Shape Analysis in Bioinformatics

Emma Marie Petty

Submitted in accordance with the requirements for the degree of Doctor of Philosophy

The University of Leeds

Department of Statistics

July 2009

Acknowledgements

This research was jointly funded as a CASE studentship by EPSRC and the Central Science Laboratory (CSL), York.

I would like to thank my supervisors, Kanti Mardia and Charles Taylor, for their continued support and inspiration throughout this research. I would like to thank Tim for helping me with my many computing quandaries. I would also like to thank Qasim Chaudhry, Jane Cotterill (CSL), Roz Banks and Rachel Craven (St James's University Hospital) for the data provided and the ideas shared to promote the multi-disciplinary work within this thesis.

I would like to thank the many weird and wonderful people in the maths department who have made coming into work an absolute pleasure.

I would like to thank my friends, within university and outside university, for their shared love of sport or card games or even pick-n-mix.

I would like to thank my boyfriend, Paul Abthorpe, for not once making me feel bad for being an 8-year student, even though my working life may have often *seemed* free and easy compared to his demanding solicitor timetable.

Finally, I would like to thank my family for generally being absolutely fantastic and an honour to be part of.

Abstract

In this thesis we explore two main themes, both of which involve proteins. The first area of research focuses on the analyses of proteins displayed as spots on 2-dimensional planes. The second area of research focuses on a specific protein and how interactions with this protein can naturally prevent or, in the presence of a pesticide, cause toxicity.

The first area of research builds on previously developed EM methodology to infer the matching and transformation necessary to superimpose two partially labelled point configurations, focusing on the application to 2D protein images. We modify the methodology to account for the possibility of missing and misallocated markers, where markers make up the labelled proteins manually located across images. We provide a way to account for the likelihood of an increased edge variance within protein images. We find that slight marker misallocations do not greatly influence the final output superimposition when considering data simulated to mimic the given dataset. The methodology is also successfully used to automatically locate and remove a grossly misallocated marker within the given dataset before further analyses is carried out.

We develop a method to create a union of replicate images, which can then be used alone in further analyses to reduce computational expense. We describe how the data can be modelled to enable the inference on the quality of a dataset, a property often overlooked in protein image analysis. To complete this line of research we provide a method to rank points that are likely to be present in one group of images but absent in a second group. The produced score is used to highlight the proteins that are not present in both image sets representing control or diseased tissue, therefore providing biological indicators which are vitally important to improve the accuracy of diagnosis.

In the second area of research, we test the hypothesis that pesticide toxicity is related to the shape similarity between the pesticide molecule itself and the natural ligand of the protein to which a pesticide will bind (and ultimately cause toxicity). A ligand of a

protein is simply a small molecule that will bind to that protein. It seems intuitive that the similarities between a naturally formed ligand and a synthetically developed ligand (the pesticide) may be an indicator of how well a pesticide and the protein bind, as well as provide an indicator of pesticide toxicity. A graphical matching algorithm is used to infer the atomic matches across ligands, with Procrustes methodology providing the final superimposition before a measure of shape similarity is defined considering the aligned molecules. We find evidence that the measure of shape similarity does provide a significant indicator of the associated pesticide toxicity, as well as providing a more significant indicator than previously found biological indicators.

Previous research has found that the properties of a molecule in its bioactive form are more suitable indicators of an associated activity. Here, these findings dictate that the docked conformation of a pesticide within the protein will provide more accurate indicators of the associated toxicity. So next we use a docking program to predict the docked conformation of a pesticide. We provide a technique to calculate the similarity between the docks of both the pesticide and the natural ligand. A similar technique is used to provide a measure for the closeness of fit between a pesticide and the protein. Both measures are then considered as independent variables for the prediction of toxicity. In this case the results show potential for the calculated variables to be useful toxicity predictors, though further analysis is necessary to properly explore their significance.

Contents

Glossary

2-Dimensional Electrophoresis (2-DE)	A chemical procedure used to separate proteins by mass and acidity to create a 2D gel providing a mapping of all proteins present (see page 2).
Western blot	Sera is used to probe the 2-DE gel. Antibodies within the sera bind to specific proteins and only the proteins with a bound antibody are highlighted within western blot images (see page 4).
DIfference Gel Electrophoresis (DIGE)	A modification on 2-DE used to compare two or three protein samples. The proteins in each sample are tagged with different colours before being mixed together. A single 2-DE gel is produced, where the proteins in a particular sample can be distinguished by colour (see page 7).
Biomarker	A biological indicator of a biological state, i.e. a protein which indicates the presence of some disease (see page 2).
Bind	Describes the way molecules chemically react and come together.

Protein	A large organic molecule that generally has some function within a biological system.
Ligand	A small molecule that will bind with a protein.
Natural ligand or substrate	An organic ligand that binds to a specific protein.
Complex	A general term given to a bound ligand and protein.
Acetylcholinesterase (AChE)	The specific protein that influences toxicity (see page 17)
Acetylcholine (ACh)	The natural ligand of AChE (see page 17).
Inhibit	Term used to describe how other ligands bind with a protein and prevent the protein from carrying out its normal function (see page 18).
Pesticide	A synthetic ligand developed to inhibit AChE and cause toxicity.
Carbamate and Organophosphate (OPs)	Two different families of pesticides.
Lethal Dose 50 (LD50)	The amount of pesticide necessary to kill half a sample of pests.

Binding affinity	A measure of the strength of the bind between a ligand and protein.
Half maximal Inhibitory Concentration 50 (IC50), inhibition constant and binding energy	Different measures of binding affinity (see page 21).
Quantitative Structure Activity Relationships (QSAR)	QSAR dictates that the toxicity of a pesticide is proportional to one or more properties of the pesticide molecule itself (see page 22).
Molecular conformation	The spatial arrangement of a molecule.
Bioactive conformation	The bioactive conformation of a pesticide is the docked conformation within AChE (see page 22).
3D Quantitative Structure Activity Relationships (3D-QSAR)	The same as QSAR, though considering specifically the bioactive, i.e. the docked conformation of a pesticide (see page 22).
Development of Environmental Modules for the Evaluation of the Toxicity of pesticide Residues in Agriculture (DEMETRA)	A project where the main objective is to produce QSAR software for the improvement of toxicity prediction (see page 23).

Simplified Molecular Input Line Entry Specification (SMILES)	Uses ordered sequence of symbols to describe the structure of a molecule.
Protein Data Bank (PDB)	An online archive of experimentally determined molecular structures.
Van der Waals (VdW) radius	Defines the radius of an imaginary sphere often used to represent an atom.
Markers	A set of known corresponding points across all images that account for the partial labelling (see page 11).
Coffin bin	If a point in a second image is not matched to a point in the first image, we say it is allocated to the coffin bin (see page 30).
Slight marker misallocation	When a marker is incorrectly allocated as a nearby point due to the warping within an image.
Standard method	Assumes allocated markers are true markers by fixing the prior matching probability of corresponding markers to be one.
Adapted method	Accounts for slight marker misallocations by allowing the prior matching probability of non-corresponding markers to be non-zero.

Gross marker misallocations	Due to input error of spot IDs when allocating markers (see page 47).
Image contamination	Consists of missing markers which are points that should have been located in an image and imposter points which are points that do not correspond to a real protein (see page 85).
Normoxia/Hypoxia	A normal/lowered amount of oxygen used as two different treatments (see page 10).

Mathematical Notation

Chapters 2, 3, Section 4.2 and Subsection 5.3.1	
D	Number of dimensions the relevant data is represented within.
K	Number of markers, i.e. known corresponding points, that *should* be located in every image.
m_G, n_G	The total number of proteins present in a first and second 2-DE gel respectively (see chemical implementation in Subsection 1.2.4).
μ_G, x_G	$m_G \times D$ and $n_G \times D$ coordinate matrix for all the proteins that would be highlighted as points on a 2-DE image (theoretical in terms of the data we consider).
m, n	Number of non-markers in a first and second image respectively.
μ, x	$(K + m) \times D$ and $(K + n) \times D$ coordinate matrices for a first and second image respectively, where the first K set of coordinates represent markers. Also used more generally to indicate the image represented by μ or x respectively.
μ_i, x_j	$D \times 1$ coordinate vector of the ith point in μ and the jth point in x respectively. Also used more generally to indicate the ith or jth point in μ or x respectively.
A, b	Non-singular $D \times D$ matrix and $D \times 1$ vector respectively that denote the affine transformation parameters.
M	$(K + m + 1) \times (K + n)$ matrix indicating matched points across images, where an element $M_{ij} = 1$ if x_j is matched to the μ_i for $i = 1, \ldots, K + m$ or allocated to the coffin bin for $i = K + m + 1$. For simplicity we set $M_{ij} = M_{0j}$ for $i = K + m + 1$.

Q	$(K+m+1)\times(K+n)$ matrix containing prior matching probabilities, where an element $q_{ij} = p(M_{ij} = 1)$ contains the prior probability that x_j matches μ_i for $i = 1, \ldots, K+m$ or is allocated to the coffin bin for $i = K+m+1$. For simplicity we set $q_{ij} = q_{0j}$ for $i = K+m+1$.
$\|\Omega\|$	The region in $\mathbb{R}^D$ containing all points in x.
σ_{ij}^2	An assigned variance between the points μ_i and x_j in each dimension.
p_{ji}	The posterior probability that x_j matches μ_i for $i = 1, \ldots, K+m$ or is allocated to the coffin bin for $i = 0$.
$\hat{p}$	$(K+n) \times (K+m+1)$ matrix containing the final posterior probabilities output by the EM algorithm. For simplicity we set $\hat{p}_{ji} = \hat{p}_{j0}$ for $i = K+m+1$.
p_{ji}^*	Set as p_{ji}/σ_{ij}^2 for notational simplicity.
d_{ij}	The Euclidean distance between μ_i and x_j after μ has been transformed to fit x using the final transformation parameters output by the EM algorithm.
D^*	$(K+m+1) \times (K+n)$ matrix where an element $D_{ij} = d_{ij}^2$ for $i = 1, \ldots, K+m$ or an assigned squared distance threshold, d_T^2 for $i = K+m+1$. For simplicity we set $D_{ij} = D_{0j}$ for $i = K+m+1$.
d_T	Distance threshold assigned within D^* that maximises the distance allowed between two points that can be matched across images.
Δ	$(K+m+1)\times(K+n)$ matrix that can be set as $\Delta = \hat{p}^T$ or $\Delta = D$ when assigning matches across images. For simplicity we set $\Delta_{ij} = \Delta_{0j}$ for $i = K+m+1$.
L	The number of matched points across μ and x.
μ', x'	$K \times D$ coordinate matrices containing only coordinate information for the markers allocated in μ and x respectively.
p_M	The probability that two correspondingly allocated markers truly match.
K_μ, K_x	The number of markers in μ and x respectively that have actually been allocated.
u_l	$D \times 1$ coordinate vector of the lth point in the union of two images μ and x.

Sections 4.3, 5.2.1, 5.2.2 and 5.3.2	
n	Number of points in some *true* image.
x	$n \times D$ coordinate matrix for points present in the true image. Also used more generally to indicate the image represented by x respectively.
$\bar{n}$	Number of points in an observed image of the true image.
$\bar{x}$	$\bar{n} \times D$ coordinate matrix for points in the observed image. Also used more generally to indicate the image represented by $\bar{x}$ respectively.
R	Number of replicate images.
r	Number of times a particular point is observed across R images.
p_*	The probability a true point is observed in $\bar{x}$.
λ	The rate of false points per observed image.
v_{rj}	The number of points that are observed r times in the union of R replicate images for $r = 0, \ldots, R$ and $j = 1, \ldots, J_r$. Here J_r is the number of ways of choosing r from R replicate images.
L	The number of sets we have of R replicate images.

Sections 4.4, 5.2.3 and 5.3.3	
L	Number of images in group 1.
R	Number of images in group 2.
$\bar{m}_l, \bar{n}_r$	The total number of non-markers observed in the lth image in group 2 and the rth image in group 2 respectively.
$\bar{\mu}^{(l)}, \bar{x}^{(r)}$	$(K+\bar{m}_l)\times D$ and $(K+\bar{n}_r)\times D$ coordinate matrix for all the points observed in the lth image in group 1 and the rth image in group 2 respectively.
$\hat{p}_{i0}^{l_1 l_2}$	Final estimated posterior probability that the ith point in $\bar{\mu}^{(l_1)}$ is allocated to the coffin bin when $\bar{\mu}^{(l_2)}$ is transformed to fit $\bar{\mu}^{(l_1)}$.
$\hat{q}_{i0}^{l_1 r}$	Final estimated posterior probability that the ith point in $\bar{\mu}^{(l_1)}$ is allocated to the coffin bin when $\bar{x}^{(r)}$ is transformed to fit $\bar{\mu}^{(l_1)}$.
$p_i^{(l)}$	Probability that the ith point in $\bar{\mu}^{(l)}$ is present in all L images in group 1.
$q_i^{(l)}$	Probability that the ith point in $\bar{\mu}^{(l)}$ is present in all R images in group 2.
w	Set as $L/(L+R)$.
$S_i^{(l)}$	$S_i^{(l)} \in \{0,1\}$ where the probability that $\bar{\mu}_i^{(l)}$ is present in group 1 images but absent from group 2 images increases as $S_i^{(l)} \to 1$.

Chapter 6	
k_I	Inhibition constant which is a measure of the binding affinity between a ligand and a protein.
m, n	Number of atoms in ACh and a general pesticide respectively.
μ, x	$m \times 3$ and $n \times 3$ atomic coordinate matrices for ACh and the general pesticide. Also used more generally to indicate the molecule represented by μ or x respectively.
M	$m \times n$ matrix indicating matched atoms across molecules, where an element $M_{ij} = 1$ if x_j is matched to the μ_i for $i = 1, \ldots, K + m$.
L	The number of matched atoms across μ and x.
μ^*, x^*	Coordinate matrices of matched points across μ and x respectively. If $M_{ij} = 1$, then $\mu_l^* = \mu_i$ and $x_l^* = x_j$ for $l = 1, \ldots, L$.
A, b	$D \times D$ rotation matrix and $D \times 1$ translation vector respectively that denote the transformation parameters necessary to superimpose μ onto x.
ζ	Distance tolerance assigned within the graphical matching algorithm.
$y_i^{(k)}$	Toxicity of the ith pesticide when ingested by the kth species.
θ_{ij}	The jth biological descriptor of the ith pesticide.
n_i	Number of atoms in the ith pesticide.
$x^{(i)}$	$n_i \times 3$ atomic coordinate matrices for the ith pesticide in the minimum-energy conformation. Also used more generally to indicate the molecule represented by $x^{(i)}$ respectively.
$\mu^{(1)}$	The $m \times 3$ coordinate matrix of the low-energy conformation of ACh.
$\mu^{(2)}$	The $m \times 3$ coordinate matrix of the docked conformation of ACh.
$x^{(i)*}$	$m \times 3$ matrix containing the matched coordinates in $x^{(i)}$.
θ_{li}^*	Measure of shape similarity between $\mu^{(l)}$ and $x^{(i)*}$.

Chapter 7	
μ_P	4143×3 atomic coordinate matrix for AChE.
n_i	Number of atoms in the ith pesticide.
$x^{(i)}$	$n_i \times 3$ atomic coordinate matrices for the ith pesticide in the minimum-energy conformation. Also used more generally to indicate the molecule represented by $x^{(i)}$.
$\hat{x}^{(il)}$	$n_i \times 3$ atomic coordinate matrices for the lth predicted dock of the ith pesticide.

Chapter 1

Introduction

In this thesis we explore two main themes, both of which involve proteins. The first area of research focuses on the analyses of proteins displayed as spots on 2-dimensional planes. The second area of research focuses on a specific protein and how interactions with this protein can naturally prevent or, in the presence of a pesticide, cause toxicity. Before we discuss the projects in more detail, we first explain the importance of proteins and why continued research is vitally important.

1.1 Why proteins are important

Proteomics is simply the 'study of proteins' with the main focus being on their structure and functions within a biological system.

> 'It is proteins that are directly involved in both normal and disease-associated biochemical processes, a more complete understanding of disease may be gained by looking at the proteins present within a diseased cell or tissue. This forms the basis of proteomics. The potential biological and clinical applications of proteomics are enormous.' [26] [90]

Most drugs exert their effects on proteins and the analysis of proteins has led to crucial developments in the successful diagnosis and treatment of neurological disorders,

infectious diseases, heart disease and cancer to name a few [7]. Research is continually being carried out to locate *biomarkers*, biological indicators, of particular biological states. These biomarkers often occur in the form of proteins. For example, the protein AMACR has been established as an important biomarker of prostate cancer [72] but there is still an urgent need for more accurate biomarkers to improve diagnosis [16].

One way to locate biomarkers of a certain disease is to analyse how proteins differ across control or diseased tissue. How we can do this forms the basis of our first area of research and is discussed in more detail in Section 1.2.

Another way that we can use proteins to gain biochemical information is to explore the reaction that occurs between a drug and a protein on a molecular level. For example, it is the direct reaction between a pesticide and a particular protein that causes toxicity to an organism. It is analyses at the molecular level that forms the basis for our second area of research and is discussed further in Section 1.3.

1.2 Analysis of 2D protein gels

1.2.1 Introduction to 2-Dimensional Electrophoresis and Western Blots

There could be as many as 500,000 proteins in a single human cell [69]. A protein can be uniquely identified by its mass and acidity (or rather, the 'isoelectric point' which is the acidity at which a protein carries no net electrical charge). Two-dimensional electrophoresis (2-DE) is a chemical procedure used to separate proteins by acidity in the first dimension and mass in the second dimension. The result is a 2-dimensional gel (or image of the gel) containing a 'mapping' of all proteins present. If the technology were flawless, the positional information would be enough to uniquely identify each protein. However, further analysis is usually necessary to confirm protein identification. In fact, the development of a protein image is generally the first stage of a multi-step procedure

described in detail by Dowsey *et al.* [26] and summarised in Subsection 1.2.2.

2-DE was first introduced in 1975 by O'Farrell and Klose [61] [46]. Although there has been thorough research over the last 34 years into alternative and more accurate methods of isolating proteins (for example, the 'virtual' 2D images developed by Walker *et al.* [86]), 2-DE remains a core technology for the separation of proteins [63] and is currently the 'workhorse' for proteomics [38]. An example of an image produced by 2-DE is displayed in Figure 1.1. In theory, a particular protein will show up in the form of a black spot at the appropriate location. The red crosses indicate the location of proteins inferred by some analyses system from the image itself. A single image could display over 5000 unique proteins, though routinely they display around 2000 [38].

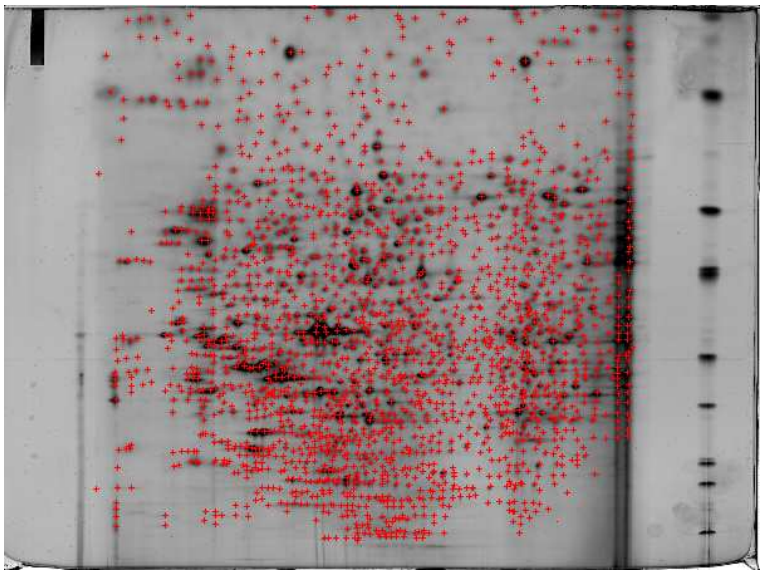


Figure 1.1: An example of a 2-DE protein gel image. The red crosses have been added to the image to indicate the location of proteins inferred by some analyses system from the gel itself.

Although 2-DE is a well-established and well-used technique in protein separation, there are still many problems. Raman *et al.* [64] list gel-running conditions, temperature effects and uneven focusing of equipment as a few factors that effect the quality of a final

image. The continual challenge within this technology is to improve the reproducibility of an image. Currently, two identical protein samples can create very different images though theoretically they should be identical. Reproducibility is difficult within a single laboratory and increasingly more so amongst varying laboratories/equipment/experts [38].

Western blots are gels created to highlight proteins present in human tissue, for example. First, 2-DE is used to separate all the proteins extracted from a cell. The 2-DE gel is then probed with serum which contains antibodies that will bind to specific proteins. The image of a western blot will contain only the location of proteins that have a bound antibody. We can think of western blots as containing only a subset of the proteins that are displayed on 2-DE images. The extra step necessary to create a western blot allows a further level of variability within the final produced image. The reproducibility of western blots is therefore even more challenging than that of 2-DE images. An example of a western blot image is illustrated in Figure 1.2 within Subsection 1.2.4.

Previously, we briefly mentioned the further analyses that generally follows the production of 2-DE or western blot images. Considering the large scope for variation between images and the often vast number of proteins located in a comparatively small area, visual examination to analyse or compare images, although often informative, can be extremely difficult and conclusions unreliable. Visual comparison can also be extremely repetitive and labourious for the expert making the comparisons. Statistical and computational analysis are essential to the *result accuracy* and reduction of expert manual labour.

1.2.2 General analyses of protein gel images

Gorg *et al.* [38] summarise the traditional multistep procedure that follows image production. Here we list the initial steps.

1. Each individual image must be preprocessed, i.e., eliminating background noise to

enhance the image.

2. The exact locations corresponding to unique proteins are highlighted. The spots often have to be segmented before unique spots can be identified.

3. To enable easier comparison across images, an expert will manually locate a number of corresponding proteins, say **markers**, across the images under examination.

4. Using the markers as reference points, the images are warped into superimposition.

5. Further corresponding proteins are highlighted through an automatic matching process, enabling scientists to pick out proteins of interest.

6. The proteins of interest can then be identified using a technique called *mass spectrometry*.

Each of the above steps leave room for error and thorough research has been carried out to refine the procedures involved. Next we discuss examples of previous research into the process described above before discussing the data we have and our particular aims.

1.2.3 Current software and methodology for image analysis

Currently there are various pieces of software commercially available which have been designed specifically to carry out some or all of the stages of analyses described in the previous subsection. In many cases, the software is built upon programs designed in the early years following the development of 2-DE technology and has undergone years of refinement.

First we discuss some of the early packages produced. TYCHO [3] comprises of programs for image acquisition, background subtraction and smoothing, spot detection and modelling, pattern matching and computer comparison. Lemkin and Lipkin [49] [50] [51] have produced multiple papers describing the segmentation of spots, system preliminaries, spot matching techniques and further analyses tools within GELLAB

software. Vincens *et al.* [83] [82] [84] [85] [76] also produced multiple papers describing HERMeS, a software produced to provide similar analyses. They also proposed database organisation and interrogation strategies to allow easy handling of the large quantity of data obtained from a series of gel electrophoresis experiments. Today, dealing with the wealth of data is still a difficult task at the heart of much research. GESA [71] is a system characterised by combining expert intervention alongside automatic analyses techniques.

Today, commercially available packages include CAROL [62]. CAROL was developed to tackle the local distortions that may be present in images and to provide a fully automated point matching technique without the need for corresponding points across images to be manually located by an expert as reference points. The system is also able to provide comparisons of images across the world wide web. WebGel [52] is an exploratory 2-DE gel image and data analysis system involving the tool 'Flicker'. The tool can also be run on the world wide web to help in the comparison of two gel images from similar samples, possibly created in different laboratories, by matching the morphology of local regions. The method is only intended to provide a rough comparison and becomes increasingly difficult to utilise as the number of images being compared increases [48]. Melanie [5] [6] is a popular package which (like many others) integrates filtering, querying, reporting, statistical and graphical options so that you can easily view, compare, analyze and present your results. Other packages include Z3 [73], PDQuest [54] and Progenesis [53].

Multiple reviews have been carried out to compare the accuracy of the different packages available [59] [64] [89] each highlighting varying levels of accuracy over the different stages of analyses. To continue refining the tools involved, many people focus on one particular stage of protein image analysis.

Before points representing proteins can be successfully located, background noise needs to be eliminated. Van Belle *et al.* [81] present a denoising algorithm that adaptively enhances the image contrast and, through thresholding and median filtering, removes the grey-scale range covering the background. Applications are demonstrated on western

blots which are the type of images that instigated this research. After reducing background noise, the next problem faced is how to locate the spots that truly indicate the presence of a protein. The spots on protein images have varying areas and are often irregularly shaped. Rogers *et al.* [69] provide a method to model the shape and appearance of spots automatically generated from a set of real images, thus allowing better definition between single and multiple spots and creating a higher tolerance for irregularly shaped spots. Bettens *et al.* [13] apply a watershed technique for the segmentation of the spots on a protein gel image and the method is demonstrated as superior to commonly used Gaussian models [4]. Cutler *et al.* [24] use a segmentation method involving pixel value collection via serial analysis of the image through its range of density levels.

One of the most difficult tasks involved in the analyses of protein images is how to highlight how the proteins present differ across image. The difficulties arise in the inconsistency of image sizes and the warping that can occur independently across gels. The rough superimposition of even two images is often impossible without computational assistance coupled with the manual location of a selection of reference points, i.e. corresponding points across images. A modification of 2-DE called DIGE [80] is a technique developed to circumvent the problems associated with point matching across protein images. A single image is developed from up to three different samples of protein extracts that have been individually tagged with different coloured fluorescent dye. The production of a single image bypasses the neccessity for image registration and the proteins present in all three samples can easily be highlighted due to the different colours of the three samples. Melanie 7.0 DIGE [32] is a commercially available analyses system for an image output using DIGE technology. However, at the moment only three samples can be compared so DIGE is unable to circumvent the problems associated with superimposition when a greater number of protein samples are being compared, as is often the case.

So the accurate registration of protein images is still vitally important in the exploration of protein correspondences across images. The registration and matching of

point sets is important in a number of different disciplines including shape analysis, image analysis, molecular comparison and even astronomy to name a few. Many techniques have been developed and applied within various fields of study.

Cross and Hancock [23] match geometric structures in 2D point sets by first highlighting point correspondences by maximum *a posteriori* graph-matching and then estimating the transformation necessary for superimposition using an Expectation Maximisation (EM) technique. Chui and Rangarajan [21] propose a general framework for non-rigid point matching by considering thin-plate splines to tackle the problem with an application to the comparison of cortical anatomical structures. Besl and McKay [12] use a iterative closest point (ICP) algorithm to register points sets, curves and surfaces. Belongie *et al.* [8] first infer point correspondence before estimating the registration and describing a shape similarity measure between two objects. To do this a 'shape context' is given to each point which captures the distribution of the remaining points relative to it. Corresponding points across sets will have similar shape contexts, therefore enabling correspondences to be inferred through an optimal assignment problem. Given the correspondences, thin-plate splines are then used to estimate the transformation that best aligns the two objects. Potra *et al.* [63] provide a method to optimally align families of 2-DE gels by constructing an ideal gel to represent the entire family and applying hierachichal piecewise affine transformations. Akutsu *et al.* [1] present a polynomial time algorithm for a special and one-dimensional case of the point matching problem, which is based on dynamic programming. A practical heuristic algorithm for identifying a match between two point sets is also described.

Rohr *et al.* [70] incorporate both point location and intensity to align 2-DE images. Point landmarks are localized using a model fitting scheme and this geometric information is combined with intensity information for elastic image registration. Richmond, Willett and Clark [68] consider Procrustes analysis [28] for molecular comparisons where correspondences are first estimated using image analysis algorithms. Dryden *et al.* [27] consider Bayesian methodology carried out through MCMC simulation to compare two

or more unlabelled point sets. Here the application considered is also in the comparison of 3D molecular structures. A similar technique has been used to explore properties of human movement by labelling points on the body [2].

Walker [87] applies an EM technique to estimate the transformation necessary to superimpose two unlabelled point sets, before providing inference on point matches across sets. This work is extended by Kent *et al.* [77] [45] with applications in protein image comparison and the matching of amino acids within 3D protein structures. Green and Mardia [39] use a different method to explore the same problems associated with matching proteins as points across images or amino acids within protein structures. A Bayesian approach is applied to simultaneously infer the matching and transformation of unlabelled or partially labelled points sets. A Poisson process is assumed to describe hidden true point locations, with EM and MCMC algorithms used to provide inference on unknown parameters. Glasbey and Mardia [34] give a review of possible warping methods that could be utilised for the superimposition of images.

Many point matching techniques require the location of a set of corresponding points across sets, i.e markers. Melanie [5] [6] automatically selects a spot in each of the four corners of an image before locating corresponding points in a second image. These allocated markers are used as fixed reference points in the gel alignment through least-squares minimisation [92]. Flicker [52] requests that the user specify 3 or 6 markers when applying an affine or polygonal transformation respectively to superimpose images. The method developed by Potra *et al.* [63] relies in the initial manual location of a group of markers across images and a threshold is applied to limit the distance allowed between corresponding pairs.

Before we outline our aims within this research, we first introduce the data we have been given.

1.2.4 Data

Introduction

Protein images have been produced to reflect, and allow the comparison of, the proteins present in tissue within controls and renal cancer patients under two possible treatments. Images are created to represent the following four scenarios.

- A control treated with *normoxia*, a normal supply of oxygen.
- A control treated with *hypoxia*, a lowered supply of oxygen.
- A renal cancer patient treated with normoxia.
- A renal cancer patient treated with hypoxia.

The chemical procedure used to create the images is described in the following section.

Chemical implementation

We describe the process in a step by step procedure.

1. Cells from the particular cell line HTB47 are grown in one of two possible treatments.
    - Normoxia.
    - Hypoxia.
2. Protein extracts are taken from the cells.
3. 2-DE is used to create a 2D protein gel by separating the proteins by acidity in the first dimension and by mass in the second dimension.
4. A rectangular membrane is sized and cut to fit the gel.

    Note: The size of the membrane fitted is dependent on the gel.

5. The gel is *probed* with serum from one of eight subjects.

   - Four different controls.
   - Four different patients.

   The term ‘probed’ is used to describe how each antibody within the serum will identify a particular protein within the gel before binding to that particular protein.

6. The binding of an antibody and protein is then detected upon exposure to film. It is this detection process that creates the subject-treatment specific 2D western blot images. We refer to these proteins as **non-markers** throughout the main text.

7. An analysis system (such as those discussed in the previous subsection) is then used to highlight each non-marker as a single cross in the western blot image (see Figure 1.2).

8. To help make image comparison easier and also to create a coordinate system for the mass and acidity of each protein, 12 particular proteins are located. These 12 proteins are present in every gel and have a known mass and acidity. The gel is removed from the membrane and a stain (Coomassie Blue) is applied to the gel to highlight *all* the proteins present within the gel. The markers are then manually located by an expert. These 12 proteins will be referred to as **markers**.

   We consider an image to contain a selection of non-markers and a separate selection of markers.

9. The gel is realigned to the membrane so that the markers are correctly positioned relative to the non-markers before being manually superimposed onto the image as larger crosses (see Figure 1.2).

10. Both the markers and the non-markers are allocated an arbitrary but unique spot ID. In addition, the markers are allocated a marker ID which will indicate

corresponding markers across images. Setting the origin of the image as the top-left corner of the membrane, 2D spatial coordinates are assigned to each marker and non-marker. Using the coordinate system created by the known measurements of the markers, a mass and acidity measurement is also assigned to each non-marker.

Figure 1.2 displays an example of a western blot image within our dataset. In this particular example, the labelled markers 9 and 12 were not successfully located, leaving 10 highlighted markers.

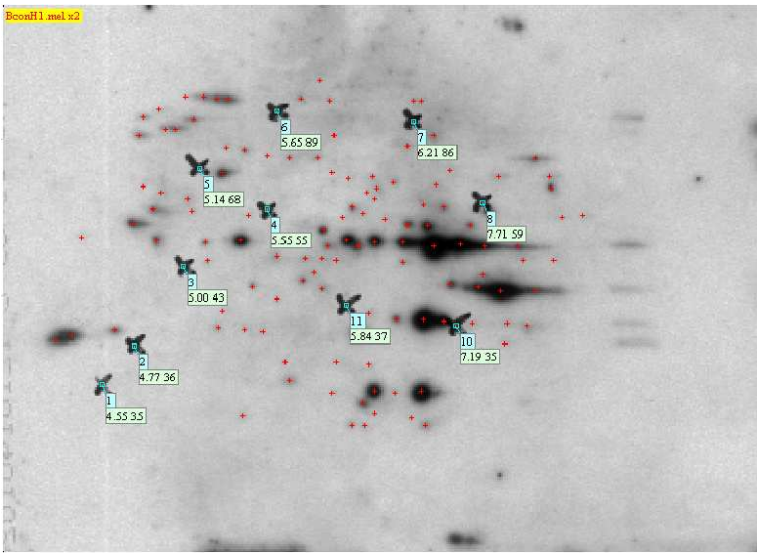


Figure 1.2: Figure displaying a western blot image within our dataset. The red crosses depict the subject-treatment specific non-markers. The larger black crosses indicate the labelled markers, with their acidity and mass measurements highlighted beneath.

Actual dataset

Data is produced to represent eight different subjects (four controls and four patients) treated with two possible treatments. A replicate image is also produced for each subject-treatment specific case. Therefore a full dataset would consist of $8 \times 2 \times 2 = 32$ images.

However, due to production faults such as excess shadowing, six images were removed from our investigation leaving 26 images remaining (indicated in Table 1.1).

Control 1	Initial	Replicate	Patient 1	Initial	Replicate
Normoxia					
Hypoxia					
Control 2	Initial	Replicate	Patient 2	Initial	Replicate
Normoxia		×			×
Hypoxia		×			
Control 3	Initial	Replicate	Patient 3	Initial	Replicate
Normoxia					
Hypoxia					
Control 4	Initial	Replicate	Patient 4	Initial	Replicate
Normoxia					×
Hypoxia		×			×

Table 1.1: Table indicating the 32 images we would have in a full dataset. The crosses highlight the 6 images that are missing from our dataset.

Sources of variability within the data

Possible variation within or between images include the following.

- During the production process, each gel has the freedom to warp independently therefore allowing error in protein location. So positional information of a protein relative to another is likely to vary from image to image. Figure 1.3 displays the 12 markers from two different images after applying Procrustes methodology to superimpose the corresponding markers. None of the corresponding markers

between the two images have been superimposed exactly, indicating location error within the known, labelled markers that will also occur within the non-markers.

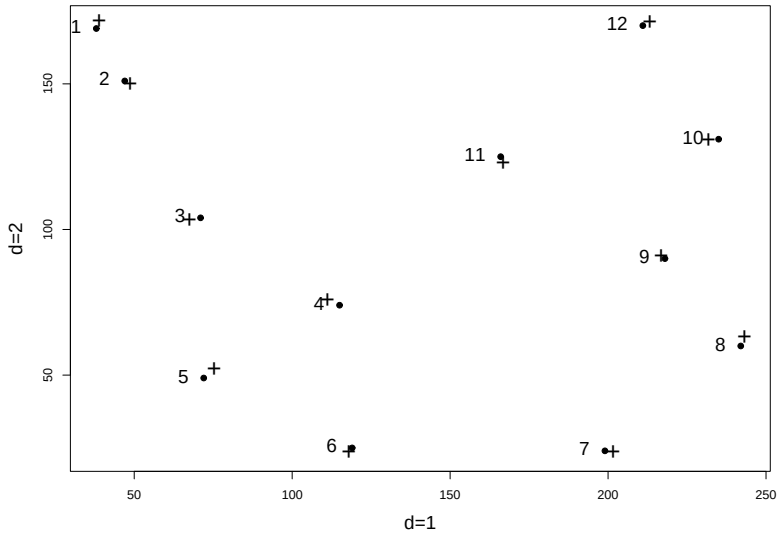
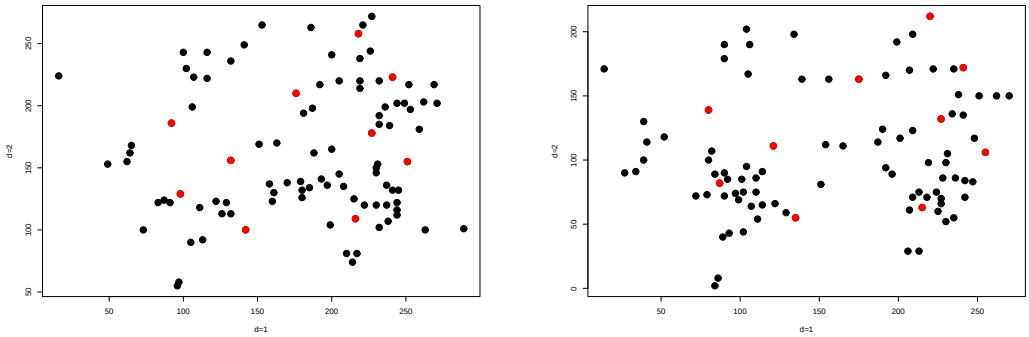


Figure 1.3: The dots and crosses indicate markers from two different images after Procrustes methodology has been applied to superimpose the corresponding markers.

- An increased edge variance. A gel is more vulnerable to warping at the edges of the gel, so variability within protein location is likely to be higher here.
- As can occur with the non-markers, all 12 markers are not always successfully located. For example, markers 9 and 12 have not been located in the image displayed in Figure 1.2.
- A marker can be incorrectly labelled. For instance, a non-marker may be misidentified as a marker or two marker labels could be incorrectly exchanged due to human error.
- It is possible for proteins present within a gel to remain undetected and for dust or shadowing, for example, to be detected as false proteins. We call this *image*

contamination. We can see that contamination is present within the dataset when we compare replicate images. All replicate images should contain the same selection of proteins and therefore the same number of points. Figure 1.4 displays two replicate images. The image in Figure 1.4a contains 99 points whereas the image in Figure 1.4b only contains 93 points.



a) b)

Figure 1.4: a) Initial image of a control treated with hypoxia. b) Replicate image of the same control treated with hypoxia. In both figures, the black dots depict the non-markers and the red dots highlight the markers.

- Subject variability between controls and between patients. For example, the non-markers found in Control A treated with normoxia may be different to those of Control B treated with normoxia.

1.2.5 Aims

In this research, we consider images that have already been pre-processed so that we have a collection of crosses and corresponding coordinates that represent the likely location of unique proteins (as displayed in Figure 1.2). The misalignment of images is a major bottleneck within the analyses of protein images [63] and this is where we focus our attention. We aim to develop a technique that can be used to rank and highlight proteins

that are likely to correspond across pairwise images. We want our metholodology to account for the possibilities of error in point location, an increased edge variance, missing markers and slight or grossly misallocated markers.

We develop a method to create a union of replicate images, which can be used alone in further analyses to reduce computational expense. Inspired by DIGE [80] (an innovative procedure to overcome the problems associated with the gel warping and discussed in more detail in Chapter 4), we develop a technique that can be used to infer the quality of a dataset, i.e., the level of contamination present. Although much work has been spent on matching images, hardly any research (if any) has gone into evaluating the quality of a dataset. Considering the extensive variability found in images across the equipment used, the laboratory conditions and the expert who creates the images, research examining the quality is vital to the relevance of any conclusion formed from a particular dataset.

Note: Many matching techniques have been tested by artificially distorting an image and investigating the matches made under comparison with the original image (for example, [64]). The presence of contamination is often ignored even though it highlights the need for an associated matching probability to locate unique points across groups of images.

Finally, we want to provide a way to rank proteins that are likely to be unique to one group of images. For each point in a group of images, we calculate an associated probability of uniqueness to that group. All pairwise transformations are considered so no information is lost in the allocation of a reference image or creation of a master image.

1.3 Toxicity prediction

1.3.1 Introduction to toxicity

First we introduce the following terminology.

A *protein* is a macromolecule, i.e. a large molecule that generally has some function within a biological system. A *natural ligand*, or *substrate* of a protein, is a small molecule that exists naturally and binds specifically to that protein. A *complex* is a general term given to a bound protein and ligand.

The pesticides that we consider within this research cause *acute oral toxicity* (a measure often used to characterise pesticide toxicity) by inhibiting the protein *acetylcholinesterase* (AChE) from carrying out its natural function. Before we describe how a pesticide causes this inhibition, we first describe the natural cause and prevention of toxicity in the absence of a pesticide.

Natural cause and prevention of toxicity

Figure 1.5 helps to visualise the natural cause and prevention of acute oral toxicity. Impulses are continually emitted from nerve cell endings within the biological system of an organism. Molecules of AChE exist in the gap dividing a nerve cell from a muscle. Molecules of *acetylcholine* (ACh), the substrate of AChE, are continually released into the same gap.

The presence of ACh allows the impulses to travel from a nerve cell to a muscle. This occurs because ACh is a *neurotransmitter*, which means it has the ability to relay and amplify the impulses. The impulses stimulate muscle contractions and it is this process that is the *natural* cause of toxicity and can eventually lead to the death of the organism.

The primary function of AChE is to break down its substrate, ACh, into smaller molecules. Therefore AChE removes molecules of ACh from the gap dividing a nerve from a muscle and the impulses cannot be transmitted across. The reaction that occurs between an AChE molecule and an ACh molecule can be summarised as follows.

1. ACh *docks* at a specific location on AChE called the *binding site*. (The term ‘dock’ is used to describe how a smaller molecule binds to a macromolecule.)

