

Adaptive sequential feature selection in visual perception and pattern recognition

Von der Fakultät für Mathematik und Informatik
der Universität Leipzig
angenommene

D I S S E R T A T I O N

zur Erlangung des akademischen Grades

DOCTOR RERUM NATURALIUM
(Dr.rer.nat.)

im Fachgebiet

Informatik

vorgelegt

von Master of Philosophy Liliya Avdiyenko
geboren am 28.01.1985 in Charkiw, Ukraine

Die Annahme der Dissertation wurde empfohlen von:

1. Professor Dr. Jürgen Jost (MPI für Mathematik in den Naturwissenschaften, Leipzig)
2. Professor Dr. Matthias Bethge (MPI für biologische Kybernetik, Tübingen)

Die Verleihung des akademischen Grades erfolgt mit Bestehen der Verteidigung am 15.09.2014 mit dem Gesamtprädikat cum laude.

Acknowledgments

There are many people who supported me during my PhD and whom I want to thank from the bottom of my heart.

First of all, I am grateful to my supervisor, Prof. Dr. Jürgen Jost, for giving me an opportunity to develop myself as a scientist in the excellent research environment of his group. I am thankful for everything he taught me, for his guidance, inspiration and openness to new ideas and research directions of his students.

Special thanks go to my “unofficial supervisor”, collaborator and my friend, Dr. Nils Bertschinger. I appreciate our long discussions, his interesting stories about programming languages and inference algorithms and of course his enormous patience and help with formalizing my heuristic reasoning.

Also I want to say my sincere thank you to my colleagues, Wiktor Młynarski, Dr. Tobias Elze, Dr. Hedwig Wilhelm and Dr. Johannes Rauh, for being always there when I needed a piece of advice. In addition, I am grateful to them for proofreading my thesis. There are other colleagues whom I did not mention but whom I am thankful for sharing with me a scientific and nonscientific fun of being a graduate student at our Max Planck Institute.

This thesis would not be possible without the financial support of the Max Planck Society that gave me a possibility to do my research as well as present it on various international workshops and conferences.

I would like to mention two magicians of our institute, Antje Vandenberg and Heike Rackwitz, who make lives of foreign PhD students easy and let them feel like at home. I am very grateful for all their support and assistance during my stay at MPI. I want to thank also librarians, a computer group and other institute staff for making it possible to concentrate only on scientific problems.

Last but not least, I would like to thank my friends, my dear Michael, my family and especially my parents, Natalia and Sergii. Their love, belief in me and endless support were with me at every moment of my PhD life and gave me strength to complete this work successfully.

Contents

Acknowledgments	iii
1 Introduction	1
2 Conventional feature selection	7
2.1 Main approaches to feature selection	7
2.2 Feature selection framework	10
2.2.1 Classification setup	10
2.2.2 General framework of feedforward selection	11
2.3 Selection criteria	12
2.3.1 Misclassification error	12
2.3.2 Gini index	14
2.3.3 Shannon entropy	16
2.3.4 Gain ratio	17
2.3.5 Alpha-entropies	18
2.3.6 Correlation-based feature selection	20
2.3.7 Probabilistic distance measures	22
2.4 Information-theoretical feature selection	24
2.4.1 Definitions	24
2.4.2 Use in solving classification tasks	27
2.5 Estimation of mutual information	30

2.5.1	Plug-in approaches	31
2.5.1.1	Density estimation.	31
2.5.2	Nonplug-in approaches	41
2.6	Approximated schemes	45
2.7	Conclusion	47
3	Adaptive feature selection	51
3.1	Biological motivation	51
3.2	Adaptive feature selection	54
3.3	Framework	56
3.3.1	Relation to complex adaptive systems	57
3.4	Existing algorithms	58
3.4.1	Local feature selection by decision trees	58
3.4.2	Active testing model	60
3.4.3	Jiang's sequential feature selector	62
3.5	Adaptive conditional mutual information feature selector	64
3.5.1	Model	64
3.5.1.1	Adaptive vs static selection	66
3.5.2	Estimation	68
3.5.2.1	Density estimation	68
3.5.2.2	Conditional Expectation	72
3.5.2.3	Smoothing	74
4	Experimental investigations of ACMIFS	83
4.1	Data sets	84
4.1.1	Artificial data set	84
4.1.2	MNIST data set	87

4.2	Smoothing	89
4.2.1	Additive smoothing	89
4.2.2	Interpolation with a prior distribution	92
4.3	Comparison with PWFS and ATM	94
4.3.1	General comparison on the artificial data set	96
4.3.2	General comparison on MNIST	97
4.3.3	Behavior in higher dimensions	98
4.4	Combined selection scheme	101
4.5	Conclusions	104
5	Information-theoretical strategies of selective attention	107
5.1	Introduction	107
5.1.1	Existing task-dependent strategies of selective attention	108
5.2	Experimental setup	111
5.3	Tested information-theoretical search strategies	113
5.3.1	Mutual information	113
5.3.2	Conditional mutual information	113
5.3.3	Adaptive conditional mutual information	114
5.4	Sequence statistics	114
5.4.1	Generative model	115
5.4.2	Base likelihood	117
5.5	Experiment	118
5.5.1	Stimuli	118
5.5.2	Presentation software	119
5.5.3	Participants	120
5.6	Results	120
5.6.1	Subject statistics	121

5.6.2	Subject clusters	123
5.7	Conclusions	125
6	Discussion	131
A	Appendix	137
A.1	Visual cortex	137
A.2	Artificial dataset	138
A.3	Example of the clicking presentation	139
	Bibliography	149

Chapter 1

Introduction

Machine learning is often confronted with high-dimensional data. A common problem is the so-called “curse of dimensionality”, meaning that an amount of data needed to accurately learn parameters of a model grows exponentially with a number of input dimensions. For this reason, as well as computational issues, feature selection is often used to reduce the data dimensionality to features that are relevant for solving a given problem, such as classification. Moreover, in a situation when a training set is of the limited size, a classifier built on a smaller number of features usually has better generalization ability.

Basically, one can distinguish between two types of feature selection algorithms: filters and wrappers [Webb, 1999]. Filters try to reduce the data dimensionality while keeping potential clusters in the data well separated. In this case, the relevance of each feature is evaluated using different measures of a feature’s ability to discriminate between classes. Wrappers also preprocess the data but directly take into account that the resulting features should be useful for a certain classifier. Therefore, features are selected based on the prediction accuracy of the classifier employing these features. This might lead to better results but is usually computationally demanding and prone to overfitting.

For both wrappers and filters, the best feature subset of a certain cardinality can be found using an optimal search strategy. However, a number of possible subsets is exponentially large, therefore, testing all of them is computationally infeasible. To tackle this problem, Narendra and Fukunaga proposed the branch and bound method that assumes monotonicity of a selection criterion, which allows to avoid an exhaustive search [Narendra & Fukunaga, 1977]. If such an assumption is not valid and the number of features is large, suboptimal methods have to be used. This class of algorithms includes forward and backward sequential feature selection, where a subset of relevant features is formed by iteratively adding relevant features or removing irrelevant ones, respectively, e. g. [Ding & Peng, 2005; Abe, 2005].

For feature selection algorithms of the filter type, one of the central questions concerns a selection criterion, i. e. a notion of the feature relevance. An intuitive choice for such criterion is the Bayes error probability of classification using a considered feature [Breiman et al., 1984]. Another popular family of techniques uses different dependency and correlation measures to determine the degree of association between classes and a feature [Mingers, 1987; Duch, 2006]. However, since such measures are usually pairwise, these techniques are not able to discover high-order dependencies in order to avoid selecting mutually redundant features. As a partial solution, Hall proposed a correlation-based measure punishing features that are highly pairwise correlated with the previously selected features within the sequential feedforward setup [Hall, 1999].

Among probabilistic criteria used by filters, selection criteria based on Shannon entropy, a measure of uncertainty in the information theory, are widely used [Duch et al., 2004]. Such criteria select features to reduce uncertainty about the class. Moreover, it was also shown that features that have high mutual information with a class variable, a concept closely related to the Shannon entropy, are indeed useful for classification [Lewis, 1962; Brown et al., 2012]. Despite numerous estimators of mutual information developed in the last several decades [Beirlant et al., 1997; Nemenman et al., 2002; Kraskov et al., 2004], its estimation is still considered to be a hard task. However, a problem of feature selection does not require precise values of mutual information. Therefore, even if an estimator is biased, it is sufficient to have the right ordering of features according to their informativeness, which significantly reduces requirements to the quality of estimates.

Battiti was one of the first to use mutual information, for sequential feature selection [Battiti, 1994]. However, this involves estimation of the conditional mutual information (CMI), i. e. the amount of information between the feature and the class given the already selected features, which requires multivariate density estimation. To circumvent this problem, Battiti approximated CMI by pairwise mutual information. In addition, his work gave rise to the development of various related approximations of conditional mutual information as a criterion for feature selection, e. g. [Yang & Moody, 1999; Kwak & Choi, 2002b; Fleuret & Guyon, 2004]. Alternatively, kernel density estimation is a non-parametric technique widely used for multivariate density estimation. It was successfully applied to estimate CMI and related quantities for the exhaustive search procedure [Bonnlander & Weigend, 1994] and forward feature selection [Kwak & Choi, 2002a; Bonnlander, 1996].

The feature selection algorithm developed in this thesis is inspired by the hypothesis checking mechanism in the human visual system, which is implemented using numerous feedback connections coming from the higher brain areas to the lower ones [Mumford, 1991; Bullier, 2001]. Due to the so-called information bottleneck referring to the limited capabilities of visual processing, only a restricted amount of information can be processed at the same time [Van Essen et al., 1991]. After the first portion of the input is processed

by bottom-up circuits, an initial set of hypotheses about a visual scene is formed in the higher brain areas. If at this stage the scene can not be unambiguously classified, i. e. there is still some uncertainty about the class and therefore no single hypothesis can be chosen, a top-down signal from the higher areas will initiate processing of the next input portion in order to refine the current hypothesis set. Such selection-refinement process will be iteratively repeated until the visual scene is classified.

One can think about small portions of the visual input as its features. Then, the described scheme is nothing else but a feature selection algorithm that selects features relevant for classification of a certain visual scene. Thus, the selection is adapted to an object that should be classified. This phenomenon inspired us to develop a computational algorithm solving a visual classification task that would incorporate such principle, adaptive feature selection. It is especially interesting because usually feature selection methods are not adaptive as they define a unique set of informative features for a task and use them for classifying all objects. However, an adaptive algorithm selects features that are the most informative for the particular input. Thus, the selection process should be driven by statistics of the environment concerning the current task and the object to be classified, which in machine learning terms are called a training set and a testing sample, respectively. In this context, the main question we ask in this thesis is whether the proposed adaptive way of selecting features is advantageous and in which situations. Similarly to the visual system where feedback is necessary for recognizing ambiguous objects, we expect that adaptive feature selection should be advantageous for complex classification tasks where it is difficult to define a single static feature subset of a moderate size that would be sufficient for the accurate classification. In particular, the usage scenarios for the adaptive selection scheme are the following.

When the structure of data is heterogeneous, one may need different features to discriminate between classes, or even different objects belonging to one class may have different discriminative features. As a result, it is very likely that no single small subset of features is good enough for classification of all observations. One can partially overcome this problem by having a collection of all relevant feature subsets. This, however, will lead to an increase in the classifier complexity, which in turn will lead to its poor performance, unless a large amount of training data is available for training a classifier in high-dimensional space [Raudys & Jain, 1991]. Thus, conventional feature selection schemes, which select a fixed subset of features before they are handed to a classifier, can be inefficient.

In addition to the case with heterogeneous data, we expect the adaptive approach to feature selection to be advantageous when the amount of available training data is limited and the number of features exceeds the number of training samples. If features are selected in the adaptive way, their relevance is judged only for a small subregion of the input space where a testing sample lies. At the same time, static schemes look for features that are globally

relevant, i. e. features with the high discriminative power for all samples from a training set. Therefore, it is very probable that in the undersampled regime, when the training set does not fully represent the true data distribution, estimates of the local relevance would be more accurate than those of the global relevance. As a result, quality of the adaptively selected features would be better and in order to reach the same classification accuracy, one would need a smaller number of adaptively selected features comparing to static selection schemes.

Thus, in cases when it is difficult to find a small fixed subset of relevant features, we propose to use different features for every testing sample, i. e. select the informative features in an “adaptive” manner. By adaptivity we mean that for a certain testing sample every selected feature should be maximally relevant for its classification given values of the already selected features observed on this testing sample.

The idea of adaptivity was used by Geman and Jedynek in their active testing model [Geman & Jedynek, 1996] where they sequentially select tests in order to reduce uncertainty about the true hypothesis. For their problem domain, they assumed that features are conditionally independent given the class, which simplified the estimation. Jiang also used an adaptive scheme [Jiang, 2008], however, without conditioning on the already selected features, which are employed only to update a set of currently active classes. In contrast to these schemes, we adaptively select features taking into account high-order dependencies between them.

Therefore, we propose an adaptive feature selection algorithm that utilizes a selection criterion based on Shannon entropy. Applied to a classification task, our adaptive feature selection algorithm sequentially adds features one by one to a subset of features in order to reduce uncertainty about a class of a certain testing sample. In information-theoretical terms, a selection criterion is the mutual information of a class variable and a feature-candidate conditioned on the already selected features, which take values observed on the current testing sample. Hence, we call it adaptive conditional mutual information feature selector (ACMIFS). For its estimation, we utilize a plug-in estimator based on kernel density estimates with the proposed here adaptive smoothing. Even though the mutual information is hard to estimate in general and from small data sets especially, practical investigations of the algorithm show that it is able to select informative features in high dimensions.

It is well-established that there are two factors affecting shifts of the visual attention: visual stimuli themselves and a task. While the influence of image statistics on the viewing behavior is intuitive, a fact that a saccade sequence differs depending on a task had to be proven experimentally [Yarbus, 1967; Rothkopf et al., 2007; Betz et al., 2010]. However, the question remains what kind of strategy people use to decide what is relevant for a task, e. g. simple heuristics or complex algorithms based on the ideas of information theory etc. Surprisingly, despite their computational complexity, statistical and

information-theoretical definitions of the task-relevance are often used in the state-of-the-art algorithms predicting eye movements [Najemnik & Geisler, 2005; Itti & Baldi, 2006; Renninger et al., 2007]. Inspired by a process that selects relevant sources of the visual information, our adaptive feature selection scheme can also be seen as a visual search strategy underlying eye movements while performing a task. Therefore, further we investigate the next question, namely whether the proposed information-theoretical selection scheme, which is a computationally complex algorithm, is utilized by humans while they perform a visual classification task. For this, we constructed a psychophysical experiment where people had to select image parts that in their opinion are relevant for classification of these images. We present the analysis of behavioral data where we investigate whether human strategies of task-dependent selective attention can be explained by a simple scheme based on the pairwise mutual information, a more complex feature selection algorithm based on the conventional static conditional mutual information and the proposed here adaptive feature selector that mimics a mechanism of the iterative hypothesis refinement.

The main contribution of this work is the adaptive feature selection criterion based on the conditional mutual information, as well as its non-parametric estimation that does not presume any problem-specific assumptions. Moreover, it is shown that such adaptive selection strategy, being inspired by the attentional modulation of task-relevant parts of a visual scene, is indeed used by people while performing visual classification.

The thesis is organized in the following way. Chapter 2 reviews the conventional feature selection. Main approaches to dimensionality reduction in general and feature selection in particular are discussed in Section 2.1. Further, in Section 2.2, we introduce a general framework of sequential feature search which is used in Section 2.3 to present different selection criteria. Information-theoretical feature selection together with appropriate estimation techniques are reviewed in Section 2.3 and in Section 2.4, respectively.

Chapter 3 starts with the biological motivation and the general idea of the adaptive approach to feature selection, given in Section 3.1 and Section 3.2, respectively. Section 3.3 introduces a framework of adaptive feature selection, which is followed by Section 3.4 presenting a review on existing algorithms utilizing this approach to dimensionality reduction. After that, in Section 3.5, the proposed adaptive conditional mutual information feature selector is presented. In particular, Subsection 3.5.1 introduces the model, its estimation using kernel density method with the adaptive smoothing is described in Subsection 3.5.2. Results of practical investigations are provided in Chapter 4, where ability of ACMIFS to select relevant features in general and especially in high dimensions is examined. In addition, Section 4.3 presents comparison of ACMIFS with two static and adaptive feature selectors based on conditional mutual information, Parzen window feature selector [Kwak & Choi, 2002a] and active testing model [Geman & Jedynak, 1996]. Further, an alternative selection scheme combining ACMIFS and active testing model in

order to reduce computational complexity is proposed in Section 4.4. The discussion of advantages of adaptive feature selection is given in Section 4.5.

Chapter 5 presents the psychophysical experiment where human strategies of task-dependent selective attention are investigated. Section 5.1 reviews existing strategies of attentional selection with the emphasis on the task-dependent ones. Further, in Section 5.2, we describe an idea of the clicking experiment. Section 5.3 provides details of three tested information-theoretical strategies based on mutual information, static and adaptive conditional mutual information of a class with an image patch. Section 5.4 presents a statistical method that is used to compare these strategies with respect to their explanatory power of the observed behavioral data. Technical details of the experimental setup are described in Section 5.5. Section 5.6 presents the analysis and interpretation of the clicking experiments. Finally, the general discussion is provided in Chapter 6.

Chapter 2

Conventional feature selection

2.1 Main approaches to feature selection

Feature selection algorithms reduce dimensionality of the input space by picking a small number of relevant features from the initial feature set. As representatives of dimensionality reduction techniques, they are used to solve a so-called “curse of dimensionality” problem, meaning that there exists exponential dependence between the dimension of the input and the amount of data required to learn model parameters. Thus, decreasing the input dimensionality should ease a learning process. Moreover, a model with fewer parameters usually has better generalization ability.

Besides feature selection, there is another method of reducing dimensionality called feature extraction. This family of techniques performs transformation of the initial input space to the space of reduced dimensionality, which usually has also some desired properties like orthogonality or independence of new features etc. It is worth to mention that some feature extraction algorithms expand the initial input dimensionality in a way that a learning problem becomes simpler in the transformed space [Broomhead & Lowe, 1988; Simoncelli et al., 1992; Lewicki et al., 1998]. Since we are interested in selecting features and not in their transformations, we will speak further exclusively about feature selection algorithms. The review of feature extractors can be found for example in [Liu & Motoda, 1998; Guyon et al., 2006]. The most prominent representatives in pattern recognition are principal component analysis [Pearson, 1901; Jolliffe, 1986], independent component analysis [Hyvärinen et al., 2004] and sparse coding due to its biological plausibility and connection to receptive field properties of neurons in primary visual cortex [Foldiak, 1990; Olshausen & Field, 1996].

Feature selection algorithms can be divided into two main classes: filters and wrappers [Webb, 1999]. The filters look for a minimal subset of features that can maximally enhance classification, i. e. discriminate between samples belonging to different classes with the minimal error. In order to evaluate a discrimination power of a feature subset, different metrics are used such as various distance measures between classes, dependency measures between features and classes etc. Note that these metrics are not restricted to a particular classification method, therefore, selected features can be used for training any classifier. However, it can also be considered as a drawback, since the resulting feature subset may be suboptimal for the chosen classifier. An extensive overview of distance measures used by the filters will be presented later in Section 2.3.

Comparing to the filter methods, the wrappers select features that are useful for a certain classifier [Kohavi & John, 1997]. A goodness of a feature subset is measured by the prediction accuracy of a classifier employing these features. Thus, one can be sure that the selected features will indeed improve the quality of classification. However, there is a danger of overfitting. This means that the features are selected in the way to provide the best classification performance on the training data which might however lead to poor accuracy on the previously unseen test data. Moreover, these methods are rather computationally expensive, since in order to find the best feature subset a classifier should be run as many times as there are different subsets under consideration. Among representatives of wrappers, there are a recursive algorithm for support vector machines [Guyon et al., 2002], a wrapper feature selector for Bayesian networks [Singh & Provan, 1995], Kohavi's sequential feature selection for a general classifier [Kohavi & John, 1997] etc. Linear discriminant analysis can also be seen as a wrapper which performs feature selection while building a linear classifier on input features [Fisher, 1936]. Since the classifier itself is simple and the only made assumption is that data drawn from each class are normally distributed, this technique is often used as a filter [McLachlan, 2004].

There is another type of feature selectors called embedded methods [Duch, 2006]. They can be considered as a subtype of the wrappers because during the selection process they take also into account a classifier to be used. However, instead of directly employing results of classification, they rather use knowledge about the structure of a classifier while evaluating how good different feature subsets are. Thus, compared to the wrappers, the embedded methods are less computationally complex and less prone to overfitting. Examples of such methods are a feature selector for support vector machines, which minimizes a generalization bound [Weston et al., 2000], decision trees and artificial neural networks.

In each case, one can look for the best feature subset of a certain cardinality using an optimal search strategy which assumes evaluating all possible feature subsets and choosing the best one [Reunanen, 2006]. Since the number of such subsets is exponentially large, testing all of them is infeasible unless a number of the initial features is small. A good example of the optimal strategy avoiding an exhaustive search is the branch and bound

method [Narendra & Fukunaga, 1977; Yu & Yuan, 1993; Somol et al., 2004]. The key point here is monotonicity of a selection criterion. It means that if for two sets \mathcal{A} and \mathcal{B} , $\mathcal{A} \subset \mathcal{B}$, then the goodness of the set \mathcal{A} is not larger than the goodness of the set \mathcal{B} . Using such assumption together with backward selection, i. e. iterative elimination of features from the initial set, one can disregard some subsets on the intermediate iterations if their discriminability is low. For references to selection criteria which are monotonic and therefore can be used in combination with the branch and bound search, see the review on feature selection and extraction of Webb [Webb, 1999].

We would rather consider a general case and assume that a selection criterion does not satisfy the monotonicity assumption. Then, suboptimal methods have to be used. This class of algorithms includes feedforward and backward sequential feature selection which use the greedy search strategy [Webb, 1999; Jain & Zongker, 1997]. The feedforward algorithms start with an empty feature set and iteratively add features, which are relevant with respect to the features selected on the previous iterations [Whitney, 1971]. As was already mentioned above, the backward approach starts with the full feature set and on every iteration removes features that are the least useful in the current subset of the remained features [Marill & Green, 1963; Abe, 2005]. A popular backward method is the Markov blankets algorithm that sequentially removes irrelevant features. A feature F_k is said to be irrelevant if it has a Markov blanket, i. e. there exists such a feature subset \mathcal{F}' that if F_k is conditioned on this subset, then it is independent of all remaining features [Koller & Sahami, 1996; Tsamardinos et al., 2003].

The suboptimality of sequential methods comes from the fact that they do not explicitly look for the best feature subset. They rather try to find a feature or several features that can improve discriminability of the current subset as much as possible. The resulting feature subsets found by optimal and suboptimal approaches will differ a lot if there are high-order dependencies between features. Comparing feedforward and backward algorithms, the latter can theoretically show better results. Assuming that there are some complementary features which are informative only together and not alone, the feedforward methods would not choose any of these features at all, and therefore, they would not have a chance to evaluate the goodness of these features together. In the case of the backward methods, it is more likely that such features will be included in the final feature subset, because eliminating any of them from the feature set will result in decrease of its discrimination power. However, in practice, the backward methods are used less often. Feature subsets on early iterations are of large cardinality, which makes the evaluation of their relevance complicated. Moreover, practical studies have not shown that the backward approach produces always better feature subsets when compared to the feedforward approach [Aha & Bankert, 1996; Kudo & Sklansky, 2000]. Sequential floating feature selectors belong to the class of algorithms that assume alternation of feedforward and backward steps while searching for the informative feature subsets. Though, they have proven quite efficient, the applicability of floating search methods is limited due to their exponential complexity

[Pudil et al., 1994; Jain & Zongker, 1997]. For a review on various search methods for feature selection see [Reunanen, 2006; Somol et al., 2007].

A feature selection algorithm, which is proposed later in this thesis, is inspired by the hypothesis checking mechanism in the visual system. It iteratively selects small parts of a visual input for the detailed processing in order to refine hypotheses about objects present in the visual scene. Keeping a parallel to this mechanism of the visual system, we adopt a sequential feedforward approach to feature selection. Further, as a task we consider image classification, therefore, all feature selection techniques will be presented in connection to classification. Hence, a model describing data will refer to an abstract classifier. For completeness, we name some examples of features selection methods for regression. These are regression trees [Breiman et al., 1984], regularization schemes [Tibshirani, 1996; Zou & Hastie, 2005] and various filters using correlation or entropy between features and a dependent variable as a measure of the feature relevance, e. g. [Hocking, 1976; Carmona et al., 2011].

2.2 Feature selection framework

2.2.1 Classification setup

Let us introduce a standard classification setup and a conventional scheme of feature selection within this setup.

Suppose we have a space of possible inputs $\mathcal{F} = \times_{i=1}^n \mathcal{F}_i$, i. e. each input is an n -dimensional feature vector $\mathbf{f} = (f_1, \dots, f_n)$, where the i^{th} feature takes values $f_i \in \mathcal{F}_i$. Our notion of feature is rather general. For example, for the image classification task, features can be quite simple, such as gray-values of certain pixels, or more sophisticated, such as frequencies of some objects on an image. Feature combinations are considered as a random variable F with a joint distribution on $\mathcal{F}_1 \times \dots \times \mathcal{F}_n$ and the observation \mathbf{f} is drawn from that distribution.

Furthermore, each observation has an associated class label $c \in \mathcal{C} = \{c_1, \dots, c_m\}$. The task of the classifier is to assign a class label to each observation \mathbf{f} . Thus, formally it is considered as a map $\phi : \mathcal{F} \rightarrow \mathcal{C}$ or, more generally, as assigning to each \mathbf{f} the conditional probabilities $p(c|\mathbf{f})$ of the classes c . To learn such classification, we are given a training set $\mathcal{X} = \{(\mathbf{x}_i, c_i)\}_{i=1}^T$ of labeled observations, which are assumed to be drawn independently from the distribution relating feature vectors and class labels. Then, the goal is to find a classification rule ϕ that correctly predicts the class of future samples with unknown class label, called testing samples. That is, confronted with a feature vector ξ we would classify

it as $c = \phi(\xi)$. Feature selection then means that for this particular task only a subset of features rather than the full feature vector is used.

2.2.2 General framework of feedforward selection

According to the conventional sequential feedforward feature selection for classification, a feature $F_{\alpha_{i+1}}$ selected on the $(i+1)^{th}$ step should maximize some selection criterion S , i. e.:

$$\alpha_{i+1} = \arg \max_k S(C, F_{\alpha_1}, \dots, F_{\alpha_i}, F_k), \quad F_k \in \{F_1, \dots, F_n\} \setminus \{F_{\alpha_1}, \dots, F_{\alpha_i}\}, \quad (2.1)$$

where $F_{\alpha_1}, \dots, F_{\alpha_i}$ is a subset of the features selected before the $(i+1)^{th}$ iteration. Intuitively, the selection criterion S should favor such features that are **relevant for classification** with respect to the variable C . At the same time, it is desirable that the final feature subset is of the minimal size, therefore, the selected features should be maximally **non-redundant with respect to each other**.

Let us formalize the concepts of relevance and redundancy. Suppose we are given an unlabeled sample. Before any feature is observed, we are completely uncertain about a class label of this sample. Let $U(C)$ be some measure of uncertainty about the variable C . Then, a feature F_k is said to be relevant for classification if given this feature the uncertainty about the class of some hypothetical sample will be reduced, i. e. $U(C) > U(C|F_k)$. Note that feature selection is performed before classification and therefore the selected features should be discriminative for any sample that we would have to classify in the future.

Suppose that we have already selected i features and let \mathbf{F}^i denote a subset of these features, $\mathbf{F}^i = \{F_{\alpha_1}, \dots, F_{\alpha_i}\}$. At this stage, the current uncertainty about the class can be expressed as $U(C|\mathbf{F}^i)$. Then, the feature F_k is both relevant for classification and non-redundant w. r. t. the already selected features if knowing this feature the current uncertainty about the class will be reduced: $U(C|\mathbf{F}^i) > U(C|F_k, \mathbf{F}^i)$. Thus, the criterion for selecting a feature on the iteration $(i+1)$ can be formulated in the following way:

$$\alpha_{i+1} = \arg \max_k S(C, \mathbf{F}^i, F_k) = \arg \max_k \{U(C|\mathbf{F}^i) - U(C|F_k, \mathbf{F}^i)\}. \quad (2.2)$$

In addition to a search strategy, a key issue in feature selection is the choice of a selection criterion, which in the framework of uncertainty reduction corresponds to the choice of the uncertainty function $U(\cdot)$. Breiman and coauthors, while working on decision trees, which can also be considered as feature selectors, developed a set of desired properties for uncertainty functions and proposed several examples satisfying these properties [Breiman et al., 1984].

Denoting a probability of the class c_j after the i^{th} iteration as $p(c_j|\mathbf{f}^i)$, the uncertainty $U(C|\mathbf{F}^i)$ is defined as a nonnegative function which depends on $p(c_1|\mathbf{f}^i), \dots, p(c_m|\mathbf{f}^i)$. In our notation, \mathbf{f}^i stands for a vector of particular realizations of the selected features $\mathbf{f}^i = \{F_{\alpha_1} = f_{\alpha_1}, \dots, F_{\alpha_i} = f_{\alpha_i}\}$. Then, $U(C|\mathbf{F}^i)$ should have the following properties [Breiman et al., 1984]:

1. $U(C|\mathbf{F}^i) = \max$, if all classes are equiprobable, i. e. $p(c_j|\mathbf{f}^i) = p(c_{j'}|\mathbf{f}^i), \forall j, j' = 1, \dots, m$.
2. $U(C|\mathbf{F}^i) = \min$, if all samples belong to one class, i. e. $p(c_j|\mathbf{f}^i) = 1$ and $p(c_{j'}|\mathbf{f}^i) = 0, \forall j' \neq j$.
3. $U(C|\mathbf{F}^i)$ is symmetric in $p(c_1|\mathbf{f}^i), \dots, p(c_m|\mathbf{f}^i)$.

2.3 Selection criteria

Here, we present various uncertainty functions which satisfy the above stated properties and are widely used for feature selection. The review is given with respect to the presented setup of the sequential feedforward feature selection for pattern classification.

2.3.1 Misclassification error

While solving a classification problem, the goal is to build a classifier with the best accuracy. So intuitively an uncertainty function should depend on the misclassification error. Following Breiman and coauthors, let us define the misclassification error for i selected features with the 0-1 loss function [Breiman et al., 1984]:

$$U(C|\mathbf{F}^i) = \sum_{\mathcal{F}^i} p(\mathbf{f}^i) \left(1 - \max_j p(c_j|\mathbf{f}^i) \right), \quad (2.3)$$

where $\sum_{\mathcal{F}^i}$ stands for $\sum_{\mathcal{F}_{\alpha_1}} \dots \sum_{\mathcal{F}_{\alpha_i}}$ for brevity. Note that the expression (2.3) is in fact the Bayes error probability, which is the lowest possible error probability for a given classification problem:

$$\begin{aligned} \sum_{\mathcal{F}^i} p(\mathbf{f}^i) \left(1 - \max_j p(c_j|\mathbf{f}^i) \right) &= \sum_{\mathcal{F}^i} p(\mathbf{f}^i) \left(\sum_{j=1, j \neq j'}^m p(c_j|\mathbf{f}^i) \right) = \\ &= \sum_{\mathcal{F}^i} \sum_{j=1, j \neq j'}^m p(c_j) p(\mathbf{f}^i|c_j), \end{aligned} \quad (2.4)$$

where $c_{j'}$ is the winning class, i. e. $p(c_{j'}|\mathbf{f}^i) = \max_j p(c_j|\mathbf{f}^i)$.

Using the misclassification error as an uncertainty function, the corresponding feature selection criterion has the following form:

$$\begin{aligned} \alpha_{i+1} = \arg \max_k \{ & U(C|\mathbf{F}^i) - U(C|F_k, \mathbf{F}^i) \} = \\ \arg \max_k \{ & - \sum_{\mathcal{F}^i} p(\mathbf{f}^i) \max_j p(c_j|\mathbf{f}^i) + \sum_{\mathcal{F}^i} \sum_{\mathcal{F}_k} p(f_k, \mathbf{f}^i) \max_j p(c_j|f_k, \mathbf{f}^i) \}. \end{aligned} \quad (2.5)$$

This selection criterion is obviously useful for selecting features that can discriminate well different classes. Since in practice neither an ideal classification rule nor posterior distributions $p(c_j|f_k, \mathbf{f}^i)$ are known, different approximations should be used. For example, one can employ nonparametric techniques of density estimation such as the kernel density method for estimating class-conditional pdfs $p(f_k, \mathbf{f}^i|c_j)$ and then apply the Bayes rule to obtain the posteriors [Fukunaga & Hummels, 1987; Yang & Hu, 2012]. k -nearest neighbor method is also used to estimate a selection criterion based on the Bayes error probability by margin-based feature selection algorithms such as Relief, which try to weight available features in a way so that a margin between classes is maximal [Kira & Rendell, 1992; Gilad-Bachrach et al., 2004; Sun, 2007; Yang & Hu, 2012]. Another approach to the estimation problem is introducing simplifying assumptions about the involved pdfs such as being Gaussian etc [Bruzzone & Serpico, 1998].

Despite its simplicity and intuitive usefulness for solving classification problems, the selection criterion based on the misclassification error has a major disadvantage as an uncertainty function. It does not explicitly favor situations where the posterior of some classes approaches 0 or 1, which happens due to the linear dependence between the uncertainty and $(\max_j p(c_j|f_k, \mathbf{f}^i))$.

Let us consider a two-class problem. In this case, the uncertainty function based on the misclassification error (2.3) is the following:

$$U(C|\mathbf{F}^i) = \sum_{\mathcal{F}^i} p(\mathbf{f}^i) (1 - \max\{p(c_1|\mathbf{f}^i), p(c_2|\mathbf{f}^i)\}) = \sum_{\mathcal{F}^i} p(\mathbf{f}^i) \min\{p(c_1|\mathbf{f}^i), p(c_2|\mathbf{f}^i)\}. \quad (2.6)$$

Keeping in mind that $U(C|\mathbf{F}^i) = \sum_{\mathcal{F}^i} p(\mathbf{f}^i) U(C|\mathbf{f}^i)$, Figure 2.1 depicts $U(C|\mathbf{f}^i)$ as a function of $p(c_1|\mathbf{f}^i)$, illustrating its linear behavior. Therefore, as the class posterior distribution becomes less uniform, i. e. when $p(c_1|\mathbf{f}^i)$ decreases on the interval $[0, 0.5)$ or correspondingly increases on the interval $(0.5, 1]$, the function $U(C|\mathbf{F}^i)$ decreases just linearly.

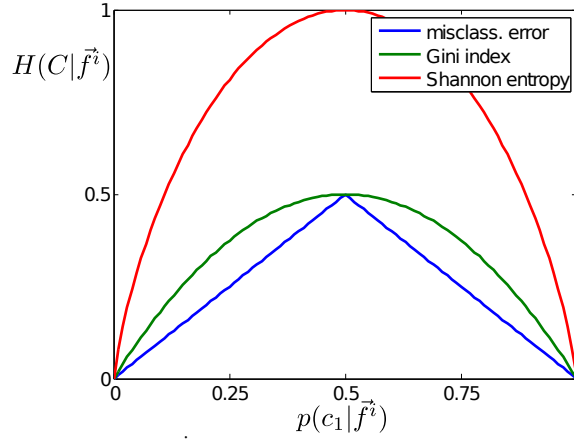


Figure 2.1: Uncertainty function $U(C|\mathbf{f}^i)$ based on misclassification error, Gini index and Shannon entropy plotted against $p(c_1|\mathbf{f}^i)$ for a two-class problem.

2.3.2 Gini index

Following further the approach of Breiman and coauthors, we introduce a family of uncertainty functions so that resulting selection criteria give more weight to less uniform class posterior distributions, i. e. when $p(c_j|\cdot) = 1$ or $p(c_j|\cdot) = 0$.

For this, the desired property for $U(C|\mathbf{F}^i)$ would be to decrease faster than linearly. This can be ensured if $U(C|\mathbf{F}^i)$ is strictly concave. So for $U(C|\mathbf{F}^i)$, which is continuous on the interval $[0, 1]$, and $p(c_1|\mathbf{f}^i) \in [0, 1]$, the second derivative of the function should be negative, $U''(C|\mathbf{F}^i) < 0$.

Let us proceed with construction of the improved uncertainty function. Recalling three general requirements, we rewrite them for the two class problem. Since $p(c_2|\mathbf{f}^i) = 1 - p(c_1|\mathbf{f}^i)$, we can consider that $U(C|\mathbf{F}^i)$ depends only on $p(c_1|\mathbf{f}^i)$:

1. $U(C|\mathbf{F}^i) = \max$, if $p(c_1|\mathbf{f}^i) = 0.5$.
2. $U(C|\mathbf{F}^i) = \min$, if $p(c_1|\mathbf{f}^i) = 1$ or $p(c_1|\mathbf{f}^i) = 0$. Without loss of generality (w.l.o.g.) we can require that the minimum value of $U(p(c_1|\mathbf{f}^i))$ is 0.
3. $U(C|\mathbf{F}^i)$ is symmetric, i. e. $U(c_1|\mathbf{f}^i) = U(c_2|\mathbf{f}^i)$.

And we add the new requirement

4. $U''(C|\mathbf{F}^i) < 0$.

The simplest example of the concave function is a quadratic polynomial, which gives $U(C|\mathbf{f}^i) = ap(c_1|\mathbf{f}^i)^2 + bp(c_1|\mathbf{f}^i) + c$.

The second and the forth requirements give $c = 0$, $a + b = 0$ and $a < 0$, respectively. Assuming w.l.o.g. that $a = -2$, the uncertainty function is of the form:

$$\begin{aligned} U(C|\mathbf{f}^i) &= \sum_{\mathcal{F}^i} p(\mathbf{f}^i) (2(-p(c_1|\mathbf{f}^i)^2 + p(c_1|\mathbf{f}^i))) = \\ &= \sum_{\mathcal{F}^i} 2p(\mathbf{f}^i) (-p(c_1|\mathbf{f}^i)(1 - p(c_2|\mathbf{f}^i)) + p(c_1|\mathbf{f}^i)) = \sum_{\mathcal{F}^i} p(\mathbf{f}^i) (2p(c_1|\mathbf{f}^i)p(c_2|\mathbf{f}^i)), \end{aligned} \quad (2.7)$$

which is known as the Gini index, a measure of statistical dispersion proposed by Corrado Gini [Gini, 1912]. The general form of the Gini index for a multiclass problem has the following form:

$$U(C|\mathbf{f}^i) = \sum_{\mathcal{F}^i} p(\mathbf{f}^i) \sum_{j=1}^m \sum_{j'=1, j \neq j'}^m p(c_j|\mathbf{f}^i)p(c_{j'}|\mathbf{f}^i) = \sum_{\mathcal{F}^i} p(\mathbf{f}^i) \left(1 - \sum_{j=1}^m p(c_j|\mathbf{f}^i)^2 \right). \quad (2.8)$$

Due to simplicity of its estimation and correspondence to the desired properties of the uncertainty function (see Figure 2.1), this criterion is widely used in decision tree construction [Breiman et al., 1984] and [Gelfand et al., 1991].

There is a modified version of the Gini index which is widely used for feature selection in the field of text classification [Shang et al., 2007; Yang et al., 2011]. Let us rewrite the feature selection criterion in the following way:

$$\begin{aligned} k &= \arg \max_k \{U(C|\mathbf{F}^i) - U(C|F_k, \mathbf{F}^i)\} = \arg \min_k \{U(C|F_k, \mathbf{F}^i)\} = \\ &= \arg \min_k \left\{ p(f_k, \mathbf{f}^i) \left(1 - \sum_{j=1}^m p(c_j|f_k, \mathbf{f}^i)^2 \right) \right\}. \end{aligned} \quad (2.9)$$

First, the criterion 2.9 is simplified to $\arg \max_k \left\{ p(f_k, \mathbf{f}^i) \sum_{j=1}^m p(c_j|f_k, \mathbf{f}^i)^2 \right\}$, where $p(f_k, \mathbf{f}^i)$ is further replaced with $p(f_k, \mathbf{f}^i|c_j)^2$:

$$k = \arg \max_k \left\{ \sum_{j=1}^m p(f_k, \mathbf{f}^i|c_j)^2 p(c_j|f_k, \mathbf{f}^i)^2 \right\}. \quad (2.10)$$

The replacement is done in order to favor more class-specific features, i. e. features with low marginal probabilities but high class-conditional probabilities for some c_j . This is especially useful when classes are unbalanced.

2.3.3 Shannon entropy

In terms of information theory, a measure of uncertainty about the outcome of a random variable is Shannon entropy [Shannon & Weaver, 1949]. For a random variable X , it is defined by the following expression:

$$H(X) = - \sum_{\mathcal{X}} p(x) \log p(x). \quad (2.11)$$

The uncertainty about the class label after selecting certain features $F_{\alpha_1}, \dots, F_{\alpha_i}$ is measured by the conditional entropy:

$$U(C|\mathbf{F}^i) = H(C|\mathbf{F}^i) = - \sum_{\mathcal{F}^i} \sum_{j=1}^m p(c_j, \mathbf{f}^i) \log p(c_j|\mathbf{f}^i). \quad (2.12)$$

Shannon entropy has all four desired properties of the uncertainty function: it takes its maximum when the class-conditional distribution is uniform, it equals zero when the posterior of one of the classes is 1, and it is strictly concave: $U''(C|\mathbf{F}^i) = - \sum_{\mathcal{F}^i} p(\mathbf{f}^i) \sum_{j=1}^m \frac{1}{p(c_j|\mathbf{f}^i)} < 0$. This can be seen on Figure 2.1 plotting Shannon entropy $H(C|\mathbf{f}^i)$ against $p(c_1|\mathbf{f}^i)$ for a two-class problem.

Rewriting the selection criterion with Shannon entropy as the uncertainty function, we obtain:

$$S(C, \mathbf{F}^i, F_k) = U(C|\mathbf{F}^i) - U(C|F_k, \mathbf{F}^i) = H(C|\mathbf{F}^i) - H(C|F_k, \mathbf{F}^i) = I(C; F_k|\mathbf{F}^i), \quad (2.13)$$

where $I(C; F_k|\mathbf{F}^i)$ is the mutual information between the class variable C and the feature F_k after selecting i features \mathbf{F}^i . It tells how much one can learn about C after observing an outcome of the feature F_k , which is usually measured in bits, or equivalently how certain one can classify the samples after selecting F_k given already selected features F_1, \dots, F_i .

Using the definition of mutual information, the selection criterion can be further rewritten in the following way:

$$S(C, \mathbf{F}^i, F_k) = I(C; F_k|\mathbf{F}^i) = \sum_{\mathcal{F}^i} \sum_{\mathcal{F}_k} \sum_{j=1}^m p(c_j, f_k, \mathbf{f}^i) \log \frac{p(c_j, f_k|\mathbf{f}^i)}{p(c_j|\mathbf{f}^i)p(f_k|\mathbf{f}^i)}, \quad (2.14)$$

which can be interpreted as the expected value of the logarithmic function of $\frac{p(c_j, f_k|\mathbf{f}^i)}{p(c_j|\mathbf{f}^i)p(f_k|\mathbf{f}^i)}$, a degree of correlation between the variables C and F_k once the features \mathbf{F}^i are selected.

Mutual information as a selection criterion is very often used in feature selection algorithms as it provides a natural measure of interdependence between features and the class,

e. g. [Lewis, 1962; Quinlan, 1986; Battiti, 1994; Kwak & Choi, 2002a; Fleuret & Guyon, 2004; Peng et al., 2005], and it is invariant under invertible transformations of involved variables [Rezā, 1961; Kraskov et al., 2004]. The major difficulties in using mutual information concern its estimation. As the feature selection criterion proposed in this thesis is based on Shannon entropy or analogously on the mutual information, a detailed discussion of entropy-related issues will be presented later.

2.3.4 Gain ratio

The selection criterion based on mutual information has a practical problem when features are discrete. There is a bias in selection towards features with many values. The reason is the following. Since the mutual information is a symmetric measure, then $I(C; F_k | \mathbf{F}^i) = H(F_k | \mathbf{F}^i) - H(F_k | C, \mathbf{F}^i)$. As a result, a feature with many values has a high entropy $H(F_k | \mathbf{F}^i)$ that could lead to a high value of the mutual information.

An intuitive solution to this problem is to punish features with high entropy values. Thus, the proposed modified uncertainty function, which is in the decision tree community called “Gain ratio” [Quinlan, 1993], is a normalized form of (2.13), the selection criterion using Shannon entropy as the uncertainty function:

$$S(C, \mathbf{F}^i, F_k) = \frac{U(C | \mathbf{F}^i) - U(C | \mathbf{F}^i, F_k)}{H(F_k | \mathbf{F}^i)} = \frac{H(C | \mathbf{F}^i) - H(C | \mathbf{F}^i, F_k)}{H(F_k | \mathbf{F}^i)} = \frac{I(C; F_k | \mathbf{F}^i)}{H(F_k | \mathbf{F}^i)}. \quad (2.15)$$

This expression measures the ratio between the informativeness of the feature F_k for classification and its entropy. However, the measure becomes unstable once the entropy of some features starts approaching zero, i. e. when the feature has only one or very few values. Experiments using the gain ratio show that features with high entropy are punished too much and almost never chosen [Mingers, 1987]. At the same time, a similar idea has been successfully used for normalizing the mutual information between two features $I(F_i, F_j)$ as a component of the selection criterion (2.13) [Estevez et al., 2009]:

$$k = \arg \max_k \left\{ I(C; F_k) - \frac{1}{i} \sum_{q=1}^i \frac{I(F_k; F_q)}{\min\{H(F_k), H(F_q)\}} \right\}. \quad (2.16)$$

Another example is the symmetrical relevance criterion which uses the joint entropy $H(C, F_{\alpha_q}, F_k)$ as a normalization factor for the multivariate mutual information $I(C; F_{\alpha_q} F_k)$ [Meyer & Bontempi, 2006]:

$$k = \arg \max_k \left\{ \sum_{q=1}^i \frac{I(C; F_{\alpha_q} F_k)}{H(C, F_{\alpha_q}, F_k)} \right\}. \quad (2.17)$$

Both above-mentioned criteria are approximations of 2.13. A detailed overview of various approximations of entropy-based selection criteria will be presented further in Section 2.6. For a review on information-theoretical selection criteria using different normalization techniques see [Duch, 2006].

2.3.5 Alpha-entropies

Shannon entropy assumes a certain trade-off between contributions from the main mass and tails of a distribution and events that occur too often or too rare do not influence much the entropy. So-called α -entropies, Rényi entropy [Rényi, 1961] and Tsallis entropy [Tsallis, 1988], are generalized versions of Shannon entropy that give a possibility to control this trade-off explicitly. The α -entropy of some variable X is a function of $\sum_X p(x)^\alpha$, where the parameter α corresponds to the degree of inhomogeneity in the structure of the probability distribution of X [Holste et al., 1998]. That is, α controls a contribution of events of different frequencies to the sum, i. e. for large α only the high-frequency events contribute, whereas for small α all events are weighted more uniformly. Due to this flexibility, α -entropies are widely used to describe behavior of complex systems in such fields like statistical thermodynamics, e. g. [Ramshaw, 1995], nonlinear dynamical systems, e. g. [Grassberger & Procaccia, 1983], evolutionary programming, e. g. [Stariolo & Tsallis, 1996] etc.

α -entropies are also used as uncertainty functions in feature selection. For the α -entropy $H_\alpha(C|\mathbf{F}^i)$, the parameter α allows to control a trade-off between the purity of the class posterior distribution, which is considered to be the best criterion to optimize from the Bayesian viewpoint [Duch, 2006], and reducing the average uncertainty. The less uniform posteriors, i. e. when the posterior of one of the classes is around 1, can be achieved with $\alpha \rightarrow 0$. In this case, all events contribute to the entropy and it will be significantly reduced only when $p(c_j|\mathbf{f}^i) \rightarrow 1$. Both Rényi and Tsallis entropies are identical to Shannon entropy for $\alpha \rightarrow 1$. Let us take a closer look at these parameterized entropies.

The uncertainty function $U(C|\mathbf{F}^i)$ using the Rényi entropy is

$$H_R(C|\mathbf{F}^i) = \sum_{\mathcal{F}^i} \frac{p(\mathbf{f}^i)}{1-\alpha} \log \left(\sum_{j=1}^m p(c_j|\mathbf{f}^i)^\alpha \right), \quad \alpha > 0, \alpha \neq 1, \quad (2.18)$$

leading to the following feature selection criterion (note its resemblance to mutual information):

$$S(C, \mathbf{F}^i, F_k) = H_R(C|\mathbf{F}^i) - H_R(C|\mathbf{F}^i, F_k) = \sum_{\mathcal{F}^i} \sum_{\mathcal{F}_k} \frac{p(\mathbf{f}^i, f_k)}{1-\alpha} \log \left(\frac{\sum_{j=1}^m p(c_j|\mathbf{f}^i)^\alpha}{\sum_{j=1}^m p(c_j|\mathbf{f}^i, f_k)^\alpha} \right) \quad (2.19)$$

Rényi entropy has the first three desired properties of the uncertainty function but, in contrast to the concave Shannon entropy, it is concave only for $\alpha \in (0, 1)$ and neither concave nor convex for $\alpha > 1$. Figure 2.2 illustrates the behavior of the quantity $H_R(C|\mathbf{f}^i)$ as a function of $p(c_1|\mathbf{f}^i)$ for different values of α for a two-class problem. One can see that for small α the decrease in entropy will be significant only if $p(c_1|\mathbf{f}^i) \rightarrow 1$ or $p(c_1|\mathbf{f}^i) \rightarrow 0$. However, for large α a slight move away from the uniform posterior, i. e. away from $p(c_1|\mathbf{f}^i) = 0.5$, causes noticeable entropy reduction.

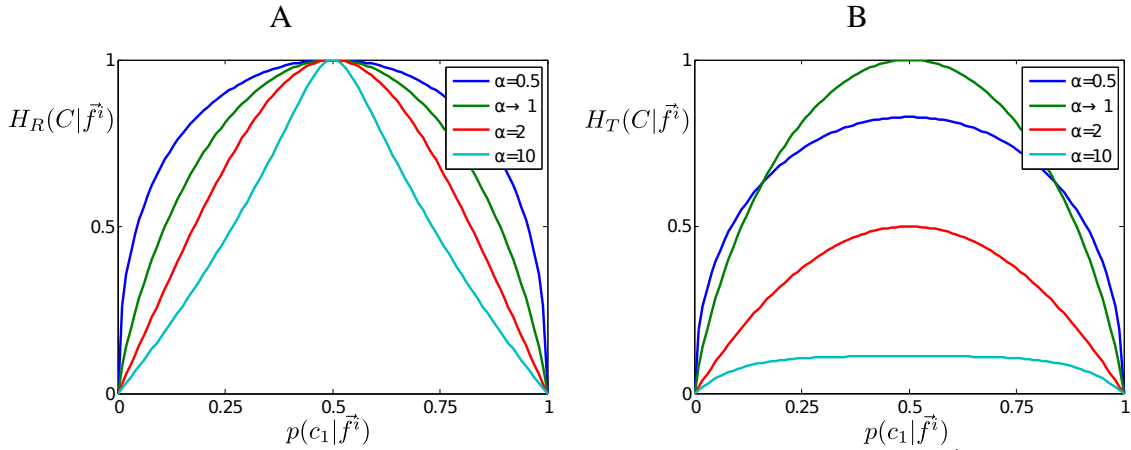


Figure 2.2: A: Rényi entropy as the uncertainty function plotted against $p(c_1|\mathbf{f}^i)$ for different values of α for a two-class problem. B: Tsallis entropy as the uncertainty function plotted against $p(c_1|\mathbf{f}^i)$ for different values of α for a two-class problem.

Tsallis entropy is a generalization of Boltzmann-Gibbs entropy from statistical mechanics and for the distribution $p(c|\mathbf{f}^i)$ it is defined as follows:

$$H_T(C|\mathbf{f}^i) = - \sum_{j=1}^m \frac{p(c_j|\mathbf{f}^i)}{1-\alpha} \left(1 - \sum_{j=1}^m p(c_j|\mathbf{f}^i)^\alpha \right), \quad \alpha > 0, \alpha \neq 1. \quad (2.20)$$

As Tsallis entropy is always concave, it satisfies all four desired features of the uncertainty function. However, compared to Shannon entropy, it is nonadditive. Additivity is one of the algebraic properties of the uncertainty measure requiring the joint entropy of two independent events to be a sum of their marginal entropies, i. e. $H(X, Y) = H(X) + H(Y)$ [Aczél & Daróczy, 1975]. For Tsallis entropy, we have:

$$H_T(X, Y) = H_T(X) + H_T(Y) + (1 - \alpha)H_T(X)H_T(Y), \quad (2.21)$$

where $(1 - \alpha)$ indicates a degree of deviation from an additive system. Nonadditivity can be useful if the system is known to have nonlinear long-range couplings between its elements [Caruso & Tsallis, 2008].

Figure 2.2 plots Tsallis entropy $H_T(C|\mathbf{f}^i)$ against $p(c_1|\mathbf{f}^i)$ for different values of α for a two-class problem. For small α , as in case of Rényi entropy, H_T vanishes only when $p(c_j|\mathbf{f}^i) \rightarrow 1$. For large values of α , the Tsallis entropy of the system does not change much by moving away from the uniform posterior. Such immunity to small changes in the class conditional probability distribution can help to achieve better generalization and robustness against noise.

Using Tsallis entropy as the uncertainty function, the feature selection criterion has the following form

$$S(C, \mathbf{F}^i, F_k) = H_T(C|\mathbf{F}^i) - H_T(C|\mathbf{F}^i, F_k) = \sum_{\mathcal{F}^i} \sum_{\mathcal{F}_k} \frac{p(\mathbf{f}^i, f_k)}{1 - \alpha} \sum_{j=1}^m (p(c_j|\mathbf{f}^i)^\alpha - p(c_j|\mathbf{f}^i, f_k)^\alpha). \quad (2.22)$$

As uncertainty functions in decision tree construction, Rényi and Tsallis entropies were used for example by [Maszczyk & Duch, 2008; Lima et al., 2010]. It was reported that due to the possibility to adjust to a structure of the probability distribution describing a given problem, constructed trees are usually smaller and have better performance than trees using the Shannon entropy. Moreover, this class of entropies is attractive for general feature selection due to the reduced computational complexity compared to the Shannon entropy [Liu & Hu, 2009; Lopes et al., 2009].

2.3.6 Correlation-based feature selection

The list of the selection criteria presented above is not exhaustive. Among others, there are criteria that do not formally fit in the framework of uncertainty reduction. The first class of such approaches are based on dependency measures between variables.

As an alternative to the selection criterion based on mutual information (2.13), it was suggested to use the χ^2 -statistic instead [Hart, 1985]. Similarly to mutual information, χ^2 -statistic measures a degree of dependence between a class and discrete features. However, this measure is usually used only for ranking, which assumes evaluating relevance of each feature alone and not together with other features. As a result, features that are relevant only in combination with other features will not be selected and a resulting feature subset will be likely redundant.

The χ^2 -statistic for a feature F_k after selecting i features is defined in the following way:

$$S(C, F_k, \mathbf{F}^i) = \chi^2(C, F_k) = \sum_{l=1}^r \sum_{j=1}^m \frac{(T_{lj} - E_{lj})^2}{E_{lj}}, \quad F_k \in \mathcal{F} \setminus \{F_1, \dots, F_i\}, \quad (2.23)$$

where r is a number of the discrete values of the feature F_k ; T_{lj} is a number of training samples belonging to the class c_j and feature F_k taking the value f_l . $E_{lj} = \frac{T_l T_j}{T}$ where T_l is a number of samples with F_k equals f_l , T_j is a number of samples belonging to the class c_j and finally T stands for a total number of the samples.

Using the determined χ^2 -statistic and the degrees of freedom, which is $(m-1)(r-1)$ in our case, one can define the p -value, the level of confidence about the feature F_k being uninformative for a class variable C . Thus, the lower this value is, the more dependent is the class on the feature under consideration. Note that this statistic naturally avoids the problem of being biased towards features with many values because a number of discrete values of the feature candidate F_k is taken into account via the degrees of freedom. Moreover, a level of confidence about the class-feature independence is more intuitive to interpret than a level of uncertainty, thus one can easily stop selecting features once the confidence level exceeds a certain threshold.

On the negative side, the main problems of χ^2 -statistic that were discovered are unreliable estimates due to noise and limited amount of training data [Mingers, 1987]. Further, as was already mentioned, χ^2 -statistic is applicable only to discrete variables. In addition, as χ^2 -statistic measures the pairwise association between a class and a single feature, it does not capture high-order dependencies between features.

Another feature selection criterion based on dependency measures is a Pearson correlation coefficient. It favors features that are highly positively or negatively correlated with the class [Duch, 2006]:

$$S(C, F_k, \mathbf{F}^i) = r(C, F_k) = \frac{\sum_{j=1}^T (f_{k,j} - \bar{f}_k)(c_j - \bar{c})}{\sqrt{\sum_{j=1}^T (f_{k,j} - \bar{f}_k)^2 \sum_{j=1}^T (c_j - \bar{c})^2}}, \quad (2.24)$$

where $f_{k,j}$ and c_j are the values of the feature F_k and the class variable on the j^{th} training sample, respectively, and \bar{f}_k and \bar{c} are the expectation values of the corresponding variable. Like the χ^2 -statistics, the correlation is usually measured between a single feature and a class. Therefore, this method is used for ranking features rather than for selecting a small informative subset of them. As an alternative to the Pearson correlation coefficient, other criteria such as Fisher score [Furey et al., 2000], Kolmogorov-Smirnov test or G-statistics can be used [Press et al., 1988; Duch, 2006; Miyahara & Pazzani, 2000]. Despite the fact that the ranking approach does not take into account high-order dependencies between features, it can still be useful for dimensionality reduction purposes. For example, it was shown during the NIPS feature selection challenge [Guyon et al., 2004]. Therefore, such techniques remain popular due to their simplicity.

A more advanced feature selection criterion was proposed by Hall [Hall, 1999]. It looks for a subset of features that are individually correlated with the class and at the same time minimally pairwise correlated with each other:

$$S(C, F_k, \mathbf{F}^i) = \frac{(i+1)\bar{r}_{ff}}{\sqrt{i+1+i(i+1)\bar{r}_{cf}}}, \quad (2.25)$$

where $(i+1)$ indicates a number of feature in the considered feature subset and \bar{r}_{ff} and \bar{r}_{cf} are the average Pearson correlation coefficients between two features and between a class and a feature, respectively. Comparing to the χ^2 -statistic, Hall's CFS measures redundancy between the features, however only pairwise. At the same time, an attractive advantage of this selection criterion is that it can be easily applied to both classification and regression problems. Note also that in contrast to mutual information, correlation-based techniques are able to find only linear dependencies between variables.

2.3.7 Probabilistic distance measures

There is a class of feature selection criteria that utilize probabilistic distance measures in order to select features which have the most distinct, i. e. minimally overlapping, class-conditional distributions. Although this approach as well does not fit into the framework of uncertainty reduction, we present it here for completeness.

Together with selection criteria based on the misclassification error, selection algorithms using probabilistic distance measures are representatives of discriminative methods that try to separate classes directly rather than model data. This approach can be a better solution while building a classifier, especially if the amount of data is limited [Vapnik, 1998].

Considering a feature candidate F_k after i selection iterations, let us denote a distance between two class-conditional distributions $p(f_k|c_j, \mathbf{f}^i)$ and $p(f_k|c_{j'}, \mathbf{f}^i)$ as $d_k^i(c_j, c_{j'})$. For a two-class problem, a selection criterion is defined simply as maximization of this distance, i. e.

$$\alpha_{i+1} = \arg \max_k S(C, \mathbf{F}^i, F_k) = \arg \max_k d_k^i(c_1, c_2), \quad (2.26)$$

while for multiclass problems there are several ways of combining the pairwise distances [Webb, 1999], for example:

$$S(C, \mathbf{F}^i, F_k) = \max_{j \neq j'} d_k^i(c_j, c_{j'}), \quad (2.27)$$

or

$$S(C, \mathbf{F}^i, F_k) = \sum_j \sum_{j'} d_k^i(c_j, c_{j'}) p(c_j | \mathbf{f}^i) p(c_{j'} | \mathbf{f}^i). \quad (2.28)$$

Similar to the notion of the uncertainty function, a distance between two distributions should satisfy certain requirements:

- 1) $d_k^i(c_j, c_{j'}) = 0$ if the corresponding pdfs are identical, $p(f_k|c_j, \mathbf{f}^i) = p(f_k|c_{j'}, \mathbf{f}^i)$;
- 2) $d_k^i(c_j, c_{j'}) \geq 0$;
- 3) $d_k^i(c_j, c_{j'}) = \max$ when the corresponding pdfs have disjoint support.

This list can be extended by the requirement of symmetry, $d_k^i(c_j, c_{j'}) = d_k^i(c_{j'}, c_j)$, which together with 1) and 2) are three standard necessary conditions for a function to be a distance metric.

We provide some examples of probabilistic distance measures used in feature selection. A more comprehensive list can be found for example here [Chen, 1976]:

- Kolmogorov variational distance:

$$d_k^i(c_j, c_{j'}) = \int_{\mathcal{F}^i} \int_{\mathcal{F}_k} |p(f_k|c_j, \mathbf{f}^i)p(c_j, \mathbf{f}^i) - p(f_k|c_{j'}, \mathbf{f}^i)p(c_{j'}, \mathbf{f}^i)| df_k d\mathbf{f}^i = \int_{\mathcal{F}^i} \int_{\mathcal{F}_k} |p(c_j|f_k, \mathbf{f}^i) - p(c_{j'}|f_k, \mathbf{f}^i)| p(f_k|\mathbf{f}^i) df_k d\mathbf{f}^i, \quad (2.29)$$

which for a two-class problem has a direct relation to the Bayes error probability:

$$p(e|f_k, \mathbf{f}^i) = 0.5 \left\{ 1 - \int_{\mathcal{F}^i} \int_{\mathcal{F}_k} |p(c_1|f_k, \mathbf{f}^i) - p(c_2|f_k, \mathbf{f}^i)| p(f_k|\mathbf{f}^i) df_k d\mathbf{f}^i \right\}.$$

- Chernoff distance:

$$d_k^i(c_j, c_{j'}) = -\log \int_{\mathcal{F}^i} \int_{\mathcal{F}_k} p^s(f_k|c_j, \mathbf{f}^i) p^{(1-s)}(f_k|c_{j'}, \mathbf{f}^i) df_k d\mathbf{f}^i, \quad s \in [0, 1]. \quad (2.30)$$

- Bhattacharyya distance:

$$d_k^i(c_j, c_{j'}) = -\log \int_{\mathcal{F}^i} \int_{\mathcal{F}_k} (p(f_k|c_j, \mathbf{f}^i) p(f_k|c_{j'}, \mathbf{f}^i))^{\frac{1}{2}} df_k d\mathbf{f}^i, \quad (2.31)$$

which is equivalent to the Chernoff distance with $s = \frac{1}{2}$. It also provides an upper and lower bound of the Bayes error probability [Fukunaga, 1990].

- Divergence:

$$d_k^i(c_j, c_{j'}) = \int_{\mathcal{F}^i} \int_{\mathcal{F}_k} (p(f_k|c_j, \mathbf{f}^i) - p(f_k|c_{j'}, \mathbf{f}^i)) \log \frac{p(f_k|c_j, \mathbf{f}^i)}{p(f_k|c_{j'}, \mathbf{f}^i)} df_k d\mathbf{f}^i = \quad (2.32)$$

$$D_{KL}(p(f_k|c_j, \mathbf{f}^i) || p(f_k|c_{j'}, \mathbf{f}^i)) + D_{KL}(p(f_k|c_{j'}, \mathbf{f}^i) || p(f_k|c_j, \mathbf{f}^i)),$$

where $D_{KL}(\cdot || \cdot)$ is the Kullback-Leibler divergence between two pdfs (see the definition (2.40) below). Although the Kullback-Leibler divergence can be used as an asymmetric distance measure between two distributions alone, the summation of two distances is used here in order to make the measure d symmetric, i. e. $d(x, y) = d(y, x)$, but $D_{KL}(p(x) || p(y)) \neq D_{KL}(p(y) || p(x))$.

- Patrick-Fischer distance:

$$d_k^i(c_j, c_{j'}) = \left\{ \int_{\mathcal{F}^i} \int_{\mathcal{F}_k} (p(f_k|c_j, \mathbf{f}^i)p(c_j|\mathbf{f}^i) - p(f_k|c_{j'}, \mathbf{f}^i)p(c_{j'}|\mathbf{f}^i))^2 df_k d\mathbf{f}^i \right\}^{\frac{1}{2}} \quad (2.33)$$

Despite an intuitive utility of the presented distance metrics for feature selection, they are rarely used in contemporary algorithms, see few examples [Papantoni-Kazakos, 1976; Devijver & Kittler, 1982; Miller, 1990]. More attention is paid to the Bhattacharyya distance due to its connection to the Bayes misclassification error [E. & C., 2003; Xuan et al., 2006]. In one of the recent works, Bhattacharyya, divergence and Patrick-Fischer metrics were employed for evaluating relevance of feature subsets in sequential search [Somol et al., 2005]. However, it was reported that filters using such metrics do not always select good feature subsets due to the difficulties in accurately estimating involved pdfs. For this reason, usage of the probabilistic distance measures is usually limited to the cases when the class-conditional pdfs come from known distributions and the metrics can be calculated analytically [Webb, 1999].

2.4 Information-theoretical feature selection

2.4.1 Definitions

Let us recall definitions and some properties of the fundamental information-theoretical concepts, Shannon entropy and mutual information. Entropy as a measure of uncertainty of a random variable was introduced by Claude Shannon [Shannon & Weaver, 1949],

though the closely related thermodynamical entropy was known before. Thus, for a random discrete variable A , its entropy is defined as follows:

$$H(A) = - \sum_{\mathcal{A}} p(a) \log p(a), \quad (2.34)$$

where $p(a)$ is the probability mass function of A .

A high level of entropy means that before observing a variable, we are highly uncertain about its future value. Therefore, as was already stated while describing different uncertainty functions, the entropy is maximal for uniform distributions and minimal if one of the possible outcomes of the variable appears with probability 1. Though the entropy was originally proposed for discrete variables, there is its analog for continuous variables called differential entropy:

$$H(A) = - \int_{\mathcal{A}} p(a) \log p(a) da, \quad (2.35)$$

where $p(a)$ refers to the probability density function. From now on, when referring to entropy, the differential entropy will be meant unless stated otherwise.

In case of two variables A and B , the uncertainty about the variable A once the variable B is known is quantified by the conditional entropy:

$$H(A|B) = H(A, B) - H(B) = \int_{\mathcal{A}} \int_{\mathcal{B}} p(a, b) \log p(a|b) db da, \quad (2.36)$$

where $p(a, b)$ is the joint probability density function of A and B and $H(A, B)$ is the joint entropy of two variables:

$$H(A, B) = \int_{\mathcal{A}} \int_{\mathcal{B}} p(a, b) \log p(a, b) db da. \quad (2.37)$$

Usually the logarithm with base 2 is used and entropy is measured in bits. Then, one can interpret the entropy of a variable as a number of bits necessary for its coding.

There are defining properties of Shannon entropy that are worth to be mentioned [Cover & Thomas, 1991]:

- the entropy is nonnegative, $H(A) \geq 0$;
- conditioning reduces the entropy, $H(A|B) \leq H(A)$;
- $H(A_1, \dots, A_n) \leq \sum_{i=1}^n H(A_i)$, with equality only if the variables A_1, \dots, A_n are independent;

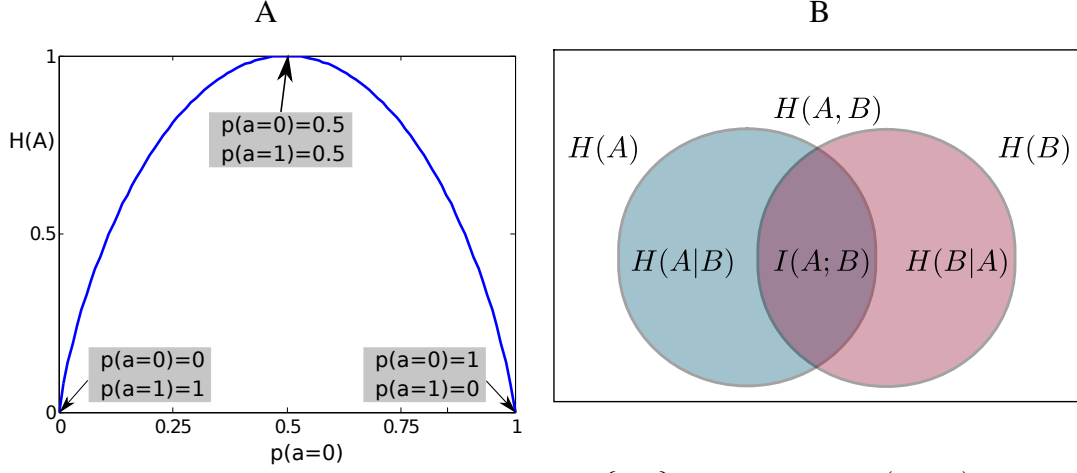


Figure 2.3: A: Entropy of the binary variable $A \in \{0, 1\}$ plotted against $p(a = 0)$. It is shown that entropy reaches its maximum value at $p(a = 0) = p(a = 1) = 0.5$ and its minimum value at $p(a = 0) = 1$ and $p(a = 0) = 0$. B: Diagram of the relation between mutual information of two variables $I(A; B)$ and their marginal, joint and condition entropies, $H(A)$ and $H(B)$, $H(A, B)$, $H(A|B)$, respectively.

- $H(A)$ is a concave function of $p(a)$.

The mutual information of two continuous random variables A and B measures the degree of their dependence is defined as follows:

$$I(A; B) = H(A) - H(A|B) = H(B) - H(B|A) = H(A) + H(B) - H(A, B) = \int \int_A \int_B p(a, b) \log \frac{p(a, b)}{p(a)p(b)} db da. \quad (2.38)$$

In other words, mutual information quantifies a change in the uncertainty about one variable after the second variable is observed. Note that the entropy $H(A)$ or $H(B)$ can be infinite as $da \rightarrow 0$ or $db \rightarrow 0$, respectively. However, the mutual information is always finite because it is defined as a difference of entropies, thus, the infinite terms will vanish [Haykin, 1999].

Among the important properties of the mutual information, one can distinguish the following:

- the mutual information is symmetric, $I(A; B) = I(B; A)$;
- it is nonnegative, $I(A; B) \geq 0$, with equality only if the variables A and B are independent.

Conditional mutual information $I(A;B|C)$ measures the amount of information of two variables A and B conditioned on the variable C :

$$I(A;B|C) = H(A|C) + H(B|C) - H(A,B|C) = \int_A \int_B \int_C p(a,b,c) \log \frac{p(a,b|c)}{p(a|c)p(b|c)} dc db da. \quad (2.39)$$

Conditional mutual information is a key element of sequential information-theoretical feature selection. Recall that on every iteration we look for a feature F_k that maximizes $I(C;F_k|F_{\alpha_1}, \dots, F_{\alpha_i})$, the mutual information with the class variables C conditioned on the already selected features $F_{\alpha_1}, \dots, F_{\alpha_i}$.

Another central concept of information theory is the relative entropy or Kullback-Leibler divergence, which for two probability distributions $p(a)$ and $q(a)$ measures a distance between them:

$$D_{KL}(p(a)||q(a)) = \int_{\mathcal{A}} p(a) \log \frac{p(a)}{q(a)} da. \quad (2.40)$$

However, the Kullback-Leibler divergence is not a true metric because it is not symmetric and the triangle inequality does not always hold. Using the definition of the Kullback-Leibler divergence, one can represent the mutual information and get its interpretation in terms of the distance between two distributions:

$$I(A;B) = \int_{\mathcal{A}} \int_{\mathcal{B}} p(b) D_{KL}(p(a|b)||p(a)) ba. \quad (2.41)$$

Then, the mutual information measures how much on average the distribution of A changes if it is conditioned on B . Obviously, if A and B are independent, conditioning on B will not have any effect on A and the average Kullback-Leibler distance will be zero.

For further reading on information theory, refer to [Shannon & Weaver, 1949; Cover & Thomas, 1991; Mackay, 2003].

2.4.2 Use in solving classification tasks

Already in 1962 Lewis proposed mutual information between a class variable and a feature as a statistic measuring “goodness” of this feature for classification [Lewis, 1962]. The statistic had to reflect a degree of correlation between the feature and the class variable and it was derived with an objective to reduce a misclassification error. Lewis showed experimentally that the accuracy of the classification was higher when using features with

higher value of the mutual information. Therefore, it was concluded that it is indeed useful for selecting features that are relevant for classification. Similar finding appeared also in the field of visual neuroscience, where Ullman and colleagues showed that features maximizing mutual information with a class are optimal for use in visual classification tasks [Ullman et al., 2002].

A more formalized justification for using mutual information as a criterion for selecting discriminative features is based on inequalities relating the Bayes error probability to the conditional entropy $p(c|f)$ and consequently to the mutual information $I(C; F)$.

For example, the Fano weak lower bound on the conditional entropy [Fano, 1961] states the following:

$$H(C|F) \leq 1 + p_e \log_2(m - 1), \quad (2.42)$$

where p_e is the Bayes error probability when using the feature F for classification and m is a number of the classes. However, this bound becomes degenerated for two-class problems. Fano also introduced a strong lower bound on this quantity [Fano, 1961]:

$$H(C|F) \leq H(p_e) + p_e \log_2(m - 1). \quad (2.43)$$

And the upper Hellman-Raviv bound [Hellman & Raviv, 1970] is given by the following expression:

$$H(C|F) \geq 2p_e. \quad (2.44)$$

As $I(C; F) = H(C) - H(C|F)$, it is obvious that a feature F , which maximizes the mutual information $I(C; F)$ or equivalently minimizes $H(C|F)$, assures a small classification error.

Recently Brown and colleagues showed that selection criteria based on mutual information can be derived from the formulation of the conditional likelihood maximization problem [Brown et al., 2012]. Let us review their analysis.

In our standard classification framework, the goal is to estimate a posterior probability $p(c|\mathbf{f})$. Assuming that some features are redundant, we want to select only relevant features for classification. Suppose that there is an N -dimensional binary vector θ indicating selected features. Thus, $\theta_i = 1$ if a feature F_i is in the subset of the relevant features and $\theta_i = 0$ otherwise. We want to find the optimal parameter θ^* ensuring the best possible classification accuracy, which is ideally provided by using all features $\mathbf{F} = \{F_1, \dots, F_n\}$. Thus, we look for such θ^* that $p(c|\mathbf{F}) = p(c|\mathbf{F}^{\theta^*})$.

Suppose also that a true posterior $p(c|\mathbf{F}^\theta)$ is approximated by the model $q(c|\mathbf{F}^\theta, \tau)$, where τ represents parameters necessary for classification. Within such setup, our goal is to find

the parameters (θ, τ) that maximize the conditional loglikelihood of the class labels given the data \mathcal{D} :

$$l(\theta, \tau | \mathcal{D}) = \frac{1}{T} \sum_{i=1}^T \log q(c_i | \mathbf{f}_i^\theta, \tau) \approx \mathbb{E}_{p(\mathbf{f}, c)} \left[\log q(c_i | \mathbf{f}_i^\theta, \tau) \right], \quad (2.45)$$

where $\mathbb{E}_{p(\mathbf{f}, c)}[\cdot]$ is the expectation w. r. t. the distribution $p(\mathbf{f}, c)$ and $\frac{1}{T} \sum_{i=1}^T (\cdot)$ is its finite sample estimate. After adding and subtracting both $p(c | \mathbf{f}^\theta)$ and $p(c | \mathbf{f})$, 2.45 can be written in the following form:

$$-l \approx \mathbb{E}_{p(\mathbf{f}, c)} \left[\log \frac{p(c | \mathbf{f}^\theta)}{q(c | \mathbf{f}^\theta, \tau)} \right] + \mathbb{E}_{p(\mathbf{f}, c)} \left[\log \frac{p(c | \mathbf{f})}{p(c | \mathbf{f}^\theta)} \right] - \mathbb{E}_{p(\mathbf{f}, c)} [\log p(c | \mathbf{f})]. \quad (2.46)$$

The first and the third terms are $D_{KL}(p^\theta || q^\theta)$ and $H(C | \mathbf{F})$, respectively. Let us look at the second term:

$$\begin{aligned} \mathbb{E}_{p(\mathbf{f}, c)} \left[\log \frac{p(c | \mathbf{f})}{p(c | \mathbf{f}^\theta)} \right] &= \mathbb{E}_{p(\mathbf{f}, c)} \left[\log p(c | \mathbf{f}) - \log p(c | \mathbf{f}^\theta) \right] = \\ \mathbb{E}_{p(\mathbf{f}, c)} \left[\log \frac{p(c, \mathbf{f})}{p(c)p(\mathbf{f})} + \log p(c) - \log \frac{p(c, \mathbf{f}^\theta)}{p(c)p(\mathbf{f}^\theta)} - \log p(c) \right] &= \\ \mathbb{E}_{p(\mathbf{f}, c)} \left[\log \frac{p(c, \mathbf{f})}{p(c)p(\mathbf{f})} \right] - \mathbb{E}_{p(\mathbf{f}, c)} \left[\log \frac{p(c, \mathbf{f}^\theta)}{p(c)p(\mathbf{f}^\theta)} \right]. \end{aligned} \quad (2.47)$$

Let $\mathbf{F}^{\bar{\theta}}$ denote the non-selected features. Then, the variable \mathbf{F} can be represented as a joint variable, $\mathbf{F} = \{\mathbf{F}^{\bar{\theta}}, \mathbf{F}^\theta\}$. Taking this into account, we have:

$$\mathbb{E}_{p(\mathbf{f}, c)} \left[\log \frac{p(c, \mathbf{f}^\theta)}{p(c)p(\mathbf{f}^\theta)} \right] = \mathbb{E}_{\mathbf{F}^{\bar{\theta}}, \mathbf{F}^\theta, c} \left[\log \frac{p(c, \mathbf{f}^\theta)}{p(c)p(\mathbf{f}^\theta)} \right] = \mathbb{E}_{\mathbf{F}^\theta, c} \left[\log \frac{p(c, \mathbf{f}^\theta)}{p(c)p(\mathbf{f}^\theta)} \right], \quad (2.48)$$

as the function whose expectation is calculated does not depend on $\mathbf{F}^{\bar{\theta}}$. Plugging this into (2.47), we get:

$$\mathbb{E}_{p(\mathbf{f}, c)} \left[\log \frac{p(c, \mathbf{f})}{p(c)p(\mathbf{f})} \right] - \mathbb{E}_{\mathbf{F}^\theta, c} \left[\log \frac{p(c, \mathbf{f}^\theta)}{p(c)p(\mathbf{f}^\theta)} \right] = I(C; \mathbf{F}) - I(C; \mathbf{F}^\theta). \quad (2.49)$$

Finally, the minimization problem (2.46) under the assumption that a number of training samples T tends to infinity can be rewritten as:

$$-\lim_{T \rightarrow \infty} l \approx D_{KL}(p^\theta || q^\theta) + I(C; \mathbf{F}) - I(C; \mathbf{F}^\theta) + H(C | \mathbf{F}). \quad (2.50)$$

Following [Brown et al., 2012], we interpret all components of the conditional loglikelihood. $D_{KL}(p^\theta||q^\theta)$ is the Kullback-Leibler divergence between the true and approximated probability of the class labels given the selected features. In other words, it measures how good the approximation of the model q is, which in turn depends on the model parameters τ . Note that though $D_{KL}(p^\theta||q^\theta)$ formally depends on θ , it does not tell much about optimality of θ but only about the quality of approximation given some fixed θ . The difference $I(C; \mathbf{F}) - I(C; \mathbf{F}^\theta)$ shows the amount of information that is left between the class variable C and non-selected features. If the optimal parameter θ^* is found, this difference will be 0. The last term can not be reduced by optimization and represents the intrinsic uncertainty of the classification problem at hand, i. e. the uncertainty about the class labels which is left after observing all features.

As the goal of feature selection is to find the optimal parameter θ^* , the minimization problem (2.50) can be reduced just to maximizing $I(C; \mathbf{F}^\theta)$, as other terms are not influenced by θ . This proves again that features maximizing the mutual information with the class are good candidates to be included in a classifier. Note that sequential feedforward techniques form a subset of relevant features sequentially. Therefore, they maximize $I(C; \mathbf{F}^\theta)$ by iteratively maximizing $I(C; F_k | F_{\alpha_1}, \dots, F_{\alpha_i})$. Then, the above statement can be reformulated in the following way: on every iteration a feature F_k maximizing $I(C; F_k | F_{\alpha_1}, \dots, F_{\alpha_i})$ will decrease the classification error more than other feature candidates.

2.5 Estimation of mutual information

Despite theoretical attractiveness of mutual information, its practical use is complicated. The reason is that usually mutual information is not known a priori and therefore it should be estimated from data at hand. As mutual information can be decomposed into a sum of marginal and joint entropies, $I(C; \mathbf{F}) = H(C) + H(\mathbf{F}) - H(C, \mathbf{F})$, its estimation is usually reduced to the problem of estimating these entropies. Although entropy estimation has been massively studied already for several decades, it is still considered to be a difficult task. Moreover, in situations when the amount of data is limited, there exists no unbiased entropy estimator [Panzeri et al., 2007].

Methods for entropy estimation can be divided into plug-in and nonplug-in types [Beirlant et al., 1997]. Recall the differential entropy of a random variable:

$$H(X) = - \int_{\mathcal{X}} p(x) \log p(x) dx. \quad (2.51)$$

Then, according to the plug-in approach, first a probability density $p(x)$ is estimated and then this estimate $\hat{p}(x)$ is plugged into the expression of the entropy definition (2.51). In turn, the nonplug-in methods estimate the entropy function directly.

Before introducing different estimation techniques, let us formally define the notion of the accuracy of an estimator. An estimation error can be decomposed into two components: the bias and the variance. The former is the error due to the difference between the expectation value of the estimate and the true value, whereas the latter is the error due to the variability of estimates build on different data subsamples. Therefore, given T i.i.d. samples x_1, \dots, x_T , an estimator $\hat{p}(x)$ of a function $p(x)$ has no error if both the bias and the variance are equal to zero:

$$\lim_{T \rightarrow \infty} \mathbb{E}_{p(x)}[\hat{p}(x)] - p(x) = 0, \quad \lim_{T \rightarrow \infty} \text{Var}_{p(x)}[\hat{p}(x)] = 0. \quad (2.52)$$

2.5.1 Plug-in approaches

In case there is no knowledge about a structure of the probability density function $p(x)$, the differential entropy as it is given by the expression (2.51) requires numerical integration. Note that (2.51) is nothing but the expectation of the logarithmic function w. r. t. the distribution $p(x)$:

$$H(X) = \mathbb{E}_{p(x)}[\log \hat{p}(x)]. \quad (2.53)$$

In order to avoid integration, the expectation value can be approximated by the average of $\log \hat{p}(x)$ over all samples. Given a training set $\{x_1, \dots, x_T\}$, such approximation leads to the following *resubstitution estimator* [Beirlant et al., 1997]

$$H(X) = -\frac{1}{T} \sum_{i=1}^T \log \hat{p}(x_i). \quad (2.54)$$

Another plug-in estimator is based on the idea of *leave-one-out cross-validation* and therefore is less prone to overfitting [Ivanov & Rozhkova, 1981; Hall & Morton, 1993]:

$$H(X) = -\frac{1}{T} \sum_{i=1}^T \log \hat{p}_i(x_i), \quad (2.55)$$

where the estimate of the probability density $\hat{p}_i(x_i)$ in the point x_i is built using all samples excluding the sample x_i , which is used for validation.

2.5.1.1 Density estimation.

Now we turn to estimation of a probability density function which can be plugged in the estimators presented above. Stating the problem formally, given a finite number of i.i.d. n -dimensional samples, we want to model a density $p(x)$.

Depending on the assumptions about a form of the density function used for its estimation, one distinguishes parametric and nonparametric estimation techniques. The parametric estimators assume that a density function can be described by a certain model. Then, the estimation problem is reduced to fitting parameters of the assumed model to observed data. The nonparametric techniques do not use such assumptions and estimation is driven purely by the observations. On the one hand, such a dependence on the data obviously implies a higher variance compared to the parametric methods. On the other hand, the bias of the nonparametric estimators vanishes asymptotically as more data are observed, while the parametric estimators will be always biased if the wrong model of the underlying density function is assumed [Scott, 2004]. One of the advantages of the parametric methods is that the mean integrated squared error is of the order $O(T^{-1})$ and does not depend on the dimension of a model, whereas convergence of the nonparametric methods becomes slow in higher dimensions [Scott, 1992].

The main representatives of **parametric techniques** are maximum likelihood, maximum a-posteriori and bayesian parametric methods.

As was mentioned before, within this approach one assumes that the estimated density can be modeled as a function of a certain form. Let this function be characterized by a parameter vector θ . Then, the goal of the *maximum likelihood estimator* (MLE) is to find such θ^* that maximizes the likelihood of the observed data:

$$\theta^* = \arg \max_{\theta} p(x|\theta) = \arg \max_{\theta} \prod_{i=1}^T p(x_i|\theta).$$

The main drawback of this method is a high possibility of overfitting, since the value θ^* is the best fit on the observations used for learning.

At the same time, MLE provides the easiest way of estimating the probability mass of discrete variables. Suppose a variable x has N possible outcomes $\{r_1, \dots, r_N\}$. Assuming that the outcomes are multinomially distributed, $\hat{p}_{MLE}(x = r_i)$ is just a frequency count $\frac{t_i}{T}$, where t_i is a number of the observations with the outcome r_i . Plugging such counts into the resubstitution estimator (2.54), we obtain the simplest plug-in estimate of the entropy of a discrete variable x

$$\hat{H}_{MLE}(X) = - \sum_{i=1}^N \frac{t_i}{T} \log \frac{t_i}{T}. \quad (2.56)$$

This estimate is known to have a negative bias that depends on the number of observations [Antos & Kontoyiannis, 2001].

Another example of the plug-in estimator based on the ML density estimate is the jackknifed ML entropy estimate, which is the asymptotic correction of MLE [Efron & Stein, 1981]:

$$\hat{H}_{JK} = T\hat{H}_{MLE}(X) + \frac{T-1}{T} \sum_{i=1}^T \hat{H}_{MLE-i}(X), \quad (2.57)$$

where $\hat{H}_{MLE-i}(X)$ is the estimate based on the all training samples except the i^{th} one. The jackknife estimate of a function is known to be consistent, i. e. it converges in probability to the true value of this function¹.

Bayesian methods tackle the problem of overfitting by imposing a prior on the distribution of the parameter vector θ . The observations are used to update a posterior distribution on the parameter values which is expressed in terms of the prior $p(\theta)$ and data likelihood $p(x|\theta)$ according to the Bayes rule [Lee, 2004; Robert, 2001]:

$$\hat{p}(\theta|x) = \frac{p(\theta)p(x|\theta)}{\int p(\theta)p(x|\theta)d\theta}. \quad (2.58)$$

Then, for any new observation ξ , a value of the density function $p(\xi|x)$ is estimated by marginalizing the likelihood at this point over the learned posterior of θ :

$$\hat{p}(\xi|x) = \int p(\xi|\theta)p(\theta|x)d\theta. \quad (2.59)$$

As more data are observed, the posterior of θ moves towards its true distribution and consequently the estimate $\hat{p}(\xi|x)$ moves towards the real value $p(\xi|x)$.

Note that, unlike MLE, the bayesian methods do not look for the single best parameter vector θ but rather learn its posterior distribution. Moreover, the final posterior is not purely data-driven as it depends on the chosen prior, which helps to avoid overfitting. Therefore, bayesian estimates are usually quite accurate if a sufficient amount of training data is available. At the same time, in situations when a number of observations is not large, the estimate $\hat{p}(\xi|x)$ can be highly biased if the specified prior is far from the real distribution $p(\theta)$. In addition, the bayesian techniques are computationally expensive as they require integration over the parameter space.

In order to estimate Shannon entropy or mutual information, the most popular Bayesian methods infer directly the entropy rather than pdfs for plug-in estimators. Therefore, they will be reviewed further while discussing the nonplug-in techniques, see Subsection (2.5.2).

¹An estimator $\hat{f}(x)$ converges in probability if $\lim_{T \rightarrow \infty} p(|\hat{f}(x) - f(x)| \geq \epsilon) = 0$ for some small ϵ .

Maximum a-posteriori techniques represent some sort of a mixture of the maximum likelihood and the Bayesian methods. Here, the modeled density is estimated using a single value of θ , which is equal to the mode of its posterior distribution:

$$\theta = \arg \max_{\theta} \hat{p}(\theta|x).$$

Such approach does not require normalization and consequently integration as in (2.58), because the value of θ maximizing $\hat{p}(\theta|x)$ is the same as maximizing $p(\theta)p(x|\theta)$. Some examples of using the maximum a-posteriori method for probability density estimation include [Gauvain & Lee, 1994; Premus & Alexandrou, 1995; Anzai & Hara, 2010].

As the name suggests, **nonparametric techniques** do not use any assumptions about the form of the estimated density functions, these densities are rather inferred purely from observations.

Let us introduce a formal framework for the nonparametric density estimation. For a random variable x , in order to find its probability density function $p(x)$, we want to build an estimator $\hat{p}(x)$ using T i.i.d. samples drawn from the distribution $p(x)$. For this, we define a small region \mathcal{R} containing x and say that k out of T samples fall in this region, i. e. $\frac{k}{T}$ is the probability of a sample to fall in \mathcal{R} . Then, the density $p(x)$ can be approximated by:

$$\hat{p}(x) = \frac{k}{TV}, \quad (2.60)$$

where V is a volume of the region \mathcal{R} . In order to converge, this estimator should satisfy the following conditions

$$\lim_{T \rightarrow \infty} V = 0, \quad \lim_{T \rightarrow \infty} k = \infty, \quad \lim_{T \rightarrow \infty} k/T = 0. \quad (2.61)$$

The first condition requires that the region \mathcal{R} shrinks with a number of the samples. The second condition makes sure that in case $p(x) \neq 0$, $\frac{k}{T}$ converges in probability to the true probability of x being in \mathcal{R} . And according to the third condition, $k \rightarrow \infty$ slower than $T \rightarrow \infty$.

Histograms. One-dimensional histograms represent the simplest nonparametric density estimation technique [Pearson, 1895; Tukey, 1977; Scott, 1979, 1992]. Imagine that a domain of definition of a continuous variable x is partitioned in N bins of equal size. Then, the probability density $p(x)$ is estimated by counting a number of the observations that fall in the same bin as x and then properly normalizing it:

$$\hat{p}(x) = \frac{k_j}{Th},$$

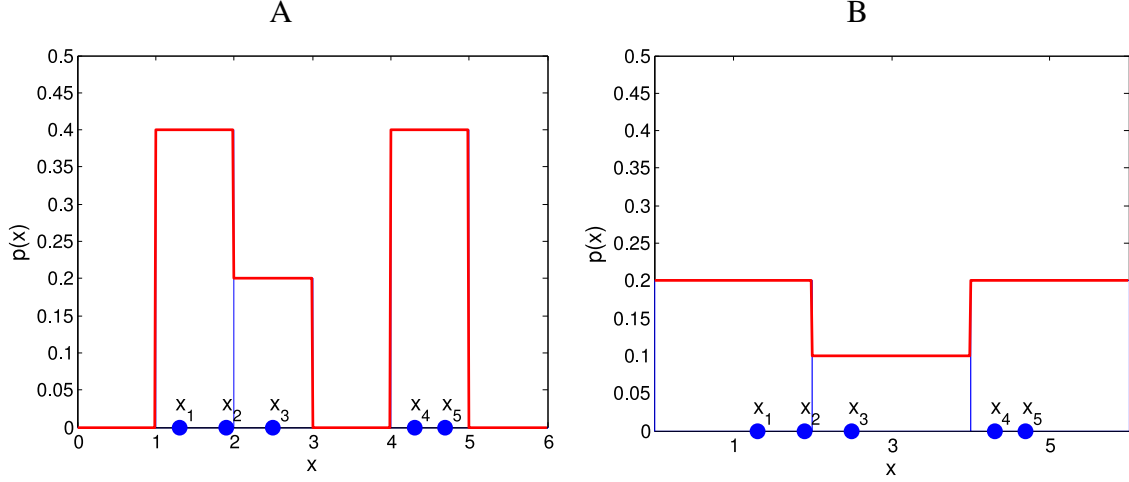


Figure 2.4: Histogram estimators with different bin widths $h = 1$ and $h = 2$ (subplots A and B, respectively). The histogram bins are depicted in blue and the estimated probability density function is in red.

where k_j is a number of the samples in the j^{th} bin where x falls and h is a bin width. Within the general approach described by the expression (2.60), the region \mathcal{R} is defined by the j^{th} bin, and the volume of this region V is the bin width h .

In multidimensional case, the volume of the n -dimensional bin is $V = h^n$. Although generalization to multivariate histograms is straightforward, their application is limited. The reason is that the number of bins grows exponentially with the number of dimensions, that is in n -dimensional case one needs N^n bins, which can obviously cause memory problems. Moreover, as in higher dimensions data become sparse, most of the univariate bins across the different dimensions will be empty resulting in zero values of the density function. To solve this issue, one needs a large number of observations for constructing a histogram estimator, otherwise the tails of the pdf will be estimated rather poor [Scott, 1992].

The error of a density estimator is often measured by the mean integrated squared error which can be decomposed into the integrated variance and the integrated squared bias:

$$\text{MISE}(\hat{p}(x)) = \mathbb{E}_{p(x)} \left[\int_{\mathcal{X}} (\hat{p}(x) - p(x))^2 dx \right] = \int_{\mathcal{X}} \text{Var} [\hat{p}(x)] dx + \int_{\mathcal{X}} \text{Bias} [\hat{p}(x)]^2 dx, \quad (2.62)$$

where $\text{Var} [\hat{p}(x)] = \mathbb{E}_{p(x)} \left[(\hat{p}(x) - \mathbb{E}_{p(x)} [\hat{p}(x)])^2 \right]$ and $\text{Bias} [\hat{p}(x)] = \mathbb{E}_{p(x)} [\hat{p}(x) - p(x)]$.

For the equally-spaced histogram estimator, the former is proportional to $\frac{1}{Th}$ and the latter is about $\frac{h^2}{12} \int p'(x)^2 dx$ [Scott, 2004]. Thus, in order to keep both components of the error

low, the following conditions should be met, which are related to the general convergence conditions of nonparametric density estimators (2.61):

$$\lim_{T \rightarrow \infty} h \rightarrow 0, \quad \lim_{T \rightarrow \infty} Th \rightarrow \infty. \quad (2.63)$$

Due to discontinuities at the boundaries of the histogram bins, the estimated densities are not smooth. As a solution, a frequency polygon estimator was developed [Scott, 1985; Beirlant et al., 1999]. This extension to the histograms performs a linear interpolation based on the middle points of the equally-sized equally-spaced histogram bins. Another extension assigns every data point to several bins with weights given by B-spline functions [Daub et al., 2004].

There are some examples of using histograms in plug-in estimators of entropy and mutual information [Györfi & van der Meulen, 1987; Hall & Morton, 1993; Battiti, 1994; Kwak & Choi, 2002b]. However, note that the resubstitution histogram entropy estimate is in fact the maximum likelihood estimate of entropy of the discretized continuous distribution. Therefore, it is also negatively biased. For this reason, contemporary histogram-based entropy estimators usually try to find a way to cancel this bias, e. g. [Moddemeijer, 1989; Paninski, 2003]. Since they no longer fit in the framework of the plug-in approach, such estimators will be reviewed later in Subsection 2.5.2.

Kernel density estimation. To construct a probability density estimate, the kernel technique, which was developed by Rosenblatt [Rosenblatt, 1956] and Parzen [Parzen, 1962], specifies a set of small regions centered at every training sample x_i . These regions are shaped by some kernel functions that assign an observation x to the corresponding region depending on the distance between the kernel center x_i and x . Then, for a training set consisting of T i.i.d. one-dimensional samples, the kernel density estimate (KDE) of the pdf $p(x)$ is

$$\hat{p}(x) = \frac{1}{Th} \sum_{i=1}^T K\left(\frac{x-x_i}{h}\right), \quad (2.64)$$

where $K(\frac{x-x_i}{h})$ is a kernel function with a bandwidth parameter h that specifies the width of the kernel. Note that a sum of the kernel responses gives an estimate for k from the expression (2.60), a number of points around x .

In order to assure that the estimate $\hat{p}(x)$ satisfies the necessary conditions for a probability density function, i. e. $p(x) \geq 0$ and $\int p(x)dx = 1$, there are the following constraints that should be imposed on the kernel function $K(w)$:

$$K(w) > 0, \quad \int K(w)dw = 1, \quad \int wK(w)dw = 0. \quad (2.65)$$

That is the kernel should be positive, a density function itself and centered at zero. There are several commonly used kernels such as rectangular, triangular, normal, Barlett-Epanechnikov, cosine etc. However, practical investigations showed that the choice of the particular kernel function is not crucial for estimation accuracy [Webb, 1999].

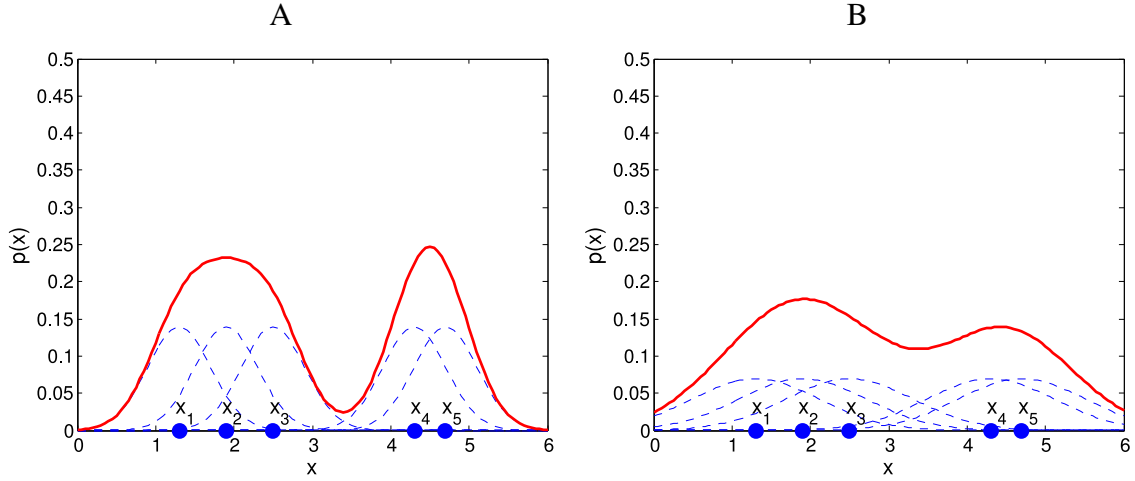


Figure 2.5: Kernel density estimators using Gaussian kernels with the different bandwidth parameters $h \approx 0.6$ and $h \approx 1.2$ (subplots A and B, respectively). Blue dashed curves represent the single kernel functions centered at the training points and the red curves depict the estimated pdf.

For n -dimensional observations $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$, the multivariate generalization of the kernel density estimate is

$$\hat{p}(\mathbf{x}) = (T|H|)^{-1} \sum_{i=1}^T K(H^{-1}(\mathbf{x} - \mathbf{x}_i)), \quad (2.66)$$

here $K(\cdot)$ is a n -dimensional kernel function and H is a symmetric positive definite $n \times n$ matrix of kernel smoothing parameters, also called a bandwidth matrix. Usually for simplicity a diagonal form of the matrix H is assumed. In this case, the multivariate kernel function is a product of the univariate kernels and the above expression simplifies to

$$\hat{p}(\mathbf{x}) = (T \prod_{j=1}^n h_j)^{-1} \sum_{i=1}^T \prod_{j=1}^n K\left(\frac{x_j - x_{j,i}}{h_j}\right), \quad (2.67)$$

where (h_1, \dots, h_n) are the diagonal elements of the matrix H , and $x_{j,i}$ is the value of the j^{th} feature of the sample \mathbf{x}_i . If the data is prerotated, i. e. correlations between the dimensions are removed, the product kernel is equivalent to the KDE with the full bandwidth matrix.

The convergence of this technique was proven under the following conditions [Duda et al., 2001]:

$$\lim_{T \rightarrow \infty} V = 0, \quad \lim_{T \rightarrow \infty} TV = \infty, \quad (2.68)$$

where V is a volume of the region defined by the kernel, so $V = |H|$ or $V = \prod_{j=1}^n h_j$ depending on the type of the bandwidth matrix. The plug-in entropy estimator based on one-dimensional kernel density estimates was shown to be asymptotically normal ² with an error of the rate $T^{-1/2}$, however, for the multivariate case the estimator is just consistent, i. e. converges in probability [Eggermont & LaRiccia, 1999].

Scott [Scott, 1992] states that in theory for a number of dimensions $n > 5$ it is not possible to obtain accurate estimates using kernel density method. However, he mentions practical examples showing reasonable results, for example in case of a 10-dimensional space with 225 training samples. His conclusion is that in such situations it is possible to find at least the structure of the pdf, though with a large estimation error, which is still acceptable in higher dimensions.

Due to their simplicity, natural extension to the multivariate pdfs and satisfactory accuracy which depends on the amount of data available, kernel density estimates are widely used in practical applications. Among them, there are examples of using KDE in estimation of entropy [Dmitriev & Tarasenko, 1973; Ahmad & Lin, 1976; Ivanov & Rozhkova, 1981; Ahmed & Gokhale, 1989; Joe, 1989], mutual information in general [Moon et al., 1995; Zhou et al., 2005; Lin & Tang, 2006; Xu et al., 2008; Qiu et al., 2009] and mutual information for feature selection [Kwak & Choi, 2002a; Ozertem et al., 2006; Carmona et al., 2011; Zhang & Hancock, 2011].

k-nearest neighbor density estimation. In contrast to the discussed above histogram and kernel estimators, in order to approximate a probability density $p(x)$ the k -nearest neighbor technique [Fix & Hodges, 1951; Loftsgaarden & Quesenberry, 1965] does not directly specifies a volume of the the region \mathcal{R} where x falls but rather fixes a number k of the samples contained in this region. That is, an estimate $\hat{p}(x)$ is based on the k direct neighbors of the observation x and can be thought as to have a data-adaptive nature.

For $\mathbf{x} \in \mathbb{R}^n$, the k -NN density estimate is an already familiar expression with fixed k :

$$\hat{p}(\mathbf{x}) = \frac{k}{TV(\mathbf{x})}, \quad V(\mathbf{x}) = \frac{r_k^n(\mathbf{x})\pi^{n/2}}{\Gamma(n/2 + 1)}, \quad (2.69)$$

where $V(\mathbf{x})$ is the volume of the n -dimensional hypersphere with the radius $r_k(\mathbf{x}) = d(\mathbf{x}, \mathbf{x}_i)$ being the n -dimensional Euclidean distance between \mathbf{x} and its k^{th} neighbor \mathbf{x}_i .

²Asymptotic normality of an estimate implies that $\lim_{T \rightarrow \infty} T^{1/2}(\hat{H}(X) - H(X)) = \mathcal{N}(0, \sigma^2)$.

The estimator converges in probability, if the followings conditions are satisfied [Devroye & Wagner, 1977]:

$$\lim_{T \rightarrow \infty} k = \infty, \quad \lim_{T \rightarrow \infty} \frac{k}{T} = 0, \quad \lim_{T \rightarrow \infty} \frac{k}{\log T} = \infty. \quad (2.70)$$

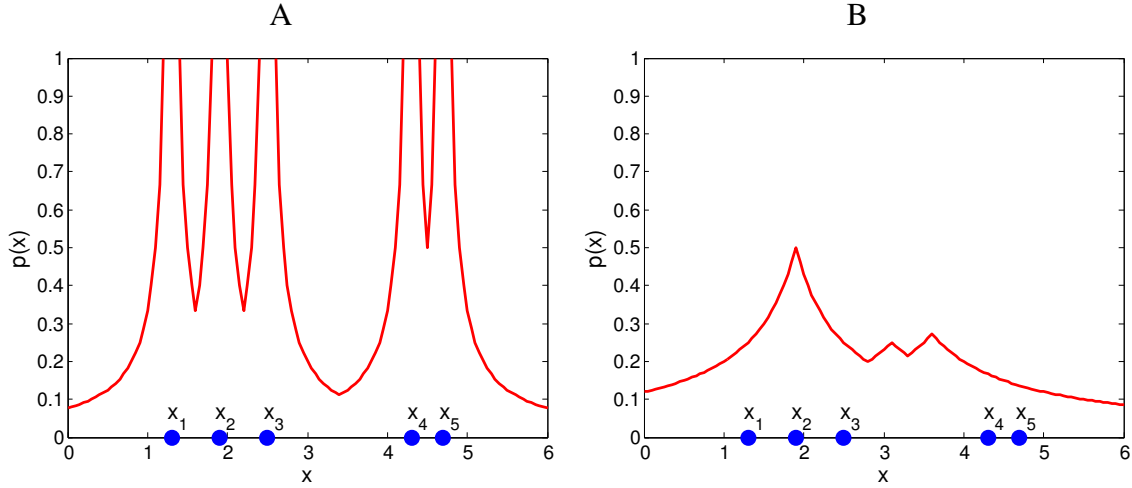


Figure 2.6: k -nearest neighbor density estimators with different number of neighbors $k = 1$ and $k = 3$ (subplots A and B, respectively).

In practice, k -NN density estimates are not very popular since they are heavy-tailed, noise-sensitive and discontinuous due to the fact that $r_k(x)$ is not differentiable. However, this approach is closely related to the widely used k -NN entropy estimators which will be reviewed later in Subsection (2.5.2).

Guided by the same idea of fixing the number of the observations falling in \mathcal{R} , Scott proposed histograms with an equal number of points in every bin. This modification was meant to improve inaccurate estimates in the tails occurring due to lack of data [Scott, 1992].

Selection of a smoothing parameter. All nonparametric techniques discussed above have a primary parameter which has to be tuned: a bin width for histograms, a kernel bandwidth for KDE and a number of neighbors for k -NN. All these parameters control the smoothness of the estimated density and therefore we will refer to all of them as “smoothing parameters”. Setting it too large, all details of the density structure are lost, whereas setting it too small will lead to a highly variable estimate with many false peaks around every sample point. This also illustrates the known bias-variance trade-off: undersmoothed estimators will have the large variance but the low bias, whereas the situation is opposite for the oversmoothed estimates [Kraskov et al., 2004].

For the k -NN estimator, the optimal value of k for should be usually tuned for the particular data at hand. However, $k = \sqrt{T}$ was reported to show good results [Loftsgaarden & Quesenberry, 1965].

For the plug-in estimator of the mutual information using histograms, it was shown that the difference between the estimate $\hat{I}(X, Y)$ and the true value $I(X, Y)$ depends on the number of bins N_X and N_Y of both variables [Li, 1990]:

$$\hat{I}(X, Y) - I(X, Y) \approx \frac{1}{2T} (N_X N_Y - N_X - N_Y). \quad (2.71)$$

The approximation can be used for correcting an estimation error if ratios of true counts $t_{xy}/t_x t_y$ are approximately the same for different bins of x and y . However, in the extremely undersampled regime, this correction is inaccurate [Battiti, 1994]. The difference between $\hat{I}(X, Y)$ and $I(X, Y)$ is usually greater than zero and it can be decreased by reducing the number of bins, i. e. setting the bin widths larger. However, as was already stated, too wide bins will not capture the data structure. Therefore, the proper smoothing parameter acting as a compromise is important. See Figures 2.4 and 2.5 for examples of histograms and kernel density estimates with different smoothing parameters.

For the histogram estimator, there are a lot of suggestions how to choose a bin width, see [Scott, 1992] for a good review. For a general case, Tukey suggested a number of equally-spaced bins being $N = \sqrt{T}$ [Mosteller & Tukey, 1977]. For normal data, there is the Sturges' rule: $N = 1 + \log_2 T$, however, it produces too few bins and the data is heavily oversmoothed [Scott, 1992]. A more robust rule gives an expression in terms of the interquartile range (IQR): $h = 2(IQR)T^{-\frac{1}{3}}$ [Freedman & Diaconis, 1981].

Using the normal density as a reference while minimizing the asymptotic mean integrated squared error of the estimate, the optimal width of the n -dimensional histogram can be expressed in terms of the standard deviation of the sampled data [Scott, 1979]

$$h_i = 2 \cdot 3^{\frac{1}{n+2}} \pi^{\frac{n}{2n+4}} \sigma_i T^{-\frac{1}{n+2}}. \quad (2.72)$$

Such expression is called the normal reference rule. Following the analogous procedure, one can also derive the optimal bandwidth for kernel density estimate [Silverman, 1986]. For the n -dimensional product kernel, the optimal bandwidth parameter for the i^{th} dimension is:

$$h_i = \left(\frac{4}{n+2}\right)^{\frac{1}{n+4}} \sigma_i T^{-\frac{1}{n+4}}, \quad (2.73)$$

where σ_i is the standard deviation of the data points along i^{th} dimension. The method produces good estimates for univariate densities but tends to oversmoothing for multivariate cases.

There is a number of bandwidth selection techniques based on cross-validation optimizing different criteria such as the Kullback-Leibler loss function [Rudemo, 1982; Bowman, 1984] or usual asymptotic mean integrated squared error [Hall et al., 1991]. Bootstrapping was also used to find the optimal bandwidth for univariate [Taylor, 1989] and multivariate data [Sain et al., 1992]. Among more sophisticated methods that can be easily extended to the multivariate densities are Markov chain Monte Carlo methods. They estimate a bandwidth matrix through the data likelihood using cross-validation and are reported to have a good performance, e. g. see [Zhang et al., 2004]. For further review on bandwidth selection methods, see [Turlach, 1993].

Instead of looking for the globally best bandwidth vector, it can be defined adaptively [Scott, 2004]. One of the approaches is to change the bandwidth value pointwise to adjust to varying density of data in different regions of the input space [Breiman et al., 1977; Hu et al., 2012]. Another approach assumes the bandwidth that depends on the estimation point [Scott, 1992]. In this case, the width of the kernel varies to catch a certain number of the neighboring points. In fact, this technique can be called k -nearest neighbor kernel density estimate. Though, asymptotically this is the best possible estimate of \mathbf{h} , similarly to k -NN, the resulting density estimate is not a true density function.

The problem of finding an optimal width of histogram bins can also be solved adaptively. The Fraser-Swinney algorithm hierarchically divides the plane into bins until they become uniform [Fraser & Swinney, 1986]. Alternatively, the bins with the equal number of samples can be constructed [Scott, 1992; Cellucci et al., 2005]. The latter estimator was reported to have similar performance to the more complicated Fraser-Swinney algorithm, which in turn was reported to perform worse than the kernel method [Silverman, 1986; Moon et al., 1995]. There are techniques which address adaptive partitioning in the multidimensional space for estimating the entropy and the mutual information [Darbellay & Vajda, 1999; Stowell & Plumbley, 2009] providing faster convergence with respect to the sample size compared with non-adaptive case [Trappenberg et al., 2006].

2.5.2 Nonplug-in approaches

Methods, which are classified as nonplug-in, usually directly solve a problem of estimating entropy rather than a probability density function, though both approaches are often related.

One of such examples is *k-nearest neighbor entropy estimation* which can be seen as an extension of the plug-in approach using k -nearest neighbor density estimation. The entropy estimator for multivariate densities and $k = 1$ was first introduced by Kozachenko and Leonenko [Kozachenko & Leonenko, 1987]. Its generalization for multiple nearest

neighbors, i. e. for $k > 1$, was developed later and it has the following form [Singh et al., 2003; Leonenko et al., 2008]:

$$\hat{H}_{kNN}(\mathbf{X}) = -\frac{N}{T} \sum_{i=1}^T \log \frac{1}{TV(\mathbf{x}_i)} - \Psi(k), \quad (2.74)$$

where $V(\mathbf{x}_i)$ is a volume of N -dimensional sphere with a radius defined as a distance from \mathbf{x}_i to its k^{th} nearest neighbor (2.69) and $\Psi(\cdot)$ is the digamma function, $\Psi(z) = \frac{\Gamma'(z)}{\Gamma(z)}$. Note that this is in fact the resubstitution plug-in entropy estimator with k -NN density estimate, whose bias is corrected by $(-\Psi(k) + \log(k))$. With this correction the estimator is proven to be asymptotically unbiased [Singh et al., 2003].

There is a related estimator based on k spacings, which is however applicable only to one-dimensional variables [Vasicek, 1976; Dudewicz & van der Meulen, 1981]:

$$\hat{H}_{sp}(X) = -\frac{1}{T-k} \sum_{i=1}^{T-k} \log \frac{k}{T(x_{i+k:T} - x_{i:T})} - \Psi(k) + \log k, \quad (2.75)$$

where $x_{1:T} \leq x_{2:T} \leq \dots \leq x_{T:T}$ are ordered statistics of x_1, x_2, \dots, x_T .

While estimating mutual information using k -NN entropy estimates, it was noted that a fixed value of k for $H(\mathbf{X})$, $H(\mathbf{Y})$ and $H(\mathbf{X}, \mathbf{Y})$ leads to biased results because distances to the k^{th} neighbor in marginal and joint spaces are different [Kraskov et al., 2004]. As a solution, Kraskov and colleagues proposed the following scheme. A distance r_k to the k^{th} nearest neighbor is defined in the joint space $\mathbf{Z} = (\mathbf{X}, \mathbf{Y})$ with the maximum norm $d(\mathbf{z}_i, \mathbf{z}') = \max\{\|\mathbf{x}_i, \mathbf{x}'\|, \|\mathbf{y}_i, \mathbf{y}'\|\}$. Then, in the marginal spaces, instead of setting k and measuring the distances from \mathbf{x}_i and \mathbf{y}_i to their k^{th} neighbors, one counts the neighbors $t_x(i)$ and $t_y(i)$ within the radius r_k defined in their joint space. The resulting estimator of mutual information is

$$I(\mathbf{X}, \mathbf{Y}) = \Psi(k) - \frac{1}{k} - \frac{1}{T} \sum_{i=1}^T [\Psi(t_x(i)) + \Psi(t_y(i))] + \Psi(T), \quad (2.76)$$

with the following generalization to the N -variate mutual information:

$$I(\mathbf{X}_1, \dots, \mathbf{X}_N) = \Psi(k) - \frac{N-1}{k} - \frac{1}{T} \sum_{i=1}^T [\Psi(t_{x_1}(i)) + \dots + \Psi(t_{x_N}(i))] + (N-1)\Psi(T). \quad (2.77)$$

The authors numerically proved that the proposed modification reduces the bias, especially for a small number of training samples. Comparative studies report good accuracy of Kraskov's nearest neighbor algorithm and KDE in contrast to simple and adaptive histograms for estimating pairwise mutual information for general purpose [Khan et al., 2007; Schaffernicht et al., 2010]. In the case of information-based feature selection, not

the absolute values of the multivariate mutual information estimates but the correct ranking of features according to their mutual information with the class is important. Within such setup, Kraskov estimator was shown to outperform KDE. It was also noted that both methods are quite sensitive to the choice of their smoothing parameters, which results in selecting different feature subsets for different parameter settings [Doquire & Verleysen, 2012].

ML-based estimators. As was already mentioned, the entropy estimator H_{MLE} based on the ML density estimates has a negative bias [Harris, 1975]:

$$\mathbb{E}[\hat{H}_{MLE}(X)] = H(X) - \frac{N-1}{2T} + \frac{1}{12T^2} \left(1 - \sum_{i=1}^N \frac{1}{p(x=r_i)} \right) + O(T^{-3}). \quad (2.78)$$

An attempt to reduce this bias has led to various correction techniques based for example on a series expansion of the bias [Miller, 1955; Carlton, 1969; Treves & Panzeri, 1995; Victor, 2000; Hacine-Gharbi et al., 2012]. Their classical representative is the Miller-Madow estimator that provides the $O(T^{-1})$ correction term of the form $\frac{L-1}{2T}$, where L is a number of discrete values of x with observed non-zero probabilities [Miller, 1955]. Unfortunately, such correction does not fully cancel the bias. An alternative approach is to fit the $O(T^{-1})$ and $O(T^{-2})$ from the data [Strong et al., 1998]. For the undersampled regime, Paninski has proposed an estimator based on a polynomial approximation of the entropy with the $O(T^{-2})$ bias [Paninski, 2003]. The key idea is to find the expansion coefficients that provide the best trade-off between the variance and the bias.

With an assumption that possible outcomes r_i of x follow the Poisson distribution, the Grassberger estimator [Grassberger, 1988] and its later improvement [Grassberger, 2003] were shown to be asymptotically unbiased for large T and to be less biased than Miller-Madow estimator in the undersampled regime.

Wolpert and Wolf proposed a *bayesian method of estimating entropy* of discrete variables [Wolpert & Wolf, 1995]. Suppose there are T observations of a variable x that can take N different values $\{r_1, \dots, r_N\}$. Let x_i denote the number of the samples with the outcome r_i , $\sum_{i=1}^N x_i = T$, and let p_i denote the probability of observing this outcome. Then, the vector of counts $\mathbf{x} = (x_1, \dots, x_N)$ is said to be multinomially distributed.

A quantity of interest is the entropy $H(\rho)$ which is estimated from the data by $\hat{H}(\rho|\mathbf{x})$. For this estimate, we need a posterior probability $p(\rho|\mathbf{x})$ that can be specified according to the Bayes rule as $p(\rho|\mathbf{x}) \propto p(\mathbf{x}|\rho)p(\rho)$ with the data likelihood:

$$p(\mathbf{x}|\rho) = T! \prod_{i=1}^N \frac{\rho_i^{x_i}}{x_i!}.$$

and the Dirichlet prior, which is a conjugate prior of the multinomially distributed likelihood¹:

$$p(\rho) = \frac{1}{B(\beta)} \prod_{i=1}^N x_i^{\beta_i} = \frac{\Gamma(\sum_i \beta_i)}{\prod_i \Gamma(\beta_i)} \prod_{i=1}^N x_i^{\beta_i}. \quad (2.79)$$

Here, $B(\cdot)$ and $\Gamma(\cdot)$ are the beta and the gamma functions, respectively, and β is a parameter vector of the Dirichlet distribution.

It was suggested that instead of estimating the entropy of the distribution $p(\rho|\mathbf{x})$, it is easier to estimate its moments [Wolpert & Wolf, 1995]. For the uniform prior with $\beta_i = 1$, the k^{th} moment of the entropy can be found as follows:

$$\hat{H}_k(\rho|\mathbf{x}) = \int \hat{H}^k(\rho|\mathbf{x}) p(\rho|\mathbf{x}) d\rho = \frac{\Gamma(T+N)}{\prod_i \Gamma(\mathbf{x}_i + 1)} \int \hat{H}^k(\rho|\mathbf{x}) p(\rho) \prod_i \rho_i^{\mathbf{x}_i} d\rho. \quad (2.80)$$

For $k = 1$, the Bayesian estimator of the entropy mean $\hat{H}(\rho|\mathbf{x})$, which is the best guess in terms of the mean squared error, is:

$$\hat{H}(\rho|\mathbf{x}) = - \sum_i \frac{x_i + 1}{T + N} (\Psi(x_i + 2) - \Psi(T + N + 1)). \quad (2.81)$$

Following the same procedure, one can estimate moments of any function of $p(\rho|\mathbf{x})$. Thus, for $f(p(\rho|\mathbf{x})) = p(\rho|\mathbf{x})$, we have

$$p(\rho|\mathbf{x}) = \frac{x_i + \beta_i}{T + N\beta_i}. \quad (2.82)$$

Note that for $\beta = 0$, this corresponds to the maximum likelihood estimate discussed before. $\beta = \frac{1}{2}$ gives the Jeffreys' or Krichevsky-Trofimov probability estimator [Jeffreys, 1946; Krichevsky & Trofimov, 1981] and $\beta = \frac{1}{N}$ gives the Schürmann-Grassberger probability estimator [Schürmann & Grassberger, 1996].

As was already mentioned, the bayesian estimates are quite sensitive to the choice of the prior distribution in the undersampled regime. For example, it was shown that a fixed value of the parameter β almost uniquely defines the entropy and as a remedy the improved estimator was suggested [Nemenman et al., 2002]. The *Nemenman-Shafee-Bialek estimator* specifies a near flat prior distribution $p_{NSB}(\rho)$, which is a mixture of Dirichlet priors with different β :

$$p_{NSB}(\rho) \propto \int \frac{d\xi}{d\beta} p_{\beta}(\rho) d\beta. \quad (2.83)$$

¹ A prior is called a conjugate prior for the likelihood if it belongs to the same family of distributions as the posterior.

Here, $p_\beta(\rho)$ is the prior defined in the expression (2.79) for the fixed value of β and $\xi := \mathbb{E}_\beta[H(\rho)]$. Acting as mixing coefficients, the fractions $\frac{d\mathbb{E}_\beta[H(\rho)]}{d\beta}$ make the contribution of every $p_\beta(\rho)$ depend on the degree of peakedness of the average entropy for the current value of β . Utilizing this prior, we obtain the Nemenman-Shafee-Bialek entropy estimator:

$$\hat{H}_{NSB}(\rho|\mathbf{x}) = \frac{\int \hat{H}_\beta(\rho|\mathbf{x}) q(\xi, \mathbf{x}) d\xi}{\int q(\xi, \mathbf{x}) d\xi}, \quad (2.84)$$

with $q(\xi, \mathbf{x}) = p(\beta(\xi)) \frac{\Gamma(N\beta(\xi))}{\Gamma(T+N\beta(\xi))} \prod_{i=1}^N \frac{\Gamma(x_i+\beta(\xi))}{\Gamma(\beta(\xi))}$ and $\hat{H}_\beta(\rho|\mathbf{x})$ is the Bayesian entropy estimate defined by (2.81) for the fixed β . The NSB-estimator shows good performance in terms of bias and robustness compared to ML-based entropy estimators and it is one of the most popular entropy estimators for discretized data. However, it is computationally expensive due to the necessity of averaging over β or ξ [Panzeri et al., 2007; Montani et al., 2007].

Estimators of differential entropy and mutual information using cumulant expansions are popular in the field of independent components analysis. On the one hand, they are quite simple, but on the other hand, their estimates are rather rough [Amari et al., 1996; Hyvärinen et al., 2004; Steuer et al., 2002]. Recently an extension of one of such techniques for multivariate entropy based on Edgeworth expansion has been proposed. The authors reported the accuracy comparable with Kraskov's multivariate variant of k -NN [Van Hulle, 2005].

2.6 Approximated schemes

All probability density and entropy estimation techniques, which are described above, require considerable amount of training data in order to achieve good accuracy. As was already stated, in case of multivariate data, the necessary amount of data is even larger. The feature selection criterion based on mutual information is of the form $I(C; F_k | F_{\alpha_1}, \dots, F_{\alpha_i})$. It means that a dimensionality of the involved entropies or equivalently pdfs grows iteratively with selecting new features. Therefore, there have been a lot of attempts to approximate the multivariate conditional mutual information using only pairwise or triplewise estimates. The conditional mutual information selection criterion can be rewritten as:

$$\begin{aligned} S(C, F_{\alpha_1}, \dots, F_{\alpha_i}, F_k) &= I(C; F_k | F_{\alpha_1}, \dots, F_{\alpha_i}) = \\ I(C; F_k) - I(F_k; F_{\alpha_1}, \dots, F_{\alpha_i}) &+ I(F_k; F_{\alpha_1}, \dots, F_{\alpha_i} | C). \end{aligned} \quad (2.85)$$

Two assumptions have to be introduced in order to reduce the dimensionality of the mutual information terms: the already selected features are independent of the feature-candidate

and they are independent of the feature-candidate and the class. Incorporating these assumptions, after some transformations we obtain the following simplified scheme [Brown et al., 2012]:

$$I_{appr}(C; F_k | F_{\alpha_1}, \dots, F_{\alpha_i}) = I(C; F_k) - \sum_{q=1}^i \{I(F_k; F_{\alpha_q}) - I(F_k; F_{\alpha_q} | C)\}. \quad (2.86)$$

Such approximation was used as a criterion for both feature selection [El Akadi et al., 2008; Guo & Nixon, 2009] and feature extraction [Lin & Tang, 2006]. While the first term measures the relevance of the feature-candidate for classification, the second term represents the approximated redundancy of this feature with respect to the selected features. Some schemes introduce a weight parameter β for the redundancy term:

$$I_{appr}(C; F_k | F_{\alpha_1}, \dots, F_{\alpha_i}) = I(C; F_k) - \beta \sum_{q=1}^i \{I(F_k; F_{\alpha_q}) - I(F_k; F_{\alpha_q} | C)\}. \quad (2.87)$$

It can be shown that with $\beta = \frac{1}{i}$, maximization of (2.87) corresponds to the maximization of $\sum_{q=1}^i I(F_k; F_{\alpha_q} | C)$, which is the so-called joint mutual information (JMI) selection criterion [Yang & Moody, 1999]. Intuitively, as i grows, such β expresses the belief that the redundancy term becomes less significant [Brown et al., 2012].

Substituting the summation in the redundancy term with the maximum operator, after some trivial transformations one obtains the following selection criterion known as conditional mutual information maximization (CMIM) [Fleuret & Guyon, 2004]:

$$S_{CMIM}(C, F_{\alpha_1}, \dots, F_{\alpha_i}, F_k) = \min_{q=1, \dots, i} \{I(F_k; F_{\alpha_q} | C)\}. \quad (2.88)$$

Obviously, in order to get rid of triplewise information terms, further simplifying assumptions should be introduced. Assuming that the relation $\frac{I(F_{\alpha_i}; F_k | C)}{I(F_{\alpha_i}; F_k)} = \frac{H(F_{\alpha_i} | C)}{H(F_{\alpha_i})}$ holds, $I(F_k; F_{\alpha_q} | C)$ can be represented as

$$I(F_k; F_{\alpha_q} | C) = \frac{H(F_{\alpha_q} | C)}{H(F_{\alpha_q})} I(F_k; F_{\alpha_q}),$$

The assumption is true for the cases when F_k is uniformly informative for all classes. The final approximation is

$$S_{MIFS-U}(C, F_{\alpha_1}, \dots, F_{\alpha_i}, F_{\alpha_q}) = I(F_k; C) - \beta \sum_{j=1}^i \frac{I(C; F_{\alpha_j})}{H(F_{\alpha_j})} I(F_k; F_{\alpha_j}), \quad (2.89)$$

which is used by MIFS-U algorithm (mutual information feature selection uniform) [Kwak & Choi, 2002b]. Here, the parameter β is not fixed but tunable.

Instead of the uniformity assumption, Battiti assumed that all features are pairwise independent given the class [Battiti, 1994]. As a result, the last term $I(F_k; F_{\alpha_q} | C)$ in (2.87) turns to zero and Battiti's MIFS criterion becomes:

$$S_{MIFS}(C, F_{\alpha_1}, \dots, F_{\alpha_i}, F_k) = I(F_k; C) - \beta \sum_{j=1}^i I(F_k; F_{\alpha_j}). \quad (2.90)$$

The minimum redundancy maximum relevance (MRMR) feature selection algorithm is identical to MIFS with one difference that it uses the adaptive value of $\beta = \frac{1}{i}$ [Peng et al., 2005].

Finally, the assumption that features are independent gives the simple ranking criterion based on the mutual information

$$S_{MIM}(C, F_{\alpha_1}, \dots, F_{\alpha_i}, F_k) = I(F_k; C), \quad (2.91)$$

which is known as the mutual information maximization [Lewis, 1962].

There are many other algorithms that approximate the conditional mutual information with low-order information and entropy terms in the way to obtain good classification performance with the selected features [Duch, 2006; Vidal-Naquet & Ullman, 2003; Meyer & Bontempi, 2006; Estevez et al., 2009; Cheng et al., 2011; Yu & Lee, 2012]. Efficiency of different approximations depends on dependencies between features in a given problem. However, it is not surprising that more complex schemes that take into account higher order interactions between features perform better than simpler algorithms. For example, JMI criterion demonstrated better results than CMIM, MRMR and MIFS [Brown, 2009]. In turn, Liu and colleagues showed that CMIM outperforms MIM, MIFS and MIFS-U [Liu et al., 2008]. Among the schemes, only CMIM employs a mutual information of three variables, while MIM, MIFS and MIFS-U use only pairwise terms in their approximations. Along the same lines are results presented by Kwak and Choi who used kernel density method to estimate the original conditional mutual information criterion. Their algorithm was shown to outperform MIFS and MIFS-U [Kwak & Choi, 2002a]. Despite such results, in practice, simple approximate schemes are usually preferred due to the reduced computational complexity.

2.7 Conclusion

In this section, we presented the conventional approach to feature selection that assumes reducing dimensionality of the input space before building a classifier. As a result, a learning problem is significantly simplified and a classification rule has better generalization

ability. This is especially important in the undersampled regime, i. e. when a number of features is larger than the number of training samples. Moreover, performing feature selection improves interpretability of data by highlighting the important input dimensions.

An emphasis of the presentation was put on feature selection algorithms of the filter type whose selection criteria are independent of a classifier to be used. The problem of feature selection was introduced within a framework of the sequential uncertainty reduction that assumes iterative selection of features informative with respect to a class variable but non-redundant with respect to the previously selected features. This framework was adopted to preserve an analogy to the sequential process of the hypothesis checking mechanism in the human visual system, which serves as an inspiration for the feature selection scheme proposed later in this thesis.

Feature selection algorithms have two key components, a search strategy and a selection criterion. As the search strategy is set to the sequential feedforward search by design, we reviewed possible uncertainty functions that can make up a selection criterion in the uncertainty reduction framework. For example, the misclassification error is an intuitive choice for selecting features that can be useful for classification. The Gini index for feature selection was initially derived heuristically as an uncertainty function corresponding to the certain requirements, though it is widely used in statistics as a diversity measure. In contrast, Shannon entropy is a fundamental measure of uncertainty in information theory and therefore it is a natural candidate for the uncertainty function. Moreover, mutual information, the entropy-based selection criterion, is able to measure nonlinear interactions. It was also shown that the entropy-based selection criterion favors features that can discriminate well between classes. Two generalizations of Shannon entropy, namely Rényi and Tsallis entropies, give an additional possibility to adjust an uncertainty measure to the structure of data and therefore they can be useful in feature selection for complex systems. There are other popular selection criteria that however do not fit in the framework of the sequential uncertainty reduction. Among them are different statistical tests of independence used for feature ranking and probabilistic distance measures that select features ensuring maximal distance between distributions of different classes.

Shannon entropy as an uncertainty function and mutual information as a corresponding selection criterion are widely used in feature selection due to the above-mentioned advantages like information-theoretical interpretation and usefulness for classification. The feature selection algorithm proposed in this thesis employs mutual information as well. However, estimation of mutual information is known to be a difficult problem. We presented a review of various estimation techniques ranging from relatively simple plug-in schemes utilizing nonparametric density estimators to complex algorithms trying directly to build unbiased estimators of entropy and mutual information. Kernel density method can be named as one of the successful representatives of the first group, whereas among the second group one can distinguish Kraskov's modification of k -NN and the Nemenman-

Shafee bayesian estimator. Since the estimation problem becomes even harder in higher dimensions, there are a lot of selection criteria that use low-order approximations of mutual information. Their efficiency of course depends on whether simplifying assumptions used in approximations contradict data describing a problem at hand.

Despite all difficulties with estimating entropy and related concepts, information-theoretical feature selection is popular in practice and it remains an active area of research. Moreover, using mutual information as a selection criterion, one does not need its precise values. Therefore, even if an estimator is biased, it is only important to have the right ordering of features according to their mutual information with a class. This peculiarity significantly reduces requirements to the quality of estimates.

Chapter 3

Adaptive feature selection

3.1 Biological motivation

The human visual system is characterized by a large number of connections going backwards along its hierarchy. Moreover, feedback is observed even on the lowest levels of visual processing. These facts gave rise to numerous investigations of functional roles of the top-down information flow in visual perception.

For example, feedback connections exist already in the retina, an inner tissue of the eye that analyzes spatial and time variations of light and sends this information further to the brain. Though, possible functions of the feedback from higher retinal layers and from the brain are not well-understood, it is hypothesized that it may influence adaptation processes of neurons in the retina [Kolb, 2011].

The primary visual cortex (V1), where the analysis of color, shape and orientation occurs, is a key component of many feedforward-feedback loops in the visual system. V1 sends the feedforward signal to and receives the excitatory feedback from areas of both dorsal and ventral visual pathways, which process information about location and complex characteristics of a stimulus, respectively [Mishkin & Ungerleider, 1982].

It is known that a size of receptive fields grows successively as one goes up along the hierarchy of the visual areas. Hence, feedback to a cell from the higher area is usually integrated over the larger region of the visual field compared to the visual region visible by this cell [Livingstone & Hubel, 1987; Van Essen et al., 1990]. The complexity of features to which cells are tuned on the subsequent levels grows successively as well. It means that cells on the higher levels are usually sensitive to different combinations of features, to which cells on the preceding layers respond. As a result, the higher-level cells extract more abstract representations that are invariant under scale, orientation and

shift transformations. Therefore, by sending feedback to the cells on the lower levels, they provide context, i. e. more global information. In this way, the feedback enhances activation of cells tuned to simple features in order to improve their further binding into more complex structures on the higher levels [Desimone & Duncan, 1995; Murphy et al., 2000; Sillito et al., 2006]. This function is also performed by long-range excitatory lateral connections operating within one visual layer. While such connections enhance grouping for many complex features at the same time, it is important to note that only a small fraction of feedback connections are active simultaneously. The reason for this is limited processing resources of the visual system that force high-level concepts to compete for representation [Van Essen et al., 1991]. Therefore, only few winners of such competition send their modulating feedback to the lower levels at once [Macknik & Martinez-Conde, 2009]. Let us look at the two main examples of the feedforward-feedback loops in which V1 is involved.

On the early stages of visual processing, the area V1 receives its feedforward input from the lateral geniculate nucleus (LGN) in the thalamus, which processes a visual scene based on the retinal output. At the same time, the primary visual cortex sends the massive excitatory feedback to LGN, which constitutes about 30% of its input. For comparison, LGN receives only about 10% of its input from the retina [Montero, 1991; Wilson, 1993; Sherman, 2001]. It is known that almost all feedback connections from V1 to LGN originate from orientation- and direction-selective cells [Grieve & Sillito, 1995]. Moreover, within the dorsal pathway, the same cells in V1 receive their excitatory feedback from the higher area MT, which is responsible for motion analysis [Rosa, 2002]. It should be noted that MT itself depends on the activity in V1, which is sent to it via feedforward circuits. Thus, the signal going from MT to V1 and further to LGN facilitates movement and orientation processing as well as contrast perception. Therefore, such feedback plays an important role in the contextual and spatial attentional modulation [Murphy et al., 1999; Jones et al., 2000; Hupe et al., 2001; Sillito et al., 2006].

Within the ventral pathway, the known feedforward-feedback loop consists of the areas V1, V2, V4 and the inferior temporal cortex (IT). The first three areas analyze such stimulus characteristics like size, shape, color with increasing complexity, whereas IT responds already to complex objects and therefore it is supposed to be involved in object recognition and identification, as well as in formation of the visual memory. Activity in IT is in turn modulated by the feedback signal from the prefrontal cortex, which is important for decision making, behavioral planning and where the working memory might be formed [Chelazzi et al., 1993; Schall et al., 1995; Yang & Raine, 2009]. Therefore, it is believed that feedback going along this pathway back to LGN modulates binding of high-level features that are relevant for a current task, such as recognition [Chelazzi et al., 1998; Miller & Cohen, 2001; Herd et al., 2006]. This feedback modulation is also called object-based attention [Fink et al., 1997; Valdes-Sosa et al., 1998].

Based on these facts, there is a hypothesis that the area V1 together with the subsequent area V2 act as an active blackboard, which integrates the results of information processing from the preceding and subsequent areas and sends it further along the feedforward circuits [Mumford, 1991; Bullier, 2001].

In line with this idea, Lee and Mumford suggested a theory describing the nature of information processing in the ventral pathway [Lee & Mumford, 2003]. Inspired by the hierarchical Bayesian inference, it states that higher areas of the visual system generate a hypothesis about a visual scene on the basis of information sent from the lower areas via bottom-up circuits. If this information is not enough for unambiguous recognition, i. e. there are several competing hypotheses about a visual scene, the feedback or top-down signal is sent back to enhance processing of those pieces of the visual input that can help to reduce the uncertainty. The whole process can be seen as inference which is based on the scene-specific bottom-up information integrated with the top-down contextual prior. It is important to note that neither bottom-up nor top-down components are static. A feedback signal modulates processes on the preceding levels, which in turn influence the refinement of the current hypotheses about a visual scene. There is experimental evidence that such iterative and bidirectional interactions happen in parallel between the adjacent areas as sending information back and forth along several areas would take long [Bichot & Schall, 1999; Lee et al., 2002]. In the computational literature, this hypothesis refinement mechanism is known as “adaptive resonance” of Grossberg [Grossberg, 1976], which inspired also neural models of cortical interactions, e. g. LAMINART [Raizada & Grossberg, 2003].

Although many studies demonstrated that object recognition and categorization could be done purely in the feedforward fashion [Riesenhuber & Poggio, 1999; Serre et al., 2007; VanRullen, 2007], it is obvious that feedback is important if the information sent via bottom-up circuits is ambiguous or imprecise [Wyatte et al., 2012]. This fact is also supported by neurophysiological experiments showing that later activity in V1 and V2 as well in the prefrontal cortex have influence on recognition [Bar et al., 2006; Koivisto et al., 2011], hence the feedback is involved in this process. Studies using such psychophysical experimental paradigm as object substitution masking also demonstrate that visual perception involves bidirectional interactions between the lower and higher areas of the visual system. Masking experiments show that perception of a stimulus can be impaired by another stimulus, a so-called mask, if the latter is presented shortly after the first stimulus [Di Lollo et al., 2000; Enns & Di Lollo, 2000; Elze et al., 2011]. This phenomenon suggests that while the first stimulus is not yet fully recognized and the higher areas have to send a top-down signal to request the additional information, the lower areas are already activated by the second stimulus. As a result, the next portion of the visual input will correspond to the later stimulus and the inference of the first stimulus will be interfered. Moreover, Elze and colleagues in their masking experiment showed that the

prior information influences very early stages of visual perception, which is likely to be sent by a feedback signal from the higher brain areas [Elze et al., 2011].

To conclude, one of the most prominent functions of the extensive feedback from the higher brain areas within and outside of the visual cortex is attentional modulation. While in the dorsal or “where” pathway top-down signals are involved in the spatial modulation, the feedback within the ventral or “what” pathway helps the brain to concentrate its resources on visual features that are relevant for recognition or categorization. Such feedback iteratively selects certain aspects of the visual scene for refined processing by the lower areas until the inference process in the higher areas converges to a single hypothesis about this scene.

3.2 Adaptive feature selection

Obviously, it is desirable to minimize a number of required selection-refinement iterations before the final recognition of a visual scene. For this, one has to find a short sequence of maximally informative portions of the visual input. As was already mentioned, the feedback is not static and therefore the selection process is adapted to a visual scene that should be recognized. To find a scene-specific subset of informative patches, the adaptive selection process on every iteration utilizes results of previous processing in order to reduce the remaining uncertainty about the visual scene. Therefore, every next portion of the input is chosen as the most informative for the current hypothesis space, which is refined based on information extracted on preceding iterations.

Let us think about a visual input divided into patches as an object described by a set of its features. Then, the mechanism deciding which portion to process next is nothing else but a feature selection algorithm. Further, suppose that for a certain classification problem there is a preselected set of informative features. That is, feature selection is performed during the learning phase before actual classification starts. And once learning is finished, the resulting set of informative features is fixed and it will be used for classification of all samples in the future. Obviously, within such setup features are selected to be *on average discriminative for all samples*. Though, the selected subset includes only a part of all available features, it can still be large especially if the data has inhomogeneous structure.

Now, suppose that we are classifying a particular sample. Already after the first iteration, there is partial knowledge about the testing sample, which is formed as the result of processing the first features. This sample-specific information is then used to refine the initial hypotheses about the possible class label, i. e. to update the prior uncertainty about the class. Intuitively, it would be reasonable to select next features that can reduce the uncertainty for this particular sample rather than for some average representative of

the training set. In such case, as we do not try to generalize, it is likely to end up with a smaller number of relevant features required to make a confident classification decision. This way of selecting features we call adaptive. The conventional approach will be further called “static” as a subset of relevant features remains fixed during the classification process.

Thus, adaptivity in feature selection means the following. For a certain testing sample, every selected feature should yield the maximum reduction in uncertainty about the class, which is iteratively updated with the values of already selected features observed on this testing sample. Then, according to the adaptive approach, every testing sample is classified with the unique subset of discriminative features. On the one hand, this is obviously more computationally demanding compared to selecting a single static feature subset. On the other hand, the adaptive methods have the potential to produce smaller feature subsets.

When do feature subsets selected in the static and the adaptive way differ significantly in their size? As was already mentioned, conventional feature selection methods can fail to select a small number of relevant features when data are heterogeneous. That is, the structure of the data can vary in different subregions of the input space and therefore every subregion is characterized by different features. For example, one may need different features to discriminate between classes, or even different objects belonging to one class may have different discriminative features. One can partially overcome this problem by forming a collection of all relevant feature subsets. This, however, will lead to an increase in the classifier complexity, which in turn increases the amount of data necessary for training [Raudys & Jain, 1991]. Thus, conventional feature selection schemes, which select a fixed subset of features before they are handed to a classifier, can be inefficient.

In addition, we suggest that the adaptive approach to feature selection is advantageous in situations when the amount of data is limited, especially if the number of features exceeds the number of training samples, hence, a classification problem is difficult. The reason for this is the following. It is known that a small sample size impairs the estimation accuracy of estimates of a selection criterion [Raudys & Jain, 1991]. Recall that the static scheme tries to generalize and therefore it selects features that are informative for the whole input space, reconstructed from a training set. At the same time, the adaptive scheme performs local feature selection, i.e. selects features that should be discriminative for a certain subregion of the input space. Moreover, this subregion is iteratively refined using values of the previously selected features, which simplifies a classification problem on every iteration. Thus, the adaptive scheme usually evaluates relevance of features only in the small input subregion for a small subset of classes to which a testing sample can belong. Therefore, in the undersampled regime, when the training set does not fully represent the true data distribution, we expect estimates of the local feature relevance used by the adaptive approach to be more accurate compared to estimates of the global relevance utilized by static selection schemes. Consequently, as a quality of adaptively

selected features is higher, a smaller subset of them will be enough to achieve the same classification performance compared to a subset of statically selected features.

Thereby, in cases when it is difficult to find a small fixed subset of relevant features, we propose to use different features for every testing sample, i. e. select relevant features in the “adaptive” manner. In addition, by analogy with the visual processing, adaptive feature selection can be useful for systems where evaluation of every additional feature is associated with considerable costs, which might be much higher compared to additional computations due to the adaptive selection.

3.3 Framework

Let us adjust the standard feature selection framework presented in the previous chapter to the adaptive approach. Suppose that we have a testing sample ξ . Suppose also that after i steps we have selected the features $F_{\alpha_1}, \dots, F_{\alpha_i}$ and observed their values $\xi_{\alpha_1}, \dots, \xi_{\alpha_i}$ on this testing sample. Then, for this testing sample the next feature $F_{\alpha_{i+1}}$ is selected according to the adaptive criterion:

$$\alpha_{i+1} = \arg \max_k S(C, F_{\alpha_1} = \xi_{\alpha_1}, \dots, F_{\alpha_i} = \xi_{\alpha_i}, F_k). \quad (3.1)$$

In contrast to the static criterion (2.1), the adaptive criterion takes also into account the *values of the already selected features, which are observed on the current testing sample*. That is, every next feature should be relevant w. r. t. the class variable whose distribution is updated with the values of the already selected features observed on the current testing sample. In other words, the selected feature should be both relevant for classification and non-redundant w. r. t. the previously selected features taking values observed on the current testing sample.

Recall that the selection criterion was expressed in the form of uncertainty reduction:

$$\alpha_{i+1} = \arg \max_k \{U(C|\mathbf{F}^i) - U(C|F_k, \mathbf{F}^i)\}, \quad (3.2)$$

however, the selected features now takes particular values from the testing sample and therefore $\mathbf{F}^i = \{F_{\alpha_1} = \xi_{\alpha_1}, \dots, F_{\alpha_i} = \xi_{\alpha_i}\}$. In order to differentiate between \mathbf{F}^i for static and adaptive schemes, let us introduce $\xi^i = \{F_{\alpha_1} = \xi_{\alpha_1}, \dots, F_{\alpha_i} = \xi_{\alpha_i}\}$.

It is necessary to make a remark regarding the difference in estimating the uncertainty function for the static and adaptive schemes. In the adaptive case, since we observe a certain realization of the selected features on the testing sample ξ , the uncertainty depends on the class posteriors only w. r. t. to this testing sample ξ :

$$U(C|\xi^i) = \sum_{j=1}^m U(c_j | f_{\alpha_1} = \xi_{\alpha_1}, \dots, f_{\alpha_i} = \xi_{\alpha_i}) \quad (3.3)$$

At the same time, the static scheme aims for features useful for classification of all samples in general. Therefore, the uncertainty function has to be averaged over the joint space of the involved features:

$$U(C|\mathbf{F}^i) = \sum_{\mathcal{F}_{\alpha_1}, \dots, \mathcal{F}_{\alpha_i}} \sum_{j=1}^m p(f_{\alpha_1}, \dots, f_{\alpha_i}) U(c_j | f_{\alpha_1}, \dots, f_{\alpha_i}). \quad (3.4)$$

But note the uncertainty updated by the feature-candidate F_k :

$$\begin{aligned} U(C|F_k, \xi^i) &= \sum_{\mathcal{F}_k} \sum_{j=1}^m p(f_k | \xi^i) U(c_j | f_k, \xi^i), \\ U(C|F_k, \mathbf{F}^i) &= \sum_{\mathcal{F}_k, \mathcal{F}_{\alpha_1}, \dots, \mathcal{F}_{\alpha_i}} \sum_{j=1}^m p(f_k | \mathbf{F}^i) U(c_j | f_k, \mathbf{F}^i), \end{aligned} \quad (3.5)$$

where the adaptive scheme averages also over the space of the feature F_k because it has not yet seen values of the unselected features on the testing sample ξ .

3.3.1 Relation to complex adaptive systems

Our adaptive feature selection framework is inspired by the definition of complex adaptive systems given by Jost [Jost, 2004]. According to this definition, while operating an adaptive system tries to increase its external complexity and at the same time to decrease its internal complexity. In other words, the goal of the adaptive system is extracting as much information as possible from its environment, which is described by data \mathcal{X} , and representing this information internally in the most efficient way using some model θ . Then, formally one can express an adaptation problem in the following way:

$$\min_{\theta} \{-e(\theta) + i(\theta)\} = \min_{\theta} \{-\mathbb{E}_{p(\mathcal{X}|\theta)}[\log_2 p(\mathcal{X}|\theta)] - \log_2 p(\theta)\}, \quad (3.6)$$

where $e(\theta)$ and $i(\theta)$ correspond to the external and internal complexities, respectively, which are however optimized on different time scales. Here, $p(\mathcal{X}|\theta)$ is the probability of data \mathcal{X} given a model θ and $p(\theta)$ is the probability of this model itself. Then, the first term, $-e(\theta) = -\mathbb{E}_{p(\mathcal{X}|\theta)}[\log_2 p(\mathcal{X}|\theta)]$, is nothing else but the Shannon entropy of the data once they are processed by the model θ . It means that a model that we are looking for should maximally reduce our uncertainty about the environment. This is exactly the adaptive selection criterion proposed here, see the expression (3.2) together with (3.3). That is, if we take $U(C|\xi^i)$ as the current uncertainty of the system on the iteration i , then the next feature F_k should minimize $U(C|F_k, \xi^i)$, which would be the uncertainty about the environment on the next iteration ($i+1$) if this feature was selected.

The second term $i(\theta) = -\log_2 p(\theta)$ in (3.6) controls complexity of the model θ and can be seen as a regularization parameter. The simpler model is, the higher its probability is. Complexity of a model can be measured by a number of its parameters, e. g. like in the Akaike information criterion used in model selection [Akaike, 1974]. Recall that the aim of any feature selection algorithm is to find a minimal number of informative features, i. e. to find a model with a minimal number of parameters that efficiently describes the given data.

As a result, on every iteration, an adaptive feature selection algorithm within the proposed framework modifies a classification model¹ by adding a feature that can minimize the current uncertainty about the environment, i. e. a testing sample. In addition, but on the longer time scale, the adaptive algorithm tries to keep the structure of a classifier as simple as possible and therefore a number of selected features is minimized.

3.4 Existing algorithms

3.4.1 Local feature selection by decision trees

The idea of adaptive feature selection has some similarities with the so-called local feature selection. For situations when available training data are inhomogeneous and of high complexity, i. e. when there exists no unique relationship between features in all parts of the input space, algorithms performing local feature selection use the following approach. They divide the input space into homogeneous regions and then for every such region construct a separate classifier that catches local dependencies between features. Usually the same type of the classifier is used, however, for every region of the input space, this classifier is built using different features.

Classification decision trees are an example of models implementing this idea. The goal of the tree classifier is to partition the input space into pure regions with a minimal number of splits. However, for a multiclass problem, finding the smallest possible tree that partitions the training samples with a minimal error is proven to be a NP-complete problem [Tu & Chung, 1992]. Therefore, similar to sequential feedforward feature selection methods, decision trees perform the greedy local search. That is, instead of looking for globally the best set of splits, they start from the initial input space and recursively partitions it so that every selected split reduces maximally the impurity of the current partition. The most used decision tree algorithms are ID3 [Quinlan, 1986], its improvement C4.5 [Quinlan, 1993] and CART [Breiman et al., 1984].

¹ As the adaptive feature selection framework is rather general, here by a classifier we mean any estimator of the class posterior distribution.

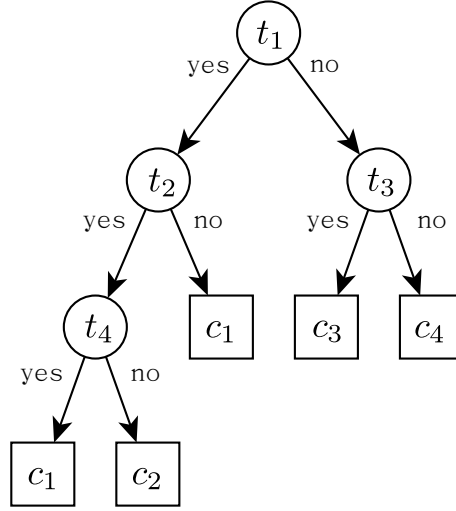


Figure 3.1: A binary decision tree where every split of the nonterminal node corresponds to some binary function, e. g. $t_1 : f_k > 0.5$, and terminal nodes correspond to the class labels.

A decision tree consists of a root, internal and terminal nodes. It is constructed by recursive partitioning of the input feature space into several descendant subspaces. That is, starting from the root, every nonterminal node t_i partitions the current feature space into further subspaces according to a test function associated with this node. The terminal subspaces correspond to the final partitions of the input space and have associated class labels c_1, \dots, c_m . Thus, all data points falling into the certain final partition are assigned to one class. It is usually the most common class among the samples in the partition or it is chosen to minimize the cross-validation classification error [Breiman et al., 1984].

Test functions of the internal nodes are usually binary functions of one argument. Then, the corresponding models are called univariate binary decision trees, see Figure 3.1 for an example of such a tree.

One of the main issues that should be addressed while constructing a decision tree is selection of splits, i. e. tests for every internal node. For univariate decision trees, the problem of split selection is nothing else but the previously introduced sequential feature selection, however for certain subregions of the input space defined by branches. Hence, it is called local feature selection.

Note that partitioning and associated with it feature selection are done during the learning phase. Therefore, the final partitioning should be general enough to achieve good classification performance. For this reason, decision trees need a rather large amount of data in order to define robust partitions of the input space corresponding to different classes.

In contrast, our adaptive approach to feature selection assumes the incremental refinement of that part of the input space where a testing sample lies and defines the subset of relevant features appropriate for this region. That is, we are not bounded to the fixed decision boundaries that are constructed offline using only training data. Instead, feature values observed on the testing sample are used to influence construction of the accurate classification rule with the minimal number of features.

3.4.2 Active testing model

There exists an adaptive scheme that selects features for every testing sample proposed by Geman and Jedynek [Geman & Jedynek, 1996]. Their so-called “Active testing model” (ATM) was developed specifically for online road tracking. Interestingly, the idea of adaptivity was also motivated by selective attention in the visual system which can be observed on the example of eye movements. The authors as well make a parallel to decision tree classifiers performing the local feature selection. For the problem of road tracking with a large number of possible roads, they claim that offline learning of such classifier based on the training data would lead to deep and bushy decision trees. Thus, the adaptive approach allows avoiding complex and lengthy learning and gives a possibility to build only those tree branches that are necessary for the testing samples.

The estimation of the feature (or tests as the original work states) selection criterion is based on the statistical model specific for their problem. Given one point on the road, the aim of the system is to extract this road from the satellite image based on small line-like segments called arcs. It is supposed that if all arcs are known, then the road ξ can be unambiguously tracked. However, a number of all possible segments on the satellite image is very large and evaluating all of them is not feasible for online tracking. As a solution, iterative testing is suggested, which is the same as sequential adaptive feature selection for a particular road ξ . That is, the arcs are sequentially selected and evaluated on the satellite image, and then this information is used to form hypotheses about the location of the road ξ . Obviously, the selected arcs should bring as much information as possible about the true hypothesis.

Formally, the problem is described by two classes $c \in \mathcal{C} = \{c_1, c_2\}$ representing roads and background and features $\mathbf{f} \in \mathcal{F} = \{f_1, \dots, f_n\}$ representing the arcs. Every feature has N discrete values, $f_i \in \{f_{i,1}, \dots, f_{i,N}\}$. The selection criterion is formulated as follows. For the road ξ , every feature selected on the iteration α_{i+1} should minimize the uncertainty about the road location updated after evaluating i previously selected features. This uncertainty is defined as the entropy of the class conditioned on the feature candidate F_k and the already selected features taking the values observed on ξ :

$$\alpha_{i+1} = \arg \min_k \{H(C|F_k, \xi^i)\}, \quad \xi^i = \{f_{\alpha_1} = \xi_{\alpha_1}, \dots, f_{\alpha_i} = \xi_{\alpha_i}\}. \quad (3.7)$$

Note that as mutual information between the class and the feature candidate F_k given already observed features is $I(C; F_k | \xi^i) = H(C | \xi^i) - H(C | F_k, \xi^i)$, maximizing $I(C; F_k | \xi^i)$ w. r. t. F_k is equivalent to minimizing $H(C | F_k, \xi^i)$.

In order to simplify the estimation of the selection criterion, the features are assumed to be conditionally independent given a class, i. e. $p(f_1, \dots, f_n | c) = \prod_{i=1}^n p(f_i | c)$. So the main consequence of this assumption is a possibility to estimate the multidimensional joint pfs of the features using their marginals. Applying the probability chain rule, this leads to the following form of the selection criterion:

$$\begin{aligned} \alpha_{i+1} = \arg \min_k \{H(C | F_k, \xi^i)\} &= \arg \min_k \{H(F_k | \xi^i, C) + H(C | \xi^i) - H(F_k | \xi^i)\} = \\ &= \arg \min_k \{H(F_k | \xi^i, C) - H(F_k | \xi^i)\} = \\ &= \arg \min_k \left\{ \sum_{j=1}^m \sum_{\mathcal{F}_k} p(c_j | \xi^i) p(f_k | c_j) (\log p(f_k | c_j) - \log p(f_k | c_j) p(c_j | \xi^i)) \right\}. \end{aligned} \quad (3.8)$$

The same independence assumption allows to estimate the class posterior $p(c_j | \mathbf{f}^i)$ recursively based on the estimates from the previous iterations:

$$p(c_j | \mathbf{f}^i) = \frac{p(\mathbf{f}^i | c_j) p(c_j)}{p(\mathbf{f}^i)} = \frac{p(f_i = \xi_i | c_j) p(\mathbf{f}^{i-1} | c_j) p(c_j)}{\sum_{j'=1}^m p(f_i = \xi_i | c_{j'}) p(\mathbf{f}^{i-1} | c_{j'}) p(c_{j'})} \quad (3.9)$$

The authors report efficiency of selected features. In addition, the model has comparatively good speed due to the introduced simplifying assumption about interdependencies of the features. Although this assumption is in reality violated also for the considered problem domain, it makes it possible for the active testing model to operate fast in online mode.

The active testing model with the Gini index as the uncertainty function was also successfully applied to face detection and localization [Sznitman & Jedynak, 2010]. Since a scale of a face is not known in advance, theoretically the whole image has to be inspected with filters of all possible sizes. Adaptivity in selecting tests allows a so-called coarse-to-fine face detection, i. e. starting with large-scale filters and further refining them only in locations where the posterior probability of face presence is high enough. Comparing to the exhaustive search, such adaptive approach gave an exponential gain in speed while preserving the detection performance.

3.4.3 Jiang's sequential feature selector

The idea of adaptive sequential feature selection based on mutual information was also used by Jiang [Jiang, 2008]. According to Jiang, systems implementing adaptive feature selection are better suitable for working in non-stationary conditions when the characteristics of the process change over time. As a subset of the informative features is defined during the testing stage, the selection process can incorporate changes in the statistics of the training data without additional efforts. While the general idea is within the introduced framework of the adaptive feature selection, the exact algorithm uses very rough approximation of the posterior probabilities of the classes which can negatively influence its performance for complex problems. Let us look at the algorithm in the details.

The general form of the selection criterion is the following:

$$S(C, \xi^i, f_k = \xi_k) = H(C|\xi^i) - H(C|f_k = \xi_k, \xi^i), \quad (3.10)$$

from which we see that the values of the feature-candidate on the testing sample ξ_k is known already during the selection process. This is in contrast to our idea stating that it is necessary to spend system resources to evaluate a feature value only if this feature is relevant for the task to be solved. This is crucial when the precise evaluation of features is costly as in the hierarchy of the human visual system.

Now consider the components of the selection criterion, namely $H(C|\xi^i)$ and $H(C|f_k = \xi_k, \xi^i)$. There is a particular way of estimating the class posterior distribution $p(c|\xi^i)$ which is used for further selection. Suppose we have selected the first feature F_{α_1} . Then, if the posterior of some class $p(c_j|f_{\alpha_1} = \xi_{\alpha_1})$ is below a certain threshold, then this class is excluded from the pool of the possible classes for the next selection iterations. And accordingly, the relevance of the features candidates on the next iteration is tested for the reduced set of the classes, which are reset to be uniformly distributed.

Formally, on the iteration i , there is a set of the currently active classes C^i with

$$p(c_j|\xi^i) = \frac{1}{|C^i|}, \quad \forall c_j \in C^i,$$

which implies that $H(C|\xi^i) = \log(|C^i|)$. $H(C|f_k = \xi_k, \xi^i)$ is defined in the following way:

$$H(C|f_k = \xi_k, \xi^i) = - \sum_{c_j \in C^i} p(c_j|f_k = \xi_k) \log p(c_j|f_k = \xi_k),$$

which together with the equiprobable classes gives:

$$H(C|f_k = \xi_k, \xi^i) = - \sum_{c_j \in C^i} \frac{p(f_k = \xi_k|c_j)}{\sum_{c_l \in C^i} p(f_k = \xi_k|c_l)} \log \frac{p(f_k = \xi_k|c_j)}{\sum_{c_l \in C^i} p(f_k = \xi_k|c_l)}.$$

Note that the previously selected features $F_{\alpha_1}, \dots, F_{\alpha_i}$ do not directly appear in the estimation of $H(C|f_k = \xi_k, \xi^i)$.

The final form of the selection criterion is then the following:

$$\begin{aligned} \alpha_{i+1} = \arg \max_k \{S(C, \xi^i, f_k = \xi_k)\} = \\ \arg \max_k \left\{ \log(|C^i|) + \sum_{c_j \in C^i} \frac{p(f_k = \xi_k | c_j)}{\sum_{c_l \in C^i} p(f_k = \xi_k | c_l)} \log \frac{p(f_k = \xi_k | c_j)}{\sum_{c_l \in C^i} p(f_k = \xi_k | c_l)} \right\}, \quad (3.11) \\ F_k \in \{F_1, \dots, F_n\} \setminus \{F_{\alpha_1}, \dots, F_{\alpha_i}\}. \end{aligned}$$

Once the feature $F_{\alpha_{i+1}}$ maximizing this selection criterion is found, the set of the classes is updated, i. e.

$$C^{\{i+1\}} = \{c_j\}, \quad \forall c_j, \quad \text{s.t.} \quad c_j \in C^i \quad \text{and} \quad p(c_j | f_{\alpha_{i+1}} = \xi_{\alpha_{i+1}}) > b,$$

where b is a threshold value.

One can see that the features selected on the previous iterations are used only to threshold the unlikely classes. Since the remaining classes are set to be uniformly distributed, the precise values of their posterior distribution are not further used. This iterative thresholding can be seen as a rough and greedy approximation to the multivariate class posteriors. Moreover, once the class is eliminated, it cannot be recovered again. And this decision is based on the value of the single feature that makes the scheme sensitive to the noisy data.

The suggested scheme in combination with the 1–NN classifier was used for classification of discrete data. The author reports its efficiency compared to the 1–NN with the arbitrary sequence of the available features for neural decoding problem. There, the assumptions that features are independent results in a good approximation due to the relatively large distance between electrodes with respect to the size of the recorded neurons. Taking this into account and the fact that the proposed way of estimating the selection criterion is oversimplified, we believe that for complex problems with many interdependent features the performance can be poor.

3.5 Adaptive conditional mutual information feature selector

In Section 3.3, we introduced the general framework of the adaptive feature selection. It was suggested that for classification problems with heterogeneous or small training sets it is advantageous to select features adaptively because it is possible to find a smaller number of discriminative features compared to the static approach. Heterogeneous data are likely to be described by different features in different subregions of the input space. Thus, adaptive selection gives a possibility to define a small subset of informative features depending on the location of a testing sample. In the case of limited training data, we expect estimates of local relevance of a feature, i. e. relevance for a small subregion of the input space defined by a certain testing sample, to be more accurate than estimates of global feature relevance, which is used by static schemes. As a result, the adaptive scheme would select features of better quality. Consequently, in order to reach the same classification accuracy, one would need a smaller number of adaptively selected features comparing to static selection schemes.

Thus, one does not predefine a single subset of relevant features but rather selects a specific one for every new testing sample. The proposed approach assumes a sequential feedforward feature selection where every next feature added to the subset should be discriminative and non-redundant w.r.t the already selected features, which take particular values observed on the current testing sample [Avdiyenko et al., 2012c,b].

3.5.1 Model

Now, we present a particular algorithm realizing the idea of adaptive sequential feature selection. The proposed method uses a selection criterion based on the mutual information of the features and class variables [Cover & Thomas, 1991]. As a subset of informative features is formed sequentially, the method iteratively looks for a feature F_k that has the maximal mutual information with the class variable conditioned on the outcomes of the selected features, which are observed on the testing sample. The selection criterion is defined as $I(C; F_k | \xi^i)$, which is mutual information of C and F_k conditioned on the values of previously selected features ξ^i . Therefore, the method is called adaptive conditional mutual information feature selector (ACMIFS).

Formally, according to ACMIFS every selected feature should satisfy the following:

$$\alpha_{i+1} = \arg \max_k \{S(C, F_{\alpha_1} = \xi_1, \dots, F_{\alpha_i} = \xi_{\alpha_i}, F_k)\} = \arg \max_k \{I(C, F_k | \xi^i)\} = \arg \max_k \left\{ \int_{\mathcal{F}_k} \sum_{c \in C} p(f_k, c | \xi^i) \log \frac{p(f_k, c | \xi^i)}{p(f_k | \xi^i) p(c | \xi^i)} df_k \right\}, \quad (3.12)$$

where the variable C represents the classes, $C \in \{c_1, \dots, c_m\}$, and $\xi^i = \{F_{\alpha_1} = \xi_{\alpha_1}, \dots, F_{\alpha_i} = \xi_{\alpha_i}\}$ is the usual shorthand for the set of values observed on the selected features of the sample ξ . One can also think about ξ^i as being the partial information about the testing sample ξ available after i iterations.

As the adaptive feature selection framework states, the expression (3.12) is not conventional conditional mutual information (CMI). We do not average over all possible outcomes of the features $F_{\alpha_1}, \dots, F_{\alpha_i}$, but rather condition on their certain values that we observe on the testing sample ξ . In this way, we specify a region of the input space where the testing sample lies and look for a feature F_k that is discriminative for this particular region. During the early steps of the iteration, this region is typically large and it gets iteratively refined with newly observed feature values until the testing sample is assigned to the certain class.

Using the definition of the Kullback-Leibler divergence for two distributions (2.40), the proposed selection criterion (3.12) can be rewritten as follows:

$$\alpha_{i+1} = \arg \max_k \left\{ \sum_{c \in C} p(c | \xi^i) D(p(f_k | c, \xi^i) || p(f_k | \xi^i)) \right\}. \quad (3.13)$$

This is the average distance between the pdf of the feature F_k given a certain class and its marginal pdf, where both pdfs are updated after observing the current feature subset on the sample ξ . Thus, the selection criterion favors features with distinctive posterior distributions for data drawn from the different classes, that is, features that on the $(i+1)^{th}$ step are expected to discriminate best between the classes.

As feature values are not known before these features are selected, the first feature is defined independently of the testing sample ξ and should maximize the mutual information with the classes:

$$\alpha_1 = \arg \max_k I(C; F_k), \quad F_k \in \{F_1, \dots, F_n\}. \quad (3.14)$$

The scheme becomes adaptive only after the first feature is selected and the value it takes on the testing sample is observed.

Ideally, the algorithm can be stopped when one of the classes has been unambiguously identified. In practice, this is rarely possible and other stopping criteria have to be used.

Table 3.1: A toy dataset \mathcal{X} consisting of four samples which are described by three features $\{F_1, \dots, F_3\}$ and can belong to one of four classes $\{c_1, \dots, c_4\}$.

	F_1	F_2	F_3	C
\mathbf{x}_1	0	1	1	c_1
\mathbf{x}_2	0	1	0	c_2
\mathbf{x}_3	1	0	1	c_3
\mathbf{x}_4	1	1	1	c_4

A usual choice is simply a fixed number of features that should be selected, which is decided either by design or optimized using cross-validation. Another approach is to use a threshold for minimum additional information that a selected feature should bring. Theoretically, such stopping criterion is attractive because the selection process can be stopped adaptively. But its practical use is complicated due to the fact that one needs absolute values of mutual information whose estimates are likely to be inaccurate. Here, we do not study the question of stopping criteria in details. However, while introducing the combined adaptive selection scheme in Chapter 4, we suggest that a heuristic based on the idea of a degenerated selection criterion can also be used as a stopping criterion. For a review on advanced stopping rule, an interested reader is referred to [Guyon & Elisseeff, 2003].

3.5.1.1 Adaptive vs static selection

Let us consider a toy classification problem to demonstrate advantages of the adaptive approach. Table 3.1 shows a dataset consisting of four samples $\mathcal{X} = \{(\mathbf{x}_1, c_1), \dots, (\mathbf{x}_4, c_4)\}$. The training samples lie in 3-dimensional binary feature space $\mathcal{F} = \mathcal{F}_1 \times \mathcal{F}_2 \times \mathcal{F}_3$ and can belong to one out of four possible classes $c \in \{c_1, c_2, c_3, c_4\}$.

Let us start selecting features adaptively. As was mentioned above, the first feature is selected according to $I(C; F)$ without any knowledge about a testing sample ξ :

$$\alpha_1 = \arg \max_{k=1,2,3} I(C, F_k) = \arg \max_{k=1,2,3} \left\{ \sum_{j=1}^4 \sum_{f_k=\{0,1\}} p(c_j, f_k) \log \frac{p(c_j, f_k)}{p(c_j)p(f_k)} \right\}.$$

Estimating pdfs using frequency counts from the training set, we have

$$I(C, F_1) = 1, \quad I(C, F_2) \approx 0.81, \quad I(C, F_3) \approx 0.81.$$

As F_1 has the highest mutual information with the class variable C , this feature is chosen on the first iteration, i. e. $F_{\alpha_1} = F_1$. To proceed with the adaptive scheme further, we need a value of the first selected feature on the testing sample.

Case 1. Suppose $\xi_1 = 1$. The feature value is used to update the posterior of the classes:

$$p(c_1|f_1 = 1) = 0, \quad p(c_2|f_1 = 1) = 0, \quad p(c_3|f_1 = 1) = 0.5, \quad p(c_4|f_1 = 1) = 0.5.$$

That is, after selecting and evaluating the first feature, the number of hypotheses is halved. But since there are still two candidates, the selection proceeds.

The selection criterion on the second iteration is of the following form:

$$\alpha_2 = \arg \max_{k=2,3} I(C; F_k | F_1 = \xi_1) = \sum_{j=3,4} \sum_{f_k=\{0,1\}} p(c_j, f_k | f_1 = 1) \log \frac{p(c_j, f_k | f_1 = 1)}{p(c_j | f_1 = 1) p(f_k | f_1 = 1)},$$

$$I(C; F_2 | F_1 = 1) = 1, \quad I(C; F_3 | F_1 = \xi_1) = 0,$$

then $F_{\alpha_2} = F_2$, i. e. the feature F_2 is selected. For both possible values $F_2 = 1$ and $F_2 = 0$ a testing sample would be unambiguously classified as c_4 and c_3 , respectively. That is, the uncertainty about the class once the F_2 will be known is

$$\begin{aligned} H(C | F_1 = 1, F_2) &= \sum_{j=3,4} \sum_{f_2=\{0,1\}} p(c_j, f_2 | f_1 = 1) \log p(c_j | f_2, f_1 = 1) = \\ &= \sum_{j=3,4} \sum_{f_2=\{0,1\}} \frac{p(c_j, f_1 = 1, f_2)}{p(f_1 = 1)} \log \frac{p(c_j, f_1 = 1, f_2)}{p(f_1 = 1, f_2)} = 0. \end{aligned}$$

Therefore, the selection terminates with two selected features $\{F_1, F_2\}$.

Case 2. Let us consider a scenario when the first selected feature $F_1 = 0$. Then,

$$\alpha_2 = \arg \max_{k=2,3} I(C; F_k | F_1 = \xi_1) = \sum_{j=3,4} \sum_{f_k=\{0,1\}} p(c_j, f_k | f_1 = 0) \log \frac{p(c_j, f_k | f_1 = 0)}{p(c_j | f_1 = 0) p(f_k | f_1 = 0)},$$

$$I(C; F_2 | F_1 = 1) = 0, \quad I(C; F_3 | F_1 = \xi_1) = 1.$$

Therefore, the second selected feature is F_3 . Similarly to the case 1, independent of the value of F_3 on the testing sample ξ , its class label will be unambiguous, c_1 and c_2 for $F_3 = 1$ and $F_3 = 0$, respectively. Thus, there is no need to select more features and the final feature subset is $\{F_1, F_3\}$. We can conclude that all samples from the considered dataset can be classified with two adaptively selected features, either $\{F_1, F_2\}$ or $\{F_1, F_3\}$.

If we consider a static feature selector, we still have $F_{\alpha_1} = F_1$. However, in order to select the second feature, we have to average over all possible values of F_1 while conditioning on this feature:

$$\alpha_2 = \arg \max_{k=2,3} I(C; F_k | F_1) = \arg \max_{k=2,3} \sum_{f_1=\{0,1\}} p(f_1) I(C; F_k | F_1).$$

According to this criterion, both features F_2 and F_3 are equally informative after selecting F_1 , $I(C; F_2|F_1) = I(C; F_3|F_1) = 0.5$. Suppose that $F_{\alpha_2} = F_2$ is selected. Then, as $H(C|F_1, F_2) = 0.5$, knowing values of the features F_1 and F_2 is not enough for unambiguous classification and further features are needed. Thus, the selection proceeds and the third feature should be added to the subset of the relevant features. Similarly, if on the second step F_3 is chosen, $F_{\alpha_2} = F_3$, there will be still uncertainty about the class since $H(C|F_1, F_3) = 0.5$. And again the third feature has to be selected. As a result, in both cases, the final subset of the relevant features selected according to the static approach consists of three features, $\{F_1, F_2, F_3\}$ or $\{F_1, F_3, F_2\}$.

This toy example demonstrates that compared to the static feature selection the adaptive approach is able to produce feature subsets of smaller size while ensuring the same classification accuracy.

It is necessary to note that a general version of the algorithm does not remove the classes with the low posteriors from the pool of the candidates. As there is always a possibility that the values of the selected features are noisy, it is better to update (reevaluate) the posterior of all classes once the value of the next selected feature is observed on the testing sample.

3.5.2 Estimation

3.5.2.1 Density estimation

A selection criterion of ACMIFS is in fact the mutual information of a class and a feature-candidate conditioned on previously selected features. The only difference to the classical conditional mutual information is that the ACMIFS selection criterion has to be estimated not on the full input space but on its subregion defined by the values of the already selected features. On the one hand, for ACMIFS any estimation method of mutual information can be used. On the other hand, note that the set of selected features grows iteratively. Therefore, the corresponding input subregion, for which the discriminative features are sought, and its dimensionality change from iteration to iteration as well. Obviously, in order to have a computationally feasible scheme performing feature selection during the actual classification, one needs an estimator with the minimum number of parameters learned from the data. Otherwise, these parameters have to be reestimated every time the input subregion under the consideration is refined with the feature values from a testing sample ξ . The most natural choice is a non-parametric estimator which helps us also to present a rather general selection scheme without any problem-specific assumptions.

Among such entropy and mutual information estimation techniques, there are plug-in and nonplug-in approaches based on non-parametric density estimation methods such as

histograms, kernel and k -nearest neighbor density estimators (see Section 2.5 for details). Let us analyze suitability of these techniques for estimating the selection criterion of ACMIFS.

The k -nearest density estimator has the problem that a concept of neighborhood strictly depends on the input space under consideration. That is, if two points are the nearest neighbors in i -dimensional space, there is no guarantee that in the $(i + 1)$ -dimensional space they remain neighbors. In the context of ACMIFS, it means that a testing sample might have a completely different set of neighbors on every iteration. As no points further than k^{th} neighbor influence the estimate of the class posterior, this posterior might change drastically. That is, for example, on one iteration, one can look for a feature that discriminates between the classes c_1 and c_2 and on the next iteration the pool of possible class-candidates might be $\{c_3, c_4, c_5\}$, i. e. there might be no overlap at all with the previous set of the class-candidates. As a result, features selected on the early iterations might become irrelevant in combination with later selected features. Following the idea of Kraskov [Kraskov et al., 2004], one may think about fixing a distance to ξ that captures at least k neighbors along every dimension. However, for our problem it would mean that in order to learn this distance we need to know values of all features of the testing sample before selecting them. Although it is not critical for many applications, it does not go along with our framework that assumes evaluating a feature value only if this feature is discriminative.

As was already mentioned, histograms and related techniques are not very popular for multivariate density estimation. In the continuous case, data have to be discretized and advanced methods using data-driven binning, such as equiprobable or adaptive, would need to repartition data on every iteration as relations between the data points change in the space of higher dimensionality. However, even if continuous data are discretized in uniform bins, these bins have to be rebuilt on every iteration as well. The reason is that a smoothing parameter, i. e. the width of bins, should be adjusted to the dimensionality of the input space and as it increases, a number of bins and their origins change as well.

The kernel density estimation technique with a diagonal bandwidth matrix is quite convenient for estimating the selection criterion of ACMIFS. First of all, the product structure of the multivariate kernel function simplifies the evaluation of the kernel response if different feature combinations should be considered. Second, although, similar to the uniform histograms, the width parameter of the one-dimensional kernels should be widened on every iteration, this results only in a slight change of their responses compared to the previous iterations, and no other changes are necessary. Let us look at the estimation procedure in details.

Pursuing the plug-in approach to estimating mutual information, the selection criterion (3.12) can be rewritten as

$$\alpha_{i+1} = \arg \max_k \left\{ \sum_{j=1}^m p(c_j|\xi^i) \int p(f_k|\xi^i, c_j) \log \frac{p(f_k, \xi^i|c_j)p(c_j)p(\xi^i)}{p(c_j, \xi^i)p(f_k, \xi^i)} df_k \right\}. \quad (3.15)$$

The pdfs under the logarithm, that do not depend on f_k and therefore do not contribute to $\arg \max_k$, can be dropped. Thus, we obtain

$$\begin{aligned} \alpha_{i+1} = \arg \max_k \left\{ \sum_{j=1}^m p(c_j|\xi^i) \int p(f_k|\xi^i, c_j) \log \frac{p(f_k, \xi^i|c_j)}{p(f_k, \xi^i)} df_k \right\} = \\ \arg \max_k \left\{ \sum_{j=1}^m p(c_j|\xi^i) E_{p(f_k|\xi^i, c_j)} \left[\log \frac{p(f_k, \xi^i|c_j)}{p(f_k, \xi^i)} \right] \right\}. \end{aligned} \quad (3.16)$$

Using a kernel method, we would like to estimate probability density functions of different feature combinations. However, we assume that data belonging to different classes have different structures and therefore every class-conditional pdf $p(f_1, \dots, f_n|c_j)$ will be modeled separately.

Using the class-conditional estimates, the pdf $p(f_k, \xi^i)$ in the denominator of the logarithm can be found by marginalizing out the class variable from its joint pdf with these features:

$$p(f_k, \xi^i) = \sum_{j=1}^m p(f_k, \xi^i, c_j) = \sum_{j=1}^m p(f_k, \xi^i|c_j)p(c_j). \quad (3.17)$$

The posterior of the classes after observing values of the selected features can be derived using the Bayes rule in the following way:

$$p(c_j|\xi^i) = \frac{p(c_j, \xi^i)}{p(\xi^i)} = \frac{p(\xi^i|c_j)p(c_j)}{p(\xi^i)} = \frac{p(\xi^i|c_j)p(c_j)}{\sum_{j'=1}^m p(\xi^i|c_{j'})p(c_{j'})}. \quad (3.18)$$

Plugging (3.17) and (3.18) in (3.16) and omitting $p(\xi^i)$ in the denominator of (3.18) because it does not influence the maximization over k , the selection criterion is:

$$\alpha_{i+1} = \arg \max_k \left\{ \sum_{j=1}^m p(\xi^i|c_j)p(c_j) E_{p(f_k|\xi^i, c_j)} \left[\log \frac{p(f_k, \xi^i|c_j)}{\sum_{j'=1}^m p(f_k, \xi^i|c_{j'})p(c_{j'})} \right] \right\}. \quad (3.19)$$

In addition to estimating multivariate pdfs, the expression (3.19) requires estimating the conditional expectation over multivariate pdf. We propose to solve this problem with the kernel method as well.

Since the quality of the kernel density estimation does not particularly depend on the choice of a kernel function defining its shape, for convenience we restrict ourselves to Gaussian kernels $K(w) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{w^2}{2})$. Taking $p(f_k, \xi^i | c_j)$ as an example, an estimate of the multivariate pdf using the Gaussian product kernel is the following:

$$\begin{aligned} p(f_k, \xi^i | c_j) &= \left(T_j h_k \prod_{q=1}^i h_{\alpha_q} \right)^{-1} \sum_{x_s \in \mathcal{X}_j} K\left(\frac{x_{r,k} - x_{s,k}}{h_k}\right) \prod_{q=1}^i K\left(\frac{\xi_{\alpha_q} - x_{r,\alpha_q}}{h_{\alpha_q}}\right) = \\ &= \left(\sqrt{2\pi}^{(i+1)} T_j h_k \prod_{q=1}^i h_{\alpha_q} \right)^{-1} \sum_{x_s \in \mathcal{X}_j} \exp\left(-\frac{(x_{r,k} - x_{s,k})^2}{h_k^2}\right) \prod_{q=1}^i \exp\left(-\frac{(\xi_{\alpha_q} - x_{r,\alpha_q})^2}{h_{\alpha_q}^2}\right), \end{aligned} \quad (3.20)$$

where $(h_{\alpha_1}, \dots, h_{\alpha_i}, h_k)$ is a bandwidth vector of the multivariate kernel, where each entry is a bandwidth parameter of the one-dimensional kernel for the features $F_{\alpha_1}, \dots, F_{\alpha_i}, F_k$, respectively; \mathcal{X}_j is a subset of the training samples belonging to the class c_j and $T_j = |\mathcal{X}_j|$. Further, in order to simplify the notation, $K_k(x_r, x_s)$ will denote the response of the kernel along the k^{th} dimension $K\left(\frac{x_{r,k} - x_{s,k}}{h_k}\right)$, and $K(\xi^i, x_r)$ will denote the product kernel $\prod_{q=1}^i K_{\alpha_q}(\xi, x_r)$.

In contrast to the kernel function, the bandwidth or smoothing parameters h should be chosen with care since they strongly influence the accuracy of the estimates. Effective techniques for defining a vector of the optimal smoothing parameters assume solving some optimization problem. However, as was already discussed, on every iteration we have to estimate pdfs of different dimensionality. Moreover, within one iteration, $p(f_k, \xi^i | c_j)$ should be estimated for different feature candidates F_k . It means that for every distinct combination of features in the multivariate pdf, one has to look for its own vector of appropriate kernel widths. To simplify this process, we adopt the simple normal reference rule that assumes an estimated density being normally distributed. For the multivariate kernel with the product structure, it gives a closed-form expression for the optimal width of the one-dimensional kernel along every dimension depending on the dimensionality of the estimated pdf. Then, the bandwidth for some feature F_k on the iteration i is defined as

$$h_{k,i} = \left(\frac{4}{d_i + 2}\right)^{\frac{1}{d_i+4}} \sigma_k T^{-\frac{1}{d_i+4}}, \quad (3.21)$$

where d_i is the dimension of the estimated multivariate density on the iteration i , σ_k is the standard deviation of the data points and T is the number of all training samples. Further,

while referring to the smoothing parameters, the iteration subscript i will be skipped in order to simplify the notation. However, it is still meant that the bandwidth parameters are changed on every iteration to adjust to the growing dimensionality of the input space. It is necessary to note that theoretically the bandwidth vector should be chosen for every class as we model them separately. Similar to the authors of the Parzen window feature selector, which can be seen as a static analog of ACMIFS [Kwak & Choi, 2002a], we use the same bandwidth parameters for all classes. On the one hand, this is due to the small amount of data for every class, which is not enough to infer the optimal bandwidth. On the other hand, this simplification is done for computational reasons.

Extension to the discrete case. The presented estimation scheme of ACMIFS is developed for continuous data. However, it can also be directly applied to discrete data using discrete kernels. As an extension to the multivariate continuous kernel function, a multivariate binary kernel was proposed by Aitchison and Aitken [Aitchison & Aitken, 1976]. It has also a product structure and its univariate component along the k^{th} dimension is of the following form:

$$K_k(x, y) = \lambda_k^{(1-d(x,y))} (1 - \lambda_k)^{d(x,y)}, \quad (3.22)$$

where $d(x, y)$ is the Hamming distance between two variables x and y and λ_k is a smoothing parameter associated with the feature F_k , $\lambda_k \in [\frac{1}{2}, 1]$. Setting $\lambda = \frac{1}{2}$ leads to the uniform probability mass function whereas λ close to 1 gives a simple maximum likelihood estimate.

For variables with more than two values, we suggest the following generalization:

$$K_k(x, y) = \lambda_k^{(1-d(x,y)/d_k)} (1 - \lambda_k)^{d(x,y)/d_k}, \quad (3.23)$$

where $d(x, y)$ is the Euclidean distance between the discrete variables x and y and d_k acts as a normalization constant, which is a distance between the maximum and the minimum possible value of the variable F_k .

Similar to the continuous analog, the accuracy of the discrete kernel method depends on the choice of the smoothing parameter λ , which can be found using methods developed for the standard kernel density estimation.

3.5.2.2 Conditional Expectation

We estimate the conditional expectation over the multivariate pdf $p(f_k | \xi^i, c_j)$ using a kernel-based estimator as well. Let us consider a training set $\mathcal{X} = \{(\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_T, \mathbf{y}_T)\}$, where \mathbf{x}_i and \mathbf{y}_i are realizations of n_x - and n_y -dimensional continuous random variables

\mathbf{x} and \mathbf{y} , respectively. Suppose, one needs to estimate the expectation of some function $g(\mathbf{x})$ over the conditional distribution $p(\mathbf{x}|\mathbf{y} = \mathbf{a})$, where \mathbf{a} is a particular observation of the variable \mathbf{y} . Then, using the nonparametric kernel regression estimator proposed by Nadaraya [Nadaraya, 1964] and Watson [Watson, 1964], the conditional expectation of $g(\mathbf{x})$ is:

$$\mathbb{E}_{p(\mathbf{x}|\mathbf{y}=\mathbf{a})}[g(\mathbf{x})] = \frac{(T \prod_{j=1}^{n_y} h_j)^{-1} \sum_{i=1}^T \prod_{j=1}^{n_y} K_j(a, y_i) g(\mathbf{x}_i)}{(T \prod_{j=1}^{n_y} h_j)^{-1} \sum_{i=1}^T \prod_{j=1}^{n_y} K_j(a, y_i)}, \quad (3.24)$$

where (h_1, \dots, h_{n_y}) is a bandwidth vector of the kernel for the variable \mathbf{y} and $K_j(a, y_i) = K(\frac{a_j - y_{i,j}}{h_j})$. Note that the denominator is the kernel density estimate of $p(\mathbf{y} = \mathbf{a})$.

Plugging (3.24) for $\mathbb{E}_{p(f_k|\xi^i, c_j)}[\log(\cdot)]$ into the selection criterion (3.19), we have:

$$\begin{aligned} \alpha_{i+1} = \arg \max_k & \left\{ \sum_{j=1}^m \frac{p(\xi^i | c_j) p(c_j)}{p(\xi^i | c_j)} (T_j \prod_{q=1}^i h_{\alpha_q})^{-1} \times \right. \\ & \left. \sum_{x_r \in \mathcal{X}_j} \prod_{q=1}^i K_{\alpha_q}(\xi, x_r) \log \frac{p(f_k = x_{r,k}, \xi^i | c_j)}{\sum_{j'=1}^m p(f_k, \xi^i | c_{j'}) p(c_{j'})} \right\} = \\ \arg \max_k & \left\{ \sum_{j=1}^m p(c_j) T_j^{-1} \sum_{x_r \in \mathcal{X}_j} \prod_{q=1}^i K_{\alpha_q}(\xi, x_r) \log \frac{p(f_k = x_{r,k}, \xi^i | c_j)}{\sum_{j'=1}^m p(f_k, \xi^i | c_{j'}) p(c_{j'})} \right\}, \end{aligned} \quad (3.25)$$

where \mathcal{X}_j is a set of training samples that belong to the class c_j . Note that $(\prod_{q=1}^i h_{\alpha_q})^{-1}$ can be dropped since it is just a multiplicative constant for the value of the selection criterion of all features-candidates within the iteration $(i+1)$.

Finally, using the kernel method to estimate densities, the expression (3.25) is of the form:

$$\alpha_{i+1} = \arg \max_k \left\{ \sum_{j=1}^m p(c_j) T_j^{-1} \times \right. \\ \left. \sum_{x_r \in \mathcal{X}_j} K(\xi^i, x_r) \log \frac{(T_j h_k \prod_{q=1}^i h_{\alpha_q})^{-1} \sum_{x_s \in \mathcal{X}_j} K_k(x_r, x_s) K(\xi^i, x_s)}{\sum_{j'=1}^m p(c_{j'}) (T_{j'} h_k \prod_{q=1}^i h_{\alpha_q})^{-1} \sum_{x_u \in \mathcal{X}_{j'}} K_k(x_r, x_u) K(\xi^i, x_u)} \right\} = \\ \arg \max_k \left\{ \sum_{j=1}^m p(c_j) T_j^{-1} \sum_{x_r \in \mathcal{X}_j} K(\xi^i, x_r) \log \frac{T_j^{-1} \sum_{x_s \in \mathcal{X}_j} K_k(x_r, x_s) K(\xi^i, x_s)}{\sum_{j'=1}^m p(c_{j'}) T_{j'}^{-1} \sum_{x_u \in \mathcal{X}_{j'}} K_k(x_r, x_u) K(\xi^i, x_u)} \right\}. \quad (3.26)$$

If the prior probabilities of the classes are estimated from the training set, i. e. $p(c_j) = \frac{T_j}{T}$, then the expression in the denominator under the logarithm can be estimated as an average over the kernel responses in all training points:

$$\sum_{j'=1}^m p(c_{j'}) T_{j'}^{-1} \sum_{x_u \in \mathcal{X}_{j'}} K_k(x_r, x_u) K(\xi^i, x_u) = \sum_{j'=1}^m \frac{T_j}{T} T_{j'}^{-1} \sum_{x_u \in \mathcal{X}_{j'}} K_k(x_r, x_u) K(\xi^i, x_u) = \\ T^{-1} \sum_{x_u \in \mathcal{X}} K_k(x_r, x_u) K(\xi^i, x_u).$$

3.5.2.3 Smoothing

The expression under the logarithm in (3.25) measures a ratio between values of the class-conditional and marginal joint densities of F_k and ξ^i in the training point x_r , i. e. between the feature-candidate and already selected features taking values observed on the testing sample ξ :

$$z_{k,r} = \log \frac{p(f_k = x_{r,k}, \xi^i | c_j)}{p(f_k = x_{r,k}, \xi^i)} = \log \frac{T_j^{-1} \sum_{x_s \in \mathcal{X}_j} K_k(x_r, x_s) K(\xi^i, x_s)}{T^{-1} \sum_{x_u \in \mathcal{X}} K_k(x_r, x_u) K(\xi^i, x_u)}. \quad (3.27)$$

There could be a situation when the values of these densities are close to zero. On the one hand, it could indicate that for the subregion of the input space where $f_k = x_{r,k}$ (either conditioned on c_j or not, $p(f_k = x_{r,k}, \xi^i | c_j)$ or $p(f_k = x_{r,k}, \xi^i)$), respectively a probability of

observing the testing sample ξ^i is very low. On the other hand, recall that in the situations of small training sets the estimates of the joint pdfs $p(\cdot, \xi^i)$ are unreliable in general and in particular they get worse as the dimension of ξ^i grows. Thus, the small values of $p(\cdot, \xi^i)$ could be just due to lack of the training data in this particular subregion to estimate the high-dimensional densities.

In order to reduce an influence of the possible unreliabilities on the estimation of the selection criterion, we suggest to *ignore fine differences between values of the pdfs* in the numerator and denominator of (3.27). For example, if at some iteration α_{i+1} we have the ratios like $\frac{10^{-1}}{10^{-2}}$ and $\frac{10^{-5}}{10^{-6}}$, it might be better to make their contributions different and moreover not to differentiate at all between two values, if their absolute difference is relatively small, i. e. $|10^{-1} - 10^{-2}| \gg |10^{-5} - 10^{-6}|$ and $|10^{-5} - 10^{-6}| \approx 0$.

To achieve this, we suggest to smooth both pdfs in the log-ratio (3.27).

The smoothing methods considered here are inspired by smoothing techniques used in language modeling (for a detailed review see [Chen & Goodman, 1998; Zhai, 2008]). There, smoothing is used for estimating probability mass functions of word sequences, which are called n -grams, i. e. sequences consisting of n words. For example, for a bigram $\{w_{i-1}, w_i\}$ one wants to estimate a probability $p(w_i|w_{i-1})$, i. e. a probability with which the word w_i follows the word w_{i-1} in the collection of documents. When these probabilities are estimated from sparse training sets, it may happen that some n -grams are not observed. Then, smoothing is applied to avoid zero probabilities.

An **additive smoothing** introduced by [Lidstone, 1920; Johnson, 1932; Jeffreys, 1948] is one of the simplest smoothing methods used in probability estimation. In order to avoid zero values when estimating probabilities, it assumes that every event occurs one time more than it was observed. A generalized version of this method used by [Chen & Goodman, 1998] for language modeling pretends that an event occurs δ times more than it was observed, where the smoothing parameter δ is usually set to be some small value, $\delta \in (0, 1)$.

Note that the additive smoothing adds δ to a count of the event and then applies a renormalization to obtain a proper probability density. As we work with continuous variables, we suppose that δ is added not to a count but directly to the probability value, which is then renormalized. Thus, according to this method the smoothed version of (3.27) is:

$$z_{k,r}^{sm} = \log \frac{p^{sm}(f_k = x_{r,k}, \xi^i | c_j)}{p^{sm}(f_k = x_{r,k}, \xi^i)} = \log \frac{a_{j,i+1} p(f_k = x_{r,k}, \xi^i | c_j) + \delta_{j,i+1}}{a'_{i+1} p(f_k = x_{r,k}, \xi^i) + \delta'_{i+1}}, \quad (3.28)$$

where the superscript sm stands for “smoothed”, $\delta_{j,i+1}$ and $a_{j,i+1}$ are the smoothing parameter and the normalization factor on the $(i+1)^{th}$ iteration for the density conditioned on the class c_j , respectively; δ'_{i+1} and a'_{i+1} correspond to the parameters for the unconditional density.

In order for $p(f_k = x_{r,k}, \xi^i | c_j)$ and $p(f_k = x_{r,k}, \xi^i)$ to be the proper densities, the following should hold:

$$\int_{\mathcal{F}_k} p^{sm}(f_k | \xi^i, c_j) df_k = 1, \quad \int_{\mathcal{F}_k} p^{sm}(f_k | \xi^i) df_k = 1. \quad (3.29)$$

Approximating the integrals by the sum over the samples and using the expressions of the smoothed densities from (3.28), we have:

$$\sum_{x_s \in \mathcal{X}_j} \frac{a_{j,i+1} p(f_k = x_{s,k}, \xi^i | c_j) + \delta_{j,i+1}}{p(\xi^i | c_j)} = 1, \quad \sum_{x_u \in \mathcal{X}} \frac{a'_{i+1} p(f_k = x_{u,k} | \xi^i) + \delta'_{i+1}}{p(\xi^i)} = 1,$$

which gives the following expressions for the normalization factors:

$$a_{j,i+1} = 1 - \frac{T_j \delta_{j,i+1}}{p(\xi^i | c_j)}, \quad a'_{i+1} = 1 - \frac{T \delta'_{i+1}}{p(\xi^i)} \quad (3.30)$$

with the requirement $\lim_{T \rightarrow \infty} \delta \rightarrow 0$.

However, as we would like to simplify the estimation, we assume that the smoothing parameters $\delta_{j,i+1}$ and δ'_{i+1} are small enough, so that we can set $a_{j,i+1} \approx 1$ and $a'_{i+1} \approx 1$. Therefore, formally the expressions (3.29) do not hold any more and the smoothing becomes improper. Such a simplification, which we call here an improper smoothing, should not cause problems because the smoothed densities appear only in the place where we measure a ratio between them.

Nevertheless, it is still possible to preserve the marginalization property for the smoothed class-conditional densities appearing in the log-ratio:

$$p^{sm}(f_k = x_{r,k}, \xi^i) = \sum_{j=1}^m p^{sm}(f_k = x_{r,k}, \xi^i | c_j) p(c_j). \quad (3.31)$$

This expression can be rewritten as follows

$$p(f_k = x_{r,k}, \xi^i) + \delta'_{i+1} = \sum_{j=1}^m (p(f_k = x_{r,k}, \xi^i | c_j) + \delta_{j,i+1}) p(c_j),$$

giving

$$\delta'_{i+1} = \sum_{j=1}^m \delta_{j,i+1} p(c_j). \quad (3.32)$$

Without loss of generality, we take the same smoothing parameter $\delta_{j,i+1}$ for every class c_j (further denoted as just δ_{i+1}). Then, the statement (3.32) and as a result the statement (3.31) hold if $\delta'_{i+1} = \delta_{i+1}$.

There is also a practical reason for the condition $\delta'_{i+1} = \delta_{i+1}$. In order to cancel the difference between the small values of the pdfs, one has to make sure that as $p^{sm}(f_k = x_{r,k}, \xi^i | c_j) \rightarrow 0$ and $p^{sm}(f_k = x_{r,k}, \xi^i) \rightarrow 0$, the ratio between the values of the smoothed pdfs equals to 1, i. e. $\frac{p^{sm}(f_k = x_{r,k}, \xi^i | c_j)}{p^{sm}(f_k = x_{r,k}, \xi^i)} = \frac{\delta_{i+1}}{\delta'_{i+1}} \approx 1$, so that $\log \frac{\delta_{i+1}}{\delta'_{i+1}} \approx 0$. As a result, we need a condition that both pdfs in the ratio are smoothed with the same smoothing parameter, i. e. $\delta_{i+1} = \delta'_{i+1}$.

Then, the final form of the smoothed log-ratio according to the additive method is:

$$z_{k,r}^{sm} = \log \frac{p(f_k = x_{r,k}, \xi^i | c_j) + \delta_{i+1}}{p(f_k = x_{r,k}, \xi^i) + \delta_{i+1}} \quad (3.33)$$

Now, the question is how to define the appropriate value of δ_{i+1} . As prompted by the subscript $i+1$, we suggest that it should be adapted on every iteration and give heuristic justifications for this choice below.

It was mentioned before that as we select features the dimensionality of the pdfs in (3.27) grows. Consequently, values of the joint pdfs of increasing number of variables decrease iteratively, i. e. $p(F_{\alpha_1}, \dots, F_{\alpha_{i-1}}) \geq p(F_{\alpha_1}, \dots, F_{\alpha_{i-1}}, F_{\alpha_i})$. Thus, first of all, the smoothing parameter should be adjusted to the current dimension of the pdf. Second, since both pdfs are joint pdfs of the selected features taking the values observed on the testing sample ξ , we suggest that it should be adjusted to $p(\xi^i)$, i. e. to the current probability of the testing sample. By doing this, we adapt the value of the smoothing parameter to the “level of surprise” associated with observing the values $\xi_{\alpha_1}, \dots, \xi_{\alpha_i}$ given the training set. This adjustment ensures that in situations when the testing sample ξ is very unlikely the smoothing parameter does not oversmooth the pdfs. The oversmoothing can lead to degeneration of the selection criterion, i. e. a situation when its value is the same or equals zero for all remaining features. As a result, the selection will be random. In our particular situation, this can happen if the smoothing value is fairly larger than all smoothed values of the pdfs leading to $z_{k,r}^{sm} = 0$ for all features-candidates k in all training points r .

Note that at the same time the smoothing parameter is constant within every iteration and all pdfs are smoothed uniformly independent of their values.

Thus, the smoothing parameter can be defined as follows:

$$\delta_{i+1} = \alpha p(\xi^i), \quad (3.34)$$

where α is a small adjustable constant, which controls a degree of the applied smoothing. The optimal value of α can be chosen using cross-validation method, i. e. using the error of a classifier built on features selected using different values of α .

However, in our simulations we used δ_{i+1} proportional to the maximum response of the product kernel $K(\xi^i, x_u)$ over all training points x_u , i. e. to the kernel response in the training point that is the closest to the current testing sample ξ :

$$\delta_{i+1} = \alpha \max_{x_u \in \mathcal{X}} \left\{ \left(\prod_{q=1}^i h_{\alpha_q} \right)^{-1} K(\xi^i, x_u) \right\}. \quad (3.35)$$

This means that we decide how likely the testing sample ξ is based not on the whole training set, as in (3.34), but on the closest neighbor. This approximation is similar to an idea of the k -nearest neighbor algorithm, where decisions are based on the k nearest training samples (in our case, $k = 1$).

The obvious advantage of the additive smoothing is its simplicity. Nevertheless, it is heavily criticized in the language modeling community due to its poor performance [Gale & Church, 1994]. This is caused by the fact that all n -grams, which consist of words of different frequencies, are smoothed uniformly. However, such a simple smoothing is fine for our problem, since we do not need precise values of the selection criterion, but rather want to find a feature that maximizes it.

Another smoothing method, which we consider here, assumes **interpolation with some background distribution**. The idea is similar to the Jelinek-Mercer smoothing [Jelinek & Mercer, 1980] for language modelling. There, an n -gram model is interpolated with a low-order model, i. e. an $(n-1)$ -gram. For example, for a bigram $\{w_{i-1}, w_i\}$, a smoothed version of $p(w_i|w_{i-1})$ is estimated as a mixture of $p(w_i|w_{i-1})$ and $p(w_i)$. Thus, instead of assigning a zero probability to the unobserved word sequence $\{w_{i-1}, w_i\}$, some small value proportional to the probability of the word w_i alone is used.

Formally, a smoothed pdf is a mixture of the original and background distributions:

$$p^{sm}(w_i|w_{i-1}) = (1 - \lambda)p(w_i|w_{i-1}) + \lambda p(w_i), \quad (3.36)$$

where λ is a mixing coefficient, $\lambda \in [0, 1]$.

In fact, the general idea of the interpolation with a background distribution is similar to the one of the additive smoothing. The difference is only in the definition of the smoothing value δ .

Then, using this definition of smoothing, the log-ratio under consideration is:

$$r_{k,r}^{sm} = \log \frac{(1 - \lambda_{1,i+1})p(f_k = x_{r,k}, \xi^i | c_j) + \lambda_{1,i+1}\delta_{1,i+1}}{(1 - \lambda_{2,i+1})p(f_k = x_{r,k}, \xi^i) + \lambda_{2,i+1}\delta_{2,i+1}}, \quad (3.37)$$

where $\delta_{1,i+1}$ and $\delta_{2,i+1}$ are the background distributions on the iteration $(i+1)$ for the $p(f_k = x_{r,k}, \xi^i | c_j)$ and $p(f_k = x_{r,k}, \xi^i)$, respectively.

In contrast to approaches in language modeling, which use interpolation with low-order models, as the background distributions $\delta_{1,i+1}$ and $\delta_{2,i+1}$ we suggest to use the corresponding $(i+1)$ -dimensional *prior densities* in the considered training point x_r :

$$\begin{aligned}\delta_{1,i+1} &= p(f_k = x_{r,k}, f_{\alpha_1} = x_{r,\alpha_1}, \dots, f_{\alpha_i} = x_{r,\alpha_i} | c_j), \\ \delta_{2,i+1} &= p(f_k = x_{r,k}, f_{\alpha_1} = x_{r,\alpha_1}, \dots, f_{\alpha_i} = x_{r,\alpha_i}).\end{aligned}\tag{3.38}$$

This choice of $\delta_{\cdot,i}$ is justified as follows. If x_r lies in the region with many training points, it is highly probable to observe there a testing sample ξ . Therefore, the estimates of $p(f_k = x_{r,k}, \xi^i | c_j)$ are mixed with the large prior. Accordingly, in the sparser regions of the input space where a priori any data point is not likely, the pdf estimates are smoothed less. Thus, in contrast to the additive technique where smoothing is uniform, here we have a locally adaptive smoothing which is adjusted to the prior probability of data in that subregion of the input space where the point x_r lies. This technique can be considered as a way of improving noisy adaptive pdf estimates by mixing with prior.

Plugging (3.38) in (3.37) and canceling out $(h_k \prod_{q=1}^i h_{\alpha_q})^{-1}$ in both numerator and denominator, we have:

$$z_{k,r}^{sm} = \log \frac{T_j^{-1} \left((1 - \lambda_{1,i+1}) \sum_{x_s \in \mathcal{X}_j} K_k(x_r, x_s) K(\xi^i, x_s) + \lambda_{1,i+1} \sum_{x_s \in \mathcal{X}_j} K_k(x_r, x_s) \prod_{q=1}^i K_{\alpha_q}(x_r, x_s) \right)}{T^{-1} \left((1 - \lambda_{2,i+1}) \sum_{x_u \in \mathcal{X}} K_k(x_r, x_u) K(\xi^i, x_u) + \lambda_{2,i+1} \sum_{x_u \in \mathcal{X}} K_k(x_r, x_u) \prod_{q=1}^i K_{\alpha_q}(x_r, x_u) \right)}.\tag{3.39}$$

To proceed further, let us have a look at the behavior of the unnormalized smoothing background densities in (3.39). For small training sets as the dimension of the corresponding pdfs grows we have the following:

$$\begin{aligned}\lim_{i \rightarrow n-1} \sum_{x_s \in \mathcal{X}_j} K_k(x_r, x_s) \prod_{q=1}^i K_{\alpha_q}(x_r, x_s) &= 1, \quad x_r \in \mathcal{X}_j \\ \lim_{i \rightarrow n-1} \sum_{x_u \in \mathcal{X}} K_k(x_r, x_u) \prod_{q=1}^i K_{\alpha_q}(x_r, x_u) &= 1,\end{aligned}\tag{3.40}$$

because due to data sparsity it is very likely that the multidimensional kernel responds only to the training sample in which it is centered:

$$\lim_{i \rightarrow n} \prod_{v=1}^i K_v(x_r, x_u) \rightarrow \begin{cases} 1, & \text{if } r = u \\ 0, & \text{otherwise.} \end{cases}\tag{3.41}$$

Therefore, for the cases where $p(f_k = x_{r,k}, \xi^i | c_j) \rightarrow 0$ and $p(f_k = x_{r,k}, \xi^i) \rightarrow 0$, the log-ratio (3.39) iteratively converges to the following constant:

$$\lim_{i \rightarrow n-1} z_{k,r}^{sm} = \log \frac{T_j^{-1} ((1 - \lambda_{1,i+1}) \times 0 + \lambda_{1,i+1} \times 1)}{T^{-1} ((1 - \lambda_{2,i+1}) \times 0 + \lambda_{2,i+1} \times 1)} = \log \frac{T_j^{-1} \lambda_{1,i+1}}{T^{-1} \lambda_{2,i+1}}. \quad (3.42)$$

The relation between $\lambda_{1,i+1}$ and $\lambda_{2,i+1}$ should be defined depending on the “default” value, which one wants to assign to the log-ratio between two small values. For example, setting $\lambda_{1,i+1} = \lambda_{2,i+1}$ will eventually lead to the log-ratio appearing in the analogous static selection criterion:

$$\lim_{i \rightarrow n-1} z_{k,r}^{sm} = \log \frac{p(f_k = x_{r,k}, f_{\alpha_1} = x_{r,\alpha_1}, \dots, f_{\alpha_i} = x_{r,\alpha_i} | c_j)}{p(f_k = x_{r,k}, f_{\alpha_1} = x_{r,\alpha_1}, \dots, f_{\alpha_i} = x_{r,\alpha_i})} = \log \frac{T}{T_j}.$$

On the one hand, this idea seems attractive. In points where there is no possibility to use posteriors for making decisions, one calculates discriminability of a feature based exclusively on the prior information. On the other hand, there is a danger that after few adaptive steps the whole selection scheme becomes static. It is very likely that the ratios between non-zero values are much smaller than those between the priors. Thus, already after few iterations we will favor the features that are discriminative with respect to the prior information and not with respect to the testing sample.

Therefore, the log-ratio between two small values is suggested to be around 0. I.e. we would like to have a contribution to the selection criterion only from those subregions of the input space where the observed values of the selected features ξ^i are likely. And smoothing should be used in order to correct the estimates for these subregions and cancel the contribution from the rest of the input space.

As a result, to achieve $z_{k,r}^{sm} = 1$, we have the following dependency between $\lambda_{1,i+1}$ and $\lambda_{2,i+1}$:

$$\lambda_{2,i+1} = \frac{T \lambda_{1,i+1}}{T_j}. \quad (3.43)$$

So far, the smoothing expressions $\lambda_{\cdot,i+1}$ and $\delta_{\cdot,i+1}$ were independent of the testing sample ξ . Since as the background distributions we use priors, by definition they cannot be influenced by ξ . However, we can adjust the mixing parameters $\lambda_{1,i+1}$ and $\lambda_{2,i+1}$ to the probability of observing a certain testing sample. Therefore, similar to a way of defining the smoothing parameter for the additive method, we suggest to adjust the λ 's on every iteration and in particular to make them dependent on $p(\xi^i)$. Here, as an approximation we use again the maximum response of the product kernel $K(\xi^i, x_u)$ over all training points x_u . Denoting $\lambda_{1,i+1}$ on the $(i+1)^{th}$ iteration as λ_{i+1} , we define it as follows:

$$\lambda_{i+1} = \alpha \max_{x_u \in \mathcal{X}} \left\{ \left(\prod_{q=1}^i h_{\alpha_q} \right)^{-1} K(\xi^i, x_u) \right\}, \quad (3.44)$$

where α is a small adjustable constant. This global adjustment of the smoothing parameter prevents oversmoothing and as a result degeneration of adaptivity in the selection process.

Thus, the ratio $z_{k,r}$ is smoothed according to the expression (3.39) with the adaptive $\lambda_{1,i+1}$ defined in (3.44). Recall that $\lambda_{2,i+1}$ depends on $\lambda_{1,i+1}$ and can be defined using (3.43).

Similar to the additive method, the optimal value of α can be defined using cross-validation. We followed this approach as well while performing experimental investigations of ACMIFS. At the same time, as it is done in language modeling (e. g., see [Jelinek & Mercer, 1980; Baum, 1972]), one can look for an optimal mixing parameter maximizing likelihood of some validation data using smoothed pdfs.

The efficiency of both presented smoothing techniques depends on the parameter α controlling the degree of smoothing. Despite the fact that α should be optimized for a specific problem at hand, there is a general principle that pdfs estimated from smaller training sets require larger smoothing.

The simulations presented in Section 4.2 demonstrate the influence of different degrees of smoothing on the classification performance, as well as its general utility for very small training sets.

Complexity of ACMIFS. Computational complexity of selecting i features according to ACMIFS for every testing sample is $O(iNT^2)$ which is due to the kernel method used for estimation. On the one hand, the algorithm is obviously costly. On the other hand, an adaptively selected feature subset has usually a smaller number of features. Thus, i can be small. Moreover, as was already suggested and as we show later experimentally, the adaptive approach to feature selection is especially advantageous for small training sets. In such cases, the factor T^2 should not be critical.

Chapter 4

Experimental investigations of ACMIFS

In this chapter, we present experimental investigations of ACMIFS. In particular, we analyze the ability of the proposed scheme to select informative features in high dimensional input space from a limited amount of training data. By providing small training sets for estimating the selection criterion and building a classifier, we model a situation when there is not enough data to infer its general structure. Therefore, these data can be seen as heterogeneous and according to our proposal one should profit from adaptive feature selection in terms of a number of informative features necessary for classification.

First, we show the impact of smoothing on the quality of selected features, which is measured by the accuracy of a classifier built using these features. Further, we provide a comparison of ACMIFS with the two most related information-based static and adaptive algorithms, the Parzen window feature selector [Kwak & Choi, 2002a] and the active testing model [Geman & Jedynak, 1996], respectively. Results of this comparison demonstrate general advantages of adaptive feature selection as well as the ability of ACMIFS to make use of high-order dependencies between features during the selection process. Finally, aiming to reduce computational complexity of ACMIFS, a hybrid adaptive feature selection scheme is proposed. It suggests that ACMIFS can switch to ATM after some iterations. That is, at some point ACMIFS can adopt an assumption that features are class-conditionally independent to simplify the estimation problem.

Most of the investigative experiments are run on the artificially constructed data set of pixel-based images of digits. However, the comparison with the competitive feature selection schemes is also performed on MNIST, the real-world benchmark data set of hand-written digits.

4.1 Data sets

4.1.1 Artificial data set

For investigating properties of the proposed adaptive feature selection scheme, we artificially constructed a data set for image classification. There are 11×11 pixel-based black-and-white images of digits belonging to 10 different classes. First, we construct four distinct examples of every class (see Appendix A.2). From this data set we generate a new one with 1000 samples by randomly adding 5 pixels of noise to the original images (Fig. 4.1). Further, we form 20 training sets with 30, 90, 300 and 800 samples in each. Every training set has its associated non-overlapping validation set containing 20 samples. Thus, one can think about the original training sets of the size $T = 30 + 20$, $T = 90 + 20$, $T = 300 + 20$ and $T = 800 + 20$, which are then divided to the non-overlapping subsets for learning the model and for validating it. This technique is called the holdout cross-validation method [Kohavi, 1995] and is usually used to avoid overfitting. Finally, there is one testing set containing 100 samples, which is formed by randomly selecting an equal number of samples from each class.

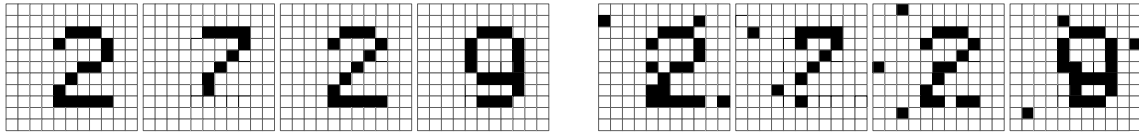


Figure 4.1: Examples of original and noisy digits.

Further, we developed an artificial neural network consisting of three consequent layers: a layer of simple feature neurons, a layer of complex feature neurons and a layer of category neurons (see Fig. 4.2). Thus, we assume that each image is described by a vector of the complex features, which in turn are functions of simple features of the image. Our simple features are inspired by the complex cells in the primary visual cortex discovered by D. Hubel and T. Wiesel in the 1960s [Hubel & Wiesel, 2005]. Both are responsive to primitive stimuli that are independent of their spatial location.

Here, each simple feature corresponds to a 3×3 image patch and the feature neuron is activated proportional to the frequency with which the corresponding patch occurs in the image. For normalization and smoothing purposes, patch frequencies are squashed in the interval $[-1, 1]$ via a sigmoid function. Thus, the activation function of the simple feature neuron is:

$$s_i = \frac{a_1}{1 + e^{-a_2(v_i - a_3)}} - a_4, \quad (4.1)$$

where v_i is a frequency of the patch corresponding to a feature F_i and $\{a_1, a_2, a_3, a_4\}$ are parameters of the sigmoid. The parameters are set in the following way: $a_1 = 2$

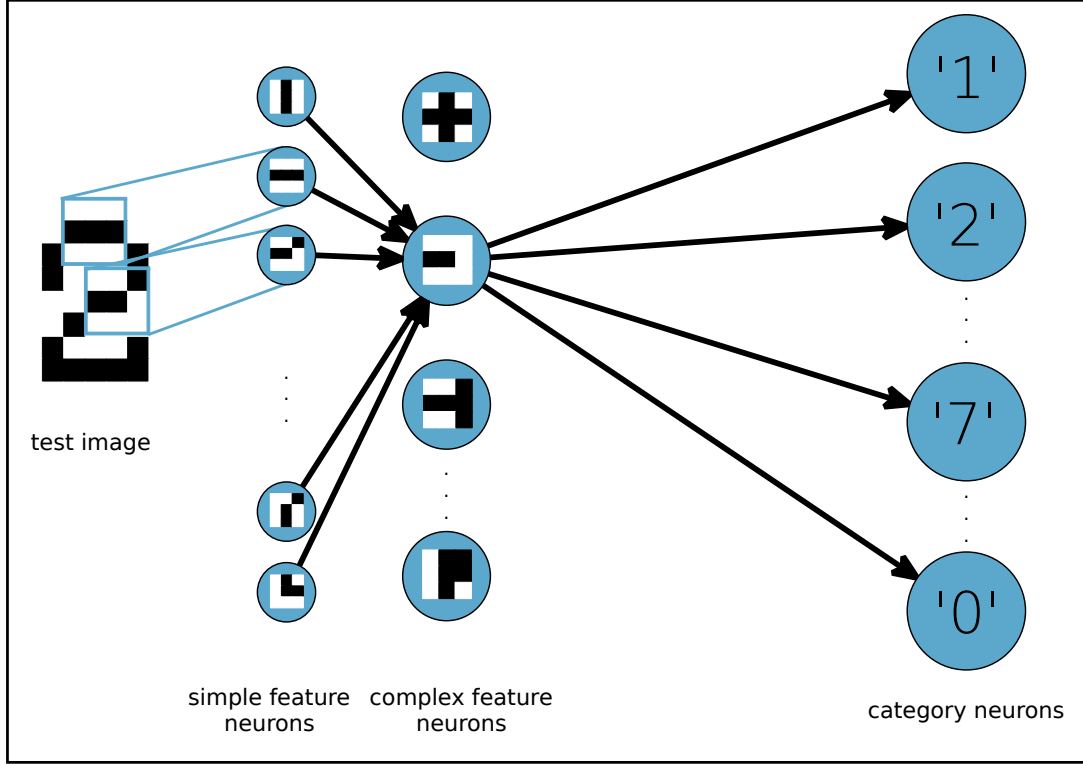


Figure 4.2: A neural network representing two layers of simple and complex feature neurons and an output layer of category neurons.

and $a_4 = 1$ to assure that $s_i \in [-1, 1]$, a_2 and a_3 control the form and the position of the function and should be chosen to achieve the desired degree of sensitivity to the feature frequency, which can vary depending on the expected noise level. In our experiments, we used $a_2 = 2$ and $a_3 = 0.3$. The behavior of the activation function depending on the patch frequency is shown on the Fig. 4.3. One can see that with such parameter setting the output of the feature neuron saturates after observing the corresponding patch more than 2 times. That is, we believe that the feature is present on the image if its frequency is larger than 2.

The complex features correspond to 3×3 image patches as well. Their activation o is computed as a weighted sum of the activations of the simple features:

$$o_j = \alpha \sum_{i=1}^{n_s} w_{ij} s_i, \quad (4.2)$$

where n_s is a number of the simple features, $\alpha \in (0, 1]$ is a normalization constant to ensure that the output of the complex feature neuron is in the range $[-1, 1]$ and w_{ij} is the

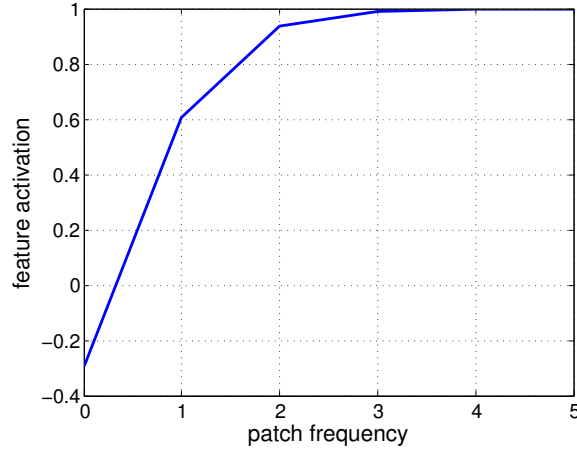


Figure 4.3: Activation function of the simple feature depending on its frequency

weight of the connection between the i^{th} simple feature neuron and j^{th} complex feature neuron. If the simple and complex feature neurons respond to the same patch, the weight w_{ij} between them is 1. For the others, it drops in the number of pixels that differ between the corresponding image patches according to the Gaussian. For example, consider two features from the Fig.4.4: the simple feature F_{s1} and the complex feature F_{c2} . Their preferred patches could be represented as strings containing 0s and 1s for white and black pixels, respectively, i. e. $p_{s1} = \{ '010010010' \}$ and $p_{c2} = \{ '001010010' \}$. The difference in the number of pixels is calculated using Hamming distance, that is $h(p_{s1}, p_{c2}) = 2$.



Figure 4.4: Example of two image patches and their representations as strings.

Then $w_{ij} = e^{-\frac{h(p_{s1}, p_{c2})^2}{\sigma^2}}$. As a result, the complex features are activated not only in response to their preferred patches but also to similar ones, which makes them more robust against pixel noise compared to the simple features.

To conclude, a simple feature is activated only if its preferred patch is present somewhere on an image. In turn, a complex feature respond to possible noisy representations of its preferred patch, which is done by aggregating responses of the simple features that are tuned to similar patches. Thus, making a parallel to complex cells in the visual cortex, we have two layers of cells with receptive fields of increasing complexity, which in our case is measured by resistance to noise.

Since there are 9 binary pixels in each 3×3 patch, both simple and complex feature layers have $2^9 = 512$ neurons. Feature selection is done on the complex feature layer, thus every image is described by a vector of 512 complex feature values.

As a classifier, we used the weighted k-nearest neighbor algorithm (wk-NN). It assigns a class to a testing sample based on a distance-weighted vote of the k -nearest training samples. The wk-NN is one of the simplest classifiers, but the fact that it does not need learning is useful because the adaptive scheme requires multiple running of the classifier with different features. Here, we used $k=20$ hand-tuned using validation sets.

Since we are interested not in the absolute classification accuracy but in its relative difference between classifiers built on different feature subsets, the choice of the particular classifier is not critical. However, on every iteration, the updated class posterior is also estimated by KDE since it is used in the selection criterion. Therefore, one could use a classifier based on this posterior estimate, which is known as the Parzen window classifier [Parzen, 1962]. Nevertheless, in all simulations only wk-NN is used as a classifier to emphasize the fact that ACMIFS is a feature selector of the filter type and selected features can be further fed in any classifier.

4.1.2 MNIST data set

In addition to the artificially constructed data set, experiments where ACMIFS is compared to related static and adaptive feature selectors are performed also on a real-world data set of handwritten digits MNIST [LeCun & Cortes, nd]. MNIST contains images that are 28×28 pixel, black and white, size-normalized and centered. The original training and testing sets consist of 60,000 and 10,000 samples, respectively.

The features are learned by LeNetConvPool [Bergstra et al., nd], an implementation of the convolutional neural network based on the LeNet5 architecture, which was originally proposed by LeCun [LeCun et al., 1998]. The convolutional networks are biologically inspired multilayered neural networks. In order to achieve some degree of location, scale and distortion invariance, they imitate arrangement and properties of simple and complex cells in primary visual cortex by implementing local filters of increasing size, shared weights and spatial subsampling.

LeNetConvPool consists of 6 layers: 4 successive convolutional and down-sampling layers (C- and S-layers), a hidden fully-connected layer and a logistic regression as a classifier, see Figure (4.6). C-layers consist of several feature maps with overlapping 5×5 linear filters. So every filter receives an input from the 5×5 region of the previous layer, computes its weighted sum and passes it through a sigmoid function. The S-layers perform max-pooling with 2×2 non-overlapping filters. That is, an output of such filter is

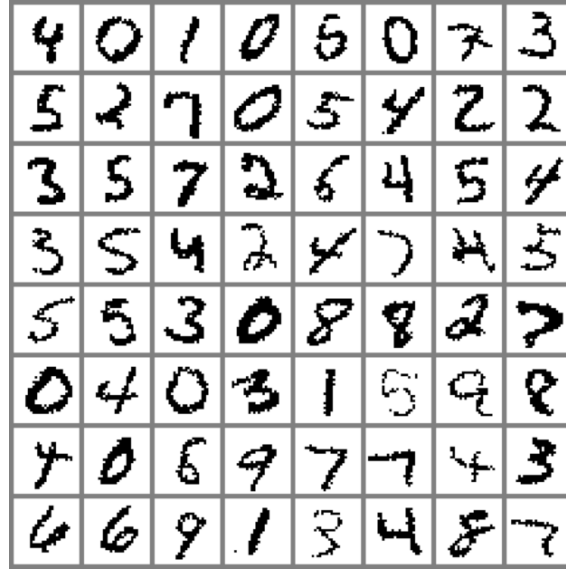


Figure 4.5: An example of the images from the MNIST dataset.

the maximum activation of units from 2×2 region of the corresponding feature map in the previous C-layer. For both types of the layers, all filters share the same weight parameters within one feature map. The first C- and S-layers have 20 feature maps, the next ones have 50. The succeeding hidden layer, which is fully-connected to all units of all feature maps in the previous S-layer, has 500 units with a sigmoid activation function. The last classification layer consists of 10 units, according to the number of classes, and performs a logistic regression. The weight parameters of all layers are learned using the gradient descent [LeCun et al., 1998]. For all implementation details see [Bergstra et al., nd].

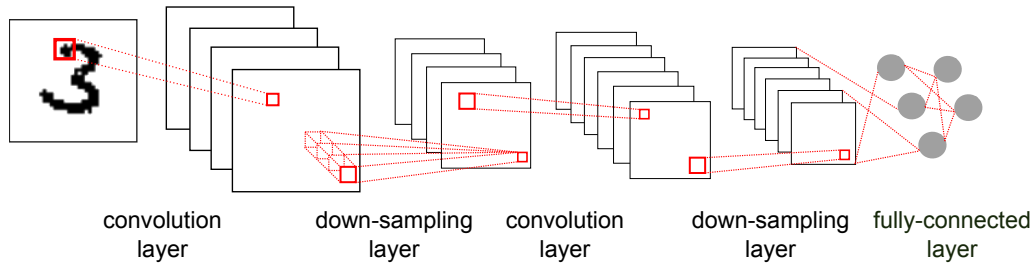


Figure 4.6: LeNetConvPool network [Bergstra et al., nd].

LeNetConvPool was trained on 15 training sets with 5,000 samples each. After that, the last classification layer was removed and the resulting 15 networks with 500 output units were used as feature extractors. These units are initial features for feature selection. Then, from every training set we formed 2 sets of different size, with $T = 100$ and $T = 300$ samples, which were used for feature selection and for classification. We use

a different amount of training data for feature extraction and for further feature selection and classification to model a situation, when one has good features but there is not enough training data to build an efficient classifier. As a classifier, we used an unweighted k -NN with $k = 5$ (again, hand-tuned on validation sets), which in contrast to wk -NN uses a simple majority vote. For computational reasons, the testing set was reduced to 500 samples, which were randomly selected from the original MNIST testing set, with an equal number of samples per class.

4.2 Smoothing

In Subsection 3.5.2.3, we presented two techniques for smoothing densities in the ratio under the logarithm (3.27):

$$z_{k,r} = \log \frac{p(f_k = x_{r,k}, \xi^i | c_j)}{p(f_k = x_{r,k}, \xi^i)}.$$

Smoothing was proposed to account for unreliable pdf estimates in higher dimensions. Using the artificial data set, we experimentally investigate the influence of these smoothing methods on the quality of selected features measured in terms of their discriminability. In order to evaluate discriminability of the selected features, we run a classifier using these features and measure its misclassification error.

4.2.1 Additive smoothing

Let us consider the first technique, namely additive smoothing (3.33) with an adaptive smoothing parameter δ_{i+1} :

$$z_{k,r}^{sm} = \log \frac{p(f_k = x_{r,k}, \xi^i | c_j) + \delta_{i+1}}{p(f_k = x_{r,k}, \xi^i) + \delta_{i+1}}$$

Recall that δ_{i+1} depends on $p(\xi^i)$, i. e. the probability of the testing sample ξ after observing values of i selected features. In particular, δ_{i+1} is proportional to the response of the product kernel $K(\xi^i, x_u)$ in the closest training point x_u with the factor of proportionality α , as defined by the expression (3.35):

$$\delta_{i+1} = \alpha \max_{x_u \in \mathcal{X}} \left\{ \left(\prod_{q=1}^i h_{\alpha_q} \right)^{-1} K(\xi^i, x_u) \right\}.$$

The value of α controls a degree with which these densities are smoothed. As was already mentioned, the optimal value of α should be adjusted to the problem at hand as well as to the amount of the available training data.

In order to inspect the influence of different degrees of smoothing on the classification performance and to choose the optimal α , we run our ACMIFS with $\alpha = 0$, $\alpha = 0.0001$, $\alpha = 0.001$, $\alpha = 0.05$. Feature selection were run for all four different sizes of the training sets, i. e. $T = 30$, $T = 90$, $T = 300$ and $T = 800$ to see the dependence between the number of training samples and the necessary degree of smoothing. The classification accuracy was measured on the validation sets.

Figure 4.7 shows the misclassification error against the number of features that were selected using different degrees of the additive smoothing controlled by α . First of all, for the training sets with a small number of samples (see subplots A and B for $T = 30$ and $T = 90$, respectively) we observe a huge decrease in the classification error when the pdf estimates are smoothed. Thus, any reasonable smoothing with α different from zero improves the estimate of the selection criterion and as a result helps to select better features.

For the middle-sized training sets (see subplots C and D for $T = 300$ and $T = 800$, respectively), it is important not to oversmooth. One can see that for $\alpha = 0.05$ the classification error is higher compared to the unsmoothed case ($\alpha = 0$).

For the training set with 300 samples, we still observe a significant improvement when pdfs are smoothed, however, now with the smaller value of α . Thus, the smoothing is still useful, though the difference between the cases where $\alpha = 0$ and $\alpha = 0.0001$ is not as prominent as for the small training sets. Finally, when there is enough training data for estimating pdfs, as in the case with 800 training samples, smoothing is not beneficial and can even impair the estimates by introducing noise.

Next, we would like to investigate on which phase of the selection process smoothing becomes crucial. I.e. whether it is at the beginning when the number of the selected features is still small or on the later iterations when one has to deal with high-dimensional pdfs.

For this, we constructed the following experiment. Using the unsmoothed version of ACMIFS, with $\alpha = 0$, we preselected subsets consisting of $N_0 = 10$, $N_0 = 20$ and $N_0 = 50$ features. Then, we let the smoothed ACMIFS, with $\alpha \neq 0$, select features further. Figure 4.8 shows an error rate against a number of the features selected by different setups for the training sets, $(T = 30; \alpha = 0.001)$ and $(T = 90; \alpha = 0.0001)$. These values of the smoothing parameter were chosen as those that showed the best results for the corresponding data sets (see Figure 4.7).

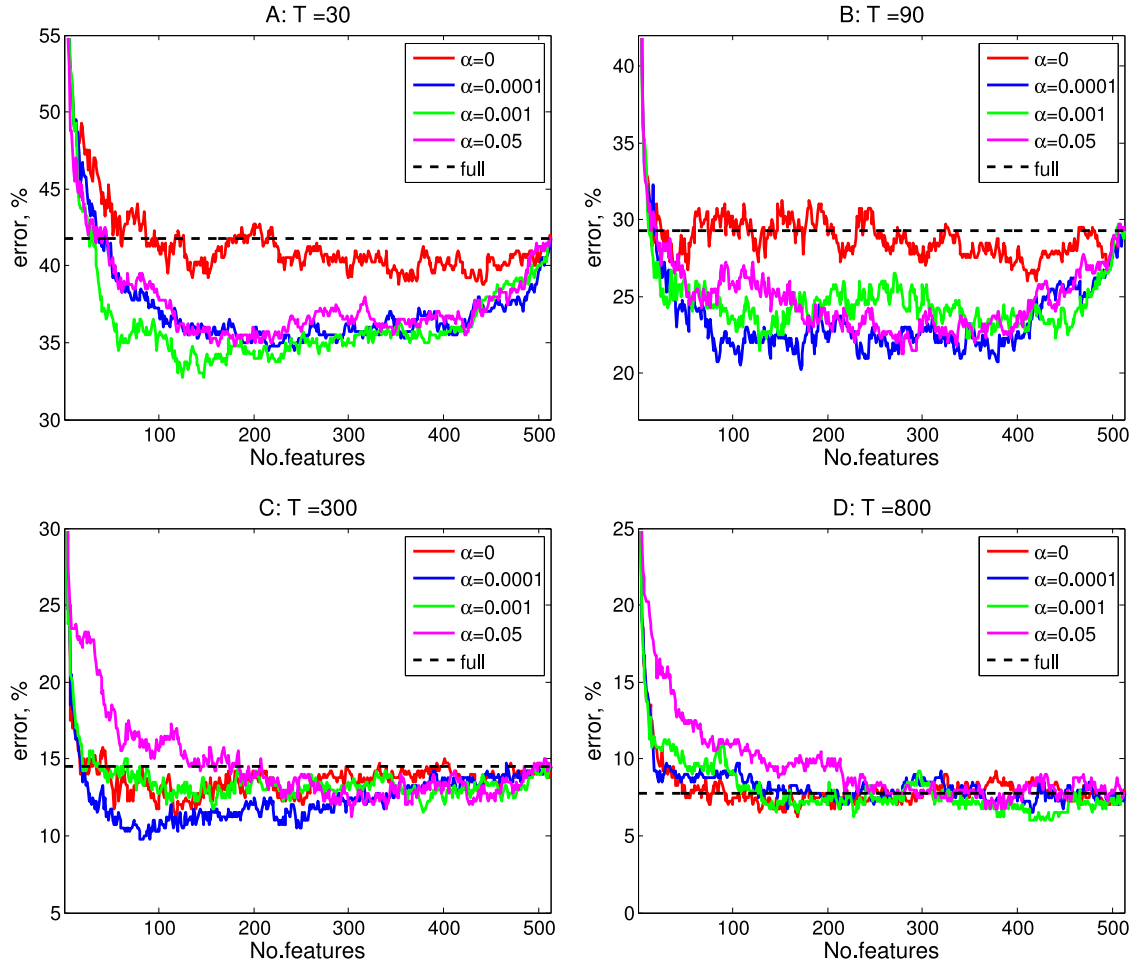


Figure 4.7: Classification error against a number of features that are selected using different degree of the additive smoothing controlled by α .

We have to note that there is no claim about the importance of the smoothing after a certain number of iterations. Our aim is rather to show the influence of smoothing after the first few iterations for training sets of different size.

One can observe the general tendency for the small datasets: smoothing starts improving the estimates already after several iterations. For both $T = 30$ and $T = 90$, there is no difference in classification performance of the smoothed and the unsmoothed ACMIFS up to first 10 – 15 iterations. It is not surprising that ACMIFS smoothed after the 10th iteration gives almost as good features as the algorithm where smoothing was applied from the beginning (see red and green curves on the Figure 4.8). However, if smoothing is introduced after 20 iterations, there is an evident difference in the accuracy. This fact suggests that those few features, which are selected approximately after 10 – 15 iterations

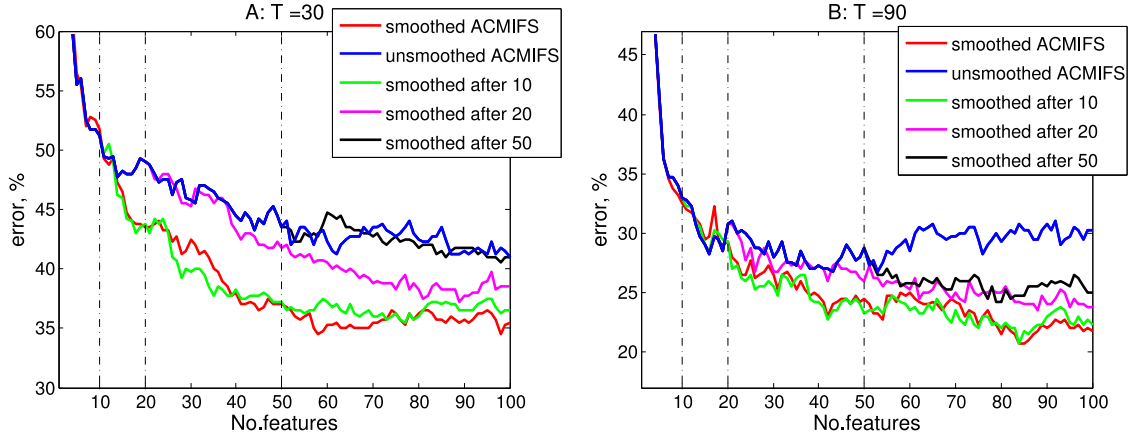


Figure 4.8: Influence of smoothing ACMIFS introduced after $N_0 = 10/20/50$ iterations. The initial feature subsets were preselected by the unsmoothed ACMIFS, further selection is performed by the smoothed version of the algorithm. The red curves on both subplots present a setup when ACMIFS is smoothed from the first iteration.

with smoothing, are able to eliminate some wrong hypotheses, i. e. set the posterior of some “wrong” classes close to 0. In this way, further selection can be simplified as one tries to discriminate between a smaller number of the classes.

It is worth noting that ACMIFS smoothed after 20 and 50 iterations for $T = 90$ can still improve the classification accuracy compared to the unsmoothed version (see magenta and blue error curves). However, if the dataset is very sparse, the first features are more crucial and ACMIFS obviously fails to recover if it gets smoothed only after 50 iterations.

This analysis gives us a hint about the dimension of pdfs when the proposed way of additive smoothing starts improving the estimates depending on the size of a training set. For example, if one decides to select few features, it is not so crucial to smooth the estimates. And correspondingly, if the final feature subset should be large, then smoothing can help to select better features on the later iterations.

4.2.2 Interpolation with a prior distribution

Now, we turn to the second smoothing technique, interpolation with a background distribution. As the background, we use a prior distribution of the smoothed density, see (3.39). As defined by the expression (3.44), the mixing parameter λ_{i+1} is proportional to the maximum response of the product kernel $K(\xi^i, x_u)$ over all training points with a factor of proportionality α .

Similar to the additive method, efficiency of smoothing with a prior is assessed by the classification performance achieved with features that are selected using different degrees of smoothing controlled by the parameter α . Figure 4.9 plots the error rate against a number of features selected with $\alpha = 0$, i. e. no smoothing, $\alpha = 0.0001$, $\alpha = 0.001$ and $\alpha = 0.05$.

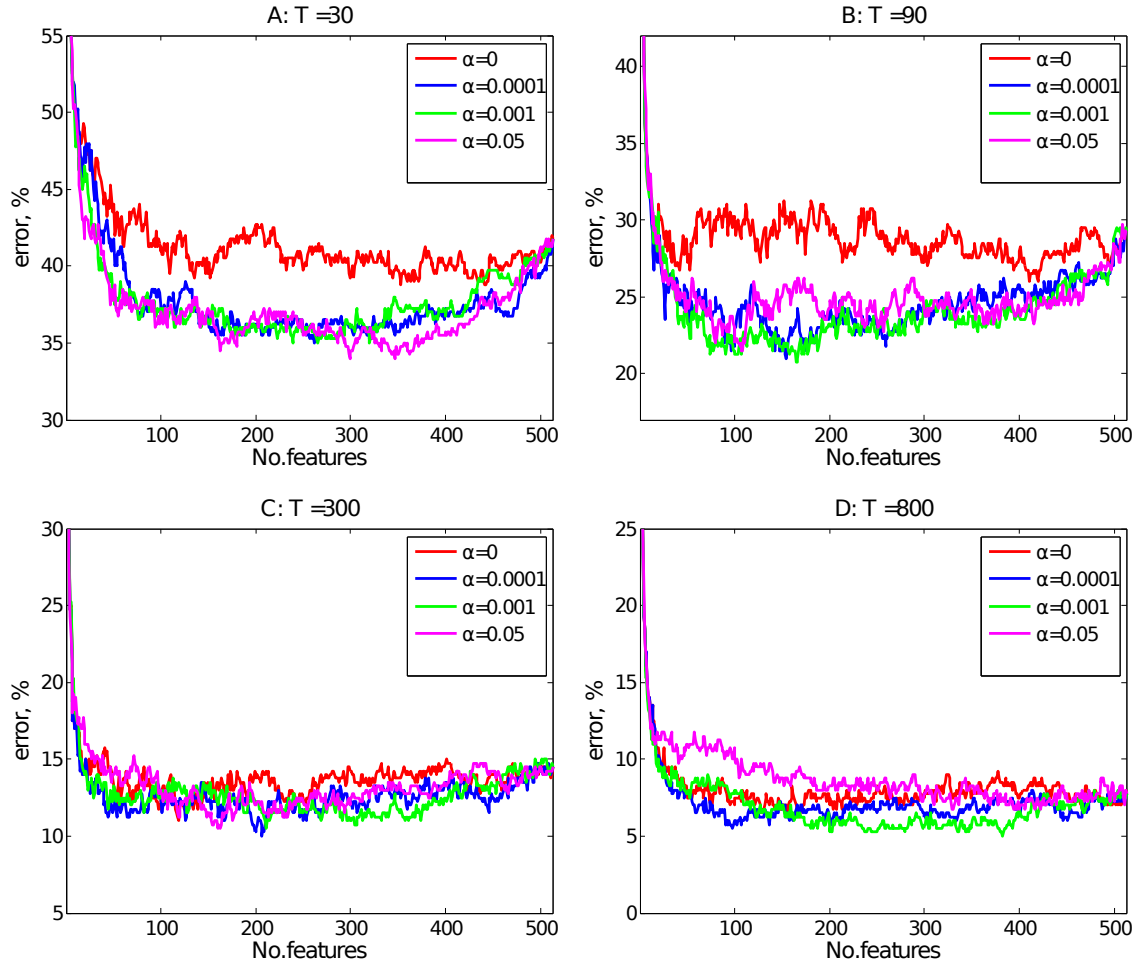


Figure 4.9: Classification error against a number of features that are selected using different degrees of smoothing with a prior controlled by α .

In general, the behavior of the error curves for different degrees of smoothing resembles the situation observed when using the additive method. Thus, one can select better features if pdfs estimated from small training sets are smoothed. And the pdfs estimated from the middle-sized datasets should be smoothed less or not at all.

Taking into account that there is no substantial difference between the results of the presented smoothing methods, we suggest that one can profit from smoothing the pdf es-

timates in general. At the same time, in order to prevent oversmoothing, the smoothing values should be adjusted to the dimensionality of the smoothed pdfs and adapted to probability of observing a testing sample. However, the exact way of smoothing does not seem to be essential.

Recall that the additive smoothing is improper because smoothed densities are not renormalized. At the same time, densities smoothed with a prior, i. e. according to the second method, are proper densities. Since we do not observe degradation of the classification performance with the features selected using the improper smoothing, this simplification is indeed not crucial for estimating the selection criterion of ACMIFS.

Having both smoothing methods almost equivalent, in further simulations we will use the additive smoothing due to its simplicity.

4.3 Comparison with PWFS and ATM

Here, we experimentally compare our method with two feature selection algorithms based on CMI: Parzen window feature selector (PWFS) [Kwak & Choi, 2002a] and active testing model (ATM) [Geman & Jedynak, 1996]. To make a fair comparison, all criteria are estimated using kernel density estimation with a bandwidth vector chosen by the normal reference rule (3.21).

In our terminology **Parzen window feature selector** is a static selection scheme (2.1). It is based on the conventional CMI, which in the original paper was also estimated with the kernel method:

$$\begin{aligned}
 \alpha_{i+1} &= \arg \max_k \{I(C, F_k | \mathbf{F}^i)\} = \\
 \arg \max_k &\left\{ \sum_{j=1}^m T_j^{-1} \sum_{x_r \in \mathcal{X}_j} \log \frac{p(f_k = x_{r,k}, c_j | \mathbf{f}^i = \mathbf{x}_r^i)}{p(f_k = x_{r,k} | \mathbf{f}^i = \mathbf{x}_r^i) p(c_j | \mathbf{f}^i = \mathbf{x}_r^i)} \right\} = \\
 \arg \max_k &\left\{ \sum_{j=1}^m T_j^{-1} \sum_{x_r \in \mathcal{X}_j} \log \frac{p(f_k = x_{r,k}, \mathbf{f}^i = \mathbf{x}_r^i | c_j)}{\sum_{j'=1}^m p(f_k = x_{r,k}, \mathbf{f}^i = \mathbf{x}_r^i | c_{j'}) p(c_{j'})} \right\}, \tag{4.3}
 \end{aligned}$$

where $(\mathbf{f}^i = \mathbf{x}_r^i)$ stands for $\{f_{\alpha_1} = x_{r,\alpha_1}, \dots, f_{\alpha_i} = x_{r,\alpha_i}\}$. Using KDE and estimating prior class probabilities from a training set, PWFS is estimated according to the following expression:

$$\alpha_{i+1} = \arg \max_k \left\{ \sum_{j=1}^m T_j^{-1} \sum_{x_r \in \mathcal{X}_j} \log \frac{T_j^{-1} \sum_{x_s \in \mathcal{X}_j} K_k(x_r, x_s) \prod_{q=1}^i K_{\alpha_q}(x_r, x_s)}{T^{-1} \sum_{x_u \in \mathcal{X}} K_k(x_r, x_u) \prod_{q=1}^i K_{\alpha_q}(x_r, x_u)} \right\}. \quad (4.4)$$

Active testing model is a feature selector based on the adaptive CMI which uses a simplifying assumption that features are conditionally independent given a class (see Subsection 3.4.2 for details). Since estimation of the selection criterion, proposed by Geman and Jedyndak, was problem-specific, here we use just the general idea of their method. Although the assumption that features are class-conditionally independent does not hold, similar to methods described in Subsection 2.6, we adopt this simplifying assumption as an approximation of the multivariate conditional mutual information.

That is, in our experiments ATM selects features according to the following criterion:

$$\alpha_{i+1} = \arg \max_k \{I(C, F_k | \xi^i)\} = \arg \max_k \left\{ \sum_{j=1}^m p(c_j) \prod_{q=1}^i p(f_{\alpha_q} = \xi_{\alpha_q} | c_j) T_j^{-1} \times \sum_{x_r \in \mathcal{X}_j} \log \frac{p(f_k = x_{r,k} | c_j) \prod_{q=1}^i p(f_{\alpha_q} = \xi_{\alpha_q} | c_j)}{\sum_{j'=1}^m p(c_{j'}) p(f_k = x_{r,k} | c_{j'}) \prod_{q=1}^i p(f_{\alpha_q} = \xi_{\alpha_q} | c_{j'})} \right\}. \quad (4.5)$$

If we cancel out $\prod_{q=1}^i p(f_{\alpha_q} = \xi_{\alpha_q} | c_j)$ in the numerator under the logarithm as it does not influence $\arg \max_k$ and estimate $p(c_j)$ as $\frac{T_j}{T}$, the estimate of the ATM selection criterion using KDE is:

$$\alpha_{i+1} = \arg \max_k \left\{ T^{-1} \prod_{q=1}^i \left[T_j^{-1} \sum_{x_s \in \mathcal{X}_j} K_{\alpha_q}(\xi, x_s) \right] \times \sum_{x_r \in \mathcal{X}_j} \log \frac{T_j^{-1} \sum_{x_s \in \mathcal{X}_j} K_k(x_r, x_s)}{\sum_{j'=1}^m T^{-1} \left[\sum_{x_s \in \mathcal{X}_{j'}} K_k(x_r, x_s) \right] \prod_{q=1}^i \left[T_{j'}^{-1} \sum_{x_s \in \mathcal{X}_{j'}} K_{\alpha_q}(\xi, x_s) \right]} \right\}. \quad (4.6)$$

4.3.1 General comparison on the artificial data set

To investigate the usefulness of the proposed ACMIFS we ran experiments on training sets with $T=30$ and 300 samples. Note that all sets have fewer training samples than features, which easily leads to overfitting. The classification errors were evaluated on separate testing samples and compared with the cases when feature selection was done using PWFS, ATM and when the classifier was run on the full feature vector, i. e. without feature selection (Fig. 4.10). All results are averaged over 20 runs with the different training sets.

One clearly sees the advantage of using an adaptive scheme for feature selection. Not only does the error rate drop very quickly with an increasing number of features, it goes even below the error that the classifier achieves when using all available features. In all our simulations, this effect never occurred for the static scheme PWFS and was particularly pronounced when using an extremely small number of training samples ($T=30$), i. e. when the classifier is prone to overfitting. Furthermore, our algorithm outperforms the ATM scheme, which assumes conditional independence of the features. The authors of ATM argue that adaptive algorithms not assuming the class-conditional independence of features require large datasets in order to provide reliable estimates of multivariate pdfs. While this is true in general, our results show that especially at the beginning, i. e. when selecting the first few features, it is beneficial to take into account high-order dependencies between features.

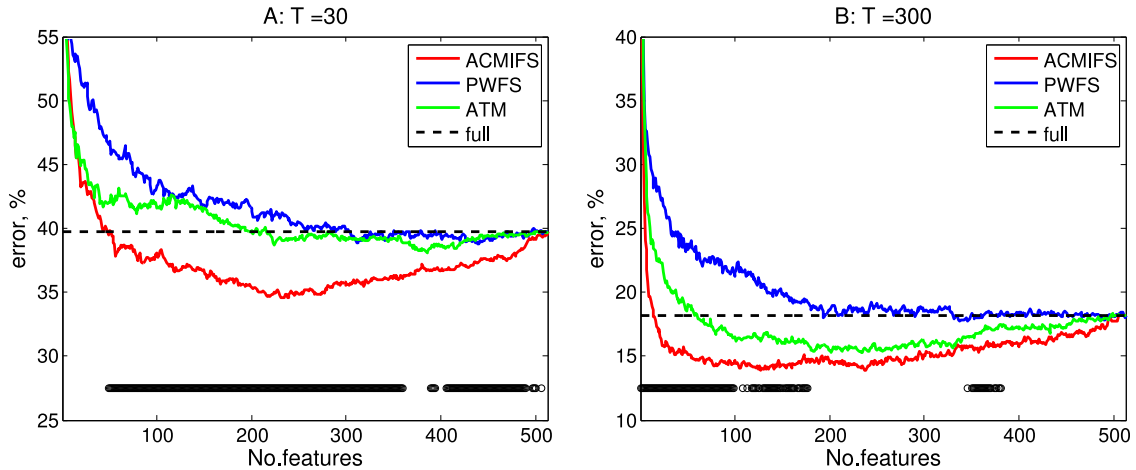


Figure 4.10: Error against the number of features for digits classification, the black markers indicate regions where AMIFS is significantly better than ATM according to the **Wilcoxon** signed-rank test at the p -level= 0.05.

4.3.2 General comparison on MNIST

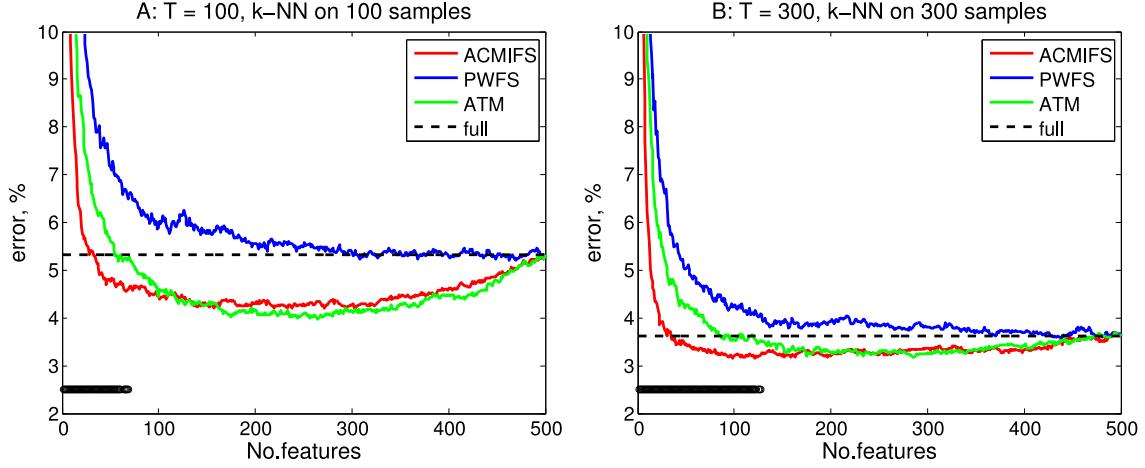


Figure 4.11: Error against the number of features for MNIST classification. k-NN was run on the same set as feature selection, the black markers indicate regions where AMIFS is significantly better than ATM according to the Wilcoxon signed-rank test at the p -level= 0.05.

In the first part of the comparative experiments, ACMIFS, PWFS and ATM were run on a training set with $T = 100$ and $T = 300$ samples. Overall, all algorithms show a similar behavior as on the artificial data set (see Fig. 4.11). The smaller differences can be attributed to the better available features, as reflected in the much lower error rates, which have been tuned by the LeNetConvPool. Again, ACMIFS outperforms ATM on the first selected features and both adaptive schemes provide some robustness against overfitting.

Further, to see whether feature selection is as beneficial when the classifier is well-trained, we repeated the experiments with a training set of 5,000 samples. However, as in the previous experiment, the feature selection was done on the small sets of 100 and 300 samples for computational reasons.

Fig. 4.12 shows that for this particular example one needs approximately 200 features to achieve the minimum error. However, there is no advantage of using any sophisticated feature selection algorithm, and one can see that the size of the training set used for selecting features does not have much influence as well. Moreover, even the random selection works about as well as other methods. We do not want to generalize results of this test by saying that for large data sets one can always select features randomly. We rather emphasize that for small data sets one can achieve better performance with features selected adaptively with our ACMIFS.

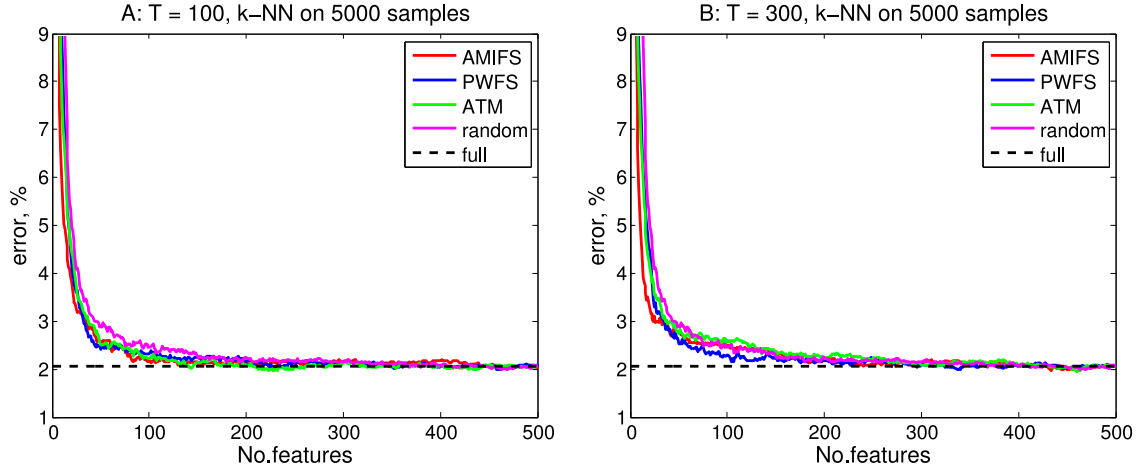


Figure 4.12: Error against the number of features for MNIST classification. k-NN was run on 5,000 training samples, feature selection on A: 100 and B: 300 training samples.

4.3.3 Behavior in higher dimensions

Further, we test the ability of the considered schemes to select informative features in high dimensions. For this, simulations are run on artificial data for training sets with $T = 30$ and $T = 300$ samples. We start with initial feature subsets containing 50 and 100 features, which are preselected by PWFS, and then select further 100 features according to the different algorithms. The initial features are selected by the static scheme in order to exclude any influence of adaptivity on the early iterations. Instead of PWFS, the initial subsets could be as well selected randomly and then fed to the tested selection schemes.

It is necessary to note that when the adaptive schemes, i. e. ACMIFS and ATM, receive preselected features, values of these features are known and used for further selection.

The results presented on Figure 4.13 show that both adaptive schemes find additional features that are markedly better than the statically selected ones. However, the difference between the static and both adaptive algorithms is less prominent when adaptivity in feature selection is introduced on later iterations as in case with $N_0 = 100$. This can be explained by the fact that ACMIFS and ATM search discriminative features for the class posterior which is updated using a larger number of suboptimal features¹ compared to the case with $N_0 = 50$.

For $T = 300$, one can see that at some point ATM, the adaptive scheme assuming conditional independence of the features given a class, starts outperforming ACMIFS. As

¹We refer to features which are selected statically as suboptimal ones because both adaptive schemes were shown to select better features (see results in Subsection 4.3).

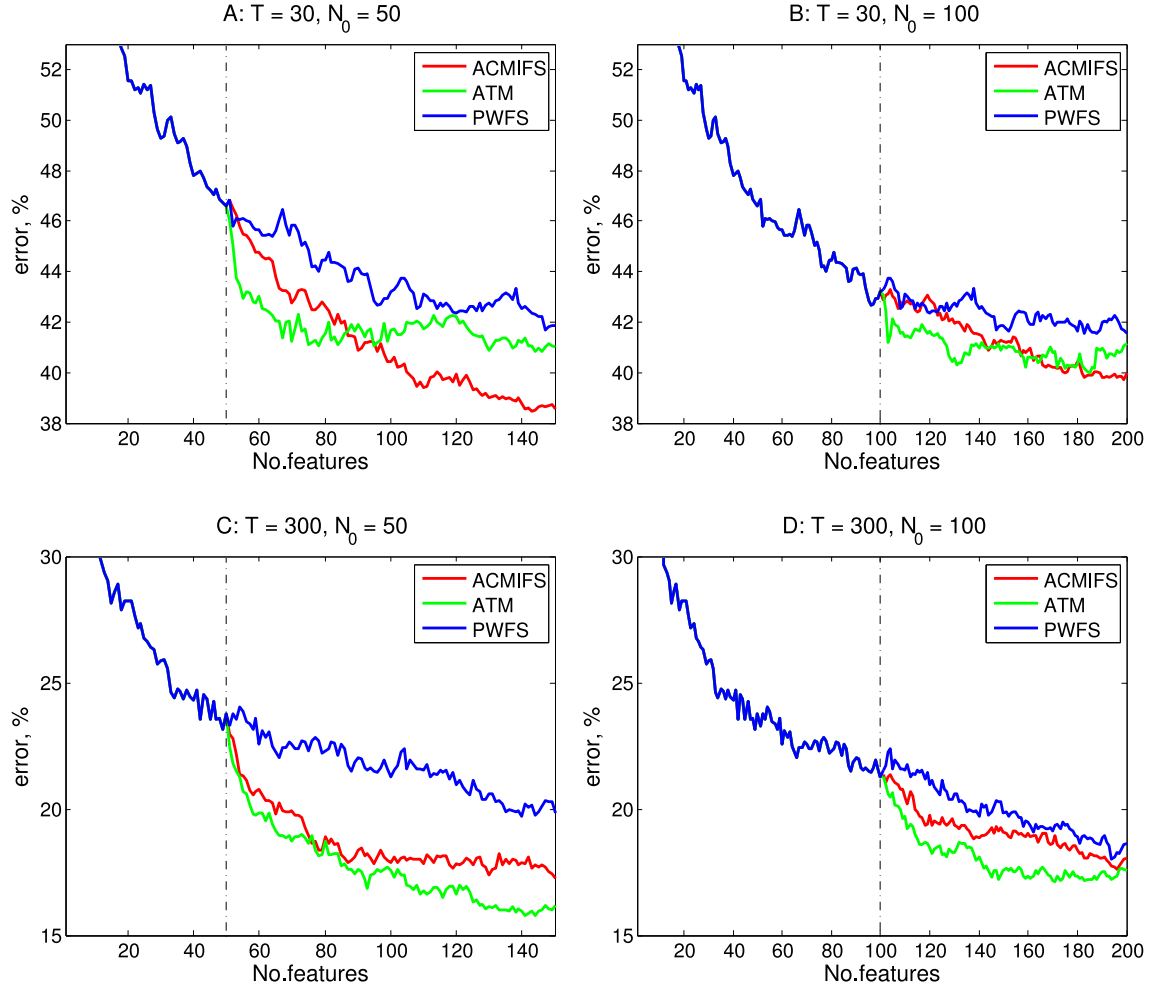


Figure 4.13: Comparison of the ability to add informative features to subsets of 50 and 100 features preselected by PWFS.

before, this fact suggests that after a certain dimension ACMIFS is not able anymore to estimate correctly high-order dependencies between the features. Interestingly, when ACMIFS selects the features from the beginning (see Fig. 4.10), it performs better than ATM almost up to 200 features, meaning that the first good features can compensate for unreliable pdf estimates further in higher dimensions. Based on this observation, one could think of a combined scheme that starts with ACMIFS and after selecting some features switches to ATM.

At the same time, for a very small training set with $T = 30$, we observe that ACMIFS after some iterations “recovers” and starts performing better than ATM. Moreover, Figure 4.10 demonstrates that the relative difference in classification performance of these two adaptive selection schemes is much larger for the case where $T = 30$ as well. In

order to understand a possible connection between an amount of the training data and the accuracy of ATM, let us look at a toy example illustrated by Figure 4.14.

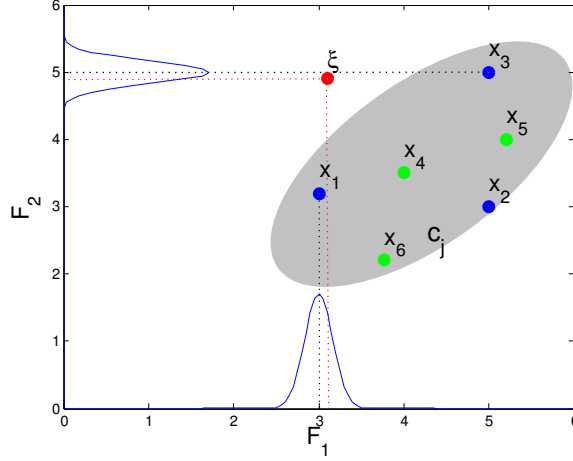


Figure 4.14: Illustration of a situation when a testing sample ξ is close to some samples from the class c_j along each of the dimensions F_1 or F_2 separately but not along both of them simultaneously. Blue curves indicate one-dimensional Gaussians centered at \mathbf{x}_1 and \mathbf{x}_3 along F_1 and F_2 , respectively.

The gray ellipsoid on Figure 4.14 indicates a region of the input space belonging to some class c_j . Assume there are two training sets of a different size describing this class: a small set consisting of three samples $\mathcal{X}_j = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$ and the extended one with six samples $\mathcal{X}_j^{ext} = \{\mathbf{x}_1, \dots, \mathbf{x}_6\}$. Suppose that we have to estimate the probability of the testing sample ξ given the class c_j after observing the values of the selected features F_1 and F_2 . Also suppose that the testing sample ξ is located in the way that it is close to one point in the dimension F_1 and to another point in the dimension F_2 . In our example, these are the points \mathbf{x}_1 and \mathbf{x}_3 , respectively. Here, we have a situation when the features F_1 and F_2 are not independent given c_j since the axes of the ellipsoid which describes the region of the class c_j are not parallel to the axes of the features. Thus, the assumption of ATM that features are class-conditionally independent is violated.

Obviously, the probability of ξ given the class c_j estimated by ACMIFS from the training set \mathcal{X}_j with the Gaussian kernels will be close to 0:

$$p(\xi|c_j)^{ACMIFS} = \frac{1}{T_j h_1 h_2 2\pi} (K_1(\xi, x_1)K_2(\xi, x_1) + K_1(\xi, x_2)K_2(\xi, x_2) + K_1(\xi, x_3)K_2(\xi, x_3)) \approx$$

$$\frac{1}{T_j h_1 h_2 2\pi} (1 * 0 + 0 * 0 + 0 * 1) = 0, \quad (4.7)$$

which holds for the estimate from the extended training set \mathcal{X}_j^{ext} as well. However, this is not the case for ATM. $p(\xi|c_j)$ estimated by ATM from the small training set \mathcal{X}_j is:

$$\begin{aligned} p(\xi|c_j)^{ATM} &= p(\xi_1|c_j)p(\xi_2|c_j) = \\ &= \frac{1}{T_j h_1 \sqrt{2\pi}} (K_1(\xi, x_1) + K_1(\xi, x_2) + K_1(\xi, x_3)) \frac{1}{T_j h_2 \sqrt{2\pi}} (K_2(\xi, x_1) + K_2(\xi, x_2) + K_2(\xi, x_3)) \approx \\ &= \frac{1}{T_j^2 h_1 h_2 2\pi} (1 + 0 + 0)(0 + 0 + 1) \approx \frac{0.16}{T_j^2 h_1 h_2} \end{aligned} \quad (4.8)$$

Remember that a bandwidth h defined by the normal reference rule depends on the number of the training samples (see the expression (3.21)). Since ATM works with the univariate pdfs, taking $d = 1$ we have

$$p(\xi|c_j)^{ATM} = \frac{0.16}{T_j^2 (1.06\sigma_1 T_j^{-1/5})(1.06\sigma_2 T_j^{-1/5})} = \frac{0.16}{\sigma_1 \sigma_2 (1.06 T_j^{4/5})^2}. \quad (4.9)$$

Without loss of generality, we suppose that the standard deviations of features σ_1 and σ_2 do not change much for different sizes of training sets or at least $(\sigma_1 \sigma_2)$ decreases slower than $(1.06 T_j^{4/5})^2$ increases.

Then, for the given example, it is obvious that the degree of overestimation of $p(\xi|c_j)^{ATM}$ will grow as the number of the training samples decreases. For comparison, the estimates based on small and extended training sets are approximately $\frac{0.024}{\sigma_1 \sigma_2}$ and $\frac{0.008}{\sigma_1 \sigma_2}$, respectively.

This example demonstrates a possible reason of ATM being much worse than ACMIFS for very small training sets. Although in such situations estimates of high-dimensional pdfs used by ACMIFS are not accurate, the estimates of ATM can get contaminated already on the very first iteration due to the wrong assumption of features being class-conditionally independent.

4.4 Combined selection scheme

The above presented experiments for $T = 300$ provided the evidence that ATM can select better features in higher dimensions. This suggests that at some point the pdf estimates can be improved if we adopt the assumption of features being class-conditionally independent. Since ATM is less computationally expensive, such switching would bring an additional advantage by reducing the amount of the computational resources needed for the selection procedure.

However, ACMIFS shows better performance for early iterations. Thus, it is important to understand when is the right moment to switch. Intuitively, ACMIFS stops being useful as its selection criterion S degenerates, i. e. its value is the same for all features candidates, thus the further selection gets random. This happens when either ACMIFS cannot estimate properly the current multivariate pdfs or there are no informative features left, if the full feature set is known to contain uninformative features. Of course, within the sequential feature selection framework, there is always a danger that there exists a combination of the remaining features reducing the remaining uncertainty even if these features alone are not informative. But since there is no guarantee that this subset exists at all, we suppose that in case of the degenerated selection criterion we would not lose much by switching to any other reasonable selection scheme.

Thus, as soon as the selection criterion of ACMIFS is degenerated, the scheme switches to ATM. We define the selection criterion (3.12) as degenerated when its standard deviation is below a certain adjustable threshold δ , $\delta \ll 1$.

First, it is necessary to note that the standard deviation should be estimated for the criterion in the expression (3.15) where all terms that do not contribute to the maximization problem are still present. Let us denote this early form as S_0 and the simplified selection criterion expressed by (3.26) as S_1 . Then, recalling all applied simplifications, there is the following relation:

$$S_0 = \frac{1}{p(\xi^i)} \left(S_1 + \sum_{j=1}^m p(c_j) T_j^{-1} \sum_{x_r \in \mathcal{X}_j} K(\xi^i, x_r) \log \frac{T_j^{-1} p(c_j) p(\xi^i)}{T^{-1} p(c_j, \xi^i)} \right),$$

As the standard deviation is not influenced by the additive term after S_1 , the standard deviation of the estimated selection criterion S_1 should be corrected just by multiplying by $\frac{1}{p(\xi^i)}$.

In summary, ACMIFS stops when

$$\sigma \left(\frac{S_1}{p(\xi^i)} \right) < \delta, \quad (4.10)$$

where $\sigma(\cdot)$ is the standard deviation and δ is an adjustable threshold.

We suggest that while setting δ one should take into account the overall accuracy and the relative difference between ACMIFS and ATM observed for a certain dataset. Figure 4.15 supports our idea.

That is, the combined scheme with approximately the same accuracy as ACMIFS requires different values of δ for the training sets with 30 and 300 training samples. Since we observed that for $T = 30$ ACMIFS shows much better results than ATM (see Fig. 4.10), it is not surprising that it makes sense to switch to ATM later, hence, using a smaller δ .

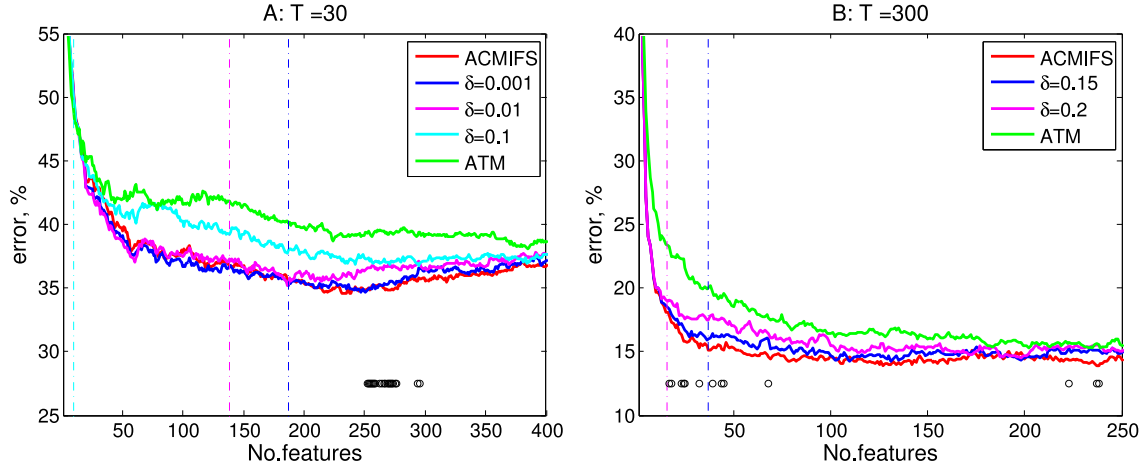


Figure 4.15: Classification performance of ACMIFS and the combined schemes starting with ACMIFS and switching to ATM according to the (4.10) for the different thresholds δ . The colored vertical dash-dotted lines indicate the average number of the features selected by ACMIFS before the switch; the color of these lines corresponds to the color of the error curve of the corresponding setup. A: The black markers indicate iterations where ACMIFS is significantly better than the combined scheme switched to ATM using $\delta = 0.01$ (red and magenta error curves). B: The black markers denote the iterations where ACMIFS significantly outperforms the combined scheme with a switch to ATM using $\delta = 0.15$ (red and blue error curves).

Though, for $T = 300$ the switch can occur already after about 15 iterations, these first features selected by ACMIFS are still very important. This can be seen by comparing the pure ATM and the combined scheme with $\delta = 0.2$.

Table 4.1 provides an overview of switching behavior of the schemes presented on Figure 4.15. It is interesting to note the huge difference between the minimal and the maximal number of the features before the switch. A small number of features before switching usually indicated the cases when testing samples were unambiguously classified. Thus, as few as only 3 features are necessary in order to reduce the uncertainty about the class for the samples which are easy to classify. At the same time, for more complex samples, a large number of the features appears to be informative enough according to the selection criterion of ACMIFS, hence, the switch occurred much later.

Also note the values err_{sw} in the table. They show the average error of our ACMIFS before the switch. If this error rate is acceptable and the feature subsets of the variable size are allowable, instead of switching to ATM one can stop the selection at all. Thus, the switching rule can be turned to the stopping rule. Such stopping criterion would also detect the situations when there is no uncertainty left about the class, as the conditional mutual information will be zero for all remaining features.

$T = 30$				$T = 300$		
	$\delta = 0.001$	$\delta = 0.01$	$\delta = 0.1$		$\delta = 0.15$	$\delta = 0.2$
N_{min}	3	3	3	N_{min}	3	3
N_{max}	413	382	71	N_{max}	353	183
N_{av}	187.1	138.9	10	N_{av}	37	15.6
err_{sw}	38.25	40.85	47.6	err_{sw}	16.15	17.45

Table 4.1: Summary of the combined schemes with different values of δ for the training sets with $T = 30$ and $T = 300$. N_{min} , N_{max} and N_{av} is the minimal, maximal and average number of the features selected by ACMIFS before switching to ATM, respectively. err_{sw} is the average error rate of ACMIFS just before switching.

4.5 Conclusions

Feature selection is a standard technique to reduce data dimensionality. In high-dimensional spaces, this can be an efficient way to cope with limited amounts of training data. Conventional methods assume selecting features on the preprocessing step before the actual classification. That is, one tries to find a small number of features that are discriminative in all regions of the input space. However, in situations with heterogeneous data or in the undersampled regime, it could be difficult to find a small subset of features that are relevant for classification of all samples. As a solution, we have proposed to adapt the selection process to every sample that has to be classified, that is to select few features that are informative only for this particular sample.

As a result, in Chapter 3, we have proposed an algorithm that employs a selection criterion based on mutual information. This choice is motivated by the connection of the resulted criterion to such fundamental information-theoretical concepts like independence and uncertainty reduction, as well as its use in classification, as was shown in Section 2.4. According to the adaptive scheme for sequential feedforward feature selection, each feature is selected as maximizing the expected mutual information with the class conditioned on the already selected features taking values observed on the testing sample.

Experimental investigations presented in this chapter provided evidence that adaptive feature selection robustly improves the classification performance despite the fact that estimating mutual information in high-dimensional spaces is a difficult problem on its own, as reviewed in Section 2.5. To estimate ACMIFS, we use a rather simple plug-in estimator in combination with the kernel density estimates, which are easy applicable for probability density functions with variable dimensionality due to its non-parametric nature. This property allows to reestimate the mutual information of a class and a feature-candidate conditioned on the growing set of selected features with the moderate computational expenses. Parametric techniques, whose contemporary representatives show good accuracy

in estimating mutual information, would need a complete retraining for every combination of features in order to find parameters of the underlying probability distribution. Among the non-parametric methods, the Kraskov estimator based on k nearest neighbors algorithm [Kraskov et al., 2004] is quite popular, however, is not applicable to our framework of the iterative selection and evaluation of features as it requires the knowledge about all feature values to fix and then iteratively refine the neighborhood of the classified object.

In order to improve estimates of the adaptive selection criterion, we apply adaptive smoothing to unreliable pdfs for extremely small datasets. In particular, we proposed to smooth extremely small values of pdfs under the logarithm and suggested that smoothing values should be adjusted both to the current dimensionality of smoothed pdfs and to the current probability of a classified sample in order to avoid oversmoothing and degeneration of the selection criterion.

Our results on both artificial and real-world data showed that a small number of adaptively selected features is sufficient to achieve good classification. In this sense, ACMIFS outperforms the two related static and adaptive feature selectors, PWFS and ATM, respectively. This fact suggests that the adaptive feature selection is indeed advantageous when solving complex classification tasks in the undersampled regime, as well as efficiency of the proposed estimation technique. Since the first few features can be reliably detected, our method does not overfit and can even compensate for shortcomings of a classifier. I. e., in case of limited training data, when a classifier is usually prone to overfitting, we demonstrated that ACMIFS can even improve the error rate compared to using all available features.

Even though the algorithm is less advantageous on large datasets, we believe that this is not a shortcoming, but merely shows that the need to select features is less pressing if enough data are available. From the point of view of computational expenses, in order to make ACMIFS more applicable to large amount of data, one has to think about an approximate implementation which can cut down the computational complexity. Alternatively, one can consider using some hybrid simplifying scheme, for example, starting with ACMIFS and then after some iterations switching to ATM, which does not require estimating multivariate densities and therefore is computationally cheaper. Then, by tuning the switching criterion, one can reach a compromise between the quality of the selected features and complexity of the selection algorithm.

Chapter 5

Information-theoretical strategies of selective attention

5.1 Introduction

Early visual perception in the human brain, which includes processing of primitive features such as color, orientation, motion, is known to be massively parallel. However, later processes like feature integration, object recognition and identification are more complex and therefore separate parts of a visual scene have to be processed sequentially due to the lack of computational resources [Rolls, 2008]. Obviously, in order to speed up high-level visual processing, it is important for the brain to concentrate on the most relevant parts of the input. Selection of relevant sources of the sensory information is exactly the function of the external selective attention [Johnston & Dark, 1986]. In addition, it was suggested to distinguish between the external and the so-called internal attention, which operates upon the internal information such as long-term and working memory contents, task rules etc [Chun et al., 2011]. However, here, we are interested in principles of the external attention, therefore, further “attention” will refer to the external attention only.

It is well-established that there are two factors influencing the attention: visual stimuli themselves and a task [van de Laar et al., 1997]. The stimuli-driven attentional mechanism prefers salient objects, i. e. the objects that differ from their neighborhood. Thus, saliency of a stimulus is defined based on its low-level characteristics or features like color, luminance and orientation. Such characteristics are believed to be extracted in fast preattentive manner via bottom-up circuits of the visual system. Then, according to the feature integration theory developed by Treisman and Gelade [Treisman & Gelade, 1980], only those parts of the visual scene that differ from neighboring regions are highlighted by the attentional mechanism for further integration and processing. In spite of the fact

that this theory received a lot of criticism, it remains quite influential and serves as the basis for numerous computational models which predict salient image locations based on their low-level features, e. g. [Koch & Ullman, 1985; Wolfe, 1994; Itti & Koch, 2000].

Although low-level image statistics undoubtedly affects shifts of the visual attention, as was shown for example by [Ossandón et al., 2012], it is not the only influential factor. Eye tracking experiments, which are widely used to study overt attention, i. e. attention whose effects are characterized by physical movements, provided much evidence that a saccade sequence differs depending on a task, see for example [Yarbus, 1967; Rothkopf et al., 2007; Betz et al., 2010]. This fact suggests that people direct their attention to locations that contain task-relevant information. For example, it was demonstrated that while executing a task, subject fixations were directed to objects that were involved in this task [Land et al., 1999; Hayhoe et al., 2003]. Therefore, contemporary models of attentional selection use not only the low-level stimulus saliency but reweight it also with the task-specific prior [Navalpakkam & Itti, 2005] and context information [Ehinger et al., 2009; Torralba et al., 2006], whose influence is believed to be implemented via top-down circuits in the visual system [van de Laar et al., 1997].

However, the question remains what kind of strategy people use to decide what is relevant for a task. Do we use simple heuristics or complex algorithms based on the ideas of information theory? Surprisingly, despite their computational complexity, statistical and information-theoretical definitions of the task-relevance are the core of state-of-the-art algorithms predicting eye movements. Let us look at some of them more closely. In order to introduce them in the context of our selection framework, we will stay within our usual notation.

5.1.1 Existing task-dependent strategies of selective attention

The widely accepted contextual guidance model combines the bottom-up saliency with the class-specific contextual prior that are formed by outputs of local and global features, respectively [Torralba et al., 2006]. The local features provide information on low-level characteristics of an image like contrast, orientation, color etc. In turn, the global features represent the statistics of integrated responses of the local features. Both type of features are evaluated after first glance at the image. However, for a visual search task, this information is not sharp enough for identifying a target immediately, therefore the sequential search should be performed. According to the contextual guidance model of Torralba and colleagues, for an image ξ , a location $F_{\alpha_{i+1}}$ selected on the $(i+1)^{th}$ iteration

should maximize the posterior probability of the target C conditioned on the local and global measurements of the image, ξ_L and ξ_G , respectively:

$$\alpha_{i+1} = \arg \max_k \{p(c, F_k | \xi_L, \xi_G)\} = \arg \max_k \left\{ \frac{p(F_k | c, \xi_G)}{p(\xi_L | \xi_G)} \right\}, \quad F_k \in \mathcal{F} \setminus \{F_{\alpha_1}, \dots, F_{\alpha_i}\}, \quad (5.1)$$

where $p(F_k | c, \xi_G)$ is the context-based prior and $p(\xi_L | \xi_G)$ is the probability of observing the local features given the global image statistics. However, one has to note that the selection criterion is not explicitly influenced by the information gained at the attended locations, as values of the global and local features are not updated after the fixation is made. In this case, one can say that such model ranks all locations according to their task-dependent saliency rather than predicting a subset of the informative locations sufficient for executing a task.

In turn, it was suggested that while performing a visual search in the complex environment with several targets c_1, \dots, c_m , rewards associated with every target influence the selection criterion as well [Navalpakkam et al., 2010]:

$$\alpha_{i+1} = \arg \max_k \left\{ \sum_{j=1}^m r_j p(c_j, F_k | \xi) \right\}, \quad (5.2)$$

where r_j is the reward of the j^{th} target.

Alternatively, Kanan and colleagues proposed that the prior on the target appearance when combined with the saliency can explain eye movements during the visual search [Kanan et al., 2009]. The selection criterion, which they call the ‘‘pointwise mutual information’’, is also based on the class posterior but with respect to the location-candidate only:

$$\alpha_{i+1} = \arg \max_k \{\log p(c | F_k = \xi_k)\} = \arg \max_k \left\{ \log \frac{p(F_k = \xi_k | c)}{p(F_k = \xi_k)} \right\}, \quad (5.3)$$

where ξ_k represents values of the local features in the location F_k and $p(F_k = \xi_k | c)$ is the likelihood of observing ξ_k for the class or target c , which is based on the knowledge of its appearance.

Itti and Baldi suggested another definition of the task-dependent saliency, which uses a notion of the so-called Bayesian surprise [Itti & Baldi, 2006]

$$\alpha_{i+1} = \arg \max_k \{D_{KL}(p(c | F_k = \xi_k) || p(c))\}, \quad (5.4)$$

where $D_{KL}(\cdot)$ is the Kullback-Leibler divergence defined earlier in (2.40). This criterion suggests that surprising and unexpected image locations with respect to the prior knowledge are likely to be attended. Results of the eye-tracking experiment performed by the

authors showed that such definition of the saliency can better explain human fixations than the entropy-based version of their criterion, i. e. $\arg \max_k \{p(c, F_k = \xi_k) \log p(c|F_k = \xi_k)\}$.

In addition to the above presented rankers, there are also models whose selection criteria make use of information collected during the previously attended locations. This is especially important in cases when a visual scene or its parts can change during the task execution [Tatler et al., 2011].

For example, Najemnik and Geisler proposed two hypotheses about strategies that people might use to select a next fixation while performing a search task [Najemnik & Geisler, 2005]. These are the maximum a posteriori and the ideal Bayesian search. Both strategies are based on maximizing the posterior probability of a target being at the next location. But the posterior is conditioned only on the value of the location-candidate F_k , which is however integrated over the previous i fixations. This value for the location F_k will be denoted as $\xi_k(i)$. Obviously, the amount of information that can be gained about F_k while fixating on another location depends on the distance between them. Then, the above-mentioned posterior probability of a target C being at the location F_k after i fixations is:

$$p(c, F_k | \xi_k(i)) = \frac{p(c, F_k) p(\xi_k(i) | c, F_k)}{\sum_j p(c, F_j) p(\xi_j(i) | c, F_j)}, \quad F_k \in \mathcal{F} \setminus \{F_{\alpha_1}, \dots, F_{\alpha_i}\}, \quad F_j \in \mathcal{F}. \quad (5.5)$$

Using this definition, the selection criterion according to the maximum a posteriori search (MAP) is

$$\alpha_{i+1} = \arg \max_k \{p(c, F_k | \xi_k(i))\}, \quad F_k \in \mathcal{F} \setminus \{F_{\alpha_1}, \dots, F_{\alpha_i}\} \quad (5.6)$$

whereas the selection criterion of the ideal Bayesian searcher is the following:

$$\alpha_{i+1} = \arg \max_k \left\{ \sum_l p(c, F_l | \xi_l(i)) p(o | F_l, F_k) \right\}, \quad F_k \in \mathcal{F} \setminus \{F_{\alpha_1}, \dots, F_{\alpha_i}\}, \quad F_l \in \mathcal{F}, \quad (5.7)$$

where $p(o | F_l, F_k)$ is the probability of observing the location F_l from F_k . That is, MAP favors the location that has the maximum expected class posterior given the currently available information about this location. However, the ideal searcher prefers locations with the maximally informative neighborhood, i. e. with the visible neighbors that in total have high expected posterior. The comparison of human saccade sequences with those generated by MAP and the ideal searcher suggested that human behavior is more compatible with the ideal Bayesian search despite the fact that it is more computationally demanding than the greedy MAP.

In contrast, the model of Renninger and colleagues uses the information of the previously attended locations fully and selects the next location as to minimize the entropy about the class conditioned on the currently available information [Renninger et al., 2007]:

$$\alpha_{i+1} = \arg \min_k \{H(C | \xi^i, F_k)\}, \quad (5.8)$$

where ξ^i are the values of all image locations, either sharp or of low-resolution, which are estimated using the information collected during the previous i fixations. Eye-tracking experiments performed by the authors revealed that human fixations are compatible with the proposed model minimizing the local uncertainty, i. e. when candidates for the next fixation are chosen from the vicinity.

Here, in our opinion, we presented the most prominent models that predict human fixations while performing a task. As one can notice, determining the task-relevant parts of a visual scene is closely related to the problem of feature selection in machine learning. Correspondingly, we would like to test to what extent human behavior can be explained in terms of feature selection criteria, in particular information-theoretical feature selection criteria.

It should be noted that our aim is not to predict eye movements but rather to test several possible strategies of the task-dependent selective attention. Therefore, for simplification, the considered strategies take into account only the task-dependent component of the attentional selection.

5.2 Experimental setup

In order to test the strategies, we use a clicking experiment, which is a more constrained setup than eye-tracking [Avdiyenko et al., 2012a]. During the experiment, a subject is presented with a covered image and its patches can be uncovered by clicking at them. The task is to uncover a minimal number of patches that, in the subject's opinion, help to identify the class of the image. The presentation stops when the class is unambiguously identified. Therefore, if the information provided by the uncovered patches is enough to classify the image, it is considered as unambiguously identified even if some patches are still covered. The stopping point is defined externally by the "computer" and not by a subject, which is done in order to have the same stopping rule for all subjects. A single stopping rule simplifies the experiment analysis reducing a number of free parameters. Thus, as long as for a certain image all patches are selected according to the same strategy, the corresponding patch sequences should be identical.

There are also time constraints saying that a subject should not spend more than 2-3 seconds to make a decision about the next click. One should note that subjects are just instructed not to exceed this limit, however, no signal is given if it happens. In this way, we hope to observe intuitive but nevertheless non-random behavior, which could be caused by the strict time constraints on the decision time.

By keeping only the attended locations uncovered, one can simulate the simplified conditions for studying a top-down component of the visual search alone. It means that only

information observed at the uncovered locations are available to a subject for decision-making. As location-candidates are not initialized with their values of low resolution, there is no influence of low-level image features on the choice of the next fixation. Thus, the attended locations are always chosen because of their informativeness for the task.

The advantage of the clicking setup is its obvious simplicity. We believe that findings of our clicking experiment can be accepted as valid for the task-dependent attentional mechanism. Although it is clear that time intervals between clicks are larger compared to saccades, it is not crucial for our experiment as we are interested only in the precise succession of the selected patches. Also one can hypothesize that a subject will click where she would direct her visual attention as all hand movements are proven to be in the gaze-centered coordinates and influenced by updates in the visual representation [Batista et al., 1999; Henriques et al., 2002; Medendorp et al., 2003]. Moreover, researchers from the field of information retrieval reported the strong correlation between eye movements and computer mouse movements of humans while performing information search using a search engine, e. g. [Navalpakkam et al., 2013].

In addition, we would like to comment on whether task-driven eye movements are conscious or unconscious actions. First of all, let us define what conscious and unconscious perception is. According to definitions given in [Dijksterhuis & Aarts, 2010], people are aware of results of conscious perception and can give a verbal report about them, whereas unconscious processes stay “invisible” for them. Further, it is known that unconscious processes are faster than conscious ones but also less stable [Breitmeyer & Tapia, 2011; Mattler, 2005; Dehaene et al., 2006]. The reason for this is that the consciously formed representation of a visual input is based on information integrated from different visual areas involving interaction with working memory [Enns & Di Lollo, 2000; Macknik & Martinez-Conde, 2008; Kiefer et al., 2011]. Further, what is the relation between consciousness and task-dependent attention? Although for a long time consciousness and the attentional control were considered to be very close processes [Posner & Petersen, 1990; Chun & Wolfe, 2001; O’Regan & Noe, 2001], recent works argue that consciousness and attention have different functions and they are realized via different neuronal mechanisms [Koch, 2004; Dehaene et al., 2006; Kietzmann et al., 2011]. However, the top-down attentional control can function in the unconscious mode only if a task is formulated before a stimulus presentation and there is no need to react to a stimulus adaptively [Kiefer & Martens, 2010].

A process of selecting a next saccade in presence of a task, which is discussed here, is iterative. Therefore, it requires integration of bottom-up information about a stimulus with top-down attentional instructions and it is likely to rely on the content of working memory [de Fockert et al., 2001]. Moreover, as was shown above, task-dependent eye movements are influenced by a visual scene, thus they are adaptive. As a result, we hypothesize that

a process directing eye movements of people while performing a task is conscious since all its listed properties conflict with the definition of unconscious perception.

5.3 Tested information-theoretical search strategies

In our setup, a feature F is an image patch, i. e. an oriented bar at the certain location. As before, the variable C represents the classes of the images $C = \{c_1, c_2, \dots, c_{10}\}$. We would like to emphasize that all above-presented models were tested for quite simple tasks like search or discrimination that can be described by the binary class variable. In contrast, we intend to have images that can belong to 10 different classes, which extremely increases the complexity of estimating a selection criterion. Under such conditions, it is interesting to see to which extent people are able to act optimally.

5.3.1 Mutual information

The first strategy is based on the Mutual Information selection criterion (MI), which was earlier introduced as S_{MIM} in (2.91). It is considered as a heuristic because it assumes that features are independent. Therefore, it simply ranks all patches (features) according to the mutual information they provide about the image class. Then, according to the MI strategy, the informativeness of the patch F_k after i patches have been uncovered is:

$$I_{MI}(F_k) = I(C; F_k) = \sum_C \sum_{F_k} p(c, f_k) \log \frac{p(c, f_k)}{p(c)p(f_k)}, \quad F_k \in \mathcal{F} \setminus \{F_{\alpha_1}, \dots, F_{\alpha_i}\}, \quad (5.9)$$

where F_{α_j} is a patch selected or uncovered on the j^{th} iteration.

5.3.2 Conditional mutual information

The second strategy is the Conditional Mutual Information selection criterion (CMI). In contrast to MI, it takes into account high-order dependencies between features, i. e. selects those that are both informative and non-redundant with respect to the already selected features. Thus, according to CMI, the informativeness of the unattended patch F_k after i steps is defined as follows:

$$I_{CMI}(F_k) = I(C; F_k | F_{\alpha_1}, \dots, F_{\alpha_i}) = \sum_C \sum_{F_{\alpha_1}} \dots \sum_{F_{\alpha_i}} \sum_{F_k} p(c, f_k, f_{\alpha_1}, \dots, f_{\alpha_i}) \log \frac{p(c, f_k | f_{\alpha_1}, \dots, f_{\alpha_i})}{p(c | f_{\alpha_1}, \dots, f_{\alpha_i}) p(f_k | f_{\alpha_1}, \dots, f_{\alpha_i})}, \quad (5.10)$$

where $\{F_{\alpha_1}, \dots, F_{\alpha_i}\}$ is the set of already uncovered patches.

5.3.3 Adaptive conditional mutual information

The third strategy is the Adaptive conditional Mutual Information feature selection criterion (AMI) proposed in this thesis in Section 3.5. This is an adaptive version of the CMI criterion that takes also into account observed values of the already attended patches, suggesting that every next decision depends on what one has seen in the previous steps. Therefore, the informativeness of the image patch F_k after i patches have been uncovered is expressed in the following form:

$$I_{AMI}(F_k) = I(C; F_k | \xi^i) = \sum_C \sum_{F_k} p(c, f_k | \xi^i) \log \frac{p(c, f_k | \xi^i)}{p(f_k | \xi^i) p(c | \xi^i)}, \quad (5.11)$$

where $\xi^i = \{F_{\alpha_1} = \xi_{\alpha_1}, \dots, F_{\alpha_i} = \xi_{\alpha_i}\}$ is a shorthand for the set of values, which are observed on the selected patches of the image ξ . As $I(C; F_k | \xi^i) = H(C | \xi^i) - H(C | F_k, \xi^i)$ and $H(C | \xi^i)$ does not influence the $\arg \max$ operator, this selection criterion is similar to one used by Renninger and colleagues (5.8).

The first two strategies give a single sequence of informative patches for all images of the given image set, therefore we call them static. One can make a parallel to the above presented strategies that do not update the task-relevance of possible locations with the information gathered from previous saccades, though the formal definition of a task-relevance criterion differs from ours [Torralba et al., 2006; Navalpakkam et al., 2010; Kanan et al., 2009; Itti & Baldi, 2006]. The advantage of the static approach is that an informative sequence can be defined once and then used for classification of all images. In contrast, the adaptive strategy defines an informative sequence for every image that is classified. One can say that static criteria use only the prior knowledge about the relevance of features for classification, whereas the adaptive strategy combines the prior with the information acquired during the previous iterations. On the one hand, the information integration is more computationally demanding and was shown to be difficult for people [Irwin, 1991; Hayhoe et al., 1998], on the other hand, feature sequences selected adaptively are usually on average shorter.

5.4 Sequence statistics

Once experimental data are collected, we have to analyze how good each of the information-theoretical strategies explains the clicking behavior of every subject. It is necessary to emphasize that we do not generalize over all subjects but rather analyze everyone individ-

ually. For this, we adopt the Bayesian approach. Theoretically, we have to compare the posterior probabilities of the strategies given the subject's observed clicks:

$$p(s|\mathcal{F}_{\xi_1}, \dots, \mathcal{F}_{\xi_T}) = \frac{p(\mathcal{F}_{\xi_1}, \dots, \mathcal{F}_{\xi_T}|s)}{p(\mathcal{F}_{\xi_1}, \dots, \mathcal{F}_{\xi_T})p(s)}, \quad (5.12)$$

where s is a certain strategy, T is a number of the presented images and the set \mathcal{F}_{ξ_j} stands for a sequence of the patches selected for the image ξ_j by the currently analyzed subject. Thus, $\mathcal{F}_{\xi_j} = \{F_{\alpha_1}(\xi_j), \dots, F_{\alpha_{K_j}}(\xi_j)\}$, where $F_{\alpha_i}(\xi_j)$ is the i^{th} selected patch while classifying an image ξ_j . Note that a presentation of the image is terminated when this image is unambiguously identified. Therefore, the length of the patch sequence K_j is not constant (the subscript j for K_j is omitted in $F_{\alpha_K}(\xi_j)$ to avoid multilevel subscripts). It depends on the amount of the remaining uncertainty about the image class, which is evaluated using the information provided by the sequentially uncovered patches.

There is the problem that we do not know the set of all possible strategies s in order to estimate the normalization factor in (5.12). Our analysis is limited to the information-theoretical feature selection strategies, which probably constitute a small fraction of all strategies people may use in their everyday life. However, since the normalization factor $p(\mathcal{F}_{\xi_1}, \dots, \mathcal{F}_{\xi_T})$ is the same for all strategies and if we assume they are equiprobable a-priori, we can just as well compare their marginal likelihood $p(\mathcal{F}_{\xi_1}, \dots, \mathcal{F}_{\xi_T}|s)$.

Assuming that the presented images are independent and identically distributed, the marginal likelihood of all images given a strategy s is just a product of the likelihood of the single images given this strategy:

$$p(\mathcal{F}_{\xi_1}, \dots, \mathcal{F}_{\xi_T}|s) = \prod_{j=1}^T p(\mathcal{F}_{\xi_j}|s). \quad (5.13)$$

5.4.1 Generative model

Here, we present a generative model that describes how an observed patch sequence is generated for a single image. Basically, we suppose that parts of an image are uncovered according to their mutual information, either unconditionally (MI), conditionally (CMI) or adaptively (AMI) taking the already uncovered patches into account. Furthermore, we assume that humans choose the next patch as the one softly maximizing the corresponding information. Then, we propose the following generative model for a patch sequence:

$$p(\mathcal{F}_{\xi_j}|s) = p(F_{\alpha_1}, \dots, F_{\alpha_{K_j}}|\xi_j, s) = \prod_{i=1}^{K_j} p(F_{\alpha_i}|F_{\alpha_1}, \dots, F_{\alpha_{i-1}}, \xi_j, s), \quad (5.14)$$

where K_j is the length of the observed patch sequence. To simplify the notation, $p(F_{\alpha_1}, \dots, F_{\alpha_K}|\xi_j, s)$ stands for $p(F_{\alpha_1}(\xi_j), \dots, F_{\alpha_K}(\xi_j)|s)$.

Finally, the likelihood of a single patch of the image ξ_j being selected according to the strategy s is:

$$p(F_{\alpha_i}|F_{\alpha_1}, \dots, F_{\alpha_{i-1}}, \xi_j, s, \beta) = \frac{e^{\beta I_s(F_{\alpha_i})}}{\sum_q e^{\beta I_s(F_q)}}, \quad \beta \geq 0, F_q \in \mathcal{F} \setminus \{F_{\alpha_1}, \dots, F_{\alpha_{i-1}}\}, \quad (5.15)$$

where β is a free parameter, $I_s(F_{\alpha_i})$ is informativeness of the i^{th} patch given by a strategy s and F_q 's are the patches that could be selected instead F_{α_i} on the i^{th} iteration. The expression (5.15) is in fact the softmax function. It expresses a probability of the patch F_{α_i} being selected from all candidates on the i^{th} iteration if the choice was guided by the informativeness of these patch-candidates $I_s(\cdot)$. The parameter β shows how much of the probability mass is concentrated on the highly informative patches. With $\beta = 0$, the distribution is uniform and a completely random strategy, i. e. a random distribution on patch sequences, is specified. When $\beta \rightarrow \infty$, only the best patch can be selected in each step. Since the true value of β for each subject is unknown, it should be inferred from the observed data.

Recall that only $I_{AMI}(\cdot)$ takes into account the values of the already uncovered patches $\xi_i = \{F_{\alpha_1} = \xi_{\alpha_1}, \dots, F_{\alpha_k} = \xi_{\alpha_k}\}$, see (5.9), (5.10) and (5.11). Therefore, the likelihood of the patch sequence according to each considered strategy is of the following forms:

$$p(\mathcal{F}_{\xi_1}, \dots, \mathcal{F}_{\xi_T} | s, \beta) = \begin{cases} \prod_{j=1}^T \prod_{i=1}^{K_j} e^{\beta I(F_{\alpha_i}; C)} / \sum_q e^{\beta I(F_q; C)}, & \text{for } s_{MI} \\ \prod_{j=1}^T \prod_{i=1}^{K_j} e^{\beta I(F_{\alpha_i}; C | F_{\alpha_1}, \dots, F_{\alpha_{i-1}})} / \sum_q e^{\beta I(F_q; C | F_{\alpha_1}, \dots, F_{\alpha_{i-1}})}, & \text{for } s_{CMI} \\ \prod_{j=1}^T \prod_{i=1}^{K_j} e^{\beta I(F_{\alpha_i}; C | \xi_j^{i-1})} / \sum_q e^{\beta I(F_q; C | \xi_j^{i-1})}, & \text{for } s_{AMI} \end{cases} \quad (5.16)$$

where $F_q \in \mathcal{F} \setminus \{F_{\alpha_1}, \dots, F_{\alpha_{i-1}}\}$.

Ideally, the likelihood should also take into account a stopping model that tells when the patch sequence should terminate. However, since we enforce an external stopping criterion (the posterior of any class being equal to one), we know that it does not depend on the strategy. In this case, for a certain sequence, the probabilities for the stopping variable on every iteration $p(t_i)$ are multiplicative constants:

$$p(F_{\alpha_i} | F_{\alpha_1}, \dots, F_{\alpha_{i-1}}, \xi_j, s, \beta, t) = p(t_i) p(F_{\alpha_i} | F_{\alpha_1}, \dots, F_{\alpha_{i-1}}, \xi_j, s, \beta), \quad (5.17)$$

where $p(t_i)$ is formally the probability of the sequence being terminated after i uncovered patches. Then, the stopping variable can be canceled when considering marginal likelihood ratios like

$$\frac{p(\mathcal{F}_{\xi_1}, \dots, \mathcal{F}_{\xi_T} | s_{AMI}, \beta)}{p(\mathcal{F}_{\xi_1}, \dots, \mathcal{F}_{\xi_T} | s_{CMI}, \beta)}.$$

5.4.2 Base likelihood

Let the likelihood, which is achieved with the $\beta = 0$, be called the base log-likelihood. As was already mentioned, in the case of $\beta = 0$, a value of feature informativeness $I_s(\cdot)$ does not have any influence on the probability of this feature to be chosen, which would agree with the perfectly random behavior. Therefore, we call this strategy s_{rand} :

$$p(\mathcal{F}_{\xi_1}, \dots, \mathcal{F}_{\xi_T} | s_{rand}) = p(\mathcal{F}_{\xi_1}, \dots, \mathcal{F}_{\xi_T} | s_{any}, \beta = 0) = \prod_{j=1}^T \prod_{i=1}^{K_j} \frac{e^0}{\sum_q e^0} = \prod_{j=1}^T \prod_{i=1}^{K_j} \frac{1}{K_j - i + 1}, \quad (5.18)$$

where $(K_j - i + 1)$ is a number of the candidates for selection on the i^{th} iteration. Note that the base likelihood depends on the length of patch sequences. The shorter a sequence is, the more likely is the random behavior. For example, if the sequence has 2 patches, then its likelihood according to the random strategy s_{rand} is:

$$p(F_{\alpha_1}, F_{\alpha_2} | \xi_j, s_{rand}) = \frac{1}{7} \times \frac{1}{6} \approx 0.024, \quad (5.19)$$

since there are 7 and 6 features to choose from on the first and the second iterations, respectively. At the same time, the likelihood of s_{rand} for the sequence consisting of 4 patches is much smaller:

$$p(F_{\alpha_1}, \dots, F_{\alpha_4} | \xi_j, s_{rand}) = \frac{1}{7} \times \frac{1}{6} \times \frac{1}{5} \times \frac{1}{4} \approx 0.001. \quad (5.20)$$

Recall that the generative model of s_{rand} can be derived from the generative model of any considered information-theoretical strategy using $\beta = 0$. Consequently, the maximum likelihood of MI, CMI and AMI is by definition not lower than the base likelihood and its absolute value alone does not tell us much. Hence, it makes sense to look at the ratio of the maximum likelihood to the base likelihood, i. e. the likelihood of the random strategy. For convenience, we will analyze the log-ratio:

$$r = \log\left(\frac{p(\mathcal{F}_{\xi} | s, \beta^*)}{p(\mathcal{F}_{\xi} | s_{rand})}\right) \quad (5.21)$$

that shows to which extent the strategy s with its fitted parameter β^* can better explain the subject's behavior compared to the random strategy.

5.5 Experiment

5.5.1 Stimuli

In our clicking experiment, as stimuli we used images of clock digits, where every image consists of seven patches. That is, the images are described by seven features and can belong to one of the ten classes. We took this stimuli set assuming that all people know them very good, so they do not need much time for learning in order to use fully the data statistics to make complex decisions. Figure 5.1 shows images of the digits used in the experiment. Examples of the covered, partially uncovered and unambiguously identified and completely uncovered images for the considered data set can be seen on Figure 5.2.

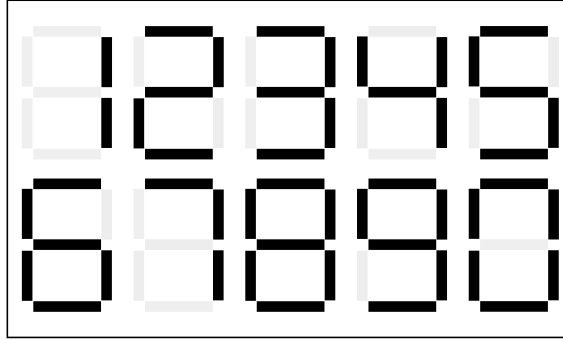


Figure 5.1: Clock digits that are used as stimuli in the experiment.

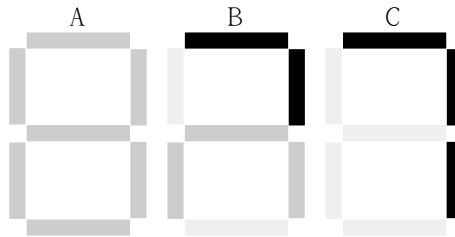


Figure 5.2: Examples of stimuli. A: covered image, B: digit, which is partially uncovered and unambiguously identified as '7', C: completely uncovered image.

Before the experiment, we generated a sequence of 80 clock digit images, i. e. 8 times the data set. Then, every subject was presented with 50 images that were randomly picked from this sequence. Though, the data set is considered to be familiar for all subjects, we define a learning phase that lasts during the presentation of the first 20 images. Therefore, the analysis is done on the last 30 images.

5.5.2 Presentation software

The experiment is written and presented using FlashDot, a software for generating visual presentations for experiments in psychophysics and vision [Elze, 2009].

For a FlashDot presentation, we defined a stimulus that consists of horizontally and vertically oriented bars (image patches) located in the way to resemble a clock digit. Interactivity with a subject is implemented by responding to events like a mouse click on the image patches and buttons.

While buttons are used to simply start a presentation of the next image if the current one is already identified, the response to the mouse click on the certain bar is more complex. Recall that at the beginning a subject is presented with the completely covered image. It is achieved by displaying a gray template of the clock digit (see subplot A on Figure 5.2) so that it is clear where the image patches are located. Roughly, one can think about this gray outline as the blurred representation of the covered image formed by responses of the low-level visual features. After a subject clicks on a certain patch, either a black or a white patch appears at this location depending on the image. At the same time, the presentation script checks whether the uncovered information is enough to classify the image unambiguously. If it is true, the final presentation screen of the current image is shown where the fully uncovered image is presented. At the same screen, a subject is informed that the image is identified and she can proceed to the next image. Otherwise, a subject sees the partially uncovered image and she should uncover the next patch. Screens of the clicking presentation for the digit “3” can be seen in the Appendix A.3.

Experimental stimuli were presented on a 19 inch monitor with the resolution 1280×960 pixels. The distance between the monitor and subject eyes were approximately 65 – 75 cm, which corresponds to the usual working conditions with a computer. The diameter of an image patch on the screen was about 7 cm. Under such conditions, the diameter of the visual field that falls on the fovea, the retinal region responsible for sharp vision, is about 2 cm. Thus, our patches fall also on the parafoveal region, which surrounds the fovea and correspond approximately to the field with the 10 cm diameter. Ideally, one had to display image patches whose size fits to the foveal region, so that when a human directs attention and uncovers a patch, the visual information about this patch can be processed with high resolution. However, we used larger patches to ease the clicking task. Moreover, in contrast to the periphery, the parafoveal region has still a good resolution which should be enough to unambiguously identify whether a patch is white or black after a first glance.

5.5.3 Participants

There were 15 participants that took part in the experiment, 11 males and 4 females between 26 and 33 years old. Their average age is 28. All of them have normal or corrected-to-normal vision. The participants were taken from PhD students and postdoctoral fellows of the Max-Planck Institute for Mathematics in the Sciences. They did not receive any reward for their participation but were willing to help. The subjects were told about the goal of the experiment in general, i. e. that we would like to observe what kind of strategy they use while selecting parts of images, which in their opinion can be useful for classification. However, they were aware neither of three certain information-theoretical strategies, which were tested, nor of our attempt to prove that people act adaptively. Moreover, not all participants have backgrounds in information theory or machine learning. Therefore, one can say that the subjects were naïve about the experiment goal.

5.6 Results

First, we want to demonstrate the fact that the adaptive strategy leads to shorter patch sequences on the clock digit data set. The Table 5.1 provides the information on the necessary number of patches to classify every clock digit as well as its mean value \bar{K} , if the patches were selected according to each of the considered strategies. Therefore, our hypothesis is that as subjects are instructed to select the shortest possible patch sequences, evidence for the adaptive strategy should be the strongest compared to the static MI and CMI.

In addition, Figure 5.3 provides examples of sequences for digits “1” and “5” that are selected according to MI, CMI and AMI. For the AMI sequences (see the subplot C), note that the choice of the second patch is adaptive and depends on whether the first patch is white or black.

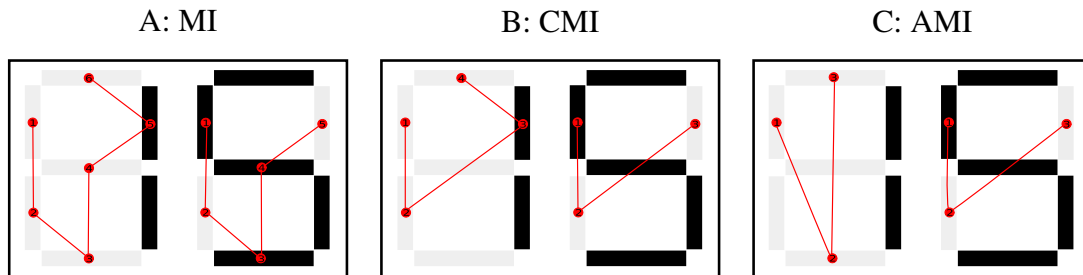


Figure 5.3: Illustration of patch sequences selected according to MI (subplot A), CMI (subplot B) and AMI (subplot C) strategies for the digits “1” and “5”.

Table 5.1: An average number of clicks K sufficient for classifying digits following one of the strategies: MI, CMI or AMI

digit	K_{MI}	K_{CMI}	K_{AMI}
“0”	6	4	3
“1”	2	2	3
“2”	3	4	3
“3”	3	5	4
“4”	5	3	3
“5”	5	3	4
“6”	6	5	3
“7”	5	4	4
“8”	4	4	4
“9”	4	5	3
\bar{K}	4.3	3.9	3.4

5.6.1 Subject statistics

For every subject, we define the average log-likelihood of every strategy s per image:

$$\log p(\mathcal{F}_\xi | s, \beta) = \frac{1}{T} \log p(\mathcal{F}_{\xi_1}, \dots, \mathcal{F}_{\xi_T} | s, \beta) = \frac{1}{T} \sum_{j=1}^T \sum_{i=1}^{K_j} \log \left[\frac{e^{\beta I_s(F_{\alpha_i})}}{\sum_q e^{\beta I_s(F_q)}} \right], \quad (5.22)$$

where T is a number of the analyzed images. Averaging is introduced in order to have the possibility to compare more naturally the results for a certain image with the results for the whole experiment to assess the stability of subject behavior.

Further, to analyze the strategies with respect to their explanatory power, the true values of β_{AMI}^* , β_{CMI}^* and β_{MI}^* for every subject are found by solving the maximum likelihood problem for the expression (5.22). Finally, using the values of β^* , we calculate log-ratios of the strategy maximum likelihood to the base likelihood for every subject. The corresponding information is given in the Table 5.2.

First of all, one can see that values of β^* and the log-ratios are highly correlated. To demonstrate this formally, we calculate their correlation coefficients: $\rho(\beta_{MI}^*, r_{MI}) = 0.99$, $\rho(\beta_{CMI}^*, r_{CMI}) = 0.99$, $\rho(\beta_{AMI}^*, r_{AMI}) = 0.97$. The reason for this is the following. If the behavior of a subject can be explained by a strategy s with the high value of β^* , then on every iteration one of the most informative patches is chosen, which leads to the high likelihood of the patch sequence with respect to this strategy. Low values of β^* make the likelihood tend to the base likelihood. As a result, these concepts provide similar information.

Table 5.2: An average sequence length, β^* , log-ratios of the strategies likelihood to the base likelihood for all subjects, entropy of the first two patches, clusters to which they belong and the correlation coefficients $\rho(\beta^*, \bar{K})$ and $\rho(r, \bar{K})$.

subj.	\bar{K}	β_{MI}^*	r_{MI}	β_{CMI}^*	r_{CMI}	β_{AMI}^*	r_{AMI}	$H(F_{\alpha_1}, F_{\alpha_2})$	cluster
s_1	4.7	0.59	0.01	0.32	0	0.32	0.02	3.73	none
s_2	4.1	0	0	0	0	0.96	0.16	1.98	AMI
s_3	4.5	1.55	0.15	1.91	0.23	1.35	0.33	1.53	AMI mix.
s_4	3.8	3.55	0.49	2.88	0.43	3.46	1.58	0.92	AMI mix.
s_5	4.4	0	0	0.48	0.01	1.18	0.25	2.69	AMI
s_6	4.8	0	0	0.78	0.04	0.28	0.02	2.56	none
s_7	4.4	0	0	0.19	0	0.83	0.14	2.68	AMI
s_8	4	4.07	0.6	5.75	1.19	2.69	1	1.67	CMI mix.
s_9	4.4	0	0	0	0	1.1	0.21	3.56	AMI
s_{10}	4	0.89	0.05	0.89	0.05	2.04	0.54	0.56	AMI
s_{11}	4.3	0.29	0.01	1.17	0.07	0.55	0.06	1.08	none
s_{12}	4	3.09	0.39	4.36	0.85	1.66	0.49	2.4	CMI mix.
s_{13}	4.2	2.51	0.31	3.31	0.67	1.91	0.62	0	CMI mix.
s_{14}	4.1	2.69	0.38	2.51	0.36	1.45	0.38	1.35	CMI mix.
s_{15}	4.5	0.59	0.03	0.45	0.01	0.03	0	1.18	none
$\rho(\cdot, \bar{K})$		-0.67	-0.69	-0.58	-0.56	-0.82	-0.77		

Inspecting the Table 5.2, one can notice the clear evidence against MI and CMI for some subjects, which is indicated by $\beta_{MI}^* = 0$ and $\beta_{CMI}^* = 0$ (see for example s_2 and s_5). At the same time, there are subjects with the relatively large values of β^* and the log-ratio r for the considered information-theoretical strategies, which suggests that their behavior can be explained well by the corresponding strategies (see examples for AMI that marked bold, e. g. s_3, s_4, s_8). We consider $\beta^* > 1$ to be a good indicator for a strategy s as it means that the informativeness of a patch $I_s(F)$ in the softmax function (5.15) has the weight > 1 . Although there are subjects that show MI- and CMI-compatible behavior, only for 4 out of 15 subjects we could not observe evidence for the adaptive strategy (see s_1, s_6, s_{11} and s_{15} with very low β_{AMI}^* and r_{AMI}). Thus, we can conclude that most of the people utilize AMI while selecting image patches that can be useful for classification.

As the Table 5.1 suggests, we expect that subjects who follow the adaptive strategy should make fewer clicks compared to those that follow MI or CMI. The correlation coefficients $\rho(\beta_{AMI}^*, \bar{K})$ and $\rho(r_{AMI}, \bar{K})$, which are given in the Table 5.2, reveal the expected regularity, i. e. the subjects with high values of AMI log-ratio and β_{AMI}^* produce on average shorter patch sequences, e. g. s_4, s_8, s_{10}, s_{12} (marked italic). To illustrate this fact graphically, Figure 5.4 provides two subplots A and B that show β_{AMI}^* and AMI log-ratio plotted against the average number of clicks, i. e. the average patch sequence. At the same time, the dependence of \bar{K} on β^* is weaker for other information-theoretical strategies (see subplots C and D on the same figure), which means that it is more likely to produce short patch sequences without following MI and CMI compared to the adaptive strategy. An interesting case is the subject s_2 who produced quite short sequences, however, there is no strong evidence for any of the considered strategies. This fact suggests that s_2 used another efficient but unknown for us strategy to achieve such result.

5.6.2 Subject clusters

Depending on strategies with large values of log-ratios, we defined 4 clusters of the subjects: “AMI”, “AMI mixture”, “CMI mixture” and “none” (see Table 5.2 for the assignment¹). Members of the “AMI” cluster show evidence almost only for the AMI strategy, subjects belonging to the “AMI mixture” and “CMI mixture” clusters follow all considered here strategies but the evidence for AMI or CMI is larger, respectively. Finally, the “none” cluster, as the name suggests, contains subjects whose behavior does not provide evidence for any of three strategies. To illustrate the distinction between subjects from the different clusters, we pick the representatives for each cluster and plot their average

¹ Although s_{14} should formally belong to “MI mixture” cluster, this subject was put in “CMI mixture” as the difference between the corresponding β^* and r is minimal and the cluster “MI mixture” would contain only one subject.

log-likelihood of AMI, CMI and MI against $\log_{10} \beta$ (see Figure 5.5). The log-likelihood of all strategies for $\log \beta = -2$ corresponds approximately to the base log-likelihood.

All subjects within the AMI cluster except s_{10} produced rather long patch sequences, which at first glance contradicts with the idea of efficiency of the adaptive behavior. Although the values of β^* are moderate, the values of the log-ratio are rather low. This comes from the fact that there are many images where the subjects do not follow any of the considered strategies. Thus, if one fits β^* for every image, its value is not stable and AMI is only on average better than CMI and MI.

Subjects belonging to the cluster “none” on average do not show evidence for the considered strategies. Note that their patch sequences are also long. Individual examination of every image for these subjects shows that a strategy with the maximum log-ratio changes from image to image. This indicates instability with respect to the strategies of our interest. To demonstrate this, we provide several subplots on Figure 5.6 where the log-likelihood of MI, CMI and AMI is plotted against β for several individual images. There is another peculiarity of this cluster: these subjects are the only who use particular patch sequences, which are rather long, repeatedly. On the contrary, in case of failure in producing a short patch sequence, subjects from other clusters tend to change their behavior.

For both mixture clusters, i. e. “AMI mixture” and “CMI mixture”, there are two reasons for observing enough evidence for all strategies. First, there are cases when for the same subject some patch sequences can be better explained by one strategy and other sequences by another strategy. However, there is also a second reason: the subjects utilize different strategies for first and later clicks.

It was noticed that most subjects use several alternatives of the fixed start consisting of 2 patches, which were learned during the presentation. The Table 5.2 provides values of the Shannon entropy of the first two uncovered patches estimated for every subject. Note that the order of patches in the starting pair was not taken into account, i. e. the pairs F_1, F_3 and F_3, F_1 are considered to be identical. The value of the highest possible entropy corresponding to the uniform distribution is 4.39. To get the intuition for other values, consider two following examples. The subject s_{15} , who used 24 out of 30 times the same starting pair, has $H(F_{\alpha_1}, F_{\alpha_2}) = 1.18$, while s_6 with $H(F_{\alpha_1}, F_{\alpha_2}) = 2.56$ has 3 preferred pairs that were used 9, 7 and 6 out of 30 times, respectively.

The reason for using the fixed start is the following. There are images which can be classified using only 1 or 2 patches. However, this particular patch or patch pair may be ineffective for classifying the rest of the images. The extreme example is the lower right patch of the clock digit. This patch is the worst to start with according to its mutual information with the class variable. Knowing its value, the initial number of classes can be reduced to 9 or 1. Thus, it is useful only in 1 out of 10 cases for the digit “2”. Nevertheless,

a lot of subjects use this patch at the beginning after successfully classifying the clock digit “2” only with one click. Therefore, it seems that a bias towards a possibility to classify an image uncovering the minimal number of patches is strong, even though this possibility may be not very likely.

Interestingly, if the first clicks are assumed to be fixed, we can observe that the patches uncovered afterwards are often chosen according to the adaptive strategy. The illustration of this phenomenon is given on Figure 5.7 that plots the log-likelihood of MI, CMI and AMI against $\log_{10} \beta$ for the full patch sequences and the sequences without considering first two clicks. Note that though the likelihood of these first clicks is not considered (the product over patches in (5.16) starts from $i = 3$), they are still used for conditioning in the selection criteria of *CMI* and *AMI*.

The intuitive explanation for adaptivity in the rather later clicks is the computational complexity of AMI. It requires the reestimation of its selection criterion for all candidates for selection after observing a value of newly uncovered patch. But on the later iterations, a number of possible classes as well as a number of the patch-candidates is smaller, thus, the selection problem becomes easier. Thus, the observed behavior can be regarded as the simplified version of the adaptive selection strategy.

5.7 Conclusions

There is experimental evidence that human saccades during visual search preferentially target locations that contain task-relevant information. Here, we suggested that information-theoretical feature selection criteria can be the underlying strategies of the task-dependent attentional selection. In particular, we considered two static selection criteria based on mutual information and conditional mutual information, as well as the criterion based on the adaptive conditional mutual information, which was proposed earlier in this thesis. Both static strategies assume that only task-specific prior influence the probability of the image location to be selected, whereas the adaptive selection criterion takes also into account the information integrated over previously visited locations.

In order to test these strategies, we performed a psychophysical experiment where subjects had to click and as a result uncover image patches, which in their opinion are relevant for classification. First of all, our clicking experiment provides evidence that for a complex visual classification task with 10 classes people are able to employ quite complicated entropy-based search strategies. In addition, we found that, even though it is more computationally demanding, most people act adaptively, i. e. take into account image-specific information. The experimental results revealed also that many subjects tend to use the so-called hybrid strategy while selecting the informative image patches. They start with

the patches which can immediately classify some images and if it does not happen they proceed selecting in the adaptive way. First, such behavior demonstrates a bias towards the possibility to classify an image just with 1 – 2 clicks, even if this is quite unlikely. Second, by applying the adaptive selection strategy on the late iterations, people avoid the high computational complexity of AMI at the beginning when a number of possible classes to which an image can belong is large.

In the future, it would be interesting to combine the considered information-theoretical strategies, which represent the top-down component of the attentional selection, with the feedforward bottom-up attentional mechanism based on low-level saliency. The eye tracking setup is then the natural tool to test such extended models. Another possible improvement concerns visual stimuli. On the one hand, one could draw better conclusions from the experiment if the stimuli set was constructed in the way that patch sequences generated by the tested strategies had a minimal overlap. On the other hand, it would be good to use natural scenes to make the experimental conditions maximally close to people's every day visual experience. Moreover, the recent trend in eye movement research is to use dynamical scenes as our natural environment is not static [Dorr et al., 2010; Tatler et al., 2011].

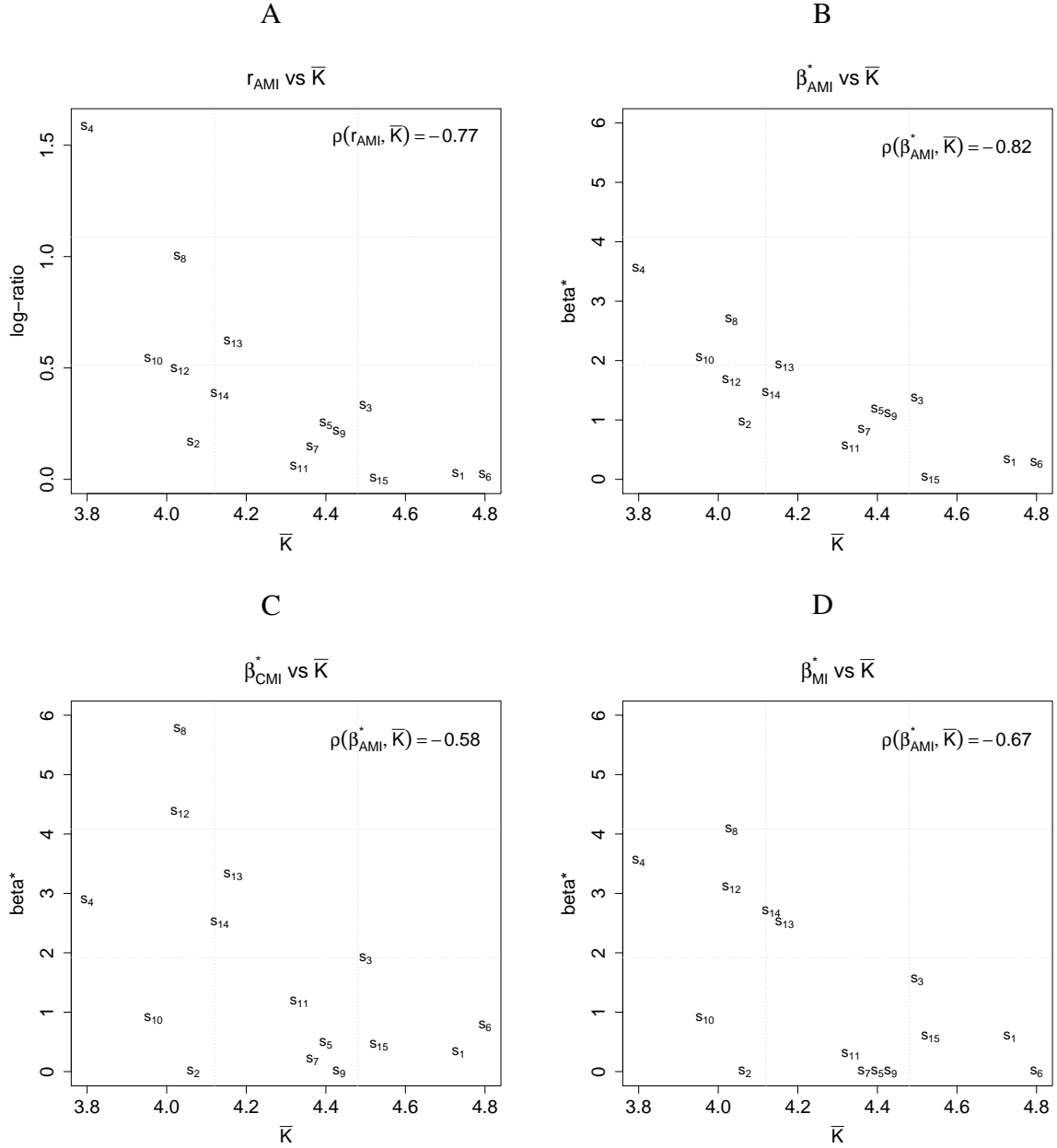


Figure 5.4: A: Log-ratio of AMI maximum likelihood to the base likelihood r_{AMI} of every subject is plotted against their average number of clicks. B, C, D: β_{AMI}^* , β_{CMI}^* and β_{MI}^* fitted for every subject is plotted against their average number of clicks, respectively. Correlation coefficients ρ between the considered concepts and \bar{K} are given on every subplot.

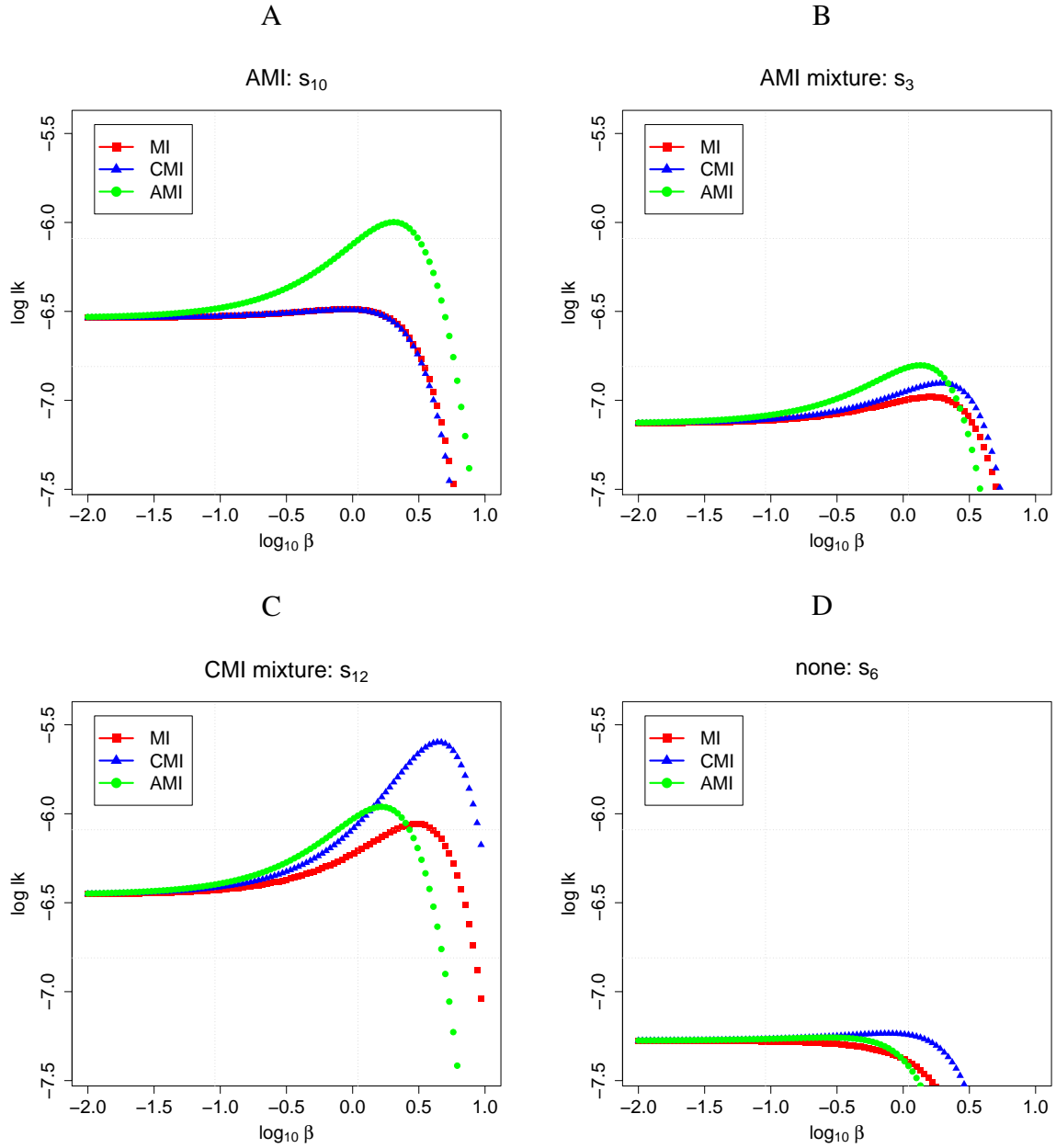


Figure 5.5: The average log-likelihood of MI, CMI and AMI is plotted against the $\log_{10} \beta$ for several subjects with typical behavior. A: There is strong evidence for AMI only. B, C: There is evidence for all three strategies but the evidence for AMI and CMI is stronger, respectively. D: There is no evidence for any of the considered strategies.

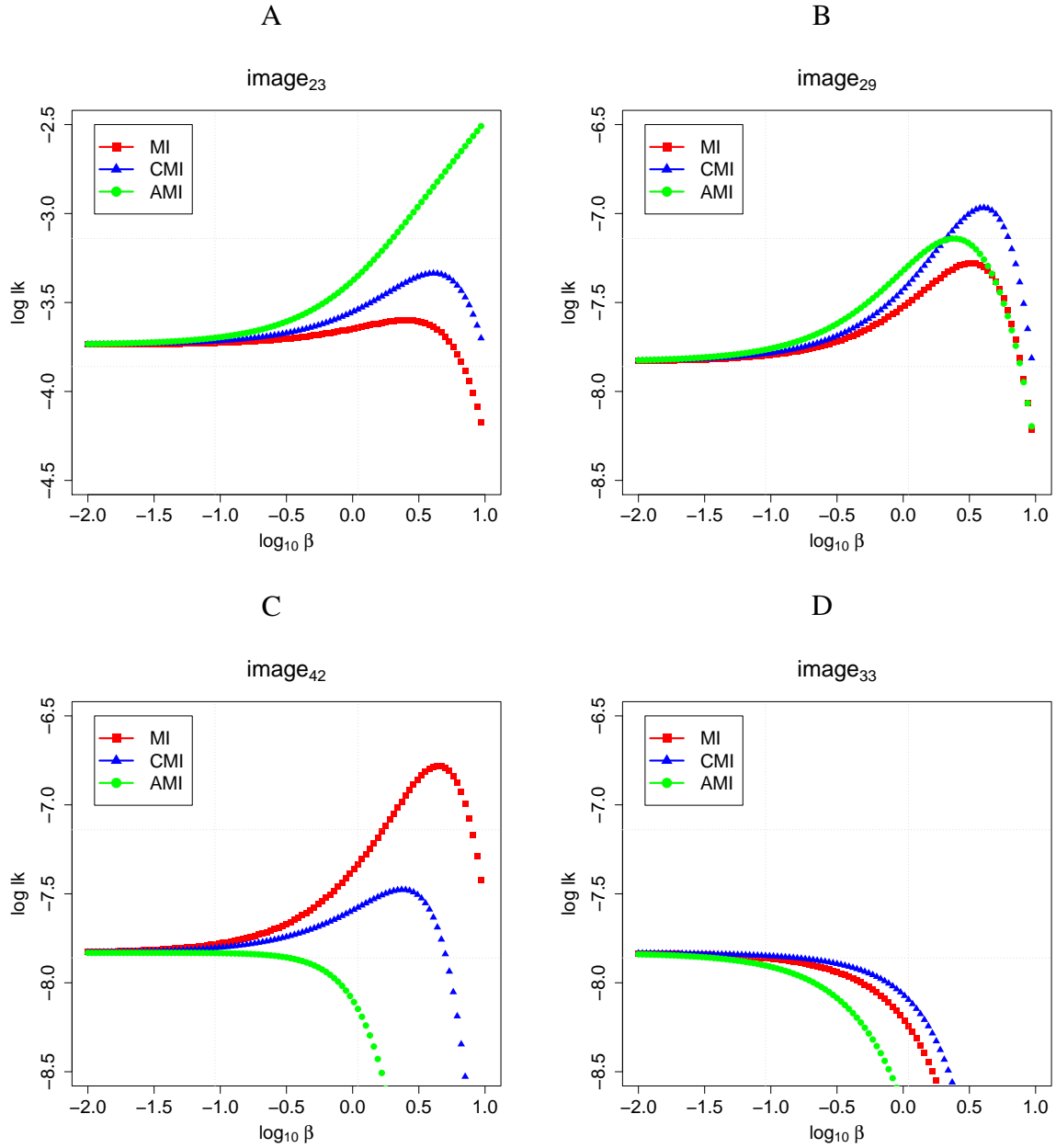


Figure 5.6: The log-likelihood of MI, CMI and AMI for a certain image is plotted against the $\log_{10} \beta$ for a subject s_{11} from the cluster “none”. A: There is strong evidence for AMI. B: There is evidence for all three strategies. C: There is evidence for MI and AMI is extremely unlikely. D: There is no evidence for any of the considered strategies.

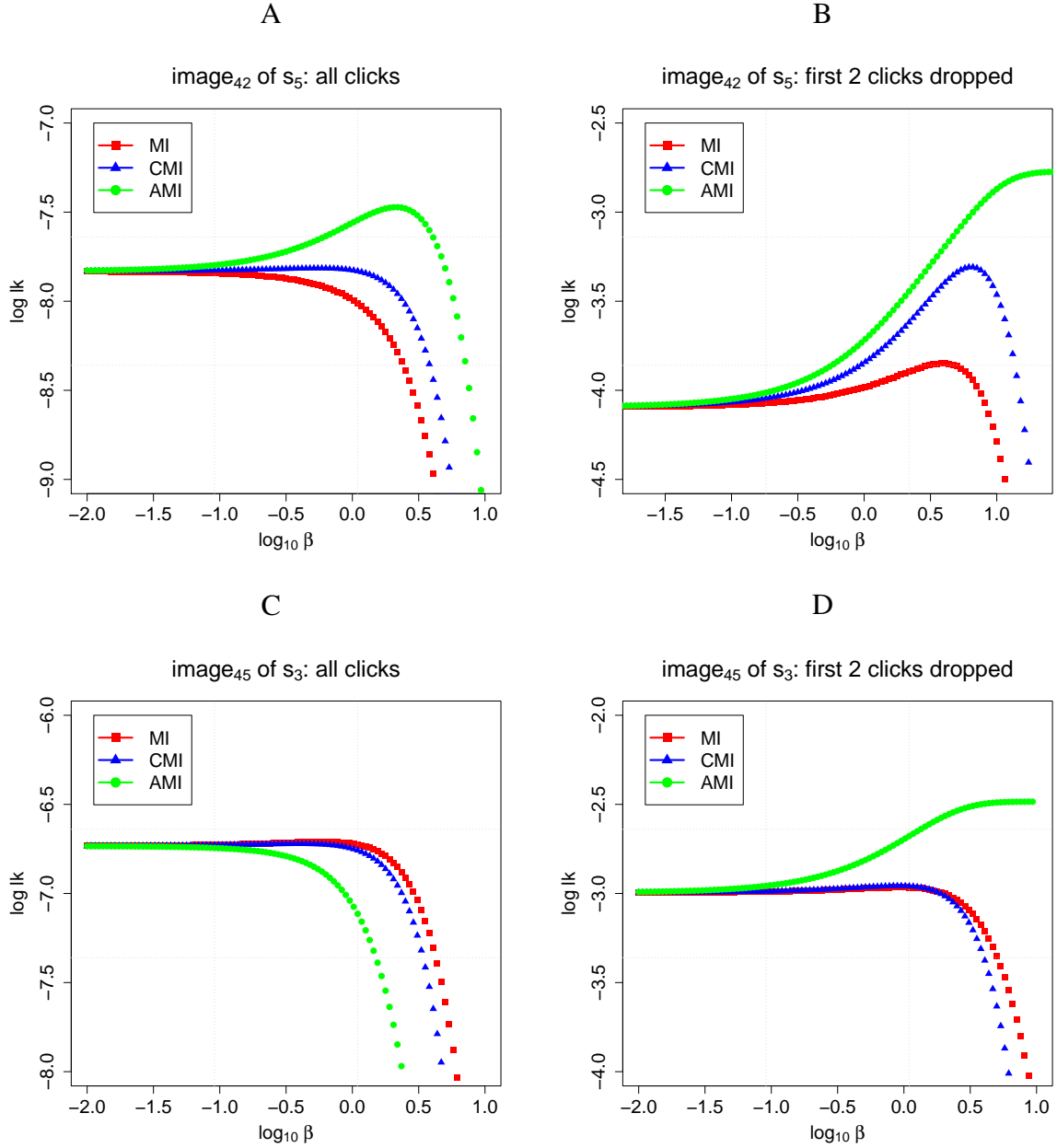


Figure 5.7: Log-likelihood of MI, CMI and AMI is plotted against the $\log_{10} \beta$ for a certain image if the strategies were followed from the beginning (subplots A and C for subjects s_5 and s_3 , respectively) and if the first two clicks are fixed (subplots B and D for subjects s_5 and s_3 , respectively).

Chapter 6

Discussion

Nature served as inspiration already for early machine learning algorithms. A good example is artificial neural networks that utilize structural and functional principles of neural circuits in the brain [McCulloch & Pitts, 1943; Rosenblatt, 1958]. The algorithm proposed in this thesis is not an exception. Our inspiration comes from the visual system and in particular from the numerous connections going backwards along the ventral or “what” pathway, which is responsible for processing complex object characteristics. Though the role of top-down connections is still a subject of debates, it is believed that the feedback from the higher to lower visual areas enhances processing of object characteristics that are useful for unambiguous identification of this object.

In analogy to such task-dependent attentional enhancement of relevant aspects of the visual input, we have proposed an adaptive sequential feature selection algorithm. The concept of adaptivity is equivalent to the presence of feedback information flow in the system. Systems having only feedforward circuits are “static” in our terminology.

The main idea of the algorithm is that it is possible to select a smaller number of features, which are useful for classification of a certain object, if the selection process is influenced by this object itself. As a result, the feature selection should be performed directly during classification. Features are picked sequentially and once a feature is selected and its value is evaluated, current hypotheses about a class of the recognized object are refined. If it is not possible to assign the object to a single class, which would mean its unambiguous classification, the next feature will be selected as the one minimizing the current uncertainty about the object class.

As described in the review presented in Chapter 2, there are plenty of selection criteria that fit to the framework of iterative uncertainty reduction. However, in our algorithm, we used Shannon entropy, which is a natural measure of uncertainty from information theory. Then, minimizing the uncertainty corresponds to maximizing the mutual infor-

mation of a class and a feature-candidate conditioned on previously selected features. In addition to the information-theoretical soundness of the concept of mutual information, features that are selected according to such a criterion are proved to be useful for classification. Though mutual information is an attractive selection criterion in theory, its estimation is still considered to be a difficult task in spite of a rich number of related studies. We solved the problem of estimating the adaptive conditional mutual information using a non-parametric kernel density estimator. Unreliable pdf estimates in the high-dimensional space were improved by the adaptive smoothing developed here. Undoubtedly, there is room for improvement concerning quality of the estimates. Nevertheless, the adaptive feature selection algorithm using the proposed estimation technique was successfully applied to problems where the number of features was much larger than the number of training samples. As a result, we demonstrated some advantages of the adaptive approach over its static counterpart.

In particular, experimental investigations of ACMIFS, which are described in Chapter 4, showed that the adaptive feature selection is advantageous when a classification task is difficult, for example, in case of an insufficient amount of trained data for good generalization. This result is in line with neurophysiological observations in the visual system. A recognition of simple objects can be performed in a purely feedforward fashion, i. e. using a static algorithm. Attentional feedback is first initiated when the information delivered by the first feedforward sweep does not suffice for unambiguous classification, i. e. when a classification is not trivial. Moreover, this is also supported by results of our psychophysical experiment, where people had to select image patches that as they think are relevant for classification of these images. We found that many people become adaptive first on later iterations when few patches selected according to some static strategy are not enough to classify an image.

Taking together such heuristic and the problem of computational complexity of the proposed ACMIFS, a hybrid selection scheme seems to be a natural candidate for a simplified adaptive feature selection algorithm. This would mean that instead of selecting only the first feature prior to observing a testing sample as the most informative for the current problem, one could use several statically preselected features. On the one hand, this sounds like a clever compromise that helps to reduce computational costs of the adaptive selection process but still leaves the possibility to make use of image-specific information in order to enhance the quality of classification. On the other hand, the experimental investigations of ACMIFS showed that one can get much better results if the first features are selected adaptively. The last fact inspired us to suggest a hybrid scheme that starts selecting features according to ACMIFS and then switches to ATM. ATM is also an adaptive algorithm that however assumes features being class-conditionally independent, thus, its estimation is computationally cheaper. Whereas the latter hybrid scheme will produce a smaller subset of informative features, the former is definitely computationally cheaper.

The results of our psychophysical clicking experiment provided evidence that ACMIFS proposed here, despite its complexity, can explain the behavior of many people while they perform a visual classification task. This fact indicates that we are on the right way to understand the concept of task-relevance utilized by the brain. However, the next interesting question appears: how the exactly estimation of the adaptive selection criterion is implemented in the visual system. There are many neural models that propose a framework for interactions of bottom-up and top-down process, e. g. [Raizada & Grossberg, 2003; Lee & Mumford, 2003], however, the process of estimating the next target for attentional modulation is not well-studied.

A detailed neural implementation of the proposed adaptive feature selection algorithm would be a good continuation of the current work. Here, we would like to describe just a sketch of it in order to show that such scheme can be implemented in the brain and therefore it can serve as the underlying strategy of task-dependent attentional selection.

Thus, the neural implementation could be based on the idea of self-organizing maps (SOM) [Kohonen, 1982], an unsupervised technique that is trained to produce a low-dimensional representation of the input, which is called a map. This rather early computational algorithm is closely related to models of cortical maps in the visual cortex [Olshausen & Field, 1996; Obermayer & Sejnowski, 2001]. As a result of self-organization, neurons within one map that have similar characteristics form clusters. Short-range excitatory connections inside one cluster make all neurons fire simultaneously, whereas long-range inhibitory connections exist between the clusters and are involved in their competition for representation [Sirosh & Miikkulainen, 1994; Bosking et al., 1997]. One can think about such clusters as features extracted from the visual input. The complexity of these features is defined by the area where the map is located. Then, the role of the task-dependent attentional modulation is to bias the competition for representation within one layer towards task-relevant features. First, the feedback modulates very abstract and complex representations, i. e. building blocks of categories to which a recognized object can belong. This modulation is transmitted further to neurons on the lower levels that respond to features, which constitute the modulated high-level representations. As a result, the enhanced features are processed with higher acuity which is reflected in the updated activity of the corresponding maps. In turn, this update causes changes in the activity of neurons responding to possible object categories.

Within such a framework, implementation of the selection criterion requires estimation of class-conditional joint probabilities of the abstract features. In fact, SOM performs vector quantization, i. e. it models a probability density function by distribution of clusters, which are evolved on the map as a result of learning. Thus, activity on the map represents the joint pdfs of features conditioned on the input signal. If every neuron on the low-dimensional map has a Gaussian activation function, the obtained model is nothing but a Gaussian mixture. Note that the kernel density estimator with Gaussian kernels

used earlier is also a mixture of Gaussians. The only difference is that KDE uses Gaussian functions that have some predefined width and are centered around training samples, rather than learning these parameters. Therefore, analogously to KDE, one can learn separate maps to model joint probability functions of features given every class using for example an expectation-maximization algorithm [Heskes, 2001] or a Bayesian approach [Yin & Allinson, 2001]. Further, having the class-conditional pdfs of features, we still need to estimate the mutual information. Recall that the ACMIFS criterion can be rewritten using the definition of the Kullback-Leibler divergence weighted by class posteriors. A posterior of a class can be represented by activation of a neuron or a group of neurons that respond to this class in the higher visual areas. However, the estimation of Kullback-Leibler divergence between the class-conditional and marginal probability distributions is not trivial as it requires integration over the feature space. A standard approach to such problem is to sample from both distributions and then estimate the divergence between them. This approach is especially attractive because its biological plausibility was already suggested by various researches. It is hypothesized that neural activity can be viewed as a representation of samples from the posterior distribution [Hoyer & Hyvärinen, 2002; Fiser et al., 2010]. Thus, the brain is likely to implement the sampling mechanism. Although a full neuronal model has to be worked out in detail and experimentally tested and some proposed aspects are likely to be inaccurate, we believe that the brain implements some approximated version of ACMIFS in order to decide which features are really relevant to classify a new visual scene.

Despite the connection of the developed scheme to principles of visual processing, the major part of this work is dedicated to the development of the machine learning algorithm. Therefore, we would like to comment on its applicability to different classification tasks. On the one hand, the adaptive feature selection presented here is of the filter type and therefore can be used with any kind of classifier. On the other hand, as feature selection is performed online during classification, a classifier should be able to produce a fast prediction of the object class given a varying subset of features. For this purpose, one can use the probabilistic estimator of ACMIFS itself because the class posteriors are intrinsic components of the selection criterion. Another possible candidate is a representative of the so-called lazy learners, i. e. classifiers that do not require any learning like k -nn. Alternatively, one can again get inspiration from nature. The adaptive selection scheme functions within the hierarchical framework in the visual system. Thus, complex mapping of the visual input to classes is distributed through the hierarchy of feature layers of growing complexity, resulting in simple linear relationships between the levels. Then, parallel feedforward-feedback interactions in such architecture allows to perform fast probabilistic inference. Therefore, in order to solve complex classification tasks, we suggest to use ACMIFS for selecting features in systems with a hierarchical architecture. Moreover, we demonstrated efficiency of the adaptive feature selection in the hierarchical

system on the example of convolutional networks used for classifying handwritten digits from the MNIST database.

Appendix A

A.1 Visual cortex

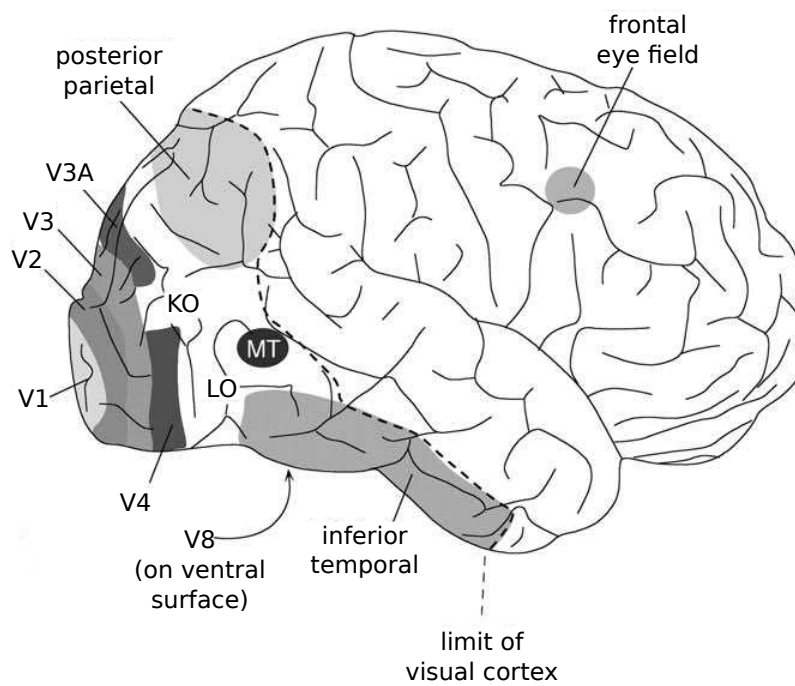


Figure A.1: Visual cortex, from [Rosa, 2002].

A.2 Artificial dataset

1	1	1	1
2	2	2	2
3	3	3	3
4	4	4	4
5	5	5	5
6	6	6	6
7	7	7	7
8	8	8	8
9	9	9	9
0	0	0	0

Figure A.2: Original images of the artificial dataset.

A.3 Example of the clicking presentation

You will be presented images of clock digits. Every image consists of 7 patches (parts) that are covered at the beginning. You can uncover a patch of the image by clicking on it. Your task is to uncover a minimal number of patches that, in your opinion, can help a computer to classify the image. You should not spend more than 3 seconds to decide which patch to uncover next. You can always stop the experiment by pressing 'Esc'. Thank you!



Start experiment

Figure A.3: Introduction screen.

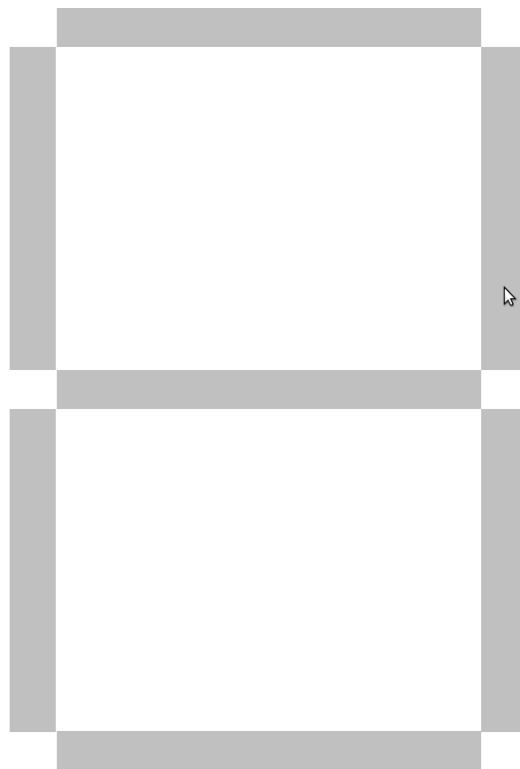


Figure A.4: First screen of the image presentation.

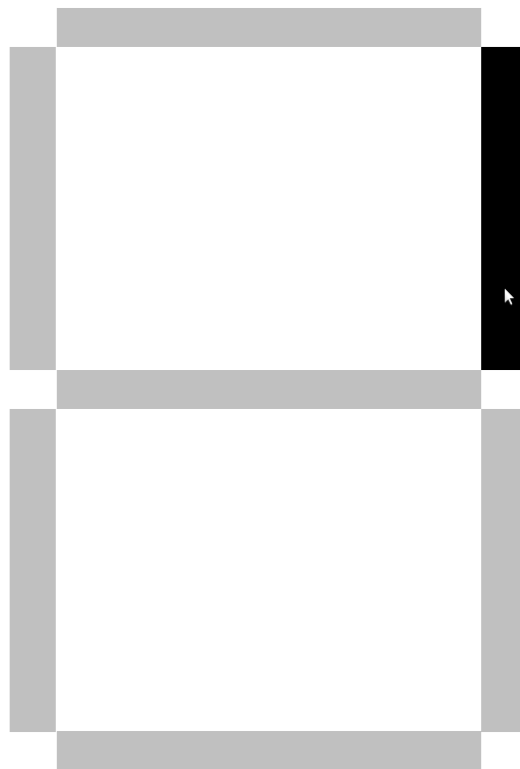


Figure A.5: Second screen of the image presentation.

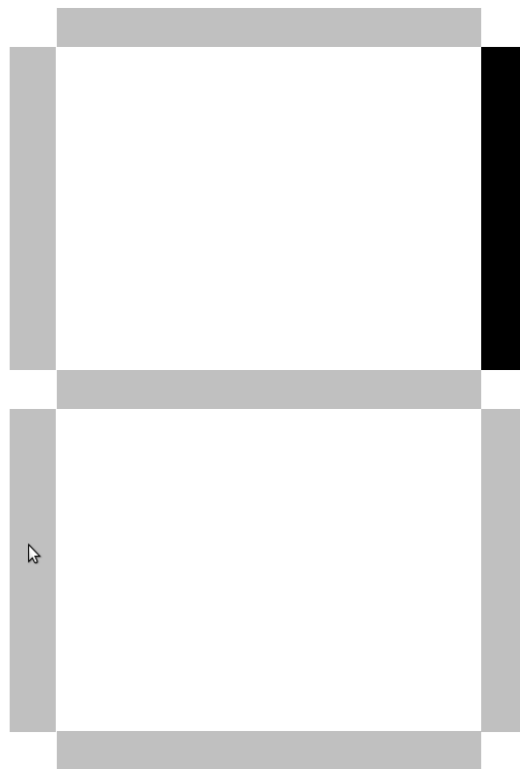


Figure A.6: Third screen of the image presentation.

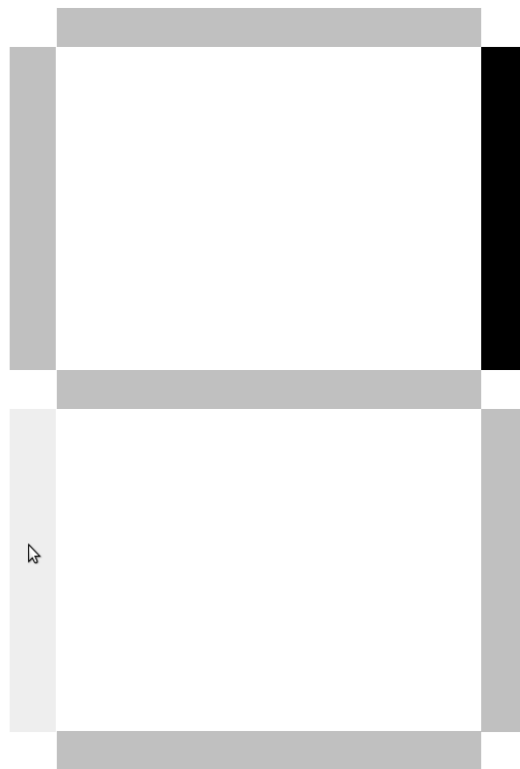


Figure A.7: Fourth screen of the image presentation.

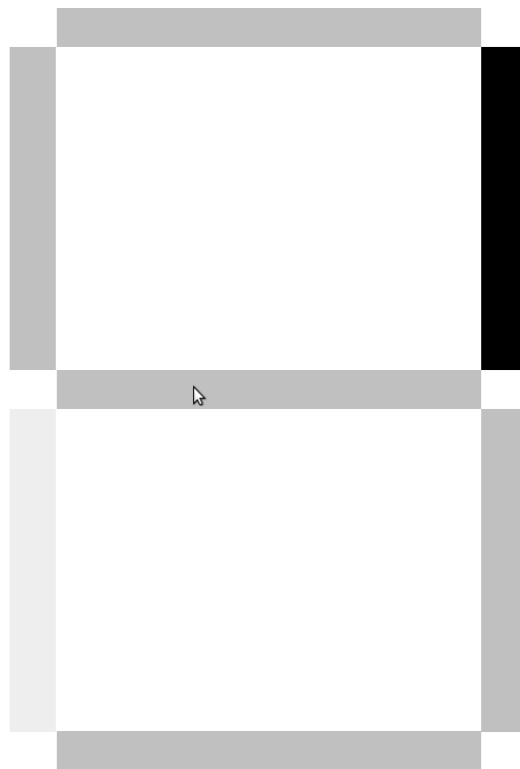


Figure A.8: Fifth screen of the image presentation.

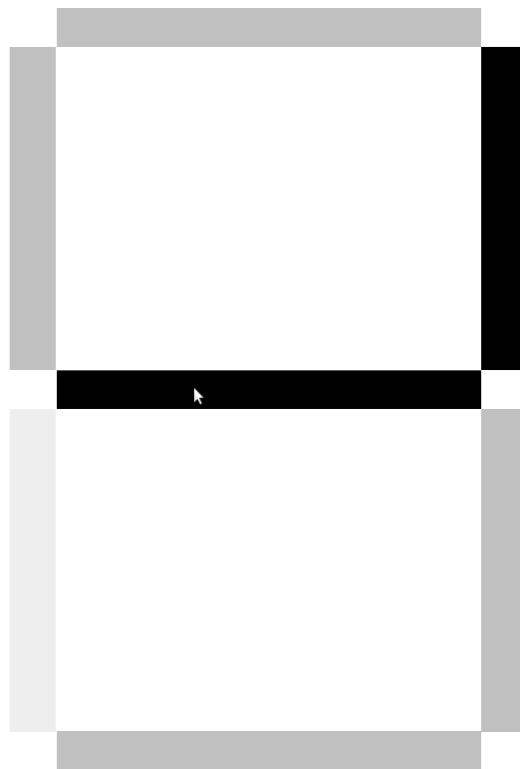


Figure A.9: Seventh screen of the image presentation.

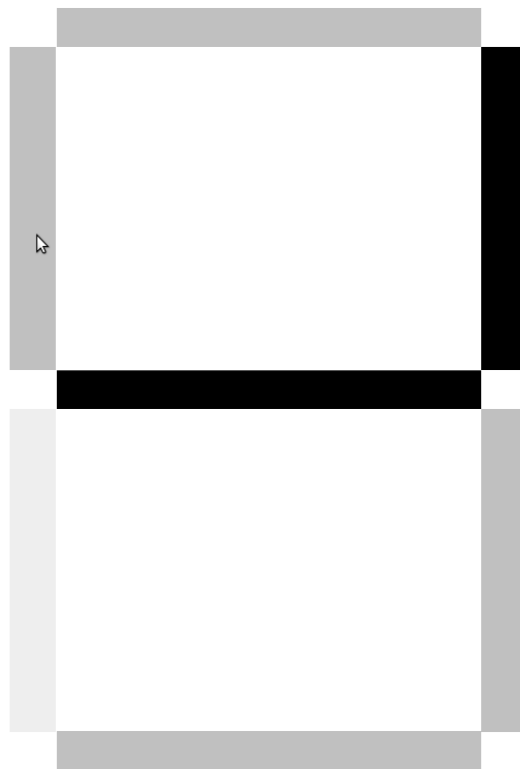


Figure A.10: Eighth screen of the image presentation.

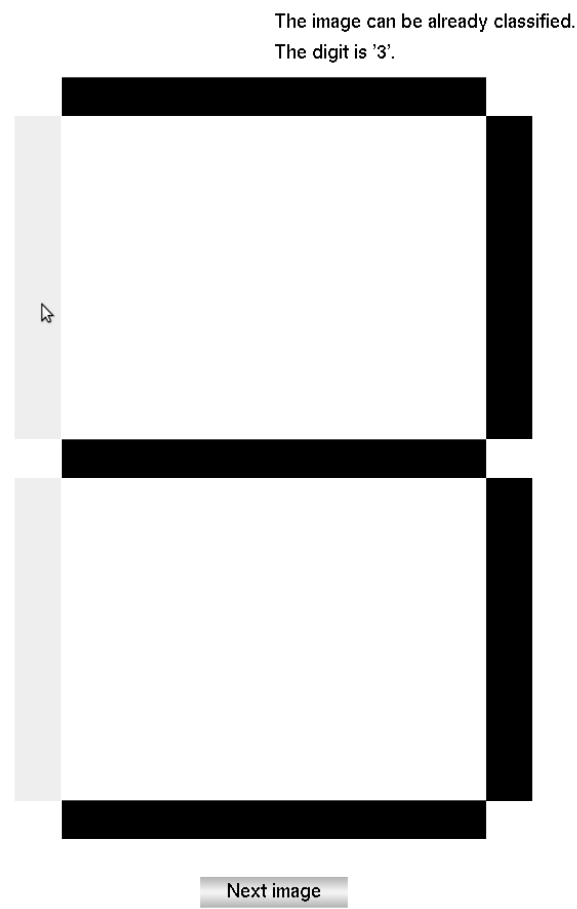


Figure A.11: Final screen of the image presentation.

Bibliography

- Abe, S. (2005). Modified backward feature selection by cross validation. In *Proceedings of the Thirteenth European Symposium on Artificial Neural Networks* (pp. 163–168). Bruges, Belgium.
- Aczél, Z. & Daróczy, Z. (1975). *On Measures of Information and Their Characterizations*. Academic Press.
- Aha, D. W. & Bankert, R. L. (1996). A comparative evaluation of sequential feature selection algorithms. In D. H. Fisher & H.-J. Lenz (Eds.), *Learning from Data: Artificial Intelligence and Statistics V*, Lecture Notes in Statistics chapter 4, (pp. 199–206). 175 Fifth Avenue. New York, New York 10010, USA: Springer-Verlag.
- Ahmad, I. & Lin, P.-E. (1976). A nonparametric estimation of the entropy for absolutely continuous distributions. *IEEE Transactions on Information Theory*, 22(3), 372–375.
- Ahmed, N. A. & Gokhale, D. V. (1989). Entropy expressions and their estimators for multivariate distributions. *IEEE Transactions on Information Theory*, 35(3), 688–692.
- Aitchison, J. & Aitken, C. G. G. (1976). Multivariate binary discrimination by the kernel method. *Biometrika*, 63, 413–420.
- Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6), 716–723.
- Amari, S., Cichocki, A., & Yang, H. H. (1996). A new learning algorithm for blind signal separation. In *Advances in Neural Information Processing Systems* (pp. 757–763).: MIT Press.
- Antos, A. & Kontoyiannis, I. (2001). Convergence properties of functional estimates for discrete distributions. *Random Structures and Algorithms*, 19(3–4), 163–193.
- Anzai, D. & Hara, S. (2010). An area layout-based MAP estimation for indoor target tracking. In *VTC Fall* (pp. 1–5).: IEEE.
- Avdiyenko, L., Bertschinger, N., & Jost, J. (2012a). Testing entropy-based search strategies for a visual classification task. *BMC Neuroscience*, 13(Suppl 1), 109.

- Avdiyenko, L., Bertschinger, N., & Jost, J. (2012b). Adaptive information-theoretical feature selection for pattern classification. In *Computational Intelligence: Revised and Selected Papers of the International Joint Conference, IJCCI 2012, Barcelona, Spain, 2012*. to appear.
- Avdiyenko, L., Bertschinger, N., & Jost, J. (2012c). Adaptive Sequential Feature Selection for Pattern Classification. In A. C. Rosa, A. D. Correia, K. Madani, J. Filipe, & J. Kacprzyk (Eds.), *IJCCI* (pp. 474–482).: SciTePress.
- Bar, M., Kassam, K. S., Ghuman, A. S., Boshyan, J., Schmid, A. M., Dale, A. M., Hämäläinen, M. S., Marinkovic, K., Schacter, D. L., Rosen, B. R., & Halgren, E. (2006). Top-down facilitation of visual recognition. *Proceedings of the National Academy of Sciences of the United States of America*, 103(2), 449–454.
- Batista, A. P., Buneo, C. A., Snyder, L. H., & Andersen, R. A. (1999). Reach plans in eye-centered coordinates. *Science*, 285(5425), 257–260.
- Battiti, R. (1994). Using mutual information for selecting feature in supervised neural net learning. *IEEE Transactions on Neural Networks*, 5(4), 537–550.
- Baum, L. E. (1972). An inequality and an associated maximization technique in statistical estimation of probabilistic functions of a Markov process. *Inequalities*, 3:1-8.
- Beirlant, J., Berlinet, A., & Györfi, L. (1999). On piecewise linear density estimators. *Statistica Neerlandica*, 53(3), 287—308.
- Beirlant, J., Dudewicz, E. J., Györfi, L., & van der Meulen, E. (1997). Nonparametric entropy estimation: An overview. *International Journal of the Mathematical Statistics Sciences*, 6, 17–39.
- Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., Turian, J., Warde-Farley, D., & Bengio, Y. (n.d.). Deep learning tutorials. Retrieved from: <http://deeplearning.net/tutorial/lenet.html>.
- Betz, T., Kietzmann, T. C., Wilming, N., & König, P. (2010). Investigating task-dependent top-down effects on overt visual attention. *Journal of Vision*, 10(3), 1–14.
- Bichot, N. P. & Schall, J. D. (1999). Effects of similarity and history on neural mechanisms of visual selection. *Nature Neuroscience*, 2(6), 549–554.
- Bonnlander, B. V. (1996). *Nonparametric Selection of Input Variables for Connectionist Learning*. PhD thesis, University of Colorado at Boulder.
- Bonnlander, B. V. & Weigend, A. S. (1994). Selecting input variables using mutual information and nonparametric density estimation. In *International Symposium on Artificial Neural Networks* (pp. 42–50). Taiwan.

- Bosking, W. H., Zhang, Y., Schofield, B., & Fitzpatrick, D. (1997). Orientation selectivity and the arrangement of horizontal connections in tree shrew striate cortex. *The Journal of Neuroscience*, 17(6), 2112—2127.
- Bowman, A. W. (1984). An alternative method of cross-validation for the smoothing of density estimates. *Biometrika*, 71, 353–360.
- Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). *Classification and Regression Trees*. Monterey, CA: Wadsworth and Brooks.
- Breiman, L., Meisel, W., & Purcell, E. (1977). Variable kernel estimates of multivariate densities and their calibration. *Technometrics*, 19, 135–144.
- Breitmeyer, B. G. & Tapia, E. (2011). Roles of contour and surface processing in microgenesis of object perception and visual consciousness. *Advances in Cognitive Psychology*, 7, 68–81.
- Broomhead, D. S. & Lowe, D. (1988). *Radial Basis Functions, Multi-variable Functional Interpolation and Adaptive Networks*. RSRE memorandum / Royal Signals and Radar Establishment. Royals Signals & Radar Establishment.
- Brown, G. (2009). A new perspective for information theoretic feature selection. In *Twelfth International Conference on Artificial Intelligence and Statistics*.
- Brown, G., Pocock, A., Zhao, M.-J., & Luján, M. (2012). Conditional likelihood maximisation: A unifying framework for information theoretic feature selection. *Journal of Machine Learning Research*, 13, 27–66.
- Bruzzzone, L. & Serpico, S. B. (1998). A simple upper bound to the Bayes error probability for feature selection. *Kybernetika*, 34(4), 387–392.
- Bullier, J. (2001). Feedback connections and conscious vision. *Trends in Cognitive Sciences*, 5(9), 369–370.
- Carlton, A. (1969). On the bias of information estimates. *Psychological Bulletin*, 71, 108–109.
- Carmona, P. L., Sotoca, J. M., Pla, F., Phoa, F. K. H., & Dias, J. B. (2011). Feature selection in regression tasks using conditional mutual information. In *Proceedings of the 5th Iberian Conference on Pattern Recognition and Image Analysis, IbPRIA'11* (pp. 224–231). Berlin, Heidelberg: Springer-Verlag.
- Caruso, F. & Tsallis, C. (2008). Nonadditive entropy reconciles the area law in quantum systems with classical thermodynamics. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 78(021102).

- Cellucci, C. J., Albano, A. M., & Rapp, P. E. (2005). Statistical validation of mutual information calculations: Comparison of alternative numerical algorithms. *Physical Review E*, 71(066208).
- Chelazzi, L., Duncan, J., Miller, E. K., & R., D. (1998). Responses of neurons in inferior temporal cortex during memory-guided visual search. *Journal of Neurophysiology*, 80, 2918–2940.
- Chelazzi, L., Miller, E. K., Duncan, J., & Desimone, R. (1993). A neural basis for visual search in inferior temporal cortex. *Nature*, 363, 345—347.
- Chen, C. H. (1976). On information and distance measures, error bounds, and feature selection. *Information Sciences*, 10(2), 159–173.
- Chen, S. F. & Goodman, J. (1998). *An empirical study of smoothing techniques for language modeling*. Technical report, Computer Science Group, Harvard University, Cambridge, Massachusetts.
- Cheng, H., Qin, Z., Feng, C., Wang, Y., & Li, F. (2011). Conditional mutual information-based feature selection analyzing for synergy and redundancy. *Electronics and Telecommunications Research Institute Journal*, 33, 2.
- Chun, M. M., Golomb, J. D., & Turk-Browne, N. B. (2011). A taxonomy of external and internal attention. *Annual Review of Psychology*, 62(1), 73–101. PMID: 19575619.
- Chun, M. M. & Wolfe, J. M. (2001). Visual Attention. In E. B. Goldstein (Ed.), *Blackwell's Handbook of Perception* chapter 9, (pp. 272–310). Blackwell.
- Cover, T. M. & Thomas, J. A. (1991). *Elements of Information Theory*. New York, NY, USA: Wiley-Interscience. pp. 12–49.
- Darbellay, G. A. & Vajda, I. (1999). Estimation of the information by an adaptive partitioning of the observation space. *IEEE Transactions on Information Theory*, 45(4), 1315–1321.
- Daub, C., Steuer, R., Selbig, J., & Kloska, S. (2004). Estimating mutual information using B-spline functions – an improved similarity measure for analysing gene expression data. *BMC Bioinformatics*, 5(1), 1–12.
- de Fockert, J. W. R. G., Frith, C. D., & Lavie, N. (2001). The role of working memory in visual selective attention. *Science*, 291(5509), 1803–1806.
- Dehaene, S., Changeux, J.-P., Naccache, L., Sackur, J., & Sergent, C. (2006). Conscious, preconscious, and subliminal processing: a testable taxonomy. *Trends in Cognitive Sciences*, 10, 204–211.

- Desimone, R. & Duncan, J. (1995). Neural mechanisms of selective visual attention. *Annual Review of Neuroscience*, 18(1), 193–222.
- Devijver, P. A. & Kittler, J. (1982). *Pattern Recognition: A Statistical Approach*. Prentice/Hall International.
- Devroye, L. P. & Wagner, T. J. (1977). The strong uniform consistency of nearest neighbor density estimates. *The Annals of Statistics*, 5(3), 536–540.
- Di Lollo, V., Enns, J. T., & Rensink, R. A. (2000). Competition for consciousness among visual events: The psychophysics of reentrant visual processes. *Journal of Experimental Psychology: General*, 129(4), 481–507.
- Dijksterhuis, A. & Aarts, H. (2010). Goals, Attention, and (Un)Consciousness. *Annual Review of Psychology*, 61(1), 467–490.
- Ding, C. H. Q. & Peng, H. (2005). Minimum redundancy feature selection from microarray gene expression data. *Journal of Bioinformatics and Computational Biology*, 3(2), 185–206.
- Dmitriev, Y. G. & Tarasenko, F. P. (1973). On the estimation functions of the probability density and its derivatives. *Theory of Probability and Its Applications*, 18, 628–633.
- Doquire, G. & Verleysen, M. (2012). A comparison of multivariate mutual information estimators for feature selection. In *Proceedings of the 1st International Conference on Pattern Recognition Applications and Methods, ICPRAM (1)'12* (pp. 176–185).
- Dorr, M., Martinetz, T., Gegenfurtner, K. R., & Barth, E. (2010). Variability of eye movements when viewing dynamic natural scenes. *Journal of Vision*, 10(10).
- Duch, W. (2006). Filter Methods. In I. Guyon, S. Gunn, M. Nikravesh, & L. Zadeh (Eds.), *Feature Extraction, Foundations and Applications* (pp. 89–118). Physica Verlag, Springer, Heidelberg.
- Duch, W., Wiecek, T., Biesiada, J., & Blachnik, M. (2004). Comparison of feature ranking methods based on information entropy. In *Proceedings of the IEEE International Joint Conference on Neural Networks* (pp. 1415–1419). Budapest, Hungary.
- Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern Classification and Scene Analysis*. New York: John Wiley and Sons.
- Dudewicz, E. J. & van der Meulen, E. (1981). Entropy-based tests of uniformity. *Journal of the American Statistical Association*, 76, 967–974.
- E., C. & C., L. (2003). Feature extraction based on the Bhattacharyya distance. *Pattern Recognition*, 36(8), 1703–1709.

- Efron, B. & Stein, C. (1981). The jackknife estimate of variance. *Annals of Statistics*, 9, 586—596.
- Eggermont, P. P. B. & LaRiccia, V. N. (1999). Best asymptotic normality of the kernel density entropy estimator for smooth densities. *IEEE Transactions on Information Theory*, 45(4), 1321–1326.
- Ehinger, K. A., Hidalgo-Sotelo, B., Torralba, A., & Oliva, A. (2009). Modelling search for people in 900 scenes: A combined source model of eye guidance. *Visual Cognition*, 17(6-7), 945–978.
- El Akadi, A., El Ouardighi, A., & Aboutajdine, D. (2008). A powerful feature selection approach based on mutual information. *International Journal of Computer Science and Network Security*, 8(4), 116–121.
- Elze, T. (2009). FlashDot - A platform independent experiment generator for visual psychophysics. *Journal of Vision*, 9(14), 58.
- Elze, T., Song, C., Stollhoff, R., & Jost, J. (2011). Chinese characters reveal impacts of prior experience on very early stages of perception. *BMC Neuroscience*, 12(1), 14.
- Enns, J. T. & Di Lollo, V. (2000). What's new in visual masking? *Trends in Cognitive Sciences*, 4(9), 345–352.
- Estevez, P., Tesmer, M., Perez, C., & Zurada, J. (2009). Normalized mutual information feature selection. *IEEE Transactions on Neural Networks*, 20(2), 189–201.
- Fano, R. (1961). *Transmission of Information: A Statistical Theory of Communications*. Cambridge, MA: The MIT Press.
- Fink, G. R., Dolan, R. J., Halligan, P. W., Marshall, J. C., & Frith, C. D. (1997). Space based and object based visual attention: shared and specific neural domains. *Brain*, 120, 2013–2028.
- Fiser, J., Berkers, P., Orban, G., & Lengye, M. (2010). Statistically optimal perception and learning: From behavior to neural representations. *Trends in Cognitive Sciences*, 14, 119–130.
- Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2), 179–188.
- Fix, E. & Hodges, J. L. (1951). *Discriminatory analysis, nonparametric discrimination: Consistency properties*. Technical Report 4, US Air Force School of Aviation Medicine, Randolph Field, TX.
- Fleuret, F. & Guyon, I. (2004). Fast binary feature selection with conditional mutual information. *Journal of Machine Learning Research*, 5, 1531–1555.

- Foldiak, P. (1990). Forming sparse representations by local anti-Hebbian learning. *Biological Cybernetics*, 64, 165–170.
- Fraser, A. M. & Swinney, H. L. (1986). Independent coordinates for strange attractors from mutual information. *Physical Review A*, 33(2), 1134–1140.
- Freedman, D. & Diaconis, P. (1981). On the histogram as a density estimator: L2 theory. *Probability Theory and Related Fields*, 57(4), 453–476.
- Fukunaga, K. (1990). *Introduction to Statistical Pattern Recognition*. Computer science and scientific computing. Elsevier Science.
- Fukunaga, K. & Hummels, D. M. (1987). Bayes error estimation using Parzen and k-NN procedures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5), 634–643.
- Furey, T. S., Duffy, N., Cristianini, N., Bednarski, D., Schummer, M., & Haussler, D. (2000). Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 16(10), 906–914.
- Gale, W. A. & Church, K. W. (1994). What's wrong with adding one. In *Corpus-Based Research into Language*. Rodolpi.
- Gauvain, J.-L. & Lee, C.-H. (1994). Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains. *IEEE Transactions on Speech and Audio Processing*, 2, 291–298.
- Gelfand, S. B., Ravishanker, C. S., & Delp, E. J. (1991). An iterative growing and pruning algorithm for classification tree design. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13, 163–174.
- Geman, D. & Jedynak, B. (1996). An active testing model for tracking roads in satellite images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(1), 1–14.
- Gilad-Bachrach, R., Navot, A., & Tishby, N. (2004). Margin based feature selection - theory and algorithms. In *Proceedings of the 21th International Conference on Machine Learning (ICML-04)* (pp. 43–50).: ACM Press.
- Gini, C. (1912). *Variabilità e Mutabilità (Variability and Mutability)*. C. Cuppini, Bologna.
- Grassberger, P. (1988). Finite sample corrections to entropy and dimension estimates. *Physics Letters A*, 128(6–7), 369–373.
- Grassberger, P. (2003). Entropy estimates from insufficient samplings. arXiv.org eprint physics/0307138.

- Grassberger, P. & Procaccia, I. (1983). Measuring the strangeness of strange attractors. *Physica D: Nonlinear Phenomena*, 9(1–2), 189–208.
- Grieve, K. L. & Sillito, A. M. (1995). Differential properties of cells in the feline primary visual cortex providing the corticofugal feedback to the lateral geniculate nucleus and visual claustrum. *Journal of Neuroscience*, 15(7 Pt 1), 4868–4874.
- Grossberg, S. (1976). Adaptive pattern classification and universal recoding: I. Parallel development and coding of neural feature detectors. *Biological Cybernetics*, 23(3), 121–134.
- Guo, B. & Nixon, M. S. (2009). Gait feature subset selection by mutual information. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, 39(1), 36–46.
- Guyon, I. & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3, 1157–1182.
- Guyon, I., Gunn, S., Nikravesh, M., & Zadeh, L., Eds. (2006). *Feature Extraction, Foundations and Applications*. Series Studies in Fuzziness and Soft Computing, Physica-Verlag, Springer.
- Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine Learning*, 46(1-3), 389–422.
- Guyon, I. M., Gunn, S. R., Ben-Hur, A., & Dror, G. (2004). Result analysis of the NIPS 2003 feature selection challenge. In *Advances in Neural Information Processing Systems*.
- Györfi, L. & van der Meulen, E. C. (1987). Density-free convergence properties of various estimators of entropy. *Computational Statistics & Data Analysis*, 5(4), 425–436.
- Hacine-Gharbi, A., Ravier, P., Harba, R., & Mohamadi, T. (2012). Low bias histogram-based estimation of mutual information for feature selection. *Pattern Recognition Letters*, 33(10), 1302–1308.
- Hall, M. A. (1999). *Correlation-base feature selection for machine learning*. PhD thesis, Department of Computer Science, University of Waikato, New Zealand.
- Hall, P. & Morton, S. C. (1993). On the estimation of entropy. *Annals of the Institute of Statistical Mathematics*, 45(1), 69–88.
- Hall, P., Sheather, S. J., Jones, M. C., & Marron, J. S. (1991). On optimal data-based bandwidth selection in kernel density estimation. *Biometrika*, 78(2), 263–269.
- Harris, B. (1975). *The statistical estimation of entropy in the non-parametric case*. University of Wisconsin-Madison, Mathematics Research Center.

- Hart, A. E. (1985). Experience in the use of an inductive system in knowledge engineering. In M. A. Bramer (Ed.), *Research and Development in Expert Systems*. Cambridge, UK: Cambridge University Press.
- Hayhoe, M. M., Bensinger, D. G., & Ballard, D. H. (1998). Task constraints in visual working memory. *Vision Research*, 38, 125–137.
- Hayhoe, M. M., Shrivastava, A., Mruczek, R., & Pelz, J. B. (2003). Visual memory and motor planning in a natural task. *Journal of Vision*, 3(1), 49–63.
- Haykin, S. (1999). *Neural Networks: A Comprehensive Foundation*. Prentice Hall.
- Hellman, M. & Raviv, J. (1970). Probability of error, equivocation, and the Chernoff bound. *IEEE Transactions on Information Theory*, 16(4), 368–372.
- Henriques, D. Y. P., Medendorp, W. P., Khan, A. Z., & Crawford, J. D. (2002). Visuomotor transformations for eye-hand coordination. In D. M. W. H. J. Hyona & R. Radach (Eds.), *The Brain's Eye: Neurobiological and Clinical Aspects of Oculomotor Research*, volume 140 of *Progress in Brain Research* (pp. 329 – 340). Elsevier.
- Herd, S. A., Banich, M. T., & O'Reilly, R. C. (2006). Neural mechanisms of cognitive control: An integrative model of stroop task performance and fMRI data. *Journal of Cognitive Neuroscience*, 18(1), 22–32.
- Heskes, T. (2001). Self-organizing maps, vector quantization, and mixture modeling. *IEEE Transactions on Neural Networks*, 12(6), 1299–1305.
- Hocking, R. R. (1976). The analysis and selection of variables in linear regression. *Biometrics*, 32.
- Holste, D., Grosse, I., & Herzel, H. (1998). Bayes' estimators of generalized entropies.
- Hoyer, P. O. & Hyvärinen, A. (2002). Interpreting neural response variability as Monte Carlo sampling of the posterior. In *Advances in Neural Information Processing Systems*, volume 15.
- Hu, S., Poskitt, D. S., & Zhang, X. (2012). Bayesian adaptive bandwidth kernel density estimation of irregular multivariate distributions. *Computational Statistics & Data Analysis*, 56(3), 732–740.
- Hubel, D. & Wiesel, T. (2005). *Brain and visual perception: the story of a 25-year collaboration*. Oxford University Press US. p. 106.
- Hupe, J. M., James, A. C., Girard, P., Lombber, S. G., Payne, B. R., & Bullier, J. (2001). Feedback connections act on the early part of the responses in monkey visual cortex. *Journal of Neurophysiology*, 85, 134–145.

- Hyvärinen, A., Karhunen, J., & Oja, E. (2004). *Independent Component Analysis*. Adaptive and learning systems for signal processing, communications and control series. Wiley.
- Irwin, D. E. (1991). Information integration across saccadic eye movements. *Cognitive Psychology*, 23(3), 420 – 456.
- Itti, L. & Baldi, P. F. (2006). Bayesian surprise attracts human attention. In *Advances in Neural Information Processing Systems 19* (pp. 547–554). Cambridge, MA: MIT Press.
- Itti, L. & Koch, C. (2000). A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision Research*, 40, 1489–1506.
- Ivanov, A. V. & Rozhkova, M. N. (1981). Properties of the statistical estimate of the entropy of a random vector with a probability density. *Problems of Information Transmission*, 17, 171–178. in Russian.
- Jain, A. & Zongker, D. (1997). Feature selection: Evaluation, application, and small sample performance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(2), 153–158.
- Jeffreys, H. (1946). An invariant form for the prior probability in estimation problems. *Proceedings of the Royal Society A: Mathematical, physical & engineering sciences*, 186, 453–461.
- Jeffreys, H. (1948). *Theory of probability*. Clarendon Press, Oxford, second edition.
- Jelinek, F. & Mercer, R. L. (1980). Interpolated estimation of Markov source parameters from sparse data. In *In Proceedings of the Workshop on Pattern Recognition in Practice* (pp. 381–397). Amsterdam, The Netherlands.
- Jiang, H. (2008). *Adaptive Feature Selection in Pattern Recognition and Ultra-Wideband Radar Signal Analysis*. PhD thesis, California Institute of Technology.
- Joe, H. (1989). On the estimation of entropy and other fuunctionals of a multivariate density. *Annals of the Institute of Statistical Mathematics*, 41, 683–697.
- Johnson, W. E. (1932). Probability: deductive and inductive problems. *Mind*, 41, 421–423.
- Johnston, W. A. & Dark, V. J. (1986). Selective attention. *Annual Review of Psychology*, 37(1), 43–75.
- Jolliffe, I. T. (1986). *Principal Components Analysis*. Springer-Verlag.

- Jones, H. E., Andolina, I. M., Oakely, N. M., Murphy, P. C., & Sillito, A. M. (2000). Spatial summation in lateral geniculate nucleus and visual cortex. *Experimental Brain Research*, 135, 279–284.
- Jost, J. (2004). External and internal complexity of complex adaptive systems. *Theory Bioscience*, 123, 69–88.
- Kanan, C., Tong, M. H., Zhang, L., & Cottrell, G. W. (2009). SUN: Top-down saliency using natural statistics. *Visual Cognition*, 17(6), 979–1003.
- Khan, S., Bandyopadhyay, S., Ganguly, A. R., Saigal, S., Erickson, D. J., Protopopescu, V., & Ostrouchov, G. (2007). Relative performance of mutual information estimation methods for quantifying the dependence among short and noisy data. *Physical Review E*, 76, 026209.
- Kiefer, M., Ansorge, U., Haynes, J.-D., Hamker, F., Mattler, U., Verleger, R., & Niedeggen, M. (2011). Neuro-cognitive mechanisms of conscious and unconscious visual perception: From a plethora of phenomena to general principles. *Advances in Cognitive Psychology*, 7, 55–67.
- Kiefer, M. & Martens, U. (2010). Attentional sensitization of unconscious cognition: task sets modulate subsequent masked semantic priming. *Journal of experimental psychology. General*, 139(3), 464–89.
- Kietzmann, T. C., Geuter, S., & König, P. (2011). Overt visual attention as a causal factor of perceptual awareness. *PLoS ONE*, 6(7), e22614.
- Kira, K. & Rendell, L. A. (1992). A practical approach to feature selection. In *Proceedings of the Ninth International Workshop on Machine Learning*, ML92 (pp. 249–256). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Koch, C. (2004). *The Quest for Consciousness: A Neurobiological Approach*. Roberts and Co.
- Koch, C. & Ullman, S. (1985). Shifts in selective visual attention: towards the underlying neural circuitry. *Human Neurobiology*, 4, 219–227.
- Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2*, IJCAI'95 (pp. 1137–1143). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Kohavi, R. & John, G. (1997). Wrapper for feature subset selection. *Artificial Intelligence*, 97(nos. 1-2), 273–324.
- Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43(1), 59–69.

- Koivisto, M., Railo, H., Revonsuo, A., Vanni, S., & Salminen-Vaparanta, N. (2011). Recurrent processing in V1/V2 contributes to categorization of natural scenes. *The Journal of Neuroscience*, 31(7), 2488–2492.
- Kolb, H. (2011). Feedback loops. In H. Kolb, R. Nelson, E. Fernandez, & B. Jones (Eds.), *Webvision. The Organization of the Retina and Visual System*. <http://webvision.med.utah.edu/book/part-iii-retinal-circuits/feedback-loops/>. accessed 30/11/2012.
- Koller, D. & Sahami, M. (1996). Toward optimal feature selection. In *Proceedings of the 13th International Conference on Machine Learning (ICML-96)* (pp. 284–292).
- Kozachenko, L. F. & Leonenko, N. N. (1987). On statistical estimation of entropy of random vector. *Problems of Information Transmission*, 23(2), 9–16. in Russian.
- Kraskov, A., Stögbauer, H., & Grassberger, P. (2004). Estimating mutual information. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 69(6 Pt 2).
- Krichevsky, R. & Trofimov, V. (1981). The performance of universal encoding. *IEEE Transactions on Information Theory*, 27(2), 199–207.
- Kudo, M. & Sklansky, J. (2000). Comparison of algorithms that select features for pattern classifiers. *Pattern Recognition*, 33(1), 25–41.
- Kwak, N. & Choi, C.-H. (2002a). Input feature selection by mutual information based on Parzen window. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24, 1667–1671.
- Kwak, N. & Choi, C.-H. (2002b). Input feature selection for classification problems. *IEEE Transactions on Neural Networks*, 13(1), 143–159.
- Land, M., Mennie, N., & Rusted, J. (1999). The roles of vision and eye movements in the control of activities of daily living. *Perception*, 28(11), 1311–1328.
- LeCun, J. & Cortes, C. (n.d.). The MNIST dataset of handwritten digits. Retrieved from: <http://yann.lecun.com/exdb/mnist/>.
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324.
- Lee, P. M. (2004). *Bayesian Statistics: An Introduction*. A Hodder Arnold Publication. John Wiley & Sons.
- Lee, T. S. & Mumford, D. (2003). Hierarchical Bayesian inference in the visual cortex. *Journal of the Optical Society of America A*, 20(7), 1434–1448.

- Lee, T. S., Yang, C. F., Romero, R. D., & Mumford, D. (2002). Neural activity in early visual cortex reflects behavioral experience and higher-order perceptual saliency. *Nature Neuroscience*, 5(6), 589–597.
- Leonenko, N., Pronzato, L., & Savani, V. (2008). A class of Rényi information estimators for multidimensional densities. *Annals of Statistics*, 36, 2153—2182. (correction: 2010, 38, 3837–3838).
- Lewicki, M. S., Sejnowski, T. J., & Hughes, H. (1998). Learning Overcomplete Representations. *Neural Computation*, 12, 337–365.
- Lewis, P. M. (1962). The characteristic selection problem in recognition systems. *IRE Transactions on Information Theory*, IT-8, 171–178.
- Li, W. (1990). Mutual information functions versus correlation functions. *Journal of Statistical Physics*, 60, 823–837.
- Lidstone, G. J. (1920). Note on the general case of the bayes-laplace formula for inductive or a posteriori probabilities. *Transactions of the Faculty of Actuaries*, 8, 182–192.
- Lima, C. F. L., de Assis, F. M., & de Souza, C. P. (2010). Decision tree based on Shannon, Rényi and Tsallis entropies for intrusion tolerant systems. In *Proceedings of the 2010 Fifth International Conference on Internet Monitoring and Protection, ICIMP '10* (pp. 117–122). Washington, DC, USA: IEEE Computer Society.
- Lin, D. & Tang, X. (2006). Conditional infomax learning: An integrated framework for feature extraction and fusion. In *Proceedings of the 9th European conference on Computer Vision - Volume Part I, ECCV'06* (pp. 68–82). Berlin, Heidelberg: Springer-Verlag.
- Liu, C.-T. & Hu, B.-G. (2009). Mutual information based on Rényi's entropy feature selection. In *Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on*, volume 1 (pp. 816–820).
- Liu, H., Liu, L., & Zhang, H. (2008). Feature selection using mutual information: An experimental study. In T.-B. Ho & Z.-H. Zhou (Eds.), *PRICAI 2008: Trends in Artificial Intelligence*, volume 5351 of *Lecture Notes in Computer Science* (pp. 235–246). Springer Berlin Heidelberg.
- Liu, L. H. & Motoda, H. (1998). *Feature Extraction, Construction and Selection: A Data Mining Perspective*. The Kluwer international series in engineering and computer science. Kluwer Academic Publishers.
- Livingstone, M. S. & Hubel, D. H. (1987). Psychophysical evidence for separate channels for the perception of form, color, movement, and depth. *Journal of Neuroscience*, 7(11), 3416–3468.

- Loftsgaarden, D. O. & Quesenberry, C. P. (1965). A nonparametric estimate of a multivariate density function. *The Annals of Mathematical Statistics*, 36(3), 1049–1051.
- Lopes, F. M., Oliveira, E. A., & Cesar, R. M. J. (2009). Analysis of the GRNs inference by using Tsallis entropy and a feature selection approach. In E. Bayro-Corrochano & J.-O. Eklundh (Eds.), *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, volume 5856 of *Lecture Notes in Computer Science* (pp. 473–480). Springer Berlin Heidelberg.
- Mackay, D. J. C. (2003). *Information Theory, Inference, and Learning Algorithms*. Cambridge: Cambridge University Press.
- Macknik, S. L. & Martinez-Conde, S. (2008). The role of feedback in visual masking and visual processing. *Advances in cognitive psychology*, 3(1-2), 125–152.
- Macknik, S. L. & Martinez-Conde, S. (2009). The role of feedback in visual attention and awareness. In M. Gazzaniga (Ed.), *The Cognitive Neurosciences* (pp. 1165–1175). Cambridge: MIT Press, 4th edition edition.
- Marill, T. & Green, D. (1963). On the effectiveness of receptors in recognition systems. *IEEE Transactions on Information Theory*, 9(1), 11–17.
- Maszczyk, T. & Duch, W. (2008). Comparison of Shannon, Renyi and Tsallis entropy used in decision trees. In *ICAISC* (pp. 643–651).
- Mattler, U. (2005). Inhibition and decay of motor and nonmotor priming. *Perception & psychophysics*, 67(2), 285–300.
- McCulloch, W. S. & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4), 115–133.
- McLachlan, G. (2004). *Discriminant Analysis and Statistical Pattern Recognition*. Wiley Series in Probability and Statistics. Wiley.
- Medendorp, W. P., Goltz, H. C., Vilis, T., & Crawford, J. D. (2003). Gaze-centered updating of visual space in human parietal cortex. *The Journal of Neuroscience*, 23(15), 6209–6214.
- Meyer, P. E. & Bontempi, G. (2006). On the use of variable complementarity for feature selection in cancer classification. In *Proceedings of the 2006 International Conference on Applications of Evolutionary Computing*, EuroGP'06 (pp. 91–102). Berlin, Heidelberg: Springer-Verlag.
- Miller, A. J. (1990). *Subset Selection in Regression*. Chapman and Hall.
- Miller, E. K. & Cohen, J. D. (2001). An integrative theory of prefrontal cortex function. *Annual Review of Neuroscience*, 24(1), 167–202.

- Miller, G. A. (1955). Note on the bias of information estimates. In H. Quastler (Ed.), *Information Theory in Psychology: Problems and Methods* (pp. 95–100). Glencoe, IL: Free Press.
- Mingers, J. (1987). Expert systems - rule induction with statistical data. *Journal of the Operational Research Society*, 38(1), 39–47.
- Mishkin, M. & Ungerleider, L. G. (1982). Contribution of striate inputs to the visuospatial functions of parieto-preoccipital cortex in monkeys. *Behavioural Brain Research*, 6(1), 57–77.
- Miyahara, K. & Pazzani, M. (2000). Collaborative filtering with the simple Bayesian classifier. In *PRICAI 2000 Topics in Artificial Intelligence* (pp. 679–689).
- Moddemeijer, R. (1989). On estimation of entropy and mutual information of continuous distributions. *Signal Processing*, 16(3), 233–248.
- Montani, F., Kohn, A., Smith, M. A., & Schultz, S. R. (2007). The role of correlations in direction and contrast coding in the primary visual cortex. *Journal of Neuroscience*, 27, 2338–2348.
- Montero, V. M. (1991). A quantitative study of synaptic contacts on internunciations and relay cells of the cat lateral geniculate nucleus. *Experimental Brain Research*, 86, 257–270.
- Moon, Y.-L., Rajagopalan, R., & Lall, U. (1995). Estimation of mutual information using kernel density estimators. *Physical Review E*, 52(3), 2318–2321.
- Mosteller, F. & Tukey, J. W. (1977). *Data Analysis and Regression: A Second Course in Statistics*. Addison-Wesley series in behavioral science. Addison Wesley Publishing Company.
- Mumford, D. (1991). On the computational architecture of the neocortex: I. The role of the thalamo-cortical loop. *Biological Cybernetics*, 65, 135–145.
- Murphy, P. C., Duckett, S. G., & Sillito, A. M. (1999). Feedback connections to the lateral geniculate nucleus and cortical response properties. *Science*, 286, 1552–1554.
- Murphy, P. C., Duckett, S. G., & Sillito, A. M. (2000). Comparison of the laminar distribution of input from areas 17 and 18 of the visual cortex to the lateral geniculate nucleus of the cat. *Journal of Neuroscience*, 20(2), 845–853.
- Nadaraya, E. A. (1964). On estimating regression. *Theory of Probability and its Applications*, 9, 134–137.
- Najemnik, J. & Geisler, W. S. (2005). Optimal eye movement strategies in visual search. *Nature*, 434, 387–391.

- Narendra, P. & Fukunaga, K. (1977). A branch and bound algorithm for feature subset selection. *IEEE Transactions on Computers*, 28(2), 917–922.
- Navalpakkam, V. & Itti, L. (2005). Modeling the influence of task on attention. *Vision Research*, 45(2), 205 – 231.
- Navalpakkam, V., Jentzsch, L., Sayres, R., Ravi, S., & Ahmed, A. S. A. (2013). Measurement and modeling of eye-mouse behavior in the presence of nonlinear page layouts. In *Proceedings of the 22nd International Conference on World Wide Web, WWW '13* (pp. 953–964). Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee.
- Navalpakkam, V., Koch, C., Rangel, A., & Perona, P. (2010). Optimal reward harvesting in complex perceptual environments. *Proceedings of the National Academy of Sciences*, 107(11), 5232–5237.
- Nemenman, I., Shafee, F., & Bialek, W. (2002). Entropy and inference, revisited. In *Advances in Neural Information Processing Systems 14*: MIT Press.
- Obermayer, K. & Sejnowski, T. J. (2001). *Self-organizing map formation: Foundations of neural computation*. A Bradford book. MIT Press.
- Olshausen, B. A. & Field, D. J. (1996). Emergence of simple cell receptive field properties by learning a sparse code for natural images. *Nature*, 381, 607–609.
- O'Regan, J. K. & Noe, A. (2001). A sensorimotor account of vision and visual consciousness. *Behavioral and Brain Sciences*, 24(5), 939–973.
- Ossandón, J. P., Onat, S., Cazzoli, D., Nyffeler, T., Müri, R., & König, P. (2012). Unmasking the contribution of low-level features to the guidance of attention. *Neuropsychologia*, 50(14), 3478–3487.
- Ozertem, U., Erdogmus, D., & Jenssen, R. (2006). Spectral feature projections that maximize Shannon mutual information with class labels. *Pattern Recognition*, 39(7), 1241–1252.
- Paninski, L. (2003). Estimation of entropy and mutual information. *Neural Computation*, 15(6), 1191–1253.
- Panzeri, S., Senatore, R., Montemurro, M. A., & Petersen, R. S. (2007). Correcting for the sampling bias problem in spike train information measures. *Journal of Neurophysiology*, 98(3), 1064–1072.
- Papantoni-Kazakos, P. (1976). *Some Distance Measures and Their Use in Feature Selection*. Defense Technical Information Center.

- Parzen, E. (1962). On estimation of a probability density and mode. *Annals of Mathematical Statistics*, 35, 1065–1076.
- Pearson, K. (1895). Contributions to the mathematical theory of evolution. II. Skew variation in homogeneous material. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 186, 343—414.
- Pearson, K. (1901). On lines and plains of closest fit to systems of points in space. *Philosophical Magazine*, 2, 559–572.
- Peng, H., Long, F., & Ding, C. (2005). Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27, 1226–1238.
- Posner, M. I. & Petersen, S. E. (1990). The Attention System of the Human Brain. *Annual Review of Neuroscience*, 13(1), 25–42.
- Premus, V. & Alexandrou, D. (1995). Maximum a posteriori probability estimation of seafloor microroughness parameters from backscatter spatial coherence. In *International Conference on Acoustics, Speech, and Signal Processing, 1995. ICASSP-95*, volume 5 (pp. 3119–3122).
- Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (1988). *Numerical recipes in C. The art of scientific computing*. Cambridge, UK: Cambridge University Press.
- Pudil, P., Novovičová, J., & Kittler, J. (1994). Floating search methods in feature selection. *Pattern Recognition Letters*, 15(11), 1119–1125.
- Qiu, P., Gentles, A. J., & Plevritis, S. K. (2009). Fast calculation of pairwise mutual information for gene regulatory network reconstruction. *Computer Methods and Programs in Biomedicine*, 94(2), 177–180.
- Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, 1, 81.
- Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Raizada, R. D. S. & Grossberg, S. (2003). Towards a Theory of the Laminar Architecture of Cerebral Cortex: Computational Clues from the Visual System. *Cerebral Cortex*, 13(1), 100–113.
- Ramshaw, J. (1995). Thermodynamic stability conditions for the Tsallis and Rényi entropies. *Physical Letters*, A(198), 119.
- Raudys, S. J. & Jain, A. K. (1991). Small sample size effects in statistical pattern recognition: Recommendations for practitioners. *IEEE Transactions on Pattern Analysis and*

- Machine Intelligence*, 13(3), 252–264.
- Renninger, L. W., Verghese, P., & Coughlan, J. (2007). Where to look next? Eye movements reduce local uncertainty. *Journal of Vision*, 7(3), 1–17.
- Rényi, A. (1961). On measures of information and entropy. In *Proceedings of the 4th Berkeley Symposium on Mathematics, Statistics and Probability 1960* (pp. 547–561).
- Reunanen, J. (2006). Search strategies. In I. Guyon, M. Nikravesh, S. Gunn, & L. Zadeh (Eds.), *Feature Extraction*, volume 207 of *Studies in Fuzziness and Soft Computing* (pp. 119–136). Springer Berlin Heidelberg.
- Rezā, F. (1961). *An Introduction to Information Theory*. Dover Books on Mathematics Series. DOVER PUBN Incorporated.
- Riesenhuber, M. & Poggio, T. (1999). Hierarchical models of object recognition in cortex. *Nature Neuroscience*, 2, 1019–1025.
- Robert, C. (2001). *The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation*. Springer Texts in Statistics. Springer.
- Rolls, E. T. (2008). *Memory, Attention, and Decision-Making: A Unifying Computational Neuroscience Approach*. Oxford University Press, Incorporated.
- Rosa, M. G. P. (2002). Visual cortex. In V. S. Ramachandran (Ed.), *Encyclopedia of the Human Brain*, volume 4 (pp. 753–773). Academic Press.
- Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386–408.
- Rosenblatt, M. (1956). Remarks on some nonparametric estimates of a density function. *Annals of Mathematical Statistics*, 27, 832–837.
- Rothkopf, C. A., Ballard, D. H., & Hayhoe, M. M. (2007). Task and context determine where you look. *Journal of Vision*, 7(14), 16.
- Rudemo, M. (1982). Empirical choice of histograms and kernel density estimators. *Scandinavian Journal of Statistics*, 9, 65–78.
- Sain, S. R., Baggerly, K. A., & Scott, D. W. (1992). Cross-validation of multivariate densities. *Journal of the American Statistical Association*, 89, 807–817.
- Schaffernicht, E., Kaltenhaeuser, R., Verma, S. S., & Gross, H.-M. (2010). On estimating mutual information for feature selection. In *Proceedings of the 20th International Conference on Artificial Neural Networks: Part I, ICANN'10* (pp. 362–367). Berlin, Heidelberg: Springer-Verlag.

- Schall, J. D., Morel, A., King, D. J., & Bullier, J. (1995). Topography of visual cortex connections with frontal eye field in macaque: convergence and segregation of processing streams. *The Journal of Neuroscience*, 15(6), 4464–4487.
- Schürmann, T. & Grassberger, P. (1996). Entropy estimation of symbol sequences. *Chaos*, 6, 414–427.
- Scott, D. W. (1979). On optimal and data-based histograms. *Biometrika*, 66(3), 605–610.
- Scott, D. W. (1985). Frequency polygons, theory and application. *Journal of the American Statistical Association*, 80, 348—354.
- Scott, D. W. (1992). *Multivariate Density Estimation: Theory, Practice, and Visualization*. John Wiley. pp. 125–206.
- Scott, D. W. (2004). *Multivariate Density Estimation and Visualization*. Papers / 16, Humboldt-Universität Berlin, Center for Applied Statistics and Economics (CASE).
- Serre, T., Kreiman, G., Kouh, M., Cadieu, C., Knoblich, U., & Poggio, T. (2007). A quantitative theory of immediate visual recognition. In P. Cisek, T. Drew, & J. F. Kalaska (Eds.), *Computational Neuroscience: Theoretical Insights into Brain Function*, volume 165 of *Progress in Brain Research* (pp. 33–56). Elsevier.
- Shang, W., Huang, H., Zhu, H., Lin, Y., Qu, Y., & Wang, Z. (2007). A novel feature selection algorithm for text categorization. *Expert Systems with Applications*, 33(1), 1–5.
- Shannon, C. E. & Weaver, W. (1949). *The Mathematical Theory of Communication*. Urbana and Chicago: University of Illinois Press.
- Sherman, S. M. (2001). Tonic and burst firing: dual modes of thalamocortical relay. *Trends in Neurosciences*, 24(2), 122–126.
- Sillito, A. M., Cudeiro, J., & Jones, H. E. (2006). Always returning: feedback and sensory processing in visual cortex and thalamus. *Trends in Neurosciences*, 29(6), 307–316. TINS special issue: The Neural Substrates of Cognition.
- Silverman, B. W. (1986). *Density estimation for statistics and data analysis*. Chapman and Hall.
- Simoncelli, E. P., Freeman, W. T., Adelson, E. H., & Heeger, D. J. (1992). Shiftable Multi-scale Transforms. *IEEE transactions on informations theory*, 38(2). MIT Media Laboratory Vision and Modeling Technical Report #161.
- Singh, H., Misra, N., Hnizdo, V., Fedorowicz, A., & Demchuk, E. (2003). Nearest neighbor estimates of entropy. *American Journal of Mathematical and Management Sciences*, 23, 301–321.

- Singh, M. & Provan, G. M. (1995). A comparison of induction algorithms for selective and non-selective Bayesian classifiers. In *Proceedings of the Twelfth International Conference on Machine Learning* (pp. 497–505).: Morgan Kaufmann.
- Sirosh, J. & Miikkulainen, R. (1994). Cooperative self-organization of afferent and lateral connections in cortical maps. *Biological Cybernetics*, (pp. 66–78).
- Somol, P., Baesens, B., Pudil, P., & Vanthienen, J. (2005). Filter- versus wrapper-based feature selection for credit scoring. *International Journal of Intelligent Systems*, 20(10), 985–999.
- Somol, P., Novovicova, J., & Pudil, P. (2007). Notes on the evolution of feature selection methodology. *Kybernetika*, 43, 713–730.
- Somol, P., Pudil, P., & Kittler, J. (2004). Fast branch & bound algorithms for optimal feature selection. *IEEE Pattern Analysis and Machine Intelligence*, 26, 900–912.
- Stariolo, D. A. & Tsallis, C. (1996). Generalized simulated annealing. *Physica A*, 233(1–2), 395–406.
- Steuer, R., Kurths, J., Daub, C. O., Weise, J., & Selbig, J. (2002). The mutual information: Detecting and evaluating dependencies between variables. *Bioinformatics*, 18(suppl 2), 231–240.
- Stowell, D. & Plumbley, M. D. (2009). Fast multidimensional entropy estimation by k-d partitioning. *IEEE Signal Processing Letters*, 16(6), 537–540.
- Strong, S. P., Koberle, R., de Ruyter van Steveninck, R., & Bialek, W. (1998). Entropy and information in neural spike trains. *Physical Review Letters*, 86, 197–200.
- Sun, Y. (2007). Iterative RELIEF for feature weighting: Algorithms, theories, and applications. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(6), 1035–1051.
- Sznitman, R. & Jedynek, B. (2010). Active testing for face detection and localization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(10), 1914–1920.
- Tatler, B. W., Hayhoe, M. M., Land, M. F., & Ballard, D. H. (2011). Eye guidance in natural vision: Reinterpreting salience. *Journal of Vision*, 11(5).
- Taylor, C. C. (1989). Bootstrap choice of the smoothing parameter in kernel density estimation. *Biometrika*, 76(4), 705–712.
- Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B. Methodological*, 58(1), 267–288.

- Torralba, A., Castelhano, M. S., Oliva, A., & Henderson, J. M. (2006). Contextual guidance of eye movements and attention in real-world scenes: The role of global features in object search. *Psychological Review*, 113, 766–786.
- Trappenberg, T., O., J., & Back, A. (2006). Input variable selection: mutual information and linear mixing measures. *IEEE Transactions on Knowledge and Data Engineering*, 18(1), 37–46.
- Treisman, A. M. & Gelade, G. (1980). A feature-integration theory of attention. *Cognitive Psychology*, 12(1), 97–136.
- Treves, A. & Panzeri, S. (1995). The upward bias in measures of information derived from limited data samples. *Neural Computation*, 7, 399–407.
- Tsallis, C. (1988). Possible generalization of Boltzmann-Gibbs statistics. *Journal of Statistical Physics*, 52, 479–487.
- Tsamardinos, I., Aliferis, C. F., & Statnikov, A. (2003). Algorithms for large scale Markov blanket discovery. In *16th International FLAIRS Conference*, volume 103.
- Tu, P.-L. & Chung, J.-Y. (1992). A new decision-tree classification algorithm for machine learning. In *Fourth International Conference on Tools with Artificial Intelligence, TAI '92* (pp. 370–377).
- Tukey, J. W. (1977). *Exploratory Data Analysis*. Addison-Wesley.
- Turlach, B. A. (1993). Bandwidth selection in kernel density estimation: A review. In *CORE and Institut de Statistique* (pp. 23–493).
- Ullman, S., Vidal-Naquet, M., & Sali, E. (2002). Visual features of intermediate complexity and their use in classification. *Nature Neuroscience*, 5(7), 682–687.
- Valdes-Sosa, M., Bobes, M. A., Rodriguez, V., & Pinilla, T. (1998). Switching attention without shifting the spotlight: object-based attentional modulation of brain potentials. *Journal of Cognitive Neuroscience*, 10, 137–151.
- van de Laar, P., Heskes, T., & Gielen, S. (1997). Task-dependent learning of attention. *Neural Networks*, 10(6), 981–992.
- Van Essen, D., Olshausen, B. A., Anderson, C. H., & Gallant, J. T. (1991). Pattern recognition, attention, and information bottlenecks in the primate visual system. In *Proceedings of SPIE Conference on Visual Information Processing: From Neurons to Chips* (pp. 17–28).
- Van Essen, D. C., Felleman, D. J., DeYoe, E. A., Olavarria, J., & Knierin, J. (1990). Modular and hierarchical organization of extrastriate visual cortex in the macaque monkey. *Cold Spring Harbor Symposia on Quantitative Biology*, 55, 679–696.

- Van Hulle, M. M. (2005). Edgeworth approximation of multivariate differential entropy. *Neural Computation*, 17(9), 1903–1910.
- VanRullen, R. (2007). The power of the feed-forward sweep. *Advances in Cognitive Psychology*, 3, 167–176.
- Vapnik, V. N. (1998). *Statistical Learning Theory*. Wiley-Interscience, 1 edition.
- Vasicek, O. (1976). On a test for normality based on sample entropy. *Journal of Royal Statistical Society, Series B*, 38, 54–59.
- Victor, J. (2000). Asymptotic bias in information estimates and the exponential (Bell) polynomials. *Neural Computation*, 12, 2797–2804.
- Vidal-Naquet, M. & Ullman, S. (2003). Object Recognition with Informative Features and Linear Classification. In *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2, ICCV '03* (pp. 281–). Washington, DC, USA: IEEE Computer Society.
- Watson, G. S. (1964). Smooth regression analysis. *Sankhyā Ser.*, 26, 359–372.
- Webb, A. (1999). *Statistical Pattern Recognition*. Arnold, London. pp. 213–226.
- Weston, J., Mukherjee, S., Chapelle, O., Pontil, M., & Vapnik, V. (2000). Feature selection for SVMs. In *Advances in Neural Information Processing Systems 13* (pp. 668–674).
- Whitney, A. W. (1971). A direct method of nonparametric measurement selection. *IEEE Transactions on Computers*, 20(9), 1100–1103.
- Wilson, J. R. (1993). Circuitry of the dorsal lateral geniculate nucleus in the cat and monkey. *Acta Anatomica*, 147(1), 1–13.
- Wolfe, J. M. (1994). Guided Search 2.0 A revised model of visual search. *Psychonomic Bulletin & Review*, 1, 202–238.
- Wolpert, D. H. & Wolf, D. R. (1995). Estimating functions of probability distributions from a finite set of samples. *Physical Review E*, 52, 6841–6854.
- Wyatte, D., Herd, S., Mingus, B., & O'Reilly, R. (2012). The role of competitive inhibition and top-down feedback in binding during object recognition. *Frontiers in Psychology*, 3(182).
- Xu, R., Chen, Y.-W., Tang, S.-Y., Morikawa, S., & Kurumi, Y. (2008). Parzen-window based normalized mutual information for medical image registration. *IEICE - Transactions on Information and Systems*, E91-D(1), 132–144.

- Xuan, G., Zhu, X., Chai, P., Zhang, Z., Shi, Y., & Fu, D. (2006). Feature selection based on the Bhattacharyya distance. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 4 (pp. 957–957).
- Yang, H. H. & Moody, J. (1999). Data visualization and feature selection: New algorithms for nongaussian data. In *Advances in Neural Information Processing Systems* (pp. 687–693).: MIT Press.
- Yang, J., Liu, Y., Liu, Z., Zhu, X., & Zhang, X. (2011). A new feature selection algorithm based on binomial hypothesis testing for spam filtering. *Knowledge-Based Systems*, 24(6), 904–914.
- Yang, S.-H. & Hu, B.-G. (2012). Discriminative feature selection by nonparametric Bayes error minimization. *IEEE Transactions on Knowledge and Data Engineering*, 24(8), 1422–1434.
- Yang, Y. & Raine, A. (2009). Prefrontal structural and functional brain imaging findings in antisocial, violent, and psychopathic individuals: A meta-analysis. *Psychiatry Research: Neuroimaging*, 174(2), 81–88.
- Yarbus, A. L. (1967). *Eye Movements and Vision*. Plenum. New York.
- Yin, H. & Allinson, N. M. (2001). Bayesian self-organising map for Gaussian mixtures. *Vision, Image and Signal Processing, IEE Proceedings -*, 148(4), 234–240.
- Yu, B. & Yuan, B. (1993). A more efficient branch and bound algorithm for feature selection. *Pattern Recognition*, 26(6), 883–889.
- Yu, S.-N. & Lee, M.-Y. (2012). Conditional mutual information-based feature selection for congestive heart failure recognition using heart rate variability. *Computer Methods and Programs in Biomedicine*, 108(1), 299–309.
- Zhai, C. (2008). Statistical language models for information retrieval: A critical review. *Foundations and Trends in Information Retrieval*, 2(3), 137–213.
- Zhang, X., King, M. L., & Hyndman, R. J. (2004). *Bandwidth selection for multivariate kernel density estimation using MCMC*. Technical report, Monash University.
- Zhang, Z. & Hancock, E. R. (2011). Mutual information criteria for feature selection. In M. Pelillo & E. R. Hancock (Eds.), *Similarity-Based Pattern Recognition*, volume 7005 of *Lecture Notes in Computer Science* (pp. 235–249). Springer Berlin Heidelberg.
- Zhou, G., Yang, L., Su, J., & Ji, D. (2005). Mutual information independence model using kernel density estimation for segmenting and labeling sequential data. In A. Gelbukh (Ed.), *Computational Linguistics and Intelligent Text Processing*, volume 3406 of *Lecture Notes in Computer Science* (pp. 155–166). Springer Berlin Heidelberg.

BIBLIOGRAPHY

- Zou, H. & Hastie, T. (2005). Regularization and variable selection via the Elastic Net. *Journal of the Royal Statistical Society, Series B*, 67, 301–320.

Selbstständigkeitserklärung

Hiermit erkläre ich, die vorliegende Dissertation selbständig und ohne unzulässige fremde Hilfe angefertigt zu haben. Ich habe keine anderen als die angeführten Quellen und Hilfsmittel benutzt und sämtliche Textstellen, die wörtlich oder sinngemäß aus veröffentlichten oder unveröffentlichten Schriften entnommen wurden, und alle Angaben, die auf mündlichen Auskünften beruhen, als solche kenntlich gemacht. Ebenfalls sind alle von anderen Personen bereitgestellten Materialien oder erbrachten Dienstleistungen als solche gekennzeichnet.

Leipzig, den 31. Januar 2014

.....

(Liliya Avdiyenko)

BIBLIOGRAPHY

Daten zum Autor

Name:	Liliya Avdiyenko
Geburtsdatum:	28.01.1985
Geburtsort:	Charkiw, Ukraine
09/2002 - 06/2006	Bachelor of Science, Informatik Nationale Universität für Radioelektronik Charkiw, Ukraine Titel der Arbeit: "Datenverarbeitung mittels ontogenischen neuronalen Netze"
09/2006 - 06/2007	Master of Science, Informatik Nationale Universität für Radioelektronik Charkiw, Ukraine Titel der Arbeit: "Intelligente Verarbeitung der Zeitreihen mittels hybriden neuronalen Netze"
10/2006 - 09/2009	Master of Philosophy, Informatik Fernstudium mit Präsenzphase Wessex Institute of Technology, University of Wales, Großbritannien Titel der Arbeit: "Heterogeneous model of neural networks using the learning algorithms based on dynamic modification of network topology"
seit 09/2008	Doktorandin am Max-Planck-Institut für Mathematik in den Naturwissenschaften, Leipzig, Deutschland Dissertation: "Adaptive sequential feature selection in visual perception and pattern recognition"

Bibliographische Daten

Adaptive sequential feature selection in visual perception and pattern recognition
(Adaptive sequentielle Featureauswahl in visueller Wahrnehmung und Mustererkennung)
Avdiyenko, Liliya
Universität Leipzig, Dissertation, 2014
172 Seiten, 29 Abbildungen, 313 Referenzen