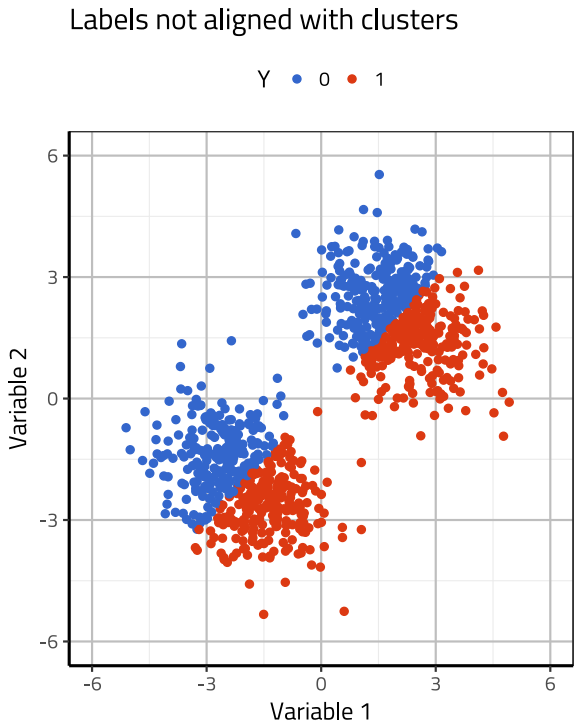


Figure 4.6: Classical counter example to show why we need to be careful with the assumptions in semi-supervised learning. The true labels do not align with the underlying clustering of the data.

that neighbors are more likely to have the same label is a type of manifold assumption initially introduced by Zhu et al. [Z+03] in their label propagation method.

We enforce a local consistency assumption [Zho+03], where unlabeled data that is close in the original feature space should be close after being projected by the discriminant vector. Local consistency is also a manifold assumption, where we seek to project manifolds embedded in the feature-space to a lower-dimensional representation [BN04; SNB05].

We refer to a graph as $\mathcal{G}$, $\boldsymbol{A}$ is the adjacency matrix, $\boldsymbol{D}$ is the degree matrix and $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}$ is the graph Laplacian.

We need to construct the matrix $\boldsymbol{\Omega}$ from the second term in Eq. 2.34 using unlabeled data. We begin by assuming that we have a dataset D, which can be split into a labeled part, D_1, and unlabeled part, D_2, where we have n_1 labeled samples and

n_2 unlabeled samples and $n = n_1 + n_2$.

$$D_1 = \{(c_{k_1}, \boldsymbol{x}_1), (c_{k_2}, \boldsymbol{x}_2), \ldots, (c_{k_{n_1}}, \boldsymbol{x}_{n_1})\} \tag{4.8}$$

$$D_2 = \{\boldsymbol{x}_{n_1+1}, \boldsymbol{x}_{n_1+2}, \ldots, \boldsymbol{x}_{n_1+n_2}\} \tag{4.9}$$

Now we ignore the labels in D_1 and construct a graph on all the data points. We want points that are near each other in the feature space to remain close after the projection, so we construct the graph based on the proximity of data points. We explore two ways to build the graph, what we need for the method from this construction is the graph Laplacian.

First, we consider the weighted undirected graph defined by a Gaussian kernel, where the edge weight between observations $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in the adjacency matrix A is:

$$\boldsymbol{A}_{ij} = \exp(-\gamma||\boldsymbol{x}_i - \boldsymbol{x}_j||_2^2).$$

For a suitable choice of γ, the closest points get a weight close to 1 and then it decays exponentially the further we go away. Now we can construct the semi-supervised regularisation term as:

$$\sum_{ij}(\boldsymbol{x}_i\boldsymbol{\beta} - \boldsymbol{x}_j\boldsymbol{\beta})\boldsymbol{A}_{ij} = 2\boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{L}\boldsymbol{X}\boldsymbol{\beta}.$$

This has been simplified on the right hand side using matrix notation, where L is the graph Laplacian and $\boldsymbol{X}$ is the data matrix containing both labeled and unlabeled feature vectors. So our semi-supervised regularisation matrix $\boldsymbol{\Omega}$ can be defined as $2\boldsymbol{X}^T\boldsymbol{L}\boldsymbol{X}$.

The second graph we consider for the data is a k-nearest neighbour graph. The only difference compared to the approach with the Gaussian kernel, is how we construct the adjacency matrix A, which we initialise as the $n \times n$ zero matrix. For a given data point $\boldsymbol{x}_i$ we find its k nearest neighbours $\boldsymbol{x}_{i_1}, \ldots, \boldsymbol{x}_{i_k}$ and assign the value 1 to $\boldsymbol{A}_{ij}$ and $\boldsymbol{A}_{ji}$ if $\boldsymbol{x}_j$ is one of the k neighbours. We assign the value 1 to $\boldsymbol{A}_{ij}$ and $\boldsymbol{A}_{ji}$ to make sure that the graph is undirected.

Although the way we define the regularisation terms only lies in the adjacency matrix, the main differences are in the way we compute them. For the k-nearest neighbour graph we might want to consider using approximate k-nearest neighbour if 2^p is greater or on the scale of n, where p is the number of features. For the adjacency matrix created with the Gaussian kernel, we can vectorise the calculations and do them on a GPU. Usually, we would want to inspect our data to make an educated guess of reasonable values for the parameters γ and k, so we only need to calculate the regularisation matrix once. We could also cross-validate over these parameters, but since we already have one parameter per regulariser to cross-validate over, we would end up with three parameters to cross-validate, which we want to avoid. Another difference is the memory footprint in the intermediate computations of $\boldsymbol{\Omega}$. For the Laplacian of a k-nearest neighbour graph and a low k we can use a sparse matrix, but for the Gaussian kernel, the graph Laplacian matrix is dense.

The presence of outliers causes significant differences in the behaviour of these two regularisation approaches. If the distribution of pairwise distances of data points has a heavy right tail, then the Gaussian kernel could potentially give low weight to edges connected to outliers while the outliers would still always be *equally* connected to other data points compared to other points if we use a k-nearest neighbor graph. If the number of variables in our data is high, then using the Gaussian kernel we risk creating many connections between clusters, thus not being able to distinguish clearly between them. This can quickly become an issue for us since we are focusing on data where the number of features is usually higher or on the scale of the number of observations.

Calculating distances in high dimensions is a significant issue in general. When we have enough variables, the observations are almost equally far from each other. But we are not limited to distance metrics for defining the graph. We could, for example, generate pairs of connection, by asking people to label the couple as being similar or not. This data can be used to construct the adjacency matrix and might be more robust than distance calculations in high dimensions.

One thing to try with this approach is to re-evaluate the Laplacian, to get a more robust estimate of the distances, based on the non-zero variables in the discriminant vectors. Re-evaluating the Laplacian could make the distance estimation more robust, but it involves a significant computational burden. It is maybe possible to represent the distance calculations in a clever data-structure, to make this fast.

We refer the reader to contribution B for further examples.

Part II

Applications

CHAPTER 5
Motion Tracking Time Series

Motion tracking has been a topic of research for four decades, Moeslund and Granum provide a comprehensive survey of the literature, until 2001 [MG01]. One of the most impactful recent advances in human motion tracking is the arrival of the Microsoft Kinect sensor and the associated software framework [Sho+11]. The Kinect sensor was the first consumer available electronic device, able to do robust real-time human motion tracking. The motion tracking is predicted on depth images, using a random-forest model [Bre01]. The original Kinect sensor, that we use in this project, uses an infrared projector and infrared camera to generate depth images, using a technique called Light Coding. With the rapid advances in machine learning specifically deep learning, new venues for human pose tracking have become available. One of the most impressive recent deep learning approaches, that only uses color images, can be found in the work DensePose by Güler et al. [GNK18], although it is not obvious how to compare scales between recordings.

We are concerned with the tracking data provided by the Microsoft Kinect sensor. The initial goal was to create a game-like environment to assist with the diagnosis and monitoring changes in movement disorders of young individuals with psychiatric problems. Movement disorders are common side effects of psychiatric drugs, and assessing the movement problems in these patients can be very difficult. Another motivation for doing computerised evaluation is that it does not have a physician's bias.

The rest of this chapter is a summary of the results presented in contributions F, G and H. The first section describes the Motor-Game, then we summarise the results from the mixed effect model analysis from contributions G and H. Finally, we summarise the content from contribution F where sparse ordinal classification is applied to the movement data.

The development of the ordinal classifier presented in chapter 4 was motivated by the problem of ordinal scoring, that was provided by the physicians before inspecting the individuals that played the motor-game. The results of that application are the central part of contribution F.

5.1 The Motor-Game

A comprehensive description of the Motor-Game can be found in contribution G, we summarise the main parts here.

5.1.1 The Game

The Motor-Game was developed with two core-design requirements in mind:

1. The participant should perform the same or similar movements repeatedly as fast and precisely as they can.

2. The game should contain tasks that challenge the hand-eye coordination of the participants, and difficulty of the game tasks should gradually increase during the game.

Another implicit constraint was that the game needed to be somewhat simple, and intuitive to use. The motions should also be rather simple, to prevent mistakes in the tracking. The game requires the participant to stand, facing the sensor (and television screen), and move their arms.

Through most of the game, the participant/player is presented with an upper body of a stickman figure, (made of bubbles), on a television screen that mirrors the movements of the participant (See Fig. 5.1). To begin with, the participant is asked to place themselves at a distance between 2 and 3 meters for optimal recording conditions. The participant is then asked to stretch out their arms, where measurements are obtained for proper calibration of the position. After this procedure, a message appears on the screen stating that the participant should try to finish the upcoming tasks as fast and precisely as possible. The following tasks were split up into three levels.

Before each level a welcome screen appears, describing the next set of tasks. The participant needs to perform similar movements of the hands repeatedly, but the design of random appearance of the button makes it hard to learn and predict the location of these buttons. A score is displayed on the top of the screen where the participant is awarded a higher score if they finish a task fast. In order to avoid interruptions during the recording, a training session is performed before the recording in order for the participant to get familiar with the game.

For analysis, we used data from the first level in the game. In the first level, the participant needs to finish 22 tasks. The first 11 tasks consist of moving the right hand, such that the stickman figure's hand overlays a button that appears on the screen. The hand needs to present on the button for half a second before the button disappears, and the task is completed. The next task starts immediately after the last. The latter 11 tasks are performed with the left hand. There is considerate variation in the amount of movement required, some buttons appear close to the previous button, while others need larger movements to catch them.

5.1.2 The Data

The Kinect provides measurements from every single joint in the tracking skeleton, x, y, z coordinates, on-screen coordinates and the corresponding quaternions, 30 times

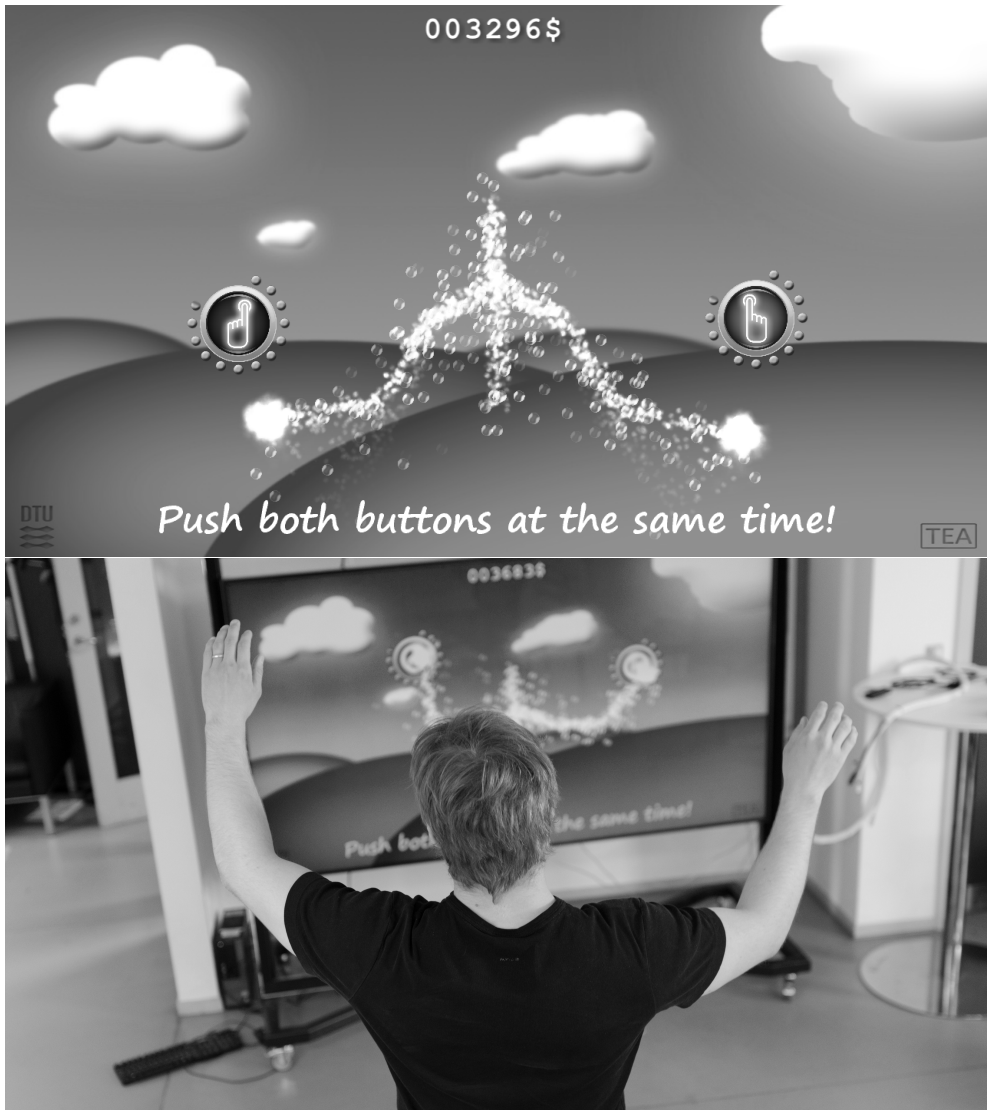


Figure 5.1: Top: Screen-shot from the motor-game, showing what the player sees. Bottom: Me playing the Motor-Game, photo acquired by Morten Hannemose.

per second. This data, along with game-state data is logged during game-play. The game-state data provides information on which task is being solved.

The data is saved to a CSV file. For contributions G and H, we had a hypothesis, that an increase in clinical scores measured by the doctors, would result in more extended gameplay. If we could validate this hypothesis, then there is evidence for the presence of a signal in the data, relevant for classifying individuals according to how the doctors have measured them.

Thus for contributions G and H we only extract the time variables from the game. For contribution F, we use the position of the wrist. Contributions G and H contain extensive summaries of the cohorts and the clinical scores measured. The cohort used for contributions F and G consists of 33 healthy controls and 30 medicated patients with PD. All of the individuals played the Motor-Game, provided demographic data, and were tested on various movement-related rating scales.

One of the most significant challenges was the imbalance in the number of participants affected by movement disorders. More than half of the participants were not affected. We can see a summary of the four clinical scores that were the focus of contribution F in Fig. 5.2. Some of the controls had a positive score in item four on the Simpson Angus Scale (SAS) rating scale. The controls were not measured on the other scale, the *Movement Disorder Society Unified Parkinson's Disease Rating Scale* (MDS-UPDRS), so they were assumed to have a score of zero.

5.2 Linear Mixed Effect Models

For contributions G and H we use linear mixed effect models for the analysis. From the Motor-Game we obtain 22 response measurements for each individual, corresponding to the time it takes to finish the 22 tasks in level one. We want to see how we can predict these scores based on the clinical measurements, and correct for the demographic variables. So we run the same model multiple times, where we only change the clinical variable. The results of this analysis are summarised in tables at the end of the two contributions.

We shall write out the model explicitly. The first two variables are the response and the clinical variable; the rest are variables used for correcting against specific demographic or clinical variables.

1. *Response* is the natural logarithm of time in seconds it takes to finish a single task in level 1 of the Motorgame. So for a single individual we have 22 observations a total of $22 * 63 = 1386$ observations from the cohort. The log transformation is used since the distribution of playing times is skewed to the right. We can see a distribution of the original variable and the transformed variable in Fig. 5.3.

2. *Clinical Score* is the only variable that we change between models. This variable is either one of the SAS variables or some other variables related to status of disease and physical activity.

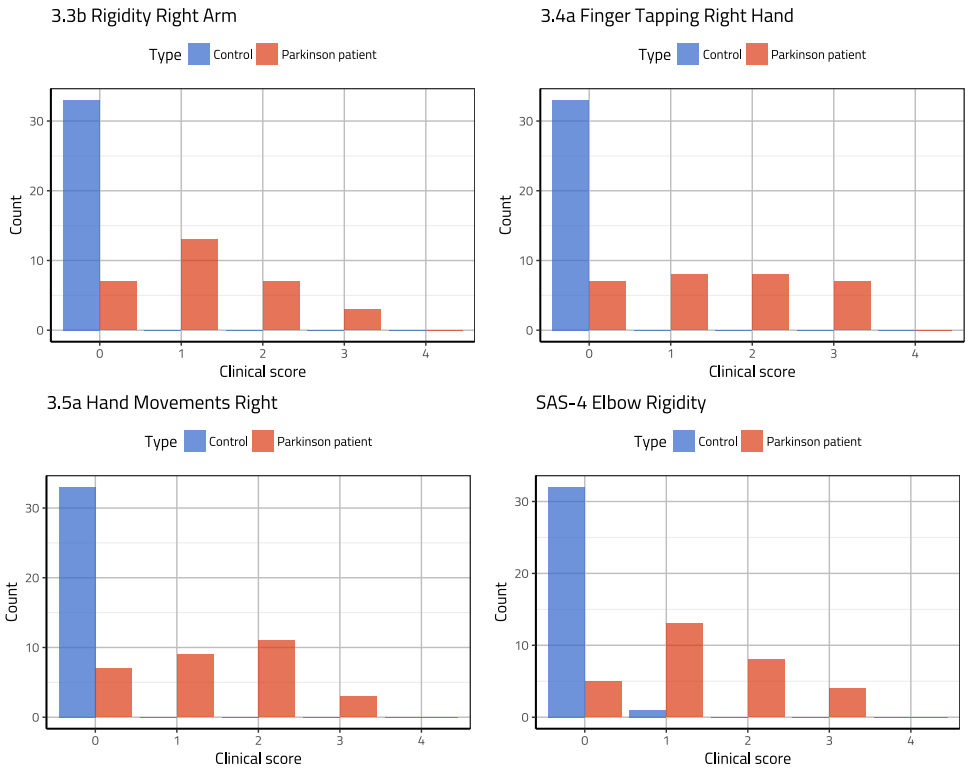


Figure 5.2: The clinical scores which were the focus of contribution F. These are the ones most relevant for the movement data, with the highest prevalence of the conditions. The bottom right is a score from the SAS scale, while the others are from the MDS-UPDRS scale.

3. *Gender*, denoted as G in the model and included as a fixed effect.
4. *Height* in cm.
5. *Weight* in kg.
6. *Task*, factor variable with 22 levels, corresponding to the 22 tasks in the first level of the Motorgame. The tasks are different, so we correct for the general mean it takes to finish the tasks.
7. *Age* in years.
8. *Symbol Coding Tasks*, used as a proxy for general cognition.
9. *Participant number*, this factor variable is used as a random effect due to differences in variation of the response between participants.

The model also includes a general mean term and the error is assumed to be i.i.d. normal.

$$\log(y_{ij}) = \mu + T_j + Cx_{iC} + Gx_{iG} + Hx_{iH} + Ax_{iA} + Wx_{iW} + Sx_{iS} + \varepsilon_i + \varepsilon_{ij} \quad (5.1)$$

The terms in the model are, y the response, and on the right-hand side we have in the following order: μ as a general mean, T_j mean for each of the 22 tasks, C parameter for the clinical score, where x_{iC} is the value for that measurement on individual i. We use i to index the individuals and j for the subtasks. The following terms correspond to the first letter in the enumeration of demographic and clinical variables above. The last two terms are ε_i, the random effect for individuals and the general error term.

We are mostly interested in the variable corresponding to the clinical score. A significant parameter for a clinical score variable confirms our hypothesis that there is a correlation between the time it takes for the Parkinson patients to finish the tasks in the Motor game and the clinical score.

Generally speaking, the clinical variables that were discovered to affect the playing time significantly had to do with rigidity and speed of movement. Two of the most significant effects were *Hand Movement Right hand*, from the MDS-UPDRS rating scale and *finger tapping right hand* from the same scale. The fact that these variables were significant validates our hypothesis that the performance in the game correlates somewhat with these clinical measurements.

Another variable that interestingly was always very significant was an item from the UKU rating scale, where the participant is asked to decipher as many symbols as possible in one minute. The symbols correspond to numbers, and the participant is given a correspondence between the numbers and symbols. This variable is a proxy for cognitive capacity, showing that the cognitive capabilities of the players have a significant effect on performance.

5.3 Ordinal Classification

For the ordinal classification, which is the primary theme of contribution F, we use the data from the Motor-Game to predict the four clinical scores in Fig. 5.2.

We use the tracked position of the participants wrists. For the first 11 tasks, we use the avatar screen coordinate vertical position for the right wrist. The choice of this coordinate is because the avatar has been scaled according to an initial estimate of the player's arm length, making on-screen positions comparable between players. For the following 11 tasks, we use the corresponding coordinates for the left wrist. For each of the 22 tasks, we use measurements for the first second of play. The participants have not reacted in the first five measurements, so we exclude those measurements. The first second of the game is enough for the person to respond and start moving. We can see the contrast between a fast and slow reacting participant in Fig. 5.4. This

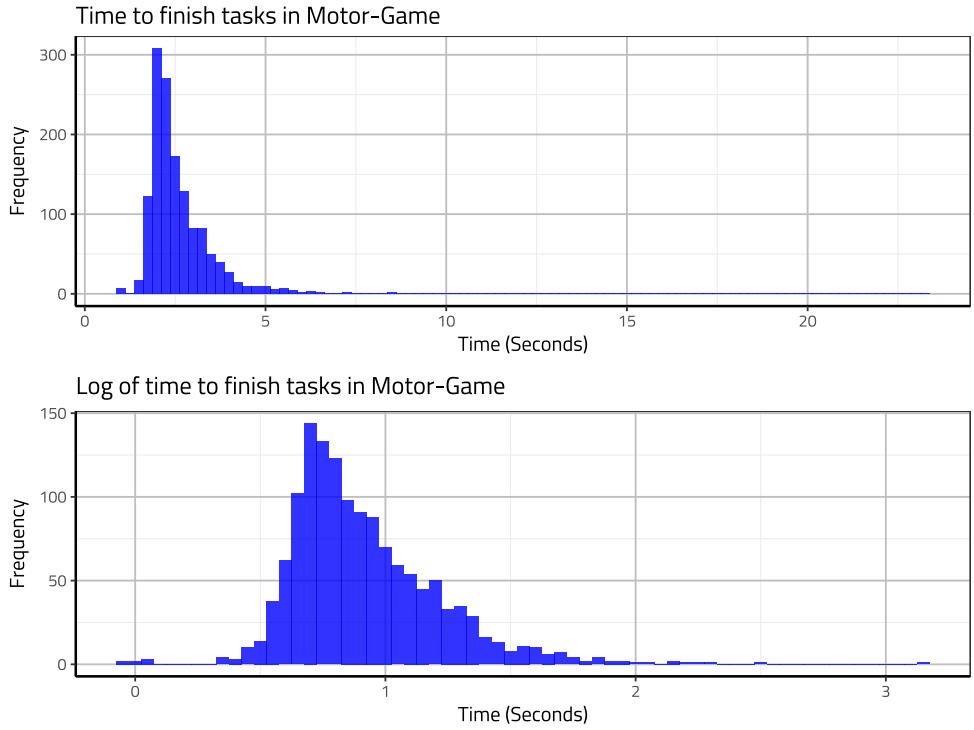


Figure 5.3: Distribution of response variable in linear mixed effect model on the Parkinson's cohort. The distribution remains skewed after the transformation, but outliers are not nearly as severe.

yields, in the end, a total of $p = 20 \times 22 = 440$ variables per participant. We denote m_{i_S} as the mean of the first three measurements for task i and m_{i_E} as the average for the last three measures for task i.

$$\tilde{x}_{ji} := \frac{x_{ji} - m_{i_S}}{|m_{i_S} - m_{i_E}|} \tag{5.2}$$

We further scale the j-th measurement x_{ji} from task i as depicted in Eq. 5.2. Due to variation in the end and starting position, this scaling ensures that the data is more robust to reactions of the participants. So now the data corresponds to a finer detailed measurement, namely comparing reactions, instead of total playing time. The results were presented as balanced leave one out cross-validation accuracy. The confusion matrices reported in the paper indicate that the method captures an ordinal aspect of the data, i.e., if we predict/classify wrong, then we are more likely to predict a wrong label, close to the true one with respect to order.

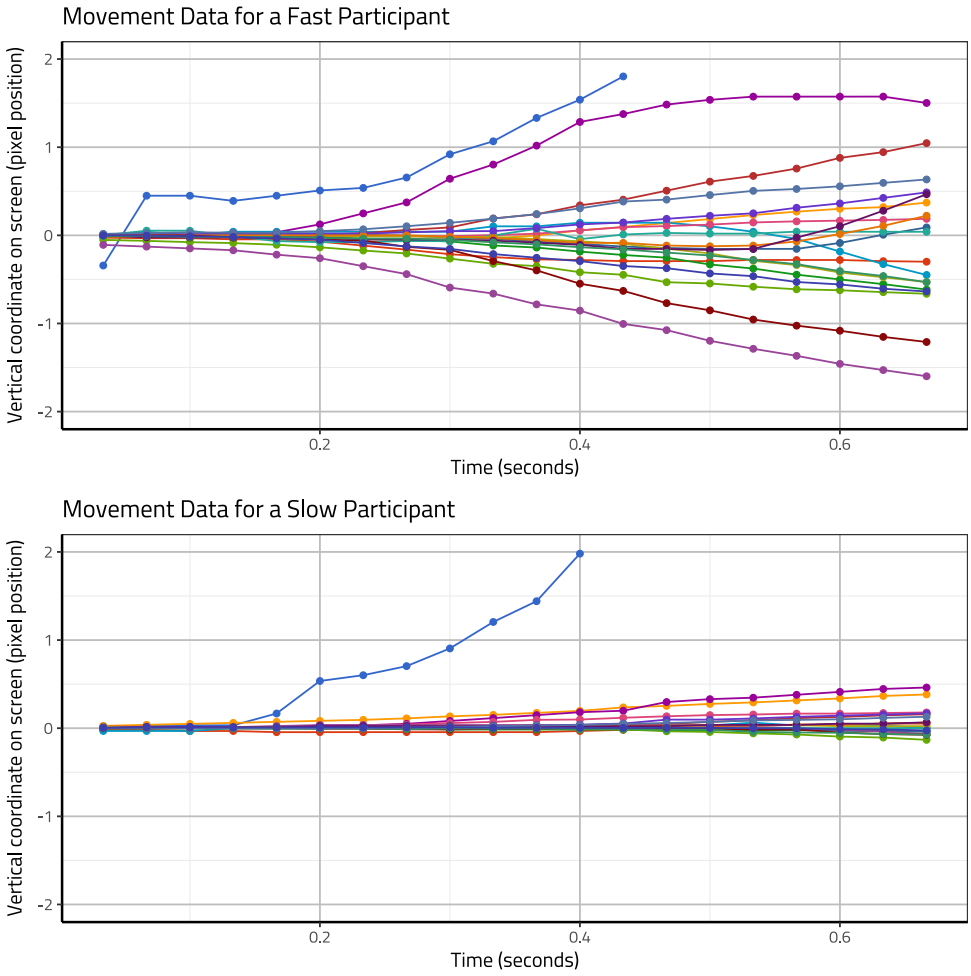


Figure 5.4: Data used for the experiment, vertical position of two subjects' hands over the first second of the 22 tasks. Top we have a participant that generally reacts fast, bottoom we have a more slow moving individual.
.

5.4 Summary

In the publications presented in contributions F, G and H, we have inspected how we can analyse movement time-series data. We have both done it from a mixed effect modelling point of view, as well as with a method developed during this PhD, specifically to analyse this type of data.

The results indicate the potential of doing analysis of this kind of data, how we can take the data from the Motor-game and summarise it down to a single number, that should reflect performance, and be comparable between plays. The mixed-effect model analysis also demonstrates how we can inspect the effect of the different scores, and correct for the appropriate demographic variables.

In a society where the average age is getting higher, and rate of birth is going down, we will soon have a stronger need for tools to monitor the health of individuals and prioritise the usage of health care facilities. These tools need to be able to perform objectively, and be cheap such that subjects could afford to have them in their homes. The work presented here is a step towards finding a solution to this problem.

CHAPTER 6
Multispectral X-ray Imaging

This chapter summarises the results from papers A and E. We recommend reading this chapter before the papers.

6.1 X-ray scanning

Multispectral X-ray scanning is a relatively new imaging modality, which is creeping into the manufacturing space, in particular, luggage scanning and food inspection. A traditional X-ray machine projects X-ray through a material, where a detector measures the attenuation for an X-ray of one or two different wavelengths. For each pixel in the resulting image, we thus have a signal describing how much energy was lost while the X-ray beam traveled to the detector. This relationship can be described through the Beer-Lambert Law (BLL)

$$I = I_0 e^{-\mu\rho d}. \tag{6.1}$$

I_0 in Equation 6.1 corresponds to the initial X-ray intensity, μ and ρ together form the linear absorption coefficient, where μ corresponds to mass absorption, and ρ corresponds to density, and finally d corresponds to the distance traveled by the beam. The parameters depend on the material composition of the scanned object.

The multispectral X-ray scanner provides measurements for multiple different energies of the X-rays, thus providing numerous measurements in each pixel or attenuation over a spectrum. In our case, we have 128 values per pixel. We use a multix scanner from the Danish Meat Research Institute to create the data. These have been shown to be useful for detecting explosives, and are more robust than dual-energy scanners, [Reb+11; Bra+12; Gor+13]

6.2 Looking for Foreign Objects

For the project presented in paper E [Ein+17], we used the scanner at DMRI to create a dataset by scanning minced meat and spring rolls, with different sets of foreign objects overlaid.

The significant challenges in foreign object detection correspond to biomaterials, such as insects or wood chips. Recent advances in grating-based imaging techniques [Pfe+06; Pfe+07], (that measure the attenuation, scattering and refraction of X-ray beams), have shown great promise in detecting organic foreign objects [Ein+16].

Although grating based methods are promising, they still have not been scaled to be used in a production line.

For our data, we could create a robust classifier from a very limited dataset. We further use the fact that SDA can handle more variables than features, so each observation in the training data consists of 5 pixels. The center one, for which the label comes from, and the four surrounding pixels, top, right, bottom, left. This is related to the *third blessing of dimensionality* (See sec. 2.5.2.2), we are in fact averaging the predictions over a small patch, giving us more robust predictions. The results indicate that we can with high accuracy detect scans with foreign objects. We could not demonstrate that biomaterial, such as insects, were identified, but those were not included in the training data since we could not accurately annotate them. Promising new reasearch/applications for Multispectral X-ray data have emerged, in particular for getting more accurate estimates of material content, such as fat in meat. One thing that was particularly important for making the classifier robust to different materials was scaling each pixel by the 95th% quantile of the 128 measurements it contained. This preprocessing also makes the algorithm robust between scanners.

One of the main reasons for using a sparse classifier is to identify which channels/wavelengths are essential for the particular classification task. Reducing the number of used channels would mean storing fewer data points, making a faster inference and in the end it would be easier to use.

6.3 Incorporating a Prior

In paper A we expand on these results and demonstrate how we can impose a structured regulariser as the elastic net coefficient matrix, this is achieved via a Matérn-covariance matrix [Mat13]. The Matérn covariance can be specified by the following equation:

$$C_\nu(d) = \sigma^2 \frac{2^{1-v}}{\Gamma(\nu)} \left(\sqrt{2\nu} \frac{d}{\rho} \right)^\nu K_\nu \left(\sqrt{2\nu} \frac{d}{\rho} \right) \tag{6.2}$$

The Matérn covariance structure 6.2 is governed by the distance d between measurements. In 6.2, Γ refers to the gamma function and K_ν is the modified Bessel function of the second kind.

We align the measurements in a spatial grid as if we would voxelise the image. This alignment induces a natural metric for calculating the distance d needed for the Matérn covariance. In paper A we also demonstrate the run-time benefit of using a low-rank approximation (using Singular Value Decomposition (SVD)) to the regularisation matrix. Using a low-rank regularisation matrix affects accuracy. Other low-rank approximations should be considered since they might preserve more relevant structure.

To summarise, SDA is a viable method to use on multispectral X-ray data. There are several possibilities for extending and continuing with this work, in particular, I

see the task of predicting fat content in meat to be interesting. Some visualisations of the classification can be seen in Fig. 6.4

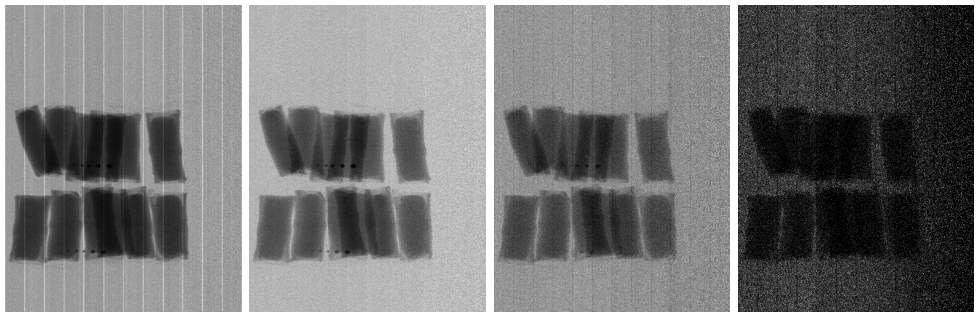


Figure 6.1: Example of raw multispectral X-ray data. The scanned item is a bag of spring rolls. The images correspond to channels (from left to right) 2, 20, 50 and 100. Note that since this is raw data, we have line scanning artefacts corresponding to the intersection of detector modules.

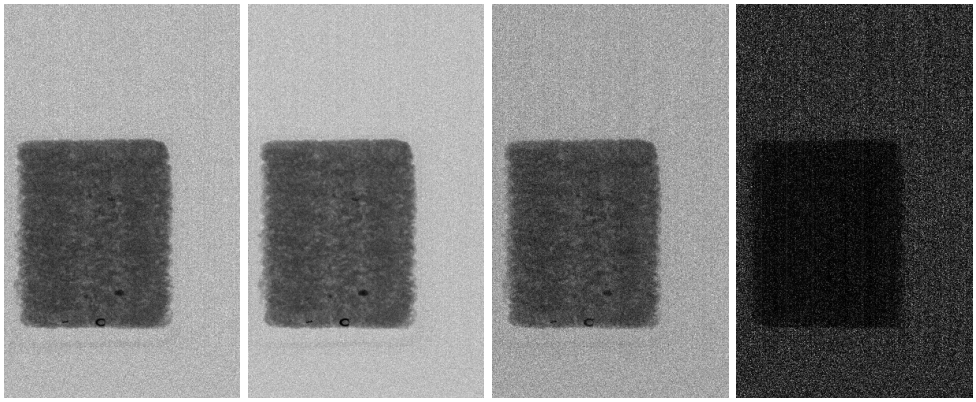


Figure 6.2: Example of preprocessed multispectral X-ray data, scanning artefacts removed. The scanned item is a box of minced meat with several foreign objects overlaid. The images correspond to channels (from left to right) 2, 20, 50 and 100.

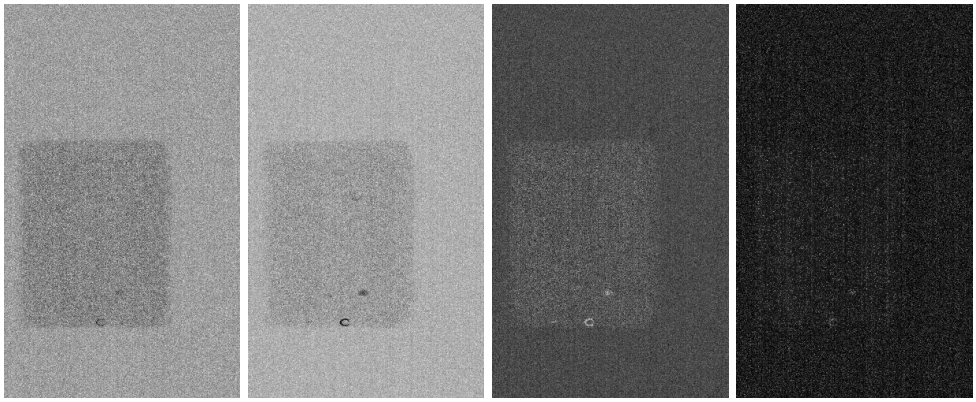


Figure 6.3: Example of preprocessed multispectral X-ray data, scanning artefacts removed and pixels have been scaled to the 95th % quantile. This scaling corresponds to the profiles in Fig. 6.5, which creates peaks at different channels for different materials. The scanned item is a box of minced meat with several foreign objects overlaid. The images correspond to channels (from left to right) 2, 20, 50 and 100.

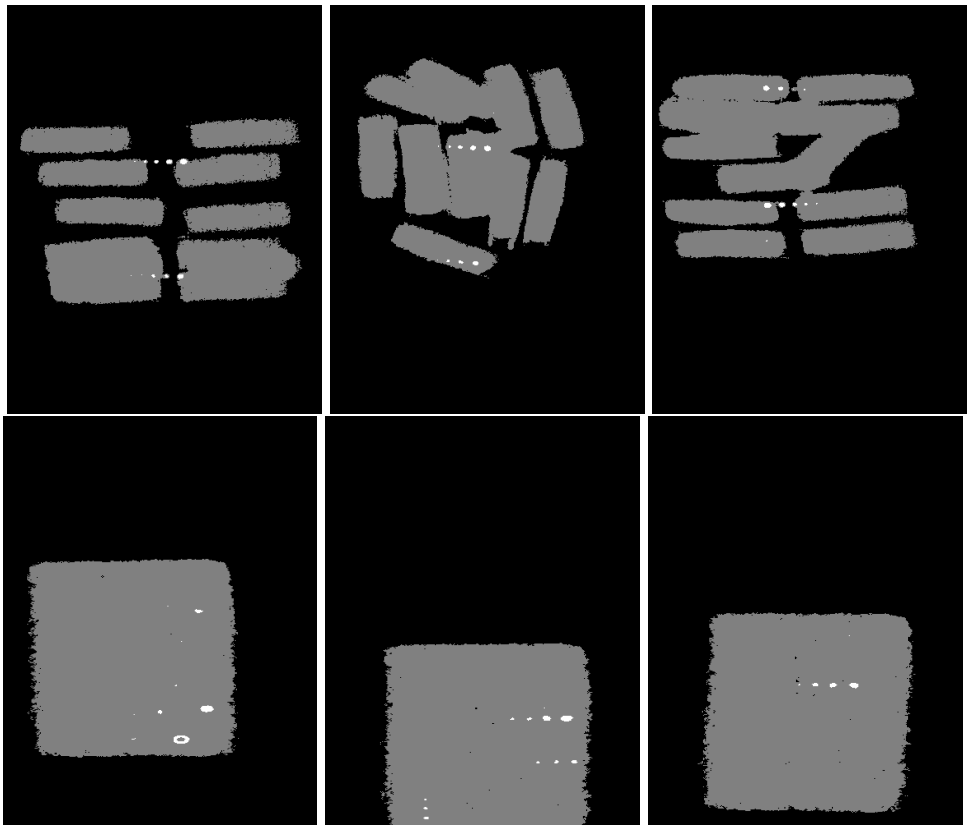


Figure 6.4: Example classification from SDA on scans not used for training. The white intensity represents the foreign object class.

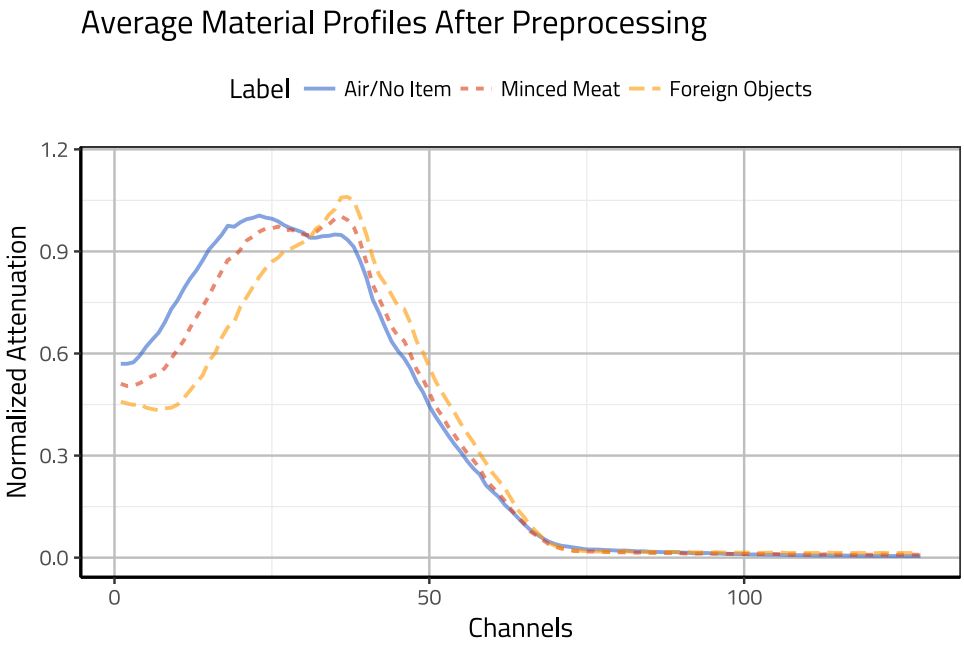


Figure 6.5: Profiles of different materials, averaged over several pixels from Fig. 6.3. These profiles represents the means of the classes for the different items, showing that they have highlights occuring at different wavelengths..

CHAPTER 7
Natural Language Processing

In this chapter, we demonstrate how SDA can be used to extract information from online user reviews, as described in contribution D. We use data from the SNAP dataset [LK14], which is a collection of large network datasets, we further focus on the Amazon reviews data. To extract information from the reviews, we create a sparse ordinal classifier and try to predict the rating based on the words used in the review. The focus is not on building an excellent classifier, but rather interpret the words found, represented as non-zero values in the discriminant vector. These words can give us a quick overview of the product, and we represent them as word clouds, see, e.g., Fig. 7.1 for positive words and Fig. 7.2 for negative words.

There is an abundance of user review data available online, and it is more and more commonly affecting our consumer behaviour. Active scientific research on user reviews concerning social networks, online behaviour, and recommender systems can, e.g., be found in work by [LK14; ML13] and [CDL15]. Online user reviews can have great economic impact, for example in the tourism sector, consumers often rely heavily on previous reviews in travel planning and decision making [GY08]. It is vital for product owners to monitor the state of the reviews, but that can be tedious when they need to go through many reviews or if the same owner is watching multiple products.

7.1 Review Data

There is a multitude of Amazon products available in the SNAP dataset [LK14]. To narrow the focus we selected the *Apps for Android* category to work with using the 5-core dense subset. 5-core means that every product has been reviewed at least five times, by reviewers that have made at least five reviews. If we consider the reviewers and products as nodes in an undirected graph, and reviews as edges connecting users and reviewers, the k-core are the maximal connected subgraph where all vertices have a degree at least k. From this category, we consider the ten products with the highest number of reviews for further analysis.

For each product we train an ordinal SDA model where the number of reviews is balanced, such that each class contains only 100 samples. The ratings are usually unbalanced, where the 2-star reviews are most uncommon. For the test set, we sample 50 samples for each class.

7.2 Preparing Text Documents for SDA

We cannot add text documents directly to the ordinal SDA classifier. We need to represent the text sensibly. One way to do that is to create a Document Term Matrix (DTM). A term in the DTM is an n-gram. A 1-gram is words, while a bi-gram is all variations of two adjacent words present in the data. After finding the unique n-grams in the reviews, our vocabulary, then we can count how often each n-gram appears in a review. Each n-gram represents a variable, and the number of times it appears in the review is what we measure. This process easily creates an enormous amount of words, so we need to prune the vocabulary sensibly. One way to do that is to request the word or n-gram to appear in a minimum number of documents.

For each of the ten datasets, corresponding to the ten highest rated products, we prepare the data with the following steps, using both the test and train data. This procedure yields approximately an order of magnitude more variables than the 500 training samples we have.

(1) We use the `tm` package in `R` to do the following transformations [FH17; MHF08; R C15]:

   (a) Make all the letters lower case.

   (b) Remove punctuation.

   (c) Remove English stop-words.

(2) We use the stemmer from the `SnowballC` package for the English language [Bou14]. The stemmer remove different endings from words, making *sweater* and *sweaters* added to the same n-gram.

(3) We generate 1 to 5 grams and prune the vocabulary such that a term needs to appear in at least 3 documents using the `text2vec` package [SW17].

Here we separate the test and train data again. We scale each column in the training data by the value of the 75% quantile. We use the same values to scale the test data. We **do not** centre the data, the only normalisation is scaling to normalise the contributions from different variables. This normalisation makes it possible to interpret the discriminant vector we find with ordinal SDA. The non-zero parameters with a certain sign correspond to terms that increase the score, while the opposite sign corresponds to a negative score. The magnitude of the absolute value of the parameter corresponds to how strongly the value contributes to the prediction. We can visualise the solutions as the word clouds seen in Figures 7.2 and 7.1 using the `word-cloud` package [Fel14]. The larger the word, corresponds to larger weight in the discriminant vector. We can see that the method picks up words that either positively or negative describe the product. We refer the reader to contribution D for discussion on performance.

Figure 7.1: Positive word-clouds from the online reviews. Top row represents the most rated android games, middle row is 4-6 and last is 7-9. The size of the font is determined by the magnitude in the discriminant vector.

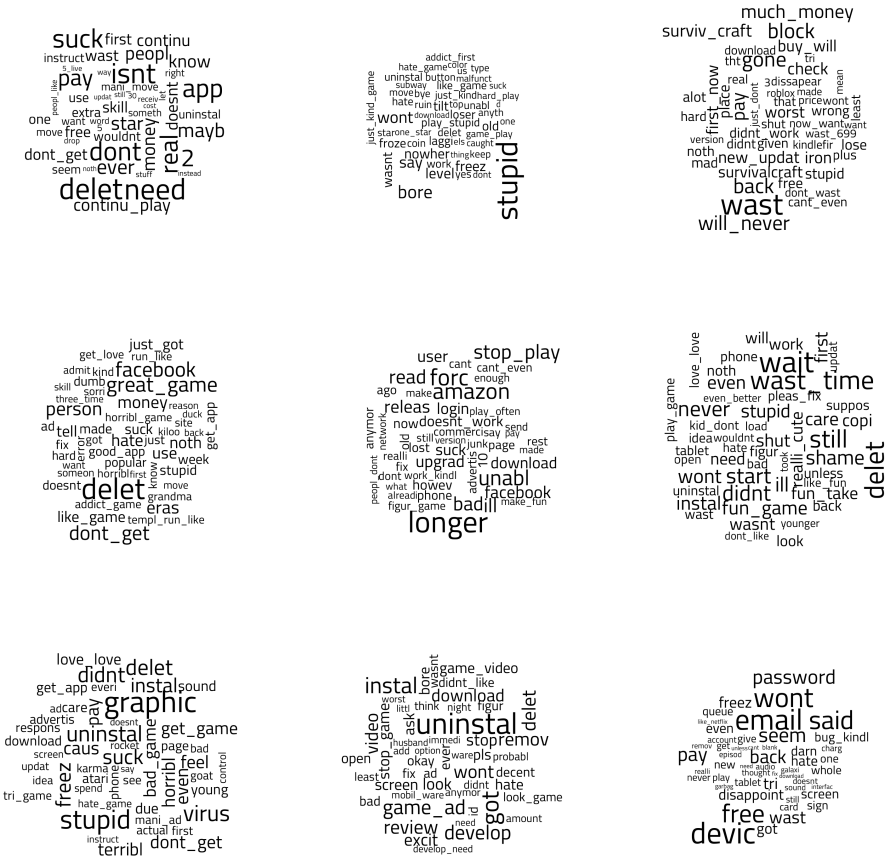


Figure 7.2: Negative word-clouds from the online reviews. Top row represents the most rated android games, middle row is 4-6 and last is 7-9. The size of the font is determined by the magnitude in the discriminant vector.

CHAPTER 8

Conclusion

In this thesis we have presented novel approaches for classification of high dimensional data with applications to computer-aided diagnosis, information retrieval from online user reviews and foreign object detection in multispectral X-ray images. The ordinal type of the response variable was the key component to create a method tailored to these applications.

The tools developed over the course of the thesis are generic and open-source. The tools were mostly developed for the R programming language with some contributions for Matlab. The R package accSDA, which contains methods from this thesis, is available on the Comprehensive R Archive Network (CRAN)[1].

My PhD project was a collaboration with medical partners at Region Hovedstaden and partially with industry experts from DMRI. My contributions focus on theory behind the methods and tools developed. The following three points summarise the developments of the thesis mentioned in the introduction.

1. The data from the Motor-Game was acquired, and analysis of the data is presented in contributions G and H. We discovered that certain clinical measurements could significantly predict the performance. We also discovered that the performance on the cognitive test was a strong predictor, and accounting for repetition was also significant. These discoveries mean that for longitudinal studies of similar nature, we certainly need to account for learning effects.

2. We created a classifier to predict the severity of symptoms, summarised in contribution F. This classifier was built specifically to account for the ordinality of the response. Another improvement to the classifier is the underlying optimisation, presented in contribution A, which also includes proof of convergences, not available before.

3. We have applied the developed methods to other domains in contributions D and E.

8.1 Outlook

There are multiple directions to continue the research presented in this thesis. In particular, the diagnosis and monitoring of movement disorders is an application

[1] https://cran.r-project.org/web/packages/accSDA/index.html

venue which has to go into a longitudinal phase soon. Longitudinal studies hold the potential to reveal what kind of motion degradation we expect over time. The increasing age of the population driven by increased longevity highlights a growing incentive to care more efficiently for the needs of the elderly such as prioritising objectively and in a fair manner who needs care.

Another direction in which to continue this research concerns non-objective information retrieval from online reviews. Ideally the user cares about the signal in all of the noise present online. Future developments of the world-cloud visualisation could include a smart user interface, to navigate to the reviews that most actively reflect on the keywords selected. Marketing research could elucidate what kind of summaries are best for consumers.

Lastly, one of the directions taken in this thesis was to study multispectral X-ray images. There are multiple low-hanging fruits for applying data-driven methods within that domain.

Future developments of the methods and algorithms presented in this thesis have multiple different ways to go forward. A C++ backend can be developed for the `accSDA` package to speed up the computation, and possibly make it scale to much more massive datasets. One example of such high-dimensional data are brain-scans that have been co-registered to the same atlas. In this case we could use SDA to quantify regions in the brain that differ between controls and patients; there are also interesting modelling problems for this domain, that can be incorporated as regularisers. Another high-dimensional application concerns genome wide association studies. In that regard we could do joint analysis of predictors, instead of the traditional univariate analysis. Algorithmic advances also include research on how the optimisation can be parallelised.

To summarise, the tools and methods presented in this thesis relate to Candés' description of Big-Data mentioned in the introduction. The work is an effort to solve some of the problems introduced by Donoho; we can now address $p \gg n$ problems with ordinal labels adequately.

Bibliography

[Abn04] Steven Abney. "Understanding the yarowsky algorithm". In: *Computational Linguistics* 30.3 (2004), pages 365–395.

[AH16] Brendan PW Ames and Mingyi Hong. "Alternating direction method of multipliers for penalized zero-variance discriminant analysis". In: *Computational Optimization and Applications* 64.3 (2016), pages 725–754.

[AK97] Cande V Ananth and David G Kleinbaum. "Regression models for ordinal responses: a review of methods and applications." In: *International journal of epidemiology* 26.6 (1997), pages 1323–1333.

[AO14] Zeyuan Allen-Zhu and Lorenzo Orecchia. "Linear coupling: An ultimate unification of gradient and mirror descent". In: *arXiv preprint arXiv:1407.1537* (2014).

[Atk+17] Summer Atkins et al. "Proximal Methods for Sparse Optimal Scoring and Discriminant Analysis". In: *arXiv preprint arXiv:1705.07194* (2017).

[B+15] Rina Foygel Barber, Emmanuel J Candès, et al. "Controlling the false discovery rate via knockoffs". In: *The Annals of Statistics* 43.5 (2015), pages 2055–2085.

[BC16] Rina Foygel Barber and Emmanuel J Candes. "A knockoff filter for high-dimensional selective inference". In: *arXiv preprint arXiv:1602.03574* (2016).

[BCS18] Rina Foygel Barber, Emmanuel J Candès, and Richard J Samworth. "Robust inference with knockoffs". In: *arXiv preprint arXiv:1801.03896* (2018).

[Bel15] Richard E Bellman. *Adaptive control processes: a guided tour*. Princeton university press, 2015.

[BLP08] Shai Ben-David, Tyler Lu, and Dávid Pál. "Does Unlabeled Data Provably Help? Worst-case Analysis of the Sample Complexity of Semi-Supervised Learning." In: *COLT*. 2008, pages 33–44.

[BLS15] Sébastien Bubeck, Yin Tat Lee, and Mohit Singh. "A geometric alternative to Nesterov's accelerated gradient descent". In: *arXiv preprint arXiv:1506.08187* (2015).

[BN01] Mikhail Belkin and Partha Niyogi. "Laplacian eigenmaps and spectral techniques for embedding and clustering". In: *NIPS*. Volume 14. 14. 2001, pages 585–591.

[BN04] Mikhail Belkin and Partha Niyogi. “Semi-supervised learning on Riemannian manifolds”. In: *Machine learning* 56.1-3 (2004), pages 209–239.

[Bou14] Milan Bouchet-Valat. *SnowballC: Snowball stemmers based on the C libstemmer UTF-8 library*. R package version 0.5.1. 2014. URL: `https://CRAN.R-project.org/package=SnowballC`.

[Box49] George EP Box. “A general distribution theory for a class of likelihood criteria”. In: *Biometrika* 36.3/4 (1949), pages 317–346.

[Boy+11] Stephen Boyd et al. “Distributed optimization and statistical learning via the alternating direction method of multipliers”. In: *Foundations and Trends® in Machine Learning* 3.1 (2011), pages 1–122.

[Bra+12] A Brambilla et al. “CdTe linear pixel X-ray detector with enhanced spectrometric performance for high flux X-ray imaging”. In: *IEEE Transactions on Nuclear Science* 59.4 (2012), pages 1552–1558.

[Bre+01] Leo Breiman et al. “Statistical modeling: The two cultures (with comments and a rejoinder by the author)”. In: *Statistical science* 16.3 (2001), pages 199–231.

[Bre01] Leo Breiman. “Random forests”. In: *Machine learning* 45.1 (2001), pages 5–32.

[Bri90] John S Bridle. “Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters”. In: *Advances in neural information processing systems*. 1990, pages 211–217.

[BT09] Amir Beck and Marc Teboulle. “A fast iterative shrinkage-thresholding algorithm for linear inverse problems”. In: *SIAM journal on imaging sciences* 2.1 (2009), pages 183–202.

[BV04] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[Cas01] George Casella. “Empirical bayes gibbs sampling”. In: *Biostatistics* 2.4 (2001), pages 485–500.

[CC07] Jaime S Cardoso and Joaquim F Costa. “Learning to classify ordinal data: The data replication method”. In: *Journal of Machine Learning Research* 8.Jul (2007), pages 1393–1429.

[CCC05] Jaime S Cardoso, Joaquim F Pinto da Costa, and Maria J Cardoso. “Modelling ordinal relations with SVMs: An application to objective aesthetic evaluation of breast cancer conservative treatment”. In: *Neural Networks* 18.5-6 (2005), pages 808–817.

[CDL15] Justin Cheng, Cristian Danescu-Niculescu-Mizil, and Jure Leskovec. “Antisocial Behavior in Online Discussion Communities.” In: *ICWSM*. 2015, pages 61–70.

[CHH07] Deng Cai, Xiaofei He, and Jiawei Han. “Semi-supervised discriminant analysis”. In: *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE. 2007, pages 1–7.

[CK10] Hyonho Chun and Sündüz Keleş. “Sparse partial least squares regression for simultaneous dimension reduction and variable selection”. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 72.1 (2010), pages 3–25.

[Cle+11] Line Clemmensen et al. “Sparse discriminant analysis”. In: *Technometrics* 53.4 (2011), pages 406–413.

[D+55] George B Dantzig, Alex Orden, Philip Wolfe, et al. “The generalized simplex method for minimizing a linear form under linear inequality restraints”. In: *Pacific Journal of Mathematics* 5.2 (1955), pages 183–195.

[Dah+16] Vedrana Andersen Dahl et al. “Automatic measurement of orbital volume in unilateral coronal synostosis”. In: *Biomedical Imaging (ISBI), 2016 IEEE 13th International Symposium on*. IEEE. 2016, pages 889–893.

[Dal+15] Alessandro Dal Corso et al. “VirtualTable: a projection augmented reality game”. In: *SIGGRAPH Asia 2015 Posters*. ACM. 2015, page 40.

[DCR05] Tijl De Bie, Nello Cristianini, and Roman Rosipal. “Eigenproblems in pattern recognition”. In: *Handbook of Geometric Computing*. Springer, 2005, pages 129–167.

[DJ94] David L Donoho and Jain M Johnstone. “Ideal spatial adaptation by wavelet shrinkage”. In: *biometrika* 81.3 (1994), pages 425–455.

[DJ95] David L Donoho and Iain M Johnstone. “Adapting to unknown smoothness via wavelet shrinkage”. In: *Journal of the american statistical association* 90.432 (1995), pages 1200–1224.

[Dom00] Pedro Domingos. “A unified bias-variance decomposition”. In: *Proceedings of 17th International Conference on Machine Learning*. 2000, pages 231–238.

[Don+00] David L Donoho et al. “High-dimensional data analysis: The curses and blessings of dimensionality”. In: *AMS Math Challenges Lecture* 1 (2000), page 32.

[Dun+14] Gordon W Duncan et al. “Health-related quality of life in early Parkinson’s disease: The impact of nonmotor symptoms”. In: *Movement disorders* 29.2 (2014), pages 195–202.

[Efr+04] Bradley Efron et al. “Least angle regression”. In: *The Annals of statistics* 32.2 (2004), pages 407–499.

[Ein+16] Hildur Einarsdóttir et al. “Novelty detection of foreign objects in food using multi-modal X-ray imaging”. In: *Food Control* 67 (2016), pages 39–47.

[Ein+17] Gudmundur Einarsson et al. “Foreign Object Detection in Multispectral X-ray Images of Food Items Using Sparse Discriminant Analysis”. In: *Scandinavian Conference on Image Analysis*. Springer. 2017, pages 350–361.

[Ein+18] Gudmundur Einarsson et al. *accSDA: Accelerated Sparse Discriminant Analysis*. R package version 1.0.0. 2018. URL: `https://github.com/gumeo/accSDA`.

[ERS14] Gudmundur Einarsson, Thomas P Runarsson, and Gunnar Stefansson. “A competitive coevolution scheme inspired by de”. In: *Differential Evolution (SDE), 2014 IEEE Symposium on*. IEEE. 2014, pages 1–8.

[FB15] Nicolas Flammarion and Francis Bach. “From Averaging to Acceleration, There is Only a Step-size”. In: *arXiv preprint arXiv:1504.01577* (2015).

[Fel14] Ian Fellows. *wordcloud: Word Clouds*. R package version 2.5. 2014. URL: `https://CRAN.R-project.org/package=wordcloud`.

[FH17] Ingo Feinerer and Kurt Hornik. *tm: text mining package. R package version 0.7-1*. `https://CRAN.R-project.org/package=tm`. 2017.

[FHT01] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*. Volume 1. Springer series in statistics New York, 2001.

[Fis36] Ronald A Fisher. “The use of multiple measurements in taxonomic problems”. In: *Annals of human genetics* 7.2 (1936), pages 179–188.

[Fri01] Jerome H Friedman. “Greedy function approximation: a gradient boosting machine”. In: *Annals of statistics* (2001), pages 1189–1232.

[Fri89] Jerome H Friedman. “Regularized discriminant analysis”. In: *Journal of the American statistical association* 84.405 (1989), pages 165–175.

[GNK18] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. “DensePose: Dense Human Pose Estimation In The Wild”. In: *arXiv preprint arXiv:1802.00434* (2018).

[Goo+16] Ian Goodfellow et al. *Deep learning*. Volume 1. MIT press Cambridge, 2016.

[Gor+13] A Gorecki et al. “Comparing performances of a CdTe X-ray spectroscopic detector and an X-ray dual-energy sandwich detector”. In: *Journal of Instrumentation* 8.11 (2013), P11011.

[GY08] Ulrike Gretzel and Kyung Hyan Yoo. “Use and impact of online travel reviews”. In: *Information and communication technologies in tourism 2008* (2008), pages 35–46.

[HBT95] Trevor Hastie, Andreas Buja, and Robert Tibshirani. “Penalized discriminant analysis”. In: *The Annals of Statistics* (1995), pages 73–102.

[HK70] Arthur E Hoerl and Robert W Kennard. “Ridge regression: Biased estimation for nonorthogonal problems”. In: *Technometrics* 12.1 (1970), pages 55–67.

[HN03] Xiaofei He and Partha Niyogi. “Locality preserving projections”. In: *NIPS*. Volume 16. 2003. 2003.

[Hot36] Harold Hotelling. “Relations between two sets of variates”. In: *Biometrika* 28.3/4 (1936), pages 321–377.

[HTB94] Trevor Hastie, Robert Tibshirani, and Andreas Buja. “Flexible discriminant analysis by optimal scoring”. In: *Journal of the American statistical association* 89.428 (1994), pages 1255–1270.

[HTW15] Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical learning with sparsity: the lasso and generalizations*. CRC press, 2015.

[I+05] Hemant Ishwaran, J Sunil Rao, et al. “Spike and slab variable selection: frequentist and Bayesian strategies”. In: *The Annals of Statistics* 33.2 (2005), pages 730–773.

[IH13] Il dar Abdulovič Ibragimov and Rafail Zalmanovich Has’ Minskii. *Statistical estimation: asymptotic theory*. Volume 16. Springer Science & Business Media, 2013.

[IYA16] Aki Ishii, Kazuyoshi Yata, and Makoto Aoshima. “Asymptotic properties of the first principal component and equality tests of covariance matrices in high-dimension, low-sample-size context”. In: *Journal of Statistical Planning and Inference* 170 (2016), pages 186–199.

[JTU03] Ian T Jolliffe, Nickolay T Trendafilov, and Mudassir Uddin. “A modified principal component technique based on the LASSO”. In: *Journal of computational and Graphical Statistics* 12.3 (2003), pages 531–547.

[KL14] Jesse H Krijthe and Marco Loog. “Implicitly constrained semi-supervised linear discriminant analysis”. In: *Pattern Recognition (ICPR), 2014 22nd International Conference on*. IEEE. 2014, pages 3762–3767.

[KL15a] Lorraine V Kalia and Anthony E Lang. “Parkinson’s disease”. In: *The Lancet* 386.9996 (2015), pages 896–912. ISSN: 0140-6736. DOI: https://doi.org/10.1016/S0140-6736(14)61393-3. URL: http://www.sciencedirect.com/science/article/pii/S0140673614613933.

[KL15b] Jesse H Krijthe and Marco Loog. “Implicitly constrained semi-supervised least squares classification”. In: *International Symposium on Intelligent Data Analysis*. Springer. 2015, pages 158–169.

[Kna78] Thomas R Knapp. “Canonical correlation analysis: A general parametric significance-testing system.” In: *Psychological Bulletin* 85.2 (1978), page 410.

[Kri16] Jesse H Krijthe. “RSSL: Semi-supervised Learning in R”. In: *arXiv preprint arXiv:1612.07993* (2016).

[L+10] Qing Li, Nan Lin, et al. “The Bayesian elastic net”. In: *Bayesian Analysis* 5.1 (2010), pages 151–170.

[Lat+16] Pierre Latouche et al. “Combining a relaxed EM algorithm with Occam’s razor for Bayesian variable selection in high-dimensional regression”. In: *Journal of Multivariate Analysis* 146 (2016), pages 177–190.

[LK14] Jure Leskovec and Andrej Krevl. *SNAP Datasets: Stanford Large Network Dataset Collection*. `http://snap.stanford.edu/data`. June 2014.

[LRP16] Laurent Lessard, Benjamin Recht, and Andrew Packard. “Analysis and design of optimization algorithms via integral quadratic constraints”. In: *SIAM Journal on Optimization* 26.1 (2016), pages 57–95.

[LSR73] Peter A Lachenbruch, Cheryl Sneeringer, and Lawrence T Revo. “Robustness of the linear and quadratic discriminant function to certain types of non-normality”. In: *Communications in Statistics* 1.1 (1973), pages 39–56.

[Mah36] Prasanta Chandra Mahalanobis. “On the generalized distance in statistics”. In: National Institute of Science of India. 1936.

[Mat13] Bertil Matérn. *Spatial variation*. Volume 36. Springer Science & Business Media, 2013.

[MG01] Thomas B Moeslund and Erik Granum. “A survey of computer vision-based human motion capture”. In: *Computer vision and image understanding* 81.3 (2001), pages 231–268.

[MHF08] David Meyer, Kurt Hornik, and Ingo Feinerer. “Text mining infrastructure in R”. In: *Journal of statistical software* 25.5 (2008), pages 1–54.

[ML13] Julian McAuley and Jure Leskovec. “Hidden factors and hidden topics: understanding rating dimensions with review text”. In: *Proceedings of the 7th ACM conference on Recommender systems*. ACM. 2013, pages 165–172.

[N+01] Andrew Y Ng, Michael I Jordan, Yair Weiss, et al. “On spectral clustering: Analysis and an algorithm”. In: *NIPS*. Volume 14. 2. 2001, pages 849–856.

[Nes05] Yurii Nesterov. “Smooth minimization of non-smooth functions”. In: *Mathematical programming* 103.1 (2005), pages 127–152.

[Nes13] Yurii Nesterov. “Gradient methods for minimizing composite functions”. In: *Mathematical Programming* 140.1 (2013), pages 125–161.

[Nes83] Yurii Nesterov. “A method of solving a convex programming problem with convergence rate $O(1/k^2)$”. In: *Soviet Mathematics Doklady*. Volume 27. 2. 1983, pages 372–376.

[Nig+00] Kamal Nigam et al. “Text classification from labeled and unlabeled documents using EM”. In: *Machine learning* 39.2-3 (2000), pages 103–134.

[NYD83] Arkadii Nemirovskii, David Borisovich Yudin, and Edgar Ronald Dawson. “Problem complexity and method efficiency in optimization”. In: (1983).

[OC15] Brendan O'Donoghue and Emmanuel Candes. "Adaptive restart for accelerated gradient schemes". In: *Foundations of Computational Mathematics* 15.3 (2015), pages 715–732.

[P+14] Neal Parikh, Stephen Boyd, et al. "Proximal algorithms". In: *Foundations and Trends® in Optimization* 1.3 (2014), pages 127–239.

[PC08] Trevor Park and George Casella. "The bayesian lasso". In: *Journal of the American Statistical Association* 103.482 (2008), pages 681–686.

[Pea01] Karl Pearson. "LIII. On lines and planes of closest fit to systems of points in space". In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2.11 (1901), pages 559–572.

[Pfe+06] Franz Pfeiffer et al. "Phase retrieval and differential phase-contrast imaging with low-brilliance X-ray sources". In: *Nature physics* 2.4 (2006), pages 258–261.

[Pfe+07] F Pfeiffer et al. "High-resolution brain tumor visualization using three-dimensional x-ray phase contrast tomography". In: *Physics in medicine and biology* 52.23 (2007), page 6923.

[Pla50] Ronald L Plackett. "Some theorems in least squares". In: *Biometrika* 37.1/2 (1950), pages 149–157.

[PTB09] Elena Parkhomenko, David Tritchler, and Joseph Beyene. "Sparse canonical correlation analysis with application to genomic data integration". In: *Statistical applications in genetics and molecular biology* 8.1 (2009), pages 1–34.

[R C15] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2015. URL: `https://www.R-project.org/`.

[Rao47] C Radhakrishna Rao. "A statistical criterion to determine the group to which an individual belongs". In: *Nature* 160.4076 (1947), page 835.

[Rao48] C Radhakrishna Rao. "The utilization of multiple measurements in problems of biological classification". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 10.2 (1948), pages 159–203.

[Reb+11] Veronique Rebuffel et al. "New perspectives of X-ray techniques for explosive detection based on CdTe/CdZnTe spectrometric detectors". In: *International Symposium on Digital Industrial Radiology and Computed Tomography–We*. Volume 2. 2011, pages 1–8.

[Roc93] R Tyrrell Rockafellar. "Lagrange multipliers and optimality". In: *SIAM review* 35.2 (1993), pages 183–238.

[SBC14] Weijie Su, Stephen Boyd, and Emmanuel Candes. "A differential equation for modeling nesterov's accelerated gradient method: Theory and insights". In: *Advances in Neural Information Processing Systems*. 2014, pages 2510–2518.

[Sho+11] Jamie Shotton et al. “Real-time human pose recognition in parts from single depth images”. In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. Ieee. 2011, pages 1297–1304.

[Sim+17] Daniel Simpson et al. “Penalising model component complexity: A principled, practical approach to constructing priors”. In: *Statistical science* 32.1 (2017), pages 1–28.

[Sjö+18] Karl Sjöstrand et al. “Spasm: A matlab toolbox for sparse statistical modeling”. In: *Journal of Statistical Software* (2018).

[SNB05] Vikas Sindhwani, Partha Niyogi, and Mikhail Belkin. “Beyond the point cloud: from transductive to semi-supervised learning”. In: *Proceedings of the 22nd international conference on Machine learning*. ACM. 2005, pages 824–831.

[SNZ09] Aarti Singh, Robert Nowak, and Xiaojin Zhu. “Unlabeled data: Now it helps, now it doesn’t”. In: *Advances in neural information processing systems*. 2009, pages 1513–1520.

[SW17] Dmitriy Selivanov and Qing Wang. *text2vec: Modern Text Mining Framework for R*. R package version 0.5.0. 2017. URL: `https://CRAN.R-project.org/package=text2vec`.

[Tib96] Robert Tibshirani. “Regression shrinkage and selection via the lasso”. In: *Journal of the Royal Statistical Society. Series B (Methodological)* (1996), pages 267–288.

[Tik43] Andrey Nikolayevich Tikhonov. “On the stability of inverse problems”. In: *Dokl. Akad. Nauk SSSR*. Volume 39. 1943, pages 195–198.

[TJ07] Nickolay T Trendafilov and Ian T Jolliffe. “DALASS: Variable selection in discriminant analysis via the LASSO”. In: *Computational Statistics & Data Analysis* 51.8 (2007), pages 3718–3736.

[Tse08] Paul Tseng. “On accelerated proximal gradient methods for convex-concave optimization. Submitted to”. In: *SIAM Journal on Optimization* (2008).

[Tsy04] Alexey Tsymbal. “The problem of concept drift: definitions and related work”. In: *Computer Science Department, Trinity College Dublin* 106.2 (2004).

[VN15] Jacob Schack Vestergaard and Allan Aasbjerg Nielsen. “Canonical information analysis”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 101 (2015), pages 1–9.

[WBC17] Asaf Weinstein, Rina Barber, and Emmanuel Candes. “A Power and Prediction Analysis for Knockoffs with Lasso Statistics”. In: *arXiv preprint arXiv:1712.06465* (2017).

[WIP13] De Wang, Danesh Irani, and Calton Pu. "A study on evolution of email spam over fifteen years". In: *Collaborative Computing: Networking, Applications and Worksharing (Collaboratecom), 2013 9th International Conference Conference on*. IEEE. 2013, pages 1–10.

[WK77] Patricia W Wahl and Richard A Kronmal. "Discriminant functions when covariances are unequal and sample sizes are moderate". In: *Biometrics* (1977), pages 479–484.

[Wol+84] Svante Wold et al. "The collinearity problem in linear regression. The partial least squares (PLS) approach to generalized inverses". In: *SIAM Journal on Scientific and Statistical Computing* 5.3 (1984), pages 735–743.

[Wri97] Stephen J Wright. *Primal-dual interior-point methods*. Siam, 1997.

[WT11] Daniela M Witten and Robert Tibshirani. "Penalized classification using Fisher's linear discriminant". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 73.5 (2011), pages 753–772.

[WTH09] Daniela M Witten, Robert Tibshirani, and Trevor Hastie. "A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis". In: *Biostatistics* 10.3 (2009), pages 515–534.

[Yar95] David Yarowsky. "Unsupervised word sense disambiguation rivaling supervised methods". In: *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics. 1995, pages 189–196.

[Yin04] Xiangrong Yin. "Canonical correlation analysis based on information theory". In: *Journal of multivariate analysis* 91.2 (2004), pages 161–176.

[Z+03] Xiaojin Zhu, Zoubin Ghahramani, John Lafferty, et al. "Semi-supervised learning using gaussian fields and harmonic functions". In: *ICML*. Volume 3. 2003, pages 912–919.

[ZH05] Hui Zou and Trevor Hastie. "Regularization and variable selection via the elastic net". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67.2 (2005), pages 301–320.

[Zho+03] Dengyong Zhou et al. "Learning with local and global consistency." In: *NIPS*. Volume 16. 16. 2003, pages 321–328.

Part III

Included Publications

APPENDIX A

Proximal Methods for Sparse Optimal Scoring and Discriminant Analysis

The following manuscript has been submitted to the journal *Statistics and Computing*.

Proximal Methods for Sparse Optimal Scoring and Discriminant Analysis

Summer Atkins [*] Gudmundur Einarsson [†] Brendan Ames [‡] Line Clemmensen [§]

February 2, 2018

Abstract

Linear discriminant analysis (LDA) is a classical method for dimensionality reduction, where discriminant vectors are sought to project data to a lower dimensional space for optimal separability of classes. Several recent papers have outlined strategies for exploiting sparsity for using LDA with high- dimensional data. However, many lack scalable methods for solution of the underlying optimization problems. We propose three new numerical optimization schemes for solving the sparse optimal scoring formulation of LDA based on block coordinate descent, the proximal gradient method, and the alternating direction method of multipliers. We show that the per-iteration cost of these methods scales linearly in the dimension of the data provided restricted regularization terms are employed, and cubically in the dimension of the data in the worst case. Furthermore, we establish that if our block coordinate descent framework generates convergent subsequences of iterates, then these subsequences converge to the stationary points of the sparse optimal scoring problem. Finally, we demonstrate the effectiveness of our new methods with empirical results for classification of Gaussian data and data sets drawn from benchmarking repositories.

1 Introduction

Sparse discriminant techniques have become popular in the last decade due to their ability to provide increased interpretation as well as predictive performance for high-dimensional problems where few observations are present. These approaches typically build upon successes from sparse linear regression, in particular the LASSO and its variants (see Hastie et al. [2013, Section 3.4.2] and Hastie et al. [2015]), by augmenting existing schemes for linear discriminant analysis (LDA) with sparsity-inducing regularization terms, such as the ℓ_1-norm and elastic net.

Thus far, little focus has been put on the optimization strategies of these sparse discriminant methods, nor their computational cost. We propose three novel optimization strategies to obtain discriminant directions in the high-dimensional setting where the number of observations n is much smaller than the ambient dimension p or when features are highly correlated, and prove their convergence. The methods are proposed for multi-class sparse discriminant analysis using the sparse optimal scoring formulation with elastic net penalty proposed in [Clemmensen et al.,

[*]Department of Mathematics, University of Florida, PO Box 118105, Gainesville, FL 32611-8105, srnatkins@ufl.edu

[†]Department of Applied Mathematics and Computer Science, Technical University of Denmark, Building 321, 2800 Kongens Lyngby, Denmark, guei@dtu.dk

[‡]Department of Mathematics, University of Alabama, Box 870350, Tuscaloosa, AL 35487-0350, bpames@ua.edu

[§]Department of Applied Mathematics and Computer Science, Technical University of Denmark, Building 324, 2800 Kongens Lyngby, Denmark, lkhc@dtu.dk

2011]; adding both the ℓ_1- and ℓ_2-norm penalties gives sparse solutions which, in particular, are competitive when high correlations exist in feature space due to the grouping behaviour of the ℓ_2-norm. The first two strategies are proximal gradient methods based on modification of the (fast) iterative shrinkage algorithm [Beck and Teboulle, 2009] for linear inverse problems. The third method uses a variant of the alternating direction method of multipliers similar to that proposed in [Ames and Hong, 2016]. We will see that these heuristics allow efficient classification of high-dimensional data, which was previously impractical using the current state of the art for sparse discriminant analysis. For example, if a diagonal or low-rank Tikhonov regularization term is used and the number of observations is very small relative to p, then the per-iteration cost of each of our algorithms is $\mathcal{O}(p)$; that is, the per-iteration cost of our approach scales linearly with the number of features of our data. Finally, we provide implementations of our algorithms in the form of an `R` package[1] and `Matlab` code[2].

1.1 Existing approaches for sparse LDA

We begin with a brief overview of existing sparse discriminant analysis techniques. Methods such as [Fan and Fan, 2008, Tibshirani et al., 2003, Witten and Tibshirani, 2011] assume independence between the features in the given data. This can lead to poor performance in terms of feature selection as well as predictions, in particular when high correlations exist. Thresholding methods such as [Shao et al., 2011], although proven to be asymptotically optimal, ignore the existing multi-linear correlations when thresholding low correlation estimates. Thresholding, furthermore, does not guarantee an invertible correlation matrix, and often pseudo-inverses must be utilized.

For two-class problems, the results of [Mai and Zou, 2013] established an equivalence between the three methods described in [Clemmensen et al., 2011, Mai et al., 2012, Wu et al., 2008]. These three approaches are formulated as constrained versions of the Fisher's discriminant problem, the optimal scoring problem, and a least squares formulation of linear discriminant analysis, respectively. For scaled regularization parameters, [Mai and Zou, 2013] showed that they all behave asymptotically as Bayes rules. Another two-class sparse linear discriminant method is the linear programming discriminant method proposed in [Cai and Liu, 2011], which finds an ℓ_1-norm penalized estimate of the product between covariance matrix and difference in means.

The sparse optimal scoring (SOS) problem was originally formulated in [Clemmensen et al., 2011] as a multi-class problem seeking at most $K-1$ sparse discriminating directions, whereas [Mai and Zou, 2013] was formulated for binary problems. Mai and Zou later proposed a multi-class sparse discriminant analysis (MSDA) based on the Bayes rule formulation of linear discriminant analysis in [Mai and Zou, 2015]. It imposes only the ℓ_1-norm penalty, whereas the SOS imposes an elastic net penalty (ℓ_1- plus ℓ_2-norm). Adding the ℓ_2-norm can give better predictive performance, in particular when very high correlations exist in data. MSDA, furthermore, finds all discriminative directions at once, whereas SOS finds them sequentially via deflation. A sequential solution can be an advantage if the number of classes is high, and a solution involving only a few directions (the most discriminating ones) is needed. On the other hand, if K is small, finding all directions at once, may be advantageous, in order to not propagate errors in a sequential manner.

Finally, the zero-variance sparse discriminant analysis approach of [Ames and Hong, 2016] reformulates the sparse discriminant analysis problem as an ℓ_1-penalized nonconvex optimization problem in order to sequentially identify discriminative directions in the null-space of the pooled within-class scatter matrix. Most relevant for our discussion here is the use of proximal methods

[1] Available from https://github.com/gumeo/accSDA

[2] Available from http://bpames.people.ua.edu/software.html

to approximately solve the nonconvex optimization problems in [Ames and Hong, 2016]; we will adopt a similar approach for solving the SOS problem.

2 Proximal Methods for Sparse Discriminant Analysis

In this section, we describe a block coordinate descent approach for (approximately) solving the sparse optimal scoring problem for linear discriminant analysis. Proposed in [Hastie et al., 1994], the optimal scoring problem recasts linear discriminant analysis as a generalization of linear regression where both the response variable, corresponding to an optimal labeling or scoring of the classes, and linear model parameters, which yield the discriminant vector, are sought. Specifically, suppose that we have $n \times p$ data matrix $\boldsymbol{X}$, where the rows of $\boldsymbol{X}$ correspond to observations in $\mathbf{R}^p$ sampled from one of K classes; we assume that the data has been centered so that the sample mean is the zero vector $\mathbf{0} \in \mathbf{R}^p$. Optimal scoring generates a sequence of discriminant vectors and conjugate scoring vectors as follows. Suppose that we have identified the first $k-1$ discriminant vectors $\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_{k-1} \in \mathbf{R}^p$ and scoring vectors $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{k-1} \in \mathbf{R}^K$. To calculate the kth discriminant vector $\boldsymbol{\beta}_k$ and scoring vector $\boldsymbol{\theta}_k$, we solve the optimal scoring criterion problem

$$\begin{array}{rll} (\boldsymbol{\theta}_k, \boldsymbol{\beta}_k) = & \underset{\boldsymbol{\theta} \in \mathbf{R}^K, \boldsymbol{\beta} \in \mathbf{R}^p}{\arg\min} & \|\boldsymbol{Y}\boldsymbol{\theta} - \boldsymbol{X}\boldsymbol{\beta}\|^2 \\ & \text{s.t.} & \frac{1}{n}\boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta} = 1, \;\; \boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta}_\ell = 0 \; \forall \ell < k, \end{array} \tag{1}$$

where $\boldsymbol{Y}$ denotes the $n \times K$ indicator matrix for class membership, defined by $y_{ij} = 1$ if the ith observation belongs to the jth class, and $y_{ij} = 0$ otherwise, and $\|\cdot\| : \mathbf{R}^n \to \mathbf{R}$ denotes the vector ℓ_2-norm on $\mathbf{R}^n$ defined by $\|\boldsymbol{y}\| = \sqrt{y_1^2 + y_2^2 + \cdots + y_n^2}$ for all $\boldsymbol{y} \in \mathbf{R}^n$. We direct the reader to [Hastie et al., 1994] for further details regarding the derivation of (1).

A variant of the optimal scoring problem which employs regularization via the elastic net penalty function is proposed in [Clemmensen et al., 2011] As before, suppose that we have identified the first $k-1$ discriminant vectors $\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_{k-1}$ and scoring vectors $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{k-1}$. To calculate the kth sparse discriminant vector $\boldsymbol{\beta}_k$ and scoring vector $\boldsymbol{\theta}_k$, we solve the optimal scoring criterion problem

$$\begin{array}{rll} (\boldsymbol{\theta}_k, \boldsymbol{\beta}_k) = & \underset{\boldsymbol{\theta} \in \mathbf{R}^K, \boldsymbol{\beta} \in \mathbf{R}^p}{\arg\min} & \|\boldsymbol{Y}\boldsymbol{\theta} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + \gamma\boldsymbol{\beta}^T\boldsymbol{\Omega}\boldsymbol{\beta} + \lambda\|\boldsymbol{\beta}\|_1 \\ & \text{s.t.} & \frac{1}{n}\boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta} = 1, \;\; \boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta}_\ell = 0 \; \forall \ell < k, \end{array} \tag{2}$$

where $\boldsymbol{Y} \in \mathbf{R}^{n \times K}$ is again the indicator matrix for class membership, λ and γ are nonnegative tuning parameters, and $\|\cdot\|_1 : \mathbf{R}^p \to \mathbf{R}$ denotes the vector ℓ_1-norm on $\mathbf{R}^p$ defined by $\|\boldsymbol{x}\|_1 = |x_1| + |x_2| + \cdots + |x_p|$ for all $\boldsymbol{x} \in \mathbf{R}^p$. The optimization problem (2) is nonconvex, due to the presence of nonconvex spherical constraints. As such, we do not expect to find a globally optimal solution of (2) using iterative methods. In [Clemmensen et al., 2011], a block coordinate descent method to iteratively approximate solutions of (2) is proposed. Specifically, suppose that we have an estimate $(\boldsymbol{\theta}^t, \boldsymbol{\beta}^t)$ of $(\boldsymbol{\theta}_k, \boldsymbol{\beta}_k)$. To update $\boldsymbol{\theta}^t$, we fix $\boldsymbol{\beta} = \boldsymbol{\beta}^t$ and solve the optimization problem

$$\begin{array}{rll} \boldsymbol{\theta}^{t+1} = & \underset{\boldsymbol{\theta} \in \mathbf{R}^K}{\arg\min} & \|\boldsymbol{Y}\boldsymbol{\theta} - \boldsymbol{X}\boldsymbol{\beta}^t\|^2 \\ & \text{s.t.} & \frac{1}{n}\boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta} = 1, \;\; \boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta}_\ell = 0 \; \forall \ell < k. \end{array} \tag{3}$$

The subproblem (3) is nonconvex in $\boldsymbol{\theta}$, however, it is known that (3) admits an analytic solution and can be solved exactly in polynomial time (see Clemmensen et al. [2011, Section 2.2] for more details). Indeed, we have the following lemma.

Algorithm 1 Block Coordinate Descent for SDA (2)

Start with initial iterate $\boldsymbol{\theta}^0$.

for $t = 0, 1, 2 \ldots$ until converged **do**

Update $\boldsymbol{\beta}^t$ as the solution of (5) with $\boldsymbol{\theta} = \boldsymbol{\theta}^t$ using the solution returned by one of Algorithm 2, Algorithm 3, or Algorithm 4.

Update $\boldsymbol{\theta}^{t+1}$ by

$$\boldsymbol{w} = (\boldsymbol{I} - \boldsymbol{Q}_k \boldsymbol{Q}_k^T \boldsymbol{D}) \boldsymbol{D}^{-1} \boldsymbol{Y}^T \boldsymbol{X} \boldsymbol{\beta}^t, \qquad \boldsymbol{\theta}^{t+1} = \frac{\boldsymbol{w}}{\sqrt{\boldsymbol{w}^T \boldsymbol{D} \boldsymbol{w}}}$$

end for

Lemma 2.1 *The problem* (3) *has optimal solution*

$$\boldsymbol{\theta}^{t+1} = s(\boldsymbol{I} - \boldsymbol{Q}_k \boldsymbol{Q}_k^T \boldsymbol{D}) \boldsymbol{D}^{-1} \boldsymbol{Y}^T \boldsymbol{X} \boldsymbol{\beta}^t, \tag{4}$$

where $\boldsymbol{D} = \frac{1}{n} \boldsymbol{Y}^T \boldsymbol{Y}$, $\boldsymbol{Q}_k$ *is the* $K \times k$ *matrix with columns consisting of the* $k - 1$ *scoring vectors* $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{k-1}$ *and the all-ones vector* $\boldsymbol{e} \in \mathbf{R}^k$, *and* s *is a proportionality constant ensuring that* $(\boldsymbol{\theta}^{t+1})^T \boldsymbol{D} \boldsymbol{\theta}^{t+1} = 1$.

For completeness, we provide a proof of Lemma 2.1 in Appendix A. After we have updated $\boldsymbol{\theta}^{t+1}$, we obtain $\boldsymbol{\beta}^{t+1}$ by solving the unconstrained optimization problem

$$\boldsymbol{\beta}^{t+1} = \underset{\boldsymbol{\beta} \in \mathbf{R}^p}{\arg\min} \quad \|\boldsymbol{Y} \boldsymbol{\theta}^{t+1} - \boldsymbol{X} \boldsymbol{\beta}\|^2 + \gamma \boldsymbol{\beta}^T \boldsymbol{\Omega} \boldsymbol{\beta} + \lambda \|\boldsymbol{\beta}\|_1. \tag{5}$$

That is, we update $\boldsymbol{\beta}^{t+1}$ by solving the generalized elastic net problem (5). It is suggested in [Clemmensen et al., 2011] that (2) can be solved using the algorithm proposed in [Zou and Hastie, 2005]. Unfortunately, this approach carries a per-iteration computational cost on the order of $\mathcal{O}(mnp + m^3)$, where m is the desired number of nonzero coefficients, which is prohibitively expensive if both p and m are large; for example, if $m = cp$ for some constant $c \in (0, 1)$, then the per-iteration cost scales cubically with p.

Our primary contribution is a collection of algorithms for solving the elastic net problem (5). Specifically, we propose three new algorithms, each based on the evaluation of proximal operators. We will see that these algorithms require significantly fewer computational resources than the elastic net algorithm if we exploit structure in the regularization parameter $\boldsymbol{\Omega}$.

2.1 Proximal Gradient Algorithms for the Generalized Elastic Net Problem

Given a convex function $f : \mathbf{R}^p \to \mathbf{R}$, the *proximal operator* $\text{prox}_f : \mathbf{R}^p \to \mathbf{R}^p$ of f is defined by

$$\underset{f}{\text{prox}}(\boldsymbol{y}) = \underset{\boldsymbol{x} \in \mathbf{R}^p}{\arg\min} \left\{ f(\boldsymbol{x}) + \frac{1}{2} \|\boldsymbol{x} - \boldsymbol{y}\|^2 \right\},$$

which yields a point that balances the competing objectives of being near $\boldsymbol{y}$ while simultaneously minimizing f. The use of proximal operators is a classical technique in optimization, particularly as surrogates for gradient descent steps for minimization of nonsmooth functions. For example, consider the optimization problem

$$\min_{\boldsymbol{x} \in \mathbf{R}^p} f(\boldsymbol{x}) + g(\boldsymbol{x}), \tag{6}$$

where $f : \mathbf{R}^p \to \mathbf{R}$ is differentiable and $g : \mathbf{R}^p \to \mathbf{R}$ is potentially nonsmooth. That is, (6) minimizes an objective that can be decomposed as the sum of a differentiable function f and nonsmooth function g. To solve (6), the *proximal gradient method* performs iterations consisting of a step in the direction of the negative gradient $-\nabla f$ of the smooth part f followed by evaluation of the proximal operator of g: given iterate $\boldsymbol{x}^t$, we obtain the updated iterate $\boldsymbol{x}^{t+1}$ by

$$\boldsymbol{x}^{t+1} = \operatorname*{prox}_{\alpha_t g}(\boldsymbol{x}^t - \alpha_t \nabla f(\boldsymbol{x}^t)) = \operatorname*{arg\,min}_{\boldsymbol{x} \in \mathbf{R}^p} \left\{ \alpha_t g(\boldsymbol{x}) + \frac{1}{2} \|\boldsymbol{x} - (\boldsymbol{x}^t - \alpha_t \nabla f(\boldsymbol{x}^t))\|^2 \right\},$$

where α_t is a step length parameter. If both f and g are differentiable and the step size α_t is small, then this approach reduces to the classical gradient descent iteration: $\boldsymbol{x}^{t+1} \approx \boldsymbol{x}^t - \alpha_t \nabla f(\boldsymbol{x}^t) - \alpha_t \nabla g(\boldsymbol{x}^t)$. We direct the reader to the recent survey article [Parikh and Boyd, 2014] for more details regarding the proximal gradient method and proximal operators in general. Expanding the residual norm term $\|\boldsymbol{Y}\boldsymbol{\theta} - \boldsymbol{X}\boldsymbol{\beta}\|^2$ in the objective of (5) and dropping the constant term shows that (5) is equivalent to minimizing

$$f(\boldsymbol{\beta}) = \frac{1}{2}\boldsymbol{\beta}^T \boldsymbol{A}\boldsymbol{\beta} + \boldsymbol{d}^T\boldsymbol{\beta} + \lambda\|\boldsymbol{\beta}\|_1, \tag{7}$$

where $\boldsymbol{A} = 2(\boldsymbol{X}^T\boldsymbol{X} + \gamma\boldsymbol{\Omega})$ and $\boldsymbol{d} = -2\boldsymbol{X}^T\boldsymbol{Y}\boldsymbol{\theta}^{t+1}$. We can decompose f as $f(\boldsymbol{\beta}) = f_1(\boldsymbol{\beta}) + f_2(\boldsymbol{\beta})$, where $f_1(\boldsymbol{\beta}) = \frac{1}{2}\boldsymbol{\beta}^T\boldsymbol{A}\boldsymbol{\beta} + \boldsymbol{d}^T\boldsymbol{\beta}$ and $f_2(\boldsymbol{\beta}) = \lambda\|\boldsymbol{\beta}\|_1$. Note that f is strongly convex if the penalty matrix $\boldsymbol{\Omega}$ is positive definite; in this case (7) has unique minimizer. Note further that f_1 is differentiable with

$$\nabla f_1(\boldsymbol{\beta}) = \boldsymbol{A}\boldsymbol{\beta} + \boldsymbol{d}.$$

Moreover, the proximal operator of the ℓ_1-norm term $f_2(\boldsymbol{\beta}) = \lambda\|\boldsymbol{\beta}\|_1$ is given by

$$\operatorname*{prox}_{\lambda\|\cdot\|_1}(\boldsymbol{y}) = \operatorname{sign}(\boldsymbol{y}) \max\{|\boldsymbol{y}| - \lambda\boldsymbol{e}, \boldsymbol{0}\} =: S_\lambda(\boldsymbol{y});$$

see [Parikh and Boyd, 2014, Section 6.5.2]. The proximal operator $S_\lambda = \operatorname{prox}_{\lambda\|\cdot\|_1}$ is often called the *soft thresholding operator* (with respect to the threshold λ) and $\operatorname{sign} : \mathbf{R}^p \to \mathbf{R}^p$ and $\max : \mathbf{R}^p \times \mathbf{R}^p \to \mathbf{R}^p$ are the element-wise sign and maximum mappings defined by

$$[\operatorname{sign}(\boldsymbol{y})]_i = \operatorname{sign}(y_i) = \begin{cases} +1, & \text{if } y_i > 0 \\ 0, & \text{if } y_i = 0 \\ -1, & \text{if } y_i < 0 \end{cases}$$

and $[\max(\boldsymbol{x}, \boldsymbol{y})]_i = \max(x_i, y_i)$. Using this decomposition, we can apply the proximal gradient method to generate a sequence of iterates $\{\boldsymbol{\beta}^t\}$ by

$$\boldsymbol{\beta}^{t+1} = \operatorname{sign}(\boldsymbol{p}^t) \max\{|\boldsymbol{p}^t| - \lambda\alpha_t\boldsymbol{e}, \boldsymbol{0}\}, \tag{8}$$

where

$$\boldsymbol{p}^t = \boldsymbol{\beta}^t - \alpha_t \nabla f(\boldsymbol{\beta}^t) = \boldsymbol{\beta}^t - \alpha_t(\boldsymbol{A}\boldsymbol{\beta}^t + \boldsymbol{d}); \tag{9}$$

here, $\boldsymbol{e}$ and $\boldsymbol{0}$ denote the all-ones and all-zeros vectors in $\mathbf{R}^p$. Our proximal gradient algorithm is summarized in Algorithm 2. It is important to note that this update scheme is virtually identical to that of the *iterative soft thresholding algorithm (ISTA)* for linear inverse problems (see Beck and Teboulle [2009]). Here, our problem and update formula differs only from that typically associated with ISTA in the presence of the Tikhonov regression term $\boldsymbol{\beta}^T\boldsymbol{\Omega}\boldsymbol{\beta}$ in our model. As an immediate consequence, we see that the sequence of function values $\{f(\boldsymbol{\beta}^t)\}$ generated by Algorithm 2, with an appropriate choice of step lengths $\{\alpha_t\}$, converges sublinearly to the optimal function value of (7) at a rate no worse than $\mathcal{O}(1/t)$ (compare to Beck and Teboulle [2009, Theorem 3.1]).

Algorithm 2 Proximal gradient method for solving the elastic net subproblem (5)

Start with initial iterate $\boldsymbol{\beta}^0$ and sequence of step lengths $\{\alpha_t\}_{t=0}^{\infty}$.
for $t = 0, 1, 2 \ldots$ until converged **do**
Update gradient term by (9):

$$\boldsymbol{p}^t = \boldsymbol{\beta}^t - \alpha_t(\boldsymbol{A}\boldsymbol{\beta}^t + \boldsymbol{d}).$$

Update iterate using proximal gradient step (8):

$$\boldsymbol{\beta}^{t+1} = \operatorname{sign}(\boldsymbol{p}^t)\max\{|\boldsymbol{p}^t| - \lambda\alpha_t\boldsymbol{e}, \boldsymbol{0}\}$$

end for

Theorem 2.1 *Let $\{\boldsymbol{\beta}^t\}$ be generated by Algorithm 2 with initial iterate $\boldsymbol{\beta}^0$ and constant step size $\alpha_t = \alpha \in (0, 2/\|\boldsymbol{A}\|)$, where $\|\boldsymbol{A}\| = \lambda_{\max}(\boldsymbol{A})$ denotes the spectral norm of $\boldsymbol{A}$ equal to the largest magnitude eigenvalue of $\boldsymbol{A}$. Suppose that $\boldsymbol{\beta}^*$ is a minimizer of f. Then*

$$f(\boldsymbol{\beta}^t) - f(\boldsymbol{\beta}^*) \leq \frac{\alpha\|\boldsymbol{A}\|\|\boldsymbol{\beta}^0 - \boldsymbol{\beta}^*\|^2}{2t} \tag{10}$$

for any $t \geq 1$.

As an immediate consequence of Theorem 2.1, the sequence of iterates generated by Algorithm 2 converges to the unique minimizer of (5) if the penalty parameter $\boldsymbol{\Omega}$ is chosen to be positive definite. If we choose $\boldsymbol{\Omega}$ to be positive semidefinite, then any limit point of the sequence of iterates generated by Algorithm 2 is a minimizer of (5); we will see that using such a matrix may have attractive computational advantages despite this loss of uniqueness.

It is reasonably easy to see that the quadratic term of f is differentiable and has Lipschitz continuous gradient with constant $L = \|\boldsymbol{A}\|$; this is the significance of the $\|\boldsymbol{A}\|$ term in (10). In order to ensure convergence in our proximal gradient method, we need to estimate $\|\boldsymbol{A}\|$ to choose a sufficiently small step size α. Computing this Lipschitz constant may be prohibitively expensive for large p; one can typically calculate $\|\boldsymbol{A}\|$ to arbitrary precision using variants of the Power Method (see Golub and Van Loan [2013, Sections 7.3.1, 8.2]) at a cost of $\mathcal{O}(p^2 \log p)$ floating point operations. Instead, we could use an upper bound $\tilde{L} \geq L$ to compute our constant step size $\alpha = 1/\tilde{L} \leq 1/L$. For example, when $\boldsymbol{\Omega}$ is a diagonal matrix, we estimate $\|\boldsymbol{A}\|$ by

$$\begin{aligned}\|\boldsymbol{A}\| = 2\|\gamma\boldsymbol{\Omega} + \boldsymbol{X}^T\boldsymbol{X}\| &\leq 2\gamma\|\operatorname{diag}(\operatorname{Diag}(\boldsymbol{\Omega}))\|_\infty + 2\|\boldsymbol{X}\|_2^F \\ &\leq 2\gamma\|\operatorname{diag}(\operatorname{Diag}(\boldsymbol{\Omega}))\|_\infty + 2\|\boldsymbol{X}\|_1\|\boldsymbol{X}\|_\infty \approx \frac{1}{\alpha},\end{aligned}$$

where $\operatorname{diag}(\boldsymbol{M}) \in \mathbf{R}^p$ is the vector of diagonal entries of the matrix $\boldsymbol{M} \in \mathbf{R}^{p\times p}$ and $\operatorname{Diag}(\boldsymbol{m}) \in \mathbf{R}^{p\times p}$ is the diagonal matrix with ith diagonal entry equal to m_i for the vector $\boldsymbol{m} \in \mathbf{R}^p$. Here, we used the triangle inequality and the identity $\|\boldsymbol{X}^T\boldsymbol{X}\| \leq \|\boldsymbol{X}\|_F^2 \leq \|\boldsymbol{X}\|_1\|\boldsymbol{X}\|_\infty$, where

$$\|\boldsymbol{X}\|_F^2 = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{p} x_{ij}^2}, \qquad \|\boldsymbol{X}\|_1 = \max_{1\leq j\leq p}\sum_{i=1}^{n}|x_{ij}|, \qquad \text{and} \qquad \|\boldsymbol{X}\|_\infty = \max_{1\leq i\leq n}\sum_{j=1}^{p}|x_{ij}|.$$

Each of these norms and, thus, these estimates can be computed using only $\mathcal{O}(np)$ floating point operations.

Algorithm 3 Accelerated proximal gradient method for solving (5)

Start with initial iterate $\boldsymbol{\beta}^0$, step length α, and sequence of extrapolation parameters $\{\omega_t\}_{t=0}^{\infty}$.
for $t = 0, 1, 2 \ldots$ until converged **do**
Update momentum term by (11):

$$\boldsymbol{y}^{t+1} = \boldsymbol{x}^t + \omega_t(\boldsymbol{x}^t - \boldsymbol{x}^{t-1}).$$

Update gradient term by (9):

$$\boldsymbol{p}^t = \boldsymbol{y}^t - \alpha(\boldsymbol{A}\boldsymbol{y}^t + \boldsymbol{d}).$$

Update iterate using proximal gradient step (8):

$$\boldsymbol{\beta}^{t+1} = \operatorname{sign}(\boldsymbol{p}^t)\max\{|\boldsymbol{p}^t| - \lambda\alpha\boldsymbol{e}, \boldsymbol{0}\}.$$

end for

ADD SUBSECTION RE: BACKTRACKING

The similarity of our method to iterative soft thresholding and, more generally, our use of proximal gradient steps to mimic the classical gradient method for minimization of our nonsmooth objective, suggests that we may be able to use momentum terms to accelerate convergence of our iterates. In particular, we modify the fast iterative soft thresholding algorithm (FISTA) described in [Beck and Teboulle, 2009, Section 4] to solve our subproblem. This approach extends a variety of accelerated gradient descent methods, most notably those of Nesterov [Nesterov, 1983, 2005, 2013], to minimization of composite convex functions; for further details regarding the acceleration process and motivation for why such acceleration is possible, we direct the reader to the references [Allen-Zhu and Orecchia, 2014, Bubeck et al., 2015, Flammarion and Bach, 2015, Lessard et al., 2016, O'Donoghue and Candes, 2015, Su et al., 2014, Tseng, 2008].

We accelerate convergence of our iterates by taking a proximal gradient step from an extrapolation of the last two iterates. Applied to (6), the accelerated proximal gradient method features updates of the form

$$\boldsymbol{y}^{t+1} = \boldsymbol{x}^t + \omega_t(\boldsymbol{x}^t - \boldsymbol{x}^{t-1}) \tag{11}$$

$$\boldsymbol{x}^{t+1} = \operatorname*{prox}_{\alpha g}(\boldsymbol{y}^{t+1} - \alpha\nabla f(\boldsymbol{y}^{t+1})), \tag{12}$$

where $\omega_t \in [0, 1)$ is an extrapolation parameter; a standard choice of this parameter is $t/(t+3)$. Applying this modification to our original proximal gradient algorithm yields Algorithm 3, which generates a sequence of iterates converging (in function value) to the optimal solution of (5) at rate $\mathcal{O}(1/t^2)$ (compare to Beck and Teboulle [2009, Theorem 4.4]).

Theorem 2.2 *Let $\{\boldsymbol{\beta}^t\}$ be generated by Algorithm 3 with initial iterate $\boldsymbol{\beta}^0$ and constant step size $\alpha_t = \alpha \in (0, 2/\|\boldsymbol{A}\|)$. Then there exists constant C such that*

$$f(\boldsymbol{\beta}^t) - f(\boldsymbol{\beta}^*) \leq \frac{C\alpha\|\boldsymbol{A}\|\|\boldsymbol{\beta}^0 - \boldsymbol{\beta}^*\|^2}{t^2} \tag{13}$$

for any $t \geq 1$ and minimizer $\boldsymbol{\beta}^*$ of f.

We conclude by proposing a third algorithm for minimization of (7) based on the *alternating direction method of multipliers* (ADMM) for minimizing separable functions under linear coupling

constraints. The ADMM solves problems of the form

$$\min_{\boldsymbol{x}\in\mathbf{R}^p,\boldsymbol{y}\in\mathbf{R}^m}\left\{f(\boldsymbol{x})+g(\boldsymbol{y}):\boldsymbol{A}\boldsymbol{x}+\boldsymbol{B}\boldsymbol{y}=\boldsymbol{c}\right\}, \tag{14}$$

via an approximate dual gradient ascent, where $f:\mathbf{R}^p\to\mathbf{R}$, $g:\mathbf{R}^m\to\mathbf{R}$, $\boldsymbol{A}\in\mathbf{R}^{r\times p}$, $\boldsymbol{B}\in\mathbf{R}^{r\times m}$, and $\boldsymbol{c}\in\mathbf{R}^r$; we direct the reader to the recent survey [Boyd et al., 2011] for more details regarding the ADMM.

Recall that the minimization of the composite function f defined in (7) can be written as the unconstrained optimization problem

$$\min_{\boldsymbol{\beta}\in\mathbf{R}^p} f(\boldsymbol{\beta})=\min_{\boldsymbol{\beta}\in\mathbf{R}^p}\frac{1}{2}\boldsymbol{\beta}^T\boldsymbol{A}\boldsymbol{\beta}+\boldsymbol{d}^T\boldsymbol{\beta}+\lambda\|\boldsymbol{\beta}\|_1. \tag{15}$$

We can rewrite (15) in an equivalent form appropriate for the ADMM by splitting the decision variable $\boldsymbol{\beta}\in\mathbf{R}^p$ as two new variables $\boldsymbol{x},\boldsymbol{y}\in\mathbf{R}^p$ with an accompanying linear coupling constraint $\boldsymbol{x}=\boldsymbol{y}$. Under this change of variables, we can express (15) as

$$\min_{\boldsymbol{x},\boldsymbol{y}\in\mathbf{R}^p}\left\{\frac{1}{2}\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x}-\boldsymbol{x}^T\boldsymbol{d}+\lambda\|\boldsymbol{y}\|_1:\boldsymbol{x}-\boldsymbol{y}=\boldsymbol{0}\right\}. \tag{16}$$

The ADMM generates a sequence of iterates using approximate dual gradient ascent steps as follows. The augmented Lagrangian of (16) is defined by

$$L_\mu(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})=\frac{1}{2}\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x}-\boldsymbol{x}^T\boldsymbol{d}+\lambda\|\boldsymbol{y}\|_1+\boldsymbol{z}^T(\boldsymbol{x}-\boldsymbol{y})+\frac{\mu}{2}\|\boldsymbol{x}-\boldsymbol{y}\|^2$$

for all $\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}\in\mathbf{R}^p$; here, $\mu>0$ is a penalty parameter controlling the emphasis on enforcing feasibility of the primal iterates $\boldsymbol{x}$ and $\boldsymbol{y}$. To approximate the gradient of the dual functional of (16), we alternately minimize the augmented Lagrangian with respect to $\boldsymbol{x}$ and $\boldsymbol{y}$. We then update the dual variable $\boldsymbol{z}$ by a dual ascent step using this approximate gradient.

Suppose that we have the iterates $(\boldsymbol{x}^t,\boldsymbol{y}^t,\boldsymbol{z}^t)$ after t steps of our algorithm. To update $\boldsymbol{x}$, we take

$$\boldsymbol{x}^{t+1}=\operatorname*{arg\,min}_{\boldsymbol{x}\in\mathbf{R}^p}L_\mu(\boldsymbol{x},\boldsymbol{y}^t,\boldsymbol{z}^t)=\operatorname*{arg\,min}_{\boldsymbol{x}\in\mathbf{R}^p}\frac{1}{2}\boldsymbol{x}^T(\mu\boldsymbol{I}+\boldsymbol{A})\boldsymbol{x}-\boldsymbol{x}^T(\boldsymbol{d}+\mu\boldsymbol{y}^t-\boldsymbol{z}^t).$$

Applying the first order necessary and sufficient conditions for optimality, we see that $\boldsymbol{x}^{t+1}$ must satisfy

$$(\mu\boldsymbol{I}+\boldsymbol{A})\boldsymbol{x}^{t+1}=\boldsymbol{d}+\mu\boldsymbol{y}^t-\boldsymbol{z}^t. \tag{17}$$

Thus, $\boldsymbol{x}^{t+1}$ is obtained as the solution of a linear system. Note that the coefficient matrix $\mu\boldsymbol{I}+\boldsymbol{A}$ is independent of t; we take the Cholesky decomposition of $\mu\boldsymbol{I}+\boldsymbol{A}=\boldsymbol{B}\boldsymbol{B}^T$ during a preprocessing step and obtain $\boldsymbol{x}^{t+1}$ by solving the two triangular systems given by

$$\boldsymbol{B}\boldsymbol{B}^T\boldsymbol{x}^{t+1}=\boldsymbol{d}+\mu\boldsymbol{y}^T-\boldsymbol{z}^T.$$

When the generalized elastic net matrix $\boldsymbol{\Omega}$ is diagonal, or $\mu\boldsymbol{I}+2\gamma\boldsymbol{\Omega}$ is otherwise easy to invert, we can invoke the Sherman-Morrison-Woodbury formula (see Golub and Van Loan [2013, Section 2.1.4]) to more efficiently solve this linear system; more details will be provided in Section 2.2. In particular, we see that

$$(\mu\boldsymbol{I}+2\gamma\boldsymbol{\Omega}+2\boldsymbol{X}^T\boldsymbol{X})^{-1}=\boldsymbol{M}^{-1}-2\boldsymbol{M}^{-1}\boldsymbol{X}^T(\boldsymbol{I}+2\boldsymbol{X}\boldsymbol{M}^{-1}\boldsymbol{X}^T)^{-1}\boldsymbol{X}\boldsymbol{M}^{-1},$$

Algorithm 4 Alternating direction method of multipliers for solving (5)

Start with initial iterates $\boldsymbol{x}^0 = \boldsymbol{y}^0$ and step length μ.

for $t = 0, 1, 2 \ldots$ until converged **do**

Update $\boldsymbol{x}$ by (17):

$$(\mu \boldsymbol{I} + \boldsymbol{A})\boldsymbol{x}^{t+1} = \boldsymbol{d} + \mu \boldsymbol{y}^t - \boldsymbol{z}^t..$$

Update $\boldsymbol{y}$ using soft thresholding (18):

$$\boldsymbol{y}^{t+1} = \text{sign}(\boldsymbol{x}^{t+1} + \boldsymbol{z}^t/\mu) \max\{|\boldsymbol{x}^{t+1} + \boldsymbol{z}^t/\mu| - \lambda \boldsymbol{e}, \boldsymbol{0}\}$$

Update $\boldsymbol{z}$ using approximate dual ascent (19):

$$\boldsymbol{z}^{t+1} = \boldsymbol{z}^t + \mu(\boldsymbol{x}^{t+1} - \boldsymbol{y}^{t+1})$$

end for

where $\boldsymbol{M} := \mu \boldsymbol{I} + 2\gamma \boldsymbol{\Omega}$; computing this inverse only requires computing the inverse of $\boldsymbol{M}$ and the inverse of the $n \times n$ matrix $\boldsymbol{I} + 2\boldsymbol{X}\boldsymbol{M}^{-1}\boldsymbol{X}^T$.

Next $\boldsymbol{y}$ is updated by

$$\boldsymbol{y}^{t+1} = \underset{y \in \mathbf{R}^p}{\arg\min}\, L_\mu(\boldsymbol{x}^{t+1}, \boldsymbol{y}, \boldsymbol{z}^t) = \underset{\boldsymbol{y}}{\arg\min}\, \lambda \|\boldsymbol{y}\|_1 + \frac{\mu}{2}\|\boldsymbol{y} - \boldsymbol{x}^{t+1} - \boldsymbol{z}^t/\mu\|^2$$

That is, $\boldsymbol{y}^{t+1}$ is updated as the value of the soft thresholding operator of the ℓ_1-norm at $\boldsymbol{z}^t/\mu + \boldsymbol{x}^{t+1}$:

$$\boldsymbol{y}^{t+1} = S_\lambda(\boldsymbol{x}^{t+1} + \boldsymbol{z}^t/\mu) = \text{sign}(\boldsymbol{x}^{t+1} + \boldsymbol{z}^t/\mu) \max\{|\boldsymbol{x}^{t+1} + \boldsymbol{z}^t/\mu| - \lambda \boldsymbol{e}, \boldsymbol{0}\}. \tag{18}$$

Finally, the dual variable $\boldsymbol{z}$ is updated using the approximate dual ascent step

$$\boldsymbol{z}^{t+1} = \boldsymbol{z}^t + \mu(\boldsymbol{x}^{t+1} - \boldsymbol{y}^{t+1}). \tag{19}$$

This approach is summarized in Algorithm 4. It is well-known that the ADMM generates a sequence of iterates which converge linearly to an optimal solution of (14) under certain strong convexity assumptions on f and g and rank assumptions on $\boldsymbol{A}$ and $\boldsymbol{B}$, all of which are satisfied by our problem (16) when $\boldsymbol{\Omega}$ is positive definite (see, for example, Deng and Yin [2012]). As such, the sequence of iterates $\{\boldsymbol{x}^t, \boldsymbol{y}^t, \boldsymbol{z}^t\}$ generated by Algorithm 4 converges to a minimizer of $f(\boldsymbol{\beta})$; that is, $\boldsymbol{x}^t - \boldsymbol{y}^t \to \boldsymbol{0}$ and $f(\boldsymbol{x}^t), f(\boldsymbol{y}^t)$ converge linearly to the minimum value of f.

2.2 Computational Requirements

To motivate the use of our proposed proximal methods for the minimization of (5), we briefly sketch the per-iteration computational costs of each of our methods. We will see that for certain choices of regularization parameters, the number of floating point operations needed for each iteration scales linearly with the size of the data.

The most expensive step of both the proximal gradient method (Algorithm 2) and the accelerated proximal gradient method (Algorithm 3) is the evaluation of the gradient ∇f_1. Given a vector $\boldsymbol{\beta} \in \mathbf{R}^p$, the gradient at $\boldsymbol{\beta}$ is given by

$$\nabla f_1(\boldsymbol{\beta}) = \boldsymbol{A}\boldsymbol{\beta} = 2\left(\boldsymbol{X}^T\boldsymbol{X} + \gamma\boldsymbol{\Omega}\right)\boldsymbol{\beta} = 2\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta} + 2\gamma\boldsymbol{\Omega}\boldsymbol{\beta}.$$

The product $\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta}$ can be computed using $\mathcal{O}(np)$ floating point operations by computing $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta}$ and then $\boldsymbol{X}^T\boldsymbol{y}$. On the other hand, the product $\boldsymbol{\Omega}\boldsymbol{\beta}$ requires $\mathcal{O}(p^2)$ flops for unstructured $\boldsymbol{\Omega}$.

However, if we use a *structured* regularization parameter $\boldsymbol{\Omega}$ we can significantly decrease this computational cost. Consider the following examples:

- Suppose that $\boldsymbol{\Omega}$ is a diagonal matrix: $\boldsymbol{\Omega} = \text{Diag}(\boldsymbol{u})$ for some vector $\boldsymbol{u} \in \mathbf{R}^p_+$. Then the product $\boldsymbol{\Omega\beta}$ can be computed using $\mathcal{O}(p)$ flops:

$$(\boldsymbol{\Omega\beta})_i = u_i\beta_i.$$

  Moreover, we can estimate the Lipschitz constant $\|\boldsymbol{A}\|$ for use in choosing the step size α by

$$\|\boldsymbol{A}\| \leq 2\gamma\|\boldsymbol{\Omega}\| + 2\|\boldsymbol{X}\|_F^2 = 2\gamma\|\boldsymbol{u}\|_\infty + 2\|\boldsymbol{X}\|_F^2,$$

  which requires $\mathcal{O}(np)$ flops, primarily to compute the norm $\|\boldsymbol{X}\|_F^2$.

- If the use of diagonal $\boldsymbol{\Omega}$ is inappropriate, we could store $\boldsymbol{\Omega}$ in factored form $\boldsymbol{\Omega} = \boldsymbol{RR}^T$ where $\boldsymbol{R} \in \mathbf{R}^{p\times r}$, and r is the rank of $\boldsymbol{\Omega}$. In this case, we have

$$\boldsymbol{\Omega\beta} = \boldsymbol{R}(\boldsymbol{R}^T\boldsymbol{\beta}),$$

  which can be computed at a cost of $\mathcal{O}(rp)$ flops. Thus, if we use a low-rank parameter $\boldsymbol{\Omega}$, say $r \leq \mathcal{O}(n)$, we can compute the gradient using $\mathcal{O}(np)$ flops. Similarly, we can estimate the step size α using

$$\|\boldsymbol{A}\| \leq 2\|\boldsymbol{R}\|_F^2 + 2\|\boldsymbol{X}\|_F^2$$

  (computed at a cost of $\mathcal{O}(rp + np)$ flops).

In either case, using a diagonal $\boldsymbol{\Omega}$ or low-rank factored $\boldsymbol{\Omega}$, each iteration of the proximal gradient method or the accelerated proximal gradient method requires $\mathcal{O}(np)$ flops. Similar improvements can be made if $\boldsymbol{\Omega}$ is tridiagonal, banded, sparse, or otherwise nicely structured.

Similarly, the use of structured $\boldsymbol{\Omega}$ can lead to significant improvements in computational efficiency in our ADMM algorithm. The main computational bottleneck of this method is the solution of the linear system in the update of $\boldsymbol{x}$:

$$(\mu\boldsymbol{I} + \boldsymbol{A})\boldsymbol{x}^{t+1} = \boldsymbol{d} + \mu\boldsymbol{y}^t - \boldsymbol{z}^t.$$

Without taking advantage of the structure of $\boldsymbol{A}$, we can solve this system using a Cholesky factorization preprocessing step (at a cost of $\mathcal{O}(p^3)$ flops) and substitution to solve the resulting triangular systems (at a cost of $\mathcal{O}(p^2)$ flops per-iteration). However, we can use the Sherman-Morrison-Woodbury matrix inversion lemma to solve this system more efficiently using the structure of $\boldsymbol{A}$. Indeed, fix t and let $\boldsymbol{b} = \boldsymbol{d} + \mu\boldsymbol{y}^t - \boldsymbol{z}^t$. Then we update $\boldsymbol{x}$ by

$$\boldsymbol{x} = (\mu\boldsymbol{I} + \boldsymbol{A})^{-1}\boldsymbol{b}.$$

If $\boldsymbol{M} = \mu\boldsymbol{I} + 2\gamma\boldsymbol{\Omega}$ then we have

$$\begin{aligned}(\mu\boldsymbol{I} + \boldsymbol{A})^{-1} &= \left(\mu\boldsymbol{I} + 2\gamma\boldsymbol{\Omega} + 2\boldsymbol{X}^T\boldsymbol{X}\right)^{-1} = \left(\boldsymbol{M} + 2\boldsymbol{X}^T\boldsymbol{X}\right)^{-1} \\ &= \boldsymbol{M}^{-1} + 2\boldsymbol{M}^{-1}\boldsymbol{X}^T\left(\boldsymbol{I} + 2\boldsymbol{X}\boldsymbol{M}^{-1}\boldsymbol{X}^T\right)^{-1}\boldsymbol{X}^T\boldsymbol{M}^{-1}. \qquad (20)\end{aligned}$$

The matrix $\boldsymbol{I} + 2\boldsymbol{X}\boldsymbol{M}^{-1}\boldsymbol{X}^T$ is $n \times n$, so we may solve any linear system with this coefficient matrix using $\mathcal{O}(n^3)$ flops; a further $\mathcal{O}(n^2p)$ flops are needed to compute the coefficient matrix if given $\boldsymbol{M}^{-1}$. Thus, the main computational burden of this update step is the inversion of the matrix $\boldsymbol{M}$. As before, we want to choose $\boldsymbol{\Omega}$ so that we can exploit its structure. Consider the following cases.

		Diagonal	Rank r	Full rank
Proximal Gradient	∇f_1	$\mathcal{O}(np)$	$\mathcal{O}(rp + np)$	$\mathcal{O}(p^2)$
	$\|\boldsymbol{A}\|$	$\mathcal{O}(np)$	$\mathcal{O}(rp + np)$	$\mathcal{O}(p^2 \log p)$
ADMM	$(\mu\boldsymbol{I} + \boldsymbol{A})\boldsymbol{x} = \boldsymbol{b}$	$\mathcal{O}(n^3 + n^2 p)$	$\mathcal{O}(n^3 + n^2 p + r^2 p)$	$\mathcal{O}(p^3)$

Table 1: Upper bounds on floating point operation counts for most time consuming steps of each algorithm. For our (accelerated) proximal gradient method, these are the matrix-vector multiplication to compute the gradient ∇f_1 and the estimation of the Lipschitz constant using $\|\boldsymbol{A}\| \leq 2\left(\|\boldsymbol{\Omega}\| + \|\boldsymbol{X}\|_F^2\right)$ to define the step length; for ADMM, this is the solution of the linear system in the update of $\boldsymbol{x}$.

- If $\boldsymbol{\Omega} = \mathrm{Diag}(\boldsymbol{u})$ is diagonal, then $\boldsymbol{M}$ is also diagonal with

$$[\boldsymbol{M}^{-1}]_{ii} = \frac{1}{\mu + 2\gamma u_i}.$$

  Thus, we require $\mathcal{O}(p)$ flops to compute $\boldsymbol{M}^{-1}\boldsymbol{v}$ for any vector $\boldsymbol{v} \in \mathbf{R}^p$.

- On the other hand, if $\boldsymbol{\Omega} = \boldsymbol{R}\boldsymbol{R}^T$, where $\boldsymbol{R} \in \mathbf{R}^{p\times r}$, then we may use the Sherman-Morrison-Woodbury identity to compute $\boldsymbol{M}^{-1}$:

$$\boldsymbol{M}^{-1} = \frac{1}{\mu}\boldsymbol{I} - \frac{2}{\mu^2}\boldsymbol{R}\left(\boldsymbol{I} + \frac{2}{\mu}\boldsymbol{R}^T\boldsymbol{R}\right)^{-1}\boldsymbol{R}^T.$$

  Therefore, we can solve any linear system with coefficient matrix $\boldsymbol{M}$ at a cost of $\mathcal{O}(r^2 p)$ flops (for the formation and solution of the system with coefficient matrix $\boldsymbol{I} + \frac{2}{\mu}\boldsymbol{R}^T\boldsymbol{R}$).

In either case, we never actually compute the matrices $\boldsymbol{M}^{-1}$ and $(\mu\boldsymbol{I} + \boldsymbol{A})^{-1}$ explicitly. Instead, we update $\boldsymbol{x}$ as the solution of a sequence of linear systems and matrix-vector multiplications, at a total cost of $\mathcal{O}(n^2 p)$ flops (in the diagonal case) or $\mathcal{O}((r^2 + n^2)p)$ flops (in the factored case). Thus, if the number of observations n is much smaller than the number of features p, then the per-iteration computation scales roughly linearly with p. Table 2.2 summarizes these estimates of per-iteration computational costs for each proposed algorithm. Further, we should note that these bounds on per-iteration cost assume that the iterates $\boldsymbol{\beta}$ and $\boldsymbol{x}$ are dense; the soft-thresholding step of the proximal gradient algorithm typically induces $\boldsymbol{\beta}$ containing many zeros, suggesting that further improvements can be made by using sparse arithmetic.

2.3 Convergence of our block coordinate descent method

In this section, we investigate the convergence properties of our block coordinate descent method (Algorithm 1). Our two main results, Theorem 2.3 and Theorem 2.4, are specializations of standard results for alternating minimization algorithms; we include these results and their proofs for completeness.

We first note that the Lagrangian $L : \mathbf{R}^K \times \mathbf{R}^p \times \mathbf{R} \times \mathbf{R}^{k-1} \to \mathbf{R}$ of (2) is given by

$$L(\boldsymbol{\theta}, \boldsymbol{\beta}, \psi, \boldsymbol{v}) = \|\boldsymbol{Y}\boldsymbol{\theta} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + \gamma\boldsymbol{\beta}^T\boldsymbol{\Omega}\boldsymbol{b} + \lambda\|\boldsymbol{\beta}\|_1 + \psi(\boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta} - n) + \boldsymbol{v}^T\boldsymbol{U}\boldsymbol{\theta}, \tag{21}$$

where $\boldsymbol{U}^T = (\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta}_1, \boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta}_2, \ldots, \boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta}_{k-1})$. Our first technical lemma establishes when this Lagrangian function is convex in $(\boldsymbol{\theta}, \boldsymbol{\beta})$.

Lemma 2.2 *The Lagrangian function $L(\cdot,\cdot,\psi,\boldsymbol{v})$ defined by (21) is convex in $(\boldsymbol{\beta},\boldsymbol{\theta})$ for all $\boldsymbol{v} \in \mathbf{R}^{k-1}$ and $\psi \geq 0$.*

Proof: Fix $\psi \geq 0$ and $\boldsymbol{v} \in \mathbf{R}^{k-1}$. The functions $\gamma\boldsymbol{\beta}^T\boldsymbol{\Omega}\boldsymbol{\beta}$ and $\psi(\boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta} - n)$ are convex quadratic functions in $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ respectively. Similarly, $\lambda\|\boldsymbol{\beta}\|_1$ is a norm on $\mathbf{R}^p$ for all $\lambda > 0$ and $\boldsymbol{v}^T\boldsymbol{U}\boldsymbol{\theta}$ is linear in $\boldsymbol{\theta}$ and, therefore, both are convex in $(\boldsymbol{\theta},\boldsymbol{\beta})$. It remains to show that $\|\boldsymbol{Y}\boldsymbol{\theta} - \boldsymbol{X}\boldsymbol{\beta}\|^2$ is convex; indeed, in this case $L(\cdot,\cdot,\psi,\boldsymbol{v})$ is the sum of several convex functions and, hence, convex. To show that this is indeed the case, note that for any $(\boldsymbol{\theta},\boldsymbol{\beta})$, we have

$$\begin{aligned} 0 \leq \|\boldsymbol{Y}\boldsymbol{\theta} - \boldsymbol{X}\boldsymbol{\beta}\|^2 &= (\boldsymbol{Y}\boldsymbol{\theta} - \boldsymbol{X}\boldsymbol{\beta})^T(\boldsymbol{Y}\boldsymbol{\theta} - \boldsymbol{X}\boldsymbol{\beta}) \\ &= \boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta} - \boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{Y} + \boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta} \\ &= \begin{bmatrix} \boldsymbol{\theta} \\ \boldsymbol{\beta} \end{bmatrix}^T \begin{bmatrix} \boldsymbol{Y}^T\boldsymbol{Y} & -\boldsymbol{Y}^T\boldsymbol{X} \\ -\boldsymbol{X}^T\boldsymbol{Y} & \boldsymbol{X}^T\boldsymbol{X} \end{bmatrix} \begin{bmatrix} \boldsymbol{\theta} \\ \boldsymbol{\beta} \end{bmatrix}. \end{aligned}$$

This establishes that $\|\boldsymbol{Y}\boldsymbol{\theta} - \boldsymbol{X}\boldsymbol{\beta}\|^2$ is a convex quadratic function because the coefficient matrix

$$\begin{bmatrix} \boldsymbol{Y}^T\boldsymbol{Y} & -\boldsymbol{Y}^T\boldsymbol{X} \\ -\boldsymbol{X}^T\boldsymbol{Y} & \boldsymbol{X}^T\boldsymbol{X} \end{bmatrix}$$

is positive semidefinite. ■

We now provide our first convergence result, specifically, that Algorithm 1 generates a convergent sequence of function values.

Theorem 2.3 *Suppose that the sequence of iterates $\{(\boldsymbol{\theta}^t,\boldsymbol{\beta}^t)\}_{t=0}^{\infty}$ is generated by Algorithm 1. Then the corresponding sequence of objective function values $\{F(\boldsymbol{\theta}^t,\boldsymbol{\beta}^t)\}_{t=0}^{\infty}$ defined by*

$$F(\boldsymbol{\theta},\boldsymbol{\beta}) := \|\boldsymbol{Y}\boldsymbol{\theta} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + \gamma\boldsymbol{\beta}^T\boldsymbol{\Omega}\boldsymbol{\beta} + \lambda\|\boldsymbol{\beta}\|_1$$

is convergent.

Proof: Suppose that, after t iterations, we have iterates $(\boldsymbol{\theta}^t,\boldsymbol{\beta}^t)$ with objective function value $F(\boldsymbol{\theta}^t,\boldsymbol{\beta}^t)$. Recall that we obtain $\boldsymbol{\beta}^{t+1}$ as the solution of (5). Moreover, note that $\boldsymbol{\beta}^t$ is also feasible for (5). This immediately implies that

$$F(\boldsymbol{\theta}^t,\boldsymbol{\beta}^t) \geq F(\boldsymbol{\theta}^t,\boldsymbol{\beta}^{t+1}).$$

On the other hand, $\boldsymbol{\theta}^{t+1}$ is the solution of (3) with $\boldsymbol{\beta} = \boldsymbol{\beta}^{t+1}$. Therefore, we have

$$F(\boldsymbol{\theta}^t,\boldsymbol{\beta}^t) \geq F(\boldsymbol{\theta}^t,\boldsymbol{\beta}^{t+1}) \geq F(\boldsymbol{\theta}^{t+1},\boldsymbol{\beta}^{t+1}).$$

It follows that the sequence of function values $\{F(\boldsymbol{\theta}^t,\boldsymbol{\beta}^t)\}_{t=1}^{\infty}$ is monotonically nonincreasing. Moreover, the objective function $F(\boldsymbol{\theta},\boldsymbol{\beta})$ is nonnegative. Therefore, $\{F(\boldsymbol{\theta}^t,\boldsymbol{\beta}^t)\}_{t=1}^{\infty}$ is convergent as a monotonic bounded sequence. ■

We also have the following theorem, which establishes that every convergent subsequence of $\{(\boldsymbol{\theta}^t,\boldsymbol{\beta}^t)\}_{t=1}^{\infty}$ converges to a stationary point of (2).

Theorem 2.4 *Let $\{(\boldsymbol{\theta}^t, \boldsymbol{\beta}^t)\}_{t=1}^{\infty}$ be the sequence of points generated by Algorithm 1. Suppose that $\{(\boldsymbol{\theta}^{t_j}, \boldsymbol{\beta}^{t_j})\}_{j=1}^{\infty}$ is a convergent subsequence of $\{(\boldsymbol{\theta}^t, \boldsymbol{\beta}^t)\}_{t=1}^{\infty}$ with limit $(\boldsymbol{\theta}^*, \boldsymbol{\beta}^*)$. Then $(\boldsymbol{\theta}^*, \boldsymbol{\beta}^*)$ is a* stationary point *of* (2)*: $(\boldsymbol{\theta}^*, \boldsymbol{\beta}^*)$ is feasible for* (2) *and there exists $\psi^* \geq 0$ and $\boldsymbol{v}^* \in \mathbf{R}^{k-1}$ such that*

$$\mathbf{0} \in \partial L(\boldsymbol{\theta}^*, \boldsymbol{\beta}^*, \psi^*, \boldsymbol{v}^*),$$

where $\partial L(\boldsymbol{\theta}, \boldsymbol{\beta}, \psi, \boldsymbol{v})$ denotes the subdifferential of the Lagrangian function L with respect to the primal variables $(\boldsymbol{\theta}, \boldsymbol{\beta})$.

To prove Theorem 2.4, we first establish the following lemma, which establishes that the limit point $(\boldsymbol{\theta}^*, \boldsymbol{\beta}^*)$ minimizes F with respect to each primal variable with the other fixed; that is, $\boldsymbol{\theta}^*$ minimizes $F(\cdot, \boldsymbol{\beta}^*)$ and $\boldsymbol{\beta}^*$ minimizes $F(\boldsymbol{\theta}^*, \cdot)$.

Lemma 2.3 *Let $\{(\boldsymbol{\theta}^t, \boldsymbol{\beta}^t)\}_{t=1}^{\infty}$ be the sequence of points generated by Algorithm 1. Suppose that $\{(\boldsymbol{\theta}^{t_j}, \boldsymbol{\beta}^{t_j})\}_{j=1}^{\infty}$ is a convergent subsequence of $\{(\boldsymbol{\theta}^t, \boldsymbol{\beta}^t)\}_{t=1}^{\infty}$ with limit $(\boldsymbol{\theta}^*, \boldsymbol{\beta}^*)$. Then*

$$F(\boldsymbol{\theta}, \boldsymbol{\beta}^*) \geq F(\boldsymbol{\theta}^*, \boldsymbol{\beta}^*) \qquad \text{for all feasible } \boldsymbol{\theta} \in \mathbf{R}^K \tag{22}$$

$$F(\boldsymbol{\theta}^*, \boldsymbol{\beta}) \geq F(\boldsymbol{\theta}^*, \boldsymbol{\beta}^*) \qquad \text{for all feasible } \boldsymbol{\beta} \in \mathbf{R}^p. \tag{23}$$

Proof: We first establish (23). Consider $(\boldsymbol{\theta}^{t_j}, \boldsymbol{\beta}^{t_j})$. By our update step for $\boldsymbol{\beta}$, we note that

$$\boldsymbol{\beta}^{t_j} = \operatorname*{arg\,min}_{\boldsymbol{\beta} \in \mathbf{R}^p} F(\boldsymbol{\theta}^{t_j}, \boldsymbol{\beta}).$$

Thus, for all $j = 1, 2, \ldots$, we have $F(\boldsymbol{\theta}^{t_j}, \boldsymbol{\beta}) \geq F(\boldsymbol{\theta}^{t_j}, \boldsymbol{\beta}^{t_j})$ for all $\boldsymbol{\beta} \in \mathbf{R}^p$. Taking the limit as $j \to \infty$ and using the continuity of F establishes (23).

Next, note that, for every $j = 1, 2, \ldots,$, we have

$$\boldsymbol{\theta}^{t_j+1} = \operatorname*{arg\,min}_{\boldsymbol{\theta} \in \mathbf{R}^p} \left\{ F(\boldsymbol{\theta}, \boldsymbol{\beta}^{t_j}) : \boldsymbol{\theta}^T \boldsymbol{Y}^T \boldsymbol{Y} \boldsymbol{\theta} = n,\ \boldsymbol{\theta}^T \boldsymbol{Y}^T \boldsymbol{Y} \boldsymbol{\theta}_\ell = 0\ \forall \ell < k \right\}.$$

This implies that

$$F(\boldsymbol{\theta}, \boldsymbol{\beta}^{t_j}) \geq F(\boldsymbol{\theta}^{t_j+1}, \boldsymbol{\beta}^{t_j}) \geq F(\boldsymbol{\theta}^{t_j+1}, \boldsymbol{\beta}^{t_j+1}) \geq F(\boldsymbol{\theta}^{t_{j+1}}, \boldsymbol{\beta}^{t_{j+1}})$$

by the monotonicity of the sequence of function values and the fact that $t_j < t_j + 1 \leq t_{j+1}$. Taking the limit as $j \to \infty$ implies that $F(\boldsymbol{\theta}, \boldsymbol{\beta}^*) \geq F(\boldsymbol{\theta}^*, \boldsymbol{\beta}^*)$ and $F(\boldsymbol{\theta}^*) \geq F(\boldsymbol{\theta}^*, \boldsymbol{\beta}^*)$ for any feasible $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$. This completes the proof. ∎

We are now ready to prove Theorem 2.4.

Proof: (of Theorem 2.4). Taking the (sub)derivatives of L with respect to $(\boldsymbol{\theta}, \boldsymbol{\beta})$ shows that $(\boldsymbol{g}_{\boldsymbol{\theta}}, \boldsymbol{g}_{\boldsymbol{\beta}}) \in \partial L(\boldsymbol{\theta}, \boldsymbol{\beta}, \psi, \boldsymbol{v})$ if and only if

$$\boldsymbol{g}_{\boldsymbol{\theta}} = 2(1 + \psi)\boldsymbol{Y}^T \boldsymbol{Y} \boldsymbol{\theta} - 2\boldsymbol{Y}^T \boldsymbol{X} \boldsymbol{\beta} + \boldsymbol{U}^T \boldsymbol{v} \tag{24}$$

$$\boldsymbol{g}_{\boldsymbol{\beta}} \in 2(\boldsymbol{X}^T \boldsymbol{X} + \gamma \boldsymbol{\Omega})\boldsymbol{\beta} - 2\boldsymbol{X}^T \boldsymbol{Y} \boldsymbol{\theta} + \lambda \partial \|\boldsymbol{\beta}\|_1 \tag{25}$$

for all $\boldsymbol{v} \in \mathbf{R}^{k-1}$ and $\psi \geq 0$; note that the assumption $\psi \geq 0$ ensures that $L(\cdot, \cdot, \psi, \boldsymbol{v})$ is convex and, hence, subdifferentiable. Equation (23) implies that $\boldsymbol{\beta}^* = \arg\min_{\boldsymbol{\beta} \in \mathbf{R}^p} F(\boldsymbol{\theta}^*, \boldsymbol{\beta})$. Thus, by the first order necessary conditions for unconstrained convex optimization, we must have

$$\mathbf{0} \in \partial \left(\frac{1}{2} (\boldsymbol{\beta}^*)^T \boldsymbol{A} \boldsymbol{\beta}^* + \boldsymbol{d}^T \boldsymbol{\beta}^* + \lambda \|\boldsymbol{\beta}^*\|_1 \right) = 2(\boldsymbol{X}^T \boldsymbol{X} + \gamma \boldsymbol{\Omega})\boldsymbol{\beta}^* - 2\boldsymbol{X}^T \boldsymbol{Y} \boldsymbol{\theta}^* + \lambda \partial \|\boldsymbol{\beta}^*\|_1; \tag{26}$$

here $\partial\|\boldsymbol{\beta}\|_1$ denotes the subdifferential of the ℓ_1-norm at the point $\boldsymbol{\beta}$.

On the other hand, (22) implies

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}\in\mathbf{R}^K}{\arg\min} \left\{ \|\boldsymbol{Y\theta} - \boldsymbol{X\beta}^*\|^2 : \boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y\theta} = n,\ \boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y\theta}_\ell \ \forall\, \ell < k-1 \right\}. \tag{27}$$

Moreover, the problem (27) satisfies the linear independence constraint qualification. Indeed, the set of active constraint gradients $\{2\boldsymbol{Y}^T\boldsymbol{Y\theta}, 2\boldsymbol{Y}^T\boldsymbol{Y\theta}_1, \ldots, 2\boldsymbol{Y}^T\boldsymbol{Y\theta}_{k-1}\}$ is linearly independent for any feasible $\boldsymbol{\theta} \in \mathbf{R}^K$ by the $\boldsymbol{Y}^T\boldsymbol{Y}$-conjugacy of $\{\boldsymbol{\theta}, \boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{k-1}\}$. Therefore, there exist Lagrange multipliers ψ^*, $\boldsymbol{v}^*$ such that

$$\mathbf{0} = 2(1+\psi^*)\boldsymbol{Y}^T\boldsymbol{Y\theta}^* - 2\boldsymbol{Y}^T\boldsymbol{X\beta}^* + \boldsymbol{U}^T\boldsymbol{v}^* \tag{28}$$

by the first-order necessary conditions for constrained optimization (see Nocedal and Wright [2006, Theorem 12.1]). Moreover, an identical argument to that found in Appendix 4.1 shows that the Lagrange multiplier ψ^* must be nonnegative for the solution $\boldsymbol{\theta}^*$ of the linear system given by (28) to minimize (27). Combining (26) and (28) establishes that $\mathbf{0} \in \partial L(\boldsymbol{\theta}^*, \boldsymbol{\beta}^*, \psi^*, \boldsymbol{v}^*)$ as required. ■

3 Numerical Simulations

We next compare the performance of our proposed approaches with standard methods for penalized discriminant analysis in several numerical experiments. In particular, we compare the implementations of the block coordinate descent method Algorithm 1 where each discriminant direction $\boldsymbol{\beta}$ is updated using the proximal gradient method given by Algorithm 2 (SDAP), the accelerated proximal method given by Algorithm 3 (SDAAP), and the alternating direction method of multipliers given by Algorithm 4 (SDAD), with the Sparse Zero Variance Discriminant Analysis (SZVD) method proposed in Ames and Hong [2016], and the algorithm for solving the sparse optimal scoring problem proposed in Clemmensen et al. [2011]. All simulations were conducted using `R` version 3.2.3 and our heuristics are implemented in `R` as the package `accSDA`[3]. The runs on the benchmarking data sets were performed on a laptop with an i7-4800MQ processor clocked at 2.70GHz. The runs on the synthetic data sets were performed on a cluster where each node had an Intel Xeon E5-2660 v2 CPU clocked at 2.20GHz.

3.1 Classification of Spectral and Time Series Data

We first apply each of these methods to learn classification rules for the following data sets: the *Penicillium (Pen)* data set from [Clemmensen et al., 2007] of multi-spectral images of three Penicillium species that are almost visual indistinguishable ($p = 3542, K = 3, n = 36$, training sample size = 24, testing sample size = 12); *Electrocardiogram measurements (ECG)* of a 67-year old male taken on two dates, five days apart, before and after corrective cardiac surgery ($p = 136, K = 2, n = 884$, training size = 23, testing size = 861); food spectrogram observations of either Arabica or Robusta variants of instant coffee ($p = 286, K = 2, n = 56$, training size = 28, testing size = 28); food spectrogram observations of extra virgin olive oil originating from one of four countries ($p = 570, K = 4, n = 60$, training size = 30, testing size = 30). The final three data sets were obtained from the UC Riverside time series classification repository [Keogh et al., 2006]. We use each heuristic to obtain $q = K - 1$ sparse discriminant vectors and then perform nearest-centroid classification after projection onto the subspace spanned by these discriminant directions. The results of our experiments are summarized in Table 2.

[3]Available https://github.com/gumeo/accSDA

Time Series	Measures	SDAP	SDAAP	SDAD	SZVD	SDA
Pen	numErr	0	0	0	0	0
$p = 3542$	fracErr	0	0	0	0	0
$K = 3$	feats	633	667	60	1454	109
$n_{train} = 24$	fracFeats	0.18	0.19	0.02	0.41	0.03
$n_{test} = 12$	time	2502.11	134.99	337.37	6855.60	1248.12
ECG	numErr	23	86	21	20	42
$p = 136$	fracErr	0.03	0.10	0.02	0.02	0.05
$K = 2$	feats	18	16	25	21	16
$n_{train} = 23$	fracFeats	0.13	0.12	0.18	0.15	0.12
$n_{test} = 861$	time	61.28	15.77	21.20	4.90	1.45
Coffee	numErr	0	0	0	0	0
$p = 286$	fracErr	0	0	0	0	0
$K = 2$	feats	16	36	43	69	4
$n_{train} = 28$	fracFeats	0.06	0.13	0.15	0.24	0.01
$n_{test} = 28$	time	210.91	25.98	43.21	36.91	6.40
OliveOil	numErr	4	4	2	2	3
$p = 570$	fracErr	0.13	0.13	0.07	0.07	0.10
$K = 4$	feats	56	53	139	215	60
$n_{train} = 30$	fracFeats	0.10	0.09	0.24	0.38	0.11
$n_{test} = 30$	time	1308.03	271.20	393.49	1445.25	459.40

Table 2: Comparison of classification performance of benchmarking data. Each block reports the number of classification errors on out-of-sample testing observations (numErr), fraction of classification errors (fracErr), number of nonzero features used for classification (feats), fraction of nonzero features (fracFeat), and time (in seconds) needed to train the discriminant vectors (time).

The sparse discriminant analysis heuristics SDAP, SDAAP, SDAD, and SDA require training of the regularization parameters γ, $\mathbf{\Omega}$, and λ. In all experiments, we set $\gamma = 10^{-3}$ and $\mathbf{\Omega}$ to be the $p \times p$ identity matrix $\mathbf{\Omega} = \boldsymbol{I}$. We train the remaining parameter λ using N-fold cross validation. Specifically, we choose λ from a set of potential λ of the form $\bar{\lambda}/2^c$ for $c = 9, 8, 7, \ldots, -1, -2, -3$ and $\bar{\lambda}$ chosen so that the problem has nontrivial solution for all considered λ; we note that (7) has optimal solution given by $\boldsymbol{\beta}^* = \boldsymbol{A}^{-1}\boldsymbol{d}$ if we set $\lambda = 0$ and choose

$$\bar{\lambda} = \frac{(\boldsymbol{\beta}^*)^T\boldsymbol{d} - \frac{1}{2}(\boldsymbol{\beta}^*)^T\boldsymbol{A}\boldsymbol{\beta}^*}{\|\boldsymbol{\beta}^*\|_1}$$

so that there exists at least one solution $\boldsymbol{\beta}^*$ with value strictly less than zero. We pick as our regularization parameter the value of λ with fewest average number of misclassification errors over training-validation splits amongst all λ which yield discriminant vectors containing at most 15% nonzero entries. We set the number of folds $N = 5, 5, 7, 15$ for Pen, ECG, Coffee, and Olive oil data sets respectively. We terminate each proximal algorithm in the inner loop after 1000 iterations or a 10^{-5} suboptimal solution is obtained; the outer block coordinate descent loop is stopped after a maximum number of 250 iterations or a 10^{-3} suboptimal solution has been found. The augmented Lagrangian parameter $\mu = 2.5$ was used in all experiments in the ADMM method (SDAD).

We train any regularization parameters in SZVD using a similar procedure. In particular, we set the maximum value of the regularization parameter γ to be $\hat{\boldsymbol{\beta}}^T\boldsymbol{B}\hat{\boldsymbol{\beta}}/\|\hat{\boldsymbol{\beta}}\|_1$, where $\hat{\boldsymbol{\beta}}$ is the optimal

solution of the unpenalized SZVD problem and $\boldsymbol{B}$ is the sample between-class covariance matrix corresponding to the given data, and choose γ from an exponentially spaced grid using N-fold cross-validation; this approach is consistent with that in [Ames and Hong, 2016]. The number of folds N for each data set was identical to that in the SDA cross validation scheme described above. We select the value of γ which minimizes misclassification error amongst all sets of discriminant vectors with at most 35% nonzero entries; this acceptable sparsity threshold is chosen to be higher than that in the SDA experiments, due to the tendency of SZVD to misconverge to the trivial all-zero solution for large values of γ. We stop SZVD after a maximum of 1000 iterations or solution satisfying the stopping tolerance of 10^{-5} is obtained. We use the augmented Lagrangian penalty parameter $\beta = 2.5$ in SZVD in all experiments.

3.2 Gaussian data

We also performed similar simulations investigating efficacy of our heuristics for classification of Gaussian data. In each experiment, we generate data consisting of p-dimensional vectors from one of K multivariate Normal distributions. Specifically, we obtain training observations corresponding to the ith class by sampling 25 observations from the multivariate Normal distribution with mean $\boldsymbol{\mu_i} \in \mathbf{R}^p$ with entries indexed by $100(i-1), \ldots 100i$ equal to 0.7 and all remaining entries equal to 0, and covariance matrix $\boldsymbol{\Sigma} \in \mathbf{R}^{p \times p}$ constructed as follows.

- *Type 1 data:* in the first set of simulations, all features are correlated with $\Sigma_{ij} = r$ for all $i \neq j$ and $\Sigma_{ii} = 1$ for all i. We conduct the experiment for all $K \in \{2, 4\}, r \in \{0, 0.1, 0.5, 0.9\}$.
- *Type 2 data:* in the second set of simulations, $\boldsymbol{\Sigma}$ is a block diagonal matrix with 100×100 blocks. For each pair of indices (i, j) in the same block we set $\Sigma_{ij} = r^{|i-j|}$, and set $\Sigma_{ij} = 0$ otherwise. As before, we repeat the experiment for each $K \in \{2, 4\}, r \in \{0, 0.1, 0.5, 0.9\}$.

For each experiment, we sample 250 testing observations from each class in the same manner as the training data.

For each (K, r) pair we generate 20 data sets and use nearest centroid classification following projection onto the span of the discriminant directions to test the five LDA heuristics SDA, SDAP, SDAAP, SDAD, and SZVD. All input parameters are defined as in Section 3.1. We train any regularization parameters in the same fashion as in Section 3.1 using N-fold cross validation; we set a maximum fraction of nonzero features to 0.3 in the cross validation scheme. Tables 3 and 4 summarize the results of these experiments.

3.3 Multispectral X-ray images and $\boldsymbol{\Omega}$ of varying rank

To demonstrate empirically the run-time gain in having a non-full rank $\boldsymbol{\Omega}$ in the elastic net penalty, we choose to do pixelwise classification on multipsectral X-ray images, like presented in [Einarsson et al., 2017]. The multispectral X-ray images are scans of food items, where each pixel contains 128 measurements (channels) corresponding to attenuation of X-rays emitted at different wavelengths (See Fig. 1). The measurements in each pixel thus give us a profile for the material positioned at that pixel's location (See Fig. 2).

We start by preprocessing the scans similar to [Einarsson et al., 2017] in order to remove scanning artifacts and normalize the intensities between scans. We scale the measurements in each pixel by the 95% quantile of the corresponding 128 measurements instead of the maximum. This scaling approach is more robust.

We create our training data by manually selecting rectangular patches from six scans. We have three classes, namely *background*, *minced meat* and *foreign objects*. We further subsample the

Dataset	Measures	SDAP	SDAAP	SDAD	SZVD	SDA
$p = 500$	numErr	9.50 (11.52)	9.00 (11.63)	6.75 (5.95)	1.55 (2.14)	113.75 (13.04)
$r = 0$	fracErr	0.02 (0.02)	0.02 (0.02)	0.01 (0.01)	0.00 (0.00)	0.23 (0.03)
$K = 2$	feats	81.90 (36.86)	88.55 (36.04)	91.45 (30.93)	135.15 (25.09)	8.00 (0.00)
	fracFeats	0.16 (0.07)	0.18 (0.07)	0.18 (0.06)	0.27 (0.05)	0.02 (0.00)
	time	31.57 (2.46)	24.87 (1.43)	131.32 (2.18)	59.62 (12.12)	22.76 (1.28)
$p = 500$	numErr	6.85 (7.68)	7.35 (9.83)	7.95 (5.69)	0.75 (0.85)	117.50 (16.38)
$r = 0.1$	fracErr	0.01 (0.02)	0.01 (0.02)	0.02 (0.01)	0.00 (0.00)	0.24 (0.03)
$K = 2$	feats	83.60 (33.42)	89.85 (36.61)	77.25 (27.37)	139.85 (16.73)	8.00 (0.00)
	fracFeats	0.17 (0.07)	0.18 (0.07)	0.15 (0.05)	0.28 (0.03)	0.02 (0.00)
	time	49.73 (7.95)	27.46 (2.10)	132.95 (0.95)	60.76 (13.60)	23.18 (1.09)
$p = 500$	numErr	1.90 (2.90)	1.90 (3.21)	0.45 (0.83)	0.00 (0.00)	94.10 (25.14)
$r = 0.5$	fracErr	0.00 (0.01)	0.00 (0.01)	0.00 (0.00)	0.00 (0.00)	0.19 (0.05)
$K = 2$	feats	75.25 (29.70)	85.95 (36.60)	71.65 (15.16)	143.40 (14.25)	8.00 (0.00)
	fracFeats	0.15 (0.06)	0.17 (0.07)	0.14 (0.03)	0.29 (0.03)	0.02 (0.00)
	time	214.63 (27.34)	37.01 (3.90)	130.33 (1.80)	56.28 (8.51)	22.86 (1.43)
$p = 500$	numErr	4.80 (20.30)	4.80 (21.00)	0.00 (0.00)	0.00 (0.00)	19.65 (20.88)
$r = 0.9$	fracErr	0.01 (0.04)	0.01 (0.04)	0.00 (0.00)	0.00 (0.00)	0.04 (0.04)
$K = 2$	feats	51.70 (25.75)	75.65 (35.88)	94.00 (9.28)	164.65 (12.91)	8.00 (0.00)
	fracFeats	0.10 (0.05)	0.15 (0.07)	0.19 (0.02)	0.33 (0.03)	0.02 (0.00)
	time	194.13 (25.24)	28.80 (3.11)	135.31 (1.58)	60.82 (17.93)	23.96 (2.30)
$p = 500$	numErr	32.60 (14.55)	25.70 (16.15)	9.15 (6.11)	11.95 (7.06)	191.45 (24.85)
$r = 0$	fracErr	0.03 (0.01)	0.03 (0.02)	0.01 (0.01)	0.01 (0.01)	0.19 (0.02)
$K = 4$	feats	252.80 (68.10)	279.05 (66.02)	522.95 (14.24)	426.50 (42.82)	69.05 (0.22)
	fracFeats	0.44 (0.08)	0.48 (0.08)	0.74 (0.01)	0.67 (0.05)	0.14 (0.00)
	time	33.29 (2.46)	65.23 (3.02)	424.30 (4.71)	476.76 (90.16)	805.62 (37.15)
$p = 500$	numErr	44.35 (17.85)	33.80 (11.99)	5.30 (2.94)	28.55 (29.26)	245.05 (23.44)
$r = 0.1$	fracErr	0.04 (0.02)	0.03 (0.01)	0.01 (0.00)	0.03 (0.03)	0.25 (0.02)
$K = 4$	feats	262.80 (60.28)	296.20 (54.55)	540.55 (18.04)	381.80 (85.53)	69.00 (0.00)
	fracFeats	0.45 (0.07)	0.49 (0.06)	0.74 (0.02)	0.60 (0.09)	0.14 (0.00)
	time	87.99 (13.43)	71.69 (4.08)	430.38 (7.55)	467.19 (86.11)	787.50 (58.68)
$p = 500$	numErr	7.25 (5.95)	3.85 (3.07)	0.00 (0.00)	3.75 (7.30)	153.50 (28.41)
$r = 0.5$	fracErr	0.01 (0.01)	0.00 (0.00)	0.00 (0.00)	0.00 (0.01)	0.15 (0.03)
$K = 4$	feats	263.65 (21.94)	309.00 (21.22)	597.80 (23.67)	361.85 (68.06)	69.00 (0.00)
	fracFeats	0.45 (0.03)	0.51 (0.03)	0.76 (0.02)	0.58 (0.06)	0.13 (0.00)
	time	352.05 (29.60)	90.43 (8.89)	444.74 (10.22)	541.43 (144.59)	804.96 (52.84)
$p = 500$	numErr	3.40 (6.06)	3.40 (7.38)	0.00 (0.00)	25.00 (77.45)	4.70 (5.39)
$r = 0.9$	fracErr	0.00 (0.01)	0.00 (0.01)	0.00 (0.00)	0.02 (0.08)	0.00 (0.01)
$K = 4$	feats	223.70 (127.38)	193.40 (33.13)	796.20 (52.95)	355.25 (135.89)	69.00 (0.00)
	fracFeats	0.37 (0.17)	0.35 (0.05)	0.79 (0.01)	0.57 (0.14)	0.13 (0.00)
	time	343.42 (21.42)	72.44 (6.31)	510.91 (11.66)	446.70 (158.79)	785.33 (32.50)

Table 3: Results for Type 1 synthetic data. All results are listed in the format "mean (standard deviation)". In all experiments $n_{train} = 25K$ and $n_{test} = 250K$.

observations to have balanced number of observations, where the class *foreign objects* was under represented. In the end we have 521 observations per class, where each observation corresponds to a single pixel. This data was used to generate Fig. 2. For training we use 100 samples per class, and the rest is allocated to a final test set.

Dataset	Measures	SDAP	SDAAP	SDAD	SZVD	SDA
$p = 500$	numErr	9.50 (11.52)	9.00 (11.63)	6.75 (5.95)	1.55 (2.14)	113.75 (13.04)
$r = 0$	fracErr	0.02 (0.02)	0.02 (0.02)	0.01 (0.01)	0.00 (0.00)	0.23 (0.03)
$K = 2$	feats	81.90 (36.86)	88.55 (36.04)	91.45 (30.93)	135.15 (25.09)	8.00 (0.00)
	fracFeats	0.16 (0.07)	0.18 (0.07)	0.18 (0.06)	0.27 (0.05)	0.02 (0.00)
	time	28.84 (2.26)	25.38 (1.41)	132.87 (1.27)	62.44 (12.85)	23.16 (1.12)
$p = 500$	numErr	8.95 (10.77)	7.40 (8.64)	8.20 (5.86)	3.15 (5.56)	111.05 (15.26)
$r = 0.1$	fracErr	0.02 (0.02)	0.01 (0.02)	0.02 (0.01)	0.01 (0.01)	0.22 (0.03)
$K = 2$	feats	93.95 (48.19)	94.75 (34.30)	86.45 (23.69)	132.80 (28.13)	8.00 (0.00)
	fracFeats	0.19 (0.10)	0.19 (0.07)	0.17 (0.05)	0.27 (0.06)	0.02 (0.00)
	time	28.36 (2.37)	24.67 (1.42)	133.24 (1.13)	60.39 (10.04)	23.02 (0.99)
$p = 500$	numErr	25.50 (13.74)	19.95 (9.98)	22.55 (8.11)	12.80 (7.88)	116.55 (18.14)
$r = 0.5$	fracErr	0.05 (0.03)	0.04 (0.02)	0.05 (0.02)	0.03 (0.02)	0.23 (0.04)
$K = 2$	feats	72.25 (36.20)	81.85 (27.66)	75.25 (18.45)	136.45 (35.10)	8.00 (0.00)
	fracFeats	0.14 (0.07)	0.16 (0.06)	0.15 (0.04)	0.27 (0.07)	0.02 (0.00)
	time	31.44 (2.76)	26.17 (1.70)	128.89 (1.97)	72.51 (17.74)	24.74 (1.15)
$p = 500$	numErr	99.50 (12.53)	99.15 (11.48)	110.25 (11.64)	116.15 (16.75)	133.75 (16.25)
$r = 0.9$	fracErr	0.20 (0.03)	0.20 (0.02)	0.22 (0.02)	0.23 (0.03)	0.27 (0.03)
$K = 2$	feats	59.65 (34.47)	66.85 (31.87)	92.10 (22.81)	132.90 (22.27)	8.00 (0.00)
	fracFeats	0.12 (0.07)	0.13 (0.06)	0.18 (0.05)	0.27 (0.04)	0.02 (0.00)
	time	51.49 (7.20)	32.63 (2.43)	116.43 (4.31)	120.82 (20.72)	30.57 (2.10)
$p = 500$	numErr	32.60 (14.55)	25.70 (16.15)	9.15 (6.11)	11.95 (7.06)	191.45 (24.85)
$r = 0$	fracErr	0.03 (0.01)	0.03 (0.02)	0.01 (0.01)	0.01 (0.01)	0.19 (0.02)
$K = 4$	feats	252.80 (68.10)	279.05 (66.02)	522.95 (14.24)	426.50 (42.82)	69.05 (0.22)
	fracFeats	0.44 (0.08)	0.48 (0.08)	0.74 (0.01)	0.67 (0.05)	0.14 (0.00)
	time	36.22 (3.18)	65.45 (3.14)	427.83 (7.06)	443.24 (77.38)	705.49 (67.77)
$p = 500$	numErr	43.00 (19.26)	28.45 (15.98)	10.75 (4.72)	18.58 (9.36)	196.65 (25.32)
$r = 0.1$	fracErr	0.04 (0.02)	0.03 (0.02)	0.01 (0.00)	0.02 (0.01)	0.20 (0.03)
$K = 4$	feats	253.35 (74.60)	299.60 (83.63)	537.20 (24.01)	419.53 (45.13)	69.00 (0.00)
	fracFeats	0.44 (0.08)	0.50 (0.08)	0.74 (0.01)	0.66 (0.05)	0.14 (0.00)
	time	36.90 (2.38)	67.01 (2.33)	430.11 (7.51)	475.23 (83.24)	742.81 (46.20)
$p = 500$	numErr	89.80 (20.86)	84.70 (22.96)	62.55 (10.24)	125.90 (93.66)	230.80 (25.47)
$r = 0.5$	fracErr	0.09 (0.02)	0.08 (0.02)	0.06 (0.01)	0.13 (0.09)	0.23 (0.03)
$K = 4$	feats	259.95 (52.78)	281.00 (47.87)	560.85 (20.69)	383.35 (68.22)	69.05 (0.22)
	fracFeats	0.46 (0.07)	0.48 (0.07)	0.76 (0.02)	0.60 (0.09)	0.14 (0.00)
	time	41.62 (4.27)	72.10 (2.82)	415.59 (5.75)	483.74 (121.77)	854.26 (51.16)
$p = 500$	numErr	368.10 (37.14)	366.20 (29.92)	453.15 (32.76)	502.65 (29.87)	391.85 (18.91)
$r = 0.9$	fracErr	0.37 (0.04)	0.37 (0.03)	0.45 (0.03)	0.50 (0.03)	0.39 (0.02)
$K = 4$	feats	189.25 (82.78)	269.55 (96.00)	867.25 (46.11)	416.20 (55.93)	69.25 (0.44)
	fracFeats	0.33 (0.12)	0.44 (0.13)	0.92 (0.02)	0.63 (0.06)	0.13 (0.00)
	time	103.07 (6.48)	100.21 (7.28)	400.24 (11.51)	1034.96 (148.44)	1096.82 (118.89)

Table 4: Results for Type 2 synthetic data. All results are listed in the format "mean (standard deviation)". In all experiments $n_{train} = 25K$ and $n_{test} = 250K$.

This process yields 128 variables per observation, but in order to get more spatially consisted classification, we also include data from the pixels located above, to the right, below and to the left of the observed pixel. Thus we have $p = 5 \cdot 128 = 640$ variables per observation. The measurements corresponding to our observation are thus indexed according to spatial and spectral position, i.e. observation $\mathbf{x}_i$ has measurements x_{ijk}, where $j \in \{0, 1, 2, 3, 4\}$ indicates which pixel

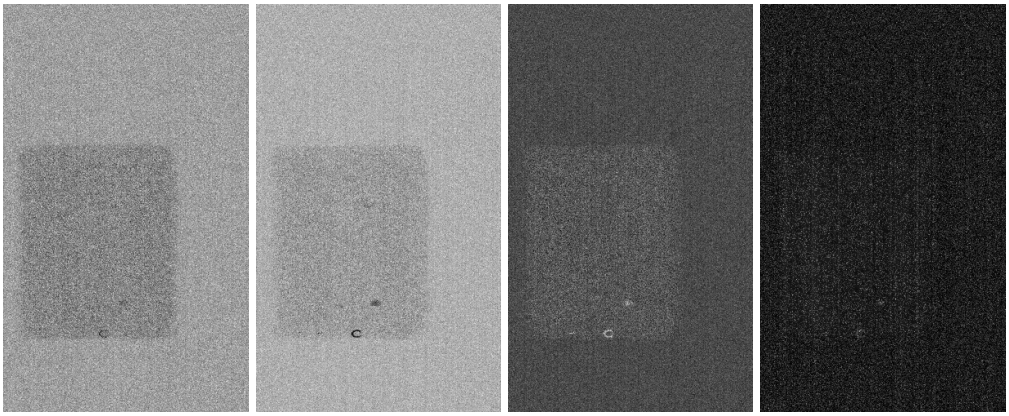


Figure 1: Grayscale images of different channels from a minced meat sample generated with a multispectral X-ray scanner after all preprocessing. From left to right are channels 2, 20, 50 and 100. The contrast decreases the higher we go in the channels and the variation in the measurements increases. Some foreign objects can be seen as small black dots.

the measurement belongs to (*center, above, right, bottom, left*), and $k \in \{1, 2, ..., 128\}$ indicates which channel. We can impose priors according to these relationships of the measurements in the $\mathbf{\Omega}$ regularization matrix. We assume that the errors should vary smoothly spatially and thus impose a Matérn covariance structure on $\mathbf{\Omega}^{-1}$ [Matérn, 2013].

$$C_\nu(d) = \sigma^2 \frac{2^{1-v}}{\Gamma(\nu)} \left(\sqrt{2\nu} \frac{d}{\rho} \right)^\nu K_\nu \left(\sqrt{2\nu} \frac{d}{\rho} \right) \tag{29}$$

The Matérn covariance structure 29 is governed by the distance d between measurements. In 29, Γ refers to the gamma function and K_ν is the modified Bessel function of the second kind. For this example we assume that all parameters are 1, except that ν is 0.5. We further assume that the distance between measurements x_{ijk} and $x_{ij'k'}$ from observation i is the Euclidean distance between the points (x_j, y_j, z_k) and $(x_{j'}, y_{j'}, z_{k'})$, where $x_j, y_j, x_{j'}, y_{j'} \in \{-1, 0, 1\}$ and $z_k, z_{k'} \in \{1, 2, ..., 128\}$. The distance is thus the same as in the image grid (center, top, bottom, left, right pixel location), and z-dimension corresponds to the channel.

For demonstrating the effect that lower rank has on runtime we factorize $\mathbf{\Omega}$ via Singular Value Decomposition, and select the first r singular vectors and singular values for constructing a low-rank approximation to $\mathbf{\Omega}$. We use a stopping tolerance of 10^{-5} and a maximum of 1000 iterations for the inner loop using the accelerated proximal algorithm, and a stopping tolerance of 10^{-4} and maximum 1000 iterations for the outer block-coordinate loop. The regularization parameter for the l_1-norm is selected as 10^{-3}, and 10^{-1} for the Tikhonov regularizer. We do this 10 times for each r and present the average runtime in Fig. 3 and the average accuracy in Fig. 4. There is a clear linear trend for the increase in runtime. We also estimate the accuracy for a diagonal $\mathbf{\Omega}$ with the same regularization parameters and get an accuracy of 0.923, which is approximately the same that an $\mathbf{\Omega}$ of rank 300 achieves. We supplied the same parameters to the function sda from the package `sparseLDA` and it took 486 seconds to run and achieved an accuracy of 0.923.

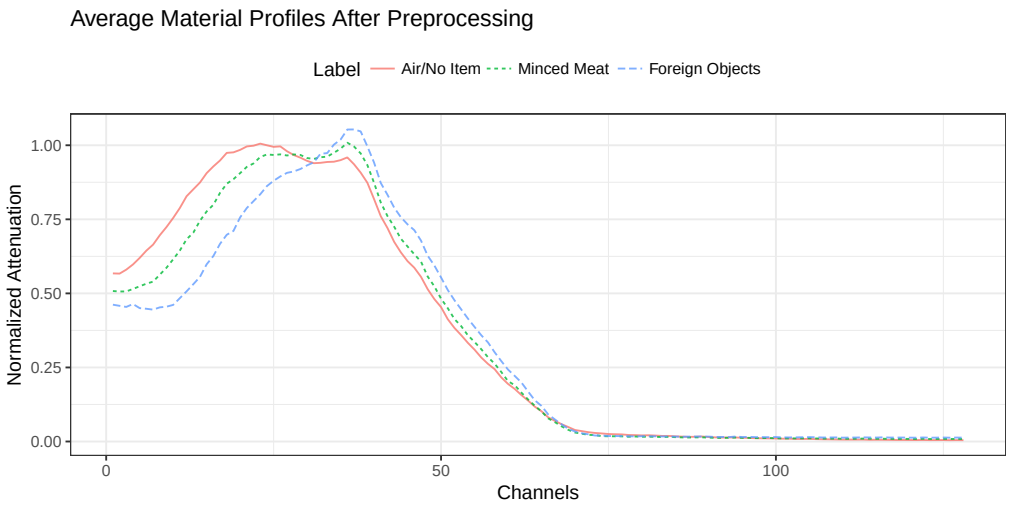


Figure 2: Profiles of materials seen in Fig. 1 over the 128 channels. The profile for each type of material, displayed here, is averaged over 500 pixels.

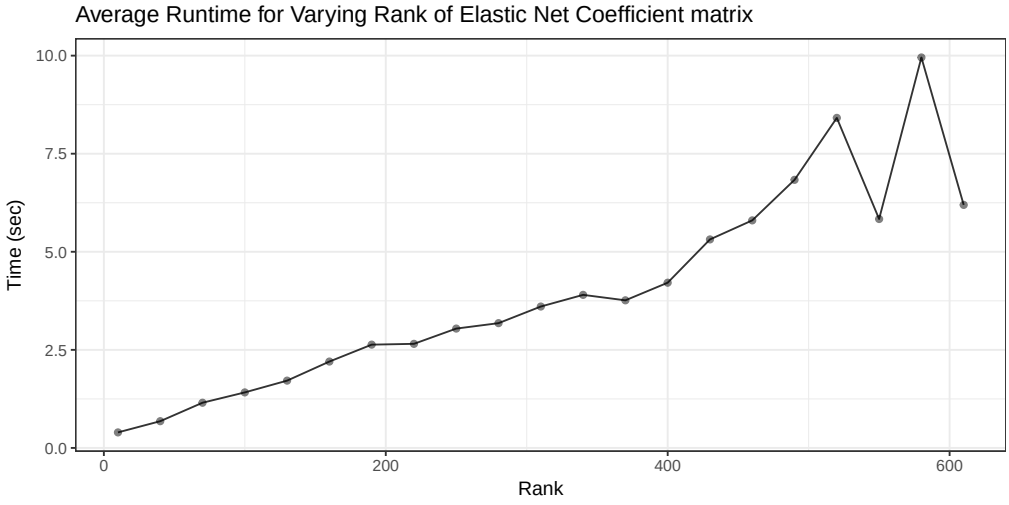


Figure 3: Average running time over 10 runs when the rank of the elastic net coefficient matrix $\mathbf{\Omega}$ is varied using Accelerated Proximal Gradient.

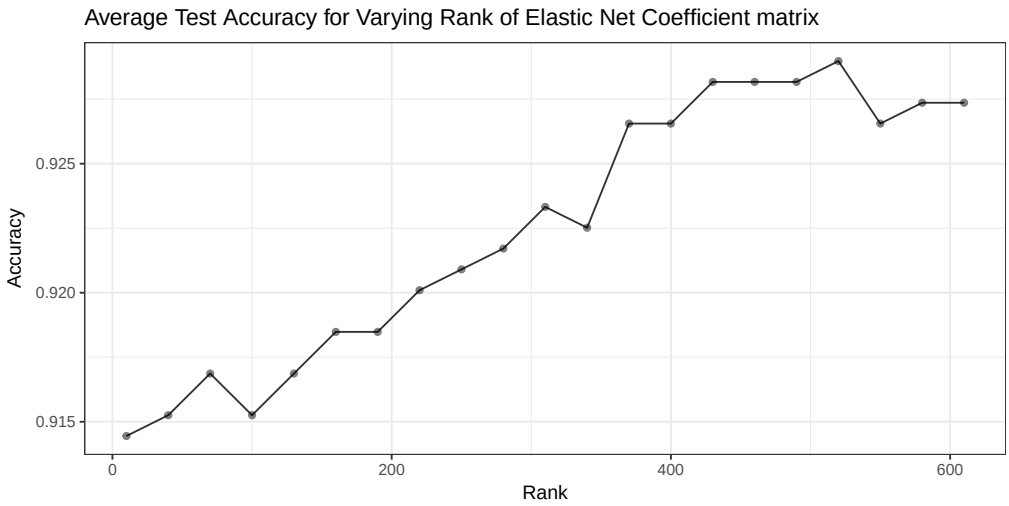


Figure 4: Average test accuracy over 10 runs when the rank of the elastic net coefficient matrix $\mathbf{\Omega}$ is varied using Accelerated Proximal Gradient.

3.4 Commentary

Our proximal methods for sparse discriminant analysis provide an improvement over the existing SDA approach in terms of classification error in almost all experiments, while we see a significant improvement in terms of computational resources used by the accelerated proximal gradient method (SDAAP) and ADMM (SDAD) over SDA. This improvement in run-time is most significant when applied to the Penicillium data set. This is not a coincidence. The per-iteration complexity of these methods is on the order of $\mathcal{O}(p)$ floating point operations per-iteration, compared to $\mathcal{O}(p^3)$ of the classical SDA method. We see this improvement is most significant when p is large, as it is for the Penicillium data set, where the per-iteration cost of $\mathcal{O}(p^3)$ flops is prohibitive. This improvement is not observed as acutely when p is small; in fact, SDA exhibits the shortest run-times for both the ECG and Coffee data sets, where p is the smallest. It is important to note that the slow convergence of the proximal gradient method (SDAP) without acceleration yields significantly longer run-times despite the improved per-iteration cost. We should also note that our use of cross validation causes significant variation in the performance of our heuristics for the ECG data set. This is because the trained discriminant vectors are sensitive to the split in the validation process, as the training set can become unbalanced.

4 Conclusion

We have proposed new algorithms for solving the sparse optimal scoring problem for high-dimensional linear discriminant analysis. These methods, based on block coordinate descent and proximal operator evaluations, provide significant improvement over existing approaches for solving the SOS problem, in terms of efficiency and scalability, in the case that specially structured Tikhonov regularization is employed in the SOS formulation; for example, the computational resources required for each iteration scales linearly with the dimension of the data if either a diagonal or low-rank

matrix is used. Moreover, we establish that any convergent subsequence of iterates generated by one of our algorithms converges to a stationary point. Finally, numerical simulation establishes that our approach provides an improvement over existing methods for sparse discriminant analysis in terms of both quality of solution and run-time.

These results present several exciting avenues for future research. Although we focus primarily on the solution of the optimal scoring problem under regularization in the form of a generalized elastic net penalty, our approach should translate immediately to formulations with any nonsmooth convex penalty function. That is, the framework provided by Algorithm 1 can be applied to solve the SOS problem (2) obtained by applying an arbitrary convex penalty to the objective of the optimal scoring problem (1). The resulting optimization problem can be approximately solved by alternately minimizing with respect to the score vector $\boldsymbol{\theta}$ using the formula (4) and with respect to the discriminant vector $\boldsymbol{\beta}$ by solving a modified version of (5). The proximal methods outlined in this paper can be applied to minimize with respect to $\boldsymbol{\beta}$ if the regularization function is convex, however it is unlikely that the computational resources necessary for this minimization will scale as favorably as with the generalized elastic net penalty. On the other hand, the convergence analysis presented in Section 2.3 extends immediately to this more general regularization framework. Of particular interest is the modification of this approach to provide means of learning discriminant vectors for data containing ordinal labels, data containing corrupted or missing observations, and semi-supervised settings.

Finally, although the results found in Section 2.3 provide compelling evidence that any convergent subsequent of iterates generated by our block coordinate descent approach must converge to a stationary point, it is still unclear what conditions ensure that the sequence of iterates is convergent, or at what rate these subsequences converge; further study is required to better understand the convergence properties of these algorithms. Similarly, despite the empirical evidence provided in Section 3, it is unknown what conditions ensure that data is classifiable or when data can have its dimension reduced using sparse optimal scoring and, more generally, linear discriminant analysis. Extensive consistency analysis is needed to determine theoretical classification error rates for distinguishing random variables drawn from distinct distributions.

4.1 Acknowledgements

We are grateful to Mingyi Hong for his helpful comments and suggestions. B. Ames was supported in part by University of Alabama Research Grant RG14678. G. Einarsson's PhD scholarship is funded by the Lundbeck foundation and the Technical University of Denmark. S. Atkins was part of the University Scholars Program at the University of Alabama while this research was conducted.

Appendix A. Proof of Lemma 2.1

To begin, we note that (3) has trivial solution $\boldsymbol{\theta} = \boldsymbol{e}$ for every $\boldsymbol{\beta} = \boldsymbol{\beta}^t \in \mathbf{R}^p$. Indeed, $\boldsymbol{Y}\boldsymbol{e} = \boldsymbol{e}$ by the structure of the indicator matrix $\boldsymbol{Y}$ and $\boldsymbol{X}^T\boldsymbol{e} = \mathbf{0}$ because our data has been centered to have sample mean equal to $\mathbf{0}$. This implies that (3) has the hidden constraint $\boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{e} = 0$. Therefore, we may reformulate (3) as

$$\min_{\boldsymbol{\theta}\in\mathbf{R}^k} \left\{ \|\boldsymbol{Y}\boldsymbol{\theta} - \boldsymbol{X}\boldsymbol{\beta}\|^2 : \boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta} = n,\ \boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{e} = 0,\ \boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta}_\ell = 0\ \ell < k-1 \right\}. \tag{30}$$

We wish to show that (30) has optimal solution $\hat{\boldsymbol{\theta}}$ given by

$$\hat{\boldsymbol{\theta}} = \frac{\boldsymbol{w}}{\sqrt{\boldsymbol{w}^T \boldsymbol{D} \boldsymbol{w}}}, \tag{31}$$

where $\boldsymbol{w} = (\boldsymbol{I} - \boldsymbol{Q}_k \boldsymbol{Q}_k^T \boldsymbol{D}) \boldsymbol{D}^{-1} \boldsymbol{Y}^T \boldsymbol{X} \boldsymbol{\beta}$.

To do so, note that (30) satisfies the linear independence constraint qualification because the set of constraint gradients $\{2\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta}, 2\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{e}, 2\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta}_1, \ldots, 2\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta}_{k-1}\}$ is linearly independent. Moreover, the optimal value of (30) is bounded below by 0. Therefore, (30) has global minimizer, $\hat{\boldsymbol{\theta}}$, which must satisfy the Karush-Kuhn-Tucker conditions, i.e., there exists $\boldsymbol{v} \in \mathbf{R}^k$, $\psi \in \mathbf{R}$ such that

$$2\boldsymbol{Y}^T\boldsymbol{Y}\hat{\boldsymbol{\theta}} - 2\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} + 2\frac{\psi}{n}\boldsymbol{Y}^T\boldsymbol{Y}\hat{\boldsymbol{\theta}} + 2\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{Q}_k\boldsymbol{v} = \mathbf{0}. \tag{32}$$

We consider the following two cases.

First, suppose that $\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} \notin \text{range}\left(\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{Q}_k\right)$. Rearranging (32) yields

$$\hat{\boldsymbol{\theta}} = \frac{n}{n+\psi}\left(\boldsymbol{Y}^T\boldsymbol{Y}\right)^{-1}\left(\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{Q}_k\boldsymbol{v}\right). \tag{33}$$

We choose the dual variables ψ and $\boldsymbol{v}$ so that $\hat{\boldsymbol{\theta}}$ is feasible for (30). It is easy to see that the conjugacy constraints are equivalent to $\boldsymbol{Q}_k^T\boldsymbol{Y}^T\boldsymbol{Y}\hat{\boldsymbol{\theta}} = \mathbf{0}$, which holds if and only if

$$\mathbf{0} = \boldsymbol{Q}_k^T(\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{Q}_k\boldsymbol{v}) = \boldsymbol{Q}_k^T\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{Q}_k^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{Q}_k\boldsymbol{v} = \boldsymbol{Q}_k^T\boldsymbol{Y}^T\boldsymbol{\beta} - n\boldsymbol{v},$$

where the last equality follows from the fact that $\boldsymbol{e}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{e} = \boldsymbol{\theta}_i^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\theta}_i = n$ for all $i = 1, 2, \ldots, k-1$. It follows immediately that

$$\boldsymbol{v} = \frac{1}{n}\boldsymbol{Q}_k^T\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta}. \tag{34}$$

Substituting (34) into (33) yields

$$\begin{aligned}
\hat{\boldsymbol{\theta}} &= \frac{n}{n+\psi}\left((\boldsymbol{Y}^T\boldsymbol{Y})^{-1}\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} - \frac{1}{n}\boldsymbol{Q}_k\boldsymbol{Q}_k^T\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta}\right) \\
&= \frac{1}{n+\psi}\left(\boldsymbol{D}^{-1}\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{Q}_k\boldsymbol{Q}_k^T\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta}\right) \\
&= \frac{1}{n+\psi}\left(\boldsymbol{I} - \boldsymbol{Q}_k\boldsymbol{Q}_k^T\boldsymbol{D}\right)\boldsymbol{D}^{-1}\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta},
\end{aligned} \tag{35}$$

where we choose $\psi \in \mathbf{R}$ so that $\hat{\boldsymbol{\theta}}^T\boldsymbol{D}\hat{\boldsymbol{\theta}} = 1$:

$$\psi = \pm\sqrt{\boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{Y}\boldsymbol{D}^{-1}\left(\boldsymbol{I} - \boldsymbol{Q}_k\boldsymbol{Q}_k^T\boldsymbol{D}\right)^T\boldsymbol{D}\left(\boldsymbol{I} - \boldsymbol{Q}_k\boldsymbol{Q}_k^T\boldsymbol{D}\right)\boldsymbol{D}^{-1}\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta}} - n. \tag{36}$$

To complete the argument, note that

$$\|\boldsymbol{Y}\hat{\boldsymbol{\theta}} - \boldsymbol{X}\boldsymbol{\beta}\|^2 = n \mp 2s\boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{Y}(\boldsymbol{D}^{-1} - \boldsymbol{Q}_k\boldsymbol{Q}_k)\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta},$$

where $s = 1/\sqrt{\boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{Y}\boldsymbol{D}^{-1}\left(\boldsymbol{I} - \boldsymbol{Q}_k\boldsymbol{Q}_k^T\boldsymbol{D}\right)^T\boldsymbol{D}\left(\boldsymbol{I} - \boldsymbol{Q}_k\boldsymbol{Q}_k^T\boldsymbol{D}\right)\boldsymbol{D}^{-1}\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta}}$. Note further that the matrix $\boldsymbol{Y}\boldsymbol{Q}_k\boldsymbol{Q}_k^T\boldsymbol{Y}^T$ has decomposition

$$\boldsymbol{Y}\boldsymbol{Q}_k\boldsymbol{Q}_k^T\boldsymbol{Y}^T = \boldsymbol{Y}\boldsymbol{e}\boldsymbol{e}^T\boldsymbol{Y}^T + \boldsymbol{Y}\boldsymbol{\theta}_1\boldsymbol{\theta}_1^T\boldsymbol{Y} + \cdots \boldsymbol{Y}\boldsymbol{\theta}_{k-1}\boldsymbol{\theta}_{k-1}^T\boldsymbol{Y}.$$

The conjugacy of the columns of $\boldsymbol{Q}_k$ implies that $\boldsymbol{Y}\boldsymbol{Q}_k\boldsymbol{Q}_k^T\boldsymbol{Y}^T$ has eigenvectors $\boldsymbol{Y}\boldsymbol{\theta}_1, \ldots \boldsymbol{Y}\boldsymbol{\theta}_{k-1}$, and $\boldsymbol{Y}\boldsymbol{e}$, each with eigenvalue n. Therefore, the matrix $\boldsymbol{Y}(\boldsymbol{D}^{-1} - \boldsymbol{Q}_k\boldsymbol{Q}_k)\boldsymbol{Y}^T$ is positive semidefinite and thus $\|\boldsymbol{Y}\boldsymbol{\theta} - \boldsymbol{X}\boldsymbol{\beta}\|^2$ is minimized by $\hat{\boldsymbol{\theta}}$ with $\psi = +s$.

Second, suppose that $\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} \in \text{range}\left(\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{Q}_k\right)$. This implies that there exists some $\boldsymbol{v} \in \mathbf{R}^k$ such that

$$\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} = \boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{Q}_k\boldsymbol{v}.$$

Substituting into the objective of (30), we see that

$$\begin{aligned}\|\boldsymbol{Y}\boldsymbol{\theta} - \boldsymbol{X}\boldsymbol{\beta}\|^2 &= \boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{\beta} - 2\boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta} \\ &= n - 2\boldsymbol{\theta}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{Q}_k\boldsymbol{v} + \boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta} \\ &= n + \boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta}\end{aligned}$$

for every feasible solution $\boldsymbol{\theta}$ of (30). This implies that every feasible solution of (30) is also optimal in this case. In particular, $\hat{\boldsymbol{\theta}}$ given by (35) is feasible for (30) and, therefore, optimal.

References

Z. Allen-Zhu and L. Orecchia. Linear coupling: An ultimate unification of gradient and mirror descent. *arXiv preprint arXiv:1407.1537*, 2014.

B. Ames and M. Hong. Alternating direction method of multipliers for penalized zero-variance discriminant analysis. *Computational Optimization and Applications*, 64(3):725–754, 2016.

A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends®in Machine Learning*, 3(1):1–122, 2011.

S. Bubeck, Y. T. Lee, and M. Singh. A geometric alternative to Nesterov's accelerated gradient descent. *arXiv preprint arXiv:1506.08187*, 2015.

T. Cai and W. Liu. A direct estimation approach to sparse linear discriminant analysis. *Journal of the American Statistical Association*, 106(496), 2011.

L. Clemmensen, M. Hansen, J. Frisvad, and B. Ersbøll. A method for comparison of growth media in objective identification of penicillium based on multi-spectral imaging. *Journal of Microbiological Methods*, 69(2):249–255, 2007.

L. Clemmensen, T. Hastie, D. Witten, and B. Ersbøll. Sparse discriminant analysis. *Technometrics*, 53(4), 2011.

W. Deng and W. Yin. On the global and linear convergence of the generalized alternating direction method of multipliers. *Journal of Scientific Computing*, pages 1–28, 2012.

G. Einarsson, J. N. Jensen, R. R. Paulsen, H. Einarsdottir, B. K. Ersbøll, A. B. Dahl, and L. B. Christensen. Foreign object detection in multispectral x-ray images of food items using sparse discriminant analysis. In *Scandinavian Conference on Image Analysis*, pages 350–361. Springer, 2017.

J. Fan and Y. Fan. High dimensional classification using features annealed independence rules. *Annals of Statistics*, 36(6):2605–2637, 2008.

N. Flammarion and F. Bach. From averaging to acceleration, there is only a step-size. *arXiv preprint arXiv:1504.01577*, 2015.

G. H. Golub and C. F. Van Loan. *Matrix Computations*, volume 3. JHU Press, 2013.

T. Hastie, R. Tibshirani, and A. Buja. Flexible discriminant analysis by optimal scoring. *Journal of the American Statistical Associationstical association*, 89(428):1255–1270, 1994.

T. Hastie, R. Tibshirani, and J. J. H. Friedman. *The Elements of Statistical Learning*. Springer New York, 2013.

T. Hastie, R. Tibshirani, and M. Wainwright. *Statistical Learning with Sparsity: the Lasso and Generalizations*. CRC Press, 2015.

E. Keogh, X. Xi, L. Wei, and C. A. Ratanamahatana. The UCR time series classification/clustering homepage, 2006.

L. Lessard, B. Recht, and A. Packard. Analysis and design of optimization algorithms via integral quadratic constraints. *SIAM Journal on Optimization*, 26(1):57–95, 2016.

Q. Mai and H. Zou. A note on the connection and equivalence of three sparse linear discriminant analysis methods. *Technometrics*, 55(2):243–246, 2013.

Q. Mai and H. Zou. Multiclass sparse discriminant analysis. *arXiv preprint arXiv:1504.05845v1*, 2015.

Q. Mai, M. Yuan, and H. Zou. A direct approach to sparse discriminant analysis in ultra-high dimensions. *Biometrika*, 99:29–42, 2012.

B. Matérn. *Spatial variation*, volume 36. Springer Science & Business Media, 2013.

Y. Nesterov. A method of solving a convex programming problem with convergence rate $o(1/k^2)$. In *Soviet Mathematics Doklady*, volume 27, pages 372–376, 1983.

Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical programming*, 103(1): 127–152, 2005.

Y. Nesterov. Gradient methods for minimizing composite functions. *Mathematical Programming*, 140(1):125–161, 2013.

J. Nocedal and S. Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

B. O'Donoghue and E. Candes. Adaptive restart for accelerated gradient schemes. *Foundations of Computational Mathematics*, 15(3):715–732, 2015.

N. Parikh and S. P. Boyd. Proximal algorithms. *Foundations and Trends in optimization*, 1(3): 127–239, 2014.

J. Shao, Y. Wang, X. Deng, and S. Wang. Sparse linear discriminant analysis by thresholding for high dimensional data. *The Annals of Statistics*, 39(2):1241–1265, 2011.

W. Su, S. Boyd, and E. Candes. A differential equation for modeling nesterov's accelerated gradient method: Theory and insights. In *Advances in Neural Information Processing Systems*, pages 2510–2518, 2014.

R. Tibshirani, T. Hastie, B. Narasimhan, and G. Chu. Class prediction by nearest shrunken centroids, with applications to dna microarrays. *Statistical Science*, pages 104–117, 2003.

P. Tseng. On accelerated proximal gradient methods for convex-concave optimization. submitted to. *SIAM Journal on Optimization*, 2008.

D. M. Witten and R. Tibshirani. Penalized classification using Fisher's linear discriminant. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(5):753–772, 2011.

M. Wu, L. Zhang, Z. Wang, D. Christiani, and X. Lin. Sparse linear discriminant analysis for simultaneous testing for the significance of a gene set/pathway and gene selection. *Bioinformatics*, 25:1145–1151, 2008.

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.

APPENDIX B

Semi-Supervised Sparse Discriminant Analysis

The following manuscript has been submitted to the journal *Statistics and Probability Letters*.

Semi-Supervised Sparse Discriminant Analysis

Gudmundur Einarsson[a,*], Rasmus R. Paulsen[a], Brendan P. Ames[b], Line K. H. Clemmensen[a]

[a] *Department for Applied Mathematics and Computer Science, Technical University of Denmark*
[b] *Department of Mathematics, University of Alabama*

Abstract

We present a semi-supervised extension of sparse discriminant analysis by addition of a graph based regularization term. We show empirically how addition of unlabeled data can improve accuracy by a significant margin and produce more consistent classifiers. We also demonstrate how the regularizer can help in the presence of concept drift, where the distribution of data conditioned on labels changes continuously.

Keywords: Semi-Supervised, Sparse, Classification, Statistical Learning

1. Introduction

Sparse or penalized discriminant analysis (SDA/PDA) Clemmensen et al. (2011); Witten & Tibshirani (2011) is a popular tool to perform supervised classification, notably in the case of more features than observations, i.e., $p \gg n$ problems. Using an l_1-norm regularizer in the model formulation ensures that variable selection is performed in the model optimization process which gives leverage for the user to interpret the non-zero parameters. Incorporation of an l_1-norm regularizer is influenced by the Lasso Tibshirani (1996); Chen et al. (2001).

Our contribution consists of emperically examining the benefits of using jointly a sparse and semi-supervised regularizer in linear discriminant analysis

*Corresponding author
Email address: guei@dtu.dk (Gudmundur Einarsson)

(LDA), where we approach the problem via sparse optimal scoring Hastie et al. (1994); Clemmensen et al. (2011). We apply the semi-supervised approach to an ECG time-series data set Chen et al. (2015) and we demonstrate how the method works on an artificial example of concept drift Tsymbal (2004) and examine how adding unlabeled samples can improve classification accuracy.

Implementations of various SDA approaches exist as software packages for several programming languages Clemmensen & contributions by Max Kuhn (2016); Sjöstrand et al. (2012), e.g. the `sparseLDA` package for `R`. However, none of these packages include methods to directly incorporate unlabeled data, i.e., a semi-supervised approach. There exist generic semi-supervised approaches like the Yarowsky algorithm Yarowsky (1995) and various Expectation-Maximization (EM) based approaches that combine a supervised and unsupervised model through a generative mixture model Nigam et al. (2000). These methods iteratively train a classifier on labeled data, and then the methods assign hard or soft labels to unlabeled data; this procedure is continued until convergence, where the new labels are used as input for the next classifier to be trained. Such a solution is implemented as the `R`-package `RSSL`, Krijthe & Loog (2015); Krijthe (2016).

The disadvantage of many of the generic approaches is that the classifier potentially needs to be trained many times to reach convergence, which might make the classifier scale poorly w.r.t. the dimensions of the training set. These methods cannot be scaled via parallelization, i.e., the training set for a new iteration is dependent on the prediction from the last. This is similar to the variable selection problem which is classically solved with stepwise regression, where in turn we need to train models multiple times. But as stated above, variable selection and model fitting can be performed simultaneously by applying an l_1-norm regularizer. We can use a similar idea with the unlabeled part of our training set. That is, we can encode the information of the unlabeled training data into a regularizer and regularize w.r.t. the patterns in the unlabeled data during training of the classifier. One approach for constructing such a regularization term for LDA is proposed in Cai et al. (2007). The main contribution

of Cai et al Cai et al. (2007) is the construction of a Tikhonov regularization term which borrows ideas from spectral dimensionality reduction and spectral clustering Belkin & Niyogi (2001); He & Niyogi (2003); Ng et al. (2001), where they construct a k-nearest neighbour graph on labeled and unlabeled data in feature space. This is a type of manifold assumption initially introduced by Zhu et al Zhu et al. (2003) in their label propagation method.

LDA is not only a classifier, but also a supervised dimensionality reduction method, where the data is projected into a lower-dimensional space with the discriminant vectors found with the method. We enforce a local consistency assumption Zhou et al. (2003), where unlabeled data that is close in the original feature space should be close in the lower-dimensional space as well. This is also a manifold assumption, where we seek to project manifolds embedded in the feature-space to a lower-dimensional representation Belkin & Niyogi (2004); Sindhwani et al. (2005).

1.1. Contributions

Our contributions can be summarized as follows:

- We combine sparse and unsupervised regularizers for the optimal scoring formulation of LDA.
- We show emperically that addition of unlabeled samples can increase classification accuracy.
- We demonstrate that an unsupervised regularizer can achieve near perfect classification accuracy in an artifical example of concept drift in presence of redundant variables.

2. Why semi-supervised learning?

There are several reasons for why one might consider using a semi-supervised learning approach. In a practical setting the main reason is probably the cost ratio of acquiring an observation versus obtaining a label for it. The usage of

unlabeled data requires assumptions, and in order to safely use unlabeled data, these assumptions need to be verified. A compelling example illustrating why checking assumptions is important is a binary classification problem where the data forms two obvious clusters, but the distribution of data conditioned on the labels is bimodal. That is, the labels do not coincide with the underlying mixtures present in the data distribution Krijthe & Loog (2014); Ben-David et al. (2008). Using a supervised approach with few data points in that case could give adequate results, while using many unlabeled samples could drive the classifier towards generating a classification boundary that separates the two clusters and heavily underperforms the supervised approach. This could very well happen in a real scenario, where there is a hidden latent variable which creates the presence of mixtures in the data distribution, which do not coincide with the target labels.

This phenomena has been studied further where bounds for improvement of a semi-supervised learner have been created based on the assumptions that are made Singh et al. (2009). In our case this relates to a proper selection of the parameters used for constructing the graph for the semi-supervised regularizer. These are the parameters that control the number of edges in the graph constructed with the observations as nodes, e.g. k in a k-nearest neighbour graph. If these parameters are too large then we have conflicting goals in our minimization problem. The optimal scoring part of SDA aims at separating different classes, while a semi-supervised regularizer based on a fully connected graph aims at projecting everything as close as possible. On the other end of the spectrum, with too low value of the graph parameter, we have a sparse graph that has almost no connections.. A sparse graph can be helpful if the number of variables is high and we do not have enough observations to find the true cluster structure in the data. But if there is a true cluster structure that coincides with the labeled data that we have, then we should try to find the appropriate graph parameter accordingly.

Another reason for using a semi-supervised approach is concept-drift Tsymbal (2004). The distribution of the data conditioned on the labels can change

with time. One particular example is a spam mail classifier. E-mail spam is not static, it changes with time, but it forms natural clusters according to the topic of the spam Wang et al. (2013). Instead of putting effort into labeling new spam and training a classifier, one can use a semi-supervised approach to try to generalize with respect to the new types of spam or reduce the number of samples needed to be labeled. If one would have used sparse discriminant analysis for the original problem, and then later semi-supervised sparse discriminant analysis for creating a new classifier, one can compare the discriminant vectors obtained from both models to identify if any new variables have become more important for this classification task. This can also aid in prioritizing which new samples we should produce labels for.

3. Semi-supervised regularizers

In this section we will explain notation, the sparse optimal scoring problem, and how we can add semi-supervised regularizers to the problem formulation.

3.1. Notation

For a given $n \times p$ data matrix $\boldsymbol{X}$, n is the number of observations and p is the number of features. The $n \times K$ matrix $\boldsymbol{Y}$ is an indicator matrix of class membership, where K is the number of classes in the data set. If observation i belongs to class j, then element Y_{ij} in $\boldsymbol{Y}$ is 1 and the other values in the same row are 0. We use $\boldsymbol{\theta} \in \mathbf{R}^{K\times 1}$ to denote the scoring vector, which is used to transform the categorical representation of the classes into a numerical representation. The desired discriminant vector, used to project data from the original feature space to a lower-dimensional representation, is deonoted by $\boldsymbol{\beta} \in \mathbf{R}^p$. The regularization parameters in the classifier are denoted as λ_1 and λ_2, where λ_1 is associated with the l_1-norm regularizer and λ_2 with the semi-supervised regularizer. The semi-supervised regularizer consists of a Tikhonov regularization matrix $\boldsymbol{\Omega} \in \mathbf{R}^{p\times p}$ and we refer to $\lambda_2\boldsymbol{\beta}^T\boldsymbol{\Omega}\boldsymbol{\beta}$ as the semi-supervised regularizer. We refer to $\lambda_1\|\boldsymbol{\beta}\|_1$ as a sparse regularizer or l_1-norm regularizer.

Observation i belonging to class j is represented as the pair $(c_j, \boldsymbol{x}_i)$. For a graph $\mathcal{G}$, $\boldsymbol{A}$ is the adjacency matrix, $\boldsymbol{D}$ is the degree matrix and $\boldsymbol{L} := \boldsymbol{D} - \boldsymbol{A}$ is the graph Laplacian.

3.2. Sparse Optimal Scoring

The sparse optimal scoring problem is formulated as follows:

$$\underbrace{\underset{\boldsymbol{\theta}_k \in \mathbf{R}^K, \boldsymbol{\beta}_k \in \mathbf{R}^p}{\arg\min} \underbrace{\|\boldsymbol{Y}\boldsymbol{\theta}_k - \boldsymbol{X}\boldsymbol{\beta}_k\|_2^2}_{\text{Optimal Scoring}} + \lambda_2 \boldsymbol{\beta}_k^T \boldsymbol{\Omega} \boldsymbol{\beta}_k + \lambda_1 \|\boldsymbol{\beta}_k\|_1}_{\text{Sparse OS}}$$

$$\begin{aligned} \text{s.t. } \tfrac{1}{n}\boldsymbol{\theta}_k^T \boldsymbol{Y}^T \boldsymbol{Y} \boldsymbol{\theta}_k &= 1, \\ \boldsymbol{\theta}_k^T \boldsymbol{Y}^T \boldsymbol{Y} \boldsymbol{\theta}_\ell &= 0 \; \forall \ell < k, \end{aligned} \tag{1}$$

In Equation 1 the constraints apply to both the optimal scoring and the sparse optimal scoring formulation. The constraints are spherical which makes the problem non-convex. When we solve this problem we seek the discriminant vectors $\boldsymbol{\beta}_i$, which we can then use to project the data from feature space to a lower-dimensional representation. The traditional approach is to solve this minimization problem with a block-update algorithm, where one first solves for $\boldsymbol{\theta}$, then $\boldsymbol{\beta}$ and iterate until convergance. That way we can find the first $(\boldsymbol{\theta}_1, \boldsymbol{\beta}_1)$ pair, then we continue in a similar manner to find the successive pairs until we have found the maximum number of pairs, $K - 1$, or the desired number of pairs.

Clemmensen et al Clemmensen & contributions by Max Kuhn (2016) show that for a given $\boldsymbol{\beta}$ one can find $\boldsymbol{\theta}$ in polynomial time. For a given $\boldsymbol{\theta}$ the problem formulation is an elastic net problem and can be solved with the LARS-EN algoritm Zou & Hastie (2005). We however approach the optimization from the point of proximal gradient methods and alternating direction method of multipliers, using the soft thresholding operator to deal with the sparse regularizer in the same manner as anonymous Atkins et al. (2017).

3.3. Adding unlabeled data

We need to construct $\boldsymbol{\Omega}$ from the second term in Equation 1 using unlabeled data. We begin by assuming that we have a data set D, which can be split into a labeled part, D_1, and unlabeled part, D_2, where we have n_1 labeled samples and n_2 unlabeled samples and $n := n_1 + n_2$.

$$\begin{aligned} D_1 &= \{(c_{k_1}, \boldsymbol{x}_1), (c_{k_2}, \boldsymbol{x}_2), \ldots, (c_{k_n}, \boldsymbol{x}_{n_1})\} \\ D_2 &= \{\boldsymbol{x}_{n_1+1}, \boldsymbol{x}_{n_1+2}, \ldots, \boldsymbol{x}_{n_1+n_2}\} \end{aligned} \tag{2}$$

Now we ignore the labels in D_1 and construct a graph on all the data points. As we stated in section 1, we want points that are near each other in the feature space to remain close after the projection, so we construct the graph based on proximity of data ponts. We explore two ways to construct the graph, what we need in the end is the graph Laplacian.

First we consider the weighted undirected graph defined by a Gaussian kernel, where the edge weight between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in the adjacency matrix A is $\boldsymbol{A}_{ij} := \exp(-\gamma||\boldsymbol{x}_i - \boldsymbol{x}_j||_2^2)$. For a suitable choice of γ, the closest points get a weight close to 1 and then it decays exponentially the further we go away. Now we can construct the semi-supervised regularization term as:

$$\sum_{ij} (\boldsymbol{x}_i\boldsymbol{\beta} - \boldsymbol{x}_j\boldsymbol{\beta})\boldsymbol{A}_{ij} = 2\boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{L}\boldsymbol{X}\boldsymbol{\beta}.$$

This has been simplified on the right hand side using matrix notation, where L is the graph Laplacian and $\boldsymbol{X}$ is the data matrix containing both labeled and unlabeled feature vectors. So our semi-supervised regularization matrix $\boldsymbol{\Omega}$ can be defined as $2\boldsymbol{X}^T\boldsymbol{L}\boldsymbol{X}$.

The second graph we consider for the data is a k-nearest neighbour graph. The only difference compared to the approach with the Gaussian kernel, is how we construct the adjacency matrix A, which we initilize as the $n \times n$ zero matrix. For a given data point $\boldsymbol{x}_i$ we find its k nearest neighbours $\boldsymbol{x}_{i_1}, \ldots, \boldsymbol{x}_{i_k}$ and assign the value 1 to $\boldsymbol{A}_{ij}$ and $\boldsymbol{A}_{ji}$ if $\boldsymbol{x}_j$ is one of the k neighbours. We assign the value 1 to $\boldsymbol{A}_{ij}$ and $\boldsymbol{A}_{ji}$ to make sure that the graph is undirected.

Although the way we define the regularization terms only lies in the adjacency matrix, the main differences are in the way we compute them. For the k-nearest neighbour graph we might want to consider using approximate k-nearest neighbour if 2^p is greater or on the scale of n, where p is the number of features. For the adjacency matrix created with the Gaussian kernel we can vectorize the calculations and do them on a GPU. Normally we would want to inspect our data to make an educated guess of good values for the parameters γ and k, so we only need to calculate the regularization matrix once. We could also cross-validate over these parameters, but since we already have one parameter per regularizer to cross-validate over, we would end up with three parameters to cross-validate, which we want to avoid. Another difference is the memory footprint in the intermediate computations of $\mathbf{\Omega}$. For the Laplacian of a k-nearest neighbour graph and a low k we can use a sparse matrix, but for the Gaussian kernel the graph Laplacian matrix is dense.

The presence of outliers causes significant differences in the behavior of these two regularization approaches. If the distribution of pairwise distances of data points has a heavy right tail, then the Gaussian kernel could potentially give a low weight to edges connected to outliers while the outliers would still always be *equally* connected to other data points compared to other points if we use a k-nearest neighbour graph. If the number of variables in our data is high, then using the Gaussian kernel we risk creating many connections between clusters, thus not being able to distinguish clearly between them. Since we are focusing on data where the number of features is usually higher or on the scale of the number of variables, we will use a k-nearest neighbour graph for the following experiments. An implementation of a graph defined with a Gaussian kernel is also supplied in the `semiSDA`-package.

4. Experiments and results

4.1. ECG time-series data

The ECG data was donwloaded from the UCR time-series classification archive Chen et al. (2015). The Electrocardigram measurements (ECG) are from a 67 year old male taken prior to and after corrective cardiac surgery, so we have a binary classification task. Each observation has 136 features.

The data set consists of 884 observations. For the experiment on this data set we initially randomly sample 150 observations from each class and use as a test set to evaluate the performance. The rest of the data we sample at random to generate balanced training sets with 2,3,4,...,9 observations per class. For each training set we then sample a set without labels were we try 100, 150 and 200 observations per class. After we obtain the data, we normalize it by centering to zero and scale the features to have unit variance. We construct the regularization term using a k-nearest neighbour graph with $k = 1$, so a sparse graph. We compare the results to a classifier with a ridge regularization term to inspect the gain of adding unlabeled data, the regularization parameters for the classifier with the ridge regularizer were found with cross-validation. The l_1 regularization parameters were choosen such that the resulting discriminant vector would have 25-45% non-zero values and it was kept fixed through the experiments. Each configuration of number of labeled and unlabeled samples was run 500 times and a new data set is randomly sampled for each run from the observations that were not in the initial test set. The average accuracy of the classifiers is summarized in Figure 1.

The regularization parameter for the semi-supervised regularizer was chosen as the smallest non-zero eigenvalue of the graph Laplacian scaled by $\frac{1}{2}$, which was at least greater than 10^{-10}. Currently we do not have any theoretical justification on why that might be a good choice. Since k is only equal to 1, we likely have multiple components in the corresponding graph and thus multiplicity of higher than one for eigenvalue zero. A choice of a much higher regularizer parameter increases the likelihood of obtaining a trivial solution.

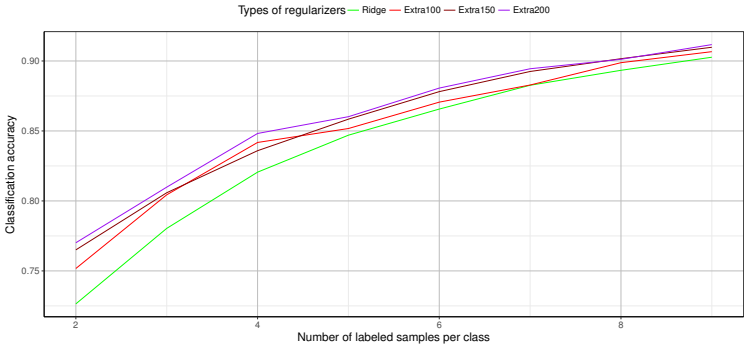


Figure 1: Average accuracy of classifiers over 500 runs. The experiment was conducted with four different regularizers summarized in the legend above the graph. Extra{} denotes the number of unlabeled samples used for constructing the semi-supervised regularizer. There is a consistent gain of 3-5% accuracy using 200 unlabeled samples for the regularizer (red line).

We also tried higher values of k, but that did not improve the results as much. Low signal to noise ratio in a high dimensional space increases the likelihood that edges in the graph are between samples in different classes, thus making the regularizer preserve locality between all samples and favoring the trivial solution. Having more unlabeled data can certainly help in that regard, if it is the case, that the process by which the data from the two classes is generated, is truly different.

We performed spectral clustering on the data and visualized the eigenvector corresponding to the smallest non-zero eigenvalue with 400 observations. We performed this repeatedly, where we sampled equal amount of data from each class, with varying k to construct the graph. We did not observe that consistent clusters were formed in nearly all cases.

4.2. Artificial concept drift example

The data set used for this example is shown in Figure 2. We imagine that an initial data set is sampled and labeled. Then a continuous process starts where the mean of the classes shift along the first variable, and crosses the optimal separation boundary. The means of the classes come from a multivariate normal distribution with the covariance matrix specified in Equation 3 and means $(-2, -1)$ and $(2, 1)$.

$$\Sigma = \begin{bmatrix} 0.02 & -0.01 \\ -0.01 & 0.02 \end{bmatrix} \tag{3}$$

We explore adding redundant variables to the data set, where each new redundant variable is sampled from a univariate normal distribution with mean zero and variance 0.02. With a fixed number of observations, 100 for the training set and 2020 for the unlabeled data, we want to find the maximum k for increasing number of redundant variables which yields a classifier that has at least 95% accuracy. We also perform cross-validation with a validation set sampled from the same distribution as the training set such that we find a classifier with around $\frac{2}{2+n_R}$ proportion of non-zero coefficients in the discriminant vector, where n_R is the number of redundant variables. We show how the optimal k falls with the number of added redundant variables. The results can be seen in Figure 3.

This shows that although there is a clear cluster structure in the data, additional noisy variables essentially make it harder to construct a good k-nearest neighbour graph, because it becomes more and more likely that the neighbours do not belong to the same class. The accuracy of the classifiers for the experiment where we find the optimal k for the number of classes is depicted in Figure 4. We can see that the performance starts to drop when we have 13 redundant variables. If we use the ridge regularizer, the accuracy is 62.3%, so even when we have 18 redundant variables, we still see improvement in accuracy.

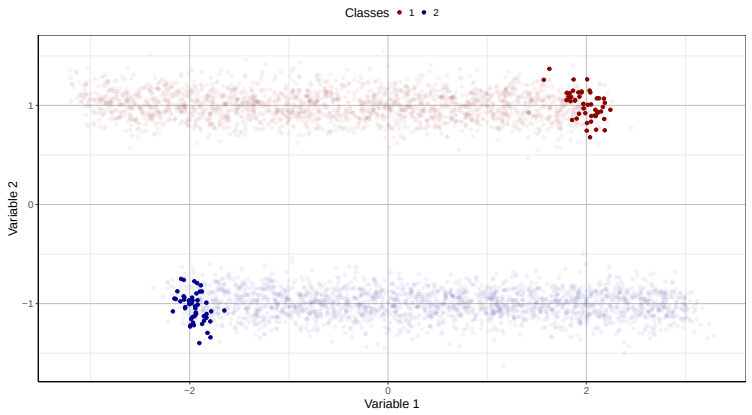


Figure 2: First two dimensions of the concept drift data set. The more transparent points correspond to unlabeled points, while the points with a solid fill correspond to training data. After the initial data set is sampled there is continuous drift along the first variable, where the second variable is more important for class separation.

5. Discussion

There are many ways to utilize unlabeled data, but the most critical part of pursuing such an endeavor is to safely check the assumptions that one makes. As we saw in the artificial concept drift example, the choice of k for the k-nearest neighbour graph is heavily influenced by the number of variables. The best choice of k is both dependent on the noise and the number of variables. For the case of concept drift it is possible to extend the method to include multiple prototypes per class, similar to Clemmensen et. al Clemmensen et al. (2011).

Another idea to make the construction of the graph more robust is to run sparse discriminant analysis on the labeled data with a ridge regularizer and

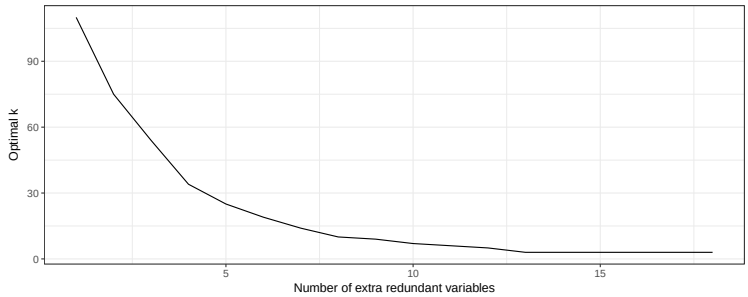


Figure 3: The optimal value of k falling with the number of extra redundant variables.

only use the variables corresponding to non-zero coefficients in the discriminant vector to estimate the distance between observations. This however only works under the assumption that the sub-space spanned by these variables contains the features that represent the variation that we are after. Our simple artificial concept drift data set only had drift within the same variable, but the mean of the data can potentially change w.r.t. other variables, which we would like our model to select. This could very well happen in the case of spam e-mail, where people that generate spam would start to use words that are not present in the old data, and the data used to train the classifier consist of word frequency within e-mail.

Another thing to remember is that we can use any method to construct the graph for the semi-supervised regularizer, we are not constrained to use the raw original observations. Thus if the user has any particular assumptions about some variables, he can define a different distance metric, or he can also potentially extract new features for constructing the graph.

One thing to note is that we used the smallest non-zero eigenvalue of the graph Laplacian, scaled by $\frac{1}{2}$, as the regularization parameter for the semi-supervised regularizer in all examples. Empirically we have discovered that it works well, but as stated before, we do not have any theoretical justification for

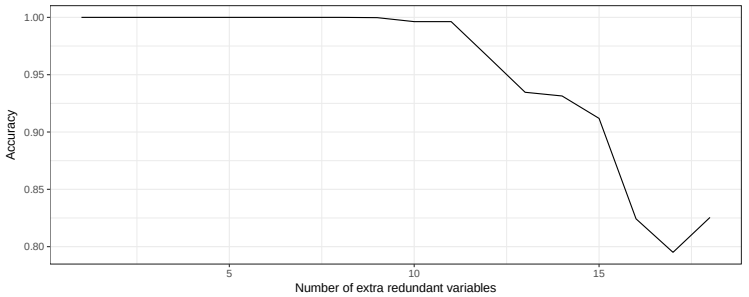


Figure 4: Accuracy of the classifiers corresponding to the optimal k for the artificial concept drift example, see Figure 3.

it. If it is truly the case that this is the best regularization parameter, then we can skip any cross-validation and simply find the smallest non-zero eigenvalue of the graph Laplacian using the Power method.

The optimization methods we use do not require $\mathbf{\Omega}$ to be full rank, which can be exploited to make the optmization faster.

6. Conclusion

We have demonstrated empirically that a semi-supervised regularizer can improve accuracy in classification tasks of high-dimensional data where few labels are available. We have also demonstrated that the method works jointly with a sparse regularizer, meaning that semi-supervised learning and sparse methods can go hand in hand. Using a semi-supervised regularizer we get more consistent accuracy when we repeatedly run the method, meaning that we can better rely on the resulting classifier. Finally we have demonstrated that the method can aid in the presence of concept drift.

References

Atkins, S., Einarsson, G., Ames, B., & Clemmensen, L. (2017). Proximal methods for sparse optimal scoring and discriminant analysis. *arXiv preprint arXiv:1705.07194*, .

Belkin, M., & Niyogi, P. (2001). Laplacian eigenmaps and spectral techniques for embedding and clustering. In *NIPS* (pp. 585–591). volume 14.

Belkin, M., & Niyogi, P. (2004). Semi-supervised learning on riemannian manifolds. *Machine learning*, *56*, 209–239.

Ben-David, S., Lu, T., & Pál, D. (2008). Does unlabeled data provably help? worst-case analysis of the sample complexity of semi-supervised learning. In *COLT* (pp. 33–44).

Cai, D., He, X., & Han, J. (2007). Semi-supervised discriminant analysis. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on* (pp. 1–7). IEEE.

Chen, S. S., Donoho, D. L., & Saunders, M. A. (2001). Atomic decomposition by basis pursuit. *SIAM review*, *43*, 129–159.

Chen, Y., Keogh, E., Hu, B., Begum, N., Bagnall, A., Mueen, A., & Batista, G. (2015). The ucr time series classification archive. *URL www. cs. ucr. edu/~ eamonn/time_series_data*, .

Clemmensen, L., Hastie, T., Witten, D., & Ersbøll, B. (2011). Sparse discriminant analysis. *Technometrics*, *53*, 406–413.

Clemmensen, L., & contributions by Max Kuhn (2016). *sparseLDA: Sparse Discriminant Analysis*. URL: `https://CRAN.R-project.org/package=sparseLDA` r package version 0.1-9.

Hastie, T., Tibshirani, R., & Buja, A. (1994). Flexible discriminant analysis by optimal scoring. *Journal of the American statistical association*, *89*, 1255–1270.

He, X., & Niyogi, P. (2003). Locality preserving projections. In *NIPS*. volume 16.

Krijthe, J. H. (2016). Rssl: Semi-supervised learning in r. *arXiv preprint arXiv:1612.07993*, .

Krijthe, J. H., & Loog, M. (2014). Implicitly constrained semi-supervised linear discriminant analysis. In *Pattern Recognition (ICPR), 2014 22nd International Conference on* (pp. 3762–3767). IEEE.

Krijthe, J. H., & Loog, M. (2015). Implicitly constrained semi-supervised least squares classification. In *International Symposium on Intelligent Data Analysis* (pp. 158–169). Springer.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*, 2278–2324.

Ng, A. Y., Jordan, M. I., Weiss, Y. et al. (2001). On spectral clustering: Analysis and an algorithm. In *NIPS* (pp. 849–856). volume 14.

Nigam, K., McCallum, A. K., Thrun, S., & Mitchell, T. (2000). Text classification from labeled and unlabeled documents using em. *Machine learning*, *39*, 103–134.

R Core Team (2015). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing Vienna, Austria. URL: `https://www.R-project.org/`.

Sindhwani, V., Niyogi, P., & Belkin, M. (2005). Beyond the point cloud: from transductive to semi-supervised learning. In *Proceedings of the 22nd international conference on Machine learning* (pp. 824–831). ACM.

Singh, A., Nowak, R., & Zhu, X. (2009). Unlabeled data: Now it helps, now it doesn't. In *Advances in neural information processing systems* (pp. 1513–1520).

Sjöstrand, K., Clemmensen, L. H., Larsen, R., & Ersbøll, B. (2012). Spasm: A matlab toolbox for sparse statistical modeling. *Journal of Statistical Software Accepted for publication*, .

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, (pp. 267–288).

Tsymbal, A. (2004). The problem of concept drift: definitions and related work. *Computer Science Department, Trinity College Dublin*, *106*.

Wang, D., Irani, D., & Pu, C. (2013). A study on evolution of email spam over fifteen years. In *Collaborative Computing: Networking, Applications and Worksharing (Collaboratecom), 2013 9th International Conference Conference on* (pp. 1–10). IEEE.

Witten, D. M., & Tibshirani, R. (2011). Penalized classification using fisher's linear discriminant. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *73*, 753–772.

Yarowsky, D. (1995). Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd annual meeting on Association for Computational Linguistics* (pp. 189–196). Association for Computational Linguistics.

Zhou, D., Bousquet, O., Lal, T. N., Weston, J., & Schölkopf, B. (2003). Learning with local and global consistency. In *NIPS* (pp. 321–328). volume 16.

Zhu, X., Ghahramani, Z., Lafferty, J. et al. (2003). Semi-supervised learning using gaussian fields and harmonic functions. In *ICML* (pp. 912–919). volume 3.

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *67*, 301–320.

APPENDIX C

Spasm: A matlab toolbox for sparse statistical modeling

The following manuscript was accepted for publication in the Journal of Statistical Software in June 2017. It is expected to appear in the journal sometime in 2018.

Journal of Statistical Software

MMMMMM YYYY, Volume VV, Issue II. *doi: 10.18637/jss.v000.i00*

SpaSM: A MATLAB Toolbox for Sparse Statistical Modeling

Karl Sjöstrand
EXINI Diagnostics AB

Line Harder Clemmensen
Technical University of Denmark

Gudmundur Einarsson
Technical University of Denmark

Rasmus Larsen
Technical University of Denmark

Bjarne Ersbøll
Technical University of Denmark

Abstract

Applications in biotechnology such as gene expression analysis and image processing have led to a tremendous development of statistical methods with emphasis on reliable solutions to severely underdetermined systems. Furthermore, interpretations of such solutions are of importance, meaning that the surplus of inputs have been reduced to a concise model. At the core of this development are methods which augments the standard linear models for regression, classification and decomposition such that sparse solutions are obtained. This toolbox aims at making public carefully implemented and well-tested variants of the most popular such methods for the MATLAB programming environment. These methods consist of easy-to-read yet efficient implementations of various coefficient-path following algorithms and implementations of sparse principal component analysis and sparse discriminant analysis which are not available in MATLAB. The toolbox builds on code made public in 2005 and which has since been used in several studies.

Keywords: Least Angle Regression, LASSO, elastic net, sparse principal component analysis, sparse discriminant analysis, MATLAB.

1. Introduction

The introduction of the Least Angle Regression (LAR) method for regularized/sparse regres-

sion (Efron, Hastie, Johnstone, and Tibshirani 2004) marked the starting point of a series of important contributions to the statistical computing community with the following common properties:

- Solutions are obtained sequentially along a path of gradually changing amounts of regularization. This paper focuses on methods where the method coefficients are piecewise linear functions of the regularization parameter, and where algorithms proceed by finding the next piecewise linear breakpoint,
- For sufficient amounts of l_1 regularization, solutions are *sparse*, i.e., some of the coefficients of the model are exactly zero, leading to more compact models which are easier to interpret,
- Methods are *efficient*, meaning they perform on a par with competing statistical methods when performance is measured on a test data set.

Examples of contributions are (Zou and Hastie 2005), (Zou, Hastie, and Tibshirani 2006), (Rosset and Zhu 2007), (Hastie, Rosset, Tibshirani, and Zhu 2004), (Park and Hastie 2007), (Friedman, Hastie, and Tibshirani 2010), of which the first three are detailed in this paper. The methods cover regression (the LASSO and the Elastic Net with ridge regression as a special case), classification (sparse discriminant analysis (SDA) with penalized linear discriminant analysis as a special case), and unsupervised modeling (sparse principal component analysis (SPCA)). The goal of this paper is to provide reference MATLAB (The MathWorks Inc. 2010) implementations of these basic regularization-path oriented methods.

Currently there are no built-in implementations of least angle regression, SPCA or SDA in MATLAB. MATLAB includes an implementation of the LASSO and elastic net in the Statistics and Machine Learning Toolbox, but both are based on coordinate descent optimization instead of coordinate path tracking which is at the heart of the least angle regression method. The R package **glmnet** (Friedman *et al.* 2010) provides methods to work with generalized linear models via penalized maximum likelihood. The **glmnet** (Qian, Hastie, Friedman, Tibshirani, and Simon 2013) package is also available in MATLAB. This includes the LASSO and the elastic net, but both in R and MATLAB the optimization is done via coordinate descent. An implementation of least angle regression is available in the R package **lars** (Hastie and Efron 2013). There exist several isolated implementations of least angle regression online. There is one such implementation on the MATLAB Central File Exchange (Kim 2009).

There exist various problem formulations of SPCA. There exist several stand-alone implementations online. One such implementation, containing 8 problem formulations, is provided alongside (Richtárik, Takáč, and Ahipaşaoğlu 2012). An isolated MATLAB implementation can be found on the MATLAB Central File Exchange (Alsahaf 2015).

For SDA there also exist various problem formulations. Some of these options are implemented in R, (Mai, Yang, and Zou 2015b,a; Witten 2015; Witten and Tibshirani 2011). Note that the same implementation as in the SpaSM toolbox is also available in the R package **sparseLDA** (Clemmensen and contributions by Max Kuhn 2015).

Currently the SpaSM toolbox is the only comprehensive toolbox for MATLAB that provides a variety of sparse methods based on least angle regression. We also present previously unpublished developments of the algorithm for sparse principal component analysis, and provide some evidence that performance is only slightly lowered (in terms of variance explained), while

the computational complexity is significantly lowered. The implementation strikes a balance between performance and readability, making this toolbox a good starting point for learning the details of the methods. For this reason, the code is written as pure MATLAB scripts which closely follows the algorithms provided here. All methods have been fully described and validated in their respective publications; despite this we provide terse but relatively complete derivations of each algorithm such that the paper can be read and the algorithms understood without having all references at hand.

The rest of the paper is structured as follows. Section 3 gives a short overview of the methods and files presented in the toolbox. Section 4 gives a concise derivation of the methods and pseudo code for the algorithms is provided. Section 5 is a short tutorial on how to apply the functions from the toolbox, interpret the command line output and a description of the input/output to the functions. The examples are shown on simulated data sets and all the examples can be found in the toolbox, with the appropriate seeds to generate the simulated data sets. Section 6 shows methods from the toolbox used on two real world data sets, one regarding Diabetes data and the second on shape data from human silhouettes.

2. Related work

Here we summarize some of the recent advances in sparse methods that are not included in the package. We point the reader to relevant software available and how it can be accessed from MATLAB.

Sparse regression. The Dantzig selector by Candes and Tao (2007) is similar to the LASSO in a sense that it performance regression and model selection. The main difference from the LASSO is in the way the optimization problem is formulated. The l_1 norm of the regression coefficients is minimized under constrains on the residuals. This can be formulated as a linear program. The main results also include bounds on the errors of the regression coefficients that are nonasymptotic. An implementation of the Dantzig selector can be found in the R package **flare** (Li, Zhao, Wang, Yuan, and Liu 2014) in the function `slim`.

Zou (2006) derive the necessary conditions for the lasso variable selection to be consistent. He then proposes a new version of the LASSO called the adaptive LASSO. The modification of the traditional LASSO consists of adding weights to the regression coefficients in the penalty term, corresponding to the inverse of the OLS solution. An additional parameter γ is also added as the exponent of the weights for further tuning. It is proved that this method is consistent in variable selection and has the oracle-property, meaning that it performs as well as if the true underlying model was given in advance. This approach only works in the $p < n$ case, but one can use the ridge solution instead of the OLS solution to provide weights for the penalty term in the case of collinearity or the $p > n$ case. The drawback is that cross-validation must be performed over two paramter in the $p < n$ case, and three parameters in the $p > n$ case. This method is implemented in the R package **parcor** (Kraemer, Schaefer, and Boulesteix 2009) in the function `adalasso`.

Scaled sparse linear regression by Sun and Zhang (2012) gives a general approach to regression and penalization, with special focus on the LASSO. The idea is to estimate the parameters in the model and the noise level. By estimating the noise level, the penalization parameter can be adjusted in successive iterations of the algorithm. The parameter controlling the penalization

thus becomes data dependent. Oracle inequalities are proved for prediction, estimation of noise level and regression coefficients. An implementation can be found in the R package **scalreg** (Sun 2013), the method can be used with the function `scalreg`.

For generalized linear models one can apply an l_1-norm regularizer via the R packages **glmnet** (Friedman *et al.* 2010) and **penalized** (Goeman 2010). The main difference between the packages is the optimization of the likelihood functions. The **glmnet** package uses cyclical coordinate descent, while the **penalized** package uses a combination of gradient ascent and the Newton-Raphson algorithm. The **glmnet** package is also available for MATLAB (Qian *et al.* 2013).

A couple of R packages are available for general non-convex penalty functions. The **plus** package (Zhang 2010) has two main components for its MC_+ algorithm, namely a minimax concave penalty and penalized linear unbiased selection, which provides unbiased estimates of parameters and variable selection. The **ncvreg** package (Breheny and Huang 2011) also handles nonconvex penalty function but the optimization is done with coordinate descent.

Sparse graphical models. Sparse graphical models concern the estimation of edge weights in a graph where the nodes correspond to variables. Variables are connected in the graph if they are conditionally dependent. To achieve this one estimates a sparse inverse covariance matrix of a multivariate normal distribution from data. If an entry in the inverse covariance matrix is non-zero, then the corresponding variables are conditionally dependent. One of the variables could be a response variable, and thus these models can be used to model conditional dependence of the response to the predictors, like in traditional linear models.

In Meinshausen and Bühlmann (2006), the authors present a method to estimate the sparse inverse covariance matrix via neighbourhood selection with the LASSO. They build the covariance matrix by using the LASSO on each variable separately. They show promising computational results. They also show that they get consistent estimation of the edges in the graph by controlling the probability of falsely joining some distinct connectivity components of the graph. An implementation is available in the R package **spaceExt** (He 2011) in the function `glasso.miss`.

Friedman, Hastie, and Tibshirani (2008) estimate the sparse inverse covariance matrix by starting with a blockwise coordinate descent approach and then solve the exact problem with a LASSO penalty using coordinate descent, instead of solving for each variable independently like Meinshausen and Bühlmann (2006). The method is implemented in the R package **glasso** (Friedman, Hastie, and Tibshirani 2014) via the function `glasso`.

Cai, Liu, and Luo (2011) propose a method (CLIME) where they use constrained l_1 minimization to estimate the sparse covariance matrix. They also show some generic results on the rate of convergence for different types of tails of population distributions. The problem can be solved with linear programming. An implementation is available in the R package **fastclime** (Pang, Qi, Liu, and Vanderbei 2016) via the function `fastclime.selector`.

Sparse quadratic discriminant analysis. Le and Hastie (2014) present a class of rules spanning from QDA and naive Bayes through a path of sparse graphical models. The authors use a group LASSO penalty, which imposes sparsity on the same elements in all the K within class precision matrices, where K is the number of classes. The authors claim that the

estimates of interactions from their method are easier to interpret than other classifiers that regularize QDA. The authors do not provide software.

Other implementations in MATLAB. Liu, Ji, Ye *et al.* (2009) present a MATLAB package called SLEP (Sparse Learning with Efficient Projections). They provide some MATLAB implementations for efficient computation of lasso variants using optimization based on efficient Euclidean projections.

2.1. Calling R code from MATLAB

For comparisons, or to complement the methods in this toolbox, the R software packages referenced in the previous section can be run from within the MATLAB environment. There are a few ways to achieve this.

The most straightforward approach is to write a separate R script and run it in batch mode with a system command in MATLAB. One way to achieve this from MATLAB would be:

```
system('R CMD BATCH rScript output');
```

The output from the R script (`rScript`) is written in the file `output`. The user then needs to load the results into MATLAB.

More generic approaches are also available, like the *MATLAB R-link* package available on the MATLAB file exchange (Henson 2013). This solution only works on Windows operating system and allows one to copy data back and forth from MATLAB and R.

3. In the toolbox

The toolbox consists of a series of MATLAB (The MathWorks Inc. 2010) scripts and functions to build and apply various statistical models for both supervised and unsupervised analyses. Below are listings of each method, file, subfunction and utility.

3.1. Methods

Forward Selection A variant of stepwise regression in which variables are included one-by-one based on their correlation with the current residual vector. Provides a baseline algorithm for other sparse methods for regression in this toolbox.

Least Angle Regression Provides a more gentle version of the classical approach of forward selection regression. The algorithm is the basis for all other methods in the toolbox. The method is also an interesting statistical method in its own right.

LASSO This method adds l_1 (1-norm) regularization to ordinary least squares regression, yielding solutions which are sparse in terms of the regression coefficients. This may lead to efficient suppression of noise and aids in interpretation.

Elastic Net Combining the algorithmic ideas of Least Angle Regression, the computational benefits of ridge regression and the tendency towards sparse solutions of the LASSO,

this versatile method is applicable for many data sets, also when the number of predictor variables far exceed the number of observations. The corresponding LARS-EN algorithm is used in the implementation of the following two algorithms.

Sparse Principal Component Analysis Principal component analysis is a powerful tool for compacting a data set and for recovering latent structures in data, but solutions are difficult to interpret as they involve all the original predictor variables. Sparse principal component analysis approximates the behavior of regular principal component analysis but models each component as a linear combination of a subset of the original variables.

Sparse Linear Discriminant Analysis Linear discriminant analysis is a standard tool for classification of observations into one of two or more groups. Further, the data can be visualized along the obtained discriminative directions. As with principal component analysis, these directions are combinations of all predictor variables. Sparse discriminant analysis reduces this to a subset of variables which may improve performance as well interpretability.

3.2. Files

`forwardselection.m` A baseline algorithm for variable selection. Based on the algorithm in `lar.m`.

`lar.m` An implementation of the LARS algorithm for Least Angle Regression described by Efron *et al.* (2004).

`lasso.m` The LASSO method of Tibshirani (1996), implemented using a combination of the algorithms of Efron *et al.* (2004) and Rosset and Zhu (2007).

`elasticnet.m` The Elastic Net algorithm of Zou and Hastie (2005), with elements from Rosset and Zhu (2007).

`spca.m` The sparse principal component algorithm based on the work by Zou *et al.* (2006), with modification described below.

`slda.m` The sparse discriminant analysis of Clemmensen, Hastie, Witten, and Ersbøll (2011).

3.3. Sub-functions and utilities

`larsen.m` The actual implementation of the Elastic Net algorithm. The functions `lasso.m`, `elasticnet.m`, `spca.m` and `slda.m` depend on this function; however it is not intended for direct use.

`cholinsert.m` Update of the Cholesky factorization of $\mathbf{X}^\top\mathbf{X} + \delta\mathbf{I}$. Used in `lar.m` and `larsen.m`.

`choldelete.m` Downdate of the above Cholesky factorization. Used in `larsen.m`.

`center.m` Convenience function for centering (removing the mean observation) a data matrix or response vector.

`normalize.m` Convenience function for centering and normalizing a data matrix or response matrix such that variables have unit Euclidean length.

4. Methods and algorithms

This section presents the principles behind each method in the toolbox, and outlines their algorithms. The basic building block is the LARS-EN algorithm (Zou and Hastie 2005) which encompasses regression via ordinary least squares, ridge regression, the LASSO and the Elastic Net. These are based on the linear model $\mathbf{y} = \mathbf{X}\beta + \boldsymbol{\varepsilon}$ where $\mathbf{y}$ $(n \times 1)$ is the observed response variable, $\mathbf{X}$ $(n \times p)$ is the data matrix where the ith column represents the ith predictor variable, β $(p \times 1)$ is the set of model coefficients which determines the load on each predictor variable, and $\boldsymbol{\varepsilon}$ are the residual errors. Unless stated otherwise, $\mathbf{y}$ is assumed centered and $\mathbf{X}$ is assumed centered and normalized such that each variable has zero mean and unit Euclidean length. A sparse method for regression estimates a coefficient vector β with many zero elements, giving an estimate $\hat{\mathbf{y}}$ of $\mathbf{y}$ which is a linear combination of a subset of available variables in $\mathbf{X}$. Sparse solutions may be preferred to full counterparts if the latent linear model can be assumed to be sparse, or when interpretation of the results is important. The set $\mathcal{A}$ denotes the indices in β corresponding to non-zero elements; we refer to this as the *active set*. The set $\mathcal{I}$ is called the *inactive set* and denotes the complement of $\mathcal{A}$. We use these sets also to denote submatrices such as the $(n \times |\mathcal{A}|)$ matrix $\mathbf{X}_{\mathcal{A}}$, consisting of the columns (variables) of $\mathbf{X}$ corresponding to the indices in $\mathcal{A}$. All algorithms proceed in iterations and we indicate iteration number by a parenthesized superscript number, e.g., $\hat{\beta}^{(k)}$ for the regression coefficients calculated in the kth iteration. It is further convenient to define an operator $\min^{+}(\cdot)$ which finds the smallest strictly positive value of the (vector-valued) input.

The methods for regression described below proceed in an iterative manner, adding or subtracting variables in the model in each step. The methods start with the trivial constant model, then move towards the full representation which corresponds to ordinary least squares regression or ridge regression, depending on the type of regularization. To put the presentation of these algorithms into perspective, we begin with a quick review of one of the simplest algorithms of this kind.

In *forward selection*, a variant of *stepwise regression*, variables are added one-by-one until some goodness-of-fit criterion is fulfilled. The next variable to include in this scheme can be chosen based on a number of criteria. The methods in this toolbox generally pick the variable that has the highest absolute correlation with the current residual vector. To fix the terminology and to give a simple baseline algorithm we state a forward selection algorithm in Algorithm 1.

In this algorithm, we move to the least squares solution using all currently active variables in each step. This approach is known as a greedy method. The following sections cover less greedy variations on the forward selection scheme which result in algorithms with generally better performance and which are able to handle more difficult data sets.

Algorithm 1 Forward Selection

1: Initialize the active set $\mathcal{A} = \emptyset$ and the inactive set $\mathcal{I} = \{1 \dots p\}$
2: Initialize the coefficient vector $\beta^{(0)} = \mathbf{0}$
3: **for** $k \in \{0 \dots p-1\}$ **do**
4: Find variable maximally correlated with the current residual $i = \arg\max_{i \in \mathcal{I}} \mathbf{x}_i^\top (\mathbf{y} - \mathbf{X}\beta^{(k)})$
5: Move i from $\mathcal{I}$ to $\mathcal{A}$.
6: Update the active set coefficients $\beta_{\mathcal{A}}^{(k+1)} = (\mathbf{X}_{\mathcal{A}}^\top \mathbf{X}_{\mathcal{A}})^{-1} \mathbf{X}_{\mathcal{A}}^\top \mathbf{y}$
7: **end for**
8: Output the series of coefficients $\mathbf{B} = [\beta^{(0)} \dots \beta^{(p)}]$.

4.1. Least Angle Regression

Least Angle Regression (LAR) is a regression method that provides a more gentle version of forward selection. Conceptually, LAR modifies Algorithm 1 on only one account. Instead of choosing a step size which yields the (partial) least squares solution in each step, we shorten the step length such that we stop when any inactive variable becomes equally important as the active variables in terms of correlation with the residual vector. That variable is then included in the active set and a new direction is calculated. Recall that all active variables are uncorrelated with the residual vector at the least squares solution, the step length will therefore always be as short or shorter at the point where we find the next active variable to include than that of the least squares solution.

The algorithm starts with the empty set of active variables. The correlation between each variable and the response is measured, and the variable with the highest correlation becomes the first variable included into the model. The first direction is then towards the least squares solution using this single active variable. Walking along this direction, the angles between the variables and the residual vector are measured. Along this walk, the angles will change; in particular, the correlation between the residual vector and the active variable will shrink linearly towards 0. At some stage before this point, another variable will obtain the same correlation with respect to the residual vector as the active variable. The walk stops and the new variable is added to the active set. The new direction of the walk is towards the least squares solution of the two active variables, and so on. After p steps, the full least squares solution will be reached.

The LAR algorithm is efficient since there is a closed form solution for the step length at each stage. Denoting the model estimate of $\mathbf{y}$ at iteration k by $\hat{\mathbf{y}}^{(k)}$ and the least squares solution including the newly added active variable $\hat{\mathbf{y}}_{\text{OLS}}^{(k+1)}$, the walk from $\hat{\mathbf{y}}^{(k)}$ towards $\hat{\mathbf{y}}_{\text{OLS}}^{(k+1)}$ can be formulated $(1-\gamma)\hat{\mathbf{y}}^{(k)} + \gamma\hat{\mathbf{y}}_{\text{OLS}}^{(k+1)}$ where $0 \le \gamma \le 1$. Estimating $\hat{\mathbf{y}}^{(k+1)}$, the position where the next active variable is to be added, then amounts to estimating γ. We seek the smallest positive γ where correlations become equal, that is

$$\mathbf{x}_{i\in\mathcal{I}}^\top(\mathbf{y} - (1-\gamma)\hat{\mathbf{y}}^{(k)} - \gamma\hat{\mathbf{y}}_{\text{OLS}}^{(k+1)}) = \mathbf{x}_{j\in\mathcal{A}}^\top(\mathbf{y} - (1-\gamma)\hat{\mathbf{y}}^{(k)} - \gamma\hat{\mathbf{y}}_{\text{OLS}}^{(k+1)}). \tag{1}$$

Solving this expression for γ, we get

$$\gamma_{i\in\mathcal{I}} = \frac{(\mathbf{x}_i - \mathbf{x}_j)^\top(\mathbf{y} - \hat{\mathbf{y}}^{(k)})}{(\mathbf{x}_i - \mathbf{x}_j)^\top(\hat{\mathbf{y}}_{\text{OLS}}^{(k+1)} - \hat{\mathbf{y}}^{(k)})} = \frac{(\mathbf{x}_i - \mathbf{x}_j)^\top\boldsymbol{\varepsilon}}{(\mathbf{x}_i - \mathbf{x}_j)^\top\mathbf{d}}, \tag{2}$$

where $\mathbf{d} = \hat{\mathbf{y}}_{\text{OLS}}^{(k+1)} - \hat{\mathbf{y}}^{(k)}$ is the direction of the walk, and $j \in \mathcal{A}$. Now, $\mathbf{d}$ is the orthogonal projection of $\boldsymbol{\varepsilon}$ onto the plane spanned by the variables in $\mathcal{A}$, therefore we have $\mathbf{x}_j^\top \boldsymbol{\varepsilon} = \mathbf{x}_j^\top \mathbf{d} \equiv c$, representing the angle at the current breakpoint $\hat{\mathbf{y}}^{(k)}$. Furthermore, the sign of the correlation between variables is irrelevant. Therefore, we have

$$\gamma = \min_{i \in \mathcal{I}} \left\{ \frac{\mathbf{x}_i^\top \boldsymbol{\varepsilon} - c}{\mathbf{x}_i^\top \mathbf{d} - c}, \quad \frac{\mathbf{x}_i^\top \boldsymbol{\varepsilon} + c}{\mathbf{x}_i^\top \mathbf{d} + c} \right\}, \quad 0 < \gamma \leq 1, \tag{3}$$

where the two terms are for correlations/angles of equal and opposite sign respectively. The coefficients at this next step are given by

$$\beta^{(k+1)} = (1 - \gamma)\beta^{(k)} + \gamma \beta_{\text{OLS}}^{(k+1)}. \tag{4}$$

Given these key pieces of the LAR algorithm, we state the entire procedure in Algorithm 2.

Algorithm 2 Least Angle Regression

1: Initialize the coefficient vector $\beta^{(0)} = \mathbf{0}$ and the fitted vector $\hat{\mathbf{y}}^{(0)} = \mathbf{0}$,
2: Initialize the active set $\mathcal{A} = \emptyset$ and the inactive set $\mathcal{I} = \{1 \dots p\}$
3: **for** $k = 0$ **to** $p - 2$ **do**
4: Update the residual $\boldsymbol{\varepsilon} = \mathbf{y} - \hat{\mathbf{y}}^{(k)}$
5: Find the maximal correlation $c = \max_{i \in \mathcal{I}} |\mathbf{x}_i^\top \boldsymbol{\varepsilon}|$
6: Move variable corresponding to c from $\mathcal{I}$ to $\mathcal{A}$.
7: Calculate the least squares solution $\beta_{\text{OLS}}^{(k+1)} = (\mathbf{X}_\mathcal{A}^\top \mathbf{X}_\mathcal{A})^{-1} \mathbf{X}_\mathcal{A}^\top \mathbf{y}$
8: Calculate the current direction $\mathbf{d} = \mathbf{X}_\mathcal{A} \beta_{\text{OLS}}^{(k+1)} - \hat{\mathbf{y}}^{(k)}$
9: Calculate the step length $\gamma = \min_{i \in \mathcal{I}}^{+} \left\{ \frac{\mathbf{x}_i^\top \boldsymbol{\varepsilon} - c}{\mathbf{x}_i^\top \mathbf{d} - c}, \quad \frac{\mathbf{x}_i^\top \boldsymbol{\varepsilon} + c}{\mathbf{x}_i^\top \mathbf{d} + c} \right\}$, $0 < \gamma \leq 1$
10: Update regression coefficients $\beta^{(k+1)} = (1 - \gamma)\beta^{(k)} + \gamma \beta_{\text{OLS}}^{(k+1)}$
11: Update the fitted vector $\hat{\mathbf{y}}^{(k+1)} = \hat{\mathbf{y}}^{(k)} + \gamma \mathbf{d}$
12: **end for**
13: Let $\beta^{(p)}$ be the full least squares solution $\beta^{(p)} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$
14: Output the series of coefficients $\mathbf{B} = [\beta^{(0)} \dots \beta^{(p)}]$

Each step of Algorithm 2 adds a covariate to the model until the full least squares solution is reached. It is natural to parameterize this process by the size $s(\beta)$ of the coefficients at each step as well as in between steps of the algorithm. The algorithm returns the following parametrization,

$$s(\beta) = \|\beta\|_1 = \sum_{i=1}^{p} |\beta_i|. \tag{5}$$

Picking a suitable model for a particular analysis thus means selecting a suitable value of $s(\beta) \in (0, \|\beta_{OLS}\|_1)$. Cross-validation or an independent validation data set are obvious choices for this purpose, however, the algorithm provides information which substitute or complement this process.

Degrees of freedom Efron *et al.* (2004) showed that the number of degrees of freedom at each step of the LAR algorithm is well approximated by the number of non-zero elements of β. The algorithm therefore returns the following sequence,

$$df_{LAR}^{(k)} = |\mathcal{A}| = k, \quad k = 0 \dots p. \tag{6}$$

Mallow's C_p Given the above measure of the number of degrees of freedom, we can calculate a number of model selection criteria. Mallow's C_p measure is defined as (Zou, Hastie, and Tibshirani 2007)

$$C_p^{(k)} = \frac{1}{\sigma_\varepsilon^2}\|\mathbf{y} - \mathbf{X}\beta^{(k)}\|^2 - n + 2df^{(k)}. \tag{7}$$

Akaike's Information Criterion Akaike's information criterion is similar to Mallow's C_p and is defined as

$$AIC^{(k)} = \|\mathbf{y} - \mathbf{X}\beta^{(k)}\|^2 + 2\sigma_\varepsilon^2 df^{(k)}. \tag{8}$$

Bayesian Information Criterion The Bayesian information criterion tends to choose a more sparse model than both AIC and C_p and is defined as

$$BIC^{(k)} = \|\mathbf{y} - \mathbf{X}\beta^{(k)}\|^2 + \log(n)\sigma_\varepsilon^2 df^{(k)}. \tag{9}$$

The latter three criteria can be used to pick a suitable model, typically indicated by the smallest value of each criterion. Alternatively, one can choose the sparsest model for which more complex models lead to scant improvements in the relevant model selection criterion. The measure σ_ε^2 represents the residual variance of a low-bias model which is here defined as

$$\sigma_\varepsilon^2 = \frac{1}{n}\|\mathbf{y} - \mathbf{X}^\dagger\mathbf{y}\|^2, \tag{10}$$

where $\mathbf{X}^\dagger$ is the Moore-Penrose pseudo-inverse of $\mathbf{X}$, equivalent to a ridge regression solution $\arg\min \|\mathbf{y}-\mathbf{X}\beta\|^2+\lambda\|\beta\|^2$ in the limit $\lambda \to 0$. Note that in cases where $p > n$, this measure of the residual variance will be zero which in effect turns the information criteria defined above into a measure of training error only. We therefore recommend using these criteria for model selection only in cases where n is well above p.

The key computational burden of Algorithm 2 lies in Step 7 where the OLS solution involving the variables in $\mathcal{A}$ is calculated. Two techniques are used to alleviate this. For problems where n is at least ten times larger than p, we calculate the full Gram matrix $\mathbf{X}^\top\mathbf{X}$ once and use the submatrix $\mathbf{X}_\mathcal{A}^\top\mathbf{X}_\mathcal{A}$ to find $\beta_{\text{OLS}}^{(k+1)}$, thus avoiding an $\mathcal{O}(|\mathcal{A}|^2 n)$ matrix multiplication. When $p > 1000$, this method is not preferred since the memory footprint of the resulting $p \times p$ Gram matrix may pose a problem. In cases where $10n < p$, or when a pre-computed Gram matrix is impractical, we maintain a matrix $\mathbf{R}$ of the Cholesky factorization of the current Gram (sub)matrix $\mathbf{X}_\mathcal{A}^\top\mathbf{X}_\mathcal{A}$ such that $\mathbf{R}^\top\mathbf{R} = \mathbf{X}_\mathcal{A}^\top\mathbf{X}_\mathcal{A}$. As variables join the active set, $\mathbf{R}$ can be updated with low computational cost. The conditions chosen for selecting between the two methods are not exact, but we have gathered evidence through simulation studies that they work well on most standard computers.

In practice one frequently has a notion of the sparsity of the desired solution when running Algorithm 2. To avoid unnecessary computations, the algorithm can be stopped prematurely, either when the active set reaches a certain size, or when the l_1 norm of the coefficients in $\beta^{(k)}$ exceeds a preset threshold. Optionally, the algorithm stores and returns the solution fulfilling the specified sparsity criterion only in order to save computer resources. This direct controlling of sparsity is a clear advantage over the coordinate descent method, where the number of non-zero parameters in the model cannot be specified directly.

4.2. The LASSO

The LASSO (Tibshirani 1996) represents the most basic augmentation of the ordinary least squares solution which implements coefficient shrinkage and selection. The sum of squared residuals loss function $L(\hat{\beta}(\lambda))$ is combined with a penalty function $J(\hat{\beta}(\lambda))$ based on the l_1 norm as,

$$\hat{\beta}(\lambda) = \arg\min_{\beta} L(\hat{\beta}(\lambda)) + \lambda J(\hat{\beta}(\lambda)) = \arg\min_{\beta} \|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda\|\beta\|_1. \tag{11}$$

The l_1 penalty will promote sparse solutions. This means that as λ is increased, elements of $\hat{\beta}(\lambda)$ will become exactly zero. Due to the non-differentiability of the penalty function, there are no closed-form solutions to Equation (11). A number of algorithms have been proposed (e.g., Fu (1998); Osborne, Presnell, and Turlach (2000); Friedman *et al.* (2010)) including the quadratic programming approach on an expanded space of variables outlined in the original LASSO paper of Tibshirani (1996). The algorithm presented here is due to Rosset and Zhu (2007) who derived a sufficient condition for piecewise linear coefficient paths on which they based several LASSO-type methods. The LASSO algorithm described here is a special case of their work. Efron *et al.* (2004) arrived at an equivalent algorithm by showing that a small modification to the Least Angle Algorithm yields LASSO solutions.

The goal of this section is to derive an expression for how the solutions of Equation (11) change with λ. The solution set $\hat{\beta}(\lambda)$ will hit a non-differentiability point when coefficients either go from non-zero to zero (join $\mathcal{I}$), or the other way around (join $\mathcal{A}$). Assume first that we are in a region of values of λ where variables are neither joining nor leaving $\mathcal{A}$. The normal equations to Equation (11) around λ and around a nearby point $\lambda + \epsilon$ are then

$$-2\mathbf{X}_{\mathcal{A}}^{\top}(\mathbf{y} - \mathbf{X}_{\mathcal{A}}\hat{\beta}_{\mathcal{A}}(\lambda)) + \lambda \cdot \text{sign}(\hat{\beta}_{\mathcal{A}}(\lambda)) = 0 \tag{12}$$

$$-2\mathbf{X}_{\mathcal{A}}^{\top}(\mathbf{y} - \mathbf{X}_{\mathcal{A}}\hat{\beta}_{\mathcal{A}}(\lambda+\epsilon)) + (\lambda+\epsilon) \cdot \text{sign}(\hat{\beta}_{\mathcal{A}}(\lambda+\epsilon)) = 0. \tag{13}$$

We now write Equation (13) as a first order Taylor expansion around $\hat{\beta}(\lambda)$. The general form of a multivariate Taylor expansion of $\mathbf{f}(x)$ around a is

$$\mathbf{f}(x) = \sum_{k=0}^{\infty} \frac{\nabla^{(k)}\mathbf{f}(a)}{k!}(x-a)^k = \mathbf{f}(a) + \nabla\mathbf{f}(a)(x-a) + \frac{1}{2}\nabla^2\mathbf{f}(a)(x-a)^2 + \ldots. \tag{14}$$

We have,

$$\mathbf{f}(\hat{\beta}(\lambda)) = -2\mathbf{X}_{\mathcal{A}}^{\top}(\mathbf{y} - \mathbf{X}_{\mathcal{A}}\hat{\beta}_{\mathcal{A}}(\lambda)) + (\lambda+\epsilon) \cdot \text{sign}(\hat{\beta}_{\mathcal{A}}(\lambda)) \tag{15}$$

$$\nabla\mathbf{f}(\hat{\beta}(\lambda)) = 2\mathbf{X}_{\mathcal{A}}^{\top}\mathbf{X}_{\mathcal{A}} \tag{16}$$

$$\nabla^{(k)}\mathbf{f}(\hat{\beta}(\lambda)) = 0 \quad \text{for } k = 2\ldots\infty. \tag{17}$$

The complete expansion of Equation (13) becomes

$$-2\mathbf{X}_{\mathcal{A}}^{\top}(\mathbf{y} - \mathbf{X}_{\mathcal{A}}\hat{\beta}_{\mathcal{A}}(\lambda)) + (\lambda+\epsilon) \cdot \text{sign}(\hat{\beta}_{\mathcal{A}}(\lambda)) + 2\mathbf{X}_{\mathcal{A}}^{\top}\mathbf{X}_{\mathcal{A}}\left(\hat{\beta}_{\mathcal{A}}(\lambda+\epsilon) - \hat{\beta}_{\mathcal{A}}(\lambda)\right) = 0. \tag{18}$$

Using Equation (12), we can rearrange this expression to

$$\frac{\hat{\beta}_{\mathcal{A}}(\lambda+\epsilon) - \hat{\beta}_{\mathcal{A}}(\lambda)}{\epsilon} = -(2\mathbf{X}_{\mathcal{A}}^{\top}\mathbf{X}_{\mathcal{A}})^{-1}\text{sign}(\hat{\beta}_{\mathcal{A}}(\lambda)), \tag{19}$$

which approaches $\nabla\hat{\beta}(\lambda)$ as $\epsilon \to 0$ (using $\nabla\hat{\beta}_{\mathcal{I}}(\lambda) = 0$). This is a constant function which means that the coefficient paths between *events* (changes to $\mathcal{A}$ and $\mathcal{I}$) are piecewise linear, similarly to Least Angle Regression.

Now that an expression for the change in $\hat{\beta}(\lambda)$ between events has been established, we focus on finding the values of λ for which changes to $\mathcal{A}$ and $\mathcal{I}$ take place. It is beneficial here to consider Equation (11) on an expanded set of β-values, chosen such that $\beta_j = \beta_j^+ + \beta_j^-$ where $\beta_j^+ \geq 0$ and $\beta_j^- \geq 0$, $\forall j$,

$$\arg\min_{\beta^+,\beta^-} \|\mathbf{y} - \mathbf{X}(\beta^+ - \beta^-)\|^2 + \lambda\|(\beta^+ + \beta^-)\|_1 = L(\hat{\beta}(\lambda)) + \lambda\|(\beta^+ + \beta^-)\|_1 \qquad (20)$$
$$\text{such that} \quad \beta_j^+ \geq 0, \beta_j^- \geq 0, \forall j.$$

This formulation of the LASSO is differentiable, at the price of having to deal with twice as many variables. The Lagrange primal function is

$$L(\hat{\beta}(\lambda)) + \lambda\|(\beta^+ + \beta^-)\|_1 - \sum_{j=1}^{p} \lambda_j^+\beta_j^+ - \sum_{j=1}^{p} \lambda_j^-\beta_j^- , \quad \lambda_j^+ \geq 0, \lambda_j^- \geq 0, \forall j,$$

where we have introduced the Lagrange multipliers λ_j^+ and λ_j^-. The Karush-Kuhn-Tucker conditions are

$$(\nabla L(\beta))_j + \lambda - \lambda_j^+ = 0 \qquad (21)$$
$$-(\nabla L(\beta))_j + \lambda - \lambda_j^- = 0 \qquad (22)$$
$$\lambda_j^+\beta_j^+ = 0 \qquad (23)$$
$$\lambda_j^-\beta_j^- = 0. \qquad (24)$$

From these conditions, a number of useful properties arise. First, we note that setting $\lambda = 0$ indeed gives us (using Equation (21) and Equation (22)) $\nabla L(\beta) = 0$ as expected. For positive values of λ we have,

$$\beta_j^+ > 0 + \Rightarrow +\lambda_j^+ = 0 \Rightarrow \nabla L(\beta) = -\lambda \Rightarrow \lambda_j^- > 0 \Rightarrow \beta_j^- = 0 \qquad (25)$$
$$\beta_j^- > 0 + \Rightarrow +\lambda_j^- = 0 \Rightarrow \nabla L(\beta) = \lambda \Rightarrow \lambda_j^+ > 0 \Rightarrow \beta_j^+ = 0. \qquad (26)$$

Elements in $\mathcal{A}$ have either $\beta_j^+ > 0$ or $\beta_j^- > 0$, but cannot both be non-zero. That is,

$$|(\nabla L(\beta))_j| = \lambda, \quad j \in \mathcal{A} \qquad (27)$$
$$|(\nabla L(\beta))_j| \leq \lambda, \quad j \in \mathcal{I}.$$

for $j \in \mathcal{A}$, $|(\nabla L(\beta))_j| = \lambda$ while for elements $j \in \mathcal{I}$, $|(\nabla L(\beta))_j| \leq \lambda$. We are seeking the value of $\gamma > 0$ for which a variable in $\mathcal{I}$ joins $\mathcal{A}$ or vice versa. We have arrived at the following conditions,

$$j \in \mathcal{A} \to \mathcal{I}: \quad \hat{\beta}_j^{(k)} + \gamma\nabla\hat{\beta}_j^{(k)} = 0, \quad j \in \mathcal{A} \qquad (28)$$
$$j \in \mathcal{I} \to \mathcal{A}: \quad |(\nabla L(\hat{\beta}^{(k)} + \gamma\nabla\hat{\beta}^{(k)}))_i| = |(\nabla L(\hat{\beta}^{(k)} + \gamma\nabla\hat{\beta}^{(k)}))_j|, \quad j \in \mathcal{A}, i \in \mathcal{I}. \qquad (29)$$

The first of these expressions defines the distances $\{\gamma\}$ at which active variables hit zero and join $\mathcal{I}$. The second expression defines the distances at which inactive variables violate the

second condition in Equation (27) and thus must join $\mathcal{A}$. Note that any element in $\mathcal{A}$ can be chosen to calculate the RHS of Equation (29), they all equal λ. The smallest value γ_{min} of the distances $\{\gamma\}$ is where the next event will happen. The coefficients can now be updated by

$$\hat{\beta}^{(k+1)} = \hat{\beta}^{(k)} + \gamma_{min} \nabla \hat{\beta}^{(k)}. \tag{30}$$

We have arrived at Algorithm 3 for the LASSO.

Algorithm 3 LASSO (Rosset and Zhu 2007)

Initialize $\beta^{(0)} = \mathbf{0}$, $\mathcal{A} = \arg\max_j |\mathbf{x}_j^\top \mathbf{y}|$, $\nabla \hat{\beta}_{\mathcal{A}}^{(0)} = -\text{sign}(\mathbf{x}_{\mathcal{A}}^\top \mathbf{y})$, $\nabla \hat{\beta}_{\mathcal{I}}^{(0)} = 0$, $k = 0$
while $\mathcal{I} \neq \emptyset$ **do**
  $\gamma_j = \min^+_{j \in \mathcal{A}} \; -\beta_j^{(k)} / \nabla \hat{\beta}_j^{(k)}$
  $\gamma_i = \min^+_{i \in \mathcal{I}} \; \{ \frac{(\mathbf{x}_i + \mathbf{x}_j)^\top (\mathbf{y} - \mathbf{X}\hat{\beta}^{(k)})}{(\mathbf{x}_i + \mathbf{x}_j)^\top (\mathbf{X} \nabla \hat{\beta}^{(k)})}, \frac{(\mathbf{x}_i - \mathbf{x}_j)^\top (\mathbf{y} - \mathbf{X}\hat{\beta}^{(k)})}{(\mathbf{x}_i - \mathbf{x}_j)^\top (\mathbf{X} \nabla \hat{\beta}^{(k)})} \}$ where j is any index in $\mathcal{A}$
  $\gamma = \min\{\gamma_j, \gamma_i\}$
  if $\gamma = \gamma_j$ **then**
    Move j from $\mathcal{A}$ to $\mathcal{I}$
  else
    Move i from $\mathcal{I}$ to $\mathcal{A}$
  end if
  $\hat{\beta}^{(k+1)} = \hat{\beta}^{(k)} + \gamma \nabla \hat{\beta}^{(k)}$
  $\nabla \hat{\beta}_{\mathcal{A}}^{(k+1)} = -(2\mathbf{X}_{\mathcal{A}}^\top \mathbf{X}_{\mathcal{A}})^{-1} \cdot \text{sign}(\hat{\beta}_{\mathcal{A}}^{(k+1)})$
  $k = k + 1$
end while
Output the series of coefficients $\mathbf{B} = [\beta^{(0)} \ldots \beta^{(k)}]$

One of the benefits with this particular algorithm is that the coefficient path can be parameterized either in terms of $\|\hat{\beta}^{(k)}\|_1$, the size of the penalty at iteration k, or the regularization parameter λ. The latter is seldom explicitly specified in path-following algorithms but here, the first identity in Equation (27) provides a way of directly calculating λ as a function of $\hat{\beta}$,

$$\lambda = 2|\mathbf{x}_{j \in \mathcal{A}}^\top (\mathbf{y} - \mathbf{X}\hat{\beta})|. \tag{31}$$

Any element in $\mathcal{A}$ will do for this calculation. To minimize the risk of numerical problems, we calculate this value for all elements in $\mathcal{A}$ and pick the median.

If asked for, the algorithm returns the same information as Algorithm 2. The LASSO solution path can be parameterized either in terms of $s(\beta)$ (cf., Equation (5)), or in terms of λ which also can be interpreted as a function of β, cf., Equation (31). Zou *et al.* (2007) show that an unbiased estimate of the degrees of freedom of a particular LASSO solution is given by $|\mathcal{A}|$, the number of non-zero components of β. Given this estimate, the various model selection criteria can be calculated as outlined in Section 4.1.

We use the same Gram matrix or Cholesky updating scheme as described in Section 4.1. As variables leave the active set, the Cholesky factorization $\mathbf{R}^\top \mathbf{R} = \mathbf{X}_{\mathcal{A}}^\top \mathbf{X}_{\mathcal{A}}$ is downdated by removing the contribution to $\mathbf{R}$ which is due to the dropped variable.

4.3. The elastic net

Ridge regression (Hoerl and Kennard 1970) represents an effective way of shrinking the OLS coefficients towards zero. The l_1 penalty of the LASSO is replaced with an l_2 penalty,

$$\hat{\beta}(\delta) = \arg\min_{\beta} \|\mathbf{y} - \mathbf{X}\beta\|^2 + \delta\|\beta\|^2, \tag{32}$$

which leads to the closed form solution

$$\hat{\beta}(\delta) = (\mathbf{X}^\top\mathbf{X} + \delta\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{y}. \tag{33}$$

Although similar in formulation, ridge regression and the LASSO have important differences. The l_2 penalty of ridge regression leads to a shrinkage of the regression coefficients, much like the l_1 penalty of the LASSO, but coefficients are not forced to exactly zero for finite values of δ. However, a benefit of ridge regression is that a unique solution is available, also when the data matrix $\mathbf{X}$ is rank deficient, e.g., when there are more predictors than observations ($p > n$). This is seen in Equation (33); the addition of a sufficiently large constant value along the diagonal of the Gram matrix $\mathbf{X}^\top\mathbf{X}$ ensures full rank (Petersen and Pedersen 2008). The LASSO algorithm (Algorithm 3) is terminated when the active set size $|\mathcal{A}|$ becomes larger than p since the matrix $\mathbf{X}_\mathcal{A}^\top\mathbf{X}_\mathcal{A}$ in Step 12 is no longer invertible.

Elastic net regression (Zou and Hastie 2005) combines the virtues of ridge regression and the LASSO by considering solutions penalized by both an l_2 and an l_1 term,

$$\hat{\beta}(\lambda, \delta) = \arg\min_{\beta} \|\mathbf{y} - \mathbf{X}\beta\|^2 + \delta\|\beta\|^2 + \lambda\|\beta\|_1, \tag{34}$$

thus bridging the gap between the LASSO ($\delta = 0$) and ridge regression ($\lambda = 0$). The l_2 penalty ensures a unique solution also when $p > n$ and the l_1 penalty offers variable selection via a sparse vector of coefficients $\hat{\beta}$. Moreover, the l_2 leads to a *grouping effect* (Zou and Hastie 2005), a term that alludes to the characteristic that highly correlated predictors tend to have similar regression coefficients for nonzero δ. Note however that this does in general not mean that highly correlated variables are included into the active set in groups along the regularization path.

We can use the LASSO algorithm to obtain the full regularization path of elastic net solutions. To see this, we first note that ridge regression solutions can be obtained by solving an ordinary least squares problem with an augmented set of observations,

$$\tilde{\mathbf{X}} = \begin{bmatrix} \mathbf{X} \\ \sqrt{\delta}\mathbf{I}_p \end{bmatrix}, \quad \tilde{\mathbf{y}} = \begin{bmatrix} \mathbf{y} \\ \mathbf{0} \end{bmatrix}. \tag{35}$$

Expanding the equation $\hat{\beta} = (\tilde{\mathbf{X}}^\top\tilde{\mathbf{X}})^{-1}\tilde{\mathbf{X}}^\top\tilde{\mathbf{y}}$ gives the ridge solution in Equation (33). For a fixed value of δ, Algorithm 3 offers solutions for all relevant values of λ. Selecting suitable values for the regularization parameters typically involves selecting the best value of λ for a discrete set of values of δ. Thus, the algorithm must be run for each value of δ.

If $p > n$, the augmented data matrix in Equation (35) has size $(n+p) \times p$, implying a system of equations that may be prohibitively large. Remarkably, it turns out that we can do without explicitly forming these augmented matrices, mainly due to the fact that any multiplications with $\tilde{\mathbf{y}}$ effectively voids the contribution of the additional rows in $\tilde{\mathbf{X}}$ since the corresponding rows of $\tilde{\mathbf{y}}$ are zero. Other computations are dot products between vectors with additional elements in $\mathcal{I}$ and vectors with additional elements in $\mathcal{A}$. Since these never coincide ($\mathcal{I} = \mathcal{A}^c$),

these additional elements do not contribute to the result. The partial OLS solution calculated in Step 12 must however take into account the additional rows $\mathbf{X}$. When this equation is solved using a pre-computed Gram matrix, we simply supply the augmented Gram matrix $\mathbf{X}^\top\mathbf{X}+\delta\mathbf{I}$. When the Cholesky approach is used, it is straightforward to take an additional parameter δ into account such that the Cholesky factorization of $\mathbf{X}^\top\mathbf{X} + \delta\mathbf{I}$ is obtained. Except for these alterations to Step 12, the algorithm is run as usual. The LASSO and the Elastic Net therefore use the same underlying function (`larsen.m`) which take the additional parameter δ used in Step 12. For LASSO solutions δ is simply set to zero.

Zou and Hastie (2005) argue and provide some evidence that the double shrinkage introduced by the l_1 and l_2 has an unfortunate effect on prediction accuracy. They propose to compensate for this by multiplying the solutions $\mathbf{B}$ by a factor $(1+\delta)$, and refer to the unadjusted solutions as the Naïve Elastic Net. In some cases, the naïve solution is preferred, which consequently are obtained either by calling `larsen.m` directly or by dividing the Elastic Net solutions by $(1 + \delta)$.

The Elastic Net algorithm outputs the same model selection criteria as the LAR and LASSO algorithms. Computationally, the difference lies in the estimation of the number of degrees of freedom and the residual variance σ_ε^2. For non-zero δ, the corresponding ridge regression solution is used as a low-bias model in the estimation of the latter. Zou (2005) shows that an unbiased estimate of the number of degrees of freedom of Elastic Net solutions can be obtained by

$$\operatorname{tr}\left(\mathbf{X}_\mathcal{A}(\mathbf{X}_\mathcal{A}^\top\mathbf{X}_\mathcal{A} + \delta\mathbf{I})^{-1}\mathbf{X}_\mathcal{A}^\top\right), \tag{36}$$

which we solve efficiently using a singular value decomposition of $\mathbf{X}_\mathcal{A}$.

4.4. Sparse principal component analysis

Principal component analysis (PCA) is a linear transformation $\mathbf{S} = \mathbf{XL}$ of a mean-zero data matrix $\mathbf{X}$ where the *loading vectors* (columns) of $\mathbf{L}$ provide an orthonormal basis which successively maximizes the variance of the projected data in $\mathbf{S}$, where the *principal components* (columns) of $\mathbf{S}$ are uncorrelated, see e.g., Hastie, Tibshirani, and Friedman (2009). PCA is optimal in the sense that no linear transformation can produce a more compact representation of data given $K < p$ basis vectors. The successive maximization of variance means that the few first principal components are usually sufficient to accurately describe the data. However, each principal component is a linear combination of *all* variables in $\mathbf{X}$ and is therefore difficult to interpret and assign a meaningful label. To alleviate this, sparse PCA (SPCA) aims at upholding some or all of the properties of PCA — successive maximization of variance, independence of the loading vectors and uncorrelated principal components — while enforcing sparsity of the loading vectors such that each principal component is a linear combination of only a few of the original variables.

The algorithm for computing sparse loading vectors used in this toolbox is detailed in Zou *et al.* (2006), and uses the Elastic Net in a regression-like framework for PCA. In the spirit of this paper, we start by formulating regular PCA as the solution to a regression problem, and then add suitable constraints to obtain sparse solutions.

Viewing PCA from a compression standpoint, the objective is to find the rank-K subspace projection $\mathbf{AA}^\top$ such that $\mathbf{A}^\top\mathbf{A} = \mathbf{I}$ ($\mathbf{A}$ is $p \times K$) which reconstructs a data point $\mathbf{x}$ as well

as possible. This amounts to the following criterion,

$$\arg\min_{\mathbf{A}} \|\mathbf{X} - \mathbf{X}\mathbf{A}\mathbf{A}^\top\|_F^2, \quad \text{such that} \quad \mathbf{A}^\top\mathbf{A} = \mathbf{I}. \tag{37}$$

The solution is readily available via a singular value decomposition; let $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$, and set $\mathbf{A} = \mathbf{V}$.

Zou *et al.* (2006) show that this criterion can be relaxed into the following l_2-penalized formulation,

$$\arg\min_{\mathbf{A},\mathbf{B}} \|\mathbf{X} - \mathbf{X}\mathbf{A}\mathbf{B}^\top\|_F^2 + \delta\|\mathbf{B}\|_F^2, \quad \text{such that} \quad \mathbf{A}^\top\mathbf{A} = \mathbf{I}, \tag{38}$$

where $\mathbf{B}$ is $p \times K$ and $\|\mathbf{B}\|_F^2 = \sum_{k=1}^K \|\beta_k\|_2^2$. After normalization such that each column of $\mathbf{B}$ has unit length, the optimal solution is $\mathbf{A} = \mathbf{B} = \mathbf{V}$, the loading matrix of PCA, irrespective of the choice of δ. The role of the ridge penalty on $\mathbf{B}$ is to provide unique solutions also when $p > n$, in which case δ must be non-zero. Since the loading vectors in $\mathbf{B}$ are orthogonal, we can estimate them sequentially by

$$\arg\min_{\alpha_k,\beta_k} \|\mathbf{X} - \mathbf{X}\beta_k\alpha_k^\top\|_F^2 + \delta\|\beta_k\|_2^2, \quad \text{subject to} \quad \mathbf{A}_k^\top\mathbf{A}_k = \mathbf{I}, \tag{39}$$

where $\mathbf{A}_k$ denotes the matrix $[\alpha_1 \dots \alpha_k]$. Aiming at an algorithm for computing a sparse matrix of loadings, we will now state an alternating algorithm for optimizing the above criterion for α_k and β_k. For this purpose, we have the following result.

Lemma 4.1 *Assume* $\alpha_k^\top\alpha_k = 1$ *and fix* α_k, $\mathbf{X}$ *and* $\mathbf{Y}$. *Then, the problems*

$$\arg\min_{\beta_k} \|\mathbf{Y} - \mathbf{X}\beta_k\alpha_k^\top\|_F^2 \tag{40}$$

$$\arg\min_{\beta_k} \|\mathbf{Y}\alpha_k - \mathbf{X}\beta_k\|_2^2 \tag{41}$$

$$\tag{42}$$

have the same minimizer $\hat{\beta}_k = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{Y}\alpha_k$.

This result (with $\mathbf{Y} = \mathbf{X}$ and adding the l_2 penalty) shows that for fixed α_k, the optimal β_k is given by $\hat{\beta}_k = (\mathbf{X}^\top\mathbf{X} + \delta\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{X}\alpha_k$. If we instead fix β_k, the optimal α_k is given by the following result.

Lemma 4.2 *Let* $\mathbf{A}_{(k-1)}$ *with* $\mathbf{A}_{(k-1)}^\top\mathbf{A}_{(k-1)} = \mathbf{I}$ *be the* $(p \times k-1)$ *matrix containing the first* $k-1$ *columns of* $\mathbf{A}$. *The "fix* β_k, *solve for* α_k*"-problem can then be formulated as,*

$$\hat{\alpha}_k = \arg\min_{\alpha_k} \|\mathbf{X} - \mathbf{X}\beta_k\alpha_k^\top\|_F^2 \quad \textit{subject to} \quad \alpha_k^\top\alpha_k = 1,\ \alpha_k^\top\mathbf{A}_{(k-1)} = \mathbf{0}. \tag{43}$$

Let $\mathbf{s} = \left(\mathbf{I} - \mathbf{A}_{(k-1)}\mathbf{A}_{(k-1)}^\top\right)\mathbf{X}^\top\mathbf{X}\beta_k$. *Then,* $\hat{\alpha}_k = \mathbf{s}/\sqrt{\mathbf{s}^\top\mathbf{s}}$.

Appendix B.1 and B.2 contain proofs of the above results. If applied alternately until convergence for each principal component, we end up with the full PCA solution. This convergence is assured since penalization (39) is convex and each alternating step lead to a lower function value.

Turning to the problem of estimating sparse principal components (a sparse loading matrix), an l_1 penalty is added to the formulation in Equation (39).

$$\{\hat{\alpha}_k, \hat{\beta}_k\} = \arg\min_{\alpha_k, \beta_k} \|\mathbf{X} - \mathbf{X}\beta_k\alpha_k^\top\|_F^2 + \delta\|\beta_k\|_2^2 + \lambda\|\beta_k\|_1, \quad \text{subject to} \quad \mathbf{A}_k^\top\mathbf{A}_k = \mathbf{I}. \tag{44}$$

Using the alternating approach defined above to optimize this criterion, we see that $\hat{\alpha}_k$ is estimated as before, while $\hat{\beta}_k$ is turned from a ridge regression problem into an elastic net problem. As before the response vector is $\mathbf{X}\alpha_k$. We arrive at Algorithm 4 for computing sparse principal components. This algorithm also handles the case where the l_2 regularization parameter δ is set to infinity. The elastic net estimation of β_k then turns into a soft-thresholding rule as described in Zou *et al.* (2006). This leads to a computational advantage, which is why this option is popular for very high-dimensional data arising from e.g., image or gene expression data. It is our experience that this option also provides better solutions (in terms of explained variance for a fixed level of sparsity) in such cases.

Algorithm 4 SPCA (Zou *et al.* 2006)

Let $K < p$ be the number of sparse principal loading vectors to estimate
Let $\mathbf{A}$ be the $(p \times K)$ matrix consisting of the K first ordinary principal loading vectors
for $k = 1, \ldots, K$ **do**
  while sparse loading vector β_k has not converged **do**
    if $\delta = \infty$ **then**
      $\beta_k = \left(|\mathbf{X}^\top\mathbf{X}\alpha_k| - \lambda\right)_+ \operatorname{sign}(\mathbf{X}^\top\mathbf{X}\alpha_k)$ (Soft thresholding)
    else
      Solve the elastic net problem $\beta_k = \arg\min_\beta \|\mathbf{X}\alpha_k - \mathbf{X}\beta\|^2 + \delta\|\beta\|^2 + \lambda\|\beta\|_1$
    end if
    $\beta_k = \beta_k / \sqrt{\beta_k^\top\beta_k}$ (Normalize to unit length)
    $\alpha_k = (\mathbf{I} - \mathbf{A}_{(k-1)}\mathbf{A}_{(k-1)}^\top)\mathbf{X}^\top\mathbf{X}\beta_k$ (Update kth column of projection matrix $\mathbf{A}$)
    $\alpha_k = \alpha_k / \sqrt{\alpha_k^\top\alpha_k}$ (Normalize to unit length)
  end while
end for
Output the coefficients $\mathbf{B} = [\beta_1 \ldots \beta_K]$

Note that this algorithm is no longer convex, and may converge to local minima.

Since this is the first account of this sequential SPCA algorithm we give preliminary results of its performance and discuss advantages in relation to the previously proposed simultaneous approach (Zou *et al.* 2006).

A clear advantage of sequential estimation of components comes from running the algorithm once to estimate k components, and once to estimate $k + l$ components. The sequential approach will yield the exact same first k components in both cases whereas the simultaneous algorithm gives different results for all components.

Both algorithms are initialized with a matrix $\mathbf{A}$ equal to the loading matrix of regular PCA. The corresponding scores are ordered from high to low variance. The simultaneous approach often stray far from this initial solution and yields an arbitrary ordering in terms of variance of its components. In Sjöstrand, Stegmann, and Larsen (2006) we discuss several ways of establishing a sensible ordering of oblique components. The sequential algorithm is more

likely to produce components of decreasing variance, and we have therefore chosen to return the components as-is, in order of computation.

The sequential approach transforms one large non-convex optimization problem into several small. Convergence rates for each such problem are typically orders of magnitude higher than that of the simultaneous approach. To verify this, we conducted an experiment on a synthetic data set of 600 observations, created from 200 observations each of three sparse components with added Gaussian noise. The total number of variables in the data set ranged from 10 to 1500 with increments of 10, and we ran the sequential and simultaneous algorithms once for each choice of p. We extracted three sparse principal components and compared computation times and total adjusted variance. Figure 1 shows the results. The simultaneous algorithm was forced to give up after 1000 iterations, which occurred in a large proportion of runs. This is visible in the figure as a marked line of maximal computation times. Computation times for the sequential algorithm were lower by a factor 15–100 and with no premature terminations. The sequential algorithm is more restrictive than its sequential counterpart since previous components are fixed when estimating the next component. In our experience, one pays a small price in terms of variance for this restriction; this is shown in the right plot in Figure 1. We have not yet encountered a case were this reduction is significant.

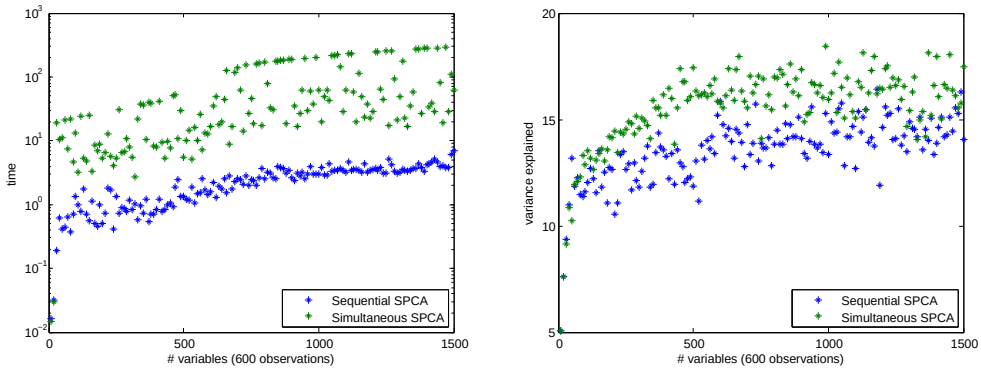


Figure 1: The left figure shows computation time for a data set with 600 observations and increasing dimensionality; 25 non-zero loadings were extracted. The sequential SPCA algorithm is faster by a factor 15–100. The right figure shows total adjusted variance for three components for the same data set. The sequential algorithm pays a small penalty for the one component at a time approach.

4.5. Sparse linear discriminant analysis

Linear Discriminant Analysis (LDA) estimates orthogonal directions β_k in which observations $\mathbf{x}_i$ belonging to one of K classes are most separated. Separation is measured as the between-class variance σ_b^2 in relation to the within-class variance σ_w^2 of the projected data $\mathbf{X}\beta_k$. Class-belongings are dummy-encoded in a $(n \times K)$ matrix $\mathbf{Y}$ where element (i, j) is 1 if the ith observation belongs to the jth class, else 0. Further, the matrix $\mathbf{D}_\pi = \frac{1}{n}\mathbf{Y}^\top\mathbf{Y}$ is a diagonal

matrix of class prior probabilities based on their frequency in $\mathbf{Y}$. Given these definitions, the matrix of class centroids is given by $\mathbf{M} = \frac{1}{n}\mathbf{D}_\pi^{-1}\mathbf{Y}^\top\mathbf{X}$, the total covariance matrix is $\Sigma = \frac{1}{n}\mathbf{X}^\top\mathbf{X}$, the between-class covariance matrix is $\Sigma_b = \mathbf{M}^\top\mathbf{D}_\pi\mathbf{M} = \frac{1}{n}\mathbf{X}^\top\mathbf{Y}(\mathbf{Y}^\top\mathbf{Y})^{-1}\mathbf{Y}^\top\mathbf{X}$, and the within-class covariance matrix is $\Sigma_w = \Sigma - \Sigma_b = \frac{1}{n}\mathbf{X}^\top(\mathbf{I} - \mathbf{Y}(\mathbf{Y}^\top\mathbf{Y})^{-1}\mathbf{Y}^\top)\mathbf{X}$. The cost function to optimize for the kth direction is,

$$\arg\max_{\beta_k} \beta_k^\top \Sigma_b \beta_k \quad \text{subject to} \quad \beta_k^\top \Sigma_w \beta_k = 1, \beta_k^\top \Sigma_w \beta_l = 0, \forall l < k. \tag{45}$$

Standard differentiation leads to a an eigenvalue problem with respect to the matrix $\Sigma_w^{-1}\Sigma_b$ which yields the full set of solutions $\mathbf{B} = \{\beta_k\}$. Classification of a new observation $\mathbf{x}$ is performed by finding the closest centroid in the derived space defined by $\mathbf{B}$.

LDA relies on the assumptions that (1) the data is normally distributed and (2) all classes have equal covariances. Although these assumptions are seldom met exactly, LDA often has as good or better performance compared to more flexible alternatives. This is in part due to the robustness of a method with few parameters to estimate. As with ordinary least squares in regression, there are situations where the inverse of the Gram matrix $\mathbf{X}^\top\mathbf{X}$ — which turns up in the estimation of regression coefficients as well as the estimation of Σ_w^{-1} — has high variance or is computationally infeasible. Similarly to ridge regression (cf., Section 4.3) the covariance matrix (or equivalently, the Gram matrix) may be replaced by a regularized variant $\Sigma + \delta\Omega$. If Ω is a positive definite matrix, then there exists a large-enough positive value of δ such that $\Sigma + \delta\Omega$ is positive definite (Petersen and Pedersen 2008). Application of this approach to LDA leads to *penalized* (linear) discriminant analysis (PDA) (Hastie, Buja, and Tibshirani 1995); the estimate of Σ_w is simply replaced by $\Sigma_w + \delta\Omega$, otherwise the calculation and application proceeds as before. In the remainder of this section, we will use $\Omega = \mathbf{I}$ which shrinks the solutions towards those obtained by assuming a spherical common covariance matrix.

An alternative route to the solutions $\mathbf{B}$ of LDA/PDA is via *optimal scoring* (Hastie, Tibshirani, and Buja 1994). The PDA optimal scoring criterion is

$$\arg\min_{\Theta,\mathbf{B}} \|\mathbf{Y}\Theta - \mathbf{X}\mathbf{B}\|_F^2 + \delta\|\mathbf{B}\|_F^2 \quad \text{subject to} \quad \Theta^\top\mathbf{D}_\pi\Theta = \mathbf{I}. \tag{46}$$

The matrix Θ is a scoring matrix, orthogonal in $\mathbf{D}_\pi$, which assigns a multiple to each column (class) in $\mathbf{Y}$. This transformation of the dummy encodings circumvents problematic situations which otherwise make a regression approach to classification difficult (Hastie *et al.* 2009). As shown in detail in Hastie *et al.* (1995) and more succinctly in Hastie *et al.* (1994), the optimal $\mathbf{B}$ are equivalent, up to a diagonal scaling matrix, to those obtained by PDA using the penalized within-class covariance matrix $\Sigma_w + \frac{\delta}{n}\mathbf{I}$. Differentiation, first with respect to $\mathbf{B}$ and then with respect to Θ gives the standard solution; first compute a multivariate ridge regression of $\mathbf{X}$ on $\mathbf{Y}$ yielding regression coefficients $\tilde{\mathbf{B}} = (\mathbf{X}^\top\mathbf{X} + \delta\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{Y}$. Θ is then obtained by an eigenanalysis of the matrix $\hat{\mathbf{Y}}\mathbf{Y}$, where $\hat{\mathbf{Y}} = \mathbf{X}\tilde{\mathbf{B}}$. The final directions $\mathbf{B}$ are finally obtained by $\mathbf{B} = \tilde{\mathbf{B}}\Theta$.

Similarly to the treatment of the PCA penalization (39), we can estimate the directions β_k sequentially since they are orthogonal. The estimation of the kth direction involves solving

$$\arg\min_{\theta_k,\beta_k} \|\mathbf{Y}\theta_k - \mathbf{X}\beta_k\|_2^2 + \delta\|\beta_k\|_2^2 \quad \text{subject to} \quad \Theta_k^\top\mathbf{D}_\pi\Theta_k = \mathbf{I}, \tag{47}$$

where Θ_k contains the k first columns of Θ. We will now describe an alternating algorithm which replaces the eigenanalysis-based recipe in the previous paragraph. This algorithm then naturally extends to *sparse* discriminant analysis, where the ridge regression estimate is replaced by an elastic net estimate. In line with Section 4.4, we first state the following Lemma.

Lemma 4.3 *Let* $\Theta_{(k-1)}$ *with* $\Theta_{(k-1)}^\top \mathbf{D}_\pi \Theta_{(k-1)} = \mathbf{I}$ *be the* $(p \times k-1)$ *matrix containing the first* $k-1$ *columns of* Θ. *The "fix* β_k*, solve for* θ_k*"-problem can then be formulated as,*

$$\hat{\theta}_k = \arg\min_{\theta_k} \|\mathbf{Y}\theta_k - \mathbf{X}\beta_k\|_2^2 \quad \textit{subject to} \quad \theta_k^\top \mathbf{D}_\pi \theta_k = 1,\ \theta_k^\top \mathbf{D}_\pi \Theta_{(k-1)} = \mathbf{0}. \tag{48}$$

Let $\mathbf{s} = \left(\mathbf{I} - \Theta_{(k-1)}\Theta_{(k-1)}^\top \mathbf{D}_\pi\right) \mathbf{D}_\pi^{-1} \mathbf{Y}^\top \mathbf{X}\beta_k$. *Then,* $\hat{\alpha}_k = \mathbf{s}/\sqrt{\mathbf{s}^\top \mathbf{D}_\pi \mathbf{s}}$.

Proof Appendix B.2 gives a proof for Lemma 4.2; the proof for this lemma is equivalent, except the trivial addition of $\mathbf{D}_\pi$. The solution to this Lemma is also given — sans proof — in Clemmensen *et al.* (2011).

The "fix θ_k, solve for β_k" problem is solved by the ridge regression estimate $\beta_k = (\mathbf{X}^\top \mathbf{X} + \delta \mathbf{I})^{-1} \mathbf{X}^\top \mathbf{Y} \theta_k$.

We initialize Θ to the size $(K \times K)$ identity matrix. The directions are then obtained sequentially by alternating the estimation of β_k and θ_k until convergence. Convergence to a global optimum for each direction is guaranteed by the convexity of the cost function and its constraints and since each alteration is sure to lower the cost.

With this algorithm in place, it is now straight-forward to extend it to include an l_1 penalty which promotes directions β_k which are sparse. This means that the space $\mathbf{B}$ in which the classification is carried out, consists of a subset of the available variables. As with all sparse methods, the possible benefits are ease of interpretation and (non-linear) suppression of noise. The l_1, l_2-regularized criterion is,

$$\{\hat{\theta_k}, \hat{\beta_k}\} = \arg\min_{\theta_k, \beta_k} \|\mathbf{Y}\theta_k - \mathbf{X}\beta_k\|_2^2 + \delta\|\beta_k\|_2^2 + \delta_k\|\beta_k\|_1 \quad \text{subject to} \quad \Theta_k^\top \mathbf{D}_\pi \Theta_k = \mathbf{I}, \tag{49}$$

which turns the ridge regression estimation of β_k from penalization (47) into an Elastic Net estimate. The resulting algorithm for sparse discriminant analysis is stated in Algorithm 5.

The normalization in Step 7 helps to avoid multiplicative drift towards the trivial solution where $\theta_k = \mathbf{0}$ and $\beta_k = \mathbf{0}$. To avoid additive drift we also require $\sum_i (\mathbf{D}_\pi \theta_k)_i = 0$, i.e., that θ_k is zero-mean in $\mathbf{D}_\pi$. In previous treatments of optimal scoring (see e.g., Clemmensen *et al.* (2011)), the columns of Θ are explicitly forced to be orthogonal to a vector of ones. However, it turns out that Step 6 implicitly guarantees this since $\mathbf{D}_\pi \theta_k = \mathbf{Y}^\top \mathbf{X}\beta_k - \mathbf{D}_\pi \Theta_{(k-1)} \Theta_{(k-1)}^\top \mathbf{D}_\pi \mathbf{D}_\pi^{-1} \mathbf{Y}^\top \mathbf{X}\beta_k$. The first term is clearly mean-zero since $\mathbf{X}$ is centered. The second term is mean-zero if $\mathbf{D}_\pi \Theta_{(k-1)} \Theta_{(k-1)}^\top \mathbf{D}_\pi \mathbf{D}_\pi^{-1}$ is mean zero. $\mathbf{D}_\pi \Theta_{(k-1)} \Theta_{(k-1)}^\top \mathbf{D}_\pi$ is the outer product of two mean-zero matrices and results in a mean-zero matrix. Multiplying this with the diagonal matrix $\mathbf{D}_\pi^{-1}$ does not change this property.

To allow for a more flexible model of the density of each class one may model each class a mixture of Gaussian distributions. Clemmensen *et al.* (2011) describe an extension of the described algorithm which implements this.

Algorithm 5 SLDA (Clemmensen *et al.* 2011)

Let Q be the number of classes and $K < Q$ the number of discriminative directions
Initialize $\mathbf{Y}$ an $n \times Q$ matrix of indicator variables with $Y_{i \in \text{class}_j j} = 1$, $\mathbf{D}_\pi = \frac{1}{n}\mathbf{Y}^\top\mathbf{Y}$, and $\Theta = \mathbf{I}_{(Q \times K)}$
for $k = 1, \ldots, K$ **do**
  while sparse discriminative direction β_k has not converged **do**
    Solve the elastic net problem $\beta_k = \arg\min_\beta \|\mathbf{Y}\theta_k - \mathbf{X}\beta\|^2 + \delta\|\beta\|^2 + \delta\|\beta\|_1$
    $\theta_k = (\mathbf{I} - \Theta_{(k-1)}\Theta_{(k-1)}^\top\mathbf{D}_\pi)\mathbf{D}_\pi^{-1}\mathbf{Y}^\top\mathbf{X}\beta_k$ (Update kth column of Θ)
    $\theta_k = \theta_k / \sqrt{\theta_k^\top \mathbf{D}_\pi \theta_k}$ (Normalize to unit length)
  end while
end for
Output the coefficients $\mathbf{B} = [\beta_1 \ldots \beta_K]$

5. Using the SpaSM toolbox

This section gives a brief overview of usage of the SpaSM toolbox. The examples covered can all be found in the source code. Issuing the command `help nameOfFunction` yields a detailed overview of the input and output to the functions, where `nameOfFunction` is a function in the toolbox, e.g., `forwardselection`.

After the code has been downloaded[1] one can add the path to the SpaSM directory in MATLAB to access the functions and scripts. The examples contain random generated data sets, the code for generating the data sets is also included in the examples.

A streamlined version of the examples is contained in the file `demo.m`. This includes the code to generate the data sets with appropriate seeds and all figures.

5.1. Forward selection usage example

The example is contained in the file `example_forwardselection.m`. First a simulated data set is generated and preprocessed. This data set is referred to as simulated data 1. The code for generating the data set and the appropriate seed for the random number generator are inside the file.

The data set is described as follows. A set of six correlated mean zero predictors are generated from a multivariate random distribution. The covariance matrix has 1 on the diagonal and 0.6 on the off-diagonal entries. The response is generated as a linear combination of the first three predictors and i.i.d. Gaussian noise with standard deviation 2. The predictors are then centered and scaled to unit euclidian length and the response is centered. The forward selection algorithm is then run with the following command:

```
>> [beta info] = forwardselection(X, y, 0, true, true);
Step Added Active set size
1 2 1
2 3 2
3 1 3
4 6 4
```

[1] Link to code http://www.imm.dtu.dk/projects/spasm/

```
5 4 5
6 5 6
```

The inputs are the predictors variables `X`, the response variable `y`, a scalar `STOP` (which triggers early stopping if non-zero), a boolean variable `STOREPATH` (which stores the values of the coefficients in the model estimated in each iteration) and a boolean variable `VERBOSE` (which if false suppresses output to the command line).

The output to the command line shows what happened in each iteration, i.e., which variable was added and the size of the active set. The `beta` in the output is a matrix containing the coefficients in the model. Column i contains the parameters found in iteration i.

```
>> beta

beta =

         0         0         0    9.4249   10.4411   10.2215   10.4525
         0   26.5885   16.2983   12.4341   13.3253   12.5398   12.5018
         0         0   15.2383   12.0823   13.2119   13.0880   13.4359
         0         0         0         0         0    2.6190    2.9017
         0         0         0         0         0         0   -1.2487
         0         0         0         0   -3.5534   -4.6301   -4.3471
```

The `info` variable in the output is a struct containing values for information criteria (see Section 4.1), degrees of freedom, steps and relative size of coefficients compared to a low bias model.

```
>> info

info =

    steps: 6
       df: [0 1 2 3 4 5 6]
       Cp: [241.363 51.658 19.405 8.533 9.012 10.182 12.000]
      AIC: [1258.8e+03 559.269 440.327 400.236 402.003 406.317 413.022]
      BIC: [1258.8e+03 568.876 459.542 429.057 440.431 454.353 470.664]
        s: [0 0.592 0.703 0.756 0.903 0.960 1.000]
```

Now we can find the best fitted model according to AIC:

```
% Find best fitting model
[bestAIC bestIdx] = min(info.AIC);
best_s = info.s(bestIdx);
```

This is a model containing variables 1,2 and 3. All information criteria select the right model. Figure 2 shows how the values of the coefficients in the models change over the iterations.

5.2. Least Angle Regression usage example

We continue to use simulated data set 1 as for the forward selection algorithm. The example is contained in `example_lar.m`. We call the LAR algorithm with the following command:

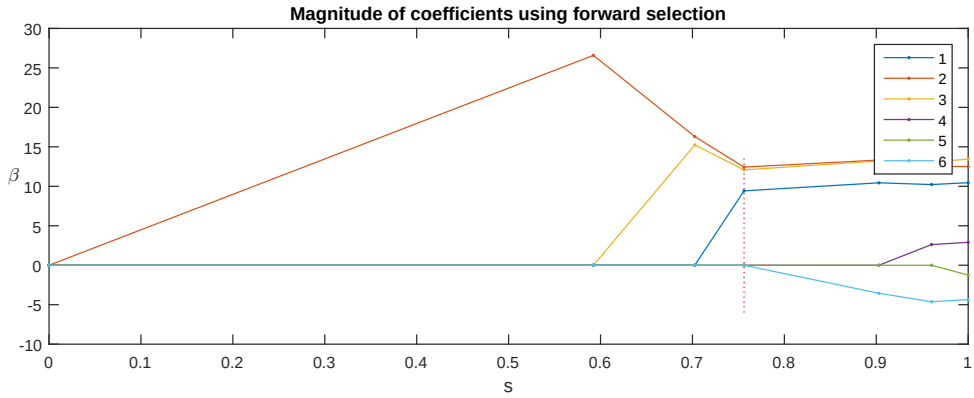


Figure 2: Value of parameters in models changing through the iterations of the forward selection algorithm on simulated data set 1. The x-axis shows the relative size of the parameter vector β compared to a low bias model. The dotted red vertical line indicates the model selected by AIC.

```
>> [beta info] = lar(X, y, 0, true, true);
Step Added Active set size
1 2 1
2 3 2
3 1 3
4 4 4
5 6 5
6 5 6
```

The input/output to the function and the output to the command line is virtually the same as for the forward selection algorithm. The `beta` matrix in the output is:

```
>> beta

beta =

         0         0         0    8.2840    8.8654    9.6638   10.4525
         0    1.0599    4.5161   11.4756   11.8199   12.2437   12.5018
         0         0    3.4562   11.0381   11.5741   12.4654   13.4359
         0         0         0         0    0.6168    1.7956    2.9017
         0         0         0         0         0         0   -1.2487
         0         0         0         0         0   -2.7259   -4.3471
```

Note that the model in the fourth column has the same parameters non-zero as in the forward selection algorithm. The values of the parameters are not the same and that is due to the fact that a new variable is added to the active set when it becomes equally important as an inactive variable.

The `info` struct holds the same information as for the forward selection algorithm. The output is:

```
>> info

info =

    steps: 6
       df: [0 1 2 3 4 5 6]
       Cp: [241.3628 228.3831 145.5324 10.5697 10.7091 10.6094 12.0000]
      AIC: [1.2588e+03 1.2110e+03 905.4487 407.7472 408.2615 407.8936 413.0218]
      BIC: [1.2588e+03 1.2206e+03 924.6628 436.5685 446.6898 455.9289 470.6643]
        s: [0 0.0236 0.1776 0.6861 0.7324 0.8665 1.0000]
```

The first and last values of the information criteria are the same as for the forward selection but note that the intermediate values are quite different due to the different step sizes. Figure 3 shows a similar evolution of the parameters through the iterations as can be seen in Figure 2 for the forward selection algorithm. Note that the magnitude of the parameters of the selected model is relatively smaller than the one from forward selection.

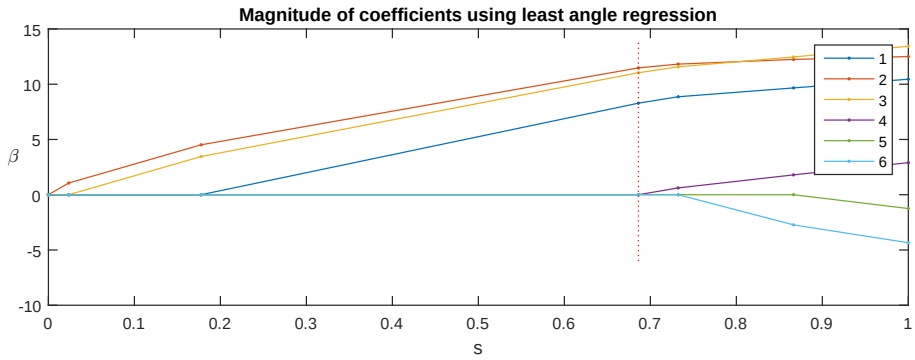


Figure 3: Value of parameters in models changing through the iterations of the LAR algorithm on simulated data set 1. The x-axis shows the relative size of the parameter vector β compared to a low bias model. The dotted red vertical line indicates the model selected by AIC.

5.3. The LASSO usage example

Again we use simulated data set 1 to try out the LASSO algorithm. The example is contained in the file `example_lasso.m`. The command and command line output are the following:

```
>> [beta info] = lasso(X, y, 0, true, true);
Step Added Dropped Active set size
1 2 1
2 3 2
3 1 3
4 4 4
5 6 5
6 5 6
```

The only noticeable difference is the extra column named dropped. In this example no variable is dropped from the active set, but that happens when a parameter value crosses zero. The input to the `lasso` function is the same as for `lar`. The values of `beta` and `info` are the same, since non of the active variables become zero throughout the iterations as can be seen in Figure 3.

5.4. The elasticnet usage example

The elastic net can handle data set with more variables than observations, given that the regularization parameter for the L_2 norm of the parameters is strictly positive. Here we create another simulated data set. We refer to this data set as simulated data set 2.

The data set is created as follows. We generate 30 observations of 40 variables. The predictors are sampled from a multivariate normal distribution with a similar covariance matrix as the simulated data set 1. The response is again a linear combination of the first three variables and i.i.d. Gaussian noise with standard deviation 0.5.

The example is contained in `example_elasticnet.m`. The algorithm is called as follows:

```
delta = 1e-3;
[beta info] = elasticnet(X, y, delta, 0, true, true);
Step Added Dropped Active set size
1 1 1
2 3 2
3 2 3
4 26 4
5 9 5
6 31 6
...
```

Here `delta` is the regularization parameter for the L_2 norm. The additional output to the command line is omitted here. The values of the coefficients in the models through the iterations can be seen in Figure 4.

5.5. Sparse principal component analysis usage example

To demonstrate the usage of SPCA we generate another simulated data set with 1500 observations and 500 variables. This data set is referred to as simulated data set 3. First we generate 3 principal components and add i.i.d. Gaussian noise to them to generate 500 noisy versions of each. The three components can be seen without noise in Figure 5 on the left, and all the data with the noise is on the right. Note that each components has at least half the values as zero or close to zero. Code to generate simulated data set 3 with the appropriate seed can be found in the corresponding file `example_spca.m`. The data is stored in a matrix `X`.

To run the method we need to supply a few parameters. The parameters and call to the function in MATLAB are the following:

```
K = 3;
delta = inf;
```

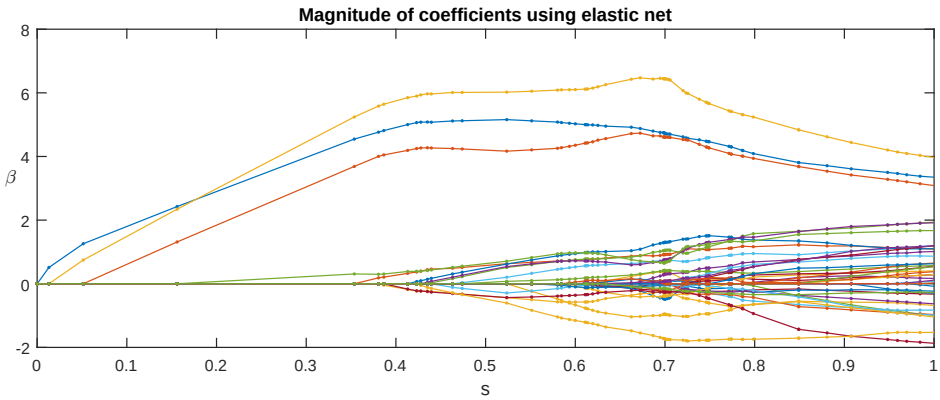


Figure 4: Value of parameters in models changing through the iterations of the elastic net algorithm on simulated data set 2. The x-axis shows the relative size of the parameter vector β compared to a low bias model.

```
stop = -[250 125 100];
maxiter = 3000;
convCriterion = 1e-9;
verbose = true;

[SL SD] = spca(X, [], K, delta, stop, maxiter, convCriterion, verbose);
```

The second argument can be the Gram matrix, otherwise the empty matrix is supplied. When the `stop` variable is negative, the absolute value is the number of desired non-zero values in each component. This is an advantage to other algorithms, where sparsity cannot directly be specified. The results compared to traditional PCA can be seen in Figure 6.

5.6. Sparse linear discriminant analysis usage example

To demonstrate the usage of SLDA we create yet another simulated data set, referred to as simulated data set 4. The training part of the data set consists of three classes with 100 observations from each class with 150 variables, the test part of the data set is identically sampled and of the same size. The data is generated from a multivariate normal distribution. The mean for the first class has the first ten variables as 0.6 and the rest as zero. The mean for the second class has variables 11-20 as 0.6 and the rest as zero and the third mean has variables 21-30 as 0.6 and the rest as zero. The classes have the same covariance matrix where there is 1 on the diagonal and 0.6 on the off diagonal entries. The example is contained in the file `example_slda.m`. We can now run the algorithm with the following command:

```
[B theta] = slda(X, Y, delta, stop, Q, maxiter, tol, true);
Estimating direction 1
  Iteration: 10, convergence criterion: 0.025816
  Iteration: 20, convergence criterion: 2.7872e-06
  Iteration: 22, convergence criterion: 5.1135e-07
Estimating direction 2
  Iteration: 3, convergence criterion: 6.019e-30
```

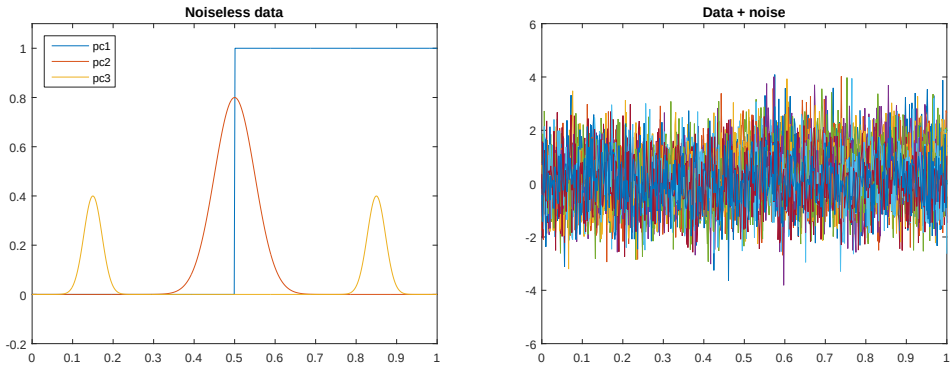


Figure 5: The left figure shows the three generated components, each consisting of 500 values. The right figure shows 5 noisy versions of each of the three components from simulated data set 3.

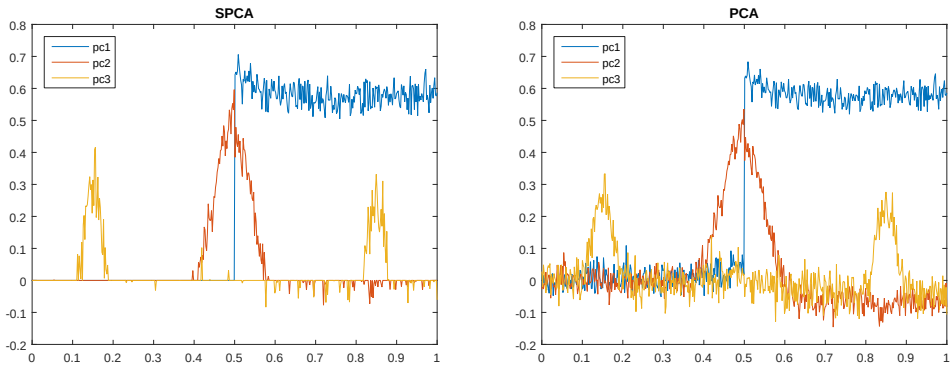


Figure 6: The left figure shows the results obtained by SPCA on simulated data set 3. The right figure shows the results from PCA.

The parameters for the function call are similar to the ones for SPCA. The parameter `Q` is the number of desired discriminative directions, in this case 2. The output consists of `B`, the regression parameters, and `theta`, the optimal scores.

We can now project the data onto the columns spanned by `B` and perform LDA. The training error is is 3.0% and test error is 5.3%. When we compare this to doing LDA on the raw data we get a training error of 1.0% and test error of 12.0%.

6. Example studies

The following examples elaborate on the differences of some of the algorithms presented in the SpaSM toolbox on real data sets. The first example demonstrates the difference of the LAR and LASSO algorithms, where the path of one of the coefficients crosses zero through the iterations. The second example demonstrates the difference of using PCA and SPCA on a shape data set consisting of male and female silhouettes. SPCA provides more local deviations from the mean which are easier to interpret.

6.1. Regression on Diabetic data

To demonstrate the difference of the LASSO and the LAR algorithm we decided to use the Diabetes data set[2], see Efron *et al.* (2004). This data set is well known in the literature and it has one variable in it which changes sign through the iterations of LAR. This is what we need to demonstrate the difference between the two methods. The example is contained in the file `example_LarLasso.m`.

The data set includes 10 predictors, 1 response and 442 observations. They are measurements on diabetic patients and the response is a quantitative measure of disease progression 1 year after the baseline.

The coefficient paths for the two methods can be seen in Figure 7. The plots are identical except for the fact that in the LASSO algorithm a variable is removed from the active set when it becomes zero.

6.2. PCA and SPCA on Silhouette data

This shape data set consists of silhouettes of 20 male and 19 female adults and was first presented in Thodberg and Olafsdottir (2003). Each silhouette consists of 65 points in 2D giving a total of 130 variables for each observation. The data has been aligned with Procrustes analysis prior to using the SpaSM toolbox. The example is contained in the file `example_spca_silDat.m`.

Running PCA shows that the first three components explain 82.9% of the variation in the data. Three components from SPCA with 40 non-zero variables each explain 65.27% of the variation in the data. The results can be seen in Figure 8.

The components obtained by PCA yield variation all over the silhouette. SPCA extracts the variables that explain most of the variation with the restriction that some of them need to be zero. These sparse components yield more local deformations on the silhouette, which are easier to interpret and visually compare.

7. Collaboration and verification

We use tools and principles from software engineering in the development of this toolbox. A server-based repository (Apache Subversion (SVN) (Apache Software Foundation 2011)) allows toolbox authors to download (Update in SVN terms) the latest toolbox snapshot, apply changes and then upload (Commit) to the server when finished. Simultaneous editing of files

[2] `http://web.stanford.edu/~hastie/Papers/LARS/diabetes.data`

is also possible where overlapping changes are merged in an intuitive way when committing changes back to the repository.

In software engineering *continuous integration* refers to the practice of committing small changes often to the repository, rather than scarce large updates. This keeps the effort required to merge changes from different authors to a minimum. To further improve the quality and effectiveness of the development, we employ *unit testing*. In parallel with the development of each toolbox entity, we develop several test scripts, each testing the one part (unit) of the code. As an example, one test file asserts (using the MATLAB `assert` command) that the elastic net with $\lambda = \delta = 0$ equals the ordinary least squares solution. Although we currently have no automated procedure, we aim to run all such unit test files each time changes are uploaded to the repository. In this way, we get a strong indication to whether the new code is working as expected, and that uploaded changes did not break code that was working in an earlier version of the toolbox. Tables 1 and 2 list all relevant unit tests.

Other means of verification we have used are *code walkthrough*, a line-by-line inspection of finished code, and deployment of beta releases of the toolbox.

A. Sparse regression with orthogonal predictors

Several methods have simple closed-form solutions in cases where the predictor variables are orthogonal and have Euclidean length 1. Often, the estimation can be split into p separate problems, one for each β_i. We will quickly review how this works for the LASSO, the treatment is similar for the Elastic Net and LAR. We use this property for testing purposes, cf., Table 1.

$$\|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda\|\beta\|_1 \Longleftrightarrow \|\mathbf{y} - \mathbf{x}_i\beta_i\|^2 + \lambda|\beta_i|, \forall i. \tag{50}$$

Optimizing the expression for a single $\hat{\beta}_i$. involves taking first derivatives and setting to zero,

$$-2\mathbf{x}_i^\top(\mathbf{y} - \mathbf{x}_i\beta_i) + \lambda \cdot \text{sign}(\beta_i) = 0, i \in \mathcal{A}. \tag{51}$$

Using $\mathbf{x}_i^\top\mathbf{y} = \beta_i^{OLS}$ and $\mathbf{x}_i^\top\mathbf{x}_i = 1$ we have,

$$-2\beta_i^{OLS} + 2\beta_i + \lambda \cdot \text{sign}(\beta_i) = 0, i \in \mathcal{A}. \tag{52}$$

For sufficiently large values of λ, β_i will shrink to exactly zero. For any other value of λ, β_i will agree in sign with β_i^{OLS}. Therefore, we have,

$$\beta_i = \text{sign}(\beta_i^{OLS})\left(|\beta_i^{OLS}| - \frac{\lambda}{2}\right)^+, \forall i, \tag{53}$$

where $(\cdot)^+$ denotes the hinge function $\max(\cdot, 0)$.

B. Proofs

B.1. Proof of Lemma 4.1

Unit	**Test**	**Acceptance criterion**
LAR	Make sure the full LAR model is equal to the ordinary least squares model	Results are equal
LAR	Make sure LAR and LASSO are equal in cases where no variables are dropped in the LASSO	Results are equal
LAR	Profile code on $n \gg p$ and $p \gg n$ data sets.	Code has no apparent bottlenecks
LASSO	Make sure the full LASSO model is equal to the ordinary least squares model	Results are equal
LASSO	Run with a data set with orthogonal predictor variables. Compare to soft thresholding. Cf., Appendix A	Results are equal
LASSO	Profile code on $n \gg p$ and $p \gg n$ data sets.	Code has no apparent bottlenecks
Elastic net	Make sure the full Elastic Net model is equal to the corresponding Ridge Regression model.	Results are equal
Elastic net	Run with a data set with orthogonal predictor variables. Compare to soft thresholding.	Results are equal
Elastic net	Compare results with running LASSO with an Elastic Net-style augmented data matrix	Results are equal
Elastic net	Profile code on $n \gg p$ and $p \gg n$ data sets.	Code has no apparent bottlenecks
SPCA	Compare the full SCPA model with that of a regular PCA. Try different values of δ.	Results are equal regardless of the value of δ
SPCA	Profile code on $n \gg p$ and $p \gg n$ data sets.	Code has no apparent bottlenecks
SLDA	Assert that the resulting optimal scores $\mathbf{Z} = \mathbf{Y}\theta$ are orthogonal	$\mathbf{Z}^\top \mathbf{Z}/n = \mathbf{I}$
SLDA	Compare the results of SLDA with no l_1 constraint to ridge regression on the matrix $\mathbf{Y}\theta$	Results are equal

Table 1: Toolbox unit tests (part 1).

Unit	Test	Acceptance criterion
SLDA	Compare the results of SLDA with no l_1 constraint to penalized discriminant analysis (LDA using the within-class covariance matrix $\Sigma_W + \frac{\delta}{n}\mathbf{I}$	Results are equal
SLDA	Profile code on $n \gg p$ and $p \gg n$ data sets.	Code has no apparent bottlenecks
cholinsert	Compare updates of the Cholesky factorization to a direct Cholesky factorization of the corresponding matrices	Results are equal
choldelete	Compare downdates of the Cholesky factorization to a direct Cholesky factorization of the corresponding matrices	Results are equal

Table 2: Toolbox unit tests (part 2).

Using $\mathrm{tr}(\mathbf{AB}) = \mathrm{tr}(\mathbf{BA})$, $\mathrm{tr}(\mathbf{A}+\mathbf{B}) = \mathrm{tr}(\mathbf{A}) + \mathrm{tr}(\mathbf{B})$ and $\alpha_k^\top \alpha_k = 1$ we have,

$$\|\mathbf{X} - \mathbf{X}\beta_k\alpha_k^\top\|_F^2 = \tag{54}$$

$$\mathrm{tr}\left(\mathbf{X}^\top\mathbf{X} + \alpha_k\beta_k^\top\mathbf{X}^\top\mathbf{X}\beta_k\alpha_k^\top - 2\mathbf{X}^\top\mathbf{X}\beta_k\alpha_k^\top\right) = \tag{55}$$

$$\mathrm{tr}(\mathbf{X}^\top\mathbf{X}) + \mathrm{tr}(\mathbf{X}\beta_k\alpha_k^\top\alpha_k\beta_k^\top\mathbf{X}^\top) - 2\alpha_k^\top\mathbf{X}^\top\mathbf{X}\beta_k = \tag{56}$$

$$\mathrm{tr}(\mathbf{X}^\top\mathbf{X}) + \mathrm{tr}(\mathbf{X}\beta_k\beta_k^\top\mathbf{X}^\top) - 2\alpha_k^\top\mathbf{X}^\top\mathbf{X}\beta_k = \tag{57}$$

$$\mathrm{tr}(\mathbf{X}^\top\mathbf{X}) + \beta_k^\top\mathbf{X}^\top\mathbf{X}\beta_k - 2\alpha_k^\top\mathbf{X}^\top\mathbf{X}\beta_k, \tag{58}$$

which clearly has the same minimizing β_k as

$$\|\mathbf{X}\alpha_k - \mathbf{X}\beta_k\|_2^2 = \tag{59}$$

$$\alpha_k^\top\mathbf{X}^\top\mathbf{X}\alpha_k + \beta_k^\top\mathbf{X}^\top\mathbf{X}\beta_k - 2\alpha_k^\top\mathbf{X}^\top\mathbf{X}\beta_k. \tag{60}$$

Differentiation of any of the expressions gives $\hat{\beta}_k = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top(\mathbf{X}\alpha_k)$. This proof is also detailed in a slightly different context by Zou *et al.* (2006).

B.2. Proof of Lemma 4.2

Incorporating the constraints into the cost function in Equation (43) using Lagrange multipliers λ (length $k-1$ vector) and γ (scalar) the problem becomes

$$\arg\min_{\alpha_k} \mathrm{tr}\left[\alpha_k\beta_k^\top\mathbf{X}^\top\mathbf{X}\beta_k\alpha_k^\top - 2\alpha_k\beta_k^\top\mathbf{X}^\top\mathbf{X} + \mathbf{X}^\top\mathbf{X}\right] + \alpha_k^\top\mathbf{A}_{(k-1)}\lambda + \gamma(\alpha_k^\top\alpha_k - 1).$$

Differentiating and setting to zero, and solving for α_k leads to

$$\hat{\alpha}_k = \frac{1}{\beta_k^\top\mathbf{X}^\top\mathbf{X}\beta_k + \gamma}\left[\mathbf{X}^\top\mathbf{X}\beta_k - \frac{1}{2}\mathbf{A}_{(k-1)}\lambda\right], \tag{61}$$

or equivalently,

$$\hat{\alpha}_k = \frac{1}{\beta}\left[\mathbf{X}^\top\mathbf{X}\beta_k - \mathbf{A}_{(k-1)}\alpha\right]. \tag{62}$$

The orthogonality constraints give

$$\mathbf{A}_{(k-1)}^{\top}\hat{\alpha}_k = 0 \Leftrightarrow \frac{1}{\beta}\left[\mathbf{A}_{(k-1)}^{\top}\mathbf{X}^{\top}\mathbf{X}\beta_k - \alpha\right] = 0 \Leftrightarrow \tag{63}$$
$$\alpha = \mathbf{A}_{(k-1)}^{\top}\mathbf{X}^{\top}\mathbf{X}\beta_k.$$

Inserting this expression for α into Equation (62) and simplifying gives

$$\hat{\alpha}_k = \frac{1}{\beta}\left(\mathbf{I} - \mathbf{A}_{(k-1)}\mathbf{A}_{(k-1)}^{\top}\right)\mathbf{X}^{\top}\mathbf{X}\beta_k \equiv \frac{1}{\beta}\mathbf{s}. \tag{64}$$

Finally, the constraint $\alpha_k^{\top}\alpha_k = 1$ gives $\beta = \sqrt{\mathbf{s}^{\top}\mathbf{s}}$ such that $\hat{\alpha}_k = \mathbf{s}/\sqrt{\mathbf{s}^{\top}\mathbf{s}}$. In practice, we first calculate $\mathbf{s}$ and then normalize this vector to unit length.

References

Alsahaf A (2015). "Sparse Principal Component Analysis using Alternating Maximization." MATLAB Central File Exchange, URL http://www.mathworks.com/matlabcentral/fileexchange/47481-amjams-spca-am.

Apache Software Foundation (2011). "Apache Subversion." http://subversion.apache.org.

Breheny P, Huang J (2011). "Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection." *The annals of applied statistics*, **5**(1), 232.

Cai T, Liu W, Luo X (2011). "A constrained l_1 minimization approach to sparse precision matrix estimation." *Journal of the American Statistical Association*, **106**(494), 594–607.

Candes E, Tao T (2007). "The Dantzig selector: Statistical estimation when p is much larger than n." *The Annals of Statistics*, pp. 2313–2351.

Clemmensen L, contributions by Max Kuhn (2015). ***sparseLDA**: Sparse Discriminant Analysis*. R package version 0.1-7, URL http://CRAN.R-project.org/package=sparseLDA.

Clemmensen L, Hastie T, Witten D, Ersbøll B (2011). "Sparse Discriminant Analysis." *Technometrics*, **53**(4), 406–413.

Efron B, Hastie T, Johnstone I, Tibshirani R (2004). "Least Angle Regression." *The Annals of Statistics*, **32**(2), 407–451.

Friedman J, Hastie T, Tibshirani R (2008). "Sparse inverse covariance estimation with the graphical lasso." *Biostatistics*, **9**(3), 432–441.

Friedman J, Hastie T, Tibshirani R (2014). ***glasso**: Graphical lasso- estimation of Gaussian graphical models*. R package version 1.8, URL https://CRAN.R-project.org/package=glasso.

Friedman JH, Hastie T, Tibshirani R (2010). "Regularization Paths for Generalized Linear Models via Coordinate Descent." *Journal of Statistical Software*, **33**(1), 1–22. ISSN 1548-7660. URL http://www.jstatsoft.org/v33/i01.

Fu WJ (1998). "Penalized Regressions: The Bridge Versus the Lasso." *Journal of Computational and Graphical Statistics*, **7**(3), 397.

Goeman JJ (2010). "L1 penalized estimation in the Cox proportional hazards model." *Biometrical journal*, **52**(1), 70–84.

Hastie T, Buja A, Tibshirani R (1995). "Penalized Discriminant Analysis." *The Annals of Statistics*, **23**, 73–102.

Hastie T, Efron B (2013). ***lars**: Least Angle Regression, Lasso and Forward Stagewise*. R package version 1.2, URL http://CRAN.R-project.org/package=lars.

Hastie T, Rosset S, Tibshirani R, Zhu J (2004). "The Entire Regularization Path for the Support Vector Machine." *Journal of Machine Learning Research*, **5**, 1391–1415.

Hastie T, Tibshirani R, Buja A (1994). "Flexible Discriminant Analysis by Optimal Scoring." *Journal of the American Statistical Association*, **89**(428), 1255–1270.

Hastie T, Tibshirani R, Friedman J (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2 edition. Springer-Verlag.

He S (2011). ***spaceExt**: Extension of SPACE*. R package version 1.0, URL https://CRAN.R-project.org/package=spaceExt.

Henson R (2013). *MATLAB R-link*. Retrieved June, 2016, URL https://se.mathworks.com/matlabcentral/fileexchange/5051-matlab-r-link.

Hoerl AE, Kennard RW (1970). "Ridge regression: Biased Estimation from Nonorthogonal Problems." *Technometrics*, **12**(1), 55–67. ISSN 00401706.

Kim SS (2009). "LARS algorithm." MATLAB Central File Exchange, URL http://www.mathworks.com/matlabcentral/fileexchange/23186-lars-algorithm.

Kraemer N, Schaefer J, Boulesteix AL (2009). "Regularized Estimation of Large-Scale Gene Regulatory Networks using Gaussian Graphical Models." *BMC Bioinformatics*, **10**(384).

Le Y, Hastie T (2014). "Sparse Quadratic Discriminant Analysis and Community Bayes." *arXiv preprint arXiv:1407.4543*.

Li X, Zhao T, Wang L, Yuan X, Liu H (2014). ***flare**: Family of Lasso Regression*. R package version 1.5.0, URL https://CRAN.R-project.org/package=flare.

Liu J, Ji S, Ye J, *et al.* (2009). "SLEP: Sparse learning with efficient projections." *Arizona State University*, **6**, 491.

Mai Q, Yang Y, Zou H (2015a). "Multiclass Sparse Discriminant Analysis." *arXiv preprint arXiv:1504.05845*.

Mai Q, Yang Y, Zou H (2015b). ***msda**: Multi-Class Sparse Discriminant Analysis*. R package version 1.0.2, URL http://CRAN.R-project.org/package=msda.

Meinshausen N, Bühlmann P (2006). "High-dimensional graphs and variable selection with the lasso." *The annals of statistics*, pp. 1436–1462.

Osborne MR, Presnell B, Turlach BA (2000). "A new Approach to Variable Selection in Least Squares Problems." *IMA Journal of Numerical Analysis*, **20**(3), 389–403.

Pang H, Qi D, Liu H, Vanderbei R (2016). ***fastclime**: A Fast Solver for Parameterized LP Problems, Constrained L1 Minimization Approach to Sparse Precision Matrix Estimation and Dantzig Selector*. R package version 1.4.1, URL https://CRAN.R-project.org/package=fastclime.

Park MY, Hastie T (2007). "L1 Regularization Path Algorithm for Generalized Linear Models." *Journal of the Royal Statistical Society B*, **69**(4), 659–677.

Petersen KB, Pedersen MS (2008). "The Matrix Cookbook." *Technical report*, Technical University of Denmark.

Qian J, Hastie T, Friedman J, Tibshirani R, Simon N (2013). "Glmnet for MATLAB." URL http://www.stanford.edu/~hastie/glmnet_matlab/.

Richtárik P, Takáč M, Ahipaşaoğlu SD (2012). "Alternating maximization: unifying framework for 8 sparse PCA formulations and efficient parallel codes." *arXiv preprint arXiv:1212.4137*.

Rosset S, Zhu J (2007). "Piecewise Linear Regularized Solution Paths." *The Annals of Statistics*, **35**(3), 1012–1030.

Sjöstrand K, Stegmann MB, Larsen R (2006). "Sparse Principal Component Analysis in Medical Shape Modeling." volume 6144. SPIE.

Sun T (2013). ***scalreg**: Scaled sparse linear regression*. R package version 1.0, URL https://CRAN.R-project.org/package=scalreg.

Sun T, Zhang CH (2012). "Scaled sparse linear regression." *Biometrika*, p. ass043.

The MathWorks Inc (2010). "MATLAB version 7.11.0.584 (R2010b), 64-bit."

Thodberg HH, Olafsdottir H (2003). "Adding Curvature to Minimum Description Length Shape Models." In *British Machine Vision Conference, BMVC*.

Tibshirani R (1996). "Regression Shrinkage and Selection via the LASSO." *Journal of the Royal Statistical Society B*, **58**(1), 267–288.

Witten D (2015). ***penalizedLDA**: Penalized Classification using Fisher's Linear Discriminant*. R package version 1.1, URL http://CRAN.R-project.org/package=penalizedLDA.

Witten DM, Tibshirani R (2011). "Penalized classification using Fisher's linear discriminant." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **73**(5), 753–772.

Zhang CH (2010). "Nearly unbiased variable selection under minimax concave penalty." *The Annals of statistics*, pp. 894–942.

Zou H (2005). *Some Perspectives of Sparse Statistical Modeling*. Ph.D. thesis, Stanford University.

Zou H (2006). "The adaptive lasso and its oracle properties." *Journal of the American statistical association*, **101**(476), 1418–1429.

Zou H, Hastie T (2005). "Regularization and Variable Selection via the Elastic Net." *Journal of the Royal Statistical Society B*, **67**(2), 301–320.

Zou H, Hastie T, Tibshirani R (2006). "Sparse Principal Component Analysis." *Journal of Computational and Graphical Statistics*, **15**(2), 265.

Zou H, Hastie T, Tibshirani R (2007). "On the "Degrees of Freedom" of the Lasso." *The Annals of Statistics*, **35**(5), 2173–2192.

Affiliation:

Karl Sjöstrand
EXINI Diagnostics AB
Ideon Science Park
Gateway, Scheelevägen 27
SE-223 70 Lund, Sweden
E-mail: karl.sjostrand@exini.com
URL: http://www.imm.dtu.dk/projects/spasm/

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
MMMMMM YYYY, Volume VV, Issue II *Submitted:* yyyy-mm-dd
doi:10.18637/jss.v000.i00 *Accepted:* yyyy-mm-dd

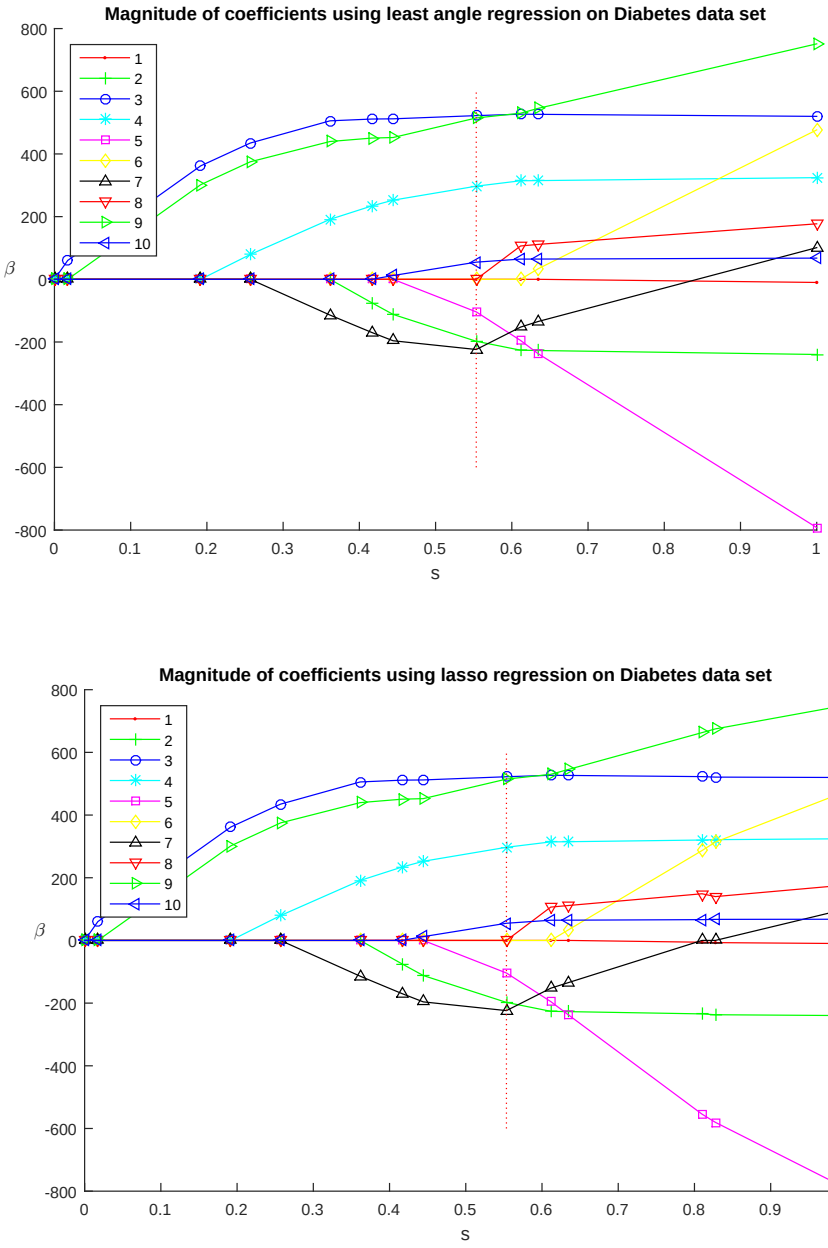


Figure 7: Coefficient paths for different LAR and LASSO on the Diabetes data set. The red dotted line shows the model selected by AIC, which in this case is the same model for both. Note that variable 7 is the only one that changes sign. In the LASSO algorithm it is removed from the active set when this happens and then it re-enters in the next iteration.

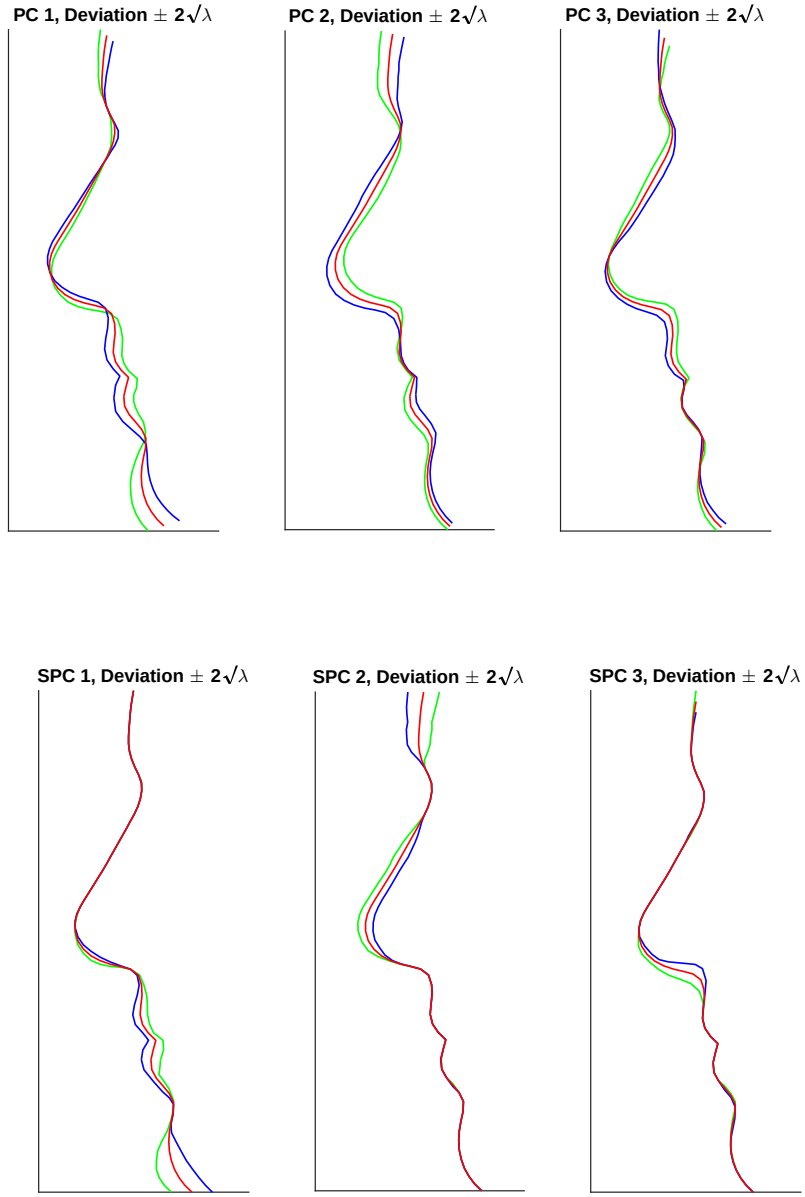


Figure 8: PCA and SPCA on the silhouette data set. The top row shows deviations in the directions of the first three components from PCA and the second row shows them for the components obtained by SPCA. The red curve represents the mean shape, the green one represents -2 standard deviations from the mean and the blue +2 standard deviations from the mean.

APPENDIX D

Sparse Interpretations of Online Reviews

The following manuscript is included as a technical report. The content will be submitted to the journal *Statistics and Probability Letters*.

Sparse Interpretations of Online Reviews

Gudmundur Einarsson
Technical University of Denmark

Rasmus R. Paulsen
Technical University of Denmark

Brendan P. Ames
University of Alabama

Line K. H. Clemmensen
Technical University of Denmark

Abstract

We present a novel approach for doing classification of data with ordinal labels in the $p \gg n$ setting. We construct the classifier based on sparse discriminant analysis using the data replication method, where we approach the underlying optimization problem with proximal gradient and alternating direction method of multipliers. We demonstrate performance on a challenging user review dataset and compare to a state of the art nominal classifier. Finally we demonstrate how the classifier allows us to interpret the data in a compact and meaningful way.

1 INTRODUCTION

There is a noticible lack of classification methods that can handle data with ordinal labels in the $p \gg n$ case, where p is the number of features and n is the number of samples. In this paper, we demonstrate how we extend sparse discriminant analysis (SDA) by Clemmensen et al. [2011] using the data replication method by Cardoso and Costa [2007]. To solve the optimal scoring problem formulated in Clemmensen et al. [2011] we rely on the alternating directions method of multipliers (ADMM) and the accelerated proximal gradient (APG) algorithms proposed by Atkins et al. [2017].

Supervised machine learning is traditionally split coarsely into regression and classification tasks, where the values/labels that we want to predict from our data are respectively continuous and discrete. In between, you find data with ordinal labels instead of nominal ones. Ordinal labels have the ordinal property in common with regression problems, and are discrete, similar to classification problems. These problems do not fall naturally in either of the two categories, and can be approached from both directions, more commonly from a regression point of view. Ananth and Kleinbaum [1997] give a good overview of commonly employed regression based approaches, e.g. the full continuation ratio model and a continuation ratio model with proportional odds structure. In these approaches, the ordinal property of the labels is commonly ignored and the labels are treated as nominal ones. The exploitation of the ordinal property should give rise to simpler and potentially more interpretable classifiers. The main problem is to incorporate the ordinal property into nominal classification algorithms. An alternative approach is to transform an ordinal classification problem into multiple binary classification problems via the data replication method introduced by Cardoso and Costa [2007]. These binary classification problems are solved together to find a common hyperplane that separates each pair of classes corresponding to adjacent ordinal labels. The only difference between the hyperplanes corresponding to different classification boundaries are biases, which are also provided in the solution.

In the past years, multiple methods have appeared which can handle feature selection and classification problems of the type $p \gg n$, most notably Sparse Discriminant Analysis (SDA) by Clemmensen et al. [2011] and Sparse Partial Least Squares for Classification by Chung and Keles [2010]. Other algorithms commonly used to solve such problems, where the focus is not neccesarily classification, are elastic net by Zou and Hastie [2005] and sparse principal component analysis by d'Aspremont et al. [2005]. Using an l_1-norm regularizer in the model formulation ensures that variable selection is performed in the model optimization process which gives leverage for the user to interpret

Preliminary work.

the non-zero parameters in the model. Incorporation of an l_1-norm regularizer is influenced by the Lasso [Tibshirani, 1996, Chen et al., 2001], which uses the l_1-norm to relax the vector cardinality function in the best feature subset problem for linear regression.

Ordinal labels appear in a multitude of applications, e.g., surveys, medical rating scales and most notably in relation to online user reviews. Users write reviews for products they have purchased and give the product a star rating, or simply a value from one to five. The number of stars can be interpreted as ordinal labels. Active research on user reviews in relation to social networks, online behaviour and recommender systems can e.g. be found in work by Leskovec and Krevl [2014], McAuley and Leskovec [2013] and Cheng et al. [2015]. Online products commonly have many user reviews and, e.g., in the case of tourism consumers rely heavily on these reviews for making decisions for future travel planning and selection of hotels and restaurants during travel [Gretzel and Yoo, 2008]. It is important for product owners to monitor the state of the reviews, but that can be tedious when they need to go through many reviews or if the same owner is monitoring multiple products.

In this paper, we employ ordinal SDA on user review data as a case study for building an ordinal classifier. We demonstrate how we can interpret the results from the classifier in a meaningful way and use it to discover trends that might promote the product owner to take actions, or to aid future buyers to make decisions based on the reviews for a given product. We use review data from the SNAP datasets by Leskovec and Krevl [2014]. We additionally demonstrate performance on the wine dataset from Cortez et al. [2009].

1.1 Contributions

The main contributions of this paper consists of integrating the data replication method by Cardoso and Costa [2007] with SDA. The additional parameters introduced by the data replication methods are left out from the regularization part of SDA, thus making the additional parameters independent of the effect of regularizers. The algorithm has been implemented and is available in the R-package (left intentionally blank for anonymity)[1] [R Core Team, 2017]. The package includes code to train a classifier and make predictions on unseen data.

We demonstrate how this method can be used in a novel way with a standard natural language processing (NLP) pipeline to create simple visualizations of online user reviews.

[1]Address to package, left blank for anonymity

2 METHOD

We begin by describing SDA and then the particularities of integrating the data replication method to obtain ordinal SDA. The notation used throughout the paper is summarized in Table 1. Finally, we summarize the NLP pipeline we use to preprocess the user review data.

Table 1: Notation

SYMBOL	DESCRIPTION
$\boldsymbol{\beta}_k$	Discriminant vector k
$\boldsymbol{\theta}_k$	Scoring vector k
$\mathbf{Y}$	Label indicator matrix
$\mathbf{X}$	Feature matrix
K	Number of classes
n	Number of samples
p	Number of variables
$\boldsymbol{\Omega}$	Tikhonov regularization matrix
$\|\| \cdot \|\|_1$	1-norm
λ_i	Regularization parameters
n_k	Number of samples in class k
$\boldsymbol{\beta}_{\text{Ord}}$	Ordinal discriminant vector
$\hat{\boldsymbol{\Omega}}$	Ordinal regularization matrix
$\mathbf{e}_i$	$1 \times (K-1)$ i-th unit vector
$\mathbf{E}_{k,i}$	n_k rows of $\mathbf{e}_i$
b_i	Bias i, where $i \in \{1, ..., K-1\}$
s	Width of binary classif. problem
$\mathbf{X}_{\text{Ord}}$	Replicated data matrix
$\mathbf{Y}_{\text{Ord}}$	Replicated label vector
$\mathbf{X}_{\text{Ord},i}$	Data matrix for boundary i
$\mathbf{Y}_{\text{Ord},i}$	Labels for boundary i
$\mathbf{x}^{(k)}$	$n_k \times p$ class k data matrix
$\mathbf{1}_k$	$n_k \times 1$ vector with only ones
$\mathbf{2}_k$	$n_k \times 1$ vector with only twos

2.1 Sparse Discriminant Analysis

Clemmensen et al. [2011] presented the sparse optimal scoring problem (SOS), which is the formulation we employ to solve ordinal SDA. SDA works in some sense like a supervised version of Principal Component Analysis (PCA), where we seek to find discriminant vectors to project the data to a lower dimensional representation, where we balance the objectives of minimizing variation within classes, maximizing variation between classes and feature selection. New samples are then traditionally classified according to the nearest centroid after projection:

$$\underset{\boldsymbol{\theta}_k \in \mathbf{R}^K,\, \boldsymbol{\beta}_k \in \mathbf{R}^p}{\arg\min} \underbrace{\underbrace{\|\boldsymbol{Y}\boldsymbol{\theta}_k - \boldsymbol{X}\boldsymbol{\beta}_k\|_2^2}_{\text{Optimal Scoring}} + \lambda_2 \boldsymbol{\beta}_k^T \boldsymbol{\Omega} \boldsymbol{\beta}_k + \lambda_1 \|\boldsymbol{\beta}_k\|_1}_{\text{Sparse Optimal Scoring}}$$

$$\begin{aligned} \text{s.t. } \tfrac{1}{n}\boldsymbol{\theta}_k^T \mathbf{Y}^T \mathbf{Y} \boldsymbol{\theta}_k &= 1, \\ \boldsymbol{\theta}_k^T \mathbf{Y}^T \mathbf{Y} \boldsymbol{\theta}_\ell &= 0 \; \forall \ell < k, \end{aligned} \tag{1}$$

In Equation 1, the constraints apply to both the optimal scoring and the sparse optimal scoring formulation. The constraints are spherical which makes the problem non-convex, but for a given value of $\boldsymbol{\theta}_k$, solving for $\boldsymbol{\beta}_k$ is a convex problem. When we solve this problem we seek the discriminant vectors $\boldsymbol{\beta}_i$, which we can then use to project the data from feature space to a lower-dimensional representation. The traditional approach is to solve this minimization problem with a block-update algorithm, where one first solves for $\boldsymbol{\theta}$, then $\boldsymbol{\beta}$ and iterate until convergance. That way we can find the first $(\boldsymbol{\theta}_1, \boldsymbol{\beta}_1)$ pair, then we continue in a similar manner to find the successive pairs until we have found the maximum number of pairs, $K-1$, or the desired number of pairs. In the ordinal case, we cast our problem as a binary classification problem, which only yields a single discriminant vector $\boldsymbol{\beta}_{\text{Ord}}$, simplifying the interpretation of the solution. $\boldsymbol{\beta}_{\text{Ord}}$ is a vector of length $p+K-1$, so the first p parameters correspond to the original variables, that we can interpret. The extra $K-1$ parameters are the additional biases introduced by the data replication method.

Clemmensen et al. [2011] show that for a given $\boldsymbol{\beta}$ one can find $\boldsymbol{\theta}$ in polynomial time. For a given $\boldsymbol{\theta}$ the problem formulation is an elastic net problem and can be solved with the LARS-EN algorithm by Zou and Hastie [2005]. We however approach the optimization from the point of proximal gradient methods and alternating direction method of multipliers, using the soft thresholding operator to deal with the sparse regularizer in the same manner as Atkins et al. [2017].

2.2 Using Ordinal Labels via Data Replication

A natural assumption for an ordinal classifier of K classes, is to have $K-1$ non-intersecting classification boundaries, where boundary i separates classes 1 to i from classes $i+1$ to K. In our case, that means finding a hyperplane and a set of biases. We employ the data replication method of [Cardoso and Costa, 2007]. We demonstrate how to build the new data matrix, with the replicated samples.

For each of the $K-1$ classification boundaries we define a binary classification problem. If the boundary is the one between classes i and $i+1$, then the binary labels correspond to labels i and lower, and $i+1$ and higher. The maximum number of classes we use on each side of the boundary is called s, which is an integer specified by the user. So for the boundary between classes i and $i+1$ we label classes $i-s+1, i-s+2, ..., i$ as belonging to class 1 and the classes $i+1, i+2, ..., i+s$ labeled as belonging to class 2. Then we append the vector $\mathbf{e}_i$, which is of length $K-1$, to the samples, where $\mathbf{e}_i$ is a vector of all zeroes, excepts it takes the value 1 in position i. All these new samples are added to the new data matrix. Class k has n_k samples and we define $\mathbf{x}^k$ as the $n_k \times p$ data matrix, only containing samples from class k. We also define the $n_k \times (K-1)$ matrix $\mathbf{E}_{k,i}$, which is a matrix of all zeroes, except the i-th column is a vector of all ones.

$$\mathbf{X}_{\text{Ord},i} := \begin{bmatrix} \mathbf{x}^{(i-s+1)} & \mathbf{E}_{(i-s+1),i} \\ \vdots & \vdots \\ \mathbf{x}^{(i)} & \mathbf{E}_{i,i} \\ \mathbf{x}^{(i+1)} & \mathbf{E}_{i+1,i} \\ \vdots & \vdots \\ \mathbf{x}^{(i+s)} & \mathbf{E}_{i+s,i} \end{bmatrix} \tag{2}$$

$$\mathbf{Y}_{\text{Ord},i} := \begin{bmatrix} \mathbf{1}_{1-s+1} \\ \vdots \\ \mathbf{1}_i \\ \mathbf{2}_{i+1} \\ \vdots \\ \mathbf{2}_{i+s} \end{bmatrix} \tag{3}$$

The data matrix $\mathbf{X}_{\text{Ord},i}$, corresponding only to the data needed for the boundary between class i and $i+1$ can be seen in Equation 2 and the corresponding label vector in Equation 3.

$$\mathbf{X}_{\text{Ord}} := \begin{bmatrix} \mathbf{X}_{\text{Ord},1} \\ \mathbf{X}_{\text{Ord},2} \\ \vdots \\ \mathbf{X}_{\text{Ord},K-1} \end{bmatrix} \mathbf{Y}_{\text{Ord}} := \begin{bmatrix} \mathbf{Y}_{\text{Ord},1} \\ \mathbf{Y}_{\text{Ord},2} \\ \vdots \\ \mathbf{Y}_{\text{Ord},K-1} \end{bmatrix} \tag{4}$$

The final data matrix is constructed from the matrices corresponding to the binary classification problems as shown in Equation 4.

Now we would wish to plug the matrix $\mathbf{X}_{\text{Ord}}$ and the labels $\mathbf{Y}_{\text{Ord}}$ into the SOS problem in Equation 1. We call the new discriminant vector that comes from this problem $\boldsymbol{\beta}_{\text{Ord}}$:

$$\boldsymbol{\beta}_{\text{Ord}} := \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \\ b_1 \\ b_2 \\ \vdots \\ b_{K-1} \end{bmatrix}, \tag{5}$$

which is composed of a traditional discriminant vector, corresponding to the first p elements, and then $K-1$

biases, denoted b_i, for $i \in \{1, 2, ..., K-1\}$. We want to regularize the first p parameters like in nominal SDA, but the parameters corresponding to the biases, should not be regularized. In Equation 1 we use instead a $(p + K - 1) \times (p + K - 1)$ regularization matrix $\hat{\mathbf{\Omega}}$, where the top left-most block corresponds to the $p \times p$ $\mathbf{\Omega}$ regularization matrix for the original variables and the rest is zero, such that there is no regularization for the extra parameters corresponding to the biases:

$$\hat{\mathbf{\Omega}} := \begin{bmatrix} \mathbf{\Omega} & 0 \\ 0 & 0 \end{bmatrix} \tag{6}$$

In a similar manner for the l_1-norm term, we only calculate the l_1-norm of the first p parameters in $\boldsymbol{\beta}_{\text{Ord}}$, not regularizing the parameters corresponding to biases. These details are masked from the user in the implementation, thus the user can optionally specify a $p \times p$ $\mathbf{\Omega}$ regularization matrix. This makes the usage almost identical to that of nominal SDA. The main difference from the usage of nominal SDA is that the user can not specify the number of dicriminant vectors in the output, ordinal SDA only generates one.

After doing the data transformation, and redefining the $\mathbf{\Omega}$ regularization matrix, the only change we need to do in the algorithms proposed by Atkins et al. [2017], is to only do the soft-thresholding update for the first p parameters in $\boldsymbol{\beta}_{\text{Ord}}$. This amounts to only using the first p parameters of $\boldsymbol{\beta}_{\text{Ord}}$ to calculate the l_1-norm in Equation 1.

$$\begin{aligned} \underset{\boldsymbol{\theta} \in \mathbf{R}^2, \boldsymbol{\beta}_{\text{Ord}} \in \mathbf{R}^{p+K-1}}{\arg\min} \quad & \|\boldsymbol{Y}_{\text{Ord}}\boldsymbol{\theta} - \boldsymbol{X}_{\text{Ord}}\boldsymbol{\beta}_{\text{Ord}}\|_2^2 \\ & + \lambda_2 \boldsymbol{\beta}_{\text{Ord}}^T \hat{\mathbf{\Omega}} \boldsymbol{\beta}_{\text{Ord}} + \lambda_1 \textstyle\sum_{i=1}^{p} |\beta_i| \\ \text{s.t. } & \frac{1}{n} \boldsymbol{\theta}^T \boldsymbol{Y}_{\text{Ord}}^T \boldsymbol{Y}_{\text{Ord}} \boldsymbol{\theta} = 1. \end{aligned} \tag{7}$$

The original SOS problem 1 is reformulated w.r.t. the new notation and change in problem in Equation 7. Note that we no longer need the orthogonality constraint, since we only have one discriminant vector.

2.3 Predictions

The biases in the ordinal discriminant vector are ordered, so they define intervals on the real line. For doing predictions on an unseen sample x, assuming it is a $1 \times p$ row vector, we first normalize it appropriately, and call the normalized sample x_{n}. Afterwards we calculate the scalar value $\tilde{x} = x_{\text{n}} \cdot \hat{\boldsymbol{\beta}}_p$, where $\hat{\boldsymbol{\beta}}_p$ is the $p \times 1$ vector corresponding to the first p values from the ordinal discriminant vector. Then we find the lowest bias b_i, such that $\tilde{x} < b_i$, that means that we predict that the new observation belongs in class i.

2.4 Normalization and Preprocessing of NLP data

There is a multitude of Amazon products available in the SNAP dataset [Leskovec and Krevl, 2014]. To narrow our focus we selected the *Apps for Android* category to work with using the 5-core dense subset. This means that every product has been reviewed at least 5 times, by reviewers that have made at least 5 reviews. If we consider the reviewers and products as nodes in an undirected graph, and reviews as edges connecting users and reviews, the k-core is the maximal connected subgraph where all vertices have degree at least k. From this category we consider the 10 products with the highest number of reviews for further analysis. For each product we train an ordinal SDA model where the number of reviews is balanced, such that each class contains 100 samples. The distribution of rating scores can be seen in Figure 2.4 for the most rated app. This distribution is very typical, where there are very few 2-star reviews. We alleviate this potential issue by sampling indicies with replacement for the training set. We sample 100 samples per class with replacement for the training set, giving a total of 500 samples for each case. For the test set we remove the training indicies from the set and sample 50 samples with replacement for the each class, so there are no samples that appear both in the training and test set.

For each of the ten datasets, corresponding to the ten highest rated products, we prepare the data with the following steps, using both the test and train data:

(1) We use the `tm` package by Feinerer and Hornik [2017] in `R` to do the following transformations [Meyer et al., 2008]:
   (a) Make all the letters lower case.
   (b) Remove punctuation.
   (c) Create a plain text document.
   (d) Remove English stopwords.

(2) We use the stemmer from the `SnowballC` package for the English language [Bouchet-Valat, 2014].

(3) We generate 1 to 5 grams and prune the vocabulary such that a term needs to appear in at least 3 documents using the `text2vec` package [Selivanov and Wang, 2017].

The process depicted above yields a high number of n-grams for each product, summarized in Table 2. The datasets all have 500 samples, 100 per class and the number of variables ranges from 2118 to 4505 variables. At this point we have a document term matrix

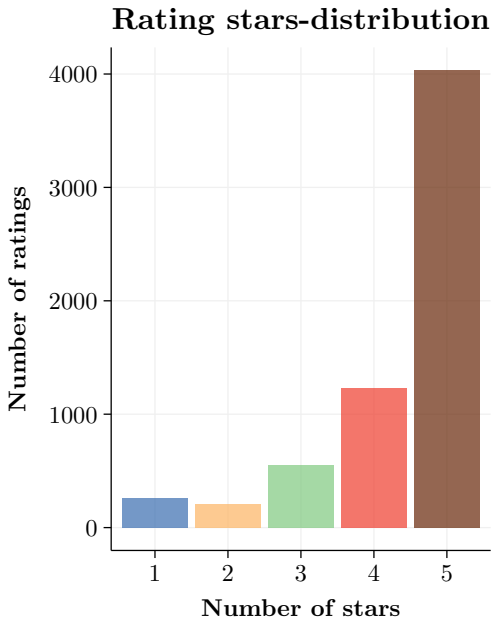


Figure 1: Distribution of Stars from the 5-Core SNAP Dataset on the Most Rated Android App. There is a Noticeable Lack of 2-Star Reviews in Most of the Products, and a Very High Bias for 5-Stars.

(DTM) with number of variables approximately an order of magnitude higher than the number of samples.

Here we separate the test and train data again. We scale each column in the training data by the value of the 75% quantile. We use the same values to scale the test data. We **do not** center the data, the only normalization is scaling to normalize the contributions from different variables. This makes it possible to interpret the discriminant vector we find with ordinal SDA. The non-zero parameters with a certain sign correspond to terms that increase the score, while the opposite sign corresponds to a negative score. The magnitude of the absolute value of the parameter corresponds to how strongly the value contributes to the prediction. We can visualize the solutions as the wordclouds seen in Figures 4 and 5 using the `wordcloud` package [Fellows, 2014].

3 EXPERIMENTS

3.1 Amazon Rating/Review Data

For the product review and rating data we perform the preprocessing and normalization steps from the last section. We do 10-fold cross-validation on the training data to find the best best regularization parameters for the elastic net penalty in Equation 1. Our $\mathbf{\Omega}$ regularization matrix is a diagonal $p \times p$ matrix. We allow the γ_1 and γ_2 parameters to be from the following sets:

$$\lambda_1 \in \{1, 2, 3, 4, 5, 6\}, \qquad \lambda_2 \in \{0.01, 0.1, 1, 2\} \tag{8}$$

After the cross-validation we select the parameter pair that has the highest average accuracy.

We also use sparse partial least squares classification (`splsda`) by Chun and Keleş [2010] from the `R`-package `splss` [Chung et al., 2013] for comparison. We use the same preprocessing and normalization and do 10-fold cross-validation, where the parameters η and K come from the sets:

$$\eta \in \{0.01, 0.03, ..., 0.09\}, \qquad K \in \{1, 2, ..., 5\}. \tag{9}$$

This is a nominal classification method, so we do not expect to get the same type of interpretation from the solution, and more parameters are needed, in case K is greater than 1.

3.2 Wine Data

For the Wine data from Cortez et al. [2009], we have two datasets of wines, namely red wine with 1599 observations and white wine with 4898 observations, both contain the same eleven measured features, we refer the reader to the original work for a further description of the features. We approach the problem similar to Alquier and Biau [2013]. We randomly sample half of the indicies from each class to use as a training set, do 10-fold cross-validation on the training set and then train the classifier on the whole training set with the best found parameters, and finally we calculate the accuracy and accuracy off-by-one on the other half. We repeat this process 20 times and report the average accuracy and off-by-one accuracy.

Here we use the Accelerated Proximal Gradient algorithm. We allow the regularization parameters to be from the set:

$$\lambda_1, \lambda_2 \in \{0.0001, 0.001, 0.01, 0.1\} \tag{10}$$

We use a diagonal matrix as $\mathbf{\Omega}$. We also report the average number of parameters used in the final model.

4 RESULTS

4.1 Review Data

For the proposed method the cross-validation parameters were 0.01 in all cases for the ridge penalty, and 1.0 for the l_1-norm penalty, except it was 2.0 for datasets one and nine. The accuarcy on the test data and the sparsity of the classifiers is summarized in Figure 4.1. The accuracy ranges from 0.268 to 0.384. We also calculate the off-by-one accuracy, where we allow the prediction to be one less or greater than the true label. The off-by-one accuracy ranges from 0.708 to 0.84.

The η parameter in the SPLS cross-validation took all different values and the K parameter was selected as three for dataset-one, four for dataset-five and five in the rest of the cases. The results are summarize in Figure 4.1. The minimum accuracy on a test set was 0.224 and the maximum was 0.388. The off-by-one accuracy is from 0.652 to 0.796.

The proposed method has a higher accuracy in four of the ten test datasets. The proposed method has a higher off-by-one accuracy in eight out of the ten test datasets. The proposed solution also provides a single interpretable discriminant vector, which was in all cases more sparse then the ones provided by SPLS. Some of the words that have the strongest effect are visualized in Figures 4 and 5. It can be clearly seen that words belonging to the negative wordcloud have a negative overall flavour. E.g. indicating software bugs or updates. The words in the positive wordcloud indicate a more positive spirit.

Table 2: Number of Variables for Each of the Product Datasets and the Number of Variables Used in The Final Model.

PRODUCT	P	P NON-ZERO
1	2118	641
2	2609	1356
3	3611	1233
4	4505	1283
5	2670	1054
6	2379	902
7	2885	966
8	3697	1361
9	3074	549
10	2927	849

4.2 Wine Data

The results on the red and white wine datasets are reported in table 3.

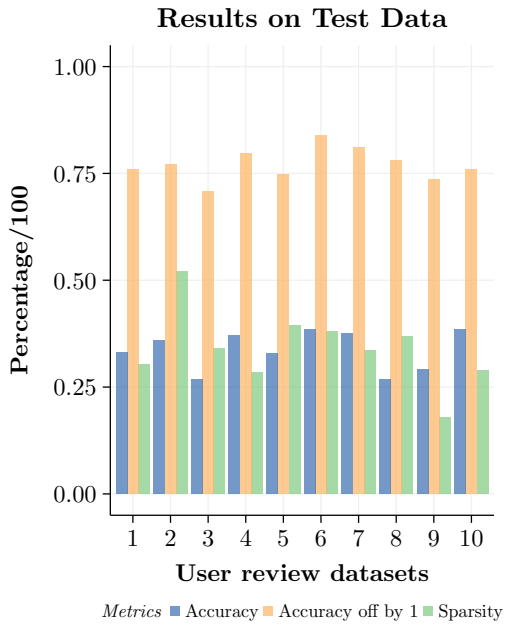


Figure 2: Results from the Proposed Method. Accuracy, Accuracy Where We Allow the Classification to be off by One, and the Proportion of Non-Zero Parameters in the Used Model.

For the red wine the λ_2 regularization parameter was selected 17 out of 20 runs as 0.0001, λ_1 was selected as 0.001 16 out of 20 runs. There were on average eight to nine non-zero parameters in the final classifier.

For the white wine dataset, λ_1 and λ_2 were selected as 0.0001 in all cases. The final classifier contained ten non-zero parameters in all cases.

In both the red and white wine datasets, the off-by-one accuracy is more than two times higher than the accuracy.

4.3 Discussion

The method we present correctly identifies terms associated with positive or negative scores and it does so in a setting, where the number of terms is an order of magnitude higher than the number of samples. Presenting the results as wordclouds gives the human observer a way to very quickly get an overview of what

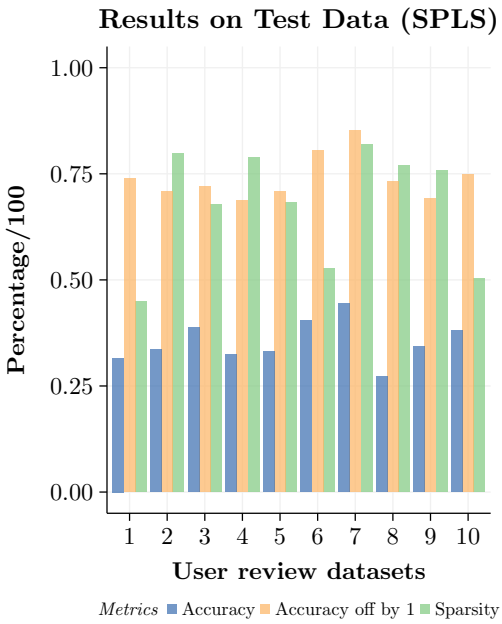


Figure 3: Results from SPLS Classification Method. Accuracy, Accuracy Where We Allow the Classification to be off by One, and the Proportion of Non-Zero Parameters in the Used Model.

terms mostly associate with positive or negative reviews.

The wordcloud presentation can be greatly expanded upon in an interactive online setting. The results could also be visualized as graphs, where thickness of edges could represent how often the terms appear together. An additional feature would also be to allow the user to navigate to the reviews that include the selected feature, ranked by how often the term appears in the review, or ranked by rating.

There are definitely parts of the NLP pipeline that could be improved. For the wine dataset we do not challenge the method in terms of having more features than samples. Another option for the wine data would be to include all the interaction and second order terms.

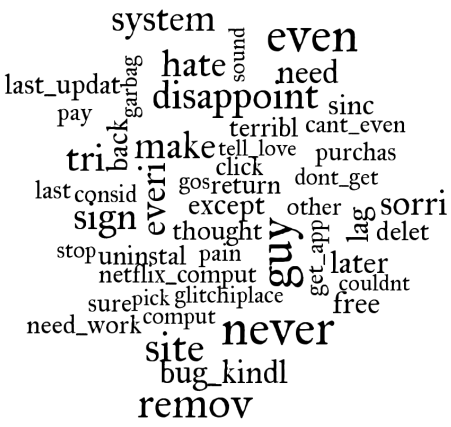


Figure 4: Negative Words Associated With an App in One of the Review Datasets. Larger Words Represent a Stronger Effect.

5 CONCLUSION

We present a novel way to approach ordinal classification in the $p \gg n$ setting. Our method is implemented and available online in the form of an `R`-package. We test the method on a dataset of online reviews, where we achieve accuracy similar to that of a state of the art nominal approach. Our off-by-one accuracy is better in eight out of ten cases, showing that our wrong predictions are more likely to be predicted as a class *closer* to the correct class.

We also demonstrate how the results from the classifier can be presented in such a way that a layman can easily understand and interpret the results. This form of presentation gives possibilities for customers to get a quick overview over user review data, and for product owners to monitor the feedback from users.

Acknowledgements

Gudmundur's PhD is funded jointly by the Lundbeck Foundation and the Technical Univeristy of Denmark.

Figure 5: Positive Words Associated With an App in One of the Review Datasets. Larger Words Represent a Stronger Effect.

Table 3: Wine Data Results.

MEASURE	RED	WHITE
Accuracy Mean	0.433	0.430
Accuracy SD	0.036	0.007
Accuracy off-by-1 Mean	0.929	0.886
Accuracy off-by-1 SD	0.014	0.005
Sparsity Mean	0.786	0.909
Sparsity SD	0.111	0

References

P. Alquier and G. Biau. Sparse single-index model. *Journal of Machine Learning Research*, 14(Jan): 243–280, 2013.

C. V. Ananth and D. G. Kleinbaum. Regression models for ordinal responses: a review of methods and applications. *International journal of epidemiology*, 26(6):1323–1333, 1997.

S. Atkins, G. Einarsson, B. Ames, and L. Clemmensen. Proximal methods for sparse optimal scoring and discriminant analysis. *arXiv preprint arXiv:1705.07194*, 2017.

M. Bouchet-Valat. *SnowballC: Snowball stemmers based on the C libstemmer UTF-8 library*, 2014. URL `https://CRAN.R-project.org/package=SnowballC`. R package version 0.5.1.

J. S. Cardoso and J. F. Costa. Learning to classify ordinal data: The data replication method. *Journal of Machine Learning Research*, 8(Jul):1393–1429, 2007.

S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM review*, 43(1): 129–159, 2001.

J. Cheng, C. Danescu-Niculescu-Mizil, and J. Leskovec. Antisocial behavior in online discussion communities. In *ICWSM*, pages 61–70, 2015.

H. Chun and S. Keleş. Sparse partial least squares regression for simultaneous dimension reduction and variable selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(1): 3–25, 2010.

D. Chung and S. Keles. Sparse partial least squares classification for high dimensional data. *Statistical applications in genetics and molecular biology*, 9(1), 2010.

D. Chung, H. Chun, and S. Keles. *spls: Sparse Partial Least Squares (SPLS) Regression and Classification*, 2013. URL `https://CRAN.R-project.org/package=spls`. R package version 2.2-1.

L. Clemmensen, T. Hastie, D. Witten, and B. Ersbøll. Sparse discriminant analysis. *Technometrics*, 53(4): 406–413, 2011.

P. Cortez, A. Cerdeira, F. Almeida, T. Matos, and J. Reis. Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47(4):547–553, 2009.

A. d'Aspremont, L. E. Ghaoui, M. I. Jordan, and G. R. Lanckriet. A direct formulation for sparse pca using semidefinite programming. In *Advances in neural information processing systems*, pages 41–48, 2005.

I. Feinerer and K. Hornik. tm: text mining package. r package version 0.7-1. `https://CRAN.R-project.org/package=tm`, 2017.

I. Fellows. *wordcloud: Word Clouds*, 2014. URL `https://CRAN.R-project.org/package=wordcloud`. R package version 2.5.

U. Gretzel and K. H. Yoo. Use and impact of online travel reviews. *Information and communication technologies in tourism 2008*, pages 35–46, 2008.

J. Leskovec and A. Krevl. SNAP Datasets: Stanford large network dataset collection. `http://snap.stanford.edu/data`, June 2014.

J. McAuley and J. Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 165–172. ACM, 2013.

D. Meyer, K. Hornik, and I. Feinerer. Text mining infrastructure in r. *Journal of statistical software*, 25(5):1–54, 2008.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2017. URL `https://www.R-project.org/`.

D. Selivanov and Q. Wang. *text2vec: Modern Text Mining Framework for R*, 2017. URL `https://CRAN.R-project.org/package=text2vec`. R package version 0.5.0.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.

APPENDIX E

Foreign Object Detection in Multispectral X-ray Images of Food Items Using SDA

The following manuscript was presented at the Scandinavian Conference on Image Analysis in Tromsó June 2017.

Foreign Object Detection in Multispectral X-ray Images of Food Items using Sparse Discriminant Analysis

Gudmundur Einarsson[1], Janus N. Jensen[1], Rasmus R. Paulsen[1], Hildur Einarsdottir[1], Bjarne K. Ersbøll[1], Anders B. Dahl[1], and Lars Bager Christensen[2]

[1] DTU Compute, Technical University of Denmark,
Richard Petersens Plads, Building 324, DK-2800 Kgs. Lyngby
{guei,jnje,rapa,hildr,bker,abda}@dtu.dk
http://www.compute.dtu.dk/english
[2] Teknologisk Institut,
Gregersensvej 9, 2630 Taastrup
lbc@teknologisk.dk
http://www.teknologisk.dk

Abstract. Non-invasive food inspection and quality assurance are becoming viable techniques in food production due to the introduction of fast and accessible multispectral X-ray scanners. However, the novel devices produce massive amount of data and there is a need for fast and accurate algorithms for processing it. We apply a sparse classifier for foreign object detection and segmentation in multispectral X-ray. Using sparse methods makes it possible to potentially use fewer variables than traditional methods and thereby reduce acquisition time, data volume and classification speed. We report our results on two datasets with foreign objects, one set with spring rolls and one with minced meat. Our results indicate that it is possible to limit the amount of data stored to 50% of the original size without affecting classification accuracy of materials used for training. The method has attractive computational properties, which allows for fast classification of items in new images.

Keywords: X-Ray, Multispectral, Sparse Classification, Foreign Object Detection

1 Introduction

One of the many purposes of X-ray scanning is to provide quality control and assurance in food production industry. The usage of X-rays provides non-destructive means of examining food items and the data can be used to verify that the content is free of anomalies or foreign objects. The usage of multispectral X-ray scanning has been used successfully in detecting explosives [12], and compares well to an X-ray dual-energy sandwich detector [8]. Foreign objects found in food items consist mostly of insects, wood chips, stone pebbles, sand/dust and

plastic. These objects might be present in the raw materials, or accidentally introduced during the manufacturing process [6], where organic materials pose the main challenge for detection. Grating-based imaging techniques [10, 9], that measure the attenuation, scattering and refraction of X-ray beams, have shown great promise in detecting organic foreign objects [6]. Although grating based methods are promising, they still have not been scaled to be used in a production line. Multispectral X-ray scanners exist with a conveyor belt setup, where a single acquisition takes around 5 seconds on the setup we used. Certain foreign objects can be detected in multispectral X-ray images of food items using a sparse classifier, which gives the potential for storing fewer data and making classification and acquisition faster.

According to the Beer-Lambert Law (BLL), the intensity of an X-ray beam decreases as it passes through matter with exponential decay depending on the distance traveled and the medium.

$$I = I_0 e^{-\mu\rho d} \tag{1}$$

I_0 in Equation 1 corresponds to the initial X-ray intensity, μ and ρ together form the linear absorption coefficient, where μ corresponds to mass absorption and ρ corresponds to density, and finally d corresponds to the distance traveled by the beam. For a given simple material, this equation allows us to do inference on either the thickness or type of material we have, in case either of them is known. In our case, we are interested in food items. Food items are particularly challenging since their shape and material composition can greatly vary. There is considerate specimen to specimen variation along with potentially inhomogeneous materials which makes inference about the material composition difficult.

Instead of trying to model the signal according to the BLL, we apply a data-driven approach, where we train a classifier to recognize whether foreign objects are present by training it on multispectral X-ray image samples of a given food product and foreign objects in the food. The dimensionality of the data is high and poses problems for data storage and processing speed. We thus seek a sparse classifier in order to examine whether decreasing the data dimensionality will still result in reasonable accuracy and a low false discovery rate. We choose sparse linear discriminant analysis (SDA) [4] for this task, since it perfectly fits the requirements. This classification method performs variable selection and dimensonality reduction in the optimization process, which allows us to identify which spectra in the images are relvant for the given classification task. This is achieved via an elastic net regularizer [14]. The elastic net regularizer also allows for the construction of a Tikhonov regularization matrix, that can be further tailored to the specific classification task. Knowing which spectra are relevant for the classification task, allows for compressing the data, to only store the relevant spectra, and it could also give some domain knowledge on which spectra are most different between certain materials. A similar method that could also be considered for this task is sparse partial least squares, [3].

To generate the data for the classifier, we first preprocess it. The main part is normalizing each pixel w.r.t. the maximum intensity, which gives better constrast

between different materials. These steps are further explained in the next section. Note that maximum 6 scans (images) were used for generating labels for training, a process that takes around 10 seconds per image. So the process of generating data for training and training the classifier is fast. This process can further be automated for a given target application.

We will examine to what extent we can detect foreign objects in two types of food materials and report which objects we detect.

The paper's outline consists of a description of the data and acquisition process, where we explain the scanner setup, the properties of the data we obtain and the preprocessing. Next, we describe how the data sets are prepared for the classifier and a description of the classifier. Finally, we present results to evaluate the performance and some discussion.

2 Data and Acquisition

The scanner used for data acquisition is a MULTIX multispectral X-ray scanner with three daisy chained detection modules, providing line scans of $3 \times 128 = 384$ pixels, where the pixel side length is 800μm. The energy of the photons is measured over 128 energy bins, (also referred to as channels), where the energy for our experiments is set to 90 kV. This spectrometric scanner is made from a combination of a semiconductor crystal and advanced electronics, capable of measuring the energy of every incident X-ray photon. The material signature is acquired in real-time and stored in raw format [2][12]. This scanning technology has been compared to a dual energy sandwich detector, (where two detector modules are used with a single shot exposure), showing better detection for explosive materials and a lower false discovery rate (FDR) [8]. We aim to examine which part of the spectra is best suited for foreign object detection in food and to what extent can we detect foreign objects in organic food items using a sparse classifier.

The MULTIX scanner provides images in a binary format, where pixel intensities are encoded as 16-bit unsigned integers. To make sure that there are no scaling differences between samples, we scale the intensities by looking at an average measurement for a patch of air in the image and find the peak value over all the channels. We scale all values by the inverse of the value at the peak, such that the maximum attenuation corresponds to one, this ensures that no scaling differences are between different images. The different attenuation profiles in an image of spring rolls can be seen in Fig 1. Foreign objects that are not "inside" a food item give a very different attenuation profile from the ones that are inside the food items. The attenuation profile of the food items naturally varies a lot by the thickness of the item, thus making it more difficult to work with products where the thickness varies much.

After this initial scaling, we remove line artifacts. These artifacts appear as strides in the image where two different detector modules are attached (See Fig. 3). To achieve this we start by creating an average air profile. We use a patch of pixels from a corner of the image where there are no overlapping

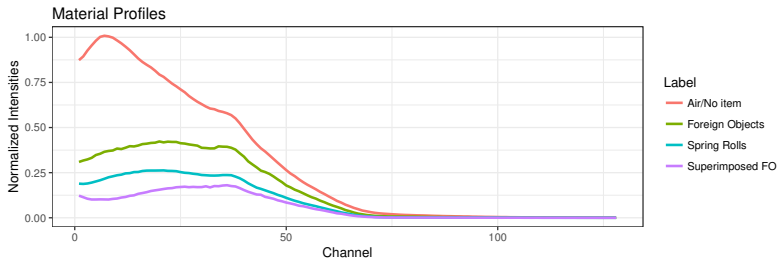


Fig. 1. Intensity profiles for different materials in an image of spring rolls with foreign objects. The green line above the blue one corresponds to foreign objects that are not superimposed on food items, while the bottom most purple line corresponds to foreign objects that are superimposed on top of food items. The data is scaled such that the peak for air/no item is at 1. The profiles are further normalized (not depicted here) such that the maximum value of every pixel is 1. This is a crude normalization for depth/thickness and gives us data that better represents differences between materials.

detector modules and we are certain that it contains no items. The mean over the samples gives us an average air profile, similar to the red one seen in Fig. 1, i.e. 128 values that should represent no items, we call that vector $\boldsymbol{\mu}_{\text{Air}} \in \mathbb{R}^{128\times 1}$. For each column in the scanning direction of the image, we now look at the mean of the first 50 pixels. This gives us another profile which is specific to that particular column, i.e. a local mean profile, which we call $\boldsymbol{\mu}_{\text{local}} \in \mathbb{R}^{128\times 1}$. Now we need to find the scaling difference between $\boldsymbol{\mu}_{\text{Air}}$ and $\boldsymbol{\mu}_{\text{local}}$, that is the vector $\mathbf{s} \in \mathbb{R}^{128\times 1}$ which is the solution to the following equation, where on the left-hand side we have elementwise multiplication.

$$\mathbf{s}\boldsymbol{\mu}_{\text{Air}} = \boldsymbol{\mu}_{\text{Air}} \tag{2}$$

The solution to Equation 2 is simply found via elementwise division of the mean vectors. Now for each pixel in this particular column, we multiply the 128 values with the scaling vector $\mathbf{s}$ elementwise. This process is depicted in Fig. 2, where the strides have been removed in the middle image.

Finally, we do a crude normalization for depth. For every pixel in the image, we find the maximum $c_{\max}$ of the 128 channels and multiply each of the 128 values by $1/c_{\max}$, such that all values in each pixel lie between 0 and 1 and the maximum is 1. This should give us data that better represents differences between materials, rather than thickness since different materials have their maximum intensities at different channels, (see Fig 1, where this scaling has not been done and the maximum intensity appears in different channels). All the preprocessing steps are depicted in Fig. 2.

We test our approach on two datasets, one with spring rolls and another with minced meat. The spring rolls dataset is challenging in a sense that the thickness varies considerably and the spring rolls can also overlap. The minced meat data varies much by thickness since it contains strings/filaments of meat that overlap.

The objects in Table 1 were used for the imaging, where they were superimposed on top of the food material.

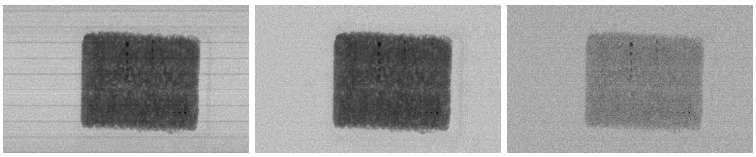


Fig. 2. An illustration of the preprocessing steps for the images. Left most image shows channel 10 in the raw data (minced meat). The image in the middle shows channel 10 in image after the removal of the strides/line artifacts. The rightmost image shows channel 10 when each pixel has been scaled by the maximum value in each of its channels. The final normalization step gives more contrast between different materials although the contrast between meat and no item is less in this particular channel. The intensities are scaled linearly from black (lowest) to white (highest) in the shown images.

2.1 Spring Rolls

The spring rolls data set consists of scans of 8 different bags of spring rolls. Each bag was scanned 20 times, then refrozen and scanned again a day later 20 times each. The foreign objects were superimposed on the bags. The foreign objects were also scanned individually 10 times and each bag was also scanned 2 times without any foreign objects. Fig. 3 shows four of the image channels in a grayscale. Most of the contrast between the different materials seems to be present in the first channels, which can also be seen in Fig. 1. The different scans provide variation in position and rotation of the food items and the different bags provide shape differences for the dataset. The spring rolls were contained in a plastic bag.

2.2 Minced Meat

A single plastic box containing 1kg of minced meat was used for all the scans. First, 5 scans were produced with no items, then the meat was scanned 5 times without any foreign objects. Finally, the meat was scanned with 3 sets of foreign objects, 10 times for each set. The types of foreign objects in each of the three sets is described in Table 1 and a sample image from the data can be seen in Fig. 3.

3 Method

For a given scanned food item, we would like to classify which parts of the image contain food and which contain foreign objects. To achieve this we first need to construct a dataset for training a classifier.

Table 1. Foreign objects used for the scans of minced meat. The items used for the spring rolls are the same excluding the last seven items. Set 3 consists of the same items as used in [6]. PTFE is an acronym for Polytetrafluoroethylene, more commonly referred to as Teflon.

Material	*Number of pieces*	*Size range*	*Dataset*
Quartz balls	5	1-4mm	Set 1 for minced meat
Aluminium balls	6	2-7mm	Set 1 for minced meat
Soft bone	5	5mm long	Set 1 for minced meat
Bone phantoms	5	2-6mm	Set 1 for minced meat
Polycarbonate balls	6	3.2-8mm	Set 2 for minced meat
Ceramic balls	6	2-8mm	Set 2 for minced meat
Glass balls	6	1-5mm	Set 2 for minced meat
PTFE balls	6	1.6-5mm	Set 2 for minced meat
Wood	4	2-8mm	Set 3 for minced meat
Stone pebbles	4	2-8mm	Set 3 for minced meat
Soft plastic	4	2-8mm	Set 3 for minced meat
Hard plastic	4	2-8mm	Set 3 for minced meat
Metal	4	2-8mm	Set 3 for minced meat
Rubber	4	2-8mm	Set 3 for minced meat
Glass	4	2-8mm	Set 3 for minced meat

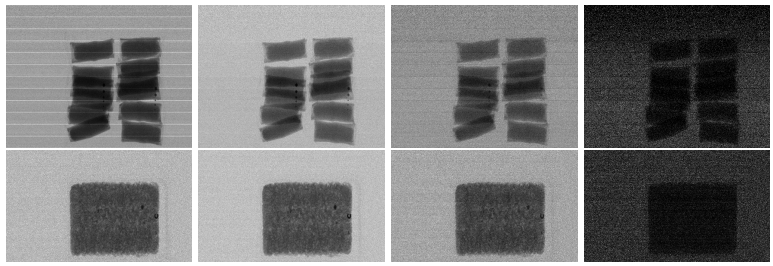


Fig. 3. Raw grayscale images of different channels from a spring roll sample (top row) and minced meat (bottom row) generated with the MULTIX scanner. From left to right are channels 2, 20, 50 and 100. The contrast decreases the higher we go in the channels and the variation in the measurements increases. The foreign objects can be seen as small black dots in the images and are most visible in the second image, better visible in Fig. 4. Strides have been removed in the meat data to show the difference compared to the strides that are present in the spring rolls images shown here. The intensities are scaled linearly from black (lowest) to white (highest). In this image, the line artifacts have not been removed and the individual pixels have not been scaled by the maximum value.

For the spring rolls dataset, we manually select four regions from five scans. In each scan, we select a region containing no items, one containing spring rolls and finally two regions with the most visually distinct foreign objects. This selection process is depicted in Fig. 4. To encode the neighborhood information, we treat a single observation of a given pixel as the 5×128 values from itself and the pixels directly above, below and on the left and right. So each observation contains $128 \times 5 = 640$ variables. This should give us more robustness for detection of different materials.

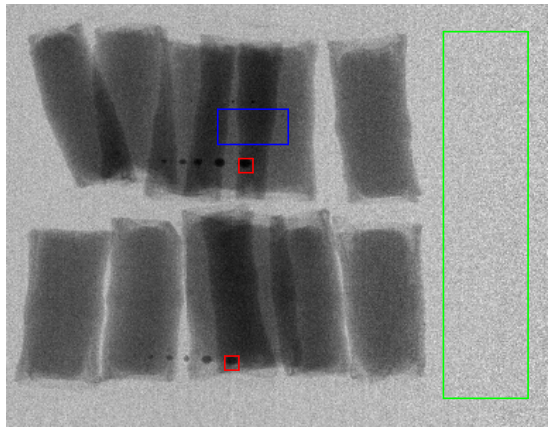


Fig. 4. Selection of data used for training the classifier. Three classes are selected, the enclosed region of the green box represents the no item or air class, the blue region represents the food item and the red boxes represent the foreign objects. For this illustration, the red boxes are a little bit larger than in practice.

After selecting the regions from five scans we can generate a matrix where rows represent observations and the $p = 640$ variables are represented as columns. Each image yields around 30 pixels of foreign objects, the other classes (spring rolls and air) are randomly subsampled, such that we have equal number of observations in each class, so we end up with a matrix $\mathbf{X}$ of dimension $n \times p = n \times 640$, where the value of n is around 500 to 600 pixels. The labels are represented in an indicator matrix $\mathbf{Y}$, which has an equal number of rows as $\mathbf{X}$, but the number of columns is equal to the number of classes K, in this case, three. If observation i belongs to class j, then $\mathbf{Y}_{ij}$ is 1 and the other values in the same row are zero. We employ a similar methodology to setup the minced meat dataset. After the data is set up we normalize it by subtracting the mean and scaling the variables such that they have unit variance. The same procedure is done for the minced meat data, so we have two different data sets.

The R programming language [11] was used for all the processing of the image data and classification. The package `imager` [1] was used for manually extracting regions from the images. The `imager` package is an R interface to the Cimg C++ library [13].

3.1 Sparse Linear Discriminant Analysis

We apply SDA, [4], to solve the present classification problem. SDA is a statistical learning method [7], which falls under the category of supervised classifiers and is a sparse version of the more basic method linear discriminant analysis (LDA). The method can handle many classes and it can also handle the case when we have more variables than observation, $p \gg n$ problems, with regularization. The underlying problem can be formulated in different ways, but we approach it by sparse optimal scoring.

$$\begin{aligned}(\boldsymbol{\theta}_k, \boldsymbol{\beta}_k) = \underset{\boldsymbol{\theta}\in\mathbb{R}^K, \boldsymbol{\beta}\in\mathbb{R}^p}{\arg\min} \; & \|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}\|^2 + \lambda_1 \boldsymbol{\beta}^T \boldsymbol{\Omega} \boldsymbol{\beta} + \lambda_2 \|\boldsymbol{\beta}\|_1 \\ \text{s.t.} \; & \tfrac{1}{n}\boldsymbol{\theta}^T \mathbf{Y}^T \mathbf{Y} \boldsymbol{\theta} = 1, \;\; \boldsymbol{\theta}^T \mathbf{Y}^T \mathbf{Y} \boldsymbol{\theta}_l = 0 \; \forall l < k,\end{aligned} \tag{3}$$

In the sparse optimal scoring formulation, (Equation 3), the $\mathbf{X}$ data matrix and $\mathbf{Y}$ indicator matrix are the same as the ones described in the last section. We seek the discriminant vectors $\boldsymbol{\beta_i}$, $i \in \{1, 2, ..., K-1\}$, which we use to project the data into a lower dimensional space. Classification is performed in this lower dimensional space by classifying an observation as belonging to the class corresponding to the nearest centroid, where the labeled data is used to estimate the centroids. $\boldsymbol{\theta}$ serves the purpose of avoiding the masking problem, i.e. such that class centroids are not colinear in the lower dimensional representation, it is not needed for classification of new observations, only in training. The second and third terms in the minimization problem form an elastic net penalty [14], which serves as a regularizer and allows us to solve the problem in the case of more variables than observations. The scaling parameters λ_1 and λ_2 are selected via cross-validation. In our case, the $\boldsymbol{\Omega}$ in the first part of the elastic net penalty is a diagonal $p \times p$ matrix, which penalizes the magnitude of the coefficients in the $\boldsymbol{\beta}_i$ vectors.

The SDA method, (without the elastic net penalty), is a linear map to a lower dimensional representation like principal component analysis (PCA), but in SDA we project the data to a lower dimensional space such that we maximize the variance between classes and minimize the variation within classes for optimal linear separation. There are also sparse versions of PCA [5], which SDA is more similar to. The centroids in the lower dimensional space can be thought of as means of different multivariate normal distributions which all have the same covariance structure, therefore we get linear decision boundaries, like in classical LDA [7]. Other classifiers can also be used on the projected data, an example of such projected data is depicted in Fig. 5, it is the training data used for the minced meat dataset.

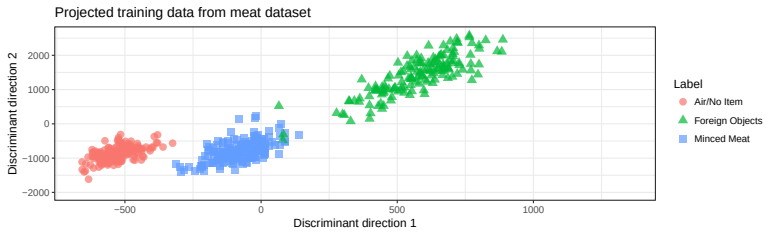


Fig. 5. Visualization of the training data used for the classification of the minced meat data after projection with the discriminant vectors. The classes are almost perfectly separated along the first discriminant direction, while the foreign objects and meat are rather close. The second discriminant vector further separates the meat and foreign objects.

4 Results

The classification results from the classifier trained on the spring rolls data are summarized in Table 2, where we show the number of detected pixels in images not contained in the training or validation set. The SDA method was trained only on 6 images from dataset 1 and the training data images are not included in the table. The final training set was balanced and consisted of 519 measurements with 640 variables. The training error is 0% with 48% sparsity, i.e. only 48% of the values in the discriminant vectors are non-zero. This means that almost half the variables are irrelevant for this particular classification task. Values corresponding to the same channels in the pixels were non-zero very consistently in the 640 variables corresponding to a pixel and its four neighbors. 10-fold cross-validation was used to tune the sparsity parameter and should be noted that sparsity of 5% only yielded 5% error on the training data, meaning that very few variables are more critical than others.

One thing to note about the results in Table 2 is that consistently fewer foreign object pixels were detected in the scans which only contained foreign objects. That is because the training set only consisted of foreign objects superimposed on the spring rolls. A very low number of false positives were detected in the data set which consisted of only spring rolls and no foreign objects (average 4.21 pixels). In most of the images, only 0,1,2 or 3 pixels were detected except for a single outlier were 42 pixels were detected, which inflates the standard deviation.

The classification results for the classifier which was trained on the minced meat dataset are summarized in Table 3, where we show the number of pixels detected in images which were not part of the training or validation set. The main difference from the spring rolls dataset is that no false positives were discovered in the scans that contained no foreign objects. Otherwise, there is consistent variation between datasets in the number of foreign object pixels discovered. Another difference is that in the cross-validation process the sparsity regulariza-

Table 2. Results for the spring rolls dataset where we present the number of pixels detected as foreign objects on average in 4 different datasets. Dataset 1 consists of 8 bags of spring rolls with foreign objects superimposed. Dataset 2 consists of the same 8 bags, where the spring rolls have been refrozen and scanned a day later. The other datasets are scans with only the spring rolls or only the foreign objects.

Spring Rolls	*Dataset 1*	*Dataset 2*	*Only FOs*	*no FOs*
Average number of detected pixels	155.87	158.00	112.60	4.21
Standard deviation of detected pixels	16.18	38.01	40.73	11.31
Number of images predicted on	136	149	10	14

tion parameter that was chosen yields 17% sparsity. This means that we could certainly get away with storing fewer data for this approach.

Table 3. Results for the minced meat dataset where we present the number of pixels detected as foreign objects on average in 5 different datasets. The first dataset is 5 scans of nothing, i.e. empty scans. Datasets 1,2 and 3 consist of 10 scans each, two which were used for training. The difference of the foreign objects in each dataset can be found in Table 1. The last dataset is meat without any foreign objects.

Minced Meat	*Nothing*	*Dataset 1*	*Dataset 2*	*Dataset 3*	no FOs
Average number of detected pixels	0.00	117.62	190.12	211.50	0.00
Standard deviation of detected pixels	0.00	11.49	9.06	8.90	0.00
Number of images predicted on	5	8	8	8	5

Some example results are presented in Fig. 6. The foreign objects detected were mostly metals, and also some stone pebbles, quartz, and glass. The smallest objects detected are 2-3 millimeters in diameter.

5 Discussion

We achieve good detection on the type of foreign objects we can detect. The items used for training are the ones that are represented in the detection, we do not manage to generalize to all the scanned foreign objects. This is mainly because of low signal to noise ratio, especially for the objects that have low absorption. The undetected items are also not represented in the training set, if they would be included we would potentially get more false positives, because they are not as distinct as the metals, stone pebbles, and quartz, thus moving the decision boundary closer to the food item. One approach to try to get a more general detector would be to train with as many different types of foreign objects as possible, that are detectable, and find the discriminant vectors from SDA or use a semi-supervised approach. Then the new data can be projected similar to Fig. 5 and the air and food item classes can be described there with normal distributions or other ways to encapsulate the two classes. Then everything outside those classes would be classified as foreign objects. Each type of

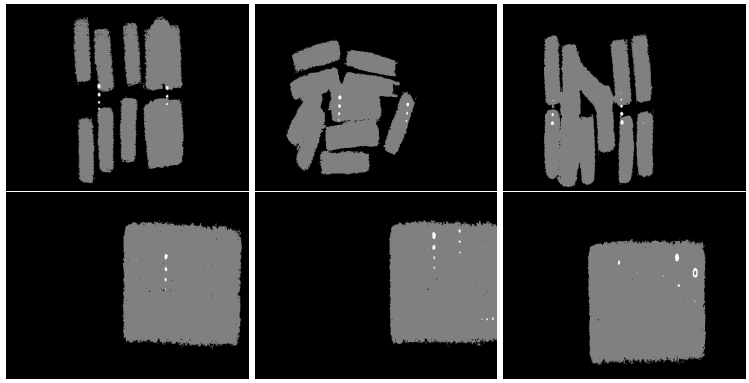


Fig. 6. Example results from the classifier on both datasets. The white color indicates foreign objects. The top row shows the spring rolls and the bottom row shows from left to right an example from dataset 1,2 and 3. The foreign objects detected were stone pebbles, metals, quartz, and glass. These were the foreign objects used for training.

foreign object could also be modeled on its own, then we could encapsulate what is known, and everything outside the known classes would belong to an *unknown* class.

One way to augment the current measurements would be to have some way to estimate the thickness of the materials that are being scanned. That would be a good additional variable for a data-driven approach, or it could be used for normalization. This is already being done in some commercial products using a laser to map the height of the food product.

We can say that for the data sets used we can get away with storing half of the data or less. But this is both application and material dependent. Different applications could yield different foreign objects and different materials have different intensity profiles, meaning that some variables/channels are redundant in some cases and useful in others. This would have to be dealt with for each specific application.

6 Conclusion

We have demonstrated that we can achieve robust detection of certain foreign objects in the data sets used in this work. This was done in a completely data-driven manner by applying a sparse classifier to the normalized data. There is great potential for using an approach similar to the one we present, which could help with storing fewer data and processing the results faster.

Acknowledgements

This work is supported by the Lundbeck foundation, the Technical University of Denmark and the NEXIM research project funded by the Danish Council for Strategic Research (contract no. 11-116226) within the Program Commission on Health, Food and Welfare. We would like to thank the anonymous reviewers for providing valuable comments on the manuscript.

References

1. Barthelme, S.: imager: Image Processing Library Based on 'CImg' (2016), `https://CRAN.R-project.org/package=imager`, r package version 0.31
2. Brambilla, A., Ouvrier-Buffet, P., Rinkel, J., Gonon, G., Boudou, C., Verger, L.: Cdte linear pixel x-ray detector with enhanced spectrometric performance for high flux x-ray imaging. IEEE Transactions on Nuclear Science 59(4), 1552–1558 (2012)
3. Chung, D., Keles, S., et al.: Sparse partial least squares classification for high dimensional data. Statistical applications in genetics and molecular biology 9(1), 17 (2010)
4. Clemmensen, L., Hastie, T., Witten, D., Ersbøll, B.: Sparse discriminant analysis. Technometrics (2012)
5. d'Aspremont, A., El Ghaoui, L., Jordan, M.I., Lanckriet, G.R.: A direct formulation for sparse pca using semidefinite programming. SIAM review 49(3), 434–448 (2007)
6. Einarsdóttir, H., Emerson, M.J., Clemmensen, L.H., Scherer, K., Willer, K., Bech, M., Larsen, R., Ersbøll, B.K., Pfeiffer, F.: Novelty detection of foreign objects in food using multi-modal x-ray imaging. Food Control 67, 39–47 (2016)
7. Friedman, J., Hastie, T., Tibshirani, R.: The elements of statistical learning, vol. 1. Springer series in statistics Springer, Berlin (2001)
8. Gorecki, A., Brambilla, A., Moulin, V., Gaborieau, E., Radisson, P., Verger, L.: Comparing performances of a cdte x-ray spectroscopic detector and an x-ray dual-energy sandwich detector. Journal of Instrumentation 8(11), P11011 (2013)
9. Pfeiffer, F., Bunk, O., David, C., Bech, M., Le Duc, G., Bravin, A., Cloetens, P.: High-resolution brain tumor visualization using three-dimensional x-ray phase contrast tomography. Physics in medicine and biology 52(23), 6923 (2007)
10. Pfeiffer, F., Weitkamp, T., Bunk, O., David, C.: Phase retrieval and differential phase-contrast imaging with low-brilliance x-ray sources. Nature physics 2(4), 258–261 (2006)
11. R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2015), `https://www.R-project.org/`
12. Rebuffel, V., Rinkel, J., Tabary, J., Verger, L.: New perspectives of x-ray techniques for explosive detection based on cdte/cdznte spectrometric detectors. In: International Symposium on Digital Industrial Radiology and Computed Tomography–We. vol. 2, pp. 1–8 (2011)
13. Tschumperlé, D.: The cimg library. In: IPOL 2012 Meeting on Image Processing Libraries. pp. 4–pp (2012)
14. Zou, H., Hastie, T.: Regularization and variable selection via the elastic net. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 67(2), 301–320 (2005)

APPENDIX F

Computer Aided Identification of Movements Related to Parkinson's Disease

Computer Aided Identification of Movements Related to Parkinson's Disease

Gudmundur Einarsson[✉1], Line K. H. Clemmensen[1], Ditte Rudå[2], Anders Fink-Jensen[3], Jannik B. Nielsen[1], Anne Katrine Pagsberg[2], Kristian Winge[4], and Rasmus R. Paulsen[1]

[1] DTU Compute, Technical University of Denmark, guei@dtu.dk
[2] Child and Adolescent Mental Health Center
Mental Health Services, Capital Region Denmark
Faculty of Health Science, University of Copenhagen
[3] Psychiatric Centre Copenhagen (Rigshospitalet)
Laboratory of Neuropsychiatry, University Hospital Copenhagen
[4] Department of Neurology, Zealand University Hospital, Roskilde, Denmark

Abstract. We present a framework for assessing which types of simple movement tasks are most discriminative between healthy controls and Parkinsons patients. We collected movement data in a game-like environment, where we used the Microsoft Kinect sensor for tracking the users joints. We recruited 63 individuals for the study, of whom 30 had been diagnosed with Parkinsons disease. A physician evaluated all participants on movement-related rating scales, e.g., elbow rigidity. The participants also completed the game task, moving their arms through a specific pattern. We present an innovative approach for data acquisition in a game-like environment, and we propose a novel method, sparse ordinal regression, for predicting the severity of motion disorders from the data.

Keywords: Game-aided diagnosis, Kinect, Parkinson's disease, sparse, ordinal, classification

1 Introduction

Parkinson's disease (PD) is a long-term neurodegenerative disease, where the significant symptoms are tremor, rigidity, slowness of movements and difficulty walking. Currently, there are 7 million individuals affected on a global scale where the disease has a severe socioeconomic effect and reduces the quality of life. The condition has a significant financial impact on health care systems and society [21][17].

PD is now known to be caused by an interplay of environment and several genetic factors [14], but there is no known cure, but mainly treatment to improve symptoms. Treatment consists mainly of medication, surgery and physical therapy. Recent studies have also shown relief of symptoms via an improved diet and rehabilitation [5][2]. There is no diagnostically conclusive test available

yet. The current diagnosis is clinical, questionaires and movement tests, and may be missed or misdiagnosed since the symptoms are common to other diseases/disorders. At the time of PD diagnosis, the disease has often progressed to an advanced stage with motor symptoms and neurophysiological damage.

It is of great importance to develop tools that can aid an unbiased diagnosis for PD in earlier stages of the disease. Some symptoms commonly appear before the motor-symptoms, such as depression, feeling tired and weak, reduced ability to smell, problems with blood pressure, heart rate, sleep disturbances and digestion [10].

Increasing the detection rate for early cases is very ambitious, especially if we do not resort to novel diagnosis tools. It would be easier, more accurate, and less prone to bias, to make a computerized diagnostic test a part of the regular screening processes. Using data from such a tool would allow us to model individual abnormalities more accurately, and make personalized and accurate predictions of disease status and progression, by comparing to earlier screenings. Another way to achieve this would be to have access to a proxy variable, that the patient can choose to send for analysis, such as data from a personal health monitor, movement data from a GPS tracker, mobile phone data, or data from a video game.

1.1 Goals

Our ambition is to predict the clinical ratings made by the physician of the underlying movement disorders from the motion tracking data, and to identify what part of the movement sequences are best suited for this task. This problem has several difficulties, of which the major ones are: 1) There are few observations compared to the number of variables. 2) The labels we want to predict are ordinal. 3) The classes are imbalanced.

1.2 Related work

In recent years the Kinect sensor has been widely used for retraining and physical therapy. Galna et al. presented such an application for PD patients[13]. A review of the usage of the Kinect sensor for medical purposes is presented in [18], of which most of the work is development and testing of physical therapy systems for various diseases and medical conditions. Of the studies covered in [18], three describe assessment of conditions, related to facioscapulohumeral muscular dystrophy (FSHD) [16], stroke [1] and balance in the elderly [11]. The capabilities of the Kinect are limited, as reported in [12]; thus we do not expect to be able to detect or predict the presence of low amplitude tremors or movement disorders related to smaller movements.

We want to predict the score from the clinically collected movement data and identify the movement sequences related to PD. Due to the high number of variables, we propose to use a novel method, sparse ordinal regression. This method builds upon sparse discriminant analysis (SDA) [8] by adapting the data replication method to the sparse setting [6] to handle ordinal labels. We further

extend the novel optimization approaches presented in [4] for sparse ordinal regression. The data replication method works on the principle of transforming an ordinal classification problem into multiple binary classification problems. These binary classification problems are solved together to find a common hyperplane that separates each pair of classes corresponding to adjacent ordinal labels. The difference between the hyperplanes corresponding to different classification boundaries are biases.

In the past years, multiple methods have appeared which can handle feature selection and classification problems of the type $p \gg n$, most notably Sparse Discriminant Analysis (SDA) by [8] and Sparse Partial Least Squares for Classification by [7]. Other algorithms commonly used to solve such problems, where the focus is not necessarily classification, are elastic net by [23] and sparse principal component analysis by [9]. Using an l_1-norm regularizer in the model formulation ensures that variable selection is performed in the model optimization process which gives leverage for the user to interpret the non-zero parameters in the model. Incorporation of an l_1-norm regularizer is influenced by the Lasso [20], which uses the l_1-norm to relax the vector cardinality function in the best feature subset problem for linear regression.

Ordinal labels appear in a multitude of applications, e.g., surveys, medical rating scales and concerning online user reviews. We believe that the methodology can be applied to a variety of other problems in the future.

1.3 Contributions

The main contributions of this paper consist of a novel game-like framework, the Motor-game, for assessing arm-movement in individuals with movement-related disorders in the arms. We further propose a novel method for performing classification from this data, sparse ordinal regression, allowing us to summarize a whole run into a single score.

2 Methods & Data

We begin by describing the particularities of adapting the data replication method to the sparse setting to obtain ordinal SDA. We then describe the data.

2.1 The Motor-Game

We have developed a game-like environment, which we call the *Motor-game*, where we use the Microsoft Kinect sensor [22] and the associated software framework to do motion tracking of the players (See Fig. 2.1) [3].

The motor-game is designed to capture a range of motions from the hands and arms. There are three levels in the Motor-game, where here we focus on data from the first level. The first level has 22 *tasks*. In the first 11 tasks, a button appears on the right side of the screen, and the player needs to react, *catch* the button and keep the hand stable there for one second. The following

Fig. 1. *Left*: Screenshot from the motorgame. The player sees his pose reflected as a stick figure and needs to make the stick figure's hands hover over the buttons as fast as possible. *Right*: View from behind a player playing the motor-game.

11 tasks are similar but for the left hand. For each player, the buttons appear in the same location, meaning that their hands have comparable positions between playthroughs. The distances between appearances of the buttons vary, forcing the player to perform large and smaller motions. Using the tracking software from the Kinect, we obtain 30 measurements per second of several joints, wrists, elbows, shoulders and neck, in the upper body. One of the main reason to make this data collection process in a game-like environment is to keep the players motivated to perform as well as they can, and to make the process more enjoyable, similar to games that have been made for physiotherapy in PD patients [13].

2.2 Sparse Ordinal Regression

In [8], Clemmensen et al. presented the sparse optimal scoring problem (SOS), which is the formulation we employ to solve sparse ordinal regression. SDA works in some sense like a supervised version of Principal Component Analysis (PCA), where we seek to find discriminant vectors to project the data to a lower dimensional representation, where we balance the objectives of minimizing variation within classes, maximizing variation between classes and feature selection. New samples are then traditionally classified according to the nearest centroid after projection. We reformulate the SOS criterion presented in [8] for ordinal labels.

$$\underset{\boldsymbol{\theta}\in\mathbf{R}^2,\,\boldsymbol{\beta}_{\mathrm{Ord}}\in\mathbf{R}^{p+K-1}}{\arg\min} \|\boldsymbol{Y}_{\mathrm{Ord}}\boldsymbol{\theta} - \boldsymbol{X}_{\mathrm{Ord}}\boldsymbol{\beta}_{\mathrm{Ord}}\|_2^2 + \lambda_2\boldsymbol{\beta}_{\mathrm{Ord}}^T\hat{\boldsymbol{\Omega}}\boldsymbol{\beta}_{\mathrm{Ord}} + \lambda_1\sum_{i=1}^{p}|\beta_i|$$

$$\text{s.t. } \frac{1}{n}\boldsymbol{\theta}^T\boldsymbol{Y}_{\mathrm{Ord}}^T\boldsymbol{Y}_{\mathrm{Ord}}\boldsymbol{\theta} = 1. \tag{1}$$

When we solve the problem in Eq. 1 we seek a sparse discriminant vector $\boldsymbol{\beta}_{\mathrm{Ord}}$, which we can then use to project the data from feature space to a one-dimensional representation. In the ordinal case, we cast our problem as a binary classification problem, which only yields a single discriminant vector $\boldsymbol{\beta}_{\mathrm{Ord}}$, simplifying the interpretation of the solution. $\boldsymbol{\beta}_{\mathrm{Ord}}$ is a vector of length $p + K - 1$, (where p is the number of variables and K the number of classes). The first p

parameters correspond to the original variables that we can interpret. The extra $K-1$ parameters are the additional biases introduced by the data replication method, allowing us to classify the projected points, based on where they end up concerning the biases.

[8] show that for a given $\boldsymbol{\beta}_{\text{Ord}}$ one can find $\boldsymbol{\theta}$ in polynomial time. For a given $\boldsymbol{\theta}$ the problem formulation is an elastic net problem, and the problem can be solved with the LARS-EN algorithm by [23]. We, however, approach the optimization from the point of proximal gradient (PG) methods and alternating direction method of multipliers (ADMM), using the soft thresholding operator to deal with the sparse regularizer in the same manner as [4].

2.3 Using Ordinal Labels via Data Replication

A natural assumption for an ordinal classifier of K classes, is to have $K-1$ non-intersecting classification boundaries, where boundary i separates classes 1 to i from classes $i+1$ to K. In our case, that means finding a hyperplane and a set of biases to shift the hyperplane between classes. We extend the data replication method of [6] to the sparse setting, by adapting the optimization, such that it does not regularize these new bias parameters.

We construct a new data matrix $\boldsymbol{X}_{\text{Ord}}$ and labels $\boldsymbol{Y}_{\text{Ord}}$ according to the data replication method. We then define a new $(p+K-1)\times(p+K-1)$ regularization matrix $\hat{\boldsymbol{\Omega}}$.

$$\hat{\boldsymbol{\Omega}} := \begin{bmatrix} \boldsymbol{\Omega} & 0 \\ 0 & 0 \end{bmatrix}, \qquad \boldsymbol{\beta}_{\text{Ord}}^T := \left[\beta_1\ \beta_2\ \ldots\ \beta_p\ b_1\ b_2\ \ldots\ b_{K-1}\right], \tag{2}$$

where $\boldsymbol{\Omega}$ is a $p\times p$ positive semi-definite regularization matrix for the parameters corresponding to the p original variables. The final adjustments relates to the l_1-norm in Eq. 1. In the soft-thresholding step of the ADMM and PG algorithms used to find $\boldsymbol{\beta}_{\text{Ord}}$, we only apply soft-thresholding to the first p elements.

The resulting $\boldsymbol{\beta}_{\text{Ord}}$ vector is show in Eq. 2. The first part is composed of a traditional discriminant vector, corresponding to the first p elements, and then $K-1$ biases, denoted b_i, for $i \in \{1, 2, ..., K-1\}$. The proofs of convergence to stationary points, of the algorithms in [4], extend naturally to our approach.

2.4 Data and Experiments

We conducted a study, where we collected data from 63 individuals, of whom 33 were healthy controls and 30 PD patients. Detailed description of the cohort can be found in [3]. Each participant played the Motor-game two times; the first one is a trial run to get familiar with the game. Motion tracking data was collected during the playthroughs. A physician then evaluated the participants on various rating scales, of which we are concerned with the results from the *Simpson-Angus-Scale* (SAS) [19], in particular, item 4, which involves elbow rigidity. Furthermore, the PD patients were evaluated on the *Movement Disorder Society Unified Parkinson's Disease Rating Scale* (MDS-UPDRS) [15]. On MDS-UPDRS we are most focused on items 3.3b rigidity of right hand, 3.4a finger

tapping in the right hand and 3.5a hand movement for the right hand. We picked out these items since they were *a priori* thought to have the most substantial correspondence with the data from the Motor-game. Items from the rating scales reflecting motor symptoms in hands and arms were included. Exclusion was made if there were too few participants affected. See Fig. 2.4 for prevalence and severity of the observed motion conditions in the data. A more detailed description of the dataset and the Motor-game can be found in [3].

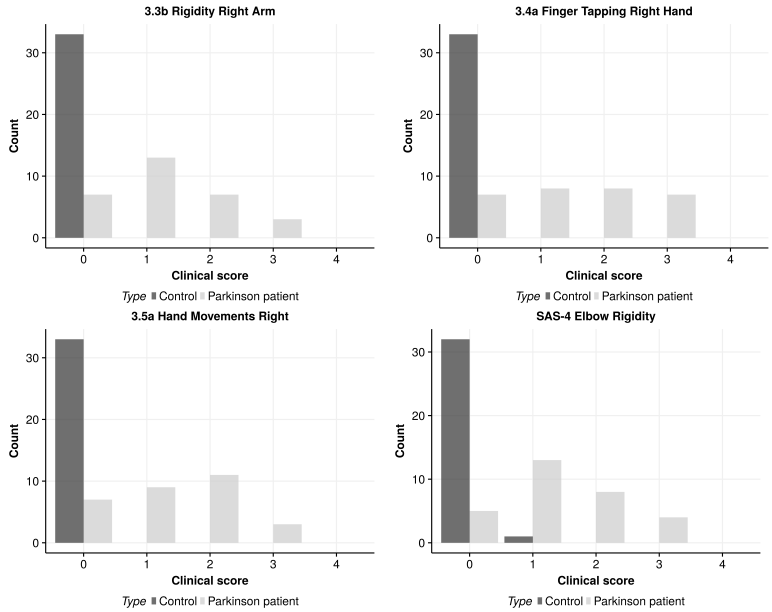


Fig. 2. Prevalence of labels in the dataset for the conditions we focus on. The first three plots from the left correspond to MDS-UPDRS items 3.3b *rigidity right arm*, 3.4a *finger tapping right hand* and 3.5a hand movement right. The final item corresponds to *elbow rigidity* on the SAS scale.

For analyzing the movements of the participants, we used the tracked position of their wrists. For the first 11 tasks, we used the avatar screen coordinate vertical position for the right wrist. The choice of this coordinate is because the avatar has been scaled according to an initial estimate of the player's arm length, making on-screen positions comparable between players. For the following 11 tasks, we used the corresponding coordinates for the left wrist. For each of the 22 tasks, we used measurements for the first second of play. The participants had not reacted in the first five measurements, so we excluded those measurements. The

first second of the game is enough for the person to respond and start moving. We can see the contrast between a fast and slow reacting participant in Fig. 2.4. This yields, in the end, a total of $p = 20 \times 22 = 440$ variables per participant. We denote m_{i_S} as the mean of the first three measurements for task i and m_{i_E} as the average for the last three measures for task i.

$$\tilde{x}_{ji} := \frac{x_{ji} - m_{i_S}}{|m_{i_S} - m_{i_E}|} \tag{3}$$

We further scale the j-th measurement x_{ji} from task i as depicted in Eq. 3. Due to variation in the end and starting position, this scaling ensures that the data is more robust to reactions of the participants.

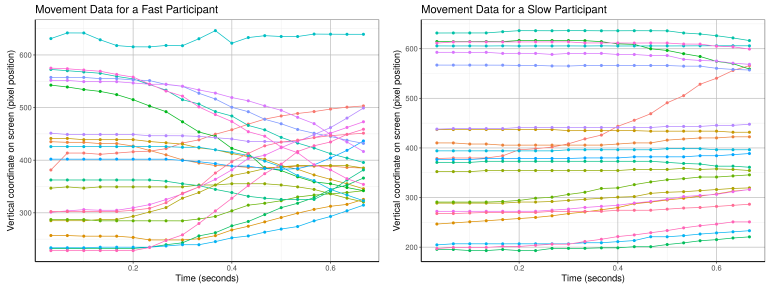


Fig. 3. Data used for the experiment, vertical position of two subjects' hands over the first second of the 22 tasks. On the left we have a participant that generally reacts fast, on the right we have a more slow reacting individual.

We normalize the data before applying sparse ordinal regression by subtracting the mean for each variable and scaling the standard deviation to one. We report the balanced accuracy for leave one cross-validation, where we allow the regularization parameters λ_1 and λ_2 from Eq. 1 to be in the set $\{0.1, 0.01, 0.001\}$. We perform this experiment for the four labels shown in Fig. 2.4. Note that the three variables for MDS-UPDRS were only measured for the Parkinson patients; thus the controls were assumed to have a score of zero.

3 Results

The leave one out cross-validation balanced-accuracy and the best cross-validation parameters are depicted in Tab. 3. The corresponding confusion matrices are shown in Tab. 3. We can see that the predictions are somewhat accurate, although the LOO-CV most likely overestimates the real accuracy. Note that the best forecasts for class zero are in MDS-UPDRS 3.4a and SAS-4. We assume that the controls have a score of zero in the MDS-UPDRS variables since they

were not measured, this may not be entirely correct, a few individuals in the control group had a score of one for SAS-4.

Table 1. Balanced accuracy from leave one out cross-validation on the four responses.

Variable	Description	Balanced Accuracy	λ_1	λ_2
SAS-4	Elbow rigidity	0.481	0.01	0.01
UPDRS-3.3b	Rigidity right arm	0.477	0.1	0.001
UPDRS-3.4a	Finger tapping right	0.463	0.1	0.001
UPDRS-3.5a	Hand movement right	0.388	0.001	0.001

Table 2. Confusion matricies for predictions (with best performing regularization parameters) from the item left out in the leave one out cross-validation. Most of the predictions are concentrated around the correct label, but most of them have difficulties with the higher labels.

SAS-4		True			
		0	1	2	3
Predicted	0	23	1	4	1
	1	11	9	3	1
	2	1	4	1	0
	3	2	0	0	2

3.3b		True			
		0	1	2	3
Predicted	0	22	2	4	1
	1	13	9	3	0
	2	4	1	0	0
	3	1	1	0	2

3.4a		True			
		0	1	2	3
Predicted	0	26	2	4	2
	1	13	6	1	4
	2	1	0	3	0
	3	0	0	0	0

3.5a		True			
		0	1	2	3
Predicted	0	20	3	6	1
	1	11	4	3	0
	2	8	2	1	1
	3	1	0	1	1

4 Conclusions

We have presented a novel approach for assessing the severity of upper body motor symptoms in PD. The game-like environment has been proven to work both in the clinic, or in the patient's home. Longitudinal studies are needed to establish further the potential of this approach, where the data needs to be examined with change point detection methods. Monitoring the movement in correspondence with the presence of pre-movement related symptoms has potential to create novel tools for early detection of PD.

Acknowledgements Gudmundur Einarson's PhD is funded jointly by the Lundbeck Foundation and the Technical University of Denmark. The study has received grants from The Capital Region of Denmark, Research Fund for Health Promotion and The Capital Region of Denmark, Mental Health Services Research Fund.

References

1. Adams, R.J., Lichter, M.D., Krepkovich, E.T., Ellington, A., White, M., Diamond, P.T.: Assessing upper extremity motor function in practice of virtual activities of daily living. IEEE Transactions on Neural Systems and Rehabilitation Engineering 23(2), 287–296 (2015)
2. Ahlskog, J.E.: Does vigorous exercise have a neuroprotective effect in parkinson disease? Neurology 77(3), 288–294 (2011)
3. Anonymous: Exploring alternative ways to assess parkinson's patients. (2018)
4. Atkins, S., Einarsson, G., Ames, B., Clemmensen, L.: Proximal methods for sparse optimal scoring and discriminant analysis. arXiv preprint arXiv:1705.07194 (2017)
5. Barichella, M., Cereda, E., Pezzoli, G.: Major nutritional issues in the management of parkinson's disease. Movement disorders 24(13), 1881–1892 (2009)
6. Cardoso, J.S., Costa, J.F.: Learning to classify ordinal data: The data replication method. Journal of Machine Learning Research 8(Jul), 1393–1429 (2007)
7. Chung, D., Keles, S.: Sparse partial least squares classification for high dimensional data. Statistical applications in genetics and molecular biology 9(1) (2010)
8. Clemmensen, L., Hastie, T., Witten, D., Ersbøll, B.: Sparse discriminant analysis. Technometrics 53(4), 406–413 (2011)
9. d'Aspremont, A., Ghaoui, L.E., Jordan, M.I., Lanckriet, G.R.: A direct formulation for sparse pca using semidefinite programming. In: Advances in neural information processing systems. pp. 41–48 (2005)
10. Duncan, G.W., Khoo, T.K., Yarnall, A.J., O'Brien, J.T., Coleman, S.Y., Brooks, D.J., Barker, R.A., Burn, D.J.: Health-related quality of life in early parkinson's disease: The impact of nonmotor symptoms. Movement disorders 29(2), 195–202 (2014)
11. Dutta, A., Chugh, S., Banerjee, A., Dutta, A.: Point-of-care-testing of standing posture with wii balance board and microsoft kinect during transcranial direct current stimulation: a feasibility study. NeuroRehabilitation 34(4), 789–798 (2014)
12. Galna, B., Barry, G., Jackson, D., Mhiripiri, D., Olivier, P., Rochester, L.: Accuracy of the microsoft kinect sensor for measuring movement in people with parkinson's disease. Gait & posture 39(4), 1062–1068 (2014)
13. Galna, B., Jackson, D., Schofield, G., McNaney, R., Webster, M., Barry, G., Mhiripiri, D., Balaam, M., Olivier, P., Rochester, L.: Retraining function in people with parkinsons disease using the microsoft kinect: game design and pilot testing. Journal of neuroengineering and rehabilitation 11(1), 60 (2014)
14. Gan-Or, Z., Dion, P.A., Rouleau, G.A.: Genetic perspective on the role of the autophagy-lysosome pathway in parkinson disease. Autophagy 11(9), 1443–1457 (2015)
15. Goetz, C.G., Tilley, B.C., Shaftman, S.R., Stebbins, G.T., Fahn, S., Martinez-Martin, P., Poewe, W., Sampaio, C., Stern, M.B., Dodel, R., et al.: Movement disorder society-sponsored revision of the unified parkinson's disease rating scale (mds-updrs): Scale presentation and clinimetric testing results. Movement disorders 23(15), 2129–2170 (2008)
16. Han, J.J., Kurillo, G., Abresch, R.T., Bie, E., Nicorici, A., Bajcsy, R.: Reachable workspace in facioscapulohumeral muscular dystrophy (fshd) by kinect. Muscle & nerve 51(2), 168–175 (2015)
17. Huse, D.M., Schulman, K., Orsini, L., Castelli-Haley, J., Kennedy, S., Lenhart, G.: Burden of illness in parkinson's disease. Movement disorders 20(11), 1449–1454 (2005)

18. Mousavi Hondori, H., Khademi, M.: A review on technical and clinical impact of microsoft kinect on physical therapy and rehabilitation. Journal of medical engineering 2014 (2014)
19. Simpson, G., Angus, J., et al.: A rating scale for extrapyramidal side effects. Acta Psychiatrica Scandinavica 45(S212), 11–19 (1970)
20. Tibshirani, R.: Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society. Series B (Methodological) pp. 267–288 (1996)
21. Tinelli, M., Kanavos, P., Grimaccia, F.: The value of early diagnosis and treatment in parkinsons disease. A literature review of the potetial clinical and socioeconomic impact of targeting unmet needs in Parkinsons disease. London School of Economics (2016)
22. Zhang, Z.: Microsoft kinect sensor and its effect. IEEE multimedia 19(2), 4–10 (2012)
23. Zou, H., Hastie, T.: Regularization and variable selection via the elastic net. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 67(2), 301–320 (2005)

APPENDIX G

Exploring Movement Impairments in Patients with Parkinson's disease using the Microsoft Kinect Sensor

Exploring Movement Impairments in Patients with Parkinson's disease using the Microsoft Kinect Sensor

Ditte Rudå[a], Gudmundur Einarsson[b*], Anne Sofie Schott Andersen[a], Jannik Boll Nielsen[b], Christoph Correll[c], Kristian Winge[d], Line K. H. Clemmensen[b], Rasmus Reinhold Paulsen[b], Anne Katrine Pagsberg[a] and Anders Fink-Jensen[e]

[a]*Centre for Child and Adolescent Mental Health, Mental Health Services, Capital Region of Denmark & Faculty of Health Science, University of Copenhagen, Denmark*

[b]*Section for Image Analysis and Computer Graphics, DTU Compute, Technical University of Denmark*

[c]*Hofstra North Shore Long Island Jewish School of Medicine and the Zucker Hillside Hospital, New York, 75-59 263rd Street,Glen Oaks, 11004 New York, U.S.A*

[d]*Department of Neurology, Zealand University Hospital, Roskilde, Denmark*

[e]*Psychiatric Centre Copenhagen (Rigshospitalet) and Laboratory of Neuropsychiatry, University Hospital Copenhagen, Denmark*

[*]*Correspondence to: Gudmundur Einarsson, section for Image Analysis and Computer Graphics, DTU Compute, Technical University of Denmark, Richard Petersens Plads, building 321, office 224. Tel.: +4552585740; E-mail: guei@dtu.dk. Fax number +45 45881399*

Abstract

BACKGROUND: Current clinical assessments of motor symptoms in Parkinson's Disease, as well as motor symptoms induced by antipsychotic medication in psychiatric patients, are often limited to clinical examination and rating scales with inherent observer bias.

OBJECTIVE: To develop and test a computer application using the Microsoft Kinect sensor to assess performance related bradykinesia and rigidity associated with Parkinson's Disease.

METHODS: The application (*Motorgame*) was tested in a group of patients with Parkinson's Disease (PD) and healthy controls in order to investigate its applicability. Participants were neurologically evaluated, assessed by the Movement Disorder Society Unified Parkinson's Disease Rating Scale and standardized clinical side effect rating scales, i.e. UKU Side Effect Rating Scale and the Simpson-Angus Scale. In addition, tests of information processing (Symbol Coding Task) and motor speed (Token Motor Task) together with a questionnaire about their views on the Kinect game, were applied.

RESULTS: Thirty patients with PD and 33 healthy controls were assessed. In the patient group, we found a statistically significant ($p<0.05$) association between prolonged time of performance in the *Motorgame* and upper body rigidity and bradykinesia (MDS-UPDRS) with the strongest effects in finger tapping right hand, hand movements right hand, and rotation of right hand ($p<0.001$). In the entire group prolonged time of motor performance was significantly associated with higher SAS item scores on rigidity and higher hypokinesia scores (UKU) ($p<0.05$). A significant association between shortened motor performance and higher scores on information processing was found. No significant association was found between motor performance time and Token Motor Task, duration of PD, or hours of daily physical activity. The application was well accepted and preferred by 76% in the healthy control group and 53% in the patient group compared to traditional clinical examinations.

CONCLUSIONS: The Kinect application was able to detect common motor symptoms in PD in a statistically significant and clinically meaningful way, making it applicable for further testing in larger samples of patients with rigidity and bradykinesia.

Keywords: Parkinson’s disease, parkinsonism, hypokinesia, technology, movement disorders, computer-assisted diagnosis.

Introduction

Parkinson's disease (PD) is a progressive, degenerative movement disorder (Poewe, 2017). The neuropathology of PD is characterized by loss of dopamine neurons in the substantia nigra resulting in dysfunction of the nigrostriatal pathway, which cause perturbation of control and regulation of intentional motor movement. Bradykinesia, rigidity and rest tremor are the core motor symptoms (Postuma, 2015). Parkinsonian bradykinesia is the very core symptom and correlates with loss of dopaminergic deficiency (Benamer HT, 2003); it involves difficulties in planning, initiating and executing movements and difficulties in performing various tasks (Moustafa, 2016). As the disease progresses, postural instability often develops as a fourth cardinal symptom (Prescott, 2014). The standard qualitative assessment for evaluating PD bradykinesia as well as other PD symptoms is the Unified Parkinson's Disease Rating Scale (UPDRS) (Goetz, 2008).

When blocking the nigrostriatal pathway by D2-receptor antagonists, e.g.antipsychotics, symptoms similar to the ones observed in idiopathic PD can occur. Antipsychotic-induced parkinsonism is characterized by bradykinesia, rigidity and (variable) tremor, which reverse upon antipsychotic discontinuation (Hausner, 1983) (López-Sendón J, 2013). In clinical practice, antipsychotic-induced motoric side effects are usually assessed by clinical evaluation. However, a number of rating scales for the evaluation of motor side effects exist, including the Abnormal Involuntary Movement Scale (AIMS) (Guy, 1976), Simpson-Angus Scale (SAS) (Simpson, 1970) and the Barnes Akathisia Rating Scale (BARS) (Barnes, 1989). Another commonly used rating scale is the UKU Side Effect Rating Scale, where UKU is an acronym for the Danish name "Udvalg for Kliniske Undersøgelser" (Task force for clinical investigations). (Lingjaerde, 1987).

Although all the mentioned rating scales have undergone thoroughly scientific validation, the fact that the rating scales are observer-based inherently requires adequate training of clinicians in their use and makes these scales vulnerable to inter-observer variability (Wolff, 1999) (Lohr, 1992) (Dean, 2004). Hence, objective methods to detect and quantify movement disorders are needed.

Besides overcoming the issue of inter-observer variability, objective technology-based tools may well be used for home monitoring of symptoms.

Computerized analysis of human movements has been investigated for more than three decades (Moeslund, 2001). Until recently, human motion capture (the process of registering motion) has required an extensive setup, typically involving several cameras, structured light projectors, and special markers attached to the different relevant body parts, who is tracked. With introduction of the Microsoft Kinect in 2010, a low-cost and accessible human position and motion tracking technology has become available.

We have developed a simple game-like application using the Microsoft Kinect, where the user is asked to push buttons on a computer screen in a specific sequence, while the application tracks the movement of the major joints in the upper body. The objectives for this work is to assess the validity of application and to study to which degree it can complement the traditional observer-based rating scales.

Methods

Participants and in- and exclusion criteria

This study included patients (age ≥ 18 years) with idiopathic PD (ICD-10 G20.9) (World Health Organization, 1993) (Postuma, 2015) and a maximal score of 2.5 on the Hoehn & Yahr scale (i.e. postural stable) (Hoehn MM, 1967). Exclusion criteria were dementia, current psychosis or in current antipsychotic treatment. The patient group was matched 1:1 to healthy controls on age and gender. Exclusion criteria in the healthy controls were: Parkinson's Disease, dementia, current psychosis or antipsychotic treatment (lifetime).

Recruitment

Patients were recruited from the Department of Neurology, Bispebjerg University Hospital, Denmark, and from private practicing neurologists in the Capitol Region of Denmark. Healthy controls were recruited through local contacts, tennis clubs and senior centres in the Capitol Region. The technical work was initiated in January 2013 and the first demonstration model was ready for data collection in June 2013. Data collection was initiated in February 2014 and proceded until August 2016.

Data and Acquisition

The data for this study comes from the Microsoft Kinect v1 sensor (Zhang, 2012), which we refer to as the Kinect or Kinect sensor. The Kinect contains an RGB (Red Green Blue device-dependent color model) camera, an infrared camera and an infrared projector. The infrared camera and projector makes it possible to estimate the depth of each pixel acquired by the RGB camera. Thus, the video stream that comes from the Kinect at 30 frames per seconds includes the standard video from the RGB camera and for each pixel we get a *D*-value, which is an estimate of the distance from the Kinect to the point seen by the camera. This type of data is referred to as RGB-D video.

One of the main innovations of the Kinect is the skeletal tracking algorithm (Shotton, 2013). The skeletal tracking algorithm is based on the Random Forest prediction algorithm (Breiman, 2001). When using the Kinect sensor, the data provided by the algorithm consists of a multivariate time-series of measurements, where positional measurements for hands, wrists, elbows, shoulders, neck and head are provided as 3D world coordinates. We only used coordinates from the upper body in this study.

We implemented a game-like environment in order to record series of movements of the participants. We refer to this environment as the *Motorgame*. The design requirements for this game consisted of the following:

1. The participant should perform the same or similar movements repeatedly as fast and precisely as they can.
2. The game should contain tasks that challenge the hand-eye coordination of the participants, and difficulty of the game tasks should gradually increase during the game.

Description of the Motorgame

The *Motorgame* was made so that the participant observed the upper body of a stickman figure on a television screen that mirrored the movements of the participant (See Figure 1). First, the participant was asked to place themselves at a distance between 2 and 3 meters for optimal recording conditions. The participant was then asked to stretch out their arms. This was done for calibrating the arm length of the stickman figure. After this procedure, a message appeared on the screen stating that the participant should try to finish the upcoming tasks as fast and precisely as possible. Then the following tasks were split up into three levels (See Figure 2, Figure 3 and Table 1). Before each level a welcome screen appeared, indicating that the participant had to perform a different task at the next level. The participant needed to perform similar movements of the hands repeatedly, but the design of random appearance of the button made it hard to *learn* this task. A score was displayed on the top of the screen where the participant was awarded a higher score if they finished the task

fast. In order to avoid interruptions during the recording, a training session was performed before the recording in order for the participant to get familiar with the game.

Description of Data

When a participant played the entire *Motorgame*, the following data was recorded in a comma separated text file. Each entry in the file corresponds to one frame from the RGB-D video, where the frames were recorded 30 times per second. The RGB-D video data was stored in a separate file. The *screen coordinates* were the 2D coordinates of the joints as seen on the screen when playing the game.

Assessments

All participants were assessed with standard neurological examination. Only patients with PD were assessed with the Movement Disorder Society Unified Parkinson's Disease Rating Scale (MDS-UPDRS) (Goetz CG, 2008). All participants were assessed by the UKU Side Effect Rating Scale (Lingjaerde, 1987) (UKU; assessing antipsychotic-induced side effects, including hypokinesia) and the Simpson-Angus Scale (Simpson, 1970) (SAS; assessing antipsychotic-induced parkinsonism). In addition, two subtests from the Brief Assessment of Cognition in Schizophrenia (BACS) (Keefe RS, 2004) assessing attention and information processing speed (Symbol Coding Task) and motor speed (Token Motor Task) were conducted. All participants were assessed by the *Motorgame*. Finally, participants filled out a questionnaire developed by the authors for the present study (5-point Likert scale about comprehensibility of the instructions, personal evaluation, preferred choice of test) assessing their opinion about the *Motorgame* compared to the clinical assessment with the rating scales and the two BACS tests.

Ethical considerations

The study was approved by The Committees on Health Research Ethics for the Capital Region of Denmark and the Danish Health Authority. The Danish Data Protection Agency journal number 2007-58-0015 approved the data collection and data storing.

Data from the Motorgame

Data from the *Motorgame* were multivariate time-series of varying length, hence a single play of the *Motorgame* generated a lot of data. For the reason of testing the value of the measurement with respect to bradykinesia, the only variable we have extracted for this analysis was the time it took to finish each of the tasks in level 1 of the *Motorgame*. For a given participant we obtained 22 variables. These measurements can be seen as 22 repeated measurements for a given participant, but due to the differences in the tasks, e.g. how far the participant had to move their hand, these were inherently different. This was accounted for in the statistical model by assigning a fixed effect to each task.

As data were did not fit a Gaussian distribution, natural logarithm transformation was used. Due to the log transformation, parameters can approximately be seen as percentwise increase/decrease in time it took to finish the tasks. For example, a parameter estimate of 0.06 for male participants indicated that it took 6% longer time to finish the tasks compared to females. This approximation is good for low values of parameters due to the Taylor-expansion of the natural logarithm having the coefficient one for the linear term.

Statistical Methods

To analyze the data we used a linear mixed effect model (McLean, 1991) implemented in the package lmer (Kuznetsova, 2015) for the R-programming language (Team, 2000) that provided *p*-values for the fixed effects in the model. The first two variables were the response and the clinical variable, the rest were variables used for correcting against certain demographic or clinical variables.

- **Response** is the natural logarithm of time in seconds it took to finish a single task in level 1 of the *Motorgame*. For a single participant we had 22 observations resulting in 22*63=1386 observations from the whole cohort.

- **Clinical Score** is the only variable that we change between models. This variable was either one of the MDS-UPDRS variables, SAS variables or some other variables related to status of disease and physical activity.
- **Task,** a factor variable with 22 levels, corresponding to the 22 tasks in the first level of the *Motorgame.*
- **Symbol Coding Tasks,** used as a proxy for attention and information processing speed.

The model also included a general mean term and the error was assumed to be independent and identically distributed from a normal distribution.

The terms in the model were, y the response, and on the right hand-side we had in the following order: μ as a general mean, T_j mean for each of the 22 tasks, C parameter for the clinical score, where x_{ijC} was the value for that measurements on participant i in task j. H was the height, W the weight and S the result of the Symbol Coding Task. The last two terms were ε_i, the random effect for participants and then the general error term.

Results

Demographic and disease specific characteristics are shown in Table 2). Thirty patients with PD and 33 healthy controls were assessed. All patients and healthy controls completed both sessions of the *Motorgame*. There were significantly more males in the patient group than in the healthy control group (60.0% vs. 30.3%, p=0.018). All 30 patients (100%) were right handed. All, but one, (97%) in the healthy control group were right handed.

Results of the clinical assessments showed significant differences between the two study groups (Table 3). A significantly higher proportion of the healthy controls managed to complete the Token Motor Task without modifications (pushing or tipping the tokens) than in the patient group (p=0.001). Likewise, the patient group had significantly lower mean scores on the Token Motor Task (indicating reduced motor speed in the fingers) versus healthy controls (25.5$\pm$18.1 vs.

39.9±13.4, p=0.003). Furthermore, patients had significantly higher mean SAS total scores (indicator of rigidity; $p<0.001$) and hypokinesia UKU scores (indicator of bradykinesia; $p<0.001$) compared to the group of healthy controls. No significant difference in mean scores (SD) in the Symbol Coding Task (information processing speed) was found between the two study groups (37.53 (13.62) in the patient group versus 41.42 (10.15) in the healthy control group; p=0.201).

Mixed model analysis - MDS-UPDRS

The results from the mixed model analysis of the effect of motor MDS-UPDRS items on the time of motor performance in the *Motorgame* are seen in Table 4. Since MDS-UPDRS was only assessed in patients with PD all controls have been assigned value zero for the measured variables in this analysis. From the results, it can be seen that all items in the MDS-UPDRS corresponding to bradykinesia and rigidity in the upper body, except from finger tapping left hand and hand movements left, had a significant ($p<0.05$) effect on the time of motor performance in the *Motorgame*. The strongest effects were from *finger tapping right hand*, *hand movements right hand*, and *rotation of right hand* (items of bradykinesia; $p<0.001$). A negative moderating effect of Symbol Coding Task scores was found in all MDS-UPDRS items ($p<0.001$), ie. a higher information processing score corresponded to a shortened time of motor performance. No moderating effects of age or weight were found. Significant ($p<0.05$) moderating effects of male gender (negative ie. shorter performance time) and height (positive ie. longer performance time) were found in the MDS-UPDRS items of finger tapping right hand, hand movements right hand, and rotation of right hand, but not in the remaining MDS-UPDRS items

Mixed model analysis - Clinical assessments

As seen in Table 5, SAS items corresponding to bradykinesia and rigidity, as well as the hypokinesia UKU score, had a significant ($p<0.05$) positive effect on the time of motor performance

in the *Motorgame*. Furthermore, a negative, moderating effect of Symbol Coding Task scores was found in all items ($p<0.001$). No significant effects of Token Motor Task (motor speed), duration of PD and hours of weekly physical activity were found. No moderating effects of age, height or weight were found.

Acceptance of the Motorgame

The application was well accepted and preferred above the clinical rating scales and the BACS subtests by 76% in the healthy control group and 53% in the patient group.

Discussion

Initially developed as an entertainment device, e.g., used for dancing games, the Kinect sensor is now in widespread research use, including neuro-rehabilitation (Chang, 2011), assessment of post-stroke movement impairment (Olesh, 2014), and classification of movements during active video gaming (Rosenberg, 2016).

In our study of 30 patients with PD and 33 healthy controls, we found a highly significant association between prolonged time of motor performance in the *Motorgame* and higher scores of MDS-UPDRS items related to right hand movements (bradykinesia). This side difference might be explained by the fact that, even though the majority of the patients (87%) in our study were bilateral affected by motor symptoms, everyone in the patient group were right handed and displayed general higher right sided symptom severity scores (data not shown). This is in line with previous studies showing that the side of PD symptoms dominance correlates with handedness (Shi J, 2014) (van der Hoorn A, 2012).

Bradykinesia has been shown to correlate strongly with a broad cluster of PD motor symptoms (Nieuwboer A, 1998). Furthermore, 18 F-DOPA PET scans in PD patients with predominantly

hypokinesia and rigidity motor symptoms correlate significantly with dopaminergic depletion in striatum (Pikstra AR, 2016). In our study, we found a statistical significant association between prolonged motor performance and upper-body rigidity (MDS-UPDRS).

The same associations between time of motor performance and bradykinesia and rigidity were found in relation to the clinical rating scales of DIP: ie. prolonged time of motor performance in the *Motorgame* was related to higher rigidity SAS scores and UKU hypokinesia. Surprisingly, we did not find a significant association between time of motor performance in the *Motorgame* (assessing gross motor skills) and motor speed (Token Motor Task; assessing fine motor skills). A possible explanation might be that the *Motorgame* is a more complex motor test demanding a fairly high level of eye-hand coordination and cognitive skills. This was confirmed by the finding of a significant moderating effect of the Symbol Coding Task on the time of performance of the *Motorgame* (ie. shortened time of performance was associated with higher/better scores of information processing).

Furthermore, we did not find an association between time of motor performance in the *Motorgame* the duration of PD (patients). An explanation could be that the included group of patients in this study were all well-medicated, which was additionally reflected by their median (IQR) Hoehn and Yahr score of 2 (2-2) and mean (±SD) duration of illness of 45.7 months ± 34.0 months.

We found that males had a significantly shorter time of performance in the majority of the estimates. In studies of healthy individuals, it has been shown that male gender influenced the speed of motor performance (Jiménez-Jiménez FJ1, 2011). Whether this difference is further moderated by the PD cannot be analyzed in our study due to the small sample size.

In line with a study of adaptive training/rehabilitation in patients with PD (Summa S, 2015), we found a high level of participant acceptance of the *Motorgame* showing that the *Motorgame* was the preferred choice of test by the majority of the healthy control group (76%) as well as in the patient group (53%).

In overall, our results are in line with previous studies of the Kinect sensor used as a supplementary assessment for patients with PD. Galna and colleges studied the accuracy of the Kinect in 9 PD patients and 10 healthy controls comparing it to the Vicon three-dimensional motion analysis system (Galna, 2014). They showed high accuracy of the Kinect sensor when measuring time and gross spatial characteristic movements relevant to PD and highly appropriate for distinguishing non-PD subjects from PD patients treated with deep brain stimulation. Likewise, the Kinect sensor has shown high validity regarding gait parameters when validated against a multiple-camera 3D motion capture system (Arango Paredes JD, 2015).

Strengths and limitations

The study has several strengths. Firstly, the instrument is low-cost, easy accessible, portable and easy to administer and does not require expert clinical knowledge to use. Secondly, applicability to the clinical setting was tested and proven on several levels. In terms of practicality, the Kinect-based instrument was easy to set up in hospitals, in private houses, in tennis clubs and senior centers. The test bears potential to be carried out even in intensive care wards and in small examination rooms. Thirdly, the study showed that all PD participants and healthy controls completed the *Motorgame*. Previous studies have shown that videogames for patients with PD should not be made too difficult, in terms of their pace or cognitive complexity (dos Santos Mendes FA1, 2012). Fourthly, by including healthy controls, we were able to match data according to age, but unfortunately not to gender, two variables relevant to motor performance (Jiménez-Jiménez FJ, 2011).

The study has some limitations. Firstly, our *Motorgame* only covers upper extremities and neck. Secondly, the version of the Kinect device used in the present study does not have the accuracy to detect tremors, at least not based on the motion trajectories computed by the internal Kinect algorithms. Potentially, tremor related measurements could be extracted from the raw Kinect depth-

image information. However, due to the small the sample size in this study experimental data exploration of this kind was not possible. For this reason, we also excluded the tremor related clinical measurements from the analysis. Newer devices with higher accuracy is currently being developed which potentially might enable tremor detection. Thirdly, the sample size in this study is small, thereby enhancing the risk of producing type II errors. Further, we have limited ourselves to do the analysis on a simple meta-variable (the playing time), since the system is able to gather very large amounts of data from the movement patterns of the tested participants. The reason is the potential problem of overfitting the sparse set of available movement related features such as speed, acceleration and deceleration and even the use of more automated feature extraction techniques. These aspects should be taken into considerations in future studies.

In conclusion, we have presented an accessible and easy to use system, i.e. the *Motorgame* with data in accordance with currently used clinical motor scores. The *Motorgame* offers a feasible and accessible objective complement to the traditional observer-based rating scales of motor disturbance. However, further development is needed to improve the tracking of tremor and motor symptoms in lower extremities. The concept of using a contact-less and gamified measurements device was well accepted by the users and the present data suggest the relevance of using portable and accessible systems like this on a much larger scaler and in different patient groups.

Acknowledgements including sources of support

Bispebjerg University Hospital, Department of Neurology, Bispebjerg Movement Disorders Biobank, Copenhagen, Denmark and private practicing neurologists Holger Frank Jespersen, Gitte Ørsnes and Elizabeth Maria Holm Nielsen for their help recruiting patients with PD to the study. MD Sabrina Krøigaard has contributed assisting the assessments of the healthy control group. Academic research assistant Nina Ramskov Siegismund has contributed to project coordination and logistics. All patients and healthy controls that participated in the study.

The study has received grants from The Capital Region of Denmark, Research Fund for Health Promotion and The Capitol Region of Denmark, Mental Health Services Research Fund. Gudmundur Einarsson is partly funded by a research grant from the Lundbeck Foundation.

Conflict of Interest

The authors Ditte Rudå (DRU), Gudmundur Einarsson(GEI), Anne Sofie Schott Andersen (ASA), Jannik Boll Nielsen (JBN), Rasmus R. Paulsen (RRP) and Anne Katrine Pagsberg (AKP) declare that they have no competing interests. Kristian Winge (KWI) has been a consultant and/or advisor to Abbvie and has scientific collaboration with Lundbeck. Anders Fink Jensen (AFJ) has received an unrestricted research grand from Novo Nordisk. Cristoph Corell (CCO) has been a consultant and/or advisor to Bristol-Myers Squibb, Eli Lilly, Genentech, Gerson Lehrman Group, IntraCellular Therapies, Janssen/J&J, Lundbeck, MedAvante, Medscape, Otsuka, Pfizer, ProPhase, Roche, Sunovion, Supernus and has received honoraria from BMS, Janssen/J&J, Novo Nordisk A/S, Otsuka, Takeda, Bristol-Myers Squibb, Janssen/J&J, Lundbeck, Medscape, Otsuka, ProPhase and Pfizer.

References

Arango Paredes JD, M.B A- . (2015). A reliability assessmentsoftware using Kinectto complement the clinical evaluation of Parkinson's disease. *Conf Proc IEEE Eng Med BiolSoc*, s. 6860-6863.

Bandini, A a . (2016). Markerless analysis ofarticulatory movements in patients with parkinson's disease. *Journalof Voice, 30*(6), 766-e1.

Barnes, T.R . (1989). A rating scale for drug-induced akathisia. *The British Journalof Psychiatry, 154*(5), 672-676.

Barrett MJ, W.S . (Oct 2011). Handedness and motorsymptom asymmetry in Parkinson's disease. *J Neurol Neurosurg Psychiatry*, s. 82(10):1122-4. doi : 10.1136/jnnp.2010.209783.

Benamer HT, O W Sep 2003). Prospective study of presynaptic dopaminergic imaging in patients with mild parkinsonism and tremordisorders: part 1. Baseline and 3-month observations. *Mov Disord*, s. 18(9) 977-84.

Bonnechere, B a . (2014). Validity and reliability of the Kinect within functional assessmentactivities: comparison with standard stereophotogrammetry. *Gait & posture, 39*(1), 593-598.

Breiman, L . (2001). Random forests. *Machine Learning, 45*(1), 5-32.

Chang Y.J . a-F Đ . (2011). A Kinect-based system for physical rehabilitation: A pilot study for young adults with motor disabilities. *Research in developmentaldisabilities, 32*(6), 2566-2570.

Dean, C.E . (2004). Clinical rating scales and instruments: how do they compare in assessing abnormal involuntary movements? *Journalof clinicalpsychopharmacology, 24*(3), 298-304.

dos Santos Mendes FA1, P.J . (2012). Motor learning, retention and transfer after virtual-reality-based training in Parkinson's disease--effect of motor and cognitive demands of games: a longitudinal, controlled clinical study. *Physiotherapy*, s. 98:217-223. 10.1016.

Galna, B.a . (2014). Accuracy of the Microsoft Kinect sensor for measuring movement in people with Parkinson's disease. *Gait & posture, 39*(4), 1062-1068.

Goetz CG, T B M- 200(8). MovementDisorder Society- sponsored revision of the Unified Parkinson's Disease Rating Scale (MDS- UPDRS): scale presentation and clinimetric testing results. *Movement disorder official journal of the Movement Disorder Society*, s. 23(15.

Goetz, C G M. 200(8). MovementDisorder Society- sponsored revision of the Unified Parkinson's Disease Rating Scale (MDS- UPDRS): Scale presentation and clinimetric testing results. *Movement disorders*, *23*(15), 2129- 2170.

Guttman, M.a . (1997). [11C] RTI- 32 PET studies of the dopamine transporter in early dopa- naive Parkinson's disease: Implications for the symptomatic threshold. *Neurology*, *48*(6), 1578- 1583.

Guy, W. 1976). Abnormal involuntary movement scale (AIMS). *ECDEU assessment manual for psychopharmacology*, *938*, 534- 537.

Hausner, R S. (1983). Neuroleptic- induced parkinsonism and Parkinson's disease: differential diagnosis and treatment. *The Journal of clinical psychiatry*.

Hoehn MM, Y M. 196(7). Parkinsonism: onset, progression and mortality. . *Neurology*, s. 17(5).

Jiménez- Jiménez F, J C M N- d- l .- .- F- N S R- M- 15 M ar. 2011). Influence of age and gender in motor performance in healthy subjects. *J Neurol Sci*, s. 302(1- 2):72- 80. doi : 10. 1016/ j. jns. 2010. 11. 021. Epub 2010 Dec 22.

Jiménez- Jiménez FJ, 1 C M N- d- l .- .- F- N S R- M- 15 M ar. 2011). Influence of age and gender in motor performance in healthy subjects. *J Neurol Sci*, s. 302(1- 2):72- 80. doi : 10. 1016/ j. jns. 2010. 11. 021.

Keefe RS, G T . (2004). The Brief Assessment of Cognition in Schizophrenia: reliability, sensitivity, and comparison with a standard neurocognitive battery. *Schizophr Res*, s. 68(2- 3):283- 297.

Kuipers, J. B. (1999). Quaternions and rotation sequences (*66*).

Kuznetsova, A a . (2015). Package 'lmerTest'. *R package version*, *2*(0).

Lingjaerde, O. a . (1987). The UKU side effect rating scale: A new comprehensive rating scale for psychotropic drugs and a cross- sectional study of side effects in neuroleptic- treated patients. *Acta Psychiatrica Scandinavica*, *76*(S334), 1- 100.

Lohr, J. B. (1992). Quantitative instrumental measurement of tardive dyskinesia: A review. *Neuropsychopharmacology*.

López- Sendón J, M. M Jul (. 2013). Drug- induced parkinsonism. *Expert Opin Drug Saf*, s. 12(4):487- 96. doi : 10. 1517/ 14740338. 2013. 787065. Epub 2013 Ma31.

McLean, R.A . (1991). A unified approach to mixed linear models. *The American Statistician*, *45*(1), 54- 64.

Moeslund, T.B . (2001). A survey of computer vision- based human motion capture. *Computer vision and image understanding*, *81*(3), 231- 268.

Moustafa, A.A . (2016). Interrelations between cognitive dysfunction and motor symptoms of Parkinson' s disease: behavioral and neural studies. *Reviews in the Neurosciences*, *27*(5), 535- 548.

Nieuwboer A, D W Apr(1998). A frequency and correlation analysis of motor deficits in Parkinson patients. *Disabil Rehabil*, s. 20(4):142- 50.

O. A ., & FSGNKBEMSK, E b . (2011). Motion- Based Games for Parkinson's Disease Patients. *Vancouver Springer, In Proceedings of the tenth International Conference of Entertainment Computing*, s. 47- 58. .

Olesh, E.V . (2014). Automated assessment of upper extremity movement impairment due to stroke. . *PloS one*, *9*(8), e104487.

Palacios- Navarro, G. a. M L- . (2015). A Kinect- based system for lower limb rehabilitation in Parkinson' s disease patients: a pilot study. *Journal of medical systems*, *39*(9), 103.

Paredes, J. D- A . (2015). A reliability assessment software using Kinect to complement the clinical evaluation of Parkinson's disease. *Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE* (s. 6860- 6863). IEEE.

Pikstra AR, v . d. (12 Jan 2016). Relation of 18- F- Dopa PET with hypokinesia- rigidity, tremor and freezing in Parkinson's disease. *Neuroimage Clin*, s. 11: 68- 72. doi : 10. 1016/ j. nicl. 2016. 01. 010. eCollection 2016.

Poewe, W. a E- . (2017). Parkinson disease. *Nature Reviews Disease Primers*, *3*, 17013.

Postuma, R. B . (2015). MDS clinical diagnostic criteria for Parkinson's disease. *Movement Disorders*, *30*(12), 1591- 1601.

Prescott, I a (2014). Lack of depotentiation at basal ganglia output neurons in PD patients with levodopa- induced dyskinesia. *Neurobiology of disease*, *71*, 24- 33.

Rosenberg M. a . (2016). Development of a Kinect Software Tool to Classify Movements during Active Video Gaming. *PloS one*, *11*(7), e0159356.

Shi J, L J . (Feb 2014). Handedness and dominant side of symptoms in Parkinson's disease. *Med Clin (Barc)*, s. 20; 142(4):141- 4. doi : 10. 1016/ j. medcli. 2012. 11. 028.

Shotton, J . a. (2013). Real- time human pose recognition in parts from single depth images. *Communications of the ACM*, *56*(1), 116- 124.

Simpson, G. a . (1970). A rating scale for extrapyramidal side effects. *Acta Psychiatrica Scandinavica. Supplementum*, *45*(212), 11–19.

Summa, S., & B. A . (2015). Adaptive training with full-body movements to reduce bradykinesia in persons with Parkinson's disease : a pilot study. *J Neuroeng Rehabil*, *5*, 12–16.

Svendsen, M. B . (2014). Using motion capture to assess colonoscopy experience level. *World journal of gastrointestinal endoscopy*, *6*(5), 193.

Team, R. C . (2000). R language definition.

Tomlinson, C. L., & S. R . (2010). Systematic review of levodopa dose equivalency reporting in Parkinson's disease. *Mov Disord*, *25*(15), 2649–2653.

van der Hoorn, A., & B. H . (Feb 2012). Handedness correlates with the dominant Parkinson side : a systematic review and meta-analysis. *Mov Disord*, *27*(2), 206–10. doi : 10.1002/mds.24007. Epub 2011 Oct 12.

Wolff, A. L., & a . (1999). Motor deficits and schizophrenia : the evidence from neuroleptic-naive patients and populations at risk. *Journal of Psychiatry and Neuroscience : JPN*, *24*(4), 304.

World Health Organization, G. (1993). The ICD-10 classification of mental and behavioural disorders : diagnostic criteria for research.

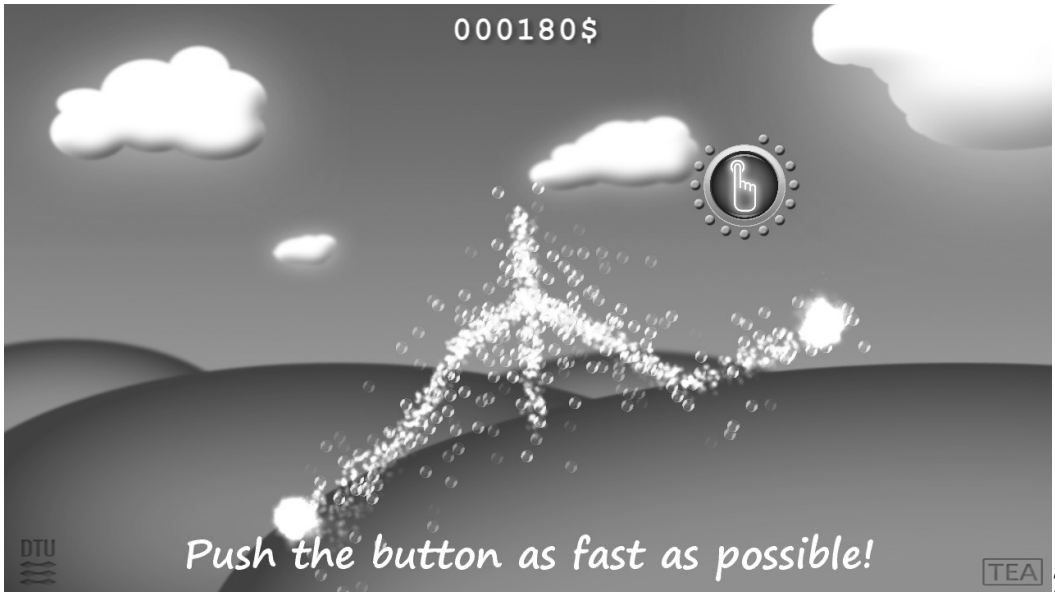
Zhang, Z. (February 2012). Microsoft Kinect Sensor and Its Effect. *IEEE Multimed*, *19*(2), 4–10.

Figures and Tables

***Figure** 1 A participant playing the Motorgame.*

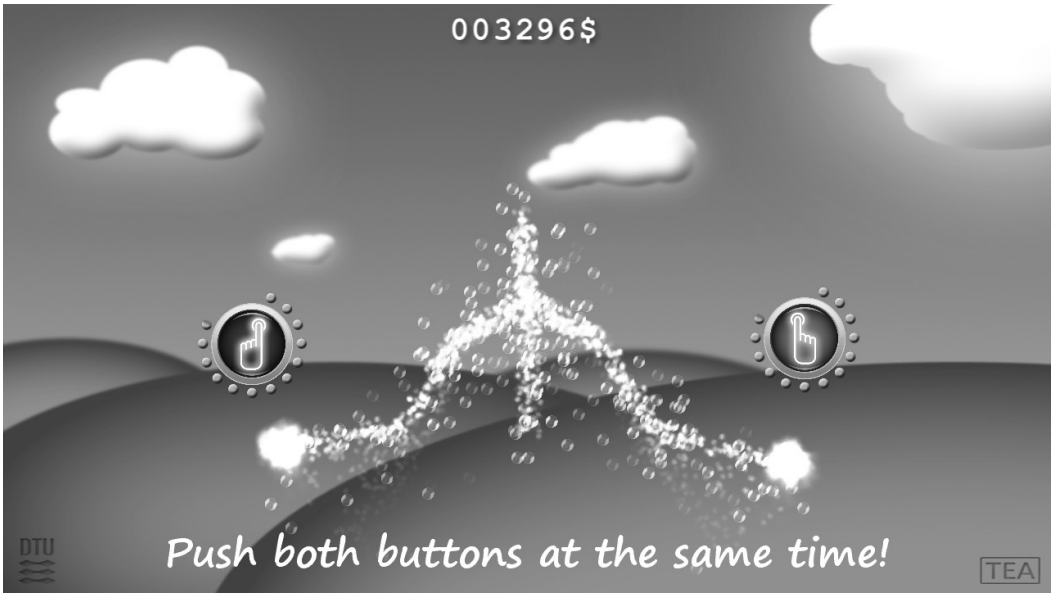
A Kinect sensor is placed on the top of the television and tracks the participant's movements. The participant's pose is mirrored as a stickman figure on the screen.

***Figure** 2 Level 1 in the Motorgame from the participant's perspective.*



The participant is moving its right hand upwards to reach the blue button visible on screen.

Figure 3 *Level 2 in the Motorgame from the participant perspective.*



Now the participant has to touch two buttons simultaneously.

Table 1. *Elements of the different levels in the Motorgame*

Level	**No of tasks (right/left handside)**	**Completion of task**	**No of buttons, the participant must push at the same time**	**Button moving during test between single tasks**
1	22 (11/11)	Must hold the hand still on the button for at least 1 second	1	y
2	21 (21/21)	Must hold the hand still on the button for at least 1 second	2	y
3	12 (6/6)	Must follow the bottom as it moves to the top or the bottom of the screen	1	y

Table 2. *Demographic and disease specific characteristics*

	Patients with Parkinson's Disease (n=30)	Healthy controls (n=33)	P value
Males, n (%)	18 (60.0)	10 (30.3)	**0.018**[a]
Age in years, mean (SD)	70.1 (6.7)	69.7 (6.1)	0.787[b]
Family history of Parkinson's Disease, n (%)	8 (26.7)	2 (6.1)	**0.025**[a]
Hoehn and Yahr score, median (IQR)	2 (2-2)	-	-
Mean duration of Parkinson's Disease in months, mean (SD)	45.7 (34.0)	-	-
L-dopa equivalent dose (Tomlinson CL, 2010) mg, mean (SD)	868.4 (1902.2)	-	-
Physical activity hours/week, mean (SD)	6.6 (5.5)	7.7 (4.6)	0.388[b]
Computer games hours/week, mean (SD)	0.17 (0.91)	0 (0.0)	0.306[b]

[a] chi² test; [b] unpaired t-test; "-" not applicable; SD Standard Deviation; IQR Interquartile Range

Table 3. *Clinical assessments*

	Patients with Parkinson's Disease (n=30)	Healthy controls (n=33)	P value
BACS Token Motor Task (completers), n (%)	17 (56.67)	31 (93.94)	**0.001**[a]
BACS Token Motor Task score, mean (SD)	25.53 (18.13)	39.89 (13.44)	**0.003**[b]
BACS Symbol Coding Task, mean (SD)	37.53 (13.62)	41.42 (10.15)	0.201[b]
SAS total score, mean (SD)	0.87 (0.70)	0.13 (0.19)	**<0.001**[b]
Hypokinesia (item 2.3 UKU), mean (SD)	1.17 (0.65)	0.09 (0.29)	**<0.001**[b]

[a] chi² test; [b] ANOVA test; SD Standard Deviation

Table 4 *The effect of motor items in the MDS-UPDS on the time of motor performance in the Motorgame.*

	MDS-UPDRS item	Gender	Age	Height	Weight
3.3a Rigidity Neck	**0.0401*** (0.0191)	-0.105 (0.0524)	-0.000552 (0.00307)	0.00517 (0.00349)	-0.00132 (0.00174)
3.3b Rigidity Right Arm	**0.0447*** (0.0185)	-0.0994 (0.051)	0.000343 (0.00306)	0.00542 (0.00343)	-0.00136 (0.00171)
3.3c Rigidity Left Arm	0.0372 (0.019)	-0.0916 (0.0516)	0.000325 (0.00313)	0.00525 (0.00351)	-0.00133 (0.00175)
3.3d Rigidity Right Leg	**0.0351*** (0.0156)	-0.0946 (0.0511)	-0.000163 (0.00306)	0.00543 (0.00346)	-0.00174 (0.00171)
3.3e Rigidity Left Leg	**0.0364*** (0.0165)	-0.0876 (0.0509)	-0.000624 (0.00306)	0.00515 (0.00348)	-0.00159 (0.00171)
3.4a Finger Tapping Right Hand	**0.0528**** (0.0149)	**-0.135**** (0.0502)	0.000949 (0.00291)	**0.00799*** (0.00328)	-0.00203 (0.0016)
3.4b Finger Tapping Left Hand	0.0308 (0.0161)	-0.103 (0.0527)	-0.000768 (0.00309)	0.00581 (0.00349)	-0.00114 (0.00177)
3.5a Hand Movements Right	**0.0583**** (0.0162)	**-0.116*** (0.0486)	0.00106 (0.0029)	**0.00757*** (0.00325)	-0.00163 (0.0016)
3.5b Hand Movements Left	0.024 (0.0167)	-0.0872 (0.0523)	-0.000534 (0.00314)	0.00546 (0.00357)	-0.00123 (0.00181)
3.6a Pronation-Supination Movements of Hands Right	**0.06**** (0.0163)	**-0.12*** (0.0486)	0.00066 (0.00287)	**0.00804*** (0.00326)	-0.002 (0.00159)
3.6b Pronation-Supination Movements of Hands Left	**0.0302*** (0.0148)	-0.0841 (0.0511)	-0.000763 (0.00308)	0.00535 (0.00349)	-0.00129 (0.00174)
3.12 Postural Stability	0.0201 (0.0229)	-0.0839 (0.053)	-0.000243 (0.00321)	0.00642 (0.00359)	-0.00183 (0.00178)
3.13 Posture	0.0438 (0.0218)	-0.0969 (0.0519)	-0.00101 (0.00308)	0.00592 (0.00348)	-0.00145 (0.00174)
3.14 Global Spontaneity of Movement (Body Bradykinesia)	**0.0419*** (0.0183)	**-0.111*** (0.0525)	-0.000369 (0.00305)	0.00515 (0.00347)	-0.000732 (0.00178)

Since MDS-UPDRS was only assessed in patients with Parkinson's Disease all controls have been assigned value zero for the measured variables in this analysis. Results are logtransformed (natural logaritme). Each parameter estimate is presented with the standard deviation in parenthesis (SD).
** p<0.05; ** p<0.001.*

Table 5 *The effect of clinical assessment scores (Simpson Angus Scale, UKU and Token Motor task) and disease related characteristics on the time of motor performance in the Motorgame.*

	Clinical assessment Score	Gender	Age	Height	Weight
SAS-1 Gait	**0.0613* (0.025)**	**-0.1028* (0.051)**	-0.00004 (0.003)	0.0050 (0.003)	-0.0008 (0.002)
SAS-2 Arm drop	**0.0717* (0.024)**	-0.0938 (0.049)	-0.0009 (0.003)	0.0040 (0.003)	-0.0009 (0.002)
SAS-3 Shoulder shaking	**0.0766* (0.023)**	**-0.1016* (0.049)**	-0.0001 (0.003)	0.0041 (0.0033)	-0.0008 (0.002)
SAS-4 Elbow rigidity	**0.0441* (0.017)**	**-0.1044* (0.051)**	0.0002 (0.003)	0.0053 (0.003)	-0.0012 (0.002)
SAS-5 Wrist rigidity	**0.0742* (0.026)**	**-0.1020* (0.050)**	-0.0004 (0.003)	0.0052 (0.003)	-0.0014 (0.002)
SAS-6 Leg pendulousness	**0.0369* (0.013)**	-0.0955 (0.050)	-0.0006 (0.003)	0.0051 (0.003)	-0.0013 (0.002)
SAS-7 Head dropping	**0.0431* (0.019)**	**-0.1071* (0.052)**	-0.0007 (0.003)	0.0051 (0.003)	-0.0012 (0.002)
SAS-8 Glabella tap	**0.0323* (0.011)**	-0.0813 (0.051)	0.0005 (0.003)	0.0046 (0.004)	-0.0010 (0.002)
SAS-9 Tremor	0.0162 (0.018)	-0.0824 (0.053)	-0.0010 (0.003)	0.0059 (0.004)	-0.0019 (0.002)
SAS-10 Salivation	0.0987 (0.056)	-0.0929 (0.052)	-0.0002 (0.003)	0.0061 (0.004)	-0.0014 (0.002)
2.3 Hypokinesia (UKU)	**0.0665* (0.022)**	**-0.1128* (0.050)**	-0.0008 (0.003)	0.0047 (0.003)	-0.0002 (0.002)
Token Motor Task	-0.0012 (0.001)	-0.0852 (0.054)	-0.0005 (0.003)	0.0061 (0.004)	-0.0017 (0.002)
Duration of Parkinson	0.0009 (0.000)	-0.1010 (0.053)	-0.0000 (0.003)	0.0072 (0.003)	-0.0018 (0.002)
Physical activity	0.0001 (0.003)	-0.0791 (0.054)	-0.0006 (0.003)	0.0062 (0.004)	-0.0019 (0.002)

*Results are logtransformed (natural logaritme). Each parameter estimate is presented with the standard deviation in parenthesis. Duration of Parkinson's Disease (PD) is in months. Physical activity is average number of hours per week. * p<0.05; ** p<0.001.*

APPENDIX H

Exploring movements in adolescents with psychosis and healthy controls using the Microsoft Kinect sensor – a new tool for assessing drug-induced parkinsonism?

Exploring movements in adolescents with psychosis and healthy controls using the Microsoft Kinect sensor – a new tool for assessing drug-induced parkinsonism?

Ditte Rudå[1], MD, ditte.rudaa@regionh.dk

Gudmundur Einarsson[2], M.Sc, guei@dtu.dk

Jannik Boll Nielsen[2], M.Sc, Ph.D, jbol@dtu.dk

Christoph Correll[3], MD, Dr.Med, CCorrell@NSHS.edu

Karsten Gjessing Jensen[1], MD, Ph.D, karsten.gjessing.jensen@regionh.dk

Dea Gowers Klauber[1], Cand.Scient.Soc, dea.klauber@regionh.dk

Jens Richardt Jeppesen[4], Cand.Psych, Ph.D,

jens.richardt.moellegaard.jepsen@regionh.dk

Birgitte Fagerlund[4], Cand.Psych, Ph.D, birgitte.fagerlund@regionh.dk

Kristian Winge[5], MD, Ph.D, k.winge@dadlnet.dk

Line K. H. Clemmensen[2], M.Sc, Ph.D, lkhc@dtu.dk

Rasmus R. Paulsen[2], M.Sc, Ph.D, rapa@dtu.dk

Anne Katrine Pagsberg[1], MD, Ph.D, anne.katrine.pagsberg@regionh.dk

Anders Fink-Jensen[6], MD, Ph.D, anders.fink-jensen@regionh.dk

[1] Centre for Child and Adolescent Mental Health, Mental Health Services, Capital Region of Denmark & Faculty of Health Science, University of Copenhagen, Denmark.

[2] Section for Image Analysis and Computer Graphics, DTU Compute, Technical University of Denmark.

[3] Hofstra North Shore Long Island Jewish School of Medicine and The Zucker Hillside Hospital, New York, 75-59 263rd Street, Glen Oaks, 11004 New York, U.S.A

[4]Center for Neuropsychiatric Schizophrenia Research, Psychiatric Center Glostrup, Denmark.

[5] Bispebjerg University Hospital, Department of Neurology, Bispebjerg Movement Disorders Biobank, Copenhagen, Denmark.

[6] Psychiatric Centre Copenhagen, University Hospital Copenhagen, Denmark and Laboratory of Neuropsychiatry, Department of Neuroscience and Pharmacology, University of Copenhagen.

Corresponding author:

Ditte Rudå

Centre for Child and Adolescent Mental Health, Mental Health Services, Capital Region of Denmark

Nordre Ringvej 69, Kirsebærhuset

DK- 2600 Glostrup, Denmark

Tel: +45 3864 1186/+45 5052 3858

Email: Ditte.Rudaa@Regionh.dk

Name of the department and institution where the work was done, and other institutional affiliation(s):

Department of Psychiatry, Child and Adolescent Psychiatry and Neurology at Copenhagen University Hospitals and the Technical University of Denmark (DTU Compute).

Support received from any grant, funding source, or commercial interest:

The study has received grants from The Capital Region of Denmark, Research Fund for Health Promotion and The Capitol Region of Denmark, Mental Health Services Research Fund. Gudmundur Einarsson is partly funded by a research grant from the Lundbeck Foundation.

Abstract

Background: Children and adolescents are more vulnerable to antipsychotic induced movement disorders compared to adults. In research, the assessment is often limited to the use of observer based rating scales, showing different degrees of inter-observer variability when tested.

Objectives: To validate the Motorgame data against the items of rigidity measured by the Simpson Angus Scale.

Methods: A computer application using the Microsoft Kinect sensor (Motorgame), especially targeting motion patterns associated with parkinsonism, was tested in a group of adolescents with psychosis and healthy controls matched on age and gender. All participants were assessed by neurological examination and clinical side effect rating scales: Udvalg for Kliniske Undersøgelser (UKU) Side Effect Rating Scale, Barnes Akathisia Rating Scale (BARS), Simpson-Angus Scale (SAS) and Abnormal Involuntary Movement Scale (AIMS), and tests of information processing (Symbol Coding Task) and motor speed (Token Motor Task) from the BACS (Brief Assessment of Cognition in Schizophrenia) test battery.

Results: 21 adolescents with psychosis and in medical treatment with antipsychotics versus 69 healthy controls were studied. Male proportion was 38.1% in the patient group versus 36.2% in the control group ($p=0.877$). Mean age was 16.02 ± 1.37 years in patients and 16.04 ± 1.52 years in controls ($p=0.960$). A positive significant effect ($p=0.009$) of arm dropping in the SAS (item 2) and a negative significant ($p<0.001$) effect of glabella tap in the SAS (item 8) was found. Furthermore, at consistent and clear retest effect was detected ($p<0.001$). In contrast to our previous study of Motorgame in patients with Parkinson's disease and healthy controls, we did not find any effect of gender or information processing (Symbol Coding Task). No effects of age, height or weight was found. Based on the questionnaire (n=8) the Motorgame was reported to be easy and fun to use and was preferred above clinical rating scales.

Conclusion: A significant association between prolonged time of performance in the Motorgame and bradykinesia/rigidity in the shoulders was found. Future studies in larger scale, including patients with higher severity in clinical scale scores are required.

Keywords: Movement disorders, Extrapyramidal symptoms, Antipsychotics, Psychosis, Schizophrenia, Children, Adolescents, Kinect.

Introduction

Schizophrenia is a severe mental illness, characterized by extensive cognitive, emotional and behavioural impairments. Usually it becomes manifest in early adulthood, but it can also be present in childhood and the adolescent years. Early onset psychosis (EOP, onset before age of 18 years) is a serious variant of schizophrenia that often has worse outcomes compared to adult onset psychosis.[1] Compared to adults, children and adolescents are more prone to develop antipsychotic-induced extrapyramidal symptoms (EPSs) i.e. akathisia, dystonia, parkinsonism and dyskinesia.[2] A meta-analysis[3] (2012) of 41 controlled short term studies, including 4015 children and adolescents, of efficacy and safety of second generation antipsychotics (SGAs) showed that all SGAs, except for quetiapine, significantly increased the risk of EPS compared with placebo: ziprasidone OR, 20.56 (3.53-68.94); olanzapine OR, 6.36 (2.43-13.84); aripiprazole OR, 3.79 (2.17-6.17); risperidone OR, 3.71 (2.18-6.02); and quetiapine OR, 2.54 (0.88-6.07). EPSs are distressing movement disorders with the potential to interfere with patient's adherence to medicine and their quality of life. In a naturalistic study[4], EPSs have been associated with poorer outcome in youth with schizophrenia spectrum disorders. Furthermore, absence of EPS has been associated with improved compliance in adult patients with schizophrenia.[5] Therefore, focus on minimizing adverse effects is important.

Traditional assessment of drug-induced movement disorders

In research, clinical rating scales are commonly used in the assessment of antipsychotic-induced movement disorders. Rating scales offer a structured and standardized method and the most used ones have been solidly validated. But due to the fact that rating scales are observer-based, studies have shown that even trained raters may underestimate the prevalence of motor abnormalities.[6] Many researchers have studied more quantitative methods to assess antipsychotic induced movement disorders in adult patients[7-12], but none have, to our knowledge, been implemented broadly in the daily clinic. Nor have studies of quantitative assessment of antipsychotic-induced movement disorders, to our

knowledge, been conducted in children and adolescents with psychosis/schizophrenia.

Kinect-based quantification of movement patterns

The Microsoft Kinect application (2010) was originally developed for motion recognition in gaming applications. The Microsoft Kinect combines a regular colour camera with a depth sensor and software based on advanced, marker-free pattern recognition.[13] The sensor maps the scene and the software recognizes and continuously monitors joint movements three-dimensionally. Additionally, the Microsoft Kinect has the advantages of being assessable, portable and low-cost. The validity of the Kinect sensor has been shown in a number of studies of motion recognition i.e. neck angle[14], spatiotemporal aspects of gait[15] and postural control.[16] In the clinical setting, Microsoft Kinect has been studied in areas such as neuro-rehabilitation[17], assessment of post-stroke movement impairment[18] and in surgical navigation as an objective method to assess, evaluate and train surgical skills.[19] In children, the Microsoft Kinect sensor has been studied as a tool to classify movements during active video gaming[20] in 43 healthy children, showing excellent reliability between two raters and the Kinect Tool for jumps ($r=0.84$, $p<0.01$), and moderate reliability for sidesteps ($r=0.69$, $p<0.01$). Furthermore, the Kinect software has been studied in the evaluation of upper extremity movement characteristics in 12 healthy, typically-developing adolescents with no injury or impairment of upper extremity function.[21] Significant variability in upper extremity kinematics was found, indicating an increased sensitivity of the scores found by the Kinect motion analysis. Additionally, the researchers reported that the Kinect sensor system was easy to use for both therapist and participants.

Studies of Microsoft Kinect in patients with Parkinson's Disease have also been conducted, mainly focusing on gait assessment.[22-27] However, movement symptoms, including the typical movement items in the upper extremity from the Unified Parkinson's Disease Rating Scale, were studied by Galna and collegues in a study of 9 patients with Parkinson's Disease and 10 healthy controls. They concluded that the

Microsoft Kinect sensor was able to accurately measure gross spatial characteristics and timing of clinically relevant movements.[28]

As published previously (reference) we studied a group of adult patients (n=30) with Parkinson's Disease and a group of healthy controls (n=33) matched on age and gender, developing a classification model, especially targeting motion patterns associated with parkinsonism. We found a significant ($p < 0.05$) association between a prolonged time of performance in the Motorgame and higher Simpson Angus Scale rigidity scores. We also found a significant ($p < 0.05$) association between decreased time of performance in the Motorgame and higher (better) scores in information processing (Symbol Coding Task). Finally, we found a significant gender difference indicating that the female participants on average had to use approximately 10% longer time to complete each task of the Motorgame.

In this pilot study, we want to investigate the Motorgame in a study population of adolescents with psychosis compared to healthy controls matched on age, gender and parental education, and validate the Motorgame data against the items of rigidity measured by the Simpson Angus Scale (measuring antipsychotic drug-induced parkinsonism).

Methods

Design, participants and in- and exclusion criteria

The patient group was in ongoing or about to initiate antipsychotic treatment, aged 12-17 years, in- or out patients, meeting the ICD-10[29] criteria for schizophrenia-spectrum disorder, delusional disorder, or affective-spectrum psychotic disorder. The control group consisted of physically and mentally healthy children and adolescents who completed screening with K-SADS-PL[30] (both participants and parents), standard somatic history and clinical examination.

Recruitment

Patients were recruited partly from the Mental Health Centre for Child and Adolescent Psychiatry in the Capitol Region and partly through to the Tolerability and Efficacy of Antipsychotics (TEA) trial.[31] Healthy controls were all recruited from the TEA trial in which recruitment was done through a random data extraction from the Danish Centralized Civil Register (government-owned registry of all residents in Denmark, located in Copenhagen (http://sundhedsdatastyrelsen.dk/da/forskerservice). The study was submitted to The Committees on Health Research Ethics for the Capital Region of Denmark and the Danish Health Authority, which stated that the study did not require mandatory notification. The Danish Data Protection Agency (DDPA) journal number 2007-58-0015 has approved data collecting and data storing. Approval for exchange of information with the TEA trial has been given from the DDPA journal number 2013-331-0479. A surrogate written informed consent was obtained from the holders of custody, since all participants were below the age of 18 years when engaging in the trials.

Setting and instrument

This pilot study tests the Kinect-based instrument in a hospital in- and outpatient environment. The Kinect application is connected to a computer with the Game-like software developed by the Technical University of Denmark (DTU) installed. The participant is placed at a distance of approximately two meters in front of the computer (Figure 1). A stickman figure that mirrors the participant's movements is visible on the screen. The Motorgame consisted of three levels: The participant had to place his/her hand 'on' (level 1) a stationary spot shown on the screen (11 times with the right hand and then 11 times with the left hand), on two spots simultaneously (level 2) and on a spot and then follow it, when it moved slowly in a semi-circle either upwards or downwards with both hands taking turns (level 3). The Motorgame was run twice: first session as a 'practice run' with instructions given and second session with no

instructions. As a motivational factor points were given and shown on screen: the amount of points would correlate with the accuracy and speed of the movements. The data presented in this paper are exclusively from the second session of level 1.

Procedures and outcomes

Demographic and clinical data

Information on gender, age, height, weight, handedness, former/current mental/physical illness, parental educational level, duration of illness and current status of pharmacological treatment was obtained through an interview. From medical records, information was collected according to clinical verified diagnosis of psychosis, schizophrenia and other psychiatric diagnoses.

Assessments

All participants were assessed with clinical somatic and neurological examinations. Assessment with the clinical side effect rating scales included the Abnormal Involuntary Movement Scale (AIMS)[32] (assessing dyskinesia), the Simpson Angus Scale (SAS)[33] (assessing drug-induced parkinsonism), the Barnes Akathisia Rating Scale (BARS)[34] (assessing akathisia) and the The Udvalg for Kliniske Undersøgelser (UKU) Side Effect Rating Scale[35] (assessing antipsychotic-induced side effects in several domains, including movement disorders). Additional examination of information processing speed (Symbol Coding Task) and motor speed (Token Motor Task) i.e. two subtests from Brief Assessment of Cognition in Schizophrenia (BACS)[36] was done. Finally, two sessions of the Motorgame were tested. Then a subgroup of the participants (the patients recruited directly from the Mental Health Centre for Child and Adolescent Psychiatry in the Capitol Region) filled in a questionnaire on their opinion of the Motorgame compared to the examination from the rating scales and the two subtests from BACS.

All healthy controls were reassessed at follow-up after 12 weeks as well as the antipsychotic-naïve patients recruited directly from the Mental Health Centre for Child

and Adolescent Psychiatry. At 12 weeks of follow up, it was assumed that the majority of the patients would be in stable antipsychotic medical treatment and if adverse effects had occurred, these would most likely have presented within that time frame, since 90% of the adverse effects occurs within the first 12 weeks of treatment[37] (in adult studies). Reassessment of the healthy control group functioned as a control for changes in movement patterns over time in the antipsychotic naïve group. The patients in ongoing antipsychotic treatment recruited from the TEA trial were assessed at follow ups (predefined in the TEA trial protocol ie. week 2, 4, 12 and/or 52 after randomization to aripiprazole or quetiapine-ER).

Investigational plan

An interdisciplinary team consisting of specialists and residents in Psychiatry, Child and Adolescent Psychiatry and Neurology at Copenhagen University Hospitals and the Technical University of Denmark (DTU Compute) has worked in collaboration around this study since January 2013. Data collection was carried out from February 2014 to July 2016.

Data analysis

The Data

The Kinect application tracks the 3D joint positions in the upper body (the hands, wrists, elbows, shoulders, neck and head) with recordings 30 times per second. One Kinect Game session takes about 5 minutes and around 9000 3D observations are made for each joint. These data are time-series of varying length, which we have taken into account.

Hypotheses

1) Higher scores in rigidity (Simpson Angus Scale) are associated with a prolonged time of performance in the Kinect Motorgame.

2) Higher scores in information processing (Symbol Coding Task) are associated with decreased time of performance in the Kinect Motorgame.
3) Retest effect will be present in both patient and healthy control group.

Statistics

Descriptive analyses and simple group comparisons were analyzed using Pearson Chi-Square tests and independent t-test (SPSS version 22+23). Two-sided tests with alpha=0.05 were used.

For the statistical analysis of data from the Motorgame, we used a linear mixed-effect model[38;39] implemented in the package lmer[38] for the R-programming language.[40] This model contains both fixed effects and random effects.

The model used:

The terms in the model are: *y* the response, μ as a general mean, mean for each of the 22 tasks, *C* parameter for the clinical score, where is the value for that measurements on individual *i* in task *j*. The following terms correspond to the first letter in the enumeration of demographic and clinical variables above. The last two terms are , the random effect for individuals and the general error term.

Results

Descriptive data

Twenty-one children and adolescents with psychosis and in medical treatment with antipsychotics versus 70 healthy controls were studied (Table 1). Male proportion was 38.1% in the patient group versus 36.2% in the group of healthy controls (p=0.877). Mean age (±SD) was 16.02±1.37 years in patients and 16.04±1.52 years in healthy controls (p=0.960). Parental education level was significantly higher in the healthy

control group 7.35±0.76 compared to the patient group (6.25±1.33; p=0.002). The patients had been diagnosed within the schizophrenia spectrum (57.1%), affective psychosis (14.4%) and other psychosis (28.5%). Median duration of untreated psychosis (IQR) was 110 days (24-301,5 days). Thirteen of the 21 patients had already initiated on antipsychotics medication at first assessment with a median duration (IQR) of 4 weeks (0-56,5 weeks). Type of antipsychotic drug and median doses are seen in Table 1. Eight patients initiated the antipsychotics treatment on the same day as the first assessment.

Clinical rating scale scores

Clinical rating scale scores are shown in Table 2. Overall, the patient clinical rating scale scores of EPS were higher than the clinical rating scale scores in the healthy controls, but only at a significant level comparing the Simpson Angus Scales scores of the antipsychotics medicated patients (n=9) with the healthy controls (n=50)(p=0.048). In general, the severity displayed in the clinical rating scales of EPS were predominantly mild and in a few cases moderate. No participants had any dyskinesia. In contrast, clear differences were seen in the two tests from Brief Assessment of Cognition in Schizophrenia. Unmedicated patients (n=8) were significantly motoric slower in the Token Motor Task (45.4±18.5 point; p<0.001)) and showed slower information processing in the Symbol Coding Task (55.1±17.3 points; p=0.016) compared to the healthy controls (n=69; 67.9±15.1 points and 66.4±11.6 points, respectively). These differences remained significant after initiation of antipsychotic medication. Changes over time (from first to second assessment) are shown in Table 3. No significant between group differences were found. Even though not significant, the Motor Token Task appeared to have a retest effect in both the healthy control group and in the patient group. However, the patients initiating antipsychotic medication after the first assessment (n=6) seemed to have a poorer performance in the Symbol Coding Task (change of -3,5 points) at their second assessment, compared to the patients medicated at

both assessment points (n=3; change of 0 points; p=0.817) and the group of healthy controls (n=50; change of 3 points; p=0.444).

Mixed model analysis of the Motorgame data

All participants completed the Motorgame. The results from the mixed-effect model are shown in Table 4. Arm dropping (item 2 in the Simpson Angus scale) had a significant positive effect on the time of performance in the Motorgame (estimate 0.069; p=0.009) ie. prolonged time of motor performance was associated with higher scores of armdropping. Surprisingly, glabella tap (item 8 in the Simpson Angus scale) had a significant negative effect on the time of performance in the Motorgame (estimate -0.012; p<0.001) ie. shortened time of motor performance was associated with higher scores of glabella tap. A consistent and clear learning effect was seen in the Kinect Motor data (estimate -0.029 to -0.024; p<0.001). We did not find any moderating effect of gender, age, height, weight or the Symbol Coding Task. Data from the questionnaire (n=8) (Figure 2) showed that 88% found it easy or very easy to understand how to use the Motorgame. Seventyfive procent found the Motorgame fun or very fun to use. The Motorgame was the preferred test by 38%., but exceeded by the BACS tests (Token Motor Task and Symbol Coding Task) which was preferred by 63%.

Conclusion and Discussion

An increasing use of antipsychotics in both psychosis and non-psychotic diseases in children and adolescents is being reported in many countries.[41-44] Children and adolescents are more likely to experience EPS caused by antipsychotics compared to adults.[45]

Scientific evaluation of antipsychotic side effects are primarily made by the use of clinical rating scales. The fact that they are observer-based make them prone to inter-observer variability, and even well-trained raters may underestimate the incidence. [46]

Antipsychotic induced-bradykinesia (one out of 3 cardinal symptoms – tremor, rigidity,

bradykinesia - of drug-induced parkinsonism), especially in its early phases, is difficult to detect due to the overlap with negative symptoms and comorbid depression.[47] Consequently, early and objective detection of motor side effect in order to optimize antipsychotic treatment and increase treatment response and adherence is of pivotal importance, especially in the young population. The present study addresses a significant need for an objective method to assess movement disorders in young patients with schizophrenia or other psychotic diseases in antipsychotic treatment.

With the design of the present pilot study, we wanted to identify differences in movement patterns between young patients with psychosis and healthy controls and validate the Motorgame data against data from the established clinical Simpson Angus Scale, measuring antipsychotic drug-induced parkinsonism. Since our study population was very young and antipsychotic dosing as a consequence was relatively low, we did not expect a presentation of high scores in the clinical measurements (Simpson Angus scale). However, we did find a significant group difference ($p=0.048$) between antipsychotic medicated and healthy controls in the mean SAS total score. In line with previous findings from a study of reliability, sensitivity and validity of Brief Assessment of Cognition in Schizophrenia comparing 150 adult patients with schizophrenia and 50 healthy controls[48], we found that the outcome of motor speed (Token Motor Task) and information processing (Symbol Coding Task) were significantly worse in both unmedicated ($p<0.001$; $p=0.016$, respectively) and antipsychotic medicated patients (($p<0.001$; $p=0.003$, respectively) compared to the group of healthy controls in our study.

In the linear mixed model analysis, we found a positive significant effect ($p=0.009$) of arm dropping (item 2) in the SAS. This corresponds with our previous findings from the study of the Motorgame in adult patients with Parkinson's disease and healthy controls. In that study 8 items in the SAS (gait, arm dropping, shoulder shaking, elbow rigidity, wrist rigidity, leg pendulousness, head dropping and glabellar tap) had a significant effect on time to complete each task of the Motorgame. A possible reason could be that

the severity of the rigidity/bradykinesia, and by that the signal to the Kinect sensor, was much stronger in the patients with Parkinson's Disease. In contrast to the study in patients with Parkinson's Disease, we found a negative significant ($p<0.001$) effect of glabella tap (item 8) in the SAS. The glabellar reflex is one out of many behavioural motor responses found in neonates, which is subsequently inhibited before the age of 5 years.[49] An abnormal glabellar reflex is primarily seen in diffuse cerebral hemisphere degenerative and vascular diseases.[50] In a study of parkinsonian disorders (Parkinson disease, supranuclear palsy, multiple system atrophy) the glabellar tap showed a modest sensitivity, but a lack of specificity. [51] Glabellar tap is in general considered to be non-specific, non-diagnostic and a poor measure of Parkinson motor severity. [52] In studies of antipsychotic naïve or antipsychotic free (1-6 months) patients with schizophrenia, an abnormal glabella tap was found in 40-77%.[53;54] Other researchers have found that the glabella tap item is not correlated to the other items of parkinsonism in the Simpson Angus scale.[55] In our study, all participants with an abnormal glabella tap were found in the patient group, except from one healthy control with a score of 1 (mild). The finding that the glabella tap score had a negative effect on the time of performance in each task (ie. having a score>0 makes the participant perform faster) could indicate that an abnormal glabella tap is more likely to be associated with a dysfunction in the dopaminergic system rather than an antipsychotic-induced motor disorder.

Not surprisingly, especially in the light of the age and the expected familiarity with interactive gaming of the participants in this age group, a consistent and clear retest effect was found ($p<0.001$).

In contrast to our previous study of patients with Parkinson's disease and healthy controls, we did not find any effect of gender or information processing (Symbol Coding Task). Especially the lack of effect of information processing is somewhat surprising, since the difference in outcome of the Symbol Coding Task between healthy controls and unmedicated/medicated patients was highly significant. Furthermore, Fervaha and colleagues showed that the severity of drug-induced parkinsonism (Simpson Angus

scale) was reliably linked with poorer scores on tests of cognition[56] in 325 non-medicated adult patients with schizophrenia. It remains unclear whether a higher severity level of EPS would have been associated with a poorer outcome of the Symbol Coding Task and as a consequence have had a significant effect on the Kinect performance time. The Motorgame was reported easy and fun to use and was preferred above clinical rating scales. This is in line with previous studies in children and adolescents using the Microsoft Kinect: In a pilot project of physical rehabilitation in young adults with motor disabilities, a Kinect-based instrument showed enhanced motivation among the participants.[17] In a study of young patients with hemiplegic cerebral palsy, Rammer et al. [21] concluded that Kinect has the potential for improving functional assessment.

In a cross-over study of traditional exercise (a stationary cycle) and Kinect as an exercise intervention in 30 young subjects with Cystic Fibrosis[57], participants preferred Xbox Kinect for its interactivity.

Strengths and limitations

With the inclusion of healthy controls in this pilot study we were able to match data according to age and gender (but not educational level). Test-retest effects were tested, which are also relevant measures regarding motor abilities in this young and developing age group. The equipment used in this study is low-cost, easy accessible, portable and easy to administer and does not require expert knowledge to use. The similarities of the Kinect-based test to an ordinary computer game and the reduced time of physical contact with the physician are factors that are expected to increase the patient's willingness to participate.

The pilot study has obvious limitations. Firstly, the Motorgame covered only a part of the clinical examination (drug-induced parkinsonism) of EPSs, thus, it remains unclear whether the Kinect sensor can measure tremor and dyskinesia. Secondly, the developed Motorgame covered only upper extremity movements, which means that drug-induced parkinsonism involving the head and lower extremities were not assessed. However, this

is a limitation within the actual version of the software in the Kinect-based instrument, which a newer version might solve. Thirdly, even though the Kinect-based instrument was easy to administer there is a potential risk for technical problems that would require IT support. Fourthly, the number of patients in our study was very small. A larger population size would have added more power to the study and have reduced the risk of type II errors.

Clinical significance

This pilot study is a methodological development study. We found that a prolonged time of performance in the Motorgame was significantly associated with bradykinesia/rigidity in the shoulders. The results need to be redefined and implemented in future studies in order to increase our knowledge, particularly about what characterizes the movements of patients with higher clinical scale scores. Further development and studies of alternative sensors is needed to improve tracking of tremor (possibly by including newer versions of the software) and testing in a larger population is planned for the future.

Disclosures

The authors Ditte Rudå (DRU), Gudmundur Einarsson(GEI), Jannik Boll Nielsen (JBN), Karsten Gjessing Jensen (KGJ), Dea Gowers Klauber (DGK), Jens Richardt Jeppesen (JRJ), Birgitte Fagerlund (BF), Rasmus R. Paulsen (RRP), and Anne Katrine Pagsberg (AKP) declare that they have no competing interests. Kristian Winge (KWI) has been a consultant and/or advisor to Abbvie and GSK and has scientific collaboration with Lundbeck. Anders Fink Jensen (AFJ) has received an unrestricted research grand from Novo Nordisk. Cristoph Corell (CCO) has been a consultant and/or advisor to Bristol-Myers Squibb, Eli Lilly, Genentech, Gerson Lehrman Group, IntraCellular Therapies, Janssen/J&J, Lundbeck, MedAvante, Medscape, Otsuka, Pfizer, ProPhase, Roche, Sunovion, Supernus and has received honoraria from BMS, Janssen/J&J, Novo Nordisk

A/S, Otsuka, Takeda, Bristol-Myers Squibb, Janssen/J&J, Lundbeck, Medscape, Otsuka, ProPhase and Pfizer.

Authors' contributions

DR has initiated the study. DR, GEI, JBN, CCO, DGK, JRJ, BF, KWI, RRP, AKP, and AFJ have made important contributions to the study conception, design and protocol. DR has led the manuscript drafting. DR, AKP, AFJ, RRP and GEI have been involved in drafting. JBN, GEI and RRP have developed the application for the Kinect sensor. GEI is responsible for the mixed model analysis. DR and KGJ have done the assessments. All authors have critically revised the manuscript.

Acknowledgements

Mental health Centre for Child and adolescent Psychiatry, Capital Region of Denmark who has provided the patients for the study. Academic research assistants Tania Storm, Nana Suldrup Jørgensen and Saba Hamza and medical students Clara Ricard, Anne Sofie Schott Andersen and Sabrina Krøigaard have contributed to the Kinect assessments in the healthy control group of children and adolescents. Academic research assistant Nina Ramskov Siegismund has contributed to project coordination and logistics. All patients and healthy controls that have participated in the study.

Reference List

(1) Eggers C. Some remarks on etiological aspects of early-onset schizophrenia. *European child & adolescent psychiatry* 1999; 8 Suppl 1.

(2) Correll CU. Assessing and maximizing the safety and tolerability of antipsychotics used in the treatment of children and adolescents. *J Clin Psychiatry* 2008; 69 Suppl 4:26-36.

(3) Cohen D, Bonnot O, Bodeau N, Consoli A, Laurent C. Adverse effects of second-generation antipsychotics in children and adolescents: a Bayesian meta-analysis. *Journal of clinical psychopharmacology* 2012; 32(3).

(4) Stentebjerg-Olesen M, Jeppesen P, Pagsberg AK, Fink-Jensen A, Kapoor S, Chekuri R et al. Early nonresponse determined by the clinical global impressions scale predicts poorer outcomes in youth with schizophrenia spectrum disorders naturalistically treated with second-generation antipsychotics. *J Child Adolesc Psychopharmacol* 2013; 23(10):665-675.

(5) Buchanan A. A two-year prospective study of treatment compliance in patients with schizophrenia. *Psychological medicine* 1992; 22(3).

(6) Dean CE, Russell JM, Kuskowski MA, Caligiuri MP, Nugent SM. Clinical rating scales and instruments: how do they compare in assessing abnormal, involuntary movements? *J Clin Psychopharmacol* 2004; 24(3):298-304.

(7) Koning JP, Tenback DE, Kahn RS, Van Schelven LJ, Van Harten PN. Instrument measurement of lingual force variability reflects tardive tongue dyskinesia. *Journal of medical engineering & technology* 2010; 34(1).

(8) Caligiuri MP, Teulings HL, Dean CE, Niculescu AB3, Lohr JB. Handwriting movement kinematics for quantifying extrapyramidal side effects in patients treated with atypical antipsychotics. *Psychiatry research* 2010; 177(1-2).

(9) Caligiuri MP, Lohr JB. Fine force instability: a quantitative measure of neuroleptic-induced dyskinesia in the hand. *The Journal of neuropsychiatry and clinical neurosciences* 1990; 2(4).

(10) Walther S, Koschorke P, Horn H, Strik W. Objectively measured motor activity in schizophrenia challenges the validity of expert ratings. *Psychiatry research* 2009; 169(3).

(11) Putzhammer A, Klein HE. Quantitative analysis of motor disturbances in schizophrenic patients. *Dialogues in clinical neuroscience* 2006; 8(1).

(12) Perry W, Minassian A, Paulus MP, Young JW, Kincaid MJ, Ferguson EJ et al. A reverse-translational study of dysfunctional exploration in psychiatric disorders: from mice to men. *Archives of general psychiatry* 2009; 66(10).

(13) Bonnechere B, Jansen B, Salvia P, Bouzahouene H, Omelina L, Moiseev F et al. Validity and reliability of the Kinect within functional assessment activities: comparison with standard stereophotogrammetry. *Gait & posture* 2014; 39(1).

(14) Allahyari T, Sahraneshin SA, Khalkhali HR. Validity of the Microsoft Kinect for measurement of neck angle: comparison with electrogoniometry. *Int J Occup Saf Ergon* 2016;1-9.

(15) Springer S, Yogev SG. Validity of the Kinect for Gait Assessment: A Focused Review. *Sensors (Basel)* 2016; 16(2):194.

(16) Clark RA, Pua YH, Fortin K, Ritchie C, Webster KE, Denehy L et al. Validity of the Microsoft Kinect for assessment of postural control. *Gait Posture* 2012; 36(3):372-377.

(17) Chang YJ, Chen SF, Huang JD. A Kinect-based system for physical rehabilitation: a pilot study for young adults with motor disabilities. *Research in developmental disabilities* 2011; 32(6).

(18) Olesh EV, Yakovenko S, Gritsenko V. Automated Assessment of Upper Extremity Movement Impairment due to Stroke. *PLoS One* 2014; 9(8).

(19) Svendsen MB, Preisler L, Hillingsoe JG, Svendsen LB, Konge L. Using motion capture to assess colonoscopy experience level. *World J Gastrointest Endosc* 2014; 6(5).

(20) Rosenberg M, Thornton AL, Lay BS, Ward B, Nathan D, Hunt D et al. Development of a Kinect Software Tool to Classify Movements during Active Video Gaming. *PLoS One* 2016; 11(7):e0159356.

(21) Rammer JR, Krzak JJ, Riedel SA, Harris GF. Evaluation of upper extremity movement characteristics during standardized pediatric functional assessment with a Kinect(R)-based markerless motion analysis system. *Conf Proc IEEE Eng Med Biol Soc* 2014; 2014:2525-2528.

(22) Rocha AP, Choupina H, Fernandes JM, Rosas MJ, Vaz R, Silva Cunha JP. Parkinson's disease assessment based on gait analysis using an innovative RGB-D camera system. *Conf Proc IEEE Eng Med Biol Soc* 2014; 2014:3126-3129.

(23) Tupa O, Prochazka A, Vysata O, Schatz M, Mares J, Valis M et al. Motion tracking and gait feature estimation for recognising Parkinson's disease using MS Kinect. *Biomed Eng Online* 2015; 14:97.

(24) Takac B, Catala A, Rodriguez MD, van der Aa N, Chen W, Rauterberg M. Position and orientation tracking in a ubiquitous monitoring system for Parkinson disease patients with freezing of gait symptom. *JMIR Mhealth Uhealth* 2013; 1(2):e14.

(25) Cao Y, Li BZ, Li QN, Xie JD, Cao BZ, Yu SY. Kinect-based gait analyses of patients with Parkinson's disease, patients with stroke with hemiplegia, and healthy adults. *CNS Neurosci Ther* 2017; 23(5):447-449.

(26) Zhao J, Bunn FE, Perron JM, Shen E, Allison RS. Gait assessment using the Kinect RGB-D sensor. *Conf Proc IEEE Eng Med Biol Soc* 2015; 2015:6679-6683.

(27) Cancela J, Arredondo MT, Hurtado O. Proposal of a Kinect(TM)-based system for gait assessment and rehabilitation in Parkinson's disease. *Conf Proc IEEE Eng Med Biol Soc* 2014; 2014:4519-4522.

(28) Galna B, Barry G, Jackson D, Mhiripiri D, Olivier P, Rochester L. Accuracy of the Microsoft Kinect sensor for measuring movement in people with Parkinson's disease. *Gait Posture* 2014; 39(4):1062-1068.

(29) The ICD-10 Classification of Mental and Behavioral Disorders. Clinical and Diagnostic guidelines. 1992.

(30) Kaufman J, Birmaher B, Brent D, Rao U, Flynn C, Moreci P et al. Schedule for Affective Disorders and Schizophrenia for School-Age Children-Present and Lifetime Version (K-SADS-PL): initial reliability and validity data. *J Am Acad Child Adolesc Psychiatry* 1997; 36(7):980-988.

(31) Pagsberg AK, Jeppesen P, Klauber DG, Jensen KG, Ruda D, Stentebjerg-Olesen M et al. Quetiapine versus aripiprazole in children and adolescents with psychosis--protocol for the randomised, blinded clinical Tolerability and Efficacy of Antipsychotics (TEA) trial. *BMC Psychiatry* 2014; 14:199.

(32) Guy WNIoMHPRBECDEP. Abnormal Involuntary Movement Scale (AIMS). In: Guy W, editor. Rockville, Md.: U.S. Dept. of Health, Education, and Welfare, Public Health Service, Alcohol, Drug Abuse, and Mental Health Administration,

National Institute of Mental Health, Psychopharmacology Research Branch, Division of Extramural Research Programs; 1976.

(33) Simpson GM, Angus JW. A rating scale for extrapyramidal side effects. *Acta psychiatrica Scandinavica Supplementum* 1970; 212.

(34) Barnes TR. A rating scale for drug-induced akathisia. *The British journal of psychiatry : the journal of mental science* 1989; 154.

(35) Lingjaerde O, Ahlfors UG, Bech P, Dencker SJ, Elgen K. The UKU side effect rating scale. A new comprehensive rating scale for psychotropic drugs and a cross-sectional study of side effects in neuroleptic-treated patients. *Acta psychiatrica Scandinavica Supplementum* 1987; 334.

(36) Keefe RS, Goldberg TE, Harvey PD, Gold JM, Poe MP, Coughenour L. The Brief Assessment of Cognition in Schizophrenia: reliability, sensitivity, and comparison with a standard neurocognitive battery. *Schizophr Res* 2004; 68(2-3):283-297.

(37) Caroff SN, Hurford I, Lybrand J, Campbell EC. Movement disorders induced by antipsychotic drugs: implications of the CATIE schizophrenia trial. *Neurologic clinics* 2011; 29(1).

(38) Kuznetsova ABPBCRHB. lmerTest: tests for random and fixed effects for linear mixed effect models (lmer objects of lme4 package). http://CRAN.R-project.org/package=lmerTest . 2013.
Ref Type: Online Source

(39) McLean R.A. SWLSWW. A unified approach to mixed linear models. *Am Stat* 1991; 45:54-64.

(40) Team RC. R language definition. 2000. 2000.
Ref Type: Online Source

(41) Olfson M. Antipsychotic prescriptions for children and adolescents in the UK increased from 1993 to 2005. *Evidence-based mental health* 2009; 12(1).

(42) Olfson M, Blanco C, Liu L, Moreno C, Laje G. National trends in the outpatient treatment of children and adolescents with antipsychotic drugs. *Archives of general psychiatry* 2006; 63(6).

(43) Olfson M, Blanco C, Liu SM, Wang S, Correll CU. National trends in the office-based treatment of children, adolescents, and adults with antipsychotics. *Archives of general psychiatry* 2012; 69(12).

(44) Olfson M, Crystal S, Huang C, Gerhard T. Trends in antipsychotic drug use by very young, privately insured children. *J Am Acad Child Adolesc Psychiatry* 2010; 49(1):13-23.

(45) Correll CU. Antipsychotic use in children and adolescents: minimizing adverse effects to maximize outcomes. *Journal of the American Academy of Child and Adolescent Psychiatry* 2008; 47(1).

(46) Wolff AL, O'Driscoll GA. Motor deficits and schizophrenia: the evidence from neuroleptic-naive patients and populations at risk. *Journal of psychiatry & neuroscience : JPN* 1999; 24(4).

(47) Prosser ES, Csernansky JG, Kaplan J, Thiemann S, Becker TJ, Hollister LE. Depression, parkinsonian symptoms, and negative symptoms in schizophrenics treated with neuroleptics. *The Journal of nervous and mental disease* 1987; 175(2).

(48) Keefe RS, Goldberg TE, Harvey PD, Gold JM, Poe MP, Coughenour L. The Brief Assessment of Cognition in Schizophrenia: reliability, sensitivity, and comparison with a standard neurocognitive battery. *Schizophr Res* 2004; 68(2-3):283-297.

(49) Tomita Y, Shichida K, Takeshita K, Takashima S. Maturation of blink reflex in children. *Brain Dev* 1989; 11(6):389-393.

(50) Pearce JM. Observations on the blink reflex. *Eur Neurol* 2008; 59(3-4):221-223.

(51) Brodsky H, Dat VK, Thomas M, Jankovic J. Glabellar and palmomental reflexes in Parkinsonian disorders. *Neurology* 2004; 63(6):1096-1098.

(52) Friedman JH, Abrantes AM. The glabellar reflex is a poor measure of Parkinson motor severity. *Int J Neurosci* 2013; 123(6):417-419.

(53) Stevens JR. Eye blink and schizophrenia: psychosis or tardive dyskinesia? *The American journal of psychiatry* 1978; 135(2):223-226.

(54) Stevens JR. Disturbances of ocular movements and blinking in schizophrenia. *J Neurol Neurosurg Psychiatry* 1978; 41(11):1024-1030.

(55) Sanchez R, Calvo JM, Jaramillo LE. [Is the glabellar reflex a component of neuroleptic-induced Parkinsonism?]. *Biomedica* 2005; 25(4):539-546.

(56) Fervaha G, Agid O, Takeuchi H, Lee J, Foussias G, Zakzanis KK et al. Extrapyramidal symptoms and cognitive test performance in patients with schizophrenia. *Schizophr Res* 2015; 161(2-3):351-356.

(57) Salonini E, Gambazza S, Meneghelli I, Tridello G, Sanguanini M, Cazzarolli C et al. Active Video Game Playing in Children and Adolescents With Cystic Fibrosis: Exercise or Just Fun? *Respir Care* 2015; 60(8):1172-1179.

Table 1. Descriptive data

	Antipsychotic naïve patients n=8	**Antipsychotic medicated patients n=13**	**Total group of patients n=21**	**Healthy controls n=69**	**P value, patients versus controls**
Age in years (range 13.58-19.08), mean±SD	15.54±1.13	16.31±1.46	16.02±1.37	16.04±1.52	0.960[a]
Male, n (%)	4 (50.0)	4 (30.8)	8 (38.1)	25 (36.2)	0.877[b]
Parental education, mean±SD	5.71±1.38	6.54±1.27	6.25±1.33	7.35±0.76	**0.002**[a]
Psychosis diagnoses, n (%) F20.0 F21.9 F23.9 F25.1 F28.9 F29.9 F32.3 F33.3	 9 (69.2) 0 (0.0) 1 (7.7) 1 (7.7) 0 (0.0) 0 (0.0) 1 (7.7) 1 (7.7)	 3 (37.5) 2 (25.0) 0 (0.0) 0 (0.0) 1 (12.5) 2 (25.0) 0 (0.0) 0 (0.0)	 12 (57.1) 2 (9.5) 1 (4.8) 1 (4.8) 1 (4.8) 2 (9.5) 1 (4.8) 1 (4.8)		
Psychiatric comorbid diagnosis, n (%)	4 (50.0)	7 (53.8)	11 (52.4)		
Duration of untreated psychosis, median weeks (IQR)	180.0 (48.0-624.0)	55.0 (22.0-220.0)	110.0 (24.0-301.5)		
Antipsychotic compound Aripiprazole, n (%) Aripiprazole, median dose, mg (IQR) Quetiapine, n (%) Quetiapine, median dose, mg (IQR) Quetiapine-ER, n (%) Quetiapine-ER, median dose, mg (IQR) Serenase, n (%) Serenase, median dose, mg (IQR)	 5 (62.5)[c] 5.0 (3.75-10.0) 2 (25.0)[c] 212.5 (25.0-212.5) 0 (0.0)[c] - 0 (0.0)[c] -	 6 (46.2) 15.0 (15.0-20.0) 3 (23.1) 200.0 (20.0-200.0) 5 (38.5) 600.0 (150.0-600.0) 1 (7.7) 2 (2.0-2.0)	 11 (52.4) 15.0 (5.0-15.0) 5 (23.8) 200.0 (22.5-400.0) 5 (23.8) 600.0 (150.0-600.0) 1 (4.8) 2 (2.0-2.0)		
Duration of antipsychotic treatment at first assessment, median weeks (IQR)	0 (0-0)	56.0 (8.0-60.0)	4 (0.0-56.5)		

[a]Independent t-test; [b]Pearson Chi-Square test; [c]Antipsychotic compounds at second assessment, where the antipsychotic naïve patients have been introduced to antipsychotic treatment; SD Standard Deviation. IQR Interquartile Range.

Table 2. Clinical rating scale scores

	Antipsychotic naïve patients		**Antipsychotic medicated patients**		**Healthy controls**		**P value[a]**	**P value[b]**	**P value[c]**
	First Kinect assessment, unmedicated, n=8	Second Kinect assessment, medicated, n=6	First Kinect assessment, medicated, n=13	Second Kinect assessment, medicated, n=3	First Kinect assessment, n=69	Second Kinect assessment, n=50			
Item 2.3 UKU (bradykinesia), mean±SD	0.13±0.35	0.33±0.52	0.08±0.28	0.0±0.0	0.0±0.0	0.0±0.0	0.169	0.732	0.351
Mean SAS total score, mean±SD	0.09±0.14	0.23±0.22	0.16±0.16	0.13±0.15	0.07±0.14	0.09±0.14	**0.048**	0.281	0.664
BARS global score, mean±SD	0.63±0.74	1.17±1.60	1.00±1.00	0.67±1.16	0.03±0.17	0.02±0.14	0.071	0.373	0.058
Tardive dyskinesia, n (%)	0	0	0	0	0	0	-	-	-
Token Motor Task, mean±SD	45.4±18.5	49.0±14.5	47.7±18.5	56.0±14.1	67.9±15.1	75.3±13.7[d]	**<0.001**	0.789	**<0.001**
Symbol Coding Task, mean±SD	55.1±17.3	52.2±17.5	53.0±15.0	64.0±7.1	66.4±11.6	72.7±11.1[d]	**0.003**	0.774	**0.016**

a) Antipsychotic medicated patients versus healthy controls (comparison between second Kinect assessments)
b) Unmedicated patients versus medicated patients (comparison between first Kinect assessments)
c) Unmedicated patients versus healthy controls (comparison between first Kinect assessments)
d) Only 20 healthy participants were assessed regarding Token Motor Task and Symbol Coding Task at second Kinect assessment
SD: Standard Deviation

Table 3. Changes over time in clinical rating scale scores.

	Antipsychotic naïve patients, n=6	**Antipsychotic medicated patients, n=3**	**Healthy controls n=50**	**P value[a]**	**P value[b]**	**P value[c]**
Change Item 2.3 UKU (bradykinesia), mean±SD	0.167±0.753	-0.333±0.577	0.00±0.0	0.423	0.351	0.611
Change mean SAS total score, mean±SD	0.183±0.214	-0.033±0.321	0.012±0.117	0.830	0.259	0.108
Change BARS global score, mean±SD	0.667±1.633	-0.067±0.577	-0.020±0.247	0.191	0.224	0.351
Change Token Motor Task, mean±SD	3.17±20.81	8.00±5.66	5.25±17.76	0.833	0.768	0.810
Change Symbol Coding Task, mean±SD	-3.5±18.89	0.00±9.90	3.00±7.41	0.598	0.817	0.444

a) Antipsychotic medicated patients versus healthy controls
b) Unmedicated patients versus medicated patients
c) Unmedicated patients versus healthy controls
SD: Standard Deviation

Table 4. The effect of Simson Angus Scale items on time of motor performance in the Kinect Motorgame

Model	SAS item	Gender	Age	Height	Weight	Symbol Coding Task	Repetition
Item 1: Gait	0.05277080±	0.02517919±	-0.0011167±	0.00158057±	-0.0009950±	-0.0005627±	-0.0290173±
Parameter Value±SD	0.03826630	0.01769783	0.0047764	0.00113420	0.0007301	0.0005026	0.0061134
P-value	0.16839253	0.15846265	0.8157226	0.16709873	0.1764375	0.2660262	**0.0000022**
Item 2: Arm dropping	0.06866780±	0.02687732±	-0.0004335±	0.00132017±	-0.0007718±	-0.0007256±	-0.0249566±
Parameter Value±SD	0.02615608	0.01723641	0.0046672	0.00110111	0.0006909	0.0004849	0.0061977
P-value	**0.00872368**	0.12254856	0.9262124	0.23382922	0.2671490	0.1382197	**0.0000579**
Item 3: Shoulder shaking	0.01663420±	0.02880679±	-0.0009367±	0.00149523±	-0.0007459±	-0.0007027±	-0.0281046±
Parameter Value±SD	0.01462769	0.01749701	0.0047066	0.00111239	0.0006974	0.0004898	0.0060867
P-value	0.25585447	0.10323483	0.8427187	0.18242055	0.2878951	0.1550418	**0.0000040**
Item 4: Elbow rigidity	-0.0014836±	0.02647410±	-0.0012066±	0.00141862±	-0.0007562±	-0.0006655±	-0.0281010±
Parameter Value±SD	0.0149801	0.01756722	0.0047322	0.00111685	0.0007020	0.0004928	0.0061481
P-value	0.9211359	0.13540278	0.7993616	0.20745887	0.2844993	0.1804201	**0.0000050**
Item 5: Wrist rigidity	0.0345330±	0.02871345±	-0.0009966±	0.00145294±	-0.0007439±	-0.0006953±	-0.0276838±
Parameter Value±SD	0.0359208	0.01751852	0.0047037	0.00111053	0.0006971	0.0004894	0.0061063
P-value	0.3369884	0.10476747	0.8327089	0.19426334	0.2890113	0.1591245	**0.0000060**
Item 6: Leg pendulousness	0.0208511±	0.02684927±	-0.0008884±	0.00140072±	-0.0007158±	-0.0006740±	-0.0282418±
Parameter Value±SD	0.0279615	0.01745283	0.0047338	0.00111449	0.0007014	0.0004905	0.0060883
P-value	0.4560965	0.12758829	0.8515689	0.21222070	0.3104183	0.1730877	**0.0000036**
Item 7: Head dropping	-0.0147955±	0.02519654±	-0.0014617±	0.00141268±	-0.0007746±	-0.0006582±	-0.0283486±
Parameter Value±SD	0.0209356	0.01765388	0.0047567	0.00111975	0.0007038	0.0004932	0.0060917
P-value	0.4801523	0.15705802	0.7593671	0.21053140	0.2743055	0.1856086	**0.0000034**
Item 8: Glabella tap	-0.0197242±	0.03293384±	-0.0010851±	0.00159485±	-0.0006150±	-0.0009878±	-0.0270163±
Parameter Value±SD	0.0056440	0.01845385	0.0049660	0.00117384	0.0007379	0.0005246	0.0060964
P-value	**0.0004981**	0.07802573	0.8275991	0.17806689	0.4071433	0.0632222	**0.0000097**
Item 9: Tremor	-0.0144065±	0.02617530±	-0.0011869±	0.00134928±	-0.0007515±	-0.0006964±	-0.0252157±
Parameter Value±SD	0.0084000	0.01763989	0.0047681	0.00112691	0.0007073	0.0004961	0.0063363
P-value	0.0870404	0.14150097	0.8040263	0.23449690	0.2911272	0.1640864	**0.0000707**
Item 10: Salivation	-0.0138284±	0.02633352±	-0.0012347±	0.00141821±	-0.0007724±	-0.0006952±	-0.0285929±
Parameter Value±SD	0.0212794	0.01744330	0.0047136	0.00111326	0.0006997	0.0004918	0.0061225
P-value	0.5159027	0.13475688	0.7939995	0.20612804	0.2728021	0.1610969	**0.0000031**

Figure 1: *Kinect assessment*

Figure 2: *Results from the questionnaire about the participants opinion about the Kinect Motorgame*