2. The complex formed by AChE and ACh is particularly unstable, leaving ACh

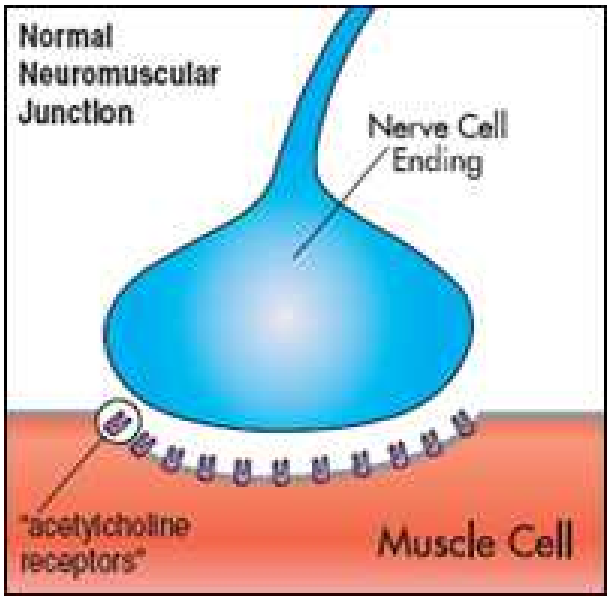


Figure 1.5: Figure to highlight the natural cause and prevention of acute oral toxicity.

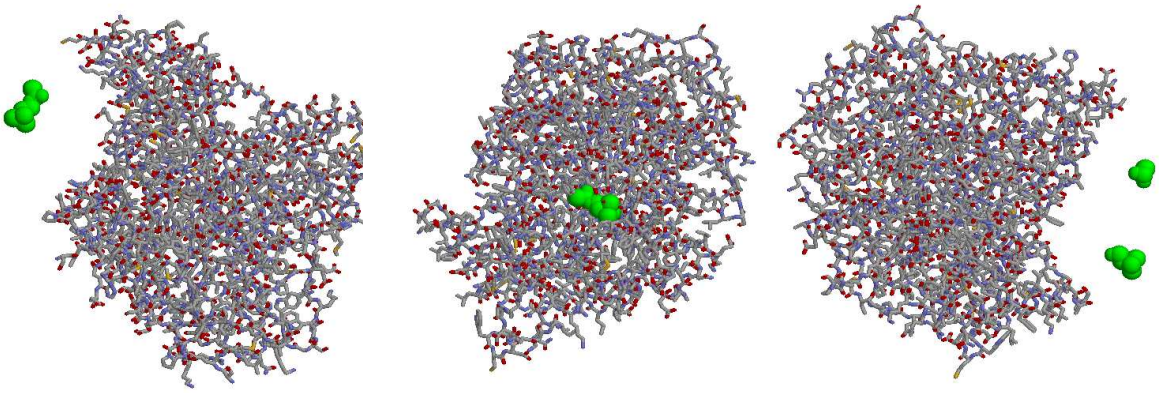
vulnerable to hydrolysis, i.e. a reaction with water. The hydrolysis of ACh breaks the substrate down into two smaller molecules, acetic acid and choline.

3. The two smaller molecules then leave the binding site of AChE, leaving AChE molecularly unaltered.

Figure 1.6 displays the three steps described above. After the third step, the molecularly unaltered AChE is then *reactivated*, that is, it is able to bind with further molecules of ACh so that the process can be continually repeated. Only the full ACh molecule acts as a neurotransmitter. The two smaller molecules released by AChE are unable to transmit the impulses and toxic consequences are naturally avoided.

Competitive inhibition by a pesticide

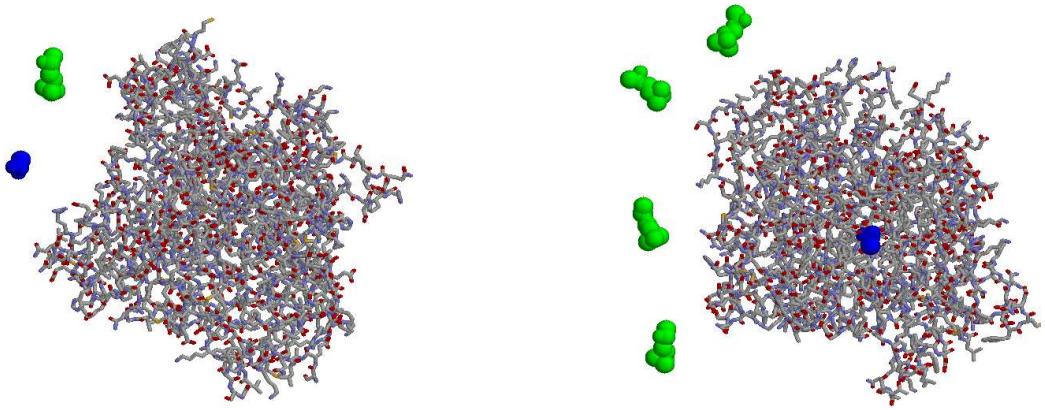
Pesticides are synthetic ligands designed specifically to dock at the same binding site on AChE to which ACh would dock. The term ‘competitive’ in *competitive inhibition* describes the competition between a pesticide and ACh to bind with AChE. If a pesticide



a) b) c)

Figure 1.6: The macromolecule is the protein, AChE, and the smaller molecule is the natural ligand, ACh, which is highlighted in green. The structure of AChE is represented as sticks and each atom in ACh as a sphere to make the smaller molecule visually clearer in respect to the larger molecule. Figure a) displays an AChE and ACh molecule before they bind. Figure b) displays the complex formed by the bound AChE and ACh. Figure c) displays AChE and the two smaller molecules, acetic acid and choline, formed by breaking down ACh.

binds with AChE, AChE is then ‘inhibited’ from binding with ACh and cannot carry out its primary function to safely break down the substrate. A build-up of ACh molecules will take place, allowing impulses to be transmitted to the muscles of an organism which result in toxic effects. Figure 1.7 displays the inhibition of AChE by an example pesticide, sarin.



a) b)

Figure 1.7: The macromolecule is the protein, AChE, and the structure of AChE is represented as sticks. The smaller molecules are ACh and the pesticide, sarin. Both structures are represented by atomic spheres and are highlighted in green and blue respectively. Figure a) displays AChE, ACh and sarin before a reaction has taken place. Figure b) displays the complex formed by the bound AChE and sarin, surrounded by molecules of ACh. AChE has been blocked by sarin and is now unable to bind with the substrate, ACh, before safely breaking it down.

Within this research, we consider two families of pesticides called *carbamates* and *organophosphates* (OPs), both of which cause toxicity in the way described above. We describe the general structures of a carbamate and an organophosphate alongside the structure of ACh in Chapter 6. We also give a more detailed description of the reaction that occurs between each ligand and AChE.

1.3.2 Current methods to predict pesticide toxicity

Drugs generally exhibit pharmaceutical activity by binding to a target protein [56]. Developing new techniques to more accurately predict the scale of activity induced by the binding of a drug and a protein is crucial to the increased understanding of the effects of current drugs as well as the development of new and more effective drugs. Similarly, pesticides exert toxicity by binding to the protein, AChE, and the ability to accurately predict their potential toxicity is of paramount importance.

Pesticide toxicity is simply the degree to which a pesticide is toxic. One way to measure the potential toxicity of a pesticide to a given species is *in vivo* by calculating the associated *Lethal Dose 50* (LD50), which is the amount of pesticide necessary to kill 50% of a sample of the species. Here the resulting measure takes into account the absorption, distribution, metabolism and excretion (ADME) of a pesticide within the system of a pest, which plays an important role in determining pesticide toxicity.

However, the potential toxicity of OPs [and carbamates] to a species is *largely* dependent on the inhibition of AChE [30]. Although many techniques have been developed, no general or reliable approach to predict the AChE inhibitory activity of new inhibitors has yet been established [11]. Because ligands will bind themselves inside AChE rather than simply on the surface of AChE, a thorough knowledge of how the ligand and AChE will bind is essential to deriving an accurate predictive model [11].

Alternative to the calculation of LD50, the reaction between AChE and a pesticide can be carried out *in vitro*. Through this experimentation we can calculate the *binding affinity*, which is a major determinant of the toxic potency of a pesticide [30]. The binding affinity of a pesticide with AChE can be measured by the *inhibition constant*, k_I. The inhibition constant is related to the *half maximal inhibitory concentration* (IC50) by the equation

$$k_I = \frac{\text{IC50}}{1 + \frac{S}{k_M}},$$

where S is the concentration of the substrate, ACh, and k_M is the affinity of ACh for

AChE. The IC50 is the amount of a pesticide necessary to inhibit 50% of the AChE molecules.

To avoid the ethical implications associated with animal testing or both the time and costs associated with *in vitro* experimentation, much research has been spent on developing more accurate toxicity predictions *in silico*.

A common technique used to predict toxicity *in silico* is via Quantitative Structure Activity Relationships (QSAR). The QSAR paradigm related to this project is that the toxicity of a pesticide is proportional to one or more properties of the pesticide molecule itself. This approach allows important molecular properties to be identified and then used within a suitable model to predict toxicity. Alternative to the classic QSAR methods, 3D-QSAR approaches are considered better suited to describe the activity resulting from ligand-receptor interactions as they consider the properties of a ligand in their (supposed) bioactive conformation [91], which in our case would be the docked conformation within AChE. In the case of 3D-QSAR, knowledge about how a pesticide will bind with AChE is assumed known.

One way to predict how AChE and a pesticide will bind is by implementing *computational molecular docking*. Generally a docking program will produce multiple predictions of how an input pesticide will bind to a protein and the predictions should converge to the 'true' dock. For each prediction a measure of binding affinity, such as the inhibition constant, is often also estimated and used to highlight the most likely representative of the true dock. Many different docking programs have been developed, though there are drawbacks associated with each docking strategy [41].

Finally, shape plays a crucial role in understanding protein-structure function relations [58]. Although shape is ill-defined in molecular biology [58] (most likely due to the difficulties associated with defining shape amidst molecular flexibility), Cosgrove *et al.* [22] state that it has been established that tightly binding ligands [high affinity ligands] have a high degree of shape complementarity with their receptor. Though analysis based on shape requires something close to the functionally relevant shapes to start with [58].

Next we briefly discuss previous work before introducing our project aims.

1.3.3 Previous work

Doorn *et al.* [25] bind the OP, isomalathion, with AChE *in vitro* to evaluate the products of the reaction. Richardson *et al.* [67] examine the toxicity of the OP, chlorpyrifos, to hens *in vitro* and *in vivo*, by calculating the inhibition constant, k_I, and the LD50 respectively. Halle and Göres [40] found a positive correlation (significant at the 95% confidence level) between IC50 and LD50 toxicity.

Recanatini *et al.* [66] carried out comparitive QSAR analysis to highlight the properties of AChE inhibitors which are essential to potential drugs for the treatment of Alzheimers. Both El Yazal *et al.* [30] and Zhao *et al.* [91] use 3D-QSAR to enable the prediction of neurotoxicity via the inhibition of AChE. The main objective for DEMETRA - Development of Environmental Modules for the Evaluation of Toxicity of pesticide Residues in Agriculture - is to produce QSAR software for pesticide toxicity prediction. Previous research has found a correlation between the inhibition of AChE and acute neurotoxicity [30].

Chen and Ung use a ligand-protein inverse docking approach to facilitate toxicity prediction [20]. Bursulaya *et al.* [18] give a detailed comparison of multiple docking programs and Halperin *et al.* [41] give an overview of the search algorithms and scoring functions involved.

Morris *et al.* [58] present a method to describe the shape of a protein binding site in terms of spherical coordinates. Cosgrove *et al.* [22] provide a method that detects local shape similarity which correctly identified the binding of 20 out of 21 particular inhibitors using shape alone. Good and Richards [37] give a review on methods developed to calculate 3D shape similarity between molecules.

1.3.4 Data

The data we have been given is summarised as follows.

1. Atomic coordinate data for 145 pesticides (39 carbamates and 106 OPs) in minimum-energy conformations (discussed in more detail in Chapter 6).

2. Over a thousand biological descriptors for each of the 145 pesticides.

3. For varying subsets of the pesticides, we have LD50 toxicity data for 5 different species: bobwhite quails, japanese quails, mallards, red-winged blackbirds and starlings. Table 1.2 displays the number of pesticides for which we have toxicity data for each species.

	Number of pesticides for which we have toxicity data		
Species	Carbamates	OPs	Total
Bobwhite quail	17	35	52
Japanese quail	18	49	67
Mallard	15	47	62
Red-winged blackbird	25	60	85
Starling	18	54	72

Table 1.2: Table displaying the number of pesticides for which we have toxicity data for each species.

1.3.5 Aims

We want to develop a shape similarity measure between ACh and a pesticide. As both ligands bind to the same site within AChE, the shape similarity between them may be an indicator of the associated pesticide toxicity. We compare the significance of the

produced shape similarity measure as a toxicity predictor to the significance of biological descriptors which have previously been highlighted as indicators of toxicity.

According to 3D-QSAR, the docked pesticide is a more suitable indicator of toxicity. We use a docking program to predict the dock of a pesticide to AChE. We then explore whether the similarity with the docked ACh, the closeness of fit to AChE and the output inhibition constant of the prediction help determine the potential toxicity.

1.4 Thesis structure

In Chapter 2 we build on the EM algorithm introduced by Walker [87] and extended by Kent *et al.* [77] [45]. We provide methodology to infer one-to-one, many-to-one or many-to-many matches of points across images. The latter types of matching are useful when comparing protein images as multiple forms of an individual protein can often be visualised [7]. We also provide a method to account for the likelihood of an increased edge variance within images.

Most current computational analyses systems rely on the manual location of a set of markers and any mismatches must be checked and edited manually by an expert [38]. The misidentification of a marker can mislead even the most elegant analyses system when estimating the superimposition of images. In this work we introduce a prior that will account for the possibility that a true marker is actually a nearby point of the allocated marker. Incorporating this prior deals with the possibility of slight marker misallocation within a warped image so that matching should not be greatly affected by slight misallocations. The EM algorithm is strongly dependent on the starting transformation which would, intuitively, be estimated from the corresponding markers. Inputting the spot IDs of markers is a manual procedure and could lead to gross positional misallocations even if there were only a slight input error. We produce a technique to automatically locate and remove markers that are highlighted as gross misallocations, before the remaining markers are used to infer starting transformations

in further comparitive analyses. Finally, we provide methodology to deal with the high likelihood of missing markers within an image.

The methodology developed in Chapter 2 can not only assign matches, but also calculates an associated matching probability. This supplies a scientist with further information and the ability to pick the most likely match or non-match (depending on what is of particular interest) for further investigation.

In Chapter 3, we explore the accuracy of the methodology introduced in Chapter 2 and use it to analyse the given data. We compare the matches inferred when we fix the prior probability of markers matching as one, to the matches inferred when we employ the prior that will account for the possibility of slight marker misallocation. We highlight appropriate parameters that should be used within further analyses of the given dataset. We explore evidence of an increased edge variance within our dataset before finally including an example of how points are matched across two images.

In Chapter 4 we first show how data can be pooled across replicate images to minimise the input into further analyses. We develop a technique that can be used to infer the quality of a dataset, i.e., the level of contamination present. Finally, we show how the EM algorithm can be used to highlight likely points unique to a specific group of images.

In Chapter 5, we explore the accuracy of the methodology introduced in Chapter 4 and use it to analyse the given data. We provide an example of how a single union image can be created to represent two replicate images. We explore the level of contamination present within the given dataset. Finally, we rank the points (or proteins) that are likely to be unique to certain groups of images within the dataset.

In Chapter 6 we test the hypothesis that the potential toxicity of a pesticide is related to the shape similarity between the pesticide and the substrate, ACh, of the protein, AChE, to which they both bind. We produce methodology to calculate a measure of shape similarity between ACh and a pesticide. We then explore the significance of the developed shape similarity measure as a toxicity predictor and compare it to the significance of

known biological indicators of toxicity. We also compare the accuracy of the toxicity predictions when applying our model to the accuracy when implementing a previously developed online predictor.

In Chapter 7 we explore the accuracy of a docking program before using it to predict the docked conformation of a pesticide within AChE. We then produce a measure of similarity between the known dock of ACh and the predicted pesticide docks. We also define a method to calculate a distance measure between a docked ligand and AChE. We investigate the significance of these measures, alongside an associated inhibition constant, as toxicity predictors for the bobwhite quail.

In Chapter 8 we provide a critical summary of the research within this thesis before finally highlighting possible further work in each area.

Chapter 2

Modelling, and using the EM algorithm to match, pairwise gels

2.1 Introduction

In Section 2.2 we introduce a statistical model to represent data across pairwise images. We consider two possible methods to calculate prior matching probabilities across images. The first method assumes that a *true* marker is always correctly *allocated*. The second method deals with the possibility of slight marker misallocation within a warped image and does not assume that an allocated marker is always the true marker. In Section 2.3 we use an Expectation Maximisation (EM) algorithm to estimate the superimposition of two images before inference is made on point correspondence across images. Finally, in Section 2.4 we provide methodology to account for missing or grossly misallocated markers.

In this chapter we assume that all points observed in an image represent real proteins.

2.2 Introduction to the statistical model

As the mass and acidity of a protein are calculated from the spatial coordinates, we focus only on protein coordinates within the described statistical model.

2.2.1 Notation

We introduce a statistical model for data within a general D dimensions. (Within figures, d denotes the dth dimension.)

Let μ_G and x_G be $m_G \times D$ and $n_G \times D$ matrices containing the coordinates for *all* the proteins present in two 2-DE gels. Let μ and x be the $(K+m) \times D$ and $(K+n) \times D$ subsets of μ_G and x_G observed in western blot images of the gels, where μ_i and x_j are $D \times 1$ vectors containing the coordinates of point i in μ and point j in x respectively. Let μ_i and x_j contain the coordinates of marker k for $1 \leq i, j, \leq K$ and the arbitrarily labelled coordinates of the m and n non-markers for $i = K+1, \ldots, K+m$ and $j = K+1, \ldots, K+n$ in μ and x respectively. The $D \times 1$ coordinate vectors in μ_G and x_G are set as $\mu_i^G = \mu_i$ and $x_j^G = x_j$ for $i = 1, \ldots, K+m$ and $j = 1, \ldots, K+n$ respectively. For $i = K+m+1, \ldots, m_G$ and $j = K+n+1, \ldots, n_G$, μ_i^G and x_j^G respectively contain coordinate information for arbitrarily labelled proteins that have not been observed in μ and x.

So the matrices μ_G and x_G contain coordinate information for all the proteins present in the 2-DE gels and are independent of the subject. In our case μ_G and x_G represent theoretical gel images as we have data for the western blots only (see chemical implementation in Subsection 1.2.4) and are only considered when simulating data in the following chapter to mimic the allocation of markers. The matrices μ and x contain coordinate information for the K markers and the m or n subject-treatment specific non-markers respectively. Both μ and x represent observed images and we assume they each contain a selection of markers and a separate selection of non-markers.

2.2.2 Transformations

To enable us to highlight points that are present in both images, we first aim to superimpose μ onto x.

Although the statistical model we later introduce can apply to various types of transformations, we focus on an affine transformation of the form

$$g(\mu) = \mu A^T + B^T,$$

where A is a non-singular $D \times D$ matrix and the $D \times 1$ vector, b, is present in every column of the $D \times (K + m)$ matrix B. Due to the possibility of differential stretching between the rows and columns found in images (because of the warping incurred by the gel), Horgan *et al.* [42] consider the affine transformation to be a suitable transformation when superimposing images. We want to estimate the affine transformation parameters, A and b, that superimpose μ onto x.

2.2.3 Matching matrix

To enable us to estimate the appropriate transformation of μ, we can introduce a labelling system that will indicate whether a point in μ corresponds to a point in x, i.e., whether two points *match* across configurations.

We can record the labelling information in a $(K + m + 1) \times (K + n)$ matching matrix, M, where

$$M_{ij} = \begin{cases} 1 & \text{for } i = 0 \text{ if } x_j \text{ does not have a matching point in } \mu \\ 1 & \text{for } i = 1, \ldots, K + m \text{ if } x_j \text{ matches } \mu_i \\ 0 & \text{otherwise} \end{cases},$$

for $j = 1, \ldots, K + n$. Note that, for simplicity of notation, we set $M_{0j} = M_{ij}$ for $i = K + m + 1$. If $M_{0j} = 0$, then x_j does not have a matching point in μ and we say that x_j is allocated to the *coffin bin*.

We consider one-to-one or many-to-one matches between points in x and points in μ. We refer to these as *soft* and *hard* matches respectively.

Hard Matches

The matching matrix, M, has the following constraints for the hard model.

$$\sum_{i=0}^{K+m} M_{ij} = 1 \text{ for } j = 1, \ldots, K+n \tag{2.1}$$

and

$$\sum_{j=1}^{K+n} M_{ij} <= 1 \text{ for } i = 1, \ldots, K+m. \tag{2.2}$$

Here the points in μ are chosen *without replacement*. So for $i_1 \neq 0$, if $M_{i_1 j_1} = 1$, i.e. μ_{i_1} is matched to x_{j_1}, then $M_{i_1 j_2} = M_{i_2 j_1} = 0$ for all $i_1 \neq i_2$ and $j_1 \neq j_2$.

Note that there are no constraints on row $K+m+1$ in M since each of the $K+n$ points in x is free to be allocated to the coffin bin.

Soft Matches

For the soft model, the only constraint is stated in Equation (2.1). Here the points in μ are chosen *with replacement*. That is, if $M_{i_1 j_1} = 1$ then $M_{i_2 j_1} = 0$ for all $i_1 \neq i_2$, but $M_{i_1 j_2} \in \{0, 1\}$ for $j_1 \neq j_2$.

When assigning either hard or soft matches, Equation (2.1) constrains a point in x to be matched to a single point in μ or, alternatively, to be allocated to the coffin bin.

To allow for the possibility of soft matching, we consider points in x to be independent. As we have K markers in each image, we have prior information about the matching across images. Next we introduce notation to deal with prior matching probabilities.

2.2.4 Prior matching matrix probabilities

Let Q be a $(K+m+1) \times (K+n)$ matrix where an element $q_{ij} = p(M_{ij} = 1)$. That is, for $j = 1, \ldots, K+n$, q_{ij} is the prior probability that μ_i is matched to x_j for

$i = 1, \ldots, K + m$ and the prior probability that x_j is allocated to the coffin bin for $i = 0$. Again, for simplicity of notation we have set $q_{0j} = q_{ij}$ for $i = K + m + 1$.

As the labelling is independent over points in x,

$$\sum_{i=0}^{K+m} q_{ij} = 1 \text{ for } j = 1, \ldots, K + n.$$

We have prior knowledge that corresponding markers, μ_k and x_k for $k = 1, \ldots, K$, *should* match.

We introduce both a *standard* and *adapted* method to assign q_{ij} for $i = 0, \ldots, K+m$ and $j = 1, \ldots, K + n$. The standard method assumes that the allocated markers are the true markers, i.e., that corresponding markers will match across configurations. The adapted method deals with the possibility of slight error when allocating markers within a warped image and does not assume prior knowledge that corresponding markers will match.

Standard method:

In this case we assume that an allocated marker k is the true marker k for $k = 1, \ldots, K$.

Markers in x

Because we assume that each marker is correctly allocated, we set

$$q_{ik} = \begin{cases} 1 & \text{if } i = k \\ 0 & \text{if } i \neq k \end{cases}, \tag{2.3}$$

for $i = 0, \ldots, K + m$ and $k = 1, \ldots, K$, where q_{kk} denotes the prior probability that correspondingly allocated markers match.

Non-markers in x

For a non-marker x_j, $j = K + 1, \ldots, K + n$, we set

$$q_{ij} = \begin{cases} 0 & \text{for } i = 1, \ldots, K \\ \frac{1}{m+1} & \text{for } i = 0 \text{ and } i = K + 1, \ldots, K + m \end{cases}. \tag{2.4}$$

We know that the K markers in μ have specified corresponding points in x. So the prior matching probability of a non-marker x_j is set to be uniform over the $m+1$ remaining matching possibilities.

Note: We have chosen to set the probability of x_j being allocated to the coffin bin to be equal to the probability of it being matched to a point in μ. The more points in μ, the more likely it is that x_j has a corresponding point in μ. So setting the prior probability to be inversely related to $m+1$ seems sensible.

Adapted method:

Here we allow for error in the allocation of a marker within a warped configuration and consider the possibility that an allocated marker k may not be the true marker k.

Markers in x

We know that μ_k contains the allocated marker coordinates for marker k in μ, $k = 1, \ldots, K$. Let γ_k be the index of the true marker k in μ. If $\gamma_k = k$, then the true marker k has been correctly allocated as marker k.

We set the prior probability of a point μ_i being the true marker k, q_{ik}, to be a function of the distance between μ_i and μ_k so that

$$q_{ik} = p(\gamma_k = i) = f(d_{ik}) \qquad \text{for } i = 1, \ldots, K+m, \tag{2.5}$$

where d_{ik} is the Euclidean distance between μ_i and μ_k, i.e.,

$$d_{ik} = \|\mu_i - \mu_k\|. \tag{2.6}$$

Possible choices for f are discussed in Section 2.3.5.

Next we consider the possibility that a marker within x does not have a corresponding point in μ. We know that x_k contains the allocated marker coordinates for marker k in x, $k = 1, \ldots, K$. To allow the possibility for x_k to be allocated to the coffin bin, we set the prior probability of $M_{0j} = 1$ to be uniform so that

$$q_{0k} = p(\gamma_k = i) = \frac{1}{|\Omega|}, \tag{2.7}$$

where Ω is some region in $\mathbb{R}^D$ containing all points in x.

Non-markers in x

To allow for the possibility that an allocated marker k is not the true marker k in x, for $k = 1, \ldots, K$, we can set

$$q_{ij} = \frac{1}{K + m + 1}, \tag{2.8}$$

for $i = 0, \ldots, K + m$ and $j = K + 1, \ldots, K + n$. So the prior matching probability of a non-marker x_j is set to be uniform over the $K + m + 1$ matching possibilities.

2.2.5 Error distribution

Assuming the transformation parameters, A and b, are known, we can apply a distribution to x_j given the match $M_{ij} = 1$. We treat the elements of x as conditionally independent with the following distributions for $j = 1, \ldots, K + n$.

$$x_j | M_{ij} = 1 \sim \begin{cases} N_D(A\mu_i + b, \sigma_{ij}^2 I_D) & \text{for } i = 1, \ldots, K + m \\ \text{Unif}(\Omega) & \text{for } i = 0 \end{cases},$$

where $2\sigma_{ij}^2$ is an assigned variance between μ_i and x_j (assuming independence across dimensions), and Ω is again some region in $\mathbb{R}^D$ containing all points in x. So the pdf of x_j given the match $M_{ij} = 1$ is

$$p(x_j | M_{ij} = 1) = \begin{cases} \frac{1}{(2\pi\sigma_{ij}^2)^{D/2}} \exp\left\{ -\frac{\|x_j - A\mu_i - b\|^2}{2\sigma_{ij}^2} \right\} & \text{for } i = 1, \ldots, K + m \\ \frac{1}{|\Omega|} & \text{for } i = 0 \end{cases}. \tag{2.9}$$

2.3 Estimating the parameters within the statistical model

2.3.1 Inference on the matching matrix assuming the transformation is known

In the simplest case, the variance, σ_{ij}^2, and the transformation parameters, A and b, are known. The expected log-likelihood of the matching matrix, M, given the data, x, takes the form

$$\begin{aligned} E[l(M|x)] &= \sum_{i=0}^{K+m}\sum_{j=1}^{K+n} M_{ij}\log p(x_j|M_{ij}=1) \\ &= \sum_{j=1}^{K+n}\left\{\sum_{i=1}^{K+m} M_{ij}\left[-\frac{\|x_j - A\mu_i - b\|^2}{2\sigma_{ij}^2} - \frac{D}{2}\log(2\pi\sigma_{ij}^2)\right] - M_{0j}\log|\Omega|\right\} \\ &= -\frac{1}{2}\sum_{j=1}^{K+n}\left\{\sum_{i=1}^{K+m}\left[\frac{M_{ij}}{\sigma_{ij}^2}\|x_j - A\mu_i - b\|^2 + D\log(\sigma_{ij}^2)\right] + \alpha M_{0j}\right\} + c, \end{aligned} \tag{2.10}$$

where $\alpha = 2\log(|\Omega|/(2\pi)^{D/2})$ and $c = -((K+n)D/2)\log(2\pi)$ when incorporating the constraint that applies to both the hard and soft model in Equation (2.1).

However, in reality it is unlikely that the transformation parameters are known. In the next section we show how the EM algorithm can be implemented to estimate the transformation parameters, A and b, before inferring on the matching matrix, M.

2.3.2 Estimating the transformation parameters via the EM algorithm

We use an EM algorithm to estimate the transformation parameters, A and b, that will superimpose μ onto x. In the E-step we calculate the posterior probability that μ_i matches x_j, i.e. the posterior probability that $M_{ij} = 1$. The posterior probabilities are then input

into the expected likelihood of observing the matching matrix, M, given the data, x. In the M-step we estimate the transformation parameters, A and b, that maximise the expected likelihood found in the E-step.

E-step

We calculate the posterior probability of μ_i matching x_j, i.e., $M_{ij} = 1$, given x_j using Bayes Theorem so that

$$p(M_{ij} = 1|x_j) = \frac{p(x_j|M_{ij} = 1)p(M_{ij} = 1)}{p(x_j)}, \tag{2.11}$$

where $p(x_j|M_{ij} = 1)$ is calculated using Equation (2.9). The second term in the numerator of Equation (2.11) is $q_{ij} = p(M_{ij} = 1)$ and is calculated using both Equations (2.3) and (2.4) in the standard method or Equations (2.5), (2.7) and (2.8) in the adapted method. The denominator of Equation (2.11) is calculated as

$$p(x_j) = \sum_{i=0}^{K+m} p(x_j|M_{ij} = 1)p(M_{ij} = 1) = \sum_{i=0}^{K+m} q_{ij}p(x_j|M_{ij} = 1).$$

Replacing M_{ij} and $p(x_j|M_{ij} = 1)$ in Equation (2.10) with p_{ji} and $q_{ij}p(x_j|M_{ij} = 1)$ respectively, the expected log-likelihood of observing the matching matrix, M, given the data, x, becomes

$$E[l(M|x)] = \sum_{i=0}^{K+m} \sum_{j=1}^{K+n} p_{ji} \left[\log q_{ij} + \log p(x_j|M_{ij} = 1)\right], \tag{2.12}$$

where $p_{ji} = p(M_{ij} = 1|x_j)$ for simplicity of notation.

M-step

In this step we want to estimate the transformation parameters, A and b, that maximise the expected log-likelihood displayed in Equation (2.12). Both the prior probabilities stored in Q and the conditional distribution of x_j being allocated to the coffin bin are independent

of A and b, so we estimate the transformation parameters that maximise

$$\sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p_{ji}\log p(x_j|M_{ij}=1) = \sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p_{ji}\left[-\frac{\|x_j - A\mu_i - b\|^2}{2\sigma_{ij}^2} - \frac{D}{2}\log(2\pi\sigma_{ij}^2)\right].$$

Removing further terms independent of A and b, we want to estimate the transformation parameters that minimise

$$\sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p^*_{ji}\|x_j - A\mu_i - b\|^2$$

$$= \sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p^*_{ji}\left[\|x_j\|^2 - 2x_j^T(A\mu_i) - 2x_j^T b + \|A\mu_i\|^2 + 2(A\mu_i)^T b + \|b\|^2\right], \qquad (2.13)$$

where

$$p^*_{ji} = \frac{p_{ji}}{\sigma_{ij}^2}.$$

Ignoring the terms independent of b and applying the properties

$$\frac{\partial a^T x}{\partial x} = a \qquad \text{and} \qquad \frac{\partial x^T x}{\partial x} = 2x,$$

the differential of Equation (2.13) with respect to b becomes

$$\sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p^*_{ji}(2b - 2x_j + 2A\mu_i).$$

Setting to zero, the maximum likelihood estimate of b, $\hat{b}$, is

$$\hat{b} = \frac{\sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p^*_{ji}(x_j - A\mu_i)}{\sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p^*_{ji}}. \qquad (2.14)$$

Substituting the mle of b, $\hat{b}$, back into Equation (2.13), we find that

$$\sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p^*_{ji}\|x_j - A\mu_i - (\bar{x} - A\bar{\mu})\|^2$$

$$= \sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p^*_{ji}\|(x_j - \bar{x}) - A(\mu_i - \bar{\mu})\|^2$$

$$= \sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p^*_{ji}(\|x_j - \bar{x}\|^2 - 2(x_j - \bar{x})^T A(\mu_i - \bar{\mu}) + \|A(\mu_i - \bar{\mu})\|^2), \qquad (2.15)$$

where

$$\bar{\mu} = \frac{\sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p^*_{ji}\mu_i}{\sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p^*_{ji}} \qquad \text{and} \qquad \bar{x} = \frac{\sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p^*_{ji}x_j}{\sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p^*_{ji}}.$$

Ignoring the terms independent of A and applying the properties

$$\frac{\partial a^T X b}{\partial X} = ab^T, \qquad \text{and} \qquad \frac{\partial a^T X^T X b}{\partial X} = X(ab^T + ba^T)$$

the differential of Equation (2.15) with respect to A becomes

$$\sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p^*_{ji}\left[-2(x_j - \bar{x})(\mu_i - \bar{\mu})^T + 2A(\mu_i - \bar{\mu})(\mu_i - \bar{\mu})^T\right]$$

Setting to zero, the maximum likelihood estimate of A, $\hat{A}$, is

$$\hat{A} = \left[\sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p^*_{ji}(x_j - \bar{x})(\mu_i - \bar{\mu})^T\right]\left[\sum_{i=1}^{K+m}\sum_{j=1}^{K+n} p^*_{ji}(\mu_i - \bar{\mu})(\mu_i - \bar{\mu})^T\right]^{-1} \qquad (2.16)$$

These mles for both $\hat{A}$ and $\hat{b}$ were given by Walker [87].

The algorithm alternates between the E-step and the M-step. At each iteration, the transformation parameters are updated in the M-step to

$$A^{(r+1)} = \hat{A}^{(r)} \text{ and } b^{(r+1)} = \hat{b}^{(r)},$$

before being input back into the E-step for the next iteration.

Convergence

We assign convergence to be when r is such that

$$\frac{1}{(K+m+1)(K+n)}\sum_{i=0}^{K+m}\sum_{j=1}^{K+n}\left[p^{(r+1)}_{ji} - p^{(r)}_{ji}\right]^2 \leq 1 \times 10^{-l}, \qquad (2.17)$$

where l can be varied and the posterior probability of μ_i matching x_j at the rth and $(r+1)$st iteration is denoted by $p^{(r)}_{ji}$ and $p^{(r+1)}_{ji}$ respectively, for $i = 0, \ldots, K+m$ and $j = 1, \ldots, K+n$. The larger l, the closer the average squared difference must be between $p^{(r)}_{ji}$ and $p^{(r+1)}_{ji}$ at the final iteration $r+1$.

2.3.3 Inference on the matching matrix using estimated transformation parameters

Let $\hat{p}$ be the $(K+n) \times (K+m+1)$ matrix containing the final posterior matching probabilities. Let $\hat{A}$ and $\hat{b}$ be the final maximum likelihood estimates of the transformation parameters output by the EM algorithm. These mles, $\hat{A}$ and $\hat{b}$, provide the transformation necessary to superimpose μ onto x.

We now provide methods to find hard (one-to-one), soft (many-to-one) and "super soft" (many-to-many) matches. The latter types of matching are useful when comparing protein images as multiple forms of an individual protein can often be visualised [7]. That is, a single protein can produce multiple spots on an image. Let Δ be a $(K+m+1) \times (K+n)$ matrix. We can estimate the matching matrix, M, using the posterior matching probabilities by setting $\Delta = \hat{p}^T$. Alternatively, we can control the output number of matches and the maximum distance between two matched points by setting $\Delta = D^*$, where D^* is the $(K+m+1)\times(K+n)$ matrix containing all pairwise Euclidean distances between points in the transformed μ and points in x. An element in D^* is set to be the following.

$$D^*_{ij} = \begin{cases} d^2_{ij} & \text{for } i = 1, \ldots, K+m \\ d^2_T & \text{for } i = 0 \end{cases},$$

for $j = 1, \ldots, K+n$ where

$$d_{ij} = \|x_j - \hat{A}\mu_i - \hat{b}\|$$

and d_T is an assigned distance threshold. The lower we fix d_T, the lower the number of output matches. Like previously, for simplicity of notation we set $D^*_{ij} = D^*_{0j}$ and $\Delta_{ij} = \Delta_{0j}$ for $i = K+m+1$.

Note: Controlling the number of matches is useful if we want to highlight the most likely matched pair or the 10 most likely, for example.

One-to-one matches

For one-to-one matches across μ and x, we need to apply the constraints stated in Equations (2.1) and (2.2). The conditional likelihood and the log-likelihood of M are respectively given as

$$\prod_{i=0}^{K+m}\prod_{j=1}^{K+n}\Delta_{ij}^{M_{ij}}$$

and

$$\sum_{i=0}^{K+m}\sum_{j=1}^{K+n}M_{ij}\log\Delta_{ij}. \tag{2.18}$$

We find M that maximises this log-likelihood when $\Delta=\hat{p}^T$ or that minimises the log-likelihood when $\Delta=D^*$. We input $\log\Delta$ and the $2K+m+n$ constraints into a hardening algorithm developed by Michael Berkelaar [10], which will output the estimated one-to-one matching matrix, $\hat{M}$.

Note 1: If $\Delta_{ij}=0$, then $\log\Delta_{ij}=-\infty$ which will halt the hardening algorithm. To allow the algorithm to run, we set $\log\Delta_{ij}=-1\times10^{10}$ if $\Delta_{ij}=0$.

Note 2: If $\Delta_{0j}>\Delta_{ij}$ when $\Delta=\hat{p}^T$ or $\Delta_{0j}<\Delta_{ij}$ when $\Delta=D^*$ for all $i\neq0$, the algorithm would set $M_{0j}=1$. To reduce computational workload, we exclude column j in Δ from the hardening algorithm when the described conditions are met and automatically set $M_{0j}=1$.

Many-to-one matches

For many-to-one matches from x to μ, we only need to apply the constraint stated in Equation (2.1).

In this case we simply set

$$\hat{M}_{i_1j}=\begin{cases}1 & \text{if, for all } i_2\neq i_1,\ \Delta_{i_1j}>\Delta_{i_2j} \text{ when } \Delta=\hat{p}^T \text{ or if } \Delta_{i_1j}<\Delta_{i_2j} \text{ when } \Delta=D^*\\ 0 & \text{otherwise}\end{cases}, \tag{2.19}$$

for $j=1,\ldots,K+n$.

Many-to-many matches

Here the only constraints are that $M_{ij} = \{0, 1\}$ and that if $M_{0j} = 1$, then $M_{ij} = 0$ for all $i \neq 0$. That is, x_j can not be allocated to the coffin bin and matched with points in μ.

Here we set

$$\hat{M}_{ij} = \begin{cases} 1 & \text{if } \Delta_{ij} > \Delta_{0j} \text{ when } \Delta = \hat{p}^T \text{ or if } \Delta_{ij} < \Delta_{0j} \text{ when } \Delta = D^* \\ 0 & \text{otherwise} \end{cases}, \quad (2.20)$$

for $i = 1, \ldots, K + m$ and $j = 1, \ldots, K + n$.

The estimated number of matches, denoted by $\hat{L}$, is

$$\hat{L} = \sum_{i=1}^{K+m} \sum_{j=1}^{K+n} \hat{M}_{ij}, \quad (2.21)$$

where $\hat{M}$ is the inferred matching matrix.

2.3.4 Composite algorithm

We can summarise each step within the algorithm as follows.

1. Assign q_{ij} using Equations (2.3) and (2.4) in the standard method or Equations (2.5), (2.7) and (2.8) in the adapted method for $i = 0, \ldots, K + m$ and $j = 1, \ldots, K + n$.

2. Find initial estimates of the transformation parameters, $A^{(0)}$ and $b^{(0)}$, and assign the variance, σ_{ij}^2. Possible choices are discussed in the following subsection.

3. Run the EM algorithm to get the updated estimates, $p_{ji}^{(1)}$, $A^{(1)}$ and $b^{(1)}$, using Equations (2.11), (2.16) and (2.14) respectively.

4. Repeat step 3 to find the updated estimates, $p_{ji}^{(r+1)}$, $A^{(r+1)}$ and $b^{(r+1)}$, until convergence (defined in Equation (2.17)) is reached. Let the final posterior matching probabilities be stored in the $(K + n) \times (K + m + 1)$ matrix $\hat{p}$ and the final estimated transformation parameters be denoted by $\hat{A}$ and $\hat{b}$.

5. We can choose to assign matches by setting $\Delta = \hat{p}^T$ or $\Delta = D^*$. One-to-one matches are assigned using the hardening algorithm described in the previous subsection, many-to-one matches using Equation (2.19) or many-to-many matches using Equation (2.20).

6. Treating the matches within the inferred matching matrix, $\hat{M}$, as known, we can update the transformation parameters using Procrustes methodology [28] to calculate the final estimates, $\hat{\hat{A}}$ and $\hat{\hat{b}}$.

2.3.5 Assigning the function and parameters within the EM algorithm

When considering the adapted method to assign prior matching probabilities for the markers, we need to assign the function f stated in Equation (2.5). We also need to assign starting values for the transformation parameters denoted by $A^{(0)}$ and $b^{(0)}$. Finally we need to assign a variance between a point i in μ and a point j in x, denoted by σ_{ij}^2.

We look at each assignment separately.

Function applied within adapted method

We discuss two possible choices for the function, f, in Equation (2.5).

As before, μ_k contains the allocated marker coordinates for marker k in μ, $k = 1, \ldots, K$ and γ_k is the index of the true marker k in μ.

Let $\bar{d}_{ik}$ denote the expected distance between a point μ_i and μ_k for $i = 1, \ldots, K+m$. Due to the freedom for a gel to warp, in reality the distance between μ_i and μ_k in an image is

$$d_{ik} = \bar{d}_{ik} + \varepsilon,$$

where ε denotes some error.

The first choice for the function, f, in Equation (2.5) is motivated by the likelihood of clusters to occur within a gel and the resulting difficulty in correctly allocating a marker within a cluster of points.

We can accommodate for the increased likelihood that a marker μ_k is misallocated if it exists within a cluster of other points, for $k = 1, \ldots, K$, by stating

$$q_{ik} = p(\gamma_k = i) \propto \begin{cases} \frac{1}{C_k} & \text{if } d_{ik} \leq \varepsilon \\ 0 & \text{if } d_{ik} > \varepsilon \end{cases}, \tag{2.22}$$

where d_{ik} is a Euclidean distance calculated with Equation (2.6) and

$$C_k = \sum_{i=1}^{K+m} I[d_{ik} \leq \varepsilon],$$

where $I[d_{ik} \leq \varepsilon] = 1$ if $d_{ik} \leq \varepsilon$ and $I[d_{ik} \leq \varepsilon] = 0$ if $d_{ik} > \varepsilon$ for $i = 1, \ldots, K + m$. So C_k is simply the number of points in μ that are within a distance of ε from μ_k.

For the second choice of the function, f, in Equation (2.5), all points in μ are considered as possible true markers. We apply a normal distribution to ε so that

$$\varepsilon \sim N_D(\mu_k, \sigma_*^2 I_D)$$

and

$$q_{ik} = p(\gamma_k = i) \propto \frac{1}{(2\pi\sigma_*^2)^{D/2}} \exp\left\{-\frac{\|\mu_i - \mu_k\|^2}{2\sigma_*^2}\right\}, \tag{2.23}$$

for $i = 1, \ldots, K + m$, where $2\sigma_*^2$ is the variance between two points in μ (assuming independence across dimensions). So the probability that μ_i is the true marker k will decrease the further it is from μ_k.

Starting values for transformation parameters

As we have prior knowledge of allocated corresponding markers in both μ and x, it is sensible that $A^{(0)}$ and $b^{(0)}$ are set as the transformation parameters necessary to best superimpose corresponding markers. Dryden and Mardia [28] show how these parameters can be estimated from the matrix,

$$R = (\mu_*^T \mu_*)^{-1} \mu_*^T x^{'}, \tag{2.24}$$

where μ_* is the $K \times (D+1)$ matrix $\mu_* = (\underline{1}_K, \mu')$ and $\underline{1}_K$ is a vector of ones of length K. The $K \times D$ matrices, μ' and x', contain only the marker coordinates for μ and x respectively.

The first column in R^T contains $b^{(0)}$ and the second two columns in R^T contain the $D \times D$ matrix $A^{(0)}$.

Starting values for the variance between images

Constant variance

We can estimate a constant variance, $\sigma_{ij}^2 = \sigma^2$ for $i = 1, \ldots, K+m$ and $j = 1, \ldots, K+n$, by considering the mean squared distance between corresponding markers in μ and x after an affine transformation has been applied to superimpose them. That is, set

$$\hat{\sigma}^2 = \frac{1}{\nu} \sum_{k=1}^{K} \|x_k - A^{(0)}\mu_k - b^{(0)}\|^2, \tag{2.25}$$

where $\nu = DK - D^2 - D$ and denotes the degrees of freedom. Here DK is the number of error terms in the D components of the K markers. This number is reduced in ν to accommodate the estimates of $A^{(0)}$ and $b^{(0)}$.

Increased edge variance

Due to the chemical procedure used to create images, points close to the edges tend to have a higher degree of positional error than those allocated close to the centre of an image. For this reason we provide a method that will take into account an increased edge variance within an image.

Let us consider the single image μ. Let w and h denote the width and height of μ respectively. If a point μ_i is a greater distance than a from any edge of the image μ, then the influence due to edge proximity on positional variance is negligible, so we fix μ_i to have a fixed variance, σ_0^2. If a point μ_i is a lesser distance than a from any edge of the image μ, then the variance of μ_i will be location dependent. We define σ_i^2, the variance of a point μ_i, separately for each of the nine areas (displayed in Figure 2.1) that μ_i can lie

within.

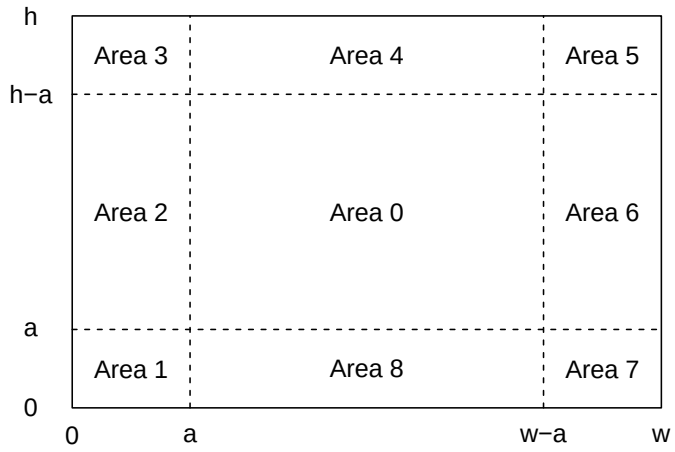


Figure 2.1: Figure displaying the nine areas of an image in which we separately define the variance of a point μ_i, σ_i^2.

Let μ_{i1} and μ_{i2} denote the x and y coordinates of a point μ_i.

Area 0

If $a \leq \mu_{i1} \leq w - a$ and $a \leq \mu_{i2} \leq h - a$, then

$$\sigma_i^2 = \sigma_0^2.$$

The variance is fixed at σ_0^2 for any point in μ present in area 0.

Area 1

If $\mu_{i1} < a$ and $\mu_{i2} < a$, then

$$\sigma_i^2 = c\left[(a - \mu_{i1})^2 + (a - \mu_{i2})^2\right] + \sigma_0^2,$$

where c is some scaling factor. The distance of μ_i from the bottom-left corner of μ dictates σ_i^2.

Area 2

If $\mu_{i1} < a$ and $a \leq \mu_{i2} \leq h - a$, then

$$\sigma_i^2 = c(a - \mu_{i1})^2 + \sigma_0^2.$$

The distance of μ_i from the left edge of μ dictates σ_i^2.

Area 3

If $\mu_{i1} < a$ and $\mu_{i2} > h - a$, then

$$\sigma_i^2 = c\left[(a - \mu_{i1})^2 + (\mu_{i2} - h + a)^2\right] + \sigma_0^2.$$

The distance of μ_i from the top-left corner of μ dictates σ_i^2.

Area 4

If $a \leq \mu_{i1} \leq w - a$ and $\mu_{i2} > h - a$, then

$$\sigma_i^2 = c(\mu_{i2} - h + a)^2 + \sigma_0^2.$$

The distance of μ_i from the top edge of μ dictates σ_i^2.

Area 5

If $\mu_{i1} > w - a$ and $\mu_{i2} > h - a$, then

$$\sigma_i^2 = c\left[(\mu_{i1} - w + a)^2 + (\mu_{i2} - h + a)^2\right] + \sigma_0^2.$$

The distance of μ_i from the top-right corner of μ dictates σ_i^2.

Area 6

If $\mu_{i1} > w - a$ and $a \leq \mu_{i2} \leq h - a$, then

$$\sigma_i^2 = c(\mu_{i1} - w + a)^2 + \sigma_0^2.$$

The distance of μ_i from the right edge of μ dictates σ_i^2.

Area 7

If $\mu_{i1} > w - a$ and $\mu_{i2} < a$, then

$$\sigma_i^2 = c\left[(\mu_{i1} - w + a)^2 + (a - \mu_{i2})^2\right] + \sigma_0^2.$$

The distance of μ_i from the bottom-right corner of μ dictates σ_i^2.

Area 8

If $a \leq \mu_{i1} \leq w - a$ and $\mu_{i2} < a$, then

$$\sigma_i^2 = c(a - \mu_{i2})^2 + \sigma_0^2.$$

The distance of μ_i from the bottom edge of μ dictates σ_i^2.

Similarly, let σ_{xj}^2 be the variance of a point x_j in x. Using the appropriate values of w and h, we can calculate σ_{xj}^2 for x_j in the same way we have calculated σ_i^2 for μ_i. We can estimate the variance between the point μ_i in μ and the point x_j in x as

$$\hat{\sigma}_{ij}^2 = \sigma_i^2 + \sigma_{xj}^2,$$

for $i = 1, \ldots, K + m$ and $j = 1, \ldots, K + n$.

2.4 Accounting for grossly misallocated or missing markers

The number of missing or grossly misidentified markers are dependent on the quality of the equipment and the expert that create the images.

2.4.1 Grossly misallocated markers

Gross misallocations of a marker may occur through human error when inputting marker labels into data spreadsheets. For instance, spot ID 153 could easily be labelled as marker 1 rather than spot ID 135. Dryden and Walker [29] consider procedures based on S estimators, least median of squares and least quartile difference estimators that are highly resistant to outlier points. Here we describe how we can use the EM algorithm previously described.

The EM algorithm is very much dependent on the transformation parameters input as starting values, $A^{(0)}$ and $b^{(0)}$. We have previously stated that the affine transformation

necessary to superimpose corresponding markers in μ and x will provide sensible starting values for the transformation parameters within the EM algorithm. However this would not be the case if gross misallocations occur.

Here we provide a method that will highlight grossly misallocated markers across images. Highlighted markers can then be automatically removed or corrected before they are used within the EM algorithm to estimate transformation starting values.

Let $\mu^{'}$ and $x^{'}$ be $K \times D$ coordinate matrices where $\mu_k^{'}$ and $x_k^{'}$ contain the coordinates of marker k in μ and x respectively for $k = 1, \ldots, K$.

Here we consider the prior matching probabilities to be independent of the distance between a possible marker and the allocated marker so that

$$q_{ik} = \begin{cases} p_M & \text{for } i = k \\ \frac{1-p_M}{K} & \text{for } i \neq k \end{cases}, \tag{2.26}$$

where p_M denotes the probability that the allocated marker $\mu_k^{'}$ truly corresponds to the allocated marker $x_k^{'}$.

We input $\mu^{'}$ and $x^{'}$ into steps 1-5 of the composite algorithm to estimate the one-to-one matching matrix $\hat{M}$, replacing Equations (2.5) and (2.7) with Equation (2.26) in stage 1. We use a sensible fixed variance $\hat{\sigma}_{ij}^2 = \hat{\sigma}^2$ in Equation (2.9). We use Equation (2.24) to estimate the starting transformation values, $A^{(0)}$ and $b^{(0)}$. Note that the starting transformation will be distorted by the presence of grossly allocated markers.

There are four possible outcomes for $k = 1, \ldots, K$.

- The allocated corresponding markers $\mu_k^{'}$ and $x_k^{'}$ are matched if

  $$\hat{M}_{kk} = 1.$$

  We include both $\mu_k^{'}$ and $x_k^{'}$ in further analyses.

- The marker $x_k^{'}$ is allocated to the coffin bin if

  $$\hat{M}_{0k} = 1.$$

  We exclude both $\mu_k^{'}$ and $x_k^{'}$ from further analyses.

- No point in $x^{'}$ is matched to the marker $\mu_k^{'}$ if

$$\hat{M}_{kj} = 0,$$

for all $j = 1, \ldots, K$. We exclude both $\mu_k^{'}$ and $x_k^{'}$ from further analyses.

- The marker $\mu_{k_1}^{'}$ is matched to an allocated non-corresponding marker $x_{k_2}^{'}$ if

$$\hat{M}_{k_1 k_2} = 1,$$

for $k_1 \neq k_2$. We exclude $\mu_{k_1}^{'}$, $\mu_{k_2}^{'}$, $x_{k_1}^{'}$ and $x_{k_2}^{'}$ from further analyses.

2.4.2 Missing markers

It is possible that all K markers are not successfully located in both μ and x. For example, only 10 out of the possible $K = 12$ markers were located in the image displayed in Figure 1.2.

There are four possibilities we must consider for $k = 1, \ldots, K$.

- **Case 1:** Marker k is located in both μ and x.
- **Case 2:** Marker k is located in μ alone.
- **Case 3:** Marker k is located in x alone.
- **Case 4:** Marker k is not located in either μ or x.

We first introduce notation to allow for the possibility of missing markers.

Let K_μ and K_x be the total number of markers located in μ and x respectively. As previously notated, let μ be the $(K+m) \times D$ coordinate matrix and x be the $(K+n) \times D$ coordinate matrix.

If marker k is located in μ, then μ_k contains the coordinates of marker k in μ. If marker k is not located in μ, then $\mu_k = \emptyset$. Similarly if marker k is located in x, then x_k

contains the coordinates of marker k in x, for $k = 1, \ldots, K$. If marker k is not located in x, then $x_k = \emptyset$.

As previously stated, Q is the $(K + m + 1) \times (K + n)$ matrix containing the prior matching probabilities for points in x. We redefine Q separately for both the standard and adapted method.

Standard method

We assume that the allocated marker k is the true marker k, for $k = 1, \ldots, K$.

Markers in x

Case 1: If $\mu_k \neq \emptyset$ and $x_k \neq \emptyset$, then marker k is located in both μ and x and we can assign q_{ik} as previously defined in Equation (2.3) for $i = 1, \ldots, K + m$.

Case 2: If $\mu_k \neq \emptyset$ and $x_k = \emptyset$, then marker k is located in μ alone. As we assume that an allocated marker k is the true marker k, we know that μ_k does not have a corresponding point in x. We can remove μ_k from the analyses by setting

$$q_{kj} = \emptyset \text{ for } j = 1, \ldots, K + n.$$

Alternatively we could set $q_{kj} = 0$ for $j = 1, \ldots, K + n$ throughout the EM algorithm and remove μ_k before assigning matches.

Case 3: If $\mu_k = \emptyset$ and $x_k \neq \emptyset$, then marker k is located in x alone. In this case we know that x_k does not have a corresponding point in μ. We can remove x_k from the analyses by setting

$$q_{ik} = \emptyset \text{ for } i = 0, \ldots, K + m.$$

Alternatively we can set

$$q_{ik} = \begin{cases} 1 & \text{for } i = 0 \\ 0 & \text{for } i = 1, \ldots, K + m. \end{cases},$$

to ensure that x_k is allocated to the coffin bin. Again we would remove x_k before assigning matches.

Case 4: If $\mu_k = \emptyset$ and $x_k = \emptyset$, then marker k is not located in either μ or x. We set

$$q_{ik} = q_{kj} = \emptyset \text{ for } i = 0, \ldots, K+m \text{ and } j = 1, \ldots, K+n.$$

Non-markers in x

In the standard method we only consider the possibility that a non-marker in x matches a non-marker in μ, otherwise it is allocated to the coffin bin. So we can still use the previously defined Equation (2.4) to evaluate q_{ij} for $i = 0, \ldots, K+m$ and $j = K+1, \ldots, K+n$.

Adapted method

Now we allow for the possibility that an allocated marker k is not the true marker k, for $k = 1, \ldots, K$.

Markers in x

Case 1: If $\mu_k \neq \emptyset$ and $x_k \neq \emptyset$, we assign q_{ik} as previously stated in Equations (2.5) and (2.7) for $i = 0, \ldots, K+m$.

Case 2: If $\mu_k \neq \emptyset$ and $x_k = \emptyset$, we treat μ_k as a non-marker.

Case 3: If $\mu_k = \emptyset$ and $x_k \neq \emptyset$, we treat x_k as a non-marker.

Case 4: If $\mu_k = \emptyset$ and $x_k = \emptyset$, we set

$$q_{ik} = q_{kj} = \emptyset \text{ for } i = 0, \ldots, K+m \text{ and } j = 1, \ldots, K+n.$$

Non-markers in x

The prior matching probability of a non-marker, x_j, is again set to be uniform over all matching possibilities so that, for $i = 0, \ldots, K+m$ and $j = K+1, \ldots, K+n$,

$$q_{ij} = \frac{1}{K_\mu + m + 1}. \tag{2.27}$$

In Case 3, when $\mu_k = \emptyset$ and $x_k \neq \emptyset$ for $k = 1, \ldots, K$, we treat x_k as a non-marker and use Equation (2.27) to calculate q_{ik} for $i = 0, \ldots, K+m$.

Note that μ contains K_μ markers and m non-markers. There are only $K_\mu + m + 1$ matching possibilities for a point in x, thus producing the denominator in Equation (2.27).

Chapter 3

Experiments and Applications

3.1 Introduction

Here we analyse the properties and the accuracy of the methodology introduced in Chapter 2. In Section 3.2 we simulate data to examine the accuracy of the algorithm and to highlight appropriate parameters that should be used in further analyses. We begin by comparing the results when applying the standard or the adapted method within the model in Subsection 3.2.1. In Subsection 3.2.2 we examine the matches made when using the final posterior probabilities or the final superimposition output by the EM algorithm. In Subsection 3.2.3 we highlight the appropriate parameter necessary to successfully highlight grossly misallocated markers. In Section 3.3 we incorporate the conclusions from Section 3.2 into the analyses of real data. We investigate the presence of grossly misallocated markers and include a simulated example to show how two incorrectly switched marker labels are correctly highlighted in Subsection 3.3.1. In Subsection 3.3.2 we investigate whether there is evidence of an increased edge variance within our dataset. Finally, in Subsection 3.3.3 we provide an example of how the methodology from Chapter 2 is implemented to highlight corresponding points across images.

Throughout the simulations and when relevant, we assume σ_{ij}^2 in Equation (2.9) is constant and estimate it as $\hat{\sigma}^2 = 4.5^2$, which is approximately the median squared distance

between two corresponding markers within the real dataset after all pairwise Procrustes transformations are performed. Alternatively, we estimate σ^2 using Equation (2.25) with denominator K instead of ν. Note that these estimates provide a conservative value of σ^2 and allow greater freedom for the distance between potential and known corresponding points. Though sensitivity tests are not carried out here, future work should involve a thorough exploration of the algorithm sensitivity to σ^2. The values presented here will be strongly dependent on the assigned σ^2.

For each investigation we fix $l = 10$ to define convergence in Equation (2.17).

3.2 Simulating data to analyse properties and highlight optimal parameters

3.2.1 Standard vs adapted method

We want to compare the accuracy of the estimated superimposition of μ onto x when applying the standard method or the adapted method. Here we produce six types of data which are described within the simulations below.

1. We simulate a 2-DE gel image, denoted by μ_G, by randomly scattering m_G points across a $w \times h$ uniform surface where each point is set to be a minimum of 2 units from any other point. These points will represent all points present in the theoretical 2-DE gel image.

2. We randomly select K *true* markers from the m_G points in μ_G, with the constraint that each marker must be a minimum distance of d_K from any other marker.

3. For simplicity we have previously considered markers and non-markers to be disjoint sets of points. In reality, this may not always be the case. Within this simulation we consider the following three ways to allocate non-markers.

(a) The m non-markers are randomly selected from the remaining $m_G - K$ points in μ_G. The K markers and m non-markers are disjoint sets of points.

(b) The m non-markers are randomly selected from all m_G points in μ_G. A marker can also be a non-marker.

(c) The K markers are a subset of the non-markers and the remaining $m - K$ non-markers are randomly selected from the remaining $m_G - K$ points in μ_G.

The set of unique markers and non-markers create the (western blot) subject-treatment specific image denoted by μ. The labelling in μ is such that, for $i = 1, \ldots, K$, μ_i contains coordinate information for marker i. For $i \geq K + 1$, μ_i contains the coordinates of a non-marker.

4. We set $x_G = \mu_G$ and $x = \mu$. That is, μ_G and x_G represent replicate 2-DE gel images and μ and x represent replicate western blot images.

5. We add noise, $N(0, \tau^2/4)$, to each individual coordinate of the $m_G = n_G$ points within both μ_G and x_G respectively.

6. We produce both *standard* and *adapted data*.

   (a) Standard data is data in which the K true markers are correctly allocated, so the data remains as the μ and x described above.

   (b) To create adapted data we use Equation (2.23) to calculate the probability that a point μ_i^G in μ_G may be allocated as the true marker μ_k, for $i = 1, \ldots, m_G$ and $k = 1, \ldots, K$. These probabilities are then used to randomly allocate each marker μ_k. If the true marker k is neither correctly allocated or also a non-marker, then the true marker k is excluded from further analyses. The same is done for x. We fix $\sigma_*^2 = \tau^2$.

7. Both μ and x are input into steps 1–4 of the composite algorithm to produce the

final estimated transformation parameters, $\hat{A}$ and $\hat{b}$. We consider both the standard method and the adapted method within step 1 to analyse the data.

The starting values for the transformation parameters, $A^{(0)}$ and $b^{(0)}$, are found using Equation (2.24). We estimate the variance in Equation (2.9), σ^2, using Equation (2.25) with denominator K instead of ν. When implementing the adapted case, we set $\hat{\sigma}_*^2 = \hat{\sigma}^2$ in Equation (2.23).

8. Finally we calculate the RMSD between the true corresponding marker pairs as

$$\text{RMSD} = \sqrt{\frac{1}{K}\sum_{k=1}^{K} \|\hat{A}\mu_{\gamma_k} + \hat{b} - x_k\|^2},$$

where μ_{γ_k} contains the coordinates of the true marker k in μ. We fix $m_G = 2000$, $m = 120$, $K = 12$, $w = 257$, $h = 191$ and $d_K = 25$ to mimic the real data. We consider values of $\tau \in [1, 10]$ at integer intervals. We repeat the simulation 200 times for each combination of τ and the six types of data.

Note: We create both standard and adapted data and consider three different ways to allocate the non-markers. Thus we consider 6 types of data.

Discussion

Figure 3.1 displays the proportion of times out of the 200 simulations that the standard method gives a lower RMSD between the true corresponding markers than the adapted method. We can see that the only time the adapted method provides the more accurate result is when the markers are a subset of the non-markers for $\tau < 7$ for adapted data. As the markers are a subset of the non-markers, all true markers will be present even if they were not correctly allocated. Unlike the standard method, the adapted method allows the matching probability of truly corresponding markers to increase from zero, even when one or both of the markers are misallocated. In every other case, the application of the standard method provides the better result.

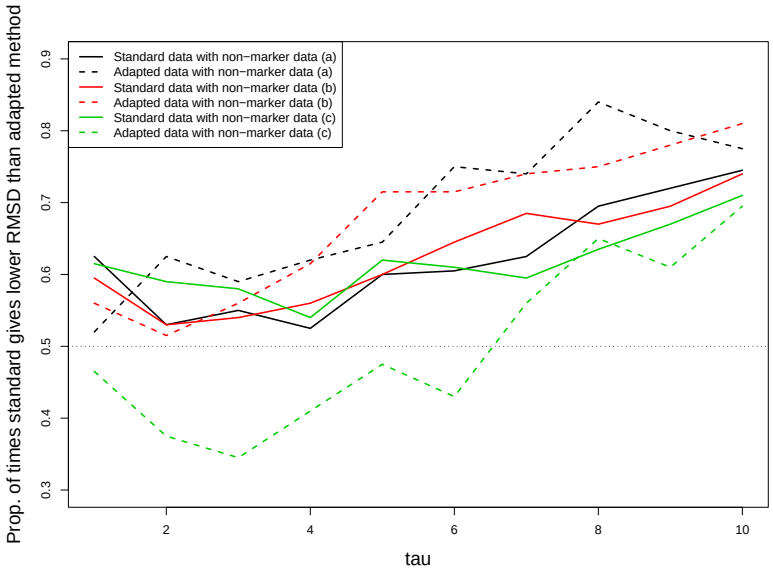


Figure 3.1: Figure displaying the proportion of times that the RSMD between corresponding markers when applying the standard method is less than the RMSD when applying the adapted method for each of the six types of data.

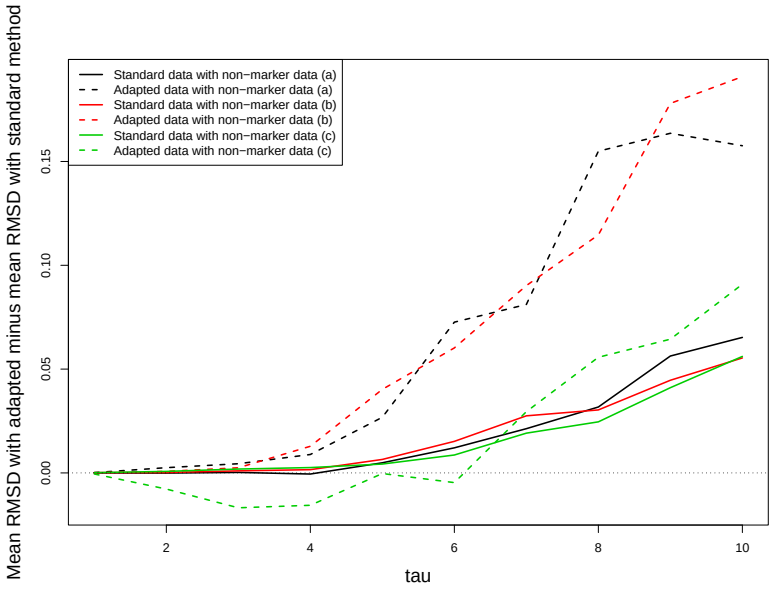


Figure 3.2: Figure displaying the mean RMSD calculated using the adapted method, minus the mean RMSD calculated using the standard method against τ.

τ	1	2	3	4	5	6	7	8	9	10
Standard data with non-marker allocation (a)										
Standard method	0.98	1.99	2.93	3.88	4.87	5.84	6.70	7.72	8.71	9.56
Adapted method	0.98	1.99	2.93	3.88	4.88	5.86	6.72	7.75	8.76	9.62
Standard data with non-marker allocation (b)										
Standard method	0.97	1.96	2.93	3.90	4.78	5.89	6.76	7.68	8.70	9.47
Adapted method	0.97	1.96	2.93	3.90	4.79	5.91	6.79	7.71	8.75	9.52
Standard data with non-marker allocation (c)										
Standard method	0.98	1.91	2.94	3.95	4.80	5.84	6.84	7.69	8.59	9.55
Adapted method	0.98	1.91	2.94	3.95	4.80	5.84	6.86	7.71	8.64	9.61
Adapted data with non-marker allocation (a)										
Standard method	0.98	1.98	2.90	3.87	4.90	5.97	7.24	8.29	9.33	10.52
Adapted method	0.98	1.99	2.91	3.88	4.92	6.04	7.32	8.45	9.49	10.68
Adapted data with non-marker allocation (b)										
Standard method	0.99	1.93	2.93	3.98	4.91	5.87	7.15	8.30	9.33	10.38
Adapted method	0.99	1.93	2.93	3.99	4.95	5.93	7.24	8.41	9.50	10.57
Adapted data with non-marker allocation (c)										
Standard method	0.97	1.92	2.91	3.90	4.95	5.89	6.91	8.20	9.44	10.44
Adapted method	0.97	1.91	2.89	3.88	4.95	5.89	6.94	8.25	9.50	10.53

Table 3.1: Table displaying the mean RMSD when applying the standard and adapted method to the six considered types of data.

Figure 3.2 displays the mean RMSD calculated using the adapted method, minus the mean RMSD calculated using the standard method for each τ. We can see that the standard and adapted methods produce very similar results for $\tau \leq 4$ when considering all three forms of standard data. For $\tau > 4$, the application of the standard method produces increasingly better results than the adapted method. For $\tau \leq 3$, the standard and adapted methods produce very similar results for the first two forms of adapted data. For $\tau > 3$, the standard method provides increasingly better results as τ increases. For the third form of adapted data, the application of the adapted method produces more accurate results for $\tau \leq 6$. However, as for all the other types of data, the standard method provides increasingly better results as τ increases.

Table 3.1 provides the mean RMSD when applying the standard and adapted method to the six considered types of data. We can see more clearly the patterns described above. As we intuitively would expect, we can see that the standard data generally produces an equal to or lower RMSD than the RMSD found with the adapted data.

Conclusion

For all future analysis, we choose to apply the standard method as this method generally produces the more accurate results. Furthermore, for the data considered in this research we should assume that the application of the blue stain to highlight markers (discussed in Subsection 1.2.4) would not be necessary if the markers were a subset of the non-markers. However, the way markers are allocated is dependent on the particular method used to create the images i.e. there could be cases where markers are subsets of the non-markers and the application of the adapted method would provide the more accurate results.

3.2.2 Assigning matches

We want to compare the accuracy of the matches made when setting $\Delta = \hat{p}^T$ or setting $\Delta = D^*$ and varying d_T. When setting μ and x to represent replicate images, we may

expect that the number of true positive matches will increase as d_T increases. However, if μ and x represent images that contain a low number of corresponding matches, increasing d_T will surely increase the number of false positive matches. For this reason, we also vary p_C, the proportion of corresponding non-markers across the images. Let $N = mp_C$ denote the number of corresponding non-markers between the images represented by μ and x.

We run the following simulation 500 times for each case.

1. We randomly scatter $K + 2m - N$ points across a $w \times h$ uniform surface, where each point is set to be a minimum of 2 units from any other point.

2. We randomly select K true markers from the $K+2m-N$ points with the constraint that each marker must be a minimum distance of d_K from any other marker. Let μ_k and x_k contain the coordinates of marker k in μ and x respectively, for $k = 1, \ldots, K$.

3. From the remaining $2m - N$ points, we randomly select N points to represent the corresponding non-markers across μ and x. So μ_i and x_i contain the coordinates of corresponding non-markers for $i = K + 1, \ldots, K + N$.

4. Finally, we randomly split the remaining $2(m - N)$ points equally between μ and x so that μ_i and x_j contain the coordinates of arbitrarily labelled points in μ and x, for $i, j = K + N + 1, \ldots, K + m$, that do not have corresponding points in x and μ respectively.

5. We add noise, $N(0, \tau^2/4)$, to each point coordinate within both μ and x.

6. Both μ and x are input into steps 1–5 of the composite algorithm to produce the estimated one-to-one matching matrix, $\hat{M}$. The starting values for the transformation parameters, $A^{(0)}$ and $b^{(0)}$, are found using Equation (2.24).

7. The number of correctly matched points is

$$n_{TP} = \sum_{j=1}^{K+N} \hat{M}_{jj}.$$

The number of points in x that are correctly allocated to the coffin bin is $n_{TN} = 0$ if $p_C = 1$ or

$$n_{TN} = \sum_{j=K+N+1}^{K+m} \hat{M}_{0j},$$

if $p_C \neq 1$.

The number of points in x that are incorrectly allocated to the coffin bin is

$$n_{FN} = \sum_{j=1}^{K+N} \hat{M}_{0j}.$$

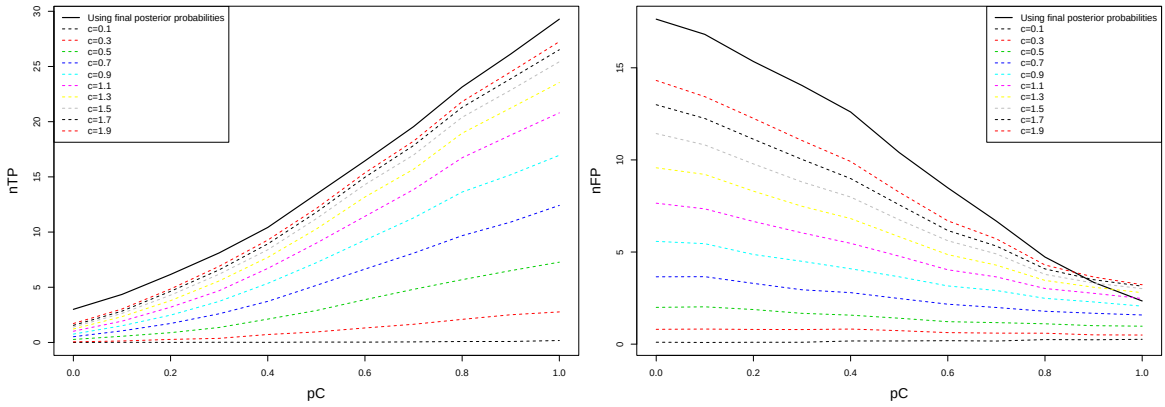
The number of falsely matched points in x is $n_{FP} = K + m - n_{TP} - n_{TN} - n_{FN}$.

In this case, to ease computational workload, we fix $m = 30$, $K = 3$, $w = 257/2$, $h = 191/2$ and $d_K = 25$. We estimate the variance in Equation (2.9) as $\hat{\sigma}^2 = 4.5^2$ and set $\tau = \hat{\sigma}$. We consider values of $p_C \in [0, 1]$ at intervals of 0.1. We first assign matches using the final posterior matching matrix by setting $\Delta = \hat{p}^T$. We also consider $c \in [0.1, 1.9]$ at intervals of 0.2 which fixes the distance threshold as $d_T = c\hat{\sigma}$ and estimate the matching matrix, M, using the pairwise distances between points across images.

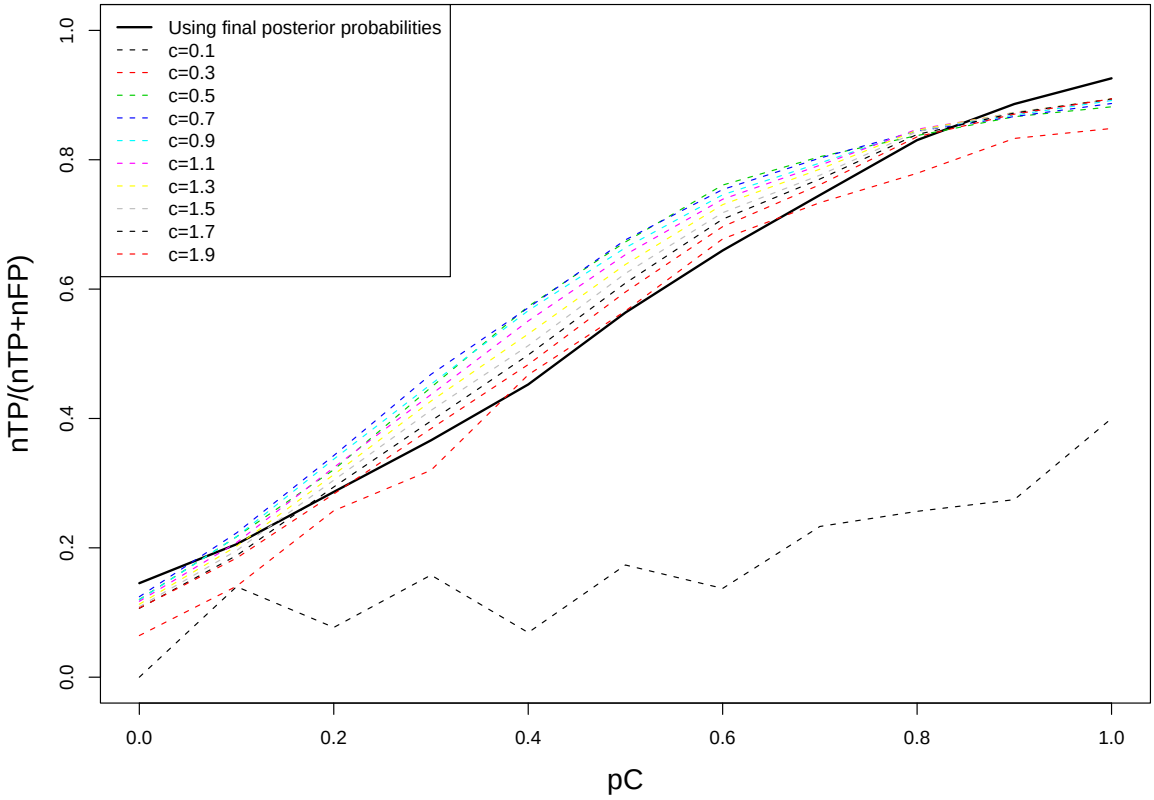
Discussion

Figure 3.3a and 3.3b display the number of true positive matches and the number of false positive matches made against p_C for each considered method of assigning matches. We can see that as d_T increases, both the number of true and false matches increase. Setting $\Delta = \hat{p}^T$ generally produces more true and false positives than the considered numerical values of d_T.

Figure 3.3c displays the proportion of true positive matches, $n_{TP}/(n_{TP} + n_{FP})$, against p_C for each considered method of assigning matches. For each matching method,



a) b)



c)

Figure 3.3: Figure showing a) n_{TP}, b) n_{FP} and c) $n_{TP}/(n_{TP} + n_{FP})$ against p_C when setting $\Delta = \hat{p}^T$ and when setting $\Delta = D^*$ for various d_T, where $d_T = c\hat{\sigma}$.

the proportion of true positive matches increases as the number of corresponding non-markers across images, p_C, increases.

For $0.1 \leq p_C \leq 0.8$, setting $0.5 \leq c \leq 1.9$ generally provides a higher proportion of true positives than when $\Delta = \hat{p}^T$. Setting $d_T \approx 0.7$ maximises the proportion of true positives, with the proportion decreasing as $d_T > 0.7$ and increasingly decreasing as $d_T < 0.7$.

For $p_C < 0.1$ and $p_C > 0.8$, setting $\Delta = \hat{p}^T$ provides a higher proportion of true positives than when $0.1 \leq c \leq 1.9$.

Conclusion

When matching points across replicates, set $\Delta = \hat{p}^T$. In other cases, p_C is unknown so set $d_T \approx 0.7\hat{\sigma}$ as this provides a higher proportion of true positive matches for a larger range of p_C.

3.2.3 Grossly misallocated markers

We want to highlight the appropriate proportion of correctly allocated marker pairs, p_M, necessary to assign matches when locating grossly misallocated markers.

We run the following simulation 1000 times.

1. We randomly scatter $K + m$ points across a $w \times h$ uniform surface, where each point is set to be a minimum of 2 units from any other point.

2. We randomly select K *true* markers from the $K + m$ points, with the constraint that each marker must be a minimum distance of d_K from any other marker. The remaining m points are the true non-markers. The K markers and m non-markers create μ. The labelling is such that μ_i contains the coordinates of the true marker k for $i = k = 1, \ldots, K$ and the coordinates of the arbitrarily labelled true non-marker i for $i = K + 1, \ldots, K + m$.

3. Let $\mu^{'}$ be the subset of μ containing the coordinates of the K markers only. We set $x^{'} = \mu^{'}$.

4. We fix the number of misallocated markers in $\mu^{'}$ as K^*. This value is related to the true proportion of correctly allocated marker pairs as
$$p_M = \frac{K - K^*}{K}.$$

5. Let $\mu^{'}_A$ contain the coordinates of the allocated markers in μ. For $K^* > 0$, we randomly select (without replacement) one of the m true non-markers to be the allocated marker k, for $k = 1, \ldots, K^*$. Let $\pi_{A'}$ be a vector of length K^*. If an element $\pi_k^{A^{'}} = i$, then the true non-marker, μ_i, is allocated as marker k for $i = K + 1, \ldots, K + m$ and $k = 1, \ldots, K^*$. We set $\mu_k^{A^{'}} = \mu_{\pi_k^{A'}}$ for $k = 1, \ldots, K^*$ and $\mu_k^{A^{'}} = \mu_k^{'}$ for $k = K^* + 1, \ldots, K$. The labelling is such that $\mu_i^{A^{'}}$ contains the coordinates of the allocated marker k for $i = k = 1, \ldots, K$.

   Note that we do not allow marker labels to be exchanged within this simulation.

6. We add noise, $N(0, \tau^2/4)$, to each point coordinate within $\mu^{'}_A$ and $x^{'}$.

7. The allocated markers in $\mu^{'}_A$ and the true markers in $x^{'}$ are input into steps 1–5 of the composite algorithm to produce the estimated one-to-one matching matrix, $\hat{M}$. The starting values for the transformation parameters, $A^{(0)}$ and $b^{(0)}$, are found using Equation (2.24). We use the final posterior probabilities, $\hat{p}$, to assign one-to-one matches.

8. The number of correctly matched marker pairs is
$$n_{TP} = \sum_{k=K^*+1}^{K} \hat{M}_{kk}.$$
   The number of markers in x that are correctly allocated to the coffin bin for $K^* = 0$ is $n_{TN} = 0$ and for $K^* \neq 0$,
$$n_{TN} = \sum_{k=1}^{K^*} \hat{M}_{0k}.$$

The number of markers in x that are incorrectly allocated to the coffin bin is

$$n_{FN} = \sum_{k=K^*+1}^{K} \hat{M}_{0k}.$$

The number of falsely matched markers in x is $n_{FP} = K - n_{TP} - n_{TN} - n_{FN}$.

We fix $m = 120$, $K = 12$, $w = 257$, $h = 191$ and $d_K = 25$. We estimate the variance in Equation (2.9) as $\hat{\sigma}^2 = 4.5^2$ and set $\tau = \hat{\sigma}$. We consider values of $K^* \in \{0, 3\}$ at integer intervals (equivalent to $p_M \in \{1, 0.92, 0.83, 0.75\}$) and $\hat{p}_M \in \{0.01, 0.99\}$ at intervals of 0.07.

As it is the matches made that are used to highlight marker correspondencies in future analyses, we focus mainly on the true and false matches made.

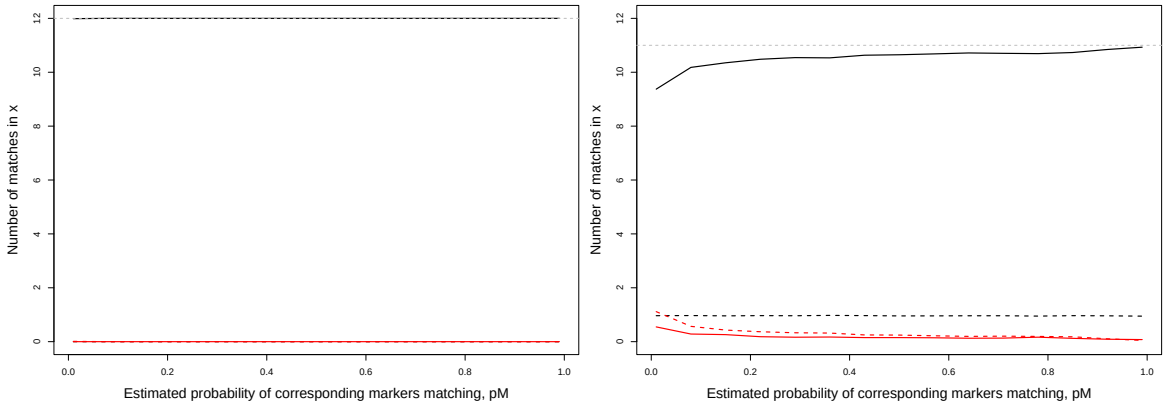
Conclusion

Figure 3.4 displays the number of matches in x against the input $\hat{p}_M$ for each considered K^*. We can see that increasing $\hat{p}_M$ increases the number of true positive matches and decreases the number of false positive matches for all $p_M \in \{1, 0.92, 0.83, 0.75\}$. Therefore for future analyses we set $\hat{p}_M = 0.99$. Setting $\hat{p}_M = 0.99$ indicates that correspondingly labelled markers are highly likely to match, but still allows the possibility for this not to be the case.

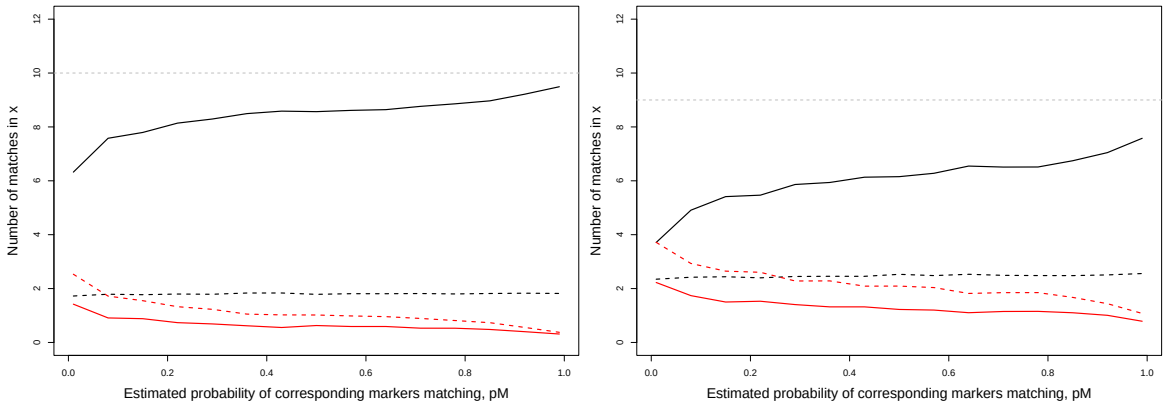
3.2.4 Overall conclusions

- The application of the standard method generally produces better results than the adapted method. That is, the assumption that the allocated markers are correctly allocated amid warping provides a more accurate match than when the method is allowed the freedom to explore other possible markers when simulating images from the given dataset.

- When matching points across replicates, set $\Delta = \hat{p}^T$. In other cases, p_C is unknown



a) b)



c) d)

Figure 3.4: Figures displaying the number of matches made in x against the input $\hat{p}_M$ for a) $K^* = 0$, b) $K^* = 1$, c) $K^* = 2$ and d) $K^* = 3$. The solid black line represents n_{TP}, the broken black line represents n_{TN}, the solid red line represents n_{FP}, the broken red line represents n_{FN}.

so set $\Delta = D^*$ and $d_T \approx 0.7\hat{\sigma}$, as this provides a higher proportion of true positive matches for a larger range of p_C.

- We found that setting a high $\hat{p}_M$ will highlight more true positive correspondences, even when the proportion of correctly allocated corresponding marker pairs is as low as $p_M = 0.75$.

3.3 Application examples

3.3.1 Grossly misallocated markers

Real gel data

Let μ_l represent image l in our dataset for $l = 1, \ldots, 26$. Let $\mu'_{l_1 l_2}$ be the $K_{l_1 l_2} \times 2$ matrix containing only the marker coordinates of the markers in μ_{l_1} that have correspondingly labelled markers in μ_{l_2}.

We input the corresponding markers for all pairwise comparisons into steps 1–5 of the composite algorithm to estimate the one-to-one matching matrix, $\hat{M}_{l_1 l_2}$, found when superimposing $\mu'_{l_1 l_2}$ onto $\mu'_{l_2 l_1}$ for $l_1, l_2 = 1, \ldots, 26$ and $l_1 \neq l_2$. That is, we transform the appropriate markers in image l_1 onto the correspondingly labelled markers in image l_2. So the indices l_1 and l_2 indicate the direction of transformation between images.

Note: If a marker k is not allocated in both μ_{l_1} and μ_{l_2}, it is excluded from the analysis.

Again, the parameters within the algorithm are set to be the same as those used in the previous simulations. We estimate the variance in Equation (2.9) as $\hat{\sigma}^2 = 4.5^2$ and the proportion of correctly corresponding marker pairs in Equation (2.26) as $\hat{p}_M = 0.99$. The starting values for the transformation parameters, $A^{(0)}$ and $b^{(0)}$, are found using Equation (2.24). We use the final posterior probabilities, $\hat{p}$, to estimate the matches by fixing $\Delta = \hat{p}^T$.

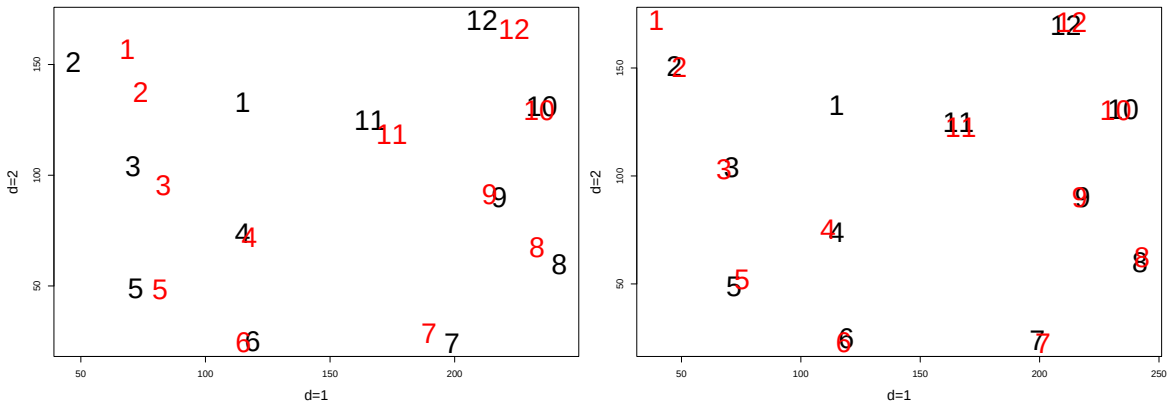
Let τ_k be a vector containing the indices of the images that contain marker k. Here we discuss all cases in the 26×25 comparisons where grossly misallocated markers are highlighted.

Case 1:

Marker 1 remains unmatched in both images for $l_1 = 23$ and each $l_2 \in \tau_1$ where $l_1 \neq l_2$. Marker 1 also remains unmatched in both images when considering the reverse transformations for $l_2 = 23$ and each $l_1 \in \tau_1$ where $l_1 \neq l_2$. The length of τ_1 is 16, indicating 16 images in the dataset that contain marker 1.

Figure 3.5a and 3.5b respectively display the initial transformation of $\mu'_{26,23}$ onto $\mu'_{23,26}$, for example, before and after marker 1 is removed as a marker from both images. In this example, the RMSD between the 12 marker pairs before the removal is 19.44. The RMSD between the remaining 11 marker pairs after the removal is 2.96. Table 3.2 lists the RMSD between corresponding markers before and after the removal of marker 1 for each of the 30 comparisons. In each case we can see a dramatic reduction in RMSD between corresponding markers after marker 1 is removed as a marker.

Note: We leave removed markers within Figures simply for illustrative purposes.



a) b)

Figure 3.5: Figure displaying the initial transformation of $\mu'_{26,23}$ onto $\mu'_{23,26}$ a) before and b) after marker 1 is removed as a marker from both images.

l_2	Before	After
3	24.1	4.5
4	21.6	8.0
5	25.1	6.0
8	22.7	3.3
9	23.1	2.9
10	24.3	3.1
11	23.1	3.5
13	22.2	3.8
14	24.1	4.7
15	24.5	3.6
16	25.5	3.2
19	25.6	4.9
22	24.8	3.9
24	25.1	3.4
26	22.5	2.9

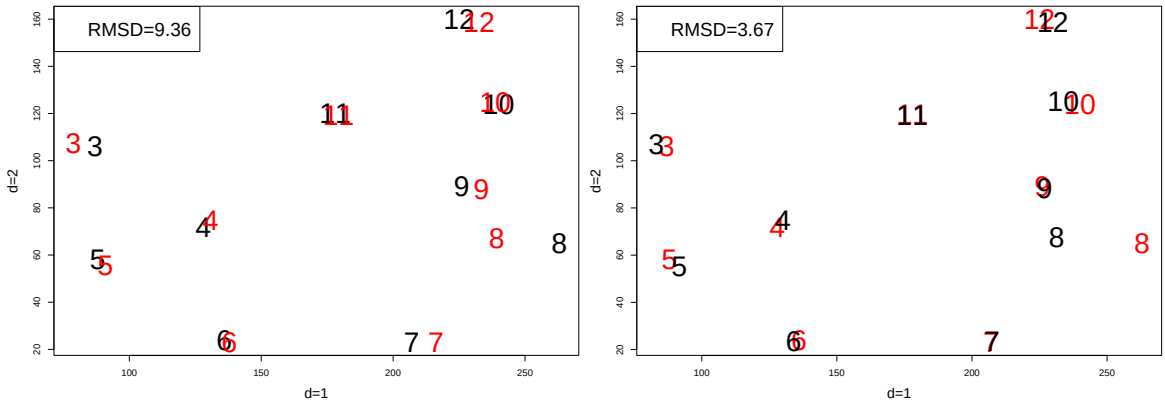
l_1	Before	After
3	21.8	4.9
4	18.4	8.1
5	21.5	6.1
8	19.9	3.2
9	22.8	3.2
10	22.1	3.5
11	20.6	3.9
13	19.8	3.7
14	22.0	5.2
15	21.9	3.8
16	21.6	3.2
19	22.6	5.2
22	21.6	4.1
24	22.0	3.4
26	19.4	3.0

Table 3.2: Tables displaying the RMSD between the allocated markers before and after marker 1 is removed as a marker from both images. The table to the left displays the RMSD when image 23 is transformed onto image l_2. The table to the right shows the RMSD when applying the reverse transformation.

Case 2:

Marker 8 remains unmatched in both images for $l_1 = 4$ and $l_2 = 25$. The same occurs for the reverse transformation when $l_1 = 25$ and $l_2 = 4$. Figure 3.6a and 3.6b respectively display the initial transformation of $\mu'_{4,25}$ onto $\mu'_{25,4}$ and the RMSD between markers before and after marker 8 is removed as a marker from both images.

Case 3:



a) b)

Figure 3.6: Figure displaying the initial transformation of $\mu'_{4,25}$ onto $\mu'_{25,4}$ a) before and b) after marker 8 is removed as a marker from both images. The RMSD between corresponding markers is indicated at the top-left of each figure.

Marker 8 remains unmatched in both images for $l_1 = 5$ and $l_2 = 25$. Figure 3.7a and 3.7b respectively display the initial transformation of $\mu'_{5,25}$ onto $\mu'_{25,5}$ and the RMSD between markers before and after marker 8 is removed as a marker from both images.

Case 4:

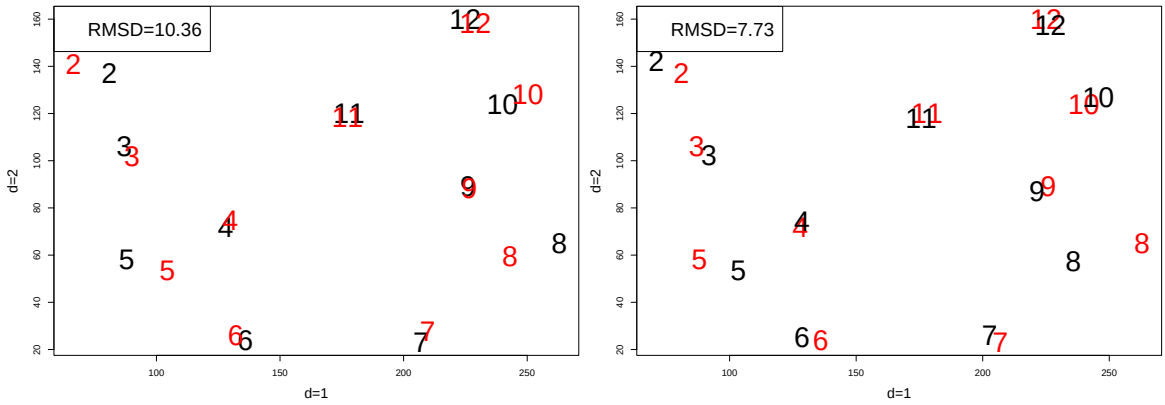
Marker 2 remains unmatched in both images for $l_1 = 25$ and $l_2 = 19$. Figure 3.8a and 3.8b respectively display the initial transformation of $\mu'_{25,19}$ onto $\mu'_{19,25}$ and the RMSD between markers before and after marker 2 is removed as a marker from both images.

Discussion

A summary of the gross misallocations found is given below.

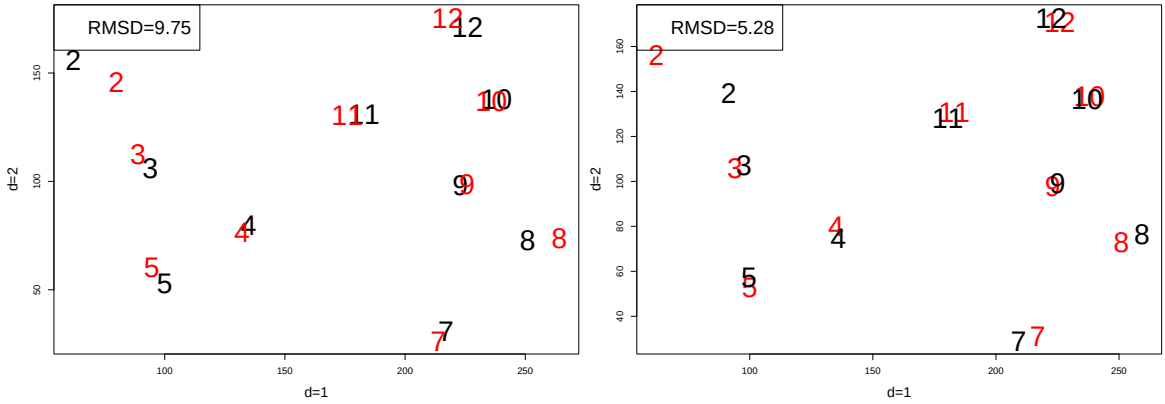
- There are 16 images containing marker 1. Marker 1 is highlighted as a gross misallocation in each of the 2×15 comparisons made with image 23.

- All 26 images contain marker 8. Marker 8 in image 25 is highlighted as a gross misallocation in 3 of the 2×25 comparisons considered involving image 25. So the



a) b)

Figure 3.7: Figure displaying the initial transformation of $\mu'_{5,25}$ onto $\mu'_{25,5}$ a) before and b) after marker 8 is removed as a marker from both images. The RMSD between corresponding markers is indicated at the top-left of each figure.



a) b)

Figure 3.8: Figure displaying the initial transformation of $\mu'_{25,19}$ onto $x'_{19,25}$ a) before and b) after marker 2 is removed as a marker from both images. The RMSD between corresponding markers is indicated at the top-left of each figure.

proportion of times marker 8 in image 25 is highlighted as a gross misallocation is 0.06.

- Marker 8 in image 4 is highlighted as a gross misallocation in 2 of the 2×25 comparisons considered involving image 4, i.e, a proportion of 0.04 times.

- Marker 8 in image 5 is highlighted as a gross misallocation in 1 of the 2×25 comparisons considered involving image 5, i.e, a proportion of 0.02 times.

- Marker 2 in both image 19 and image 25 is highlighted as a gross misallocation in 1 of the 2×15 comparisons considered involving image 19 and image 25 respectively, i.e, a proportion of 0.03 times.

Remark:

Following these discoveries, we were informed that marker 1 in image 23 was incorrectly labelled as spotID 136 when it should have been spotID 153.

To investigate whether our method would have found this match, we rerun each of the 2×15 transformations, this time reallocating markers 1 as non-markers in both images. We now consider the full image represented by μ_l for $l = 1, \ldots, 26$.

First we transform μ_{23} onto μ_l for $l \in \tau_1$ and $l \neq 23$. We also carry out the reverse transformation of μ_l onto μ_{23} for $l \in \tau_1$ and $l \neq 23$. For each pairwise comparison, we input both images into steps 1–5 of the composite algorithm to estimate the one-to-one matching matrix. For this analysis, we reassign marker 1 as a non-marker in both images and treat non-correspondingly labelled markers across images as non-markers.

The starting values for the transformation parameters, $A^{(0)}$ and $b^{(0)}$, are found using Equation (2.24). We estimate the variance in Equation (2.9), σ^2, using Equation (2.25) with denominator K instead of ν. We use the final posterior probabilities, $\hat{p}$, to estimate the matches by fixing $\Delta = \hat{p}^T$.

When transforming μ_{23} onto μ_l for $l \in \tau_1$ and $l \neq 23$, we found that the originally labelled marker 1 in image l is correctly matched to the true marker 1 in image 23 (i.e. the

point with spotID 153) in 12 out of the 15 cases. In two cases, marker 1 in both images remain unmatched. In the remaining one case, marker 1 in image l is incorrectly matched to marker 2 in image 23.

When transforming μ_l onto μ_{23} for $l \in \tau_1$ and $l \neq 23$, we found that the originally labelled marker 1 in image l is correctly matched to the true marker 1 in image 23 (i.e. the point with spotID 153) in 9 out of the 15 cases. In three cases, marker 1 in both images remains unmatched. In one case, marker 1 remains unmatched in image l, but marker 1 in image 23 is incorrectly matched to a nearby non-marker in image l. In the remaining two cases, marker 1 in image l is incorrectly matched to a nearby non-marker in image 23.

Conclusion

Within image 23, we reassign the point with spotID 153 as marker 1 and set the point with spotID 136 to be a non-marker.

In our case, we deal with more than a single pairwise comparison so we have more information than the methodology described within Chapter 2 would require. Because only a small proportion of comparisons highlight each of the other gross misallocations, we make the executive decision to leave the other highlighted markers as markers to allow σ_{ij} to be higher in future analyses.

We have previously concluded that the standard method should be used for further analyses. In this section, we have discovered a case where marker 1 is incorrectly matched to marker 2 in another image when marker 2 is not present in the first. For this reason, and because markers are included as a guide rather than for scientific interest, we include only corresponding markers between images in further analyses. That is, we follow the standard method described in Subsection 2.4.2 when discussing how to deal with missing markers.

Simulated gel data

Figure 3.9a and Figure 3.9b depict the $K = 12$ marker labels at the appropriate coordinates for a simulated $\mu^{'}$ and $x^{'}$ respectively. In this example, the labels of marker

$k = 1$ and marker $k = 9$ in $\mu^{'}$ have been 'accidentally' switched.

Figure 3.9c displays the initial affine transformation of $\mu^{'}$ onto $x^{'}$ when considering the originally allocated markers.

We input both $\mu^{'}$ and $x^{'}$ into steps 1–5 of the composite algorithm to estimate the one-to-one matching matrix, $\hat{M}$. The parameters within the algorithm are set to be the same as those used or established in the previous simulations. We estimate the variance in Equation (2.9) as $\hat{\sigma}^2 = 4.5^2$ and the proportion of correctly corresponding marker pairs in Equation (2.26) as $\hat{p}_M = 0.99$. The starting values for the transformation parameters, $A^{(0)}$ and $b^{(0)}$, are found using Equation (2.24). We use the final posterior probabilities, $\hat{p}$, to estimate the matches by fixing $\Delta = \hat{p}^T$.

We find that $\hat{M}_{kk} = 1$ for all $k \neq 1, 9$, so the correctly labelled markers are successfully matched. We also find that $\hat{M}_{19} = \hat{M}_{91} = 1$. That is, μ_1 is matched to x_9 and μ_9 is matched to x_1. The EM algorithm has correctly highlighted the incorrectly labelled markers.

Figure 3.9d displays the initial affine transformation of $\mu^{'}$ onto $x^{'}$ when considering the 10 remaining markers only. We can see that the truly corresponding markers are now much closer.

3.3.2 Investigating evidence of increased edge variance

We consider marker correspondences only when investigating evidence of increased edge variance. For our dataset, the width, w, and the height, h, of each image is unknown.

For each transformation of $\mu^{'}_{l_1 l_2}$ onto $\mu^{'}_{l_2 l_1}$ for $l_1, l_2 = 1, \ldots, 26$ and $l_1 \neq l_2$, we do the following.

- Calculate the residual between the kth corresponding marker pair as

$$r_k^{l_1 l_2} = \|\mu_k^{l_1 l_2} - A^{(0)} \mu_k^{l_2 l_1} - b^{(0)}\|,$$

  where $k = 1, \ldots, K_{l_1 l_2}$ and $K_{l_1 l_2}$ is the number of corresponding markers between

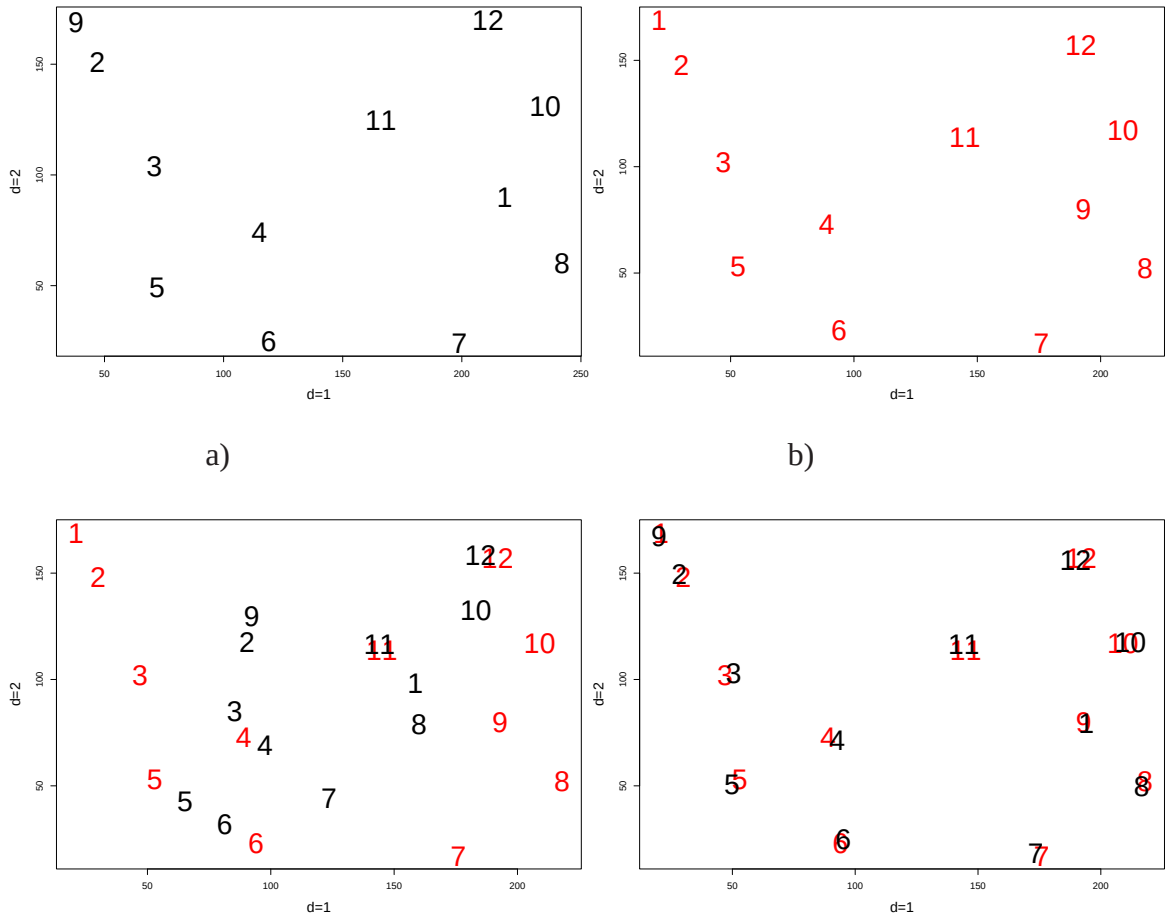


Figure 3.9: Figure displaying the $K = 12$ markers a) within $\mu^{'}$ b) and within $x^{'}$. Figure displaying the affine superimposition of the markers across images c) using the initial marker labels d) and using the updated marker labels.

μ_{l_1} and μ_{l_2}. The affine transformation parameters, $A^{(0)}$ and $b^{(0)}$, are found using Equation (2.24)

- Calculate the coordinates associated with this residual as

$$C_k^{l_1 l_2} = \frac{1}{2}\left(\mu_{k*}^{l_1 l_2} + \mu_{k*}^{l_2 l_1}\right),$$

where $*$ indicates the standardised coordinates so that every point in each image, μ_l for $l = 1, \ldots, 26$, is defined within a unit square.

Smoothing of the residuals at the standardised (and irregular) coordinates was performed by Gaussian kernel weighting with parameter 0.1. Figure 3.10 displays the colour-coded intensity plot with highlighted coordinates. We see evidence that the variance between corresponding markers, after superimposition, increases as the markers become closer to the top or right side of the image. We see further evidence that the variance between corresponding markers, after superimposition, decreases as the markers become closer to the bottom or left side of the image.

Conclusion

We have found evidence of an increased edge variance at the top-right corner of an image and a decreased edge variance at the bottom-left corner of an image. However, the width and height of each image within this dataset is unknown and the estimated values used above are unlikely to reflect the reality. For example, the image displayed in Figure 1.2 has ample space without points at each edge. Furthermore, the degree of variance will vary across images and fitting a global trend is unlikely to be very accurate.

For these reasons, we assume $\sigma_{ij}^2 = \sigma^2$ is constant in Equation (2.9) in all future analyses.

3.3.3 Real matching example

In this example we display the matches made when comparing two replicates, μ and x. We input the images into steps 1–5 of the composite algorithm. The starting values for the

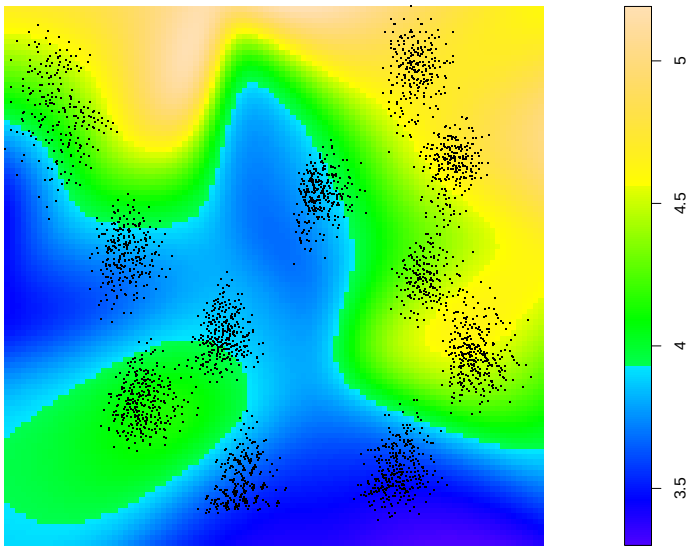


Figure 3.10: Colour-coded intensity plot displaying the smoothed residuals (performed by Gaussian kernel weighting) between corresponding markers at the standardised coordinates.

transformation parameters, $A^{(0)}$ and $b^{(0)}$, are found using Equation (2.24). We estimate the variance in Equation (2.9), σ^2, using Equation (2.25) with denominator K instead of ν.

The estimated transformation parameters are

$$\hat{A} = \begin{pmatrix} 0.9750 & -0.0506 \\ 0.0001 & 1.0047 \end{pmatrix},$$

and $\hat{b} = (-0.7138, 11.3564)^T$.

We explore the one-to-one matches made when $\Delta = \hat{p}^T$ and the matches made when $\Delta = D^*$, setting $d_T = 0.7\hat{\sigma}$.

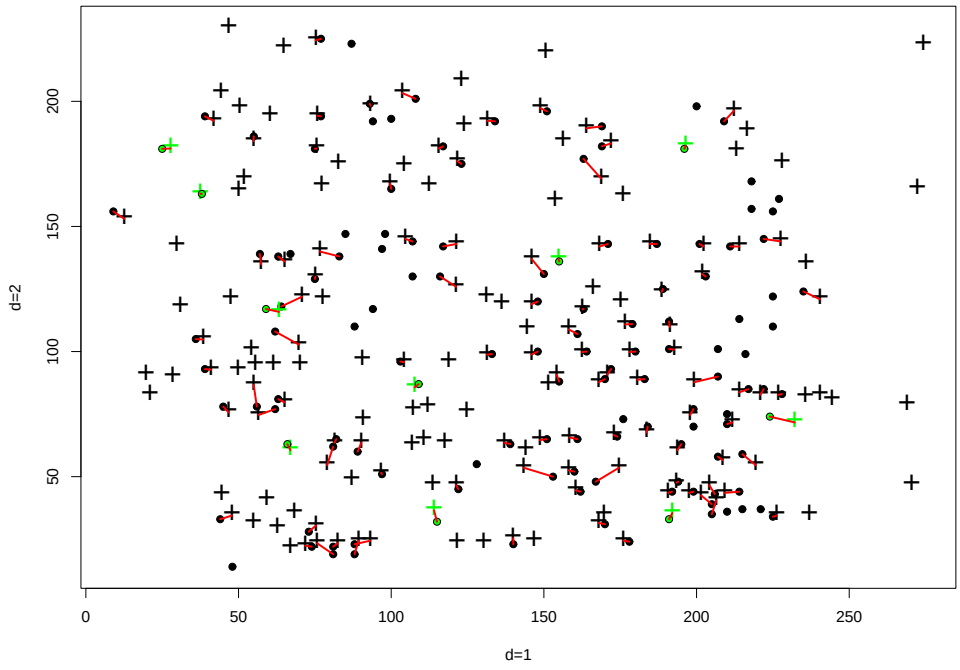
Both plots in Figure 3.11 display the final transformation of μ onto x and the matches made when $\Delta = \hat{p}^T$ and when $d_T = 0.7\hat{\sigma}$ in Figure 3.11a and Figure 3.11b respectively. We find that the estimated number of matches is $\hat{L} = 107$ when we set $\Delta = \hat{p}^T$ and $\hat{L} = 49$ when we set $d_T = 0.7\hat{\sigma}$ respectively. The maximum distance between two matched points in each case is 10.24 and 2.33 respectively.

3.3.4 Overall conclusions

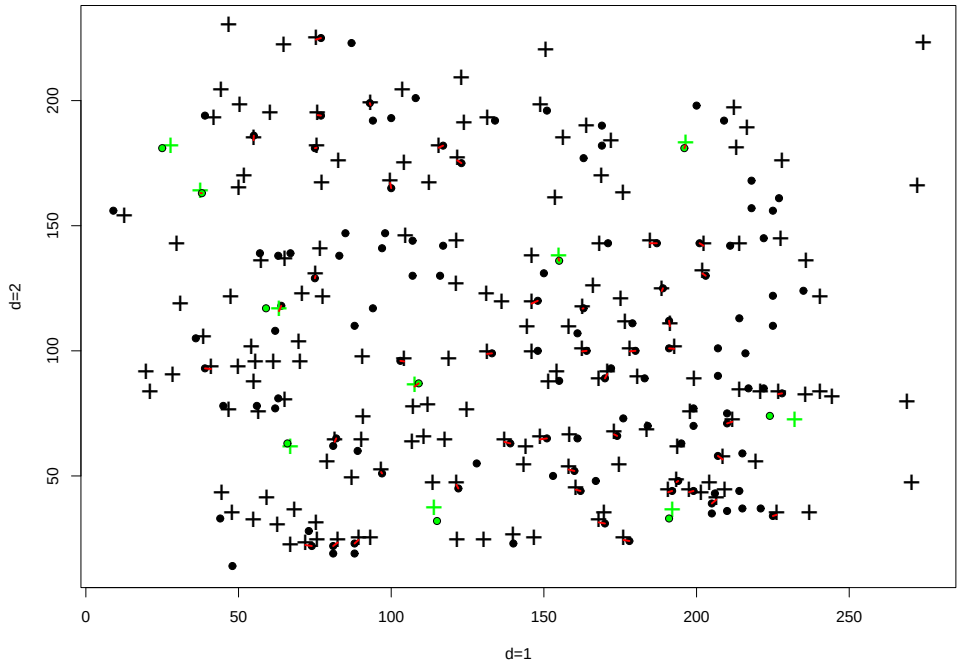
- The described methodology and appropriate parameters correctly highlighted a grossly misallocated marker in all comparisons. The marker was reallocated appropriately before further analyses.

- We found evidence of an increasing edge variance at the top and right side of the images. We found evidence of an decreasing edge variance at the bottom and left side of the images. However, as w and h are unknown and the warping between images is independent, we assign $\sigma_{ij}^2 = \sigma^2$ in future analyses.

- Using the final estimated posterior probabilities, $\hat{p}$, to define matches can often match points that are quite far apart. If a point x_j has a single nearby point in μ, the posterior probability of these two points matching will be quite dominant even



a)



b)

Figure 3.11: Figure showing the final transformation of μ onto x and the matches made when a) $\Delta = \hat{p}^T$ and b) $d_T = 0.7\hat{\sigma}$. The filled circles represent points in x and the crosses represent points in the transformed μ. Black indicates non-markers and green indicates markers. Matched points across images are joined by a red line.

though the points are not that close. Setting a distance tolerance, d_T, and $\Delta = D^*$ bypasses this problem which will become more prominant as the correspondence across two images decreases.

Chapter 4

Further analyses for image comparisons

4.1 Introduction

In Chapter 4 we explore how the methodology introduced in Chapter 2 can be used to pool data across replicates, to investigate the quality of a dataset and finally, how it can be implemented to highlight the differences in proteins across groups of images. We begin in Section 4.2 by describing how we can create a union of replicate images which can then be considered alone in further analyses to reduce the computational expense. In Section 4.3 we introduce the concept of image contamination and descibe how the data can be modelled to enable the inference of the contamination levels within a dataset. In Section 4.4 we introduce methodology to calculate a score that can be used to highlight proteins unique to one group of images.

4.2 Pooling data across replicate pairs

In this section we consider μ and x to denote two replicate images. Replicate images should be identical. However, due to gel warping and imperfections within the chemical procedure used to create the images, exact replicates are rarely produced.

To reduce computational workload, we can pool replicate information into one

single image which can be used in further analyses. Inputting μ and x into steps 1-6 of the composite algorithm in Subsection 2.3.4, we estimate the $(K+m+1)\times(K+n)$ one-to-one matching matrix, $\hat{M}$.

Let u_l be the $D\times 1$ vector containing the coordinates of the lth point in the union of μ and x, u. The points within u include points that are present in μ alone, points that are present in x alone and points that are present in both μ and x. We define u_l in each case.

- The number of points that remain unmatched in μ, i.e., that are present solely in μ, is

  $$m^* = K + m - \hat{L},$$

  where $\hat{L}$ is the estimated number of matches stated in Equation (2.21). Let ζ be a list of length m^* containing the increasing indices of unmatched points in μ. We set

  $$u_l = \hat{\hat{A}}\mu_{\zeta_l} + \hat{\hat{b}},$$

  for $l = 1,\ldots,m^*$, where $\hat{\hat{A}}$ and $\hat{\hat{b}}$ are the updated transformation parameters in Step 6 of the composite algorithm.

- The number of points that remain unmatched in x, i.e., that are present solely in x, is

  $$n^* = K + n - \hat{L}$$

  Let η be a list of length n^* containing the increasing indices of unmatched points in x. We set

  $$u_l = x_{\eta_{l-m^*}},$$

  for $l = m^*+1,\ldots,m^*+n^*$.

- Now to include the $\hat{L}$ matched points, i.e., the points present in both μ and x.

  Let φ_μ be a list of length $\hat{L}$ containing the increasing indices in $\{1,\ldots,K+m\}$ that are not present in ζ. Let φ_x be a list of length $\hat{L}$ containing the corresponding

indices in $\{1, \ldots, K+n\}$ of matched points in x that are not present in η. That is, if $\varphi_l^\mu = i$ and $\varphi_l^x = j$, then $\hat{M}_{ij} = 1$ for $l = 1, \ldots, \hat{L}$. We set

$$u_l = \frac{1}{2}\left(x_{\varphi^x_{l-m^*-n^*}} + \hat{\hat{A}}\mu_{\varphi^\mu_{l-m^*-n^*}} + \hat{\hat{b}}\right),$$

for $l = m^* + n^* + 1, \ldots, m^* + n^* + \hat{L}$.

Note: Pooling replicate data in this way is only useful when the error within images is small. If the images are greatly influenced by warping, for example, the union will be unlikely to represent the theoretical image represented by the two replicate images and information would be lost.

4.3 Image contamination

In this section we want to provide a method to measure the level of *contamination* within a set of images. The methodology we produce was inspired by a modification of 2-DE called *DIfference Gel Electrophoresis* (DIGE) [80]. DIGE is a chemical procedure used to compare two or three protein samples by tagging each sample with different coloured fluorescent dyes before mixing them and creating one single image. The third protein sample is usually a mix of the first two samples. The creation of a single gel bypasses the necessity of further computational analyses to assign matches across images and contamination is more easily distinguished from spots that represent true proteins.

Figure 4.1 shows a simplified example of an image produced by DIGE. In this example, two protein samples are tagged with either a blue or a red dye. A yellow dye is used to tag the proteins in a mixture of the two samples. We know that the black points represent false positive observations (created by dust on the image, for example) due to the absense of fluorescent dye. We know that the single yellow spots represent proteins that have failed to be observed in one or both of the first two samples - false negative observations. The yellow and blue spots denote proteins present in the first sample (and possibly false negative observations in the second sample). The yellow and red spots

denote proteins present in the second sample (and possibly false negative observations in the first sample). The yellow, blue and red spots denote proteins present in both the first sample and the second sample.

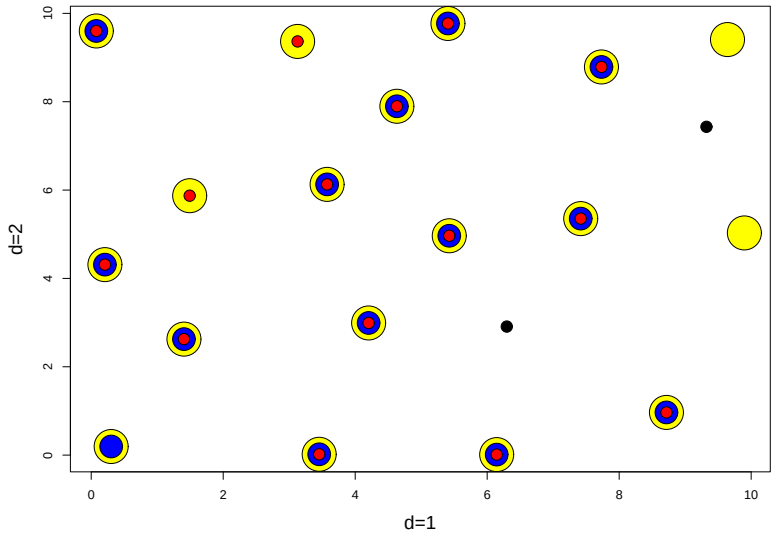


Figure 4.1: Figure displaying an example of a simplified image output by DIGE. The blue circles display the proteins present in the first sample and the red circles represent the proteins present in the second sample. The yellow circles display a union of the proteins present in the first two samples.

The limitations of DIGE include that it can only be used to compare a maximum of three protein samples and the technology is not yet commonly in general usage. In the following subsection we clearly define what we mean by image contamination before discussing how replicate data can be used to infer the level of contamination within a dataset. In this section we assume that images are free from warping and that correspondences across images are known, so that matching is not necessary. We should also note that, within this section, the presence of markers within an image is ignored.

4.3.1 Introduction to contamination

We consider image contamination to be the presence of *missing* or *imposter points* within an image.

- A missing point is a protein that should have been detected within an image, but has not been observed. Missing points are caused by the *limit of detection* of the chemical procedure used to produce the images.

- An imposter point is a point that is observed in an image, even though the protein corresponding to the point location should not have been detected. These points can be the result of dust caught in the gel before the image has been taken.

In this section, we let x denote some true image containing n points. This true image is what we would see in the absence of contamination. Let $\bar{x}$ be the observed image, containing $\bar{n}$ points. It is within $\bar{x}$ that contamination may exist.

Table 4.1 displays the four possibilities for the points present in x and the points observed in $\bar{x}$.

		Observed in $\bar{x}$	
		Yes	No
Present in x	Yes	True Positive	False Negative
	No	False Positive	True Negative

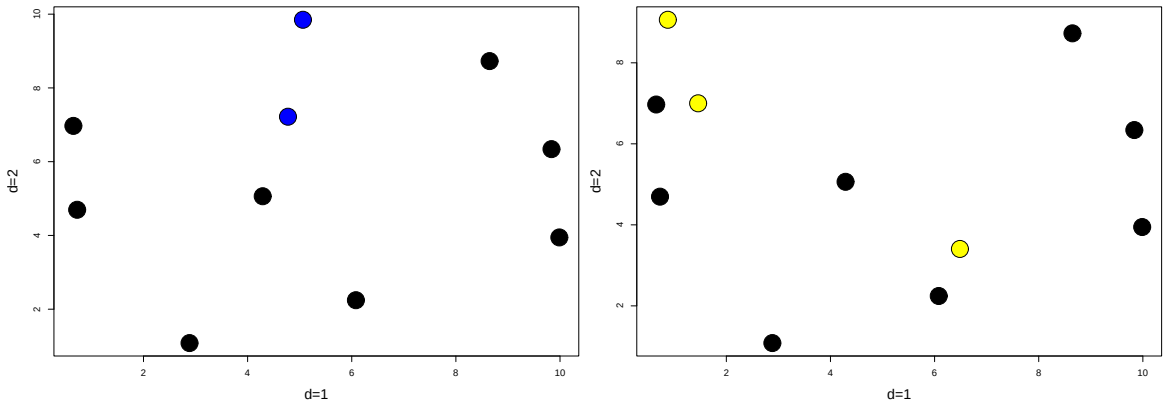
Table 4.1: Table displaying the possibilities of points observed or those failed to be observed in $\bar{x}$.

We can redefine contamination, i.e., missing points and imposter points, in terms of the true image, x, and the observed image, $\bar{x}$.

- **Missing points:** Points that are present in x but are not observed in $\bar{x}$, i.e., false negatives.

- **Imposter points:** Points that are absent from x but are observed in $\bar{x}$, i.e., false positives.

Figure 4.2a displays a simulated true image, x. Figure 4.2b displays a possible observed image, $\bar{x}$. In this example there are two false negative observations in $\bar{x}$ (highlighted in blue within x) and three false positive observations in $\bar{x}$ (highlighted in yellow).



a) b)

Figure 4.2: a) A simulated true image, x, and b) a possible observed image, $\bar{x}$. The false negative observations in $\bar{x}$ are highlighted in blue within x. The false positive observations in $\bar{x}$ are highlighted in yellow.

4.3.2 Contamination across replicates

Replicate images each represent the same true image, x, but are produced separately. In the absence of contamination, each replicate image would simply be identical to x. However if the observed images contain contamination, it is possible for a point to be observed in $\bar{x}_1$ but not observed in $\bar{x}_2$, for example, where $\bar{x}_1$ and $\bar{x}_2$ are two replicate images. There are only two possible explanations.

- The point is a false positive observation in $\bar{x}_1$.

- The point is a false negative observation (i.e.,missing) in $\bar{x}_2$.

Because $\bar{x}_1$ and $\bar{x}_2$ are replicates, we do not have to consider the possibility that a point may be a true positive in $\bar{x}_1$ or a true negative in $\bar{x}_2$. We know that the differences between replicates are a product of contamination alone.

Figure 4.3a displays the same true image, x, as displayed in Figure 4.2a. Figures 4.3b and 4.3c display two possible replicates, $\bar{x}_1$ and $\bar{x}_2$, both produced to represent x. In this example, there are two false negative observations in both $\bar{x}_1$ and $\bar{x}_2$ (highlighted in light-blue and dark-blue respectively within x). There are also three false positive observations in both $\bar{x}_1$ and $\bar{x}_2$ (highlighted in yellow). So only 6 true positive observations are present in both $\bar{x}_1$ and $\bar{x}_2$.

Next we introduce a possible model to represent image contamination.

4.3.3 Modelling contamination

Introduction

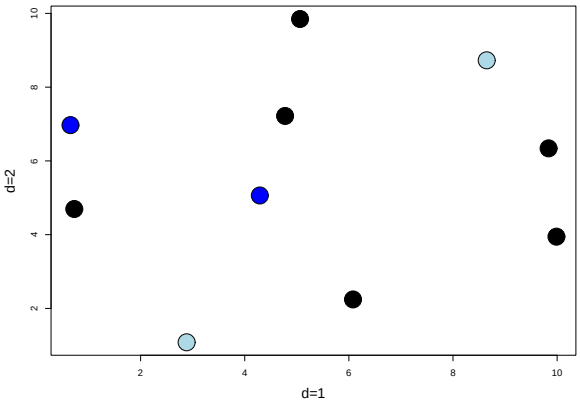
As stated previously, x denotes some true image containing n points. Let $\bar{x}_l$ denote the lth replicate, produced to represent x, containing $\bar{n}_l$ points for $l = 1, \ldots, R$. Let r be the number of times a point is observed in a union of the R replicate samples. For example, if a point is observed $r = R$ times in the union, then the point is observed in each of the R replicates.

Let ζ indicate whether a point is one of the n true points or whether it is a false point, i.e, an imposter point where

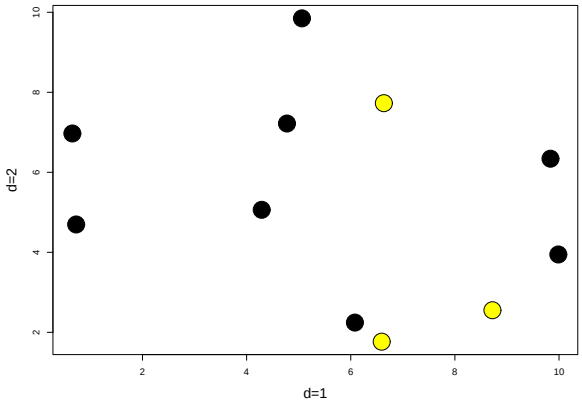
$$\zeta = \begin{cases} 1 & \text{if the point is true} \\ 0 & \text{if the point is false.} \end{cases} \tag{4.1}$$

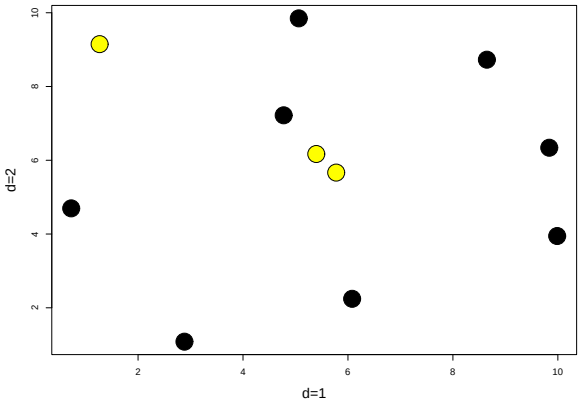
The probability we observe a point r times in a union is

$$p(r) = p(r|\zeta = 1)p(\zeta = 1) + p(r|\zeta = 0)p(\zeta = 0). \tag{4.2}$$



a)





b) c)

Figure 4.3: a) A simulated true image, x. Two possible observations of x, $\bar{x}_1$ and $\bar{x}_2$, are displayed in a) and b) respectively. The false negative points observations in $\bar{x}_1$ and $\bar{x}_2$ are highlighted in light-blue and dark-blue repectively within x. The false positive observations in $\bar{x}_1$ and $\bar{x}_2$ are highlighted in yellow.

Distribution of true points

We use a Binomial distribution to model the number of times a true point is observed in the union so that

$$r|\zeta = 1 \sim \text{Bin}(R, p_*), \tag{4.3}$$

where p_* is the probability a true point in x is observed in $\bar{x}_l$ for $l = 1, \ldots, R$.

Distribution of false points

Let C_l be the number of false points observed in the lth replicate for $l = 1, \ldots, R$. Assuming false points occur at random over a uniform surface, we can apply a Poisson distribution so that

$$C_l \sim \text{Po}(\lambda), \tag{4.4}$$

where λ is the rate of false points per image.

The number of points observed in the lth image, for $l = 1, \ldots, R$, is therefore distributed as

$$\bar{n}_l \sim \text{Bin}(n, p_*) + \text{Po}(\lambda). \tag{4.5}$$

We assume the contamination parameters, p_* and λ, to be dependent on the laboratory conditions and the person who created the dataset. We also assume that both p_* and λ are constant over all points and all images respectively.

Inputting the distributions applied in Equation (4.3) and (4.4), we can rewrite Equation (4.2) as

$$p(r) \propto \frac{R!}{r!(R-r)!} p_*^r (1-p_*)^{R-r} a + (1-a) I[r=1], \tag{4.6}$$

where $a = p(\zeta = 1)$, i.e., the probability an observed point is true and

$$I[r=1] = \begin{cases} 1 & \text{if the point is observed once in the union} \\ 0 & \text{otherwise.} \end{cases}.$$

Let v_{rj} be the number of points that are observed r times in the union of R replicates for $r = 0, \ldots, R$ and $j = 1, \ldots, J_r$, where J_r is the number of possible ways of choosing

r from the R replicates, arbitrarily ordered. As we only know correspondences between observed points, we do not know the total number of true points so we have

$$\sum_{r=1}^{R} J_r = \sum_{r=1}^{R} \frac{R!}{r!(R-r)!},$$

observations in total.

Because of the Binomial distribution applied in Equation (4.3) to the true points and the Poisson distribution applied in Equation (4.4) to the number of false points in an image, we can state the following.

- The number of points observed in r replicates, for $r = 2, \ldots, R$ and $j = 1, \ldots, J_r$, is distributed as

  $$v_{rj} \sim \text{Bin}(n, p_*^r(1-p_*)^{R-r}), \tag{4.7}$$

  and is therefore independent of λ. For each of the r distributions, we have J_r observed results.

- The number of points observed in x_j alone, for $j = 1, \ldots, R$, is distributed as

  $$v_{1j} \sim \text{Bin}(n, p_*(1-p_*)^{R-1}) + \text{Po}(\lambda), \tag{4.8}$$

  and is dependent on all three unknown parameters, n, p_* and λ.

To allow us to estimate the three unknown parameters, we assume the distributions stated in Equations (4.7) and (4.8) are independent over r and j.

We now show how we can estimate the total number of true points, n, and the two contamination parameters, p_* and λ.

4.3.4 Parameter estimation

We find n, p_* and λ that maximise the probability of observing v_{rj}. To do this we consider two methods. The first method provides numerical solutions for the three unknowns using the full dataset, i.e., considering all v_{rj} for $r = 1, \ldots, R$ and $j = 1, \ldots, J_r$. The second

method estimates n and p_* considering only v_{rj} for $r = 2, \ldots, R$, $j = 1, \ldots, J_r$ and finally estimates λ from v_{1j} for $j = 1, \ldots, R$. This method highlights the relationship between p_* and λ when $R = 2$, it provides analytical solutions when $R = 3$ and is less computationally expensive than method 1 over all R.

Method 1:

We can estimate the parameters numerically by finding n, p_* and λ that maximise

$$\prod_{r=1}^{R} \prod_{j=1}^{J_r} p(v_{rj}),$$

i.e., that maximise the full log-likelihood of all the observed data

$$\sum_{r=1}^{R} \sum_{j=1}^{J_r} \log p(v_{rj}), \tag{4.9}$$

where v_{rj} has the distribution defined in Equation (4.7) for $r = 2, \ldots, R$ and the distribution defined in Equation (4.8) for $r = 1$ and $j = 1, \ldots, J_r$. The probability of observing v_{1j} is stated later within Method 2 in Equation (4.13).

Using this method we cannot easily obtain analytical solutions, but we do not lose the information about n and p_* stored in the observed v_{1j}, for $j = 1, \ldots, R$, as we do in Method 2 described next.

Method 2:

To define analytical solutions or if we cannot solve for N, p_* and λ using Method 1, we can consider the following method as an alternative way to estimate the unknown parameters.

Equation (4.8) involves all three unknown parameters, n, p_* and λ and provides us with R observations from one distibution. Equation (4.7) involves only two unknowns, n and p_*, and provides us with observations from $R - 1$ distributions.

We can first use the observations with the distribution stated in Equation (4.7) to estimate n and p_* where possible. Finally, we use the observations with the distribution

stated in Equation (4.8) to estimate λ and any remaining unknowns. We look at the cases for $R = 2$, $R = 3$ and general $R \geq 3$ separately.

Estimating n and p_*

$\mathbf{R = 2}$ replicates

We begin by considering the observed points in the union of $R = 2$ replicates, $\bar{x}_1$ and $\bar{x}_2$.

- We observe v_{21} points in both $\bar{x}_1$ and $\bar{x}_2$.
- We observe v_{1j} points in $\bar{x}_j$ alone for $j = 1, 2$.

From Equation (4.7), we know that the estimate of n that maximises the probability of observing v_{21} is

$$\hat{n} = \frac{v_{21}}{\hat{p}_*^2}. \tag{4.10}$$

$\mathbf{R = 3}$ replicates

For $R = 3$ replicates, $\bar{x}_1$, $\bar{x}_2$ and $\bar{x}_3$, we observe

- v_{31} points in all $R = 3$ replicates.
- v_{2j} points in all replicates excluding $\bar{x}_j$ for $j = 1, 2, 3$.
- v_{1j} points in $\bar{x}_j$ alone for $j = 1, 2, 3$.

In this case, we have observations from two distributions stated in Equation (4.7) for $r = 2$ and $r = 3$. As the expected values of v_{2j} for $j = 1, 2, 3$ and v_{31} are

$$E[v_{2j}] = np_*^2(1 - p_*) \qquad \text{and} \qquad E[v_{31}] = np_*^3,$$

respectively, we can estimate n and p_* respectively as

$$\hat{n} = \frac{(\overline{v}_{2\cdot} + v_{31})^3}{v_{31}^2} \qquad \text{and} \qquad \hat{p}_* = \frac{v_{31}}{\overline{v}_{2\cdot} + v_{31}}, \tag{4.11}$$

where

$$\overline{v}_{2\cdot} = \frac{1}{3}\sum_{j=1}^{J_2=3} v_{2j}.$$

R $>$ 3 replicates

For $R > 3$, we have observations v_{rj} for $r = 2, \ldots, R$ and $j = 1, \ldots, J_r$ that are dependent solely on n and p_*. So we have two unknowns and more than two equations involving the two unknowns. In this case we can estimate n and p_* that maximise the joint distribution of v_{rj} for $r = 2, \ldots, R$ and $j = 1, \ldots, J_r$

$$\prod_{r=2}^{R} \prod_{j=1}^{J_r} p(v_{rj}),$$

i.e., that maximise the log likelihood

$$\sum_{r=2}^{R} \sum_{j=1}^{J_r} \log p(v_{rj}), \tag{4.12}$$

where the probability of v_{rj} is found from the distribution in Equation (4.7) for $r = 2, \ldots, R$ and $j = 1, \ldots, J_r$. As R increases, Equation (4.12) becomes increasingly complex so numerical solutions are easier to compute.

So, from the observed correspondences with the distribution in Equation (4.7) between R replicates, we can state the following.

- For $R = 2$, we know the relationship between n and p_* stated in Equation (4.10).
- For $R = 3$, we can estimate n and p_* analytically using Equations (4.11).
- For $R > 3$, we can estimate n and p_* numerically by maximising the log-likelihood of v_{rj} over $r = 2, \ldots, R$ and $j = 1, \ldots, J_r$ stated in Equation (4.12).

Estimating λ and any remaining unknowns

The conditional distribution of v_{1j} given C_j (see Equation (4.4)), for $j = 1, \ldots, R$, is dependent on the number of n real points that are observed only once, so that

$$v_{1j}|C_j \sim \text{Bin}(n, p_*(1 - p_*)^{R-1}).$$

The probability of observing v_{1j} points in x_j alone is therefore

$$p(v_{1j}) = \sum_{C_j=0}^{v_{1j}} p(v_{1j}|C_j)p(C_j)$$

$$= \sum_{C_j=0}^{v_{1j}} \frac{n!}{(v_{1j}-C_j)!(n-v_{1j}+C_j)!} \left[p_*(1-p_*)^{R-1}\right]^{v_{1j}-C_j} \left[1-p_*(1-p_*)^{R-1}\right]^{n-v_{1j}+C_j} \frac{e^{-\lambda}\lambda^{C_j}}{C_j!}. \tag{4.13}$$

Finally, we estimate any remaining unknowns for $R \geq 2$ by maximising the joint probability

$$\prod_{j=1}^{R} p(v_{1j}),$$

i.e., maximising the log likelihood

$$\sum_{j=1}^{R} \log p(v_{1j}), \tag{4.14}$$

where $p(v_{1j})$ is given in Equation (4.13).

4.3.5 Contamination within multiple replicate sets

Now let us consider that we have L sets of R replicates (note that here, L does not indicate the number of matches as it has previously). Let $\bar{x}_{lr}$ denote the rth image from the lth set of replicates containing $\bar{n}_{lr}$ points for $l = 1, \ldots, L$ and $r = 1, \ldots, R$. The lth set of R images is taken to represent the true image, x_l, containing n_l points, for $l = 1, \ldots, L$. Assuming that p_* and λ are constant over L, we can estimate p_*, λ and n_l for $l = 1, \ldots, L$.

Let v_{rj}^l be the number of points that are observed r times in the union of the lth set of R replicate samples for $r = 0, \ldots, R$, $j = 1, \ldots, J_r$ and $l = 1, \ldots, L$, where J_r is the number of possible ways of choosing r from the R replicates.

The distributions assigned to v_{rj}^l are similar to those assigned to v_{rj} in Equations (4.7) and (4.8) except we replace the n with n_l so that

$$v_{rj}^l \sim \text{Bin}(n_l, p_*^r(1-p_*)^{R-r}) \tag{4.15}$$

and

$$v_{1j}^l \sim \text{Bin}(n_l, p_*(1-p_*)^{R-1}) + \text{Po}(\lambda), \tag{4.16}$$

for $r = 0, \ldots, R$, $j = 1, \ldots, J_r$ and $l = 1, \ldots, L$.

In this case, we need to estimate p_*, λ and n_l for $l = 1, \ldots, L$ so we have a total of $L + 2$ unknowns. We consider two methods similar to those described previously.

Method 1

Similar to Equation (4.9), we can estimate all unknown parameters numerically by finding n_l, p_* and λ that maximise

$$\prod_{l=1}^{L}\prod_{r=1}^{R}\prod_{j=1}^{J_r} p(v_{rj}^l),$$

i.e., that maximise the full log-likelihood of all the observed data

$$\sum_{l=1}^{L}\sum_{r=1}^{R}\sum_{j=1}^{J_r} \log p(v_{rj}^l), \tag{4.17}$$

where v_{rj}^l has the distribution defined in Equation (4.15) for $r = 2, \ldots, R$ and Equation (4.16) for $r = 1$, $j = 1, \ldots, J_r$ and $l = 1, \ldots, L$, . The probability of v_{1j}^l is later stated in Method 2 in Equation (4.22).

Method 2:

Again we can look at $R = 2$, $R = 3$ and $R \geq 3$ separately.

For $R = 2$, the relationship stated in Equation (4.10) becomes

$$\hat{n}_l = \frac{v_{21}^l}{\hat{p}_*^2}. \tag{4.18}$$

For $R = 3$, the estimates stated in Equation (4.11) respectively become

$$\hat{n}_l = \frac{(\overline{v}_{2\cdot}^l + v_{31}^l)^3}{[v_{31}^l]^2} \qquad \text{and} \qquad \hat{p}_* = \frac{v_{31}^*}{\overline{v}_{2\cdot}^* + v_{31}^*}, \tag{4.19}$$

where for $l = 1, \ldots, L$,

$$v_{31}^* = \frac{1}{L}\sum_{l=1}^{L} v_{31}^l, \qquad \overline{v}_{2\cdot}^l = \frac{1}{3}\sum_{j=1}^{J_2=3} v_{2j}^l \qquad \text{and} \qquad v_{2\cdot}^* = \frac{1}{L}\sum_{l=1}^{L} \overline{v}_{2\cdot}^l.$$

For $R > 3$, the log likelihood stated in Equation (4.12) is now dependent solely on n_l and p_*, for $l = 1, \ldots, L$, and becomes

$$\sum_{l=1}^{L} \sum_{r=2}^{R} \sum_{j=1}^{J_r} \log p(v_{rj}^l), \tag{4.20}$$

where v_{rj}^l has the distribution stated in Equation (4.15).

For $R \geq 2$, the log likelihood stated in Equation (4.14) becomes

$$\sum_{l=1}^{L} \sum_{j=1}^{R} \log p(v_{1j}^l), \tag{4.21}$$

where v_{1j}^l has the distribution stated in Equation (4.16) so that

$$p(v_{1j}^l) =$$

$$\sum_{C_j=0}^{v_{1j}^l} \frac{n_l!}{(v_{1j}^l - C_j)!(n_l - v_{1j}^l + C_j)!} \left[p_*(1-p_*)^{R-1}\right]^{v_{1j}^l - C_j} \left[1 - p_*(1-p_*)^{R-1}\right]^{n_1 - v_{1j}^l + C_j} \frac{e^{-\lambda}\lambda^{C_j}}{C_j!}. \tag{4.22}$$

4.4 Scoring system for group comparisons

The main aim of this section is to develop a method that will highlight points that do not exist in both control and patient images or both normoxia and hypoxia treated images. In this section we introduce a scoring system for the comparison of two groups of images.

Let $\bar{\mu}^{(l)}$ and $\bar{x}^{(r)}$ denote the lth and rth image in group 1 and group 2 for $l = 1, \ldots, L$ and $r = 1, \ldots, R$ respectively. (Note that the definitions of L and R are different to those defined previously.)

The ith point in image l from group 1 is denoted by $\bar{\mu}_i^{(l)}$, for markers $i = 1, \ldots, K$ and non-markers $i = K + 1, \ldots, K + \bar{m}_l$. The jth point in image r from group 2 is denoted by $\bar{x}_j^{(r)}$, for markers $j = 1, \ldots, K$ and non-markers $j = K + 1, \ldots, K + \bar{n}_r$.

Say we wanted to highlight points that are likely to be present in group 1 images, but absent from group 2 images. To do this, we calculate a score as follows for each point $\bar{\mu}_i^{(l)}$, $i = 1, \ldots, \bar{m}_l$ and $l = 1, \ldots, L$.

4.4.1 Point presence in group 1

Transform $\bar{\mu}^{(l_2)}$ to fit $\bar{\mu}^{(l_1)}$ using steps 1–4 of the composite algorithm, described in Subsection 2.3.4 to match pairwise configurations, for $l_2 = 1, \ldots, L$ and $l_1 \neq l_2$.

Let $\hat{p}_{i0}^{l_1 l_2}$ denote the final estimated posterior probability that $\bar{\mu}_i^{(l_1)}$ is allocated to the coffin bin when $\bar{\mu}^{(l_2)}$ is transformed to fit $\bar{\mu}^{(l_1)}$. The probability that $\bar{\mu}_i^{(l_1)}$ is present in all L images in group 1 is

$$p_i^{(l_1)} = \frac{1}{L}\left[1 + \sum_{l_1 \neq l_2}(1 - \hat{p}_{i0}^{l_1 l_2})\right]. \tag{4.23}$$

4.4.2 Point absence in group 2

Transform $\bar{x}^{(r)}$ to fit $\bar{\mu}^{(l_1)}$ using steps 1–4 of the composite algorithm to match pairwise configurations, for $r = 1, \ldots, R$.

Let $\hat{q}_{i0}^{l_1 r}$ denote the final estimated posterior probability that $\bar{\mu}_i^{(l_1)}$ is allocated to the coffin bin when $\bar{x}^{(r)}$ is transformed to fit $\bar{\mu}^{(l_1)}$. The probability that $\bar{\mu}_i^{(l_1)}$ is present in all R images in group 2 is

$$q_i^{(l_1)} = \frac{1}{R}\left[\sum_{r=1}^{R}(1 - \hat{q}_{i0}^{l_1 r})\right]. \tag{4.24}$$

The probability $\bar{\mu}_i^{(l_1)}$ is absent from images in group 2 is simply $1 - q_i^{(l_1)}$.

We assign the following score to each point, $\bar{\mu}_i^{(l)}$,

$$S_i^{(l)} = w p_i^{(l)} + (1 - w)(1 - q_i^{(l)}), \tag{4.25}$$

for $i = 1, \ldots, \bar{m}_l$ and $l = 1, \ldots, L$. The weight, w, accounts for the number of images in each group as

$$w = \frac{L}{L + R}.$$

The use of the posterior matching probabilities provides a score $S_i^{(l)} \in \{0, 1\}$. The probability that $\bar{\mu}_i^{(l)}$ is present in group 1 images but absent in group 2 images increases as $S_i^{(l)} \to 1$.

Chapter 5

Experiments and Applications

5.1 Introduction

In Chapter 5 we analyse the properties and accuracy of the methodology introduced in Chapter 4. In Subsection 5.2.1, simulations are carried out to investigate the accuracy of contamination prediction with data from the assumed models. The accuracy is then investigated when point correspondences are inferred across replicates in Subsection 5.2.2. In Subsection 5.2.3 we explore how the score used to highlight points unique to one group of images is affected by varying levels of correspondence across groups as well as varying levels of contamination. Finally, in Section 5.3, we focus on the real data. Two replicate images are randomly chosen to provide an example of how a union image is created in Subsection 5.3.1. In Subsection 5.3.2 the correspondences across all replicate pairs are inferred and then used to estimate the level of contamination within the dataset. Finally, in Subsection 5.3.3, we highlight points that are likely to be present in patient images but absent from control images and vice versa. We also highlight the points likely to exist uniquely to images treated with normoxia or hypoxia. Finally, we reduce the variability within groups even further by considering the four subsets of images split by subject-type as well as treatment, before highlighting unique points within groups.

As in Chapter 3, where relevant we assume σ_{ij}^2 in Equation (2.9) is constant and

estimate it as $\hat{\sigma}^2 = 4.5^2$, which is approximately the median squared distance between two corresponding markers within the real dataset after all pairwise Procrustes transformations are performed. Alternatively, we estimate σ^2 using Equation (2.25) with denominator K instead of ν. Note that these estimates provide a conservative value of σ^2 and allow greater freedom for the distance between potential and known corresponding points. Though sensitivity tests are not carried out here, future work should involve a thorough exploration of the algorithm sensitivity to σ^2. The values presented here will be strongly dependent on the assigned σ^2.

When following the composite algorithm described in Subsection 2.3.4, we implement the standard method to assign the prior matching probabilities in Q. The starting values for the transformation parameters, $A^{(0)}$ and $b^{(0)}$, are found using Equation (2.24). We set $l = 10$ to define convergence in Equation (2.17).

5.2 Experiments

5.2.1 Accuracy of the contamination parameters within one set of replicates

We first investigate the prediction accuracy of n, p_* and λ over varying R and varying levels of contamination when simulating data from the assumed distributions. We focus on method 1 only.

We run the following simulation 1000 times.

1. First we simulate v_{rj} for $r = 1, \ldots, R$ and $j = 1, \ldots, J_r$. We randomly assign v_{1j} using Equation (4.8) and v_{rj} using Equation (4.7) for $r \neq 1$.

2. We then estimate n, p_* and λ as the values that maximise the likelihood stated in Equation (4.9).

We fix $n = 120$ and vary $p_* = 0.5, 0.75, 1$ and $\lambda = 0, 5, 10$. We consider values of

$R = 2, 4, 6$. We calculate the likelihood in Equation (4.9) for $\hat{n} \in [110, 130]$ at integer values, $\hat{p}_* \in [0, 1]$ at 0.05 intervals and $\hat{\lambda} \in [0, 20]$ at integer intervals.

Table 5.1 displays the mean number of true points estimated over the 1000 simulations and the standard deviance of the estimates from the true n, $\sigma_{\hat{n}}$, where

$$\sigma^2_{\hat{n}} = \frac{1}{1000} \sum_{i=1}^{1000} (\hat{n}_i - n)^2,$$

where $\hat{n}_i$ is the estimated number of true points at the ith simulation. Similarly, Table 5.2 displays the mean probability of observing a true point over the 1000 simulations and the standard deviance of the estimates from the true p_*, $\sigma_{\hat{p}}$. Table 5.3 displays the mean number of false points over the 1000 simulations and the standard deviance of the estimates from the true λ, $\sigma^2_{\hat{\lambda}}$.

		Average $\hat{n}$		
	λ	0	5	10
$p_* = 0.5$	$R = 2$	115.97	118.91	120.10
	$R = 4$	117.74	120.07	121.57
	$R = 6$	118.99	121.02	119.91
$p_* = 0.75$	$R = 2$	113.25	114.74	115.14
	$R = 4$	119.07	120.85	120.62
	$R = 6$	119.14	121.37	120.62
$p_* = 1$	$R = 2$	120.00	120.00	120.00
	$R = 4$	120.00	120.00	120.00
	$R = 6$	120.00	120.00	120.00

Standard deviance from n		
0	5	10
7.7427	6.9522	7.0396
7.8983	7.9627	8.5156
7.9639	7.9047	7.9283
7.7218	6.9457	6.3548
7.5952	7.7778	6.9442
7.5652	7.5712	7.7185
0.0000	0.0000	0.0000
0.0000	0.0000	0.0000
0.0000	0.0000	0.0000

Table 5.1: The first table displays the average of the estimated number of true points across R replicates, $\hat{n}$, for various values of p_* and λ. The second table displays the standard deviation of $\hat{n}$ from n, $\sigma_{\hat{n}}$, over the 1000 simulations.

		Average $\hat{p}_*$		
	λ	0	5	10
$p_* = 0.5$	$R = 2$	0.5010	0.5090	0.4980
	$R = 4$	0.5035	0.5010	0.4940
	$R = 6$	0.5080	0.4990	0.4975
$p_* = 0.75$	$R = 2$	0.7730	0.7725	0.7670
	$R = 4$	0.7555	0.7510	0.7455
	$R = 6$	0.7505	0.7480	0.7465
$p_* = 1$	$R = 2$	1.0000	1.0000	1.0000
	$R = 4$	1.0000	1.0000	1.0000
	$R = 6$	1.0000	1.0000	1.0000

Standard deviance from p_*		
0	5	10
0.0512	0.0414	0.0466
0.0322	0.0355	0.0402
0.0266	0.0284	0.0241
0.0389	0.0458	0.0389
0.0251	0.0236	0.0251
0.0151	0.0201	0.0181
0.0000	0.0000	0.0000
0.0000	0.0000	0.0000
0.0000	0.0000	0.0000

Table 5.2: The first table displays the average of the estimated probabilities of observing a true point, $\hat{p}_*$, for various values of R, p_* and λ. The second table displays the standard deviation of $\hat{p}_*$ from p_*, $\sigma_{\hat{p}}$, over the 1000 simulations.

		Average $\hat{\lambda}$		
	λ	0	5	10
$p_* = 0.5$	$R = 2$	1.68	5.28	10.34
	$R = 4$	0.57	4.80	9.49
	$R = 6$	0.25	4.89	10.07
$p_* = 0.75$	$R = 2$	2.34	7.36	11.48
	$R = 4$	0.23	4.90	9.67
	$R = 6$	0.03	5.08	9.98
$p_* = 1$	$R = 2$	0.00	5.18	10.42
	$R = 4$	0.00	5.12	10.13
	$R = 6$	0.00	4.97	10.03

Standard deviance from λ		
0	5	10
2.7005	4.0751	3.9287
1.0200	1.8641	2.4184
0.5412	1.2268	1.5472
3.7363	5.2436	4.4992
0.5222	1.1371	1.5923
0.1741	1.0347	1.2871
0.0000	1.5954	1.9592
0.0000	1.0445	1.6606
0.0000	0.9156	1.3744

Table 5.3: The first table displays the average of the estimated number of false points per image, $\hat{\lambda}$, for various values of R, p_* and λ. The second table displays the standard deviation of $\hat{\lambda}$ from λ, $\sigma_{\hat{\lambda}}$, over the 1000 simulations.

Conclusion

Generally, as $R \to \infty$, $\hat{n} \to n$, $\hat{p}_* \to p_*$ and $\hat{\lambda} \to \lambda$. We also see that as $R \to \infty$ or $p_* \to 1$, $\sigma^2_{\hat{n}}$, $\sigma^2_{\hat{p}}$ and $\sigma^2_{\hat{\lambda}} \to 0$.

That is, the accuracy of the inferred parameters increase as R increases and as $p_* \to 1$. Increasing λ does not have an obvious adverse effect on prediction accuracy.

5.2.2 Accuracy of the contamination parameters after inferring correspondence

The methodology, introduced in Section 4.3, to estimate unknown contamination parameters assumes point correspondences across images are known. In reality, correspondences across images have to be inferred because of the warping present in an image. Here we investigate the accuracy of the estimated contamination parameters after using the proposed methodology in Chapter 2 to infer on the corresponding points across images. As the real data we have contains only replicate pairs, we fix $R = 2$.

We run the following simulation 200 times for each case.

1. We randomly scatter $K+n$ points across a $w \times h$ uniform surface, where each point is set to be a minimum of 2 units from any other point. Note that we are simulating data similar to that given so we again consider an image to have K markers and a selection of non-markers.

2. We randomly select K markers from the $K+n$ points with the constraint that each marker must be a minimum distance of d_K from any other marker. The remaining n points are allocated as non-markers. These K markers and n non-markers represent the true image, x.

3. Now we allocate the non-markers in the observed images $\bar{x}_1$ and $\bar{x}_2$. Note that we fix the K markers to be observed in both images.

Let $\bar{n}_1^T$ and $\bar{n}_2^T$ denote the number of non-markers observed in $\bar{x}_1$ and $\bar{x}_2$ respectively. We simulate $\bar{n}_l^T$ using only the Binomial distribution within Equation (4.5). Then we randomly select the $\bar{n}_l^T$ from the N non-markers to represent the non-markers observed in $\bar{x}_l$ for $l = 1, 2$.

4. We add noise, $N(0, \tau^2/4)$, to each point coordinate within $\bar{x}_1$ and $\bar{x}_2$.

5. Now we add false points to both $\bar{x}_1$ and $\bar{x}_2$. Let $\bar{n}_1^F$ and $\bar{n}_2^F$ denote the number of false points allocated to $\bar{x}_1$ and $\bar{x}_2$ respectively. We simulate $\bar{n}_1^F$ using only the Poisson distribution stated in Equation (4.5) before randomly scattering them across the same $w \times h$ uniform surface used to create $\bar{x}_l$ for $l = 1, 2$. In this case there are no constraints on the distance between points.

   Note: The distribution stated in Equation (4.5) is now fully satisfied for both $\bar{n}_1$ and $\bar{n}_2$.

6. The $(K + \bar{n}_1) \times 2$ and $(K + \bar{n}_2) \times 2$ matrices, $\bar{x}_1$ and $\bar{x}_2$ respectively, are input into steps 1–5 of the composite algorithm introduced in Subsection 2.3.4 to estimate the one-to-one matching matrix, $\hat{M}$.

   Note: As p_* decreases and λ increases, the number of false positive matches made by the EM algorithm will increase as d_T increases if we set $\Delta = D^*$. As we do not know the level of contamination before the inferrence of matches, we assign matches using the final posterior probabilities by setting $\Delta = \hat{p}^T$, therefore forfeiting control of the number of output matches.

7. Let $\hat{v}_{21}$, $\hat{v}_{11}$ and $\hat{v}_{12}$ be the inferred values of v_{21}, v_{11} and v_{12} respectively. These values are calculated from the non-markers alone so that

$$\hat{v}_{21} = \sum_{i=K+1}^{K+\bar{n}_1} \sum_{j=K+1}^{K+\bar{n}_2} \hat{M}_{ij}, \qquad \hat{v}_{12} = \sum_{j=K+1}^{K+\bar{n}_2} \hat{M}_{0j}$$

   and $\hat{v}_{11} = \bar{n}_1 - \hat{v}_{21}$.

8. Finally we estimate $\hat{n}$, $\hat{p}_*$ and $\hat{\lambda}$ that maximize the log-likelihood in Equation (4.9) when applying method 1. For method 2, we find $\hat{n}$ in terms of $\hat{p}_*$ using the relationship defined in Equation (4.10), before estimating $\hat{p}_*$ and $\hat{\lambda}$ that maximise the log-likelihood stated in Equation (4.14).

We fix $n = 30$, $K = 3$, $w = 257/2$, $h = 191/2$ and $d_K = 25$. We estimate the variance in Equation (2.9) as $\hat{\sigma}^2 = 4.5^2$ and set $\tau = \hat{\sigma}$. We vary the contamination levels by considering $p_* \in [0.5, 1]$ at intervals of 0.05 and $\lambda \in \{0, 10\}$ at integer intervals. In step 8, we calculate the method 1 log-likelihood for $\hat{n} \in [0, 40]$ at integer values, $\hat{p}_* \in [0.05, 1]$ at intervals of 0.05 and $\hat{\lambda} \in [0, 20]$ at integer values. We calculate the method 2 log-likelihood for $\hat{p}_* \in [0.05, 1]$ at intervals of 0.05 and $\hat{\lambda} \in [0, 20]$ at integer values.

Discussion

Figure 5.1 displays the mean values of $\hat{n}$, $\hat{p}_*$ and $\hat{\lambda}$ over the 200 simulations for both methods 1 and 2. Figure 5.2 displays the standard error of the estimates from the true values, again for both methods 1 and 2.

We can see that method 1 generally estimates $\hat{p}_* \approx 1$ for all considered levels of contamination. Therefore the error between the estimates $\hat{p}_*$ and the true p_* increases as p_* decreases. The estimated number of non-markers, $\hat{n}$, decreases as both p_* and λ decrease with the error between $\hat{n}$ and n becoming increasingly larger. Generally $\hat{\lambda}$ decreases at a slower rate than λ decreases so that the error between $\hat{\lambda}$ and λ increases.

Method 2 provides low estimates for λ for all considered levels of contamination, with the error between $\hat{\lambda}$ and λ increasing as λ increases. The estimated number of non-markers, $\hat{n}$, decreases as both p_* and λ decrease with the error between $\hat{n}$ and n becoming increasingly larger. We can see that $\hat{p}_*$ decreases at a slower rate than p_* decreases so that the error increases between $\hat{p}^*$ and p^* increases.

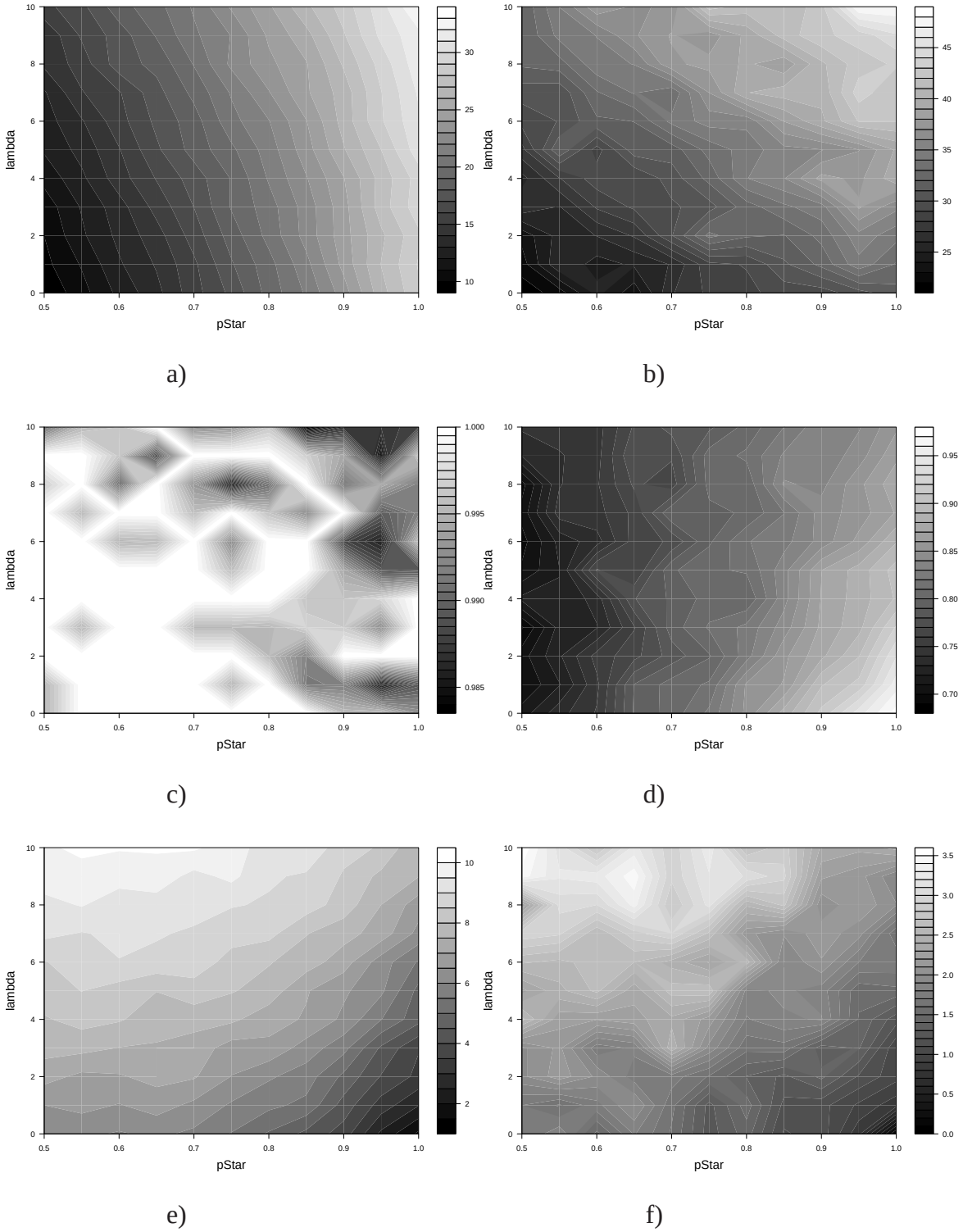


Figure 5.1: For λ against p^*, Figures a) and b) display the mean values of $\hat{n}$, Figures c) and d) display the mean values of $\hat{p}_*$ and Figures e) and f) display the mean values of $\hat{\lambda}$ for methods 1 and 2 respectively over the 200 simulations. Each figure is a contour plot where a greyscale is used to illustrate the various means.

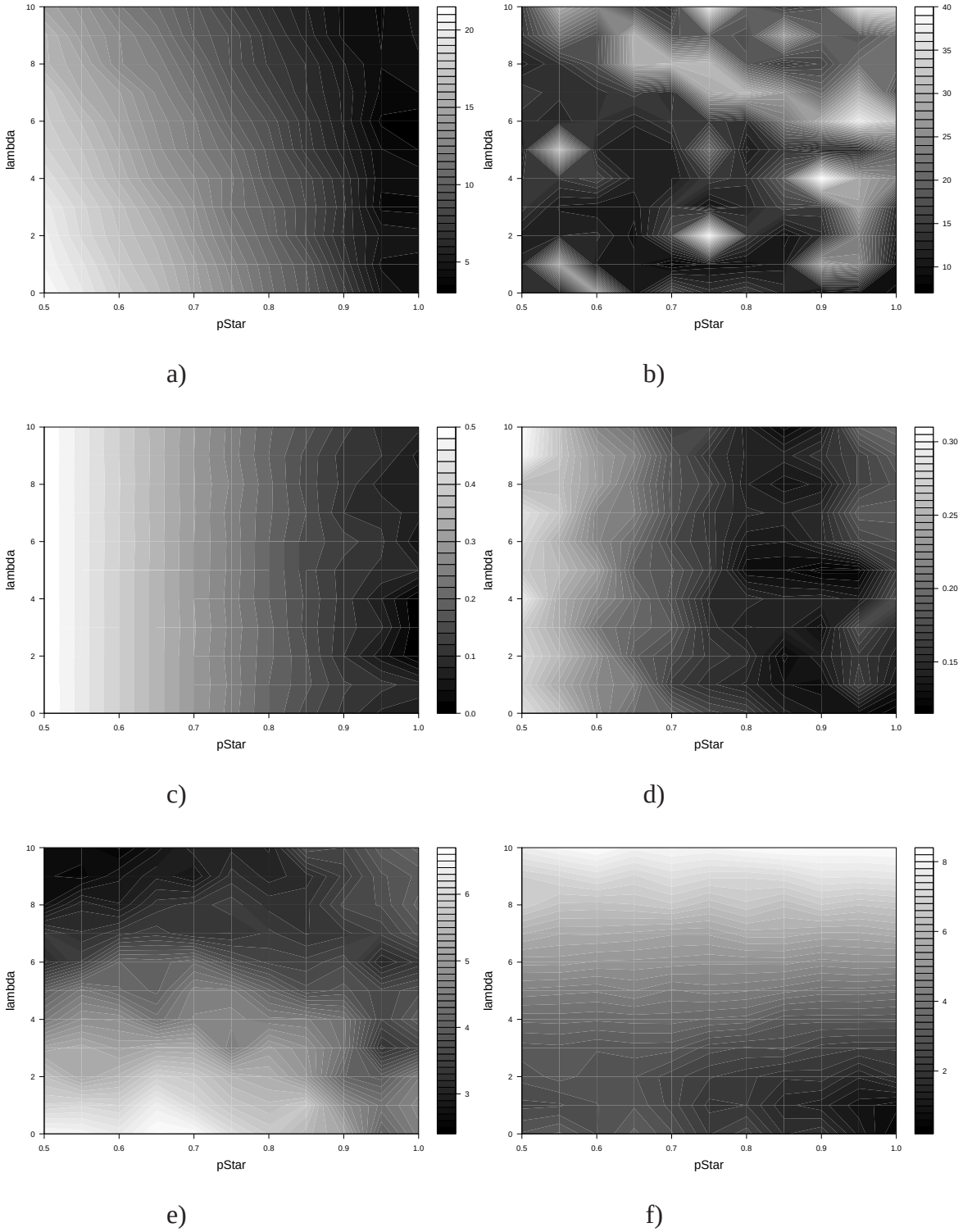


Figure 5.2: For λ against p^*, Figure a) and b) display the standard error of $\hat{n}$ from n, Figures c) and d) display the standard error of $\hat{p}_*$ from p_* and Figures e) and f) display the standard error of $\hat{\lambda}$ from λ for methods 1 and 2 respectively over the 200 simulations. Each figure is a contour plot where a greyscale is used to illustrate the various errors.

Conclusion

Method 1 generally predicts $\hat{p}_* \approx 1$. Alternatively method 2 generally predicts a low $\hat{\lambda}$ over all considered levels of contamination. The previous simulations showed that generating data from the assumed distributions will provide good estimates of n, p_* and λ for $R = 2$. However, when inferring on point correspondence across $R = 2$ replicates, there is not enough information to provide good estimates of contamination. When testing the real data in Subsection 5.3.2, we see that only a relationship between p_* and λ can be inferred.

5.2.3 Estimating the appropriate score threshold

Here we look at how the score (indicating points unique to a particular group of images) is affected by varying levels of correspondence across groups or varying levels of contamination. We want to highlight an appropriate threshold, p_T, above which scores should be considered. We run simulations separately for varying correspondence and contamination levels.

Let p_C denote the proportion of corresponding non-markers across group 1 images and group 2 images. The number of corresponding non-markers across images in the different groups is $N = np_C$.

Varying point correspondence across images

We vary p_C and carry out 200 simulations for each case.

1. We randomly scatter $K + 2n - N$ points across a $w \times h$ uniform surface, where each point is set to be a minimum of 2 units from any other point.

2. We randomly select K true markers from the $K+2n-N$ points with the constraint that each marker must be a minimum distance of d_K from any other marker. Let

μ_k and x_k contain the coordinates of marker k in μ and x respectively, for $k = 1, \ldots, K$.

3. From the remaining $2n - N$ points, we randomly select N points to represent the corresponding non-markers across μ and x. So μ_i and x_i contain the coordinates of corresponding non-markers for $i = K + 1, \ldots, K + N$.

4. Finally, we randomly split the remaining $2(n - N)$ points equally between μ and x so that μ_i and x_j contain the coordinates of arbitrarily labelled points in μ and x, for $i, j = K + N + 1, \ldots, K + n$, that do not have corresponding points in x and μ respectively.

5. We fix $\bar{\mu}^{(l)} = \mu$ for $l = 1, \ldots, L$ to create the group 1 images and $\bar{x}^{(l)} = x$ for $r = 1, \ldots, R$ to create the group 2 images.

6. We add noise, $N(0, \tau^2/4)$, to each point coordinate within $\bar{\mu}^{(l)}$ and $\bar{x}^{(r)}$ for $l = 1, \ldots, L$ and $r = 1, \ldots, R$ respectively.

7. Using the output final posterior matching probabilities, we calculate the probabilities stated in Equations (4.23) and (4.24), $p_i^{(l)}$ and $q_i^{(l)}$, for $l = 1$ and $i = 1, \ldots, K + n$.

   We transform $\bar{\mu}^{(l)}$ onto $\bar{\mu}^{(1)}$, for $l = 2, \ldots, L$, by inputting both into steps 1–4 of the composite algorithm described in Subsection 2.3.4 to produce the final posterior matching probabilities, $\hat{p}^{1l}$. We transform $\bar{x}^{(r)}$ onto $\bar{\mu}^{(1)}$, for $r = 1, \ldots, R$, by inputting both into steps 1–4 of the composite algorithm to produce the final posterior matching probabilities, $\hat{q}^{1r}$.

8. Inputting $p_i^{(1)}$ and $q_i^{(1)}$ into Equation (4.25), we calculate the score for each point in $\bar{\mu}^{(1)}$, where $S_i^{(1)}$ is the score for point i in the first image in group 1.

9. Finally we calculate the proportion of correctly highlighted points unique to group

1 images as $p_{TP} = 0$ for $p_C = 1$ and for $p_C \neq 1$,

$$p_{TP} = \frac{n_{TP}}{n_{TP} + n_{FP}} = \frac{\sum_{i=K+N+1}^{K+n} I[S_i^{(1)} > p_T]}{\sum_{i=K+1}^{K+n} I[S_i^{(1)} > p_T]},$$

where

$$I[S_i^{(1)} > p_T] = \begin{cases} 1 & \text{if } S_i^{(1)} > p_T \\ 0 & \text{if } S_i^{(1)} \leq p_T \end{cases}.$$

We fix $n = 30$, $K = 3$, $w = 257/2$, $h = 191/2$ and $d_K = 25$. We estimate the variance in Equation (2.9) as $\hat{\sigma}^2 = 4.5^2$ and set $\tau = \hat{\sigma}$. We also fix $L = R = 13$ to mimic the comparisons between patients/controls and treatments used to create the real data. We vary the point correspondence across groups by considering $p_C \in [0, 1]$ at intervals of 0.1. We explore the highlighted points when fixing $p_T \in [0.49, 1]$ at intervals of 0.01.

Discussion

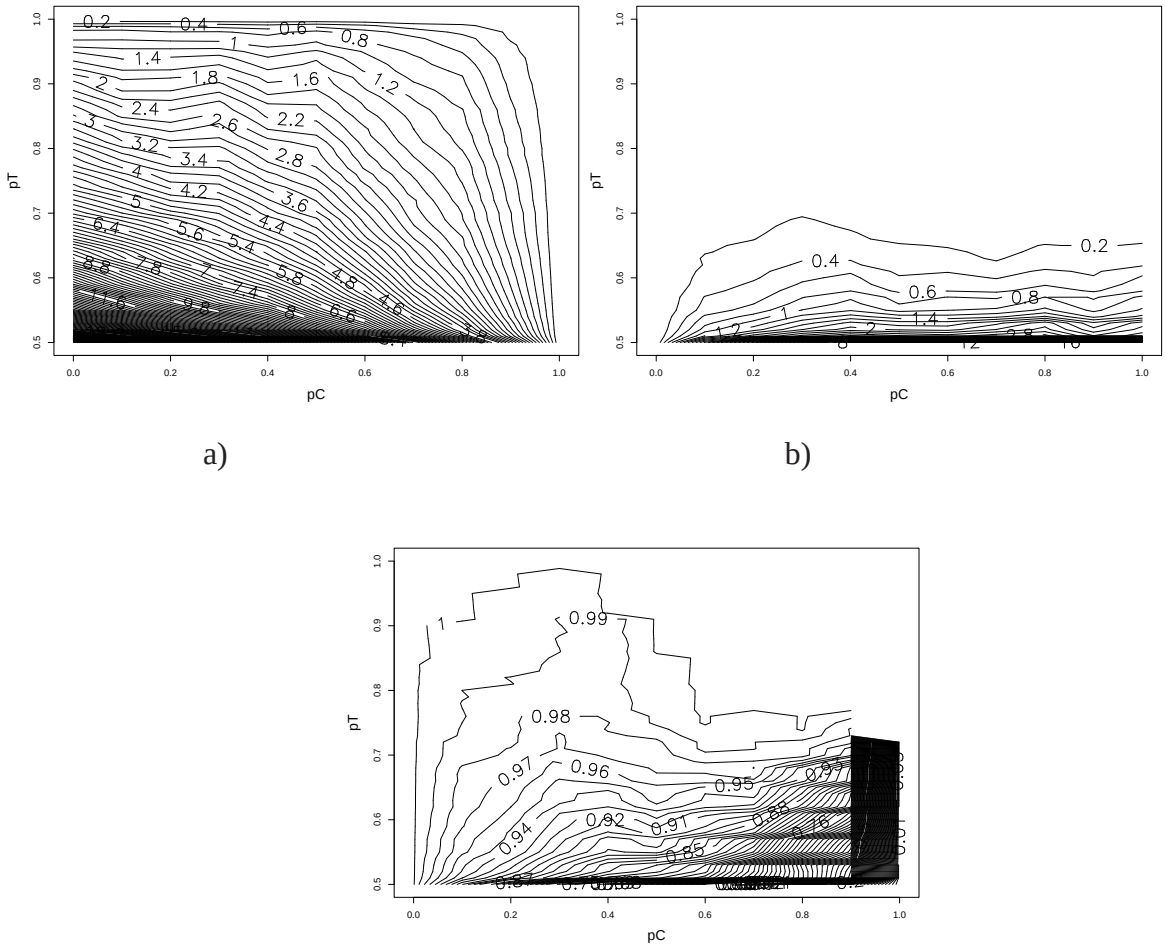
Figure 5.3 displays contours of n_{TP}, n_{FP} and p_{TP} for p_T against p_C. The number of true positives increase as both p_C and p_T decrease. The number of false positives also increase as p_T decreases, but increase as p_C increases. The proportion of true positives increase as p_C decreases and p_T increases.

Conclusion

When applying a threshold of $p_T \approx 0.7$, over 97% of the highlighted points are true positive observations over all p_C. A decreasing amount of points with $S_i^{(1)} > p_T$ indicates an increasing similarity across images in group 1 and group 2 images.

Varying contamination levels

Finally we fix $p_C = 1$ and vary the level of contamination within the following 100 simulations.



a) b)

c)

Figure 5.3: Figure showing a) n_{TP}, b) n_{FP} and c) p_{TP} for p_T against p_C.

1. We randomly scatter $K + m$ points across a $w \times h$ uniform surface, where each point is set to be a minimum of 2 units from any other point.

2. We randomly select K markers from the $K+m$ points with the constraint that each marker must be a minimum distance of d_K from any other marker. The remaining m points are allocated as non-markers. These K markers and m non-markers represent the true image, μ. We set $x = \mu$ so that the true image for group 1 is equivalent to the true image for group 2.

3. Now we allocate the non-markers observed in $\bar{\mu}^{(l)}$ for $l = 1, \ldots, L$ and in $\bar{x}^{(r)}$ for $r = 1, \ldots, R$. Note that the K markers are observed in all images.

   Let $\bar{m}_l^T$ and $\bar{n}_r^T$ denote the number of non-markers observed in $\bar{\mu}^{(l)}$ and $\bar{x}^{(r)}$ respectively. We simulate $\bar{m}_l^T$ and $\bar{n}_r^T$ separately using only the Binomial distribution within Equation (4.5). Then we randomly select $\bar{m}_l^T$ and $\bar{n}_r^T$ from the m non-markers to represent the non-markers observed in $\bar{\mu}^{(l)}$ and $\bar{x}^{(r)}$ respectively.

4. We add noise, $N(0, \tau^2/4)$, to each coordinate of the non-markers observed in $\bar{\mu}^{(l)}$ and $\bar{x}^{(r)}$ for $l = 1, \ldots, L$ and $r = 1, \ldots, R$ respectively.

5. Let $\overline{m}_l^F$ and $\overline{n}_r^F$ denote the number of false points in $\bar{\mu}^{(l)}$ and $\bar{x}^{(r)}$ respectively. Using only the Poisson distribution stated in Equation (4.5), we simulate $\overline{m}_l^F$ and $\overline{n}_r^F$ before randomly scattering them across the same $w \times h$ uniform surface separately for $\bar{\mu}^{(l)}$ and $\bar{x}^{(r)}$ respectively. In this case there are no constraints on the distance between points.

   Note: In this case $\bar{m}_l = \bar{m}_l^T + \overline{m}_l^F$ and $\bar{n}_r = \bar{n}_r^T + \overline{n}_r^F$ for $l = 1, \ldots, L$ and $r = 1, \ldots, R$.

6. Using the output posterior matching probabilities, we calculate the probabilities stated in Equations (4.23) and (4.24), $p_i^{(l)}$ and $q_i^{(l)}$, for $l = 1$ and $i = 1, \ldots, \bar{m}_1$.

We transform $\bar{\mu}^{(l)}$ onto $\bar{\mu}^{(1)}$, for $l = 2, \ldots, L$, by inputting both into steps 1–4 of the composite algorithm described in Subsection 2.3.4 to produce the final posterior matching probabilities, $\hat{p}^{1l}$. We transform $\bar{x}^{(r)}$ onto $\bar{\mu}^{(1)}$, for $r = 1, \ldots, R$, by inputting both into steps 1–4 of the composite algorithm to produce the final posterior matching probabilities, $\hat{q}^{1r}$.

7. Inputting $p_i^{(1)}$ and $q_i^{(1)}$ into Equation (4.25), we calculate the score for each point in $\bar{\mu}^{(1)}$, where $S_i^{(l)}$ is the score for point i in the first image in group 1.

8. Finally we calculate the proportion of incorrectly highlighted points unique to group 1 images as
$$p_{FP} = \frac{1}{\bar{m}_1} \sum_{i=1}^{\bar{m}_1} I[S_i^{(l)} > p_T].$$
Note that the true images, μ and x, are identical. Any point highlighted as unique to group 1 images is therefore incorrectly highlighted.

We fix $N = 30$, $K = 3$, $w = 257/2$, $h = 191/2$ and $d_K = 25$. We estimate the variance in Equation (2.9) as $\hat{\sigma}^2 = 4.5^2$ and set $\tau = \hat{\sigma}$. We vary the contamination levels by considering $p_* \in [0.5, 1]$ at intervals of 0.1 and $\lambda \in \{0, 10\}$ at intervals of 2. We calculate p_{FP} when setting $p_T = 0.7$

Conclusion

Over all considered contamination levels, the maximum value of the mean proportion of false positives (over the 100 simulations) was $\bar{p}_{FP} = 0.007$. So, when $L = R = 13$ and setting $p_T = 0.7$, the unknown contamination levels do not have a negative influence on the points highlighted to exist uniquely in one group.

5.2.4 Overall conclusions

- The prediction accuracy of n, p_* and λ is good when simulating data from the assumed distributions, even when considering only $R = 2$ replicates.

- When inferring on the corresponding points across $R = 2$ replicates, method 1 estimates $\hat{p}_* \approx 1$ and method 2 povides low estimates of $\hat{\lambda}$. In this case, later analyses on the real data suggests that there is enough information to infer a relationship between $\hat{p}_*$ and $\hat{\lambda}$ when $R = 2$, although not enough information to produce explicit solutions.

- The higher the probability threshold, p_T, the higher the proportion of true positive points highlighted as unique to group 1 images. A threshold of $p_T = 0.7$ is recommended when analysing the real data and should not be negatively affected by the proportion of corresponding points across groups or the level of contamination within the dataset.

5.3 Applications

5.3.1 Example of a union

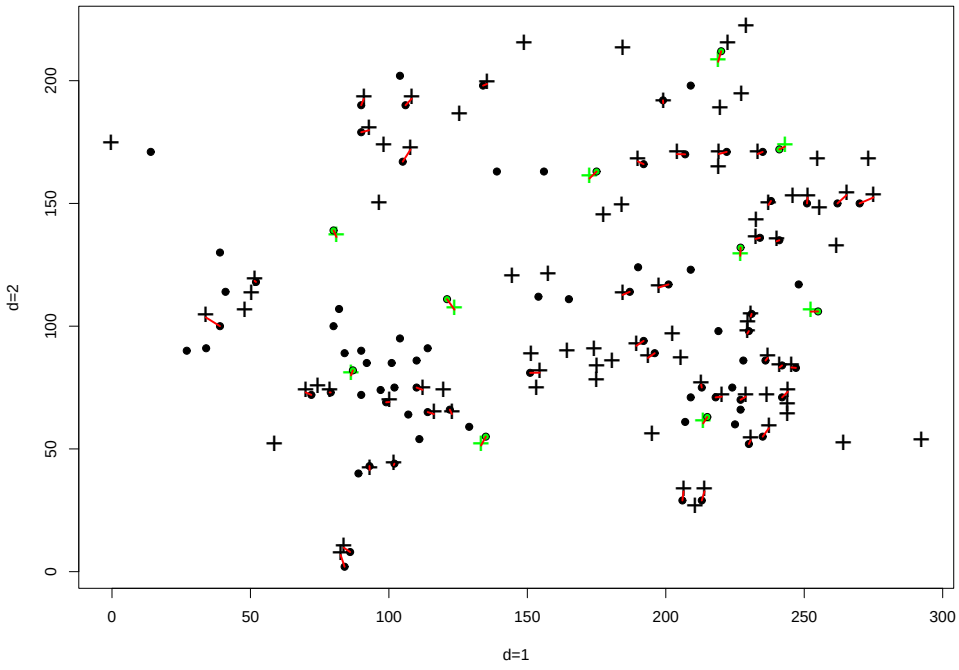
In this example we create a union of two replicates, μ and x. We input the images into steps 1–6 of the composite algorithm described in Subsection 2.3.4. We estimate the variance in Equation (2.9), σ^2, using Equation (2.25) with denominator K instead of ν. We explore the one-to-one matches made when $\Delta = \hat{p}^T$.

The final estimated transformation parameters are

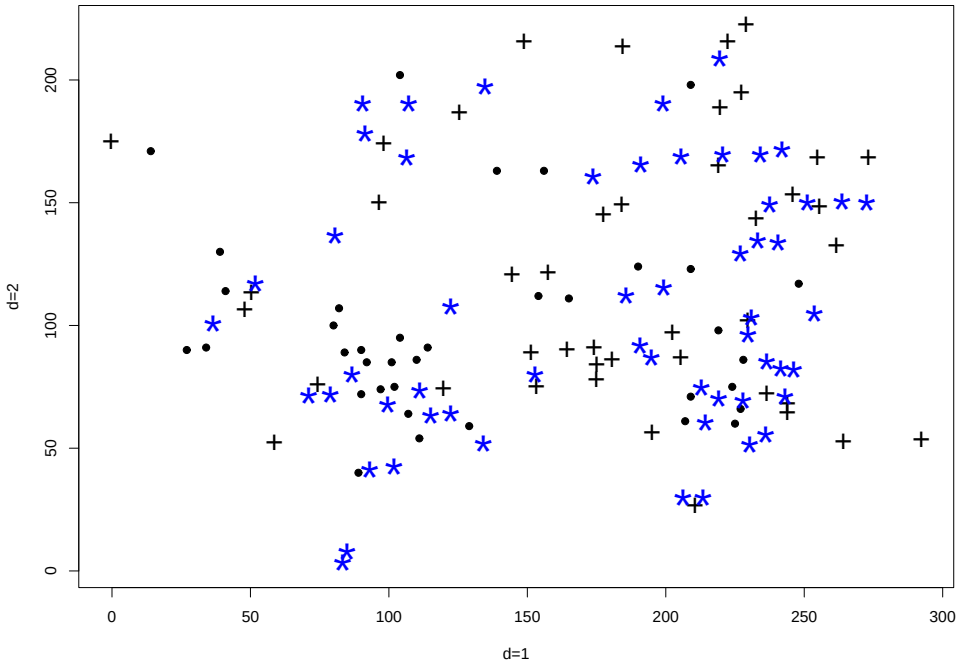
$$\hat{\hat{A}} = \begin{pmatrix} 1.0815 & 0.0217 \\ 0.0018 & 0.9885 \end{pmatrix},$$

and $\hat{\hat{b}} = (-22.4579, -47.7393)^T$.

Both plots in Figure 5.4 display the final transformation of μ onto x. Figure 5.4a displays the matches inferred. Figure 5.4b displays the union of μ and x.



a)



b)

Figure 5.4: a) Figure showing the final transformation of μ onto x and the matches made. The filled circles represent points in x and the crosses represent points in the transformed μ. Black indicates non-markers and green indicates markers. Matched points across images are joined by a red line. b) Figure showing the union of μ and x. The crosses indicate points unique to μ, the filled circles indicate points unique to x and the blue stars indicate points that are observed in both μ and x.

l	$\bar{n}_1^l$	$\bar{n}_2^l$	$\hat{v}_{21}^l$	$\hat{v}_{11}^l$	$\hat{v}_{12}^l$
1	89	83	46	43	37
2	157	134	115	42	19
3	198	142	137	61	5
4	152	141	107	45	34
5	141	148	120	21	28
6	112	114	73	39	41
7	106	109	65	41	44
8	94	99	72	22	27
9	96	92	57	39	35
10	166	125	97	69	28

Table 5.4: Table displaying $\bar{n}_1^l$, $\bar{n}_2^l$, $\hat{v}_{21}^l$, $\hat{v}_{11}^l$ and $\hat{v}_{12}^l$ for $l = 1, \ldots, L = 10$.

5.3.2 Estimating contamination levels

Here we estimate the contamination levels using the $L = 10$ sets of $R = 2$ replicates we have in the real dataset. Let $\bar{x}_{lr}$ represent the rth replicate from the lth set, for $l = 1, \ldots, L = 10$ and $r = 1, 2$. First we estimate v_{21}^l, v_{11}^l and v_{12}^l for each of the L replicate pairs.

We input $\bar{x}_{l1}$ and $\bar{x}_{l2}$ into steps 1–5 of the composite algorithm described in Subsection 2.3.4 for $l = 1, \ldots, L = 10$. We estimate the variance in Equation (2.9), σ^2, using Equation (2.25) with denominator K instead of ν. We estimate the one-to-one matching matrix, M, by setting $\Delta = \hat{p}^T$.

Table 5.4 displays $\bar{n}_1^l$, $\bar{n}_2^l$, $\hat{v}_{21}^l$, $\hat{v}_{11}^l$ and $\hat{v}_{12}^l$ for each of the $L = 10$ replicate pairs.

Note: These values consider the non-markers only, as within the simulations.

We first consider each replicate set separately before finding global solutions using all replicate pairs.

Local Solutions

For each of the $L = 10$ replicate pairs, we do the following.

For method 1, we estimate n_l, p_* and λ that maximize the log-likelihood in Equation (4.9). We calculate the log-likelihood for $\hat{n}_l \in [0, a_l]$ at integer values, $\hat{p}_* \in [0, 1]$ at intervals of 0.01 and $\hat{\lambda} \in [0, 100]$ at integer values. Here, $a_l = 50 + \max(\bar{n}_1^l, \bar{n}_2^l)$.

For method 2, we find $\hat{n}_l$ in terms of $\hat{p}_*$ using the relationship defined in Equation (4.10), before maximising the log-likelihood stated in Equation (4.14). We calculate the log-likelihood for $\hat{p}_* \in [0.01, 1]$ at intervals of 0.01 and $\hat{\lambda} \in [0, 100]$ at integer values.

Table 5.5 displays the estimated parameters when applying method 1 and method 2. Figure 5.5 displays contours of the method 2 likelihood in Equation (4.14) for λ against p_* for $l = 1, \ldots, L = 10$ (note that we only display the method 2 likelihood because it is dependent on the two contamination parameters alone).

Conclusion

In this case, from Table 5.5 we can see that generally $\hat{p}_* \approx 1$, with an exception for the first replicate pair in method 2 where $\hat{\lambda} \approx 0$. The contours in Figure 5.5 each show a ridge of maxima indicating the data provides a relationship between $\hat{p}_*$ and $\hat{\lambda}$ rather than explicit solutions. The similarity of the contours across replicate pairs supports the assumption that p_* and λ are constant across images in the dataset. However, the contours also indicate a poor quality of the given dataset.

Global solutions

Now we combine information across the $L = 10$ replicate pairs when estimating the unknown parameters.

For method 1, we estimate n_l, p_* and λ that maximize the log-likelihood in Equation (4.17). We calculate the log-likelihood for $\hat{n}_l \in [0, a_l]$ at integer values, $\hat{p}_* \in [0, 1]$ at intervals of 0.01 and $\hat{\lambda} \in [0, 100]$ at integer values. Here, $a_l = 50 + \max(\bar{n}_1^l, \bar{n}_2^l)$.

For method 2, we find $\hat{n}_l$ in terms of $\hat{p}_*$ using the relationship defined in Equation

l	$\hat{n}_l$	$\hat{p}_*$	$\hat{\lambda}$
1	10	1	45
2	11	1	79
3	10	1	99
4	9	1	76
5	9	1	71
6	10	1	56
7	8	1	53
8	9	1	47
9	10	1	48
10	10	1	83

l	$\hat{n}_l$	$\hat{p}_*$	$\hat{\lambda}$
1	277	0.19	2
2	11	0.98	78
3	10	1.00	99
4	9	1.00	76
5	9	0.99	70
6	10	1.00	56
7	8	1.00	53
8	9	1.00	47
9	10	1.00	48
10	10	1.00	83

Table 5.5: Tables containing the estimated parameters, $\hat{n}_l$, $\hat{p}_*$ and $\hat{\lambda}$ when considering each of the $L = 10$ replicate pairs separately. The table to the left displays the results when applying method 1 and the table to the right displays the results when applying method 2.

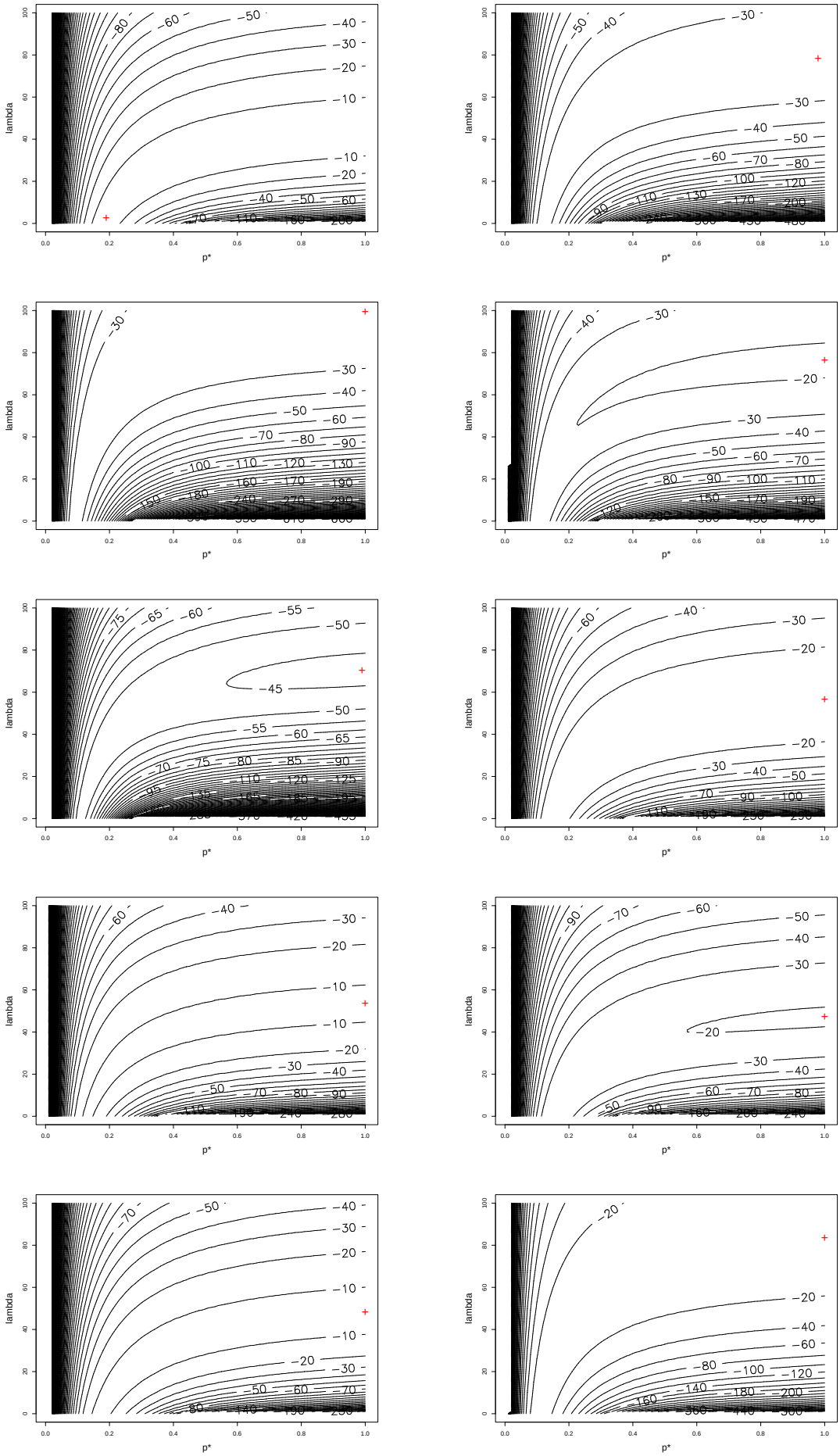


Figure 5.5: Contours of the method 2 likelihood for λ against p_* for each of the $L = 10$ replicate pairs. The red crosses represent the estimated parameters, $\hat{p}_*$ and λ, in each case.

l	1	2	3	4	5	6	7	8	9	10
Method 1 $\hat{n}_l$	80	202	248	184	160	114	95	81	90	216
Method 2 $\hat{n}_l$	50	125	149	116	130	79	71	78	62	105

Table 5.6: Tables containing the estimated number of points in each of the $L = 10$ true images, $\hat{n}$, for method 1 and method 2 globally.

(4.18), before maximising the log-likelihood stated in Equation (4.21). We calculate the log-likelihood for $\hat{p}_* \in [0.01, 1]$ at intervals of 0.01 and $\hat{\lambda} \in [0, 100]$ at integer values.

Table 5.6 displays the estimated $\hat{n}_l$ for $l = 1, \ldots, L$ when applying either method 1 or method 2. For method 1, the estimated contamination parameters, p_* and λ, are respectively

$$\hat{p}_* = 0.26 \qquad \hat{\lambda} = 37.$$

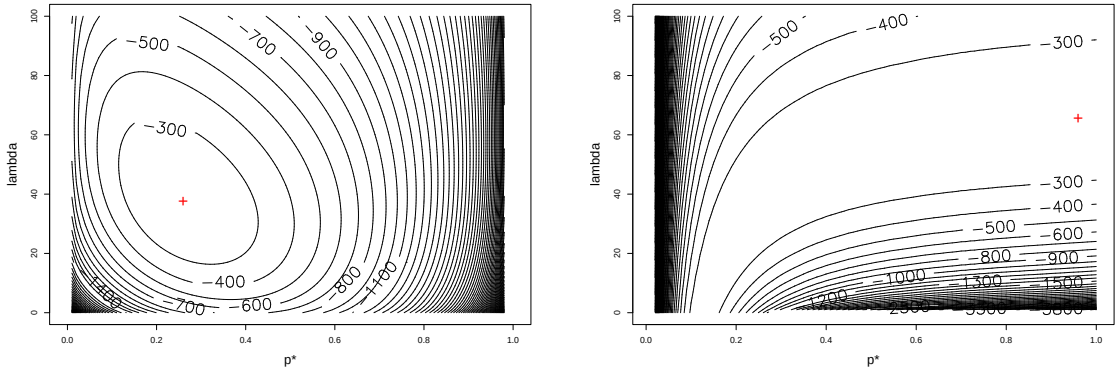
For method 2, the estimated contamination parameters, p_* and λ, are respectively

$$\hat{p}_* = 0.96 \qquad \hat{\lambda} = 65.$$

Figure 5.6 displays contours of the global likelihood for λ against p_* for a) method 1 (assuming that $n_l = \hat{n}_l$ for $l = 1, \ldots, L$) and b) method 2.

Conclusion

We can see that the solutions for n_l for $l = 1, \ldots, L$, p_* and λ differ greatly across methods, with method 2 predicting $p_* \approx 1$. The contour in Figure 5.6b, which illustrates the method 2 likelihood, again shows a ridge of maxima indicating the data provides a relationship between $\hat{p}_*$ and $\hat{\lambda}$ rather than explicit solutions. The contour illustrating the method 1 likelihood indicates explicit solutions when all information is considered to estimate p_*, λ and n_l for $l = 1, \ldots, L$. This solution is also present within the ridge of maxima shown in the contour displaying the method 2 likelihood. The resulting contamination parameters again indicate a poor quality of images within our dataset.



a) b)

Figure 5.6: Contours of the global likelihood for λ against p_* for a) method 1 and b) method 2. The red cross represents the estimated parameters, $\hat{p}_*$ and λ, in each case.

5.3.3 Highlighting unique points within image groups

Table 5.7 displays the number of images we have for each subject-type and treatment.

	Treatment		
	Normoxia	Hypoxia	Total
Control	7	6	13
Patient	6	7	13
Total	13	13	

Table 5.7: Table displaying the number of images we have for each subject-type and treatment.

First, we calculate the score to highlight points that are present in

- **A1:** control images but absent in patient images.
- **A2:** patient images but absent in control images.
- **A3:** normoxic images but absent in hypoxic images.

- **A4:** hypoxic images but absent in normoxic images.

In each of the above analyses, $L = R = 13$. Finally, we reduce variability within groups even further by separating the data in four subsets and calculating the score to highlight points that are present in

- **B1:** normoxic control images but absent in normoxic patient images ($L = 7$ and $R = 6$).
- **B2:** normoxic patient images but absent in normoxic control images ($L = 6$ and $R = 7$).
- **B3:** hypoxic control images but absent in hypoxic patient images ($L = 6$ and $R = 7$).
- **B4:** in hypoxic patient images but absent in hypoxic control images ($L = 7$ and $R = 6$).
- **B5:** in normoxic control images but absent in hypoxic control images ($L = 7$ and $R = 6$).
- **B6:** in hypoxic control images but absent in normoxic control images ($L = 6$ and $R = 7$).
- **B7:** in normoxic patient images but absent in hypoxic patient images ($L = 7$ and $R = 6$).
- **B8:** in hypoxic patient images but absent in normoxic patient images ($L = 7$ and $R = 6$).

For each analysis, we transform $\bar{\mu}^{(l)}$ onto $\bar{\mu}^{(1)}$, for $l = 2, \ldots, L$, by inputting both into steps 1–4 of the composite algorithm described in Subsection 2.3.4 to produce the final posterior matching probabilities, $\hat{p}^{1l}$. We transform $\bar{x}^{(r)}$ onto $\bar{\mu}^{(1)}$, for $r = 1, \ldots, R$, by inputting both into steps 1–4 of the composite algorithm to produce the final posterior

matching probabilities, $\hat{q}^{1r}$. We estimate the variance in Equation (2.9), σ^2, using Equation (2.25) with denominator K instead of ν.

Table 5.8 displays the index of each image present in the dataset. These indices are then used to indicate the image containing the 5 top scoring points in analyses A (Tables 5.9) and analyses B (Tables 5.10).

	Control	Initial	Replicate	Patient	Initial	Replicate
Hypoxia	1	1	2	1	5	6
Normoxia		3	4		7	8
Hypoxia	2	9	×	2	11	×
Normoxia		10	×		12	13
Hypoxia	3	14	15	3	18	19
Normoxia		16	17		20	21
Hypoxia	4	22	23	4	25	×
Normoxia		24	×		26	×

Table 5.8: Table displaying indices of the 26 images within the dataset.

The highest score is 0.9418 which highlights spotID 112 in image 10 as being the most likely point to be present in normoxic controls images but absent in hypoxic controls. Figure 5.7a-f display the superimposition of each remaining normoxic control image onto image 10. Figure 5.8a-f display the superimposition of each hypoxic control image onto image 10. Each figure is magnified onto the point of interest. In each case, a red 'circle' with a radius equal to twice the RMSD, $\hat{\sigma}$, surrounds spotID 112 in image 10.

A1		
Image	Point	Score
23	171	0.7748
23	168	0.7682
23	163	0.7644
15	124	0.7311
16	116	0.7020

A2		
Image	Point	Score
7	103	0.7525
12	108	0.7453
18	73	0.7317
8	110	0.7155
12	113	0.7142

A3		
Image	Point	Score
3	107	0.7203
10	89	0.7161
3	167	0.6987
7	107	0.6900
3	161	0.6849

A4		
Image	Point	Score
18	84	0.7729
19	85	0.7695
9	102	0.7633
15	92	0.7564
19	57	0.7557

Table 5.9: Table showing the top five scoring points in analyses A.

B1		
Image	Point	Score
23	6	0.9115
23	11	0.9077
10	6	0.8929
24	62	0.8781
17	7	0.8760

B2		
Image	Point	Score
20	67	0.7629
7	157	0.7473
20	59	0.7391
7	86	0.7005
7	97	0.6871

B3		
Image	Point	Score
15	14	0.7992
15	13	0.7834
2	4	0.7689
1	5	0.7687
1	3	0.7659

B4		
Image	Point	Score
18	73	0.8905
5	206	0.8788
12	106	0.8483
5	134	0.8339
11	148	0.8339

B5		
Image	Point	Score
10	112	0.9418
10	24	0.9165
10	19	0.8883
10	110	0.8733
3	161	0.8665

B6		
Image	Point	Score
15	14	0.7629
1	61	0.7598
15	13	0.7520
15	92	0.7195
22	1	0.7087

B7		
Image	Point	Score
8	150	0.6945
7	87	0.6900
21	26	0.6857
8	18	0.6851
7	13	0.6757

B8		
Image	Point	Score
6	15	0.8722
18	84	0.8447
19	73	0.8431
6	133	0.8417
18	85	0.8353

Table 5.10: Table showing the top five scoring points in analyses B.

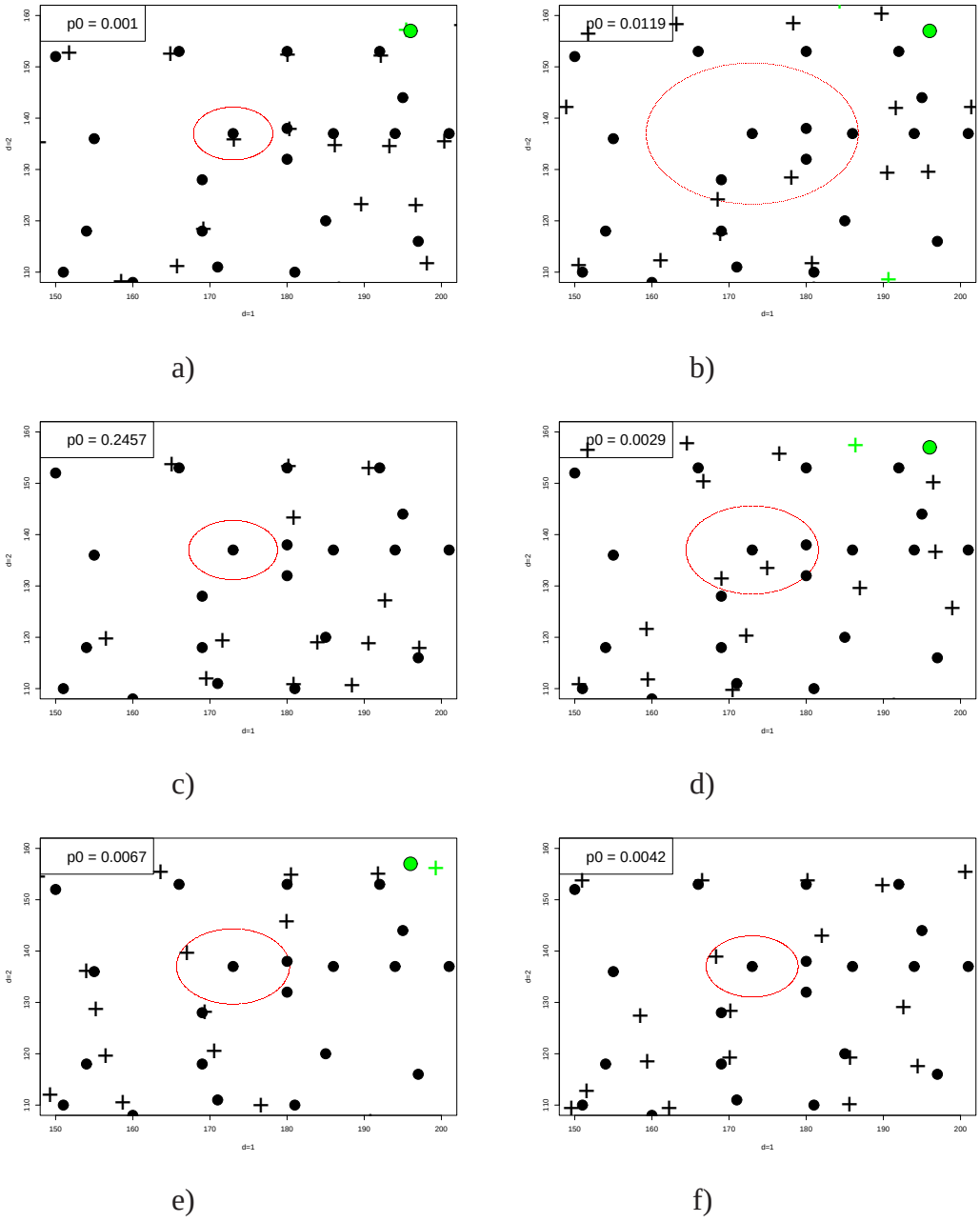


Figure 5.7: Figures displaying the final superimposition of each of the 6 remaining normoxic control images onto image 10. The filled circles represent points in image 10 and the crosses represent points in the transformed second image. Black indicates non-markers and green indicates markers. The radius of the red 'circle' surrounding point 112 in image 10 is equal to twice the standard deviation, $\hat{\sigma}$, used within the model to provide the superimposition.

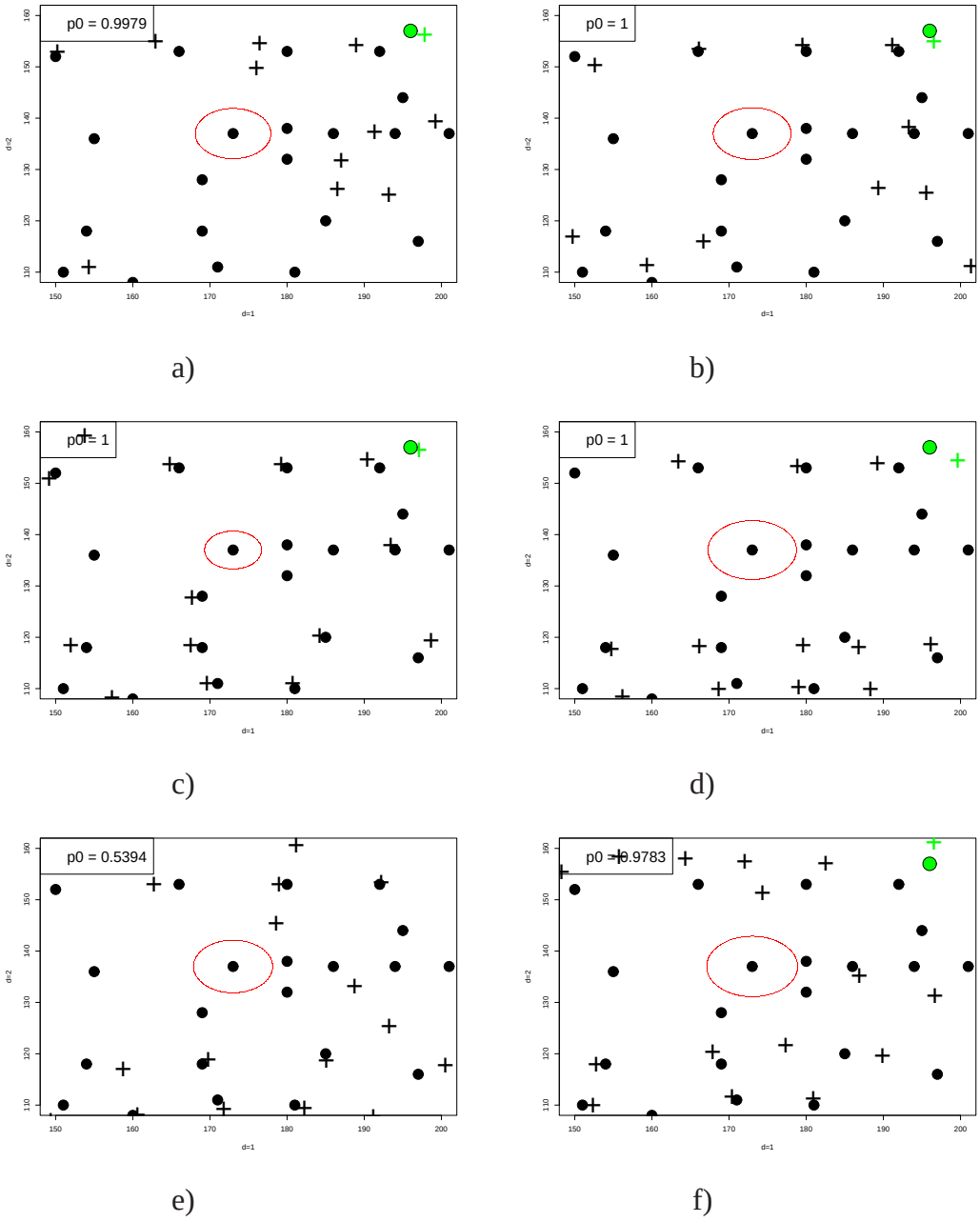


Figure 5.8: Figures displaying the final superimposition of each of the 6 hypoxic control images onto image 10. The filled circles represent points in image 10 and the crosses represent points in the transformed second image. Black indicates non-markers and green indicates markers. The radius of the red 'circle' surrounding point 112 in image 10 is equal to twice the standard deviation, $\hat{\sigma}$, used within the model to provide the superimposition.

Chapter 6

Predicting toxicity by shape similarity

6.1 Introduction

In Chapter 6 we test the hypothesis that the potential toxicity of a pesticide is related to the shape similarity between the pesticide and the substrate, ACh, of the protein, AChE, to which they both bind. In Section 6.2, we illustrate the structure of ACh and depict the general structures of a carbamate and an organophosphate (OP), the two families of pesticides considered within these analyses. We also display the reaction that takes places between each ligand and AChE before describing the concept driving the development of the shape similarity measure. In Section 6.3, we discuss how we can measure the shape similarity between ACh and a given pesticide. We introduce the data considered within the analyses, including the molecular conformations and known biological indicators of toxicity in Section 6.4. Finally, in Section 6.5, we explore the significance of the developed shape similarity measure as a toxicity predictor and compare it to the significance of the known biological indicators of toxicity. We also compare the accuracy of the toxicity predictions when applying our model to the accuracy when implementing a previously developed online predictor.

6.2 Ligand structures, reaction and shape similarity concept

6.2.1 Structures and reactions

Here we describe the structures of the different ligands under consideration and illustrate the reaction that takes place when each ligand type binds to AChE.

Figure 6.1 displays the structure of an ACh molecule. It is a small molecule with only 10 non-hydrogen atoms (relative to the 4143 non-hydrogen atoms within the protein, AChE). The number of non-hydrogen atoms within the considered pesticides range from 7 to 28 with an average of around 16.

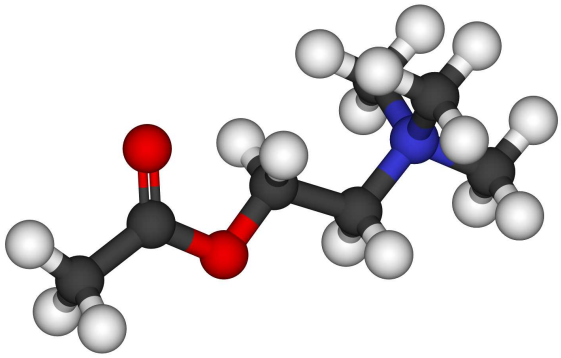
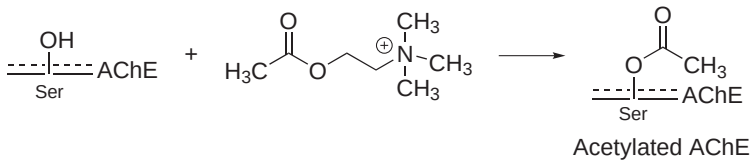
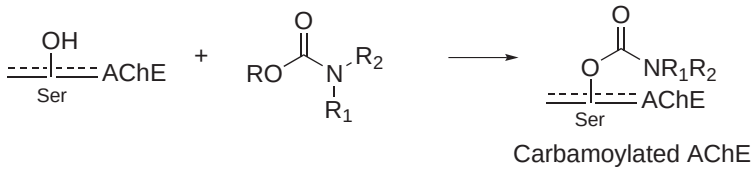


Figure 6.1: Figure displaying the structure of the substrate, ACh. The black spheres represent carbon atoms, the blue sphere represents a nitrogen atom, the red atoms represent oxygen atoms and the white spheres represent hydrogen atoms.

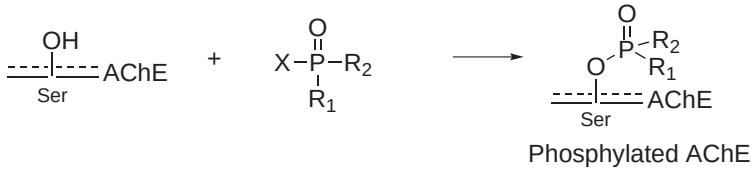
Thompson and Richardson [79] describe the general structure of a carbamate and an OP pesticide molecule. They also outline the reaction that takes place between each ligand type and AChE. Both of these are illustrated in Figure 6.2.



a)



b)



c)

Figure 6.2: Figures a), b) and c) depict the reaction between AChE and: ACh; a general carbamate; and a general OP respectively.

6.2.2 Concept

As introduced in Subsection 1.3.2, QSAR assumes that the activity or function of a molecule is correlated with one or more of the structural properties of the molecule itself. At the basis of the theory is that similar molecules induce similar reactions and consequences within a biological system. QSAR has resulted in an increased understanding of the molecular properties necessary to trigger a certain activity and is therefore vital to the discovery and development of new and more effective drugs, as well as producing an increased understanding of current drugs.

In terms of this research, QSAR represents the theory that pesticide toxicity is correlated with one or more properties of the pesticide molecule itself. Much research has been carried out to highlight the structural properties that influence the resulting toxicity [30] [91]. Here, we test the unique hypothesis that pesticide toxicity is related to the *similarity* between the pesticide molecule and the natural ligand, ACh. It is with ACh that the protein, AChE, naturally binds in a very similar way that a pesticide molecule would bind. We can see from Figure 6.2 that ACh, a carbamate and an OP each form a bond with the same oxygen atom (within a particular serine residue) in AChE. Both ACh and a carbamate contain a 'carbon double-bond oxygen' within their structures, whereas an OP contains a 'phosphorus double-bond oxygen'. The afore mentioned oxygen atom in AChE will form a bond with this carbon atom for both ACh and a carbamate, or the phosphorus atom for an OP. It seems intuitive that the similarities or dissimilarities between a synthetically developed pesticide and the naturally formed ligand will help characterise the bind with AChE and ultimately influence the final activity, i.e., the induced toxicity.

In this chapter we produce a measure of *shape similarity* between the two molecules. Molecular shape is not well-defined in molecular biology [58]. One of the reasons for this is likely to be due to the flexibilty and continual change of molecular shape dependent on the encountered environment. To avoid the vast difficulties associated

with incorporating molecular flexibility, we focus on particular conformations which we discuss further in Subsection 6.4.3.

Next we introduce the methodology developed to produce a measure of shape similarity between ACh and a pesticide.

6.3 Methodology to produce measure of shape similarity

Let μ and x be the $m\times 3$ and $n\times 3$ matrices containing the atomic coordinates of ACh and a pesticide respectively. We exclude hydrogen atoms from the analyses so there are $m = 10$ atoms in ACh under consideration. Dryden *et al.* [27] consider Bayesian methodology within MCMC to infer the matching and transformation necessary to superimpose (or align) two molecules. Here we provide alternative methodolgy to reach the same goal and ultimately enable the calculation of the shape similarity between two molecules.

6.3.1 Graphical matching algorithm

To assign atomic matches across μ and x we use the program *BKTest* (written by Gold [36]) which implements a graphical matching algorithm, originally developed by Bron and Kerbosch [17]. Inputting μ and x into BKTest, distance matrices are produced for each molecule and used to find a maximal common-induced subgraph to infer the best atomic matches.

Using similar notation to that introduced in Chapter 2, let M be the $10\times n$ matching matrix, where

$$M_{ij} = \begin{cases} 1 & \text{if } \mu_i \text{ matches } x_j \\ 0 & \text{otherwise} \end{cases},$$

for $i = 1,\ldots,10$ and $j = 1,\ldots,n$. Note that in this case, we do not have a final column indicating coffin bin allocations as we did in Chapter 2.

Let μ^* and x^* represent matrices containing the matched atomic coordinates across μ and x respectively. If $M_{ij} = 1$, then $\mu_l^* = \mu_i$ and $x_l^* = x_j$ for some l, where there are

$l = 1, \ldots, L$ matches.

Before matches can be inferred, we need to input a *distance tolerance* into the matching algorithm. The distance tolerance ensures that

$$\|\mu^*_{l_1} - \mu^*_{l_2}| - |x^*_{l_1} - x^*_{l_2}\| < \zeta,$$

where ζ indicates the input distance tolerance, $l_1, l_2 = 1, \ldots, L$ and $l_1 \neq l_2$.

6.3.2 Superimposition via Procrustes methodology

Using the inferred matches, we use Procrustes methodology [28] to superimpose μ^* onto x^*. Let $\hat{A}$ and $\hat{b}$ be the estimated transformation parameters (scale is not relevant here). The measure of shape similarity is simply the sum of squares (OSS) between the matched atom pairs after the superimposition (measured in squared angstroms, Å^2), i.e.,

$$\text{OSS} = \sum_{l=1}^{L} \|x^*_l - \hat{A}\mu^*_l - \hat{b}\|^2,$$

where $\hat{A}$ is the estimated 3×3 rotation matrix and $\hat{b}$ is the estimated translation vector. Note that the typical distance between two atoms in a molecule is 1Å-2Å.

It should be noted that we consider OSS rather than the root mean squared distance (RMSD) so that information involving the number of matches, L, is not lost.

The number of matches inferred by the graphical matching algorithm, L, is of course discrete. Figure 6.3 displays the output OSS against the input distance tolerance when comparing a random pesticide with ACh. We can see that increasing the distance tolerance, ζ, will generally increase the output OSS, that is, ζ and OSS have a high positive correlation.

However, increasing ζ will not *always* increase the number of matches, L, so the OSS can remain constant over ζ (for example, for $\zeta \in [0.7, 1.1]$ in Figure 6.3). It is also possible for the OSS to decrease as ζ increases as the subset of matches may change even when L remains constant (for example, for $\zeta \in [1.4, 1.5]$ when $L = 8$ or $\zeta \in [2.0, 2.1]$ when L increases from $L = 8$ to $L = 9$).

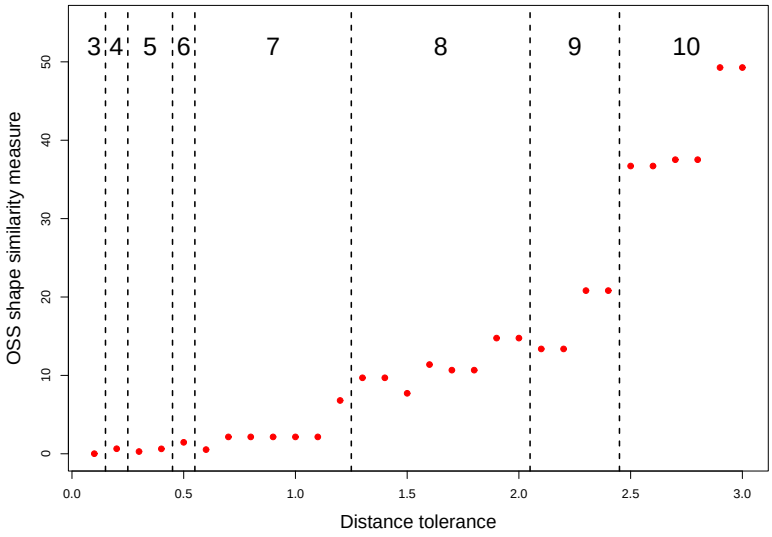


Figure 6.3: Figure displaying the output OSS against the input distance tolerance, ζ, as red dots. The number of matches inferred at a specific distance tolerance is indicated at the top of the plot.

6.4 Data introduction and development

Here we discuss the data we consider within the following analyses and give appropriate notation. We first introduce the toxicity data, the biological descriptors and the atomic coordinates representing specific pesticides given to us by CSL, York. Obviously the measure of shape similarity produced will be extremely dependent on the considered molecular conformation of each pesticide and ACh. So next we discuss the single conformation we consider for each pesticide and the two conformations of ACh under investigation. Finally, we show how the pesticide conformation and the two conformations of ACh can be used to produce two measures of shape similarity for each pesticide.

6.4.1 Toxicity data

We have toxicity data for different subsets of 145 pesticides calculated from 5 different species; mallards, japanese quails, red-winged blackbirds, starlings and bobwhite quails. We consider the LD50 toxicity, that is, the amount of pesticide neccessary to kill 50 percent of a species.

Let $y_i^{(k)}$ be the toxicity of the ith pesticide when ingested by the kth species, for $i = 1, \ldots, m_k$ and $k = 1, \ldots, 5$ ($k = 1$ indicates bobwhite quails, $k = 2$ Japanese quails, $k = 3$ mallards, $k = 4$ red-winged blackbirds and $k = 5$ starlings).

6.4.2 Biological descriptors

We have been given over 1000 biological descriptors for each of the 145 pesticides.

A DEMETRA software tool [9] is available online to predict the acute oral toxicity related to the administration of a pesticide to bobwhite quails. Previous research has highlighted numerous biological descriptors as being significant indicators of toxicity [30] [91]. The software requires the input of 14 of these biological descriptors, 13 of which

are numerical. It is these 13 numerical descriptors that we consider in the later analyses.

Let θ_{ij} be the jth biological descriptor of the ith pesticide, found to be relevant to bobwhite quail toxicity, for $j = 1, \ldots, 13$ and $i = 1, \ldots, 52$. Here we consider the 52 pesticides for which we have bobwhite quail toxicity data.

6.4.3 Molecular conformations

Pesticide conformation

We have atomic coordinate data for 145 pesticides in *minimum-energy conformations*. In mathematical terms, the minimum-energy conformation of a molecule is equivalent to the conformation associated with the highest likelihood of occurrence. For this reason, it is appropriate to use the minimum-energy conformation of each pesticide for our analysis.

Let $x^{(i)}$ be the $n_i \times 3$ matrix containing the coordinates of the minimum-energy conformation associated with the ith pesticide.

Conformations of ACh

It is also appropriate to compare each pesticide to the minimum-energy conformation of ACh. In this case, we were not given the appropriate atomic coordinates so they must be generated. For an input SMILES formula, the program Corina [35] will generate a single low-energy conformation. The SMILES formula of a molecule is the 'Simplified Molecular Input Line Entry Specification'. The formula for ACh is CC(=O)OCC[N+](C)(C)C [44]. Inputting this SMILES formula into an online demonstration of Corina, a low-energy conformation of ACh is generated.

We should note that this is a low-energy conformation rather than a global minima. Michael North [60] states that 'there is no methodology which can guarantee to find the global minimum-energy conformation. There are, however, various methods which can be used to generate multiple [local] minimum-energy conformations of a molecule'.

Another sensible conformation of ACh to examine is the bioactive, i.e. the docked

conformation of ACh within AChE. A *native* structure of AChE, that is, a crystallised structure of AChE with its substrate, ACh, is recorded in the *Protein Data Bank* (PDB), an online archive of experimentally determined structures. From this database we can extract the docked conformation of ACh and consider this as our second conformation of ACh under investigation.

Aside: This native structure, named *torpedo californica* file 2ace [65], was extracted from a particular species of stingray. Although AChE has been isolated from a wide range of species [33], few 3D structures of the native complex have been recorded, including structures from the species with which we have toxicity data for. However, a high degree of homology exists for AChE across a variety of species [55] and *torpedo californica* is often used as a standard native structure of AChE [79] within this type of analyses.

Let $\mu^{(1)}$ be the 10×3 matrix containing the atomic coordinates of the low-energy conformation of ACh. Let $\mu^{(2)}$ be the 10×3 matrix containing the atomic coordinates of the docked conformation of ACh. We refer to the analyses involving $\mu^{(1)}$ and $\mu^{(2)}$ as **Case 1** and **Case 2** throughout this text.

6.4.4 The measure of shape similarity

We have previously introduced the distance tolerance, ζ, necessary to output the inferred matches across ACh and a pesticide before calculating the OSS shape similarity measure. Intuitively, there is no obvious value of ζ that we should consider. Investigating a range of ζ may produce the same vectors of OSS over the pesticides. So rather than fixing ζ, we choose to fix the number of matches as $L = 10$, that is, all 10 atoms within ACh must be matched to atoms within each pesticide (as 144 out of the 145 pesticides have more than 10 atoms). Figure 6.3 displays the possibility for the OSS to vary over a fixed L. So we further consider ζ as the minimum distance threshold necessary (to 2dp) to match all 10 atoms within ACh.

In Case 1, we produce the OSS measure of shape similarity between $\mu^{(1)}$ and $x^{(i)}$ for $i = 1, \ldots, 144$ (note that the pesticide with only 7 non-hydrogen atoms is excluded from the analysis, leaving 144 pesticides remaining). Both $\mu^{(1)}$ and $x^{(i)}$ are input into the graphical matching program, BKTest. We fix the distance threshold as $\zeta = \zeta_{1i}$, the minimum distance tolerance to find corresponding atoms for all 10 atoms within $\mu^{(1)}$.

Let $x^{(i)*}$ be the 10×3 matrix containing the matched coordinates of atoms in $x^{(i)}$. The atom represented by the lth row in $x^{(i)*}$, $x_l^{(i)*}$, is matched to the atom represented by the lth row in $\mu^{(1)}$, $\mu_l^{(1)}$, for $l = 1, \ldots, 10$.

Finally, we apply Procrustes methodology to superimpose $\mu^{(1)}$ onto $x^{(i)*}$ before calculating the OSS measure of shape similarity, so that

$$\theta_{1i}^* = \sum_{l=1}^{10} \|x_l^{(i)*} - \hat{A}_{1i}\mu_l^{(1)} - \hat{b}_{1i}\|^2,$$

where θ_{1i}^* indicates the OSS when calculating the shape similarity between $\mu^{(1)}$ and $x^{(i)}$. The estimated transformation parameters necessary to superimpose $\mu^{(1)}$ onto $x^{(i)*}$ are denoted by $\hat{A}_{1i}$ and $\hat{b}_{1i}$.

We repeat this process for the Case 2 conformation of ACh to produce the OSS measure of shape similarity between $\mu^{(2)}$ and $x^{(i)}$, denoted by θ_{2i}^*, for $i = 1, \ldots, 144$.

6.5 Analyses of toxicity prediction

Let $y^{(k)}$ be the vector containing all pesticide toxicity values for species k. Let $\theta_1^{(k)*}$ and $\theta_2^{(k)*}$ be the corresponding vectors of shape similarity in Case 1 and Case 2 respectively, for $k = 1, \ldots, 5$.

First we explore the distributions of the shape similarity measures for the 144 pesticides in both Case 1 and Case 2. Then we investigate the correlation between $y^{(k)}$ and the two measures of shape similarity, $\theta_1^{(k)*}$ and $\theta_2^{(k)*}$ for $k = 1, \ldots, 5$.

Finally we focus on the data we have concerning the bobwhite quail toxicity. For the 52 pesticides for which we have toxicity data, we include the measures of shape similarity

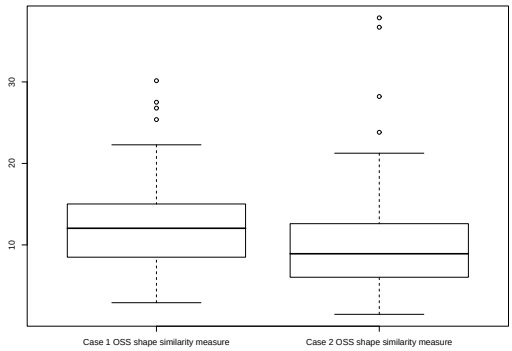
alongside the 13 biological descriptors and investigate whether there is an improvement in toxicity prediction. We compare the significance of the shape similarity measures with the significance of the 13 biological descriptors when predicting toxicity. Lastly, we compare the toxicity prediction accuracy when using our developed model and the known online predictor [9].

6.5.1 Distribution of the shape similarity measures

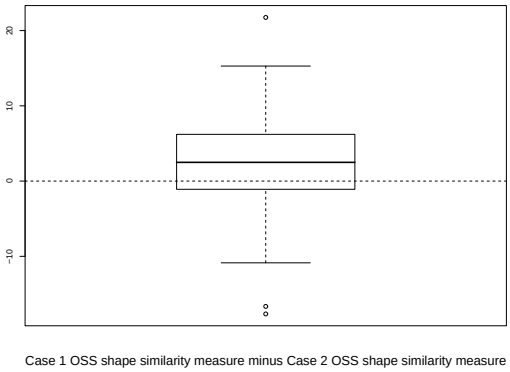
Figure 6.4 explores the distributions of the shape similarity measures. Figure 6.4a displays boxplots of the shape similarity measure for Case 1 and Case 2 respectively. Both Case 1 and Case 2 show evidence of four outliers, though only one observation is an outlier in both cases. Figure 6.4b displays the boxplot of the Case 2 shape similarity subtracted from the Case 1 shape similarity measure. The differences have a symmetrical distribution and indicate that the Case 1 measure is generally higher than the Case 2 measure for a given pesticide. Figure 6.4c shows the Case 2 shape similarity measure against the Case 1 shape similarity measure. When all observations are considered, the regression line shows a slight positive correlation between the two variables. Excluding the outliers (as indicated by the boxplots in Figure 6.4a), the regression line is much flatter indicating no relationship between the Case 1 and the Case 2 shape similarity measures and highlighting the importance of the molecular conformations considered within these analyses.

6.5.2 Correlation between toxicity and shape similarity measure

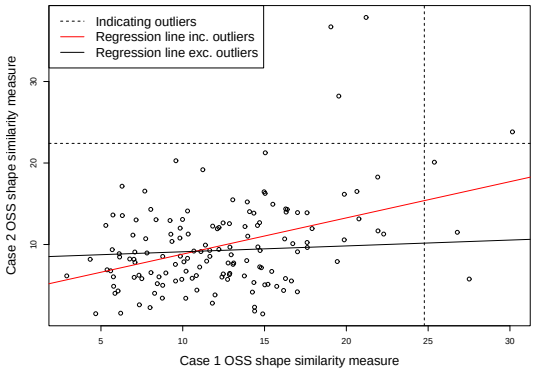
Figure 6.5 displays $y^{(k)}$ against $\theta_1^{(k)*}$ in the first column and $y^{(k)}$ against $\theta_2^{(k)*}$ in the second column for $k = 1, \ldots, 5$. Table 6.1 displays the number of toxicity observations we have for each species. Pearson's correlation coefficient between $y^{(k)}$ and both $\theta_1^{(k)*}$ and $\theta_2^{(k)*}$ has been calculated including and excluding outliers. The one-sided critical value at a 95% confidence level (when considering all observations) is also displayed for each species.



a)



b)



c)

Figure 6.4: Figures a) displays boxplots of the shape similarity measure for Case 1 and Case 2 respectively. Figure b) displays the boxplot of the Case 2 shape similarity subtracted from the Case 1 shape similarity measure. Figure c) shows the Case 2 shape similarity measure against the Case 1 shape similarity measure.

Note: We consider the four shape similarity measures highlighted in Figure 6.4a for both Cases 1 and 2 as outliers. They include the observations classed as being less than LQ-1.5IQR or higher than UQ+1.5IQR. The toxicity outliers are calculated in the same way but separately for each species.

		Pearson's correlation coefficient				
		Case 1		Case 2		
Species	No. obs.	Inc. outliers	Exc. outliers	Inc. outliers	Exc. outliers	Critical value
1	51	-0.18	-0.07	-0.45	-0.31	0.23
2	67	-0.05	-0.08	-0.25	-0.23	0.20
3	84	-0.12	-0.11	-0.13	-0.03	0.18
4	72	0.14	0.14	-0.09	-0.13	0.20
5	62	0.12	0.20	-0.02	-0.07	0.21

Table 6.1: Table displaying the number of toxicity observations we have for each species. The Pearson's correlation coefficient between $y^{(k)}$ and both $\theta_1^{(k)*}$ and $\theta_2^{(k)*}$ has been calculated including and excluding outliers. The one-sided critical value at a 95% confidence level is also displayed for each species when considering the full dataset.

In Case 1, we can see that negative correlations between $y^{(k)}$ and $\theta_1^{(k)*}$ have been calculated for $k = 1, 2, 3$, though positive correlations were found for $k = 4, 5$. In Case 2, we can see that a negative correlation between $y^{(k)}$ and $\theta_2^{(k)*}$ is found for all $k = 1, \ldots, 5$.

Conclusion

The only significant linear correlation we find is in Case 2. For species 1 and 2, bobwhite quails and japanese quails, we find evidence at the 95% confidence level of a negative correlation between toxicity and the shape similarity measure, when the outliers are both included and excluded. This result could be due to an increased homology between the docked ACh within quails and the type of ray from which the considered ACh was

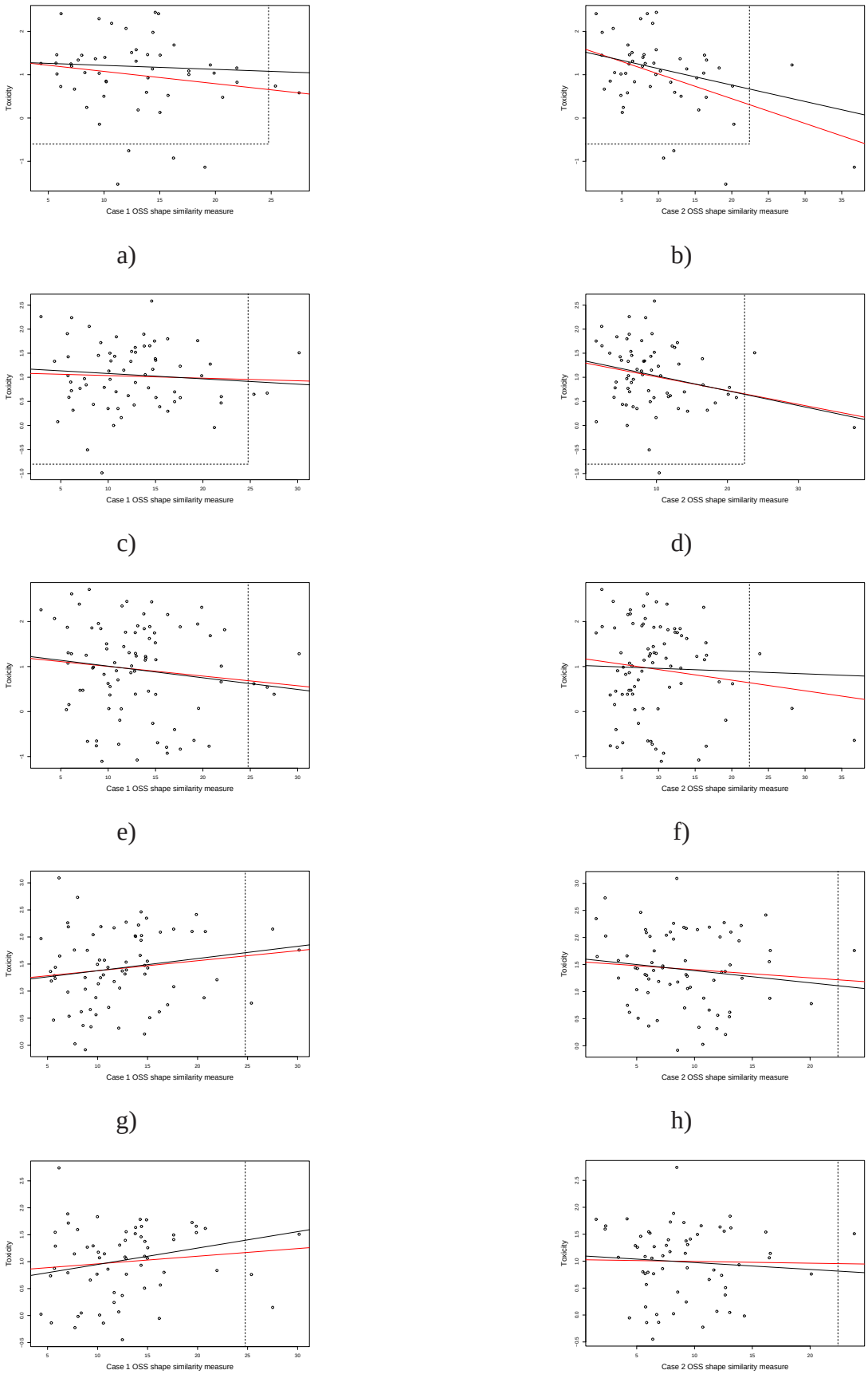


Figure 6.5: Figure displaying $y_{(k)}$ against $\theta_1^{(k)*}$ in the left column and $y_{(k)}$ against $\theta_2^{(k)*}$ in the right column. The kth row indicates the kth species. Points below or to the right of the black dotted line indicate outliers within the dataset. The red and black lines are the fitted regression lines when outliers are included and excluded respectively.

extracted.

6.5.3 Predicting the bobwhite quail toxicity using the shape similarity measure alongside biological descriptors

In summary, the data we now consider is as follows for $i = 1, \ldots, 51$, for the 51 pesticides for which we have bobwhite quail toxicity data (excluding the pesticide with only 7 non-hydrogen atoms).

- The vector of pesticide toxicities, y, where an element y_i is the bobwhite quail toxicity of the ith pesticide. (Note that for simplicity of notation, we have set $y = y^{(1)}$ etc.)
- The 51×13 matrix containing the biological descriptors, where an element θ_{ij} is the jth descriptor of the ith pesticide.
- The corresponding vectors of OSS shape similarity, θ_1^* and θ_2^*, for Case 1 and Case 2 respectively. An element, θ_{1i}^* and θ_{2i}^*, is respectively the Case 1 and Case 2 measure of shape similarity for the ith pesticide.

The online predictor discussed in Subsection 6.4.2 [9] is a hybrid model consisting of two possible algorithms, one of which is simply a linear model in which the biological descriptors are the independent variables.

For Case l, $l = 1, 2$, we choose to fit a linear model between the variables and response, i.e. find β_j that best fits

$$y_i = \beta_0 + \beta_1 \theta_l^* + \sum_{j=1}^{13} \beta_{j+1} \theta_{ij} + \epsilon_i, \tag{6.1}$$

for $j = 0, \ldots, 14$. The intercept is denoted by β_0, β_1 is the coefficient of the shape similarity measure and β_j is the coefficient of the $(j-1)$st biological descriptor for $j = 2, \ldots, 14$. The normally distributed error of the ith observation is denoted by ϵ_i for $i = 1, \ldots, 51$.

Initially we calculate the pairwise correlation between the toxicity and each variable. We test the significance of the full linear model by testing the hypothesis that $H_0 : \beta = 0$, where β is the vector containing all β_j for $j = 1, \ldots, 14$. We then test the significance of each variable by testing the hypotheses that $H_0 : \beta_j = 0$ for $j = 1, \ldots, 14$.

For Case 1, the top table in Tables 6.2 displays the correlation between y and θ_1^* and the p-value of β_1, denoted by $\rho(\theta_1^*, y)$ and p_1-value respectively. The ranks of these values in comparison to the corresponding values of the 13 biological descriptions is shown. The p-value of the full linear model is stated. Also displayed is the OSS and correlation between y and the predicted toxicities, denoted by $\hat{y}$. The adjusted R^2 is displayed in the final row. The same is displayed for Case 2, in both cases considering the inclusion and exclusion of toxicity outliers.

For comparison, the bottom table in Tables 6.2 displays the p-value, the OSS and correlation between the true and fitted toxicity, and the adjusted R^2 when the shape similarity measure is excluded from the linear model. So, in this case, only the 13 biological descriptors are used to predict toxicity.

Figure 6.6 displays the residuals against y (note not the fitted toxicity, $\hat{y}$) when the toxicity outliers are a) included and b) excluded when fitting the linear model. The filled circles represent observations when the shape measure is excluded from the linear model and the crosses indicate the Case 2 residuals. The dotted red line connects the two residuals for the same pesticide.

Discussion

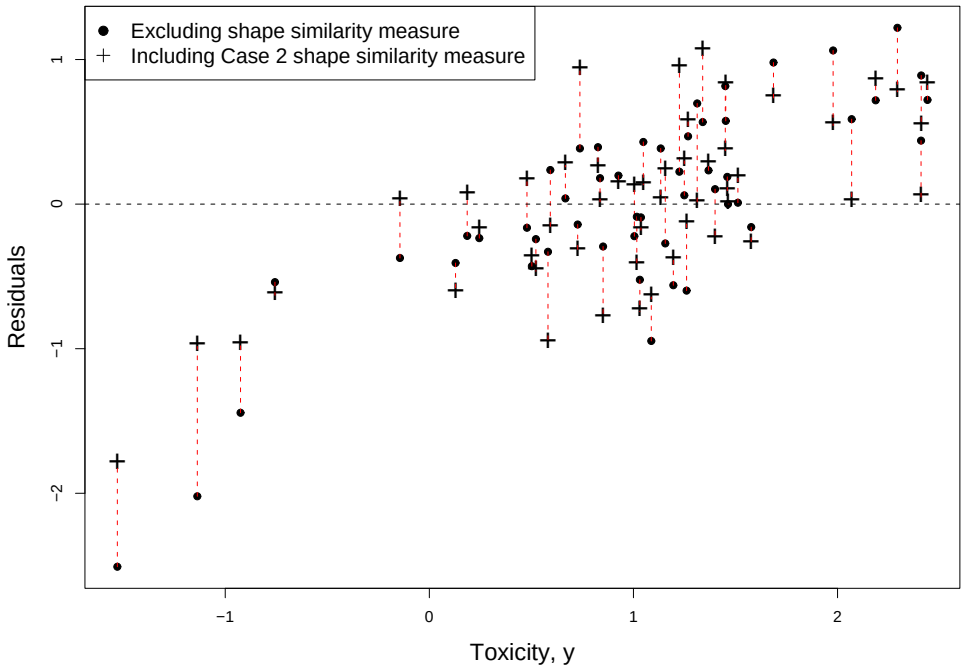
We discuss Case 1 and Case 2 separately.

In Case 1, when all observations are considered, the absolute correlation between y and θ_1^* is higher than the correlation between y and 10 of the biological descriptors. The p_1-value is lower than the p-values of 8 biological descriptors, though it is not significant enough to reject $H_0 : \beta_1 = 0$. When outliers are excluded, the absolute correlation between y and θ_1^* is again higher than the correlation between y and 8 of the biological

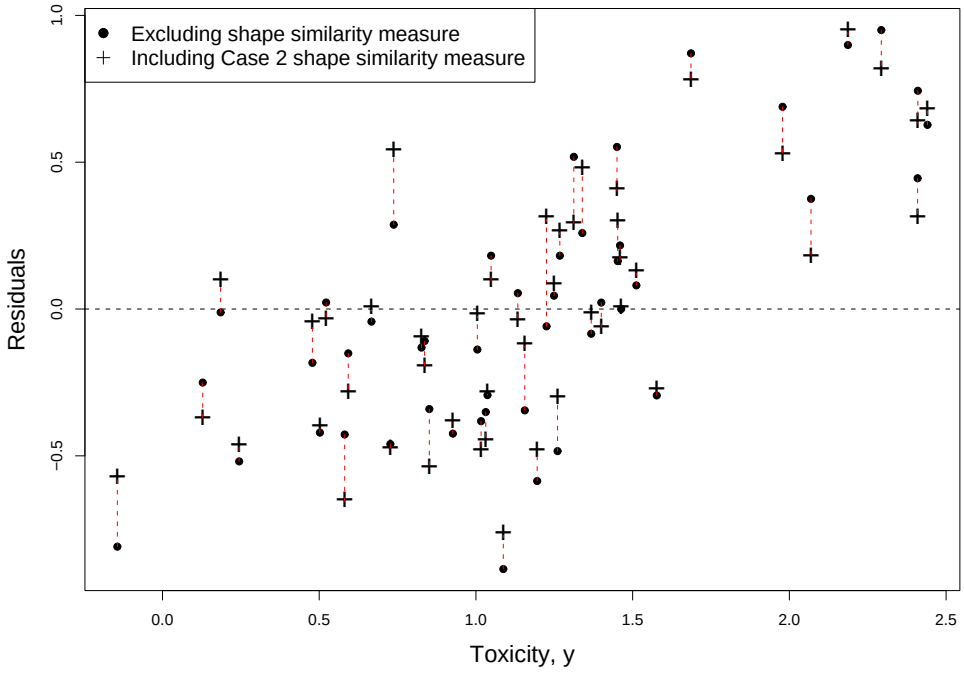
Including OSS shape measure				
	Case 1		Case 2	
	Inc. outliers	Exc. outliers	Inc. outliers	Exc. outliers
$\rho(\theta_1^*, y)$	-0.1768	-0.1557	-0.4521	-0.2607
Rank: $\rho(\theta_1^*, y)$	4	4	1	3
p_1-value	0.3006	0.6205	0.0003	0.0723
Rank: p_1-value	6	12	1	3
p-value	0.2189	0.0371	0.0048	0.0131
OSS($y, \hat{y}$)	23.8031	8.9277	17.0769	8.1209
$\rho(y, \hat{y})$	0.5892	0.6954	0.7291	0.7282
Adjusted R^2	0.0933	0.2577	0.3495	0.3247

Excluding OSS shape measure		
	Inc. outliers	Exc. outliers
p-value	0.2131	0.0243
OSS($y, \hat{y}$)	24.5323	8.9975
$\rho(y, \hat{y})$	0.5720	0.6925
Adjusted R^2	0.0908	0.2745

Table 6.2: For Case 1, the top table displays the correlation between y and θ_1^* and the p-value of β_1, denoted by p_1-value. The ranks of these values in comparison to the 13 biological descriptions is shown. The p-value of the full linear model is stated. Also displayed is the OSS and correlation between y and the predicted $\hat{y}$. The adjusted R^2 is displayed in the last row. The same is displayed for Case 2, in both cases considering the inclusion and exclusion of toxicity outliers. The bottom table displays the p-value, the OSS and correlation between the true and fitted toxicity, and the adjusted R^2 when the shape similarity measure is excluded from the linear model. We again consider both the inclusion and exclusion of toxicity outliers.



a)



b)

Figure 6.6: Figure displaying the residuals against y when the toxicity outliers are a) included and b) excluded. The filled circles represent observations when the shape measure is excluded from the linear model and the crosses indicate the Case 2 outliers. The dotted red line connects the two residuals for the same pesticide.

descriptors. Though the p_1-value is now lower than the p-values of only 2 biological descriptors, again it is not significant enough to reject $H_0 : \beta_1 = 0$. Including outliers, we find a slight increase in the adjusted R^2 when the Case 1 shape similarity measure is included within the linear model. Excluding outliers, we find a decrease in the adjusted R^2 when the Case 1 shape similarity measure is included within the linear model.

In Case 2, when outliers are included, the absolute correlation between y and θ_2^* is higher than the correlation between y and all 13 of the biological descriptors. The p_1-value is lower than the p-values of all 13 biological descriptors and is significant enough to reject $H_0 : \beta_1 = 0$ at even the 99.9% confidence level. When outliers are excluded, the absolute correlation between y and θ_2^* is higher than the correlation between y and 11 of the biological descriptors. The p_1-value is lower than the p-values of 11 biological descriptors and is almost significant enough to reject $H_0 : \beta_1 = 0$ at the 95% confidence level. Including outliers, we find a large increase in the adjusted R^2 when the Case 2 shape similarity measure is included within the linear model. Excluding outliers, we find a relatively large increase in the adjusted R^2 when the Case 2 shape similarity measure is included within the linear model.

Conclusion

The main conclusion is that the shape similarity measure between the minimum-energy pesticide conformations and the docked conformation of ACh is a significant predictor of the associated acute oral toxicity to bobwhite quails. We can see, from the illustrations in Figure 6.6, an obvious improvement in the toxicity predictions of pesticides with particularly low or high toxicities. We should note that these results are vulnerable to the problems associated with multiple testing. The p-values of the variable coefficients are dependent on the variables under consideration, therefore we could find them less significant if a different set of variables were considered.

6.5.4 Comparison of the fitted model with an online toxicity predictor

In the final subsection, we use cross-validation to compare the accuracy of toxicity prediction under our model to that found when implementing the online predictor [9]. In turn, we exclude each of the 51 pesticides and fit the linear model in Equation (6.1) using the remaining 50 pesticides as the training set. Then the fitted model is used to predict the toxicity of the excluded pesticide. We do this for both Case 1 and Case 2.

Let $\rho(y, \hat{y})$ denote the correlation between the true and predicted toxicities. Let $|\bar{r}|$ denote the mean absolute residual between the true and predicted toxicities. Table 6.3 provides these results for each of the considered toxicity predictors.

	Online predictor	Case 1	Case 2
$\rho(y, \hat{y})$	0.31	0.14	0.40
$\lvert\bar{r}\rvert$	1.23	0.69	0.65

Table 6.3: Table displaying $\rho(y, \hat{y})$ and $|\bar{r}|$ when implementing the online predictor or when applying the linear model in Equation (6.1) for Case 1 and Case 2.

Conclusion

The one-sided critical value of the correlation coefficient at a 95% confidence level with 51 observations is 0.23. We can see from Table 6.3 that both the application of the online predictor and the Case 2 linear model provide a significant correlation between the true and predicted toxicities, with the highest correlation being produced when including the Case 2 shape similarity measure within our model.

Using the Case 1 and Case 2 linear model as a toxicity predictor provides a much lower absolute residual between the true and predicted toxicities on average that the online predictor.

Here we have further evidence that the inclusion of a shape similarity measure increases the accuracy of toxicity prediction, especially when the docked conformation of ACh is used to calculate the measure of shape similarity between ACh and a pesticide.

Chapter 7

Pesticide dock as toxicity predictor

7.1 Introduction

In Section 7.2 we describe the concept behind considering a docked molecular conformation of a pesticide when attempting to predict the associated toxicity. In Section 7.3 we introduce a docking program and explore the prediction accuracy by using it to predict the known docked conformation of ACh within AChE. In Section 7.4 we define a method to calculate a distance measure between a docked ligand and AChE and discuss how it can be used as an indicator of docking accuracy. We highlight a relationship between this measure and the accuracy of a predicted dock. In Section 7.5 we produce a measure of similarity between the known dock of ACh and the predicted pesticide docks. Finally we investigate the significance of these measures, alongside an associated inhibition constant, as toxicity predictors for the bobwhite quail.

7.2 Concept

In Chapter 6 we found evidence that the shape similarity between the minimum-energy pesticide conformation and the docked conformation of ACh was a significant predictor of the associated quail toxicity. 3D-QSAR approaches consider the properties of a ligand in

their bioactive form to be more appropriate indicators of the associated activity. In terms of this research, 3D-QSAR dictates that the properties of a docked pesticide conformation will provide a better indication of the associated toxicity.

Within this section we want to calculate a measure of similarity between the docked locations of both ACh *and a pesticide* to explore whether this provides a more significant predictor of toxicity. We use a docking program (introduced in the following subsection) to predict the docked conformation of the pesticides under consideration. A common approach in this type of analysis is to fix the protein as rigid. This enables a direct comparison of the docked locations between ACh and a pesticide with respect to a fixed protein. Figure 7.1 illustrates the complementary geometries between a substrate and a protein, demonstrating the basic lock and key concept first postulated by Emil Fischer [31]. The concept of a key fitting into the lock to open a door was developed to represent a substrate binding with a protein to initialise some activity. According to this theory, we can think of a pesticide and ACh as being two different keys designed to fit the same lock. As well as being able to make a direct comparison between the docks of ACh and a pesticide, we can also explore whether the closeness of the fit between a pesticide and AChE provides an indicator of pesticide toxicity. It has already been established that tightly binding ligands have a high degree of shape complementarity with their receptor [22]. It is intuitive that the closer a pesticide and AChE, the tighter they are bound, the longer AChE will be inhibited and the stronger the toxic effects.

Note: The theory of the rigid binding site has since been proved inaccurate and has been modified by an induced-fit theory proposed by Koshland [47]. In this case, the substrate induces changes in the molecular conformation of the AChE binding site until the substrate is bound and the final complex shape is determined [15]. However, to provide a fixed position of AChE relative to both ACh and a pesticide, we allow the protein configuration to remain rigid so that a direct comparison of the docked configurations of both ACh and a pesticide can be made.

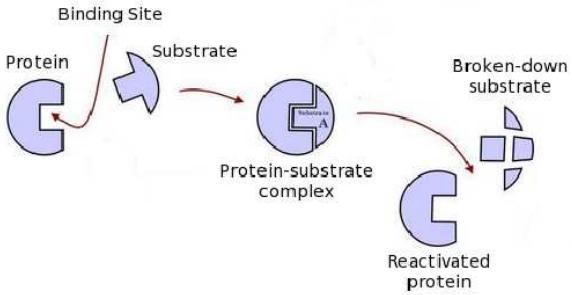


Figure 7.1: Displaying the theory relating the geometric bind between a protein and substrate to a lock and key respectively.

7.3 The docking program, AutoDock 4

7.3.1 Introduction

AutoDock 4 is a suite of C programs implemented to predict the conformation formed between an input ligand and protein. It is amongst the five most popular docking programs [74] and came second for docking accuracy when being compared to the remaining four [18]. The top ranking docking program is only commercially available whereas AutoDock is free for academics. AutoDock has been applied with great success to the prediction of enzyme-inhibitor complex conformations [57] and is a logical choice for our analyses.

The only input required are the atomic coordinates of the ligand and protein. The ligand is treated as flexible and the protein can be fixed as entirely rigid or flexible within set residues. AutoDock is made up of three main programs:

1. **AutoTors** first processes the ligand. The default unbound state (used in later energy calculations) is set as the extended ligand, where all atoms are pushed as far away as possible from each other. Rotatable torsion angles are assigned. The ligand explores six degrees of freedom for translation and rotation plus the assigned number of torsional degrees of freedom.

2. **AutoGrid** creates a grid of interaction energies between the input ligand and the protein. A 3D grid is constructed around the full protein or a particular area of interest, such as the protein binding site. At regular intervals within the grid, interaction energies are calculated and stored for each atom type within the ligand. The full energy grid provides a quick look-up table for the evaluation of the full interaction energy between a ligand and protein. The force field used to evaluate the energies is based on the Amber force field [19], which was primarily developed to represent molecular dynamics involving proteins [88].

3. **AutoDock** performs the actual docking simulation using a Lamarckian genetic algorithm [57]. The free energy of binding is calculated as the difference between the energies of the separate molecules and the energy of the ligand-protein complex. It is made up of energy terms for dispersion/repulsion, hydrogen bonding, electrostatics and desolvation.

The free energy of binding, ΔG, is used to rank the final predicted conformations over all simulations. AutoDock defines the relationship with the inhibition constant as

$$\ln k_I \propto \Delta G.$$

The more negative ΔG, or equivalently the lower k_I, the more likely the prediction is to represent the true ligand-protein conformation. In later analyses we focus on the inhibition constant, k_I, as a variable for toxicity prediction.

In the following section we analyse the accuracy of AutoDock.

7.3.2 Exploring the accuracy of AutoDock

The accuracy of a docking program is generally measured by its ability to reproduce an experimentally determined conformation of a bound ligand [75]. Although there are no experimentally determined conformations of the bound pesticides under consideration, there is the bound conformation of ACh and AChE stored in file 2ace in the PDB [65], as

considered in Chapter 6. We can use AutoDock to predict how ACh will bind to AChE and compare the predictions to the experimentally determined conformation. We carry out two tests to investigate whether the input conformation of ACh will affect the accuracy of the predicted dock.

Test 1: We arbitrarily translate and rotate the true docked conformation of ACh away from its docked location (though the distance moved was fixed as 100A to ensure the validity of the created PDB file).

Test 2: We use the program *Frog* [14] to generate multiple different conformations of ACh which are then input directly into AutoDock. The only input required to generate an assigned number of conformations is the SMILES formula of ACh, CC(=O)OCC[N+](C)(C)C.

We carry out 40 trials for each test, that is, we input 40 different starting orientations and conformations respectively for Test 1 and Test 2. In each trial, we request that AutoDock output 50 predictions for the docked conformation.

For each test, we fix AChE to be the rigid conformation experimentally determined. That is, the conformation of AChE is fixed as the true bound conformation. When analysing the accuracy of AutoDock, we only need to compare the true dock of ACh with the predicted docks. The difference between a predicted and experimentally determined docked conformation is generally calculated as the RMSD between the corresponding 'heavy atoms' [74], i.e. non-hydrogen atoms. (Note that the three methyl groups, CH_3, are interchangeable so we calculate the RMSD in all variations and use the minimum value.) First we describe the preparation that must be carried before AutoDock can be run.

Preparing the molecules, grid and docking procedure

Before AutoDock can be run, we first need to prepare the molecules, the grid and fix the parameters within the docking procedure. We do this by carrying out the following steps.

1. Read the full AChE and ACh complex [65] into the graphical user interface of

AutoDock after deleting the single bond connecting ACh to AChE. The remaining steps can be carried out within the user interface with default settings specified when necessary.

2. Delete the water molecules from the complex and add hydrogens to both molecules.

3. Save the automatically generated PDB files (storing atom information such as type, coordinates and partial charges) separately for ACh and AChE. The true bound conformation of AChE is fixed for both tests described above. For Test 1, it is this conformation of ACh that is randomly rotated and translated before being saved as a separate PDB file. For Test 2, Frog can be set to automatically output PDB files for each conformation produced.

4. Generate atomic partial charges for both AChE and ACh and the torsional angles within ACh alone to define a flexible ligand.

5. Prepare the grid by assigning the location of the center and dimensions of the grid. In these analyses we fix the centre of the grid as the oxygen atom within AChE that will bind to the considered ligands (see Figure 6.2), so that the grid captures the relevant binding site within AChE.

6. Run AutoGrid to precalculate interaction energies for each atom type within ACh at each grid point.

7. Fix the parameters used within the Lamarckian algorithm. We set the number of predictions to be 50 for each trial.

8. Finally, run AutoDock to produce the predicted docks.

The output can now be used to analyse AutoDock accuracy.

Analysis of results

We carry out three main forms of analysis. First we explore whether the predictions in Test 1 differ from those in Test 2. Secondly, we focus on each test individually and investigate whether the initial orientation or specific conformation greatly influences the final result. Finally, as in the general case when the true dock is unknown, we use the RMSD between predictions alone to see if the observations are grouped similarly as to when the RMSD between the true and predicted docks are used.

Test 1 v Test 2 observations

Figure 7.2b displays k_I against the RMSD between the true and predicted docks for the 40×50 predictions for both Test 1 and Test 2. Due to the observed clustering about RMSD, we choose to fit a global Gaussian mixture model so that

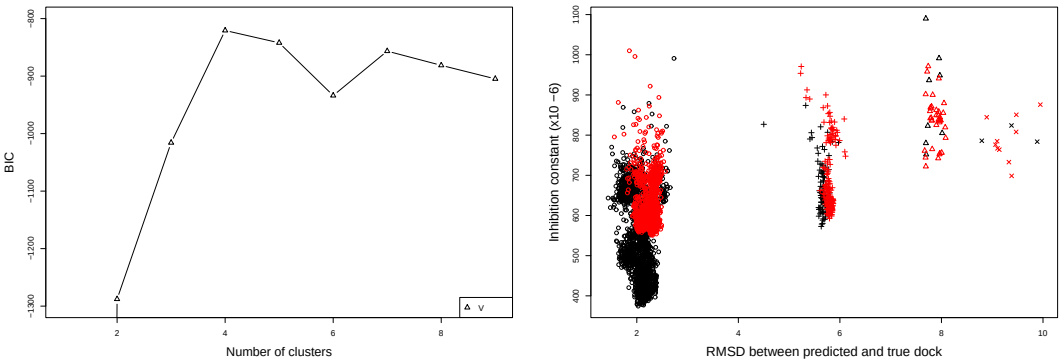
$$\text{RMSD} \sim p_j N(\mu_j, \sigma_j^2),$$

where μ_j and σ_j^2 are the mean and variance of the jth cluster and p_j is the probability of an observation being in cluster j. It is considered a global distribution because all observations in both Test 1 and Test 2 are considered. We assign the number of clusters as that that maximises the Bayesian Information Criterion (BIC) for EM initialized by model-based hierarchical clustering for parameterized Gaussian mixture models. Finally complete hierarchical clustering on the set of differences between RMSD is used to allocate each observation to a particular cluster. Figure 7.2a shows that the BIC is maximised at four clusters and each of the four clusters can be distinguished by character in Figure 7.2b.

Table 7.1 displays the number of observations in cluster j, n_j, and the estimated parameters of the mixture model, $\hat{p}_j$, $\hat{\mu}_j$ and $\hat{\sigma}_j^2$ for cluster $j = 1, \ldots, 4$. We can use the Chi-squared test to investigate whether the number of observations in each cluster for Test 1 and Test 2 separately follow the applied global distribution.

Conclusion

The p-value for both Test 1 and Test 2 observations is 1.684×10^{-7}, indicating that



a) b)

Figure 7.2: Figure a) displays the BIC against the number of clusters. Figure b) displays k_I against the RMSD between the true and predicted docks. Black indicates observations from Test 1 and red indicates observations from Test 2. Each cluster is indicated by different symbols.

	Cluster j			
	1	2	3	4
n_j	3666	276	45	13
$\hat{p}_j$	0.916	0.069	0.011	0.003
$\hat{\mu}_j$	2.152	5.745	7.869	9.305
$\hat{\sigma}_j^2$	0.034	0.019	0.015	0.118

Table 7.1: Table displaying the number of observations in each cluster and the estimated parameters of the mixture model.

observations from Test 1 and Test 2 do not follow the applied global distribution. That is, inputting the true docked conformation (although at a random orientation) provides significantly different predicted docks than when a random conformation is input.

Note: The k_I for the more accurate predictions is much lower in Test 1 than in Test 2. However, the observation with the minimum k_I is present in the most accurate cluster in both tests.

Trial dependency within each test

Following the same procedure as in the previous investigation, we fit a sub-global Gaussian mixture model separately to both Test 1 and Test 2 observations so that

$$\text{RMSD} \sim p_{1j}N(\mu_{1j}, \sigma^2_{1j}) \qquad \text{and} \qquad \text{RMSD} \sim p_{2j}N(\mu_{2j}, \sigma^2_{2j}),$$

respectively, where μ_{1j} and σ^2_{1j} are the mean and variance of the jth cluster and p_{1j} is the probability of an observation being in cluster j in Test 1 for example. They are considered to be sub-global distributions because observations in both Test 1 and Test 2 are considered separately. Again we fix the number of clusters as four for the observations in each test. Table 7.2 displays the number of observations in each cluster and the estimated parameters of the mixture model for each test. We again use the Chi-squared test to investigate whether observations from the 40 trials in each test follow the corresponding sub-global distribution applied.

Conclusion

For Test 1, we found evidence that the observations from four trials do not follow the applied sub-global distribution at the 95% critical level. For Test 2, we found evidence that the observations from only one trial did not follow the applied sub-global distribution at the 95% critical level.

When inputting the true docked conformation of ACh at a random orientation, 10% of the considered starting values do not follow the general distribution applied to the RMSD over all predictions. This provides evidence that the starting orientation does affect the output if the dock is known.

Test 1	Cluster j				Test 2	Cluster j			
	1	2	3	4		1	2	3	4
n_{1j}	1904	85	8	3	n_{2j}	1762	191	37	10
$\hat{p}_{1j}$	0.952	0.042	0.004	0.002	$\hat{p}_{2j}$	0.881	0.096	0.018	0.005
$\hat{\mu}_{1j}$	2.070	5.646	7.814	9.354	$\hat{\mu}_{2j}$	2.241	5.789	7.881	9.290
$\hat{\sigma}^2_{1j}$	0.035	0.023	0.019	0.298	$\hat{\sigma}^2_{2j}$	0.018	0.011	0.014	0.090

Table 7.2: Tables displaying the number of observations in each cluster and the estimated parameters of the mixture model fitted for Test 1 observations and Test 2 observations.

When inputting a random conformation of ACh, only 1% of the considered starting values do not follow the general distribution applied to the RMSD over all predictions. This provides evidence that, if the true dock is unknown, the random conformation input as a starting value does not greatly influence the output.

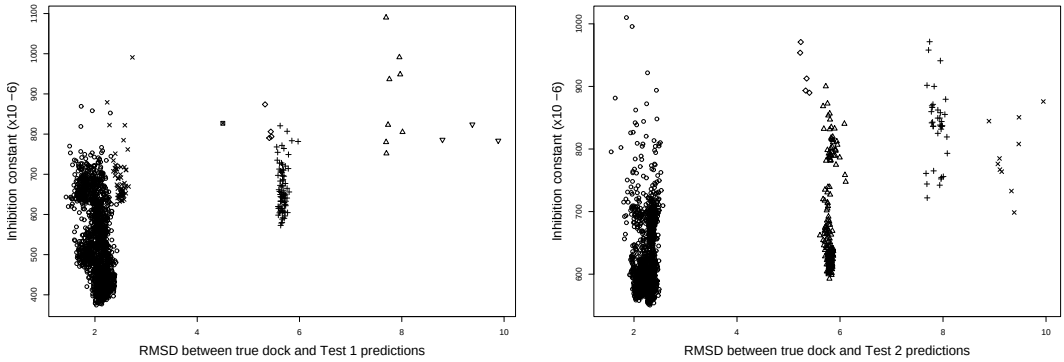
Using predicted docks to assign clusters

Here we assign clusters using the RMSD between the predicted docks alone. A RMSD< 2.5Å between the true dock and predicted dock is classed as a successful prediction [43]. We again use complete hierarchical clustering and cut the tree at a height of 2.5Å so that we locate a set of unique conformations.

Figure 7.3 displays k_I against the RMSD between the true and predicted docks for a) Test 1 and b) Test 2 where each cluster (allocated using the RMSD between predicted docks alone) can be visualised. Note that the RMSD between the true and predicted docks is only used for reasons of visual comparison.

Conclusion

There are seven clusters formed using the RMSD between predictions in Test 1 and five clusters formed using the RMSD between predictions in Test 2. In both cases a greater number of clusters is found than when the RMSD between the true and predicted docks is considered. Note that the RMSD between the true and predicted docks can be equal



a) b)

Figure 7.3: Figures displaying k_I against the RMSD between the true and predicted docks for a) Test 1 and b) Test 2. Each cluster is indicated by different symbols.

even though the predictions differ significantly. Using the RMSD between predictions we are able to locate the local minima that may be indistinguishable when considering the RMSD between the true and predicted docks.

For both Test 1 and Test 2, the cluster containing the more accurate predictions (i.e. the lower RMSD between the true and predicted docks) is the largest, containing 1848 and 1762 observations respectively.

Overall conclusions

- A RMSD$<$ 2.5Å between the true dock and predicted dock is classed as a successful prediction [43]. We find that 94% of predictions match this criteria in Test 1 and 88% from Test 2.

- The conformation associated with the minimum k_I has RMSD=2.03Å and RMSD=2.31Å from the true dock in Test 1 and Test 2 respectively. So in both tests the observation with the minimum k_I represents a successful dock.

- Inputting the true docked conformation of ACh (though at a random orientation) produced a different distribution of predicted docks than when a random

conformation of ACh was input. From Table 7.2 we can see that inputting the true docked conformation produces a larger amount of more accurate predictions which is what we intuitively would expect.

- If the true dock is known, the orientation of the input ligand conformation does affect the accuracy of the output predictions. However, if the true dock is unknown, the specific random conformation input as a starting value does not greatly influence the accuracy of the output predictions.

- When using the RMSD between predicted docks only to cluster the observations, the largest cluster contained the more accurate predictions in both Test 1 and Test 2. In Test 1, 100% of the observations in the first cluster have RMSD< 2.5Å from the true dock. In Test 2, 99.8% of the observations in the first cluster have RMSD< 2.5Å from the true dock.

7.4 Using the distance between the protein and docked ligand as an accuracy indicator

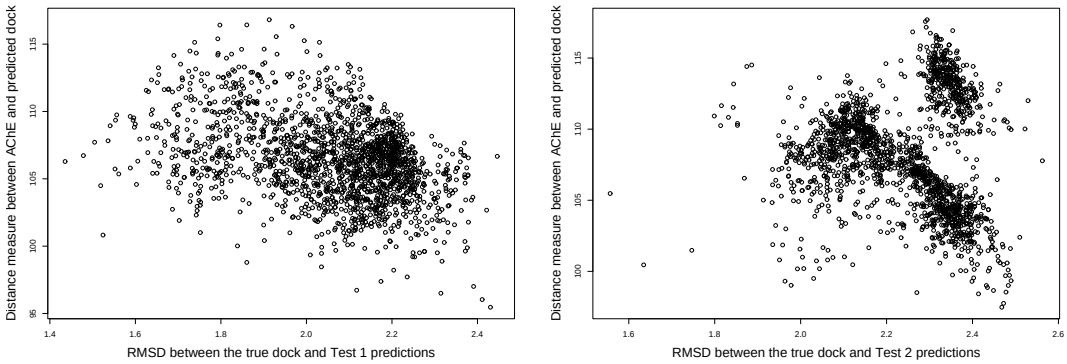
Here we show how a distance measure between a predicted ACh dock and the protein, AChE, is an indicator of the accuracy of the observations within the largest cluster (formed using the RMSD between predicted docks alone).

Let μ_P denote the 4143×3 atomic coordinate matrix for AChE (excluding hydrogens). Let $\hat{\mu}^{(l)}$ denote the 10×3 coordinate matrix for the lth predicted dock in Test 1 for $l = 1, \ldots, 2000$. We measure the distance between the lth dock and μ_P as

$$\sum_{k=1}^{10} \|\hat{\mu}_k^{(l)} - \mu_{\pi_k}^P\|^2,$$

where $\mu_{\pi_k}^P$ are the coordinates of the point in μ_P that is closest to the kth point in the lth dock, $\hat{\mu}_k^{(l)}$. Note that all points within $\hat{\mu}^{(l)}$ are considered and that one-to-many matches

are allowed. Figure 7.4a shows the distance measure against the RMSD between the true and predicted docks for each of the 1848 observations in the largest cluster. The correlation coefficient is -0.37 which provides strongly significant evidence that, as the RMSD between the true and predicted docks decreases, the distance between $\hat{\mu}^{(l)}$ and μ_P increases. That is, AutoDock is overfitting in this particular case. Figure 7.4b shows the distance measure against the RMSD between the true and predicted docks for each of the 1762 observations in the largest cluster in Test 2. The correlation coefficient in this case is -0.09, however Figure 7.4b displays two clusters which could indicate two separate local solutions that the clustering technique failed to distinguish.



a) b)

Figure 7.4: Figures a) and b) display the distance measure between AChE and the predicted docks against the RMSD between the true and predicted docks for the largest cluster formed in Test 1 and Test 2 respectively.

We use this finding to highlight an observation to investigate in the next section.

7.5 Pesticide docks as toxicity predictors

As in the previous chapter, let $x^{(i)}$ be the $n_i \times 3$ matrix containing the minimum-energy conformation of the ith pesticide. In Subsection 7.3.3 we found that, when the true dock is unknown, the conformation input into AutoDock does not significantly influence the

accuracy of the predicted docks. So we consider only the conformation stored within $x^{(i)}$ in further analyses, for $i = 1, \ldots, 17$. For simplicity, we consider only the 17 carbamate pesticides for which we have bobwhite quail toxicity.

Preparing the molecules, grid and docking parameters

We again follow steps 1-8 of the preparation described previously in Subsection 7.3.3 when exploring AutoDock accuracy, with a few alterations.

In step 3, we convert the given formats of $x^{(i)}$ to the required PDB format using *Babel* [78]. In the remaining steps we simply replace ACh with $x^{(i)}$. To allow each pesticide to be able to rotate freely, in step 5 we fix the grid dimensions to be twice that of the maximum length of the extended ligand. We now assign that $L = 1000$ predictions be made for the dock of each $x^{(i)}$.

Let $\hat{x}^{(il)}$ be the $n_i \times 3$ matrix containing the coordinates of the lth predicted dock of the ith pesticide.

Single dock to predict toxicity

Let $\hat{x}^{(i)}$ denote a single docked prediction for the ith pesticide. We fit the linear model, i.e. estimate the parameter β_j for $j = 0, \ldots, 4$ in

$$y_i = \beta_0 \sum_{j=1}^{4} \beta_j \theta_{ij} + \epsilon_i, \tag{7.1}$$

where y_i is the toxicity of the ith pesticide. The error, ϵ_i, is fixed as $N(0, \sigma_i^2)$ where σ_i is set as proportional to the number of observations within the same cluster as $\hat{x}^{(i)}$.

We separately consider three possible predicted docks for each of the $i = 1, \ldots, 17$ pesticides.

1. In Case 1 we set $\hat{x}^{(i)}$ to be the conformation with the minimum k_I.

2. In Case 2 we set $\hat{x}^{(i)}$ to be the median conformation within the largest cluster formed when using the RMSD between the predicted docks.

3. In Case 3 we set $\hat{x}^{(i)}$ to be the conformation, within the largest cluster, that is the greatest distance from μ_P.

The four variables we include within the linear model in Equation (7.1) are now discussed individually.

Inhibition constant

Let θ_{i1} denote the inhibition constant associated with $\hat{x}^{(i)}$. The inhibition constant is a measure of a pesticides ability to inactivate AChE. Intuitively, it should be an important indicator of the potential toxicity.

Comparing the ACh and pesticide docks

In the previous chapter we describe a method to calculate a measure of shape similarity between the natural ligand, ACh, and a pesticide. Now we produce a way of measuring the 'distance' between the predicted pesticide dock, $\hat{x}^{(i)}$, and the known ACh dock, μ. Let

$$\theta_{i2} = \sum_{k=1}^{10} \|\mu_k - \hat{x}^{(i)}_{\pi_k}\|^2,$$

where $\hat{x}^{(i)}_{\pi_k}$ represents the atom within $\hat{x}^{(i)}$ that is closest to μ_k. Note that all points in μ are considered and that one-to-many matches are allowed.

Comparing pesticide dock to protein receptor

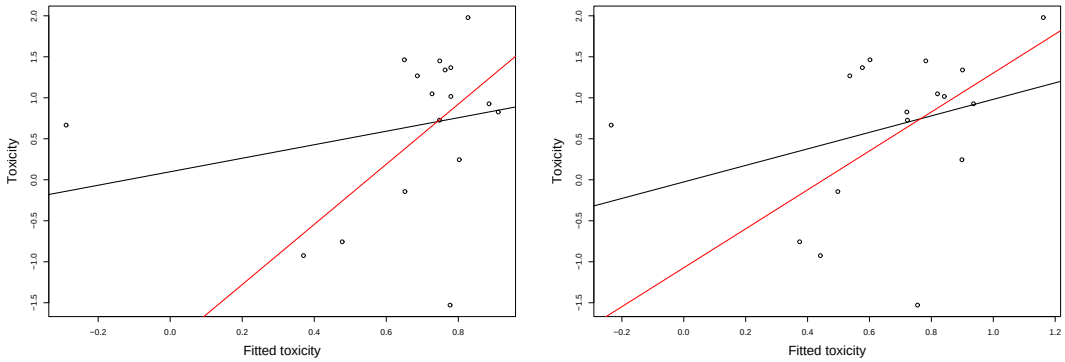
Finally we include a measure for the distance between $\hat{x}^{(i)}$ and the protein, μ_P. Similar to the previous variable defined in Subsection 7.3.4, we set

$$\theta_{i3} = \sum_{k=1}^{K_i} \|\hat{x}^{(i)}_k - \mu^P_{\pi_k}\|^2,$$

where $\mu^P_{\pi_k}$ represents the atom within μ_P that lies closest to $\hat{x}^{(i)}_k$. Again, all points in $\hat{x}^{(i)}$ are considered and one-to-many matches are allowed.

Conclusion

We found that in all three considered cases, the linear model in Equation (7.1) was not significant (which is not surprising considering the low number of observations



a) b)

Figure 7.5: Figures displaying the true toxicities, y, against the predicted toxicities, $\hat{y}$, for a) Case 2 and b) Case 3. The black line is the regression line considering all observations and the red line is the regression line when the fitted outlier is excluded.

considered). The correlation coefficient between the predicted and true toxicity is 0.05, 0.24 and 0.32 respectively in Cases 1, 2 and 3. Excluding the one fitted toxicity outlier in both Cases 2 and 3, the correlation coefficient becomes 0.51 and 0.50 respectively. The 95% one-tailed critical value using all observations is 0.41 and excluding the outlier is 0.43. Therefore indicating a significant correlation between the true and fitted toxicities in Case 2 and 3 when the fitted outlier is excluded. Figures 7.5 displays the true toxicities, y, against the predicted toxicities, $\hat{y}$ for both a) Case 2 and b) Case 3.

Here we have highlighted that it is not simply the predicted dock with the minimum k_I that provides the more accurate toxicity prediction. It is the observations within the dominant cluster (allocated using the RMSD between predicted docks alone) that provide a more accurate toxicity predictor. It would be interesting to see if the linear model in Equation (7.1) provides a more significant predictor of toxicity when considering a larger sample. This research has shown that useful toxicity indicators can be found even when the conformation of AChE is fixed and therefore the true complex conformation is not considered.

Chapter 8

Critical summary and further work

8.1 Introduction

In the final chapter, we consider each area of research separately when providing a critical summary and proposing ideas for further work.

8.2 Using EM to match pairwise gels, infer contamination levels and highlight missing proteins across sets

In Chapter 2 we introduced a statistical model to represent data across pairwise images. We considered two possible methods to calculate prior matching probabilities across images. The standard method assumes that a *true* marker is always correctly *allocated*. The adapted method deals with the possibility of slight marker misallocation within a warped image and does not assume that an allocated marker is always the true marker. We used an EM algorithm to estimate the superimposition of two images before inferring point correspondence across images. Finally, we provided methodology to account for missing or grossly misallocated markers.

In Chapter 4 we explored how the methodology introduced in Chapter 2 can be used to pool data across replicates, to investigate the quality of a dataset and finally, how it can be implemented to highlight the differences in proteins across groups of images.

8.2.1 Critical summary

The EM algorithm is strongly dependent on the starting transformation when aiming to superimpose one image onto another. As the data we consider within this research contains images with partial labelling (i.e. a corresponding set of points across images known as markers), the estimation of a good starting transformation is possible. We consider an affine transformation for the superimposition, so the fit will not account for local distortions that may exist within an image. However an affine transformation will account for a global warp and will avoid the overfitting often associated with attempts to account for local warping.

Throughout the experiments and applications described in Chapters 3 and 5, we used a conservative estimate of $\sigma_{ij}^2 = \sigma^2$ in Equation (2.9) to allow greater freedom for the distance between potential and known corresponding points. It should be noted that the values and conclusions will be strongly dependent on the estimate of σ^2.

We found that applying the standard method generally produced a more accurate superimposition than when applying the adapted method. That is, we found that setting the prior probability of corresponding markers matching to be one generally produced a more accurate superimposition than when the probabilities were allowed to vary. Though this is dependent on how the markers are allocated. If the markers are subsets of the points present in a western blot image, then the adapted method performs better for $\sigma < 7$. This is also likely to be the case when considering a lower number of markers across images.

Using the final output posterior probabilities to match points across images we found that relatively far apart points are often matched. If a point in x has a single nearby point in μ, the posterior probability of these two points matching will become

quite dominant even if the points are not that close. This a problem that becomes more prominant as μ and x become increasingly dissimilar. An alternative way of inferring the matches is by considering the pairwise distances between points after the application of the final transformation. Implementing this method dictates that only points within a certain distance threshold are matched. However, this does not address the negative influence caused throughout the running of the EM algorithm. Possible ways to counteract this problem are by increasing the coffin bin probability, i.e. the probability that points in x remain unmatched or by decreasing the variance between points, σ^2, within the algortihm.

Pooling data across replicate images can reduce computational expense in further analyses, but data will always be lost due to image warping and any inaccuracy within the matching method.

The method introduced for estimating contamination levels in a dataset of images assumes a constant distribution over all the images. If we consider images made by the same expert, in the same laboratory with the same equipment, this assumption is sensible. The probability of successfully observing a protein as a point on the image, p_*, is assumed to be constant over all points. In truth, this probability is likely to dependent on the intensity of the protein itself, as more intense proteins tend to produce larger and often darker spots on an image. The method is also dependent on the accuracy of the matches across images. However, when applying the method to the real dataset we found a similar relationship indicated between p_* and λ (the number of false points expected in an image) for each of the ten replicate pairs. Therefore showing the estimation of the contamination levels to be consistent across replicate pairs within the same dataset, and thus providing useful indicators of the dataset quality.

The production of the score indicating unique proteins across two groups of images will become increasingly computationally expensive as the number of images under consideration increases. Again, the method is dependent on the accuracy of the final superimposition and posterior matching probabilities proposed by the EM algorithm. However the score is not strongly influenced by varying levels of contamination within a

dataset and it provides a logical indicator of points unique to a group of images.

8.2.2 Future work

Future work could involve a comparison of the accuracy of the standard or adapted method as the number of markers across images vary. As well as considering how the protein sets differ across images or groups of images, it is also of interest to explore how the intensity of a particular protein varies. After employing the EM algorithm to infer point matches, we could further investigate how the intensities vary across the points matched.

The most appropriate value of σ^2 within Equation (2.9) was not investigated here. Sensitivity tests should be completed to find the optimal estimate of σ^2 for a particular dataset of interest.

As previously discussed, the probability of successfully observing a protein as a point on the image is likely to dependent on the intensity of the protein itself. The methodology could be modified to deal with the influence of protein intensity.

8.3 Molecular structure to predict pesticide toxicity

In Chapter 6 we test the hypothesis that the potential toxicity of a pesticide is related to the shape similarity between the pesticide and the substrate, ACh, of the protein, AChE, to which they both bind. We consider two different fixed conformations of ACh. In Chapter 7, we explore this hypothesis further by using a docking program to predict a pesticide dock and calculating a measure of shape similarity between the docked conformations of both the pesticide and ACh.

8.3.1 Critical summary

As we purely wanted to investigate whether the molecular shape of a pesticide helped predict the associated toxicity, information such as atom type was not considered, though

would be provide useful further information. The calculation of a shape similarity measure between a pesticide and ACh could provide a useful indicator of toxicity, however molecular shape is constantly changing due to the flexibility of a molecule. If fixing molecular shape is required, the minimum energy conformation and docked conformation are sensible conformations to consider and compare.

We found that the shape similarity measure provided a significant indicator of toxicity in the case of quail toxicity when the docked conformation of ACh was considered. We also found that using our shape similarity measure alongside known biological descriptors provided a more accurate prediction of associated toxicity than an online toxicity predictor.

Providing a measure of shape similarity between the docks of both a pesticide and ACh within the relevant protein, is likely to provide a better predictor of toxicity as it is is this form that toxicity is caused. However, the structure of the bound protein is dependent on the ligand with which it is binding and will rarely remain fixed as assumed within this research.

8.3.2 Future work

Future work would consist of a more detailed comparison of the molecular shapes involved in the complexes of ACh and AChE, and a pesticide and AChE. This time allowing flexibility within the protein to enable a more accurate dock prediction.

Bibliography

[1] T. Akutsu, K. Kanaya, and A. Ohyama, A.and Fujiyama, *Point matching under non-uniform distortions*, Discrete Applied Mathematics **127** (2003), 5–21.

[2] A. K. S. Alshabani, I. L. Dryden, C. D. Litton, and J. Richardson, *Bayesian analysis of human movement curves*, Applied Statistics **56** (2007), 415–428.

[3] N. L. Anderson, J. Taylor, A. E. Scandora, B. P. Coulter, and N. G. Anderson, *The TYCHO system for computer analysis of two-dimensional gel electrophoresis patterns*, Clinical Chemistry **27** (1981), 1807–1820.

[4] R. Anindya, R. L. Khan, H. Yaming, M. Marten, and R. Bubu, *Analysing two-dimensional gel images. technical report*, 2003, Technical Report. Department of Mathematics and Statistics, University of Maryland.

[5] R.D. Appel, P. M. Palagi, D. Walther, J. R. Vargas, J-C. Sanchez, F. Ravier, C. Pasquali, and D. F. Hochstrasser, *Melanie II - a third-generation software package for analysis of two-dimensional electrophoresis images: I. Features and user interface*, Electrophoresis **18** (1997), 2724–2734.

[6] R.D. Appel, J. R. Vargas, P. M. Palagi, D. Walther, and D. F. Hochstrasser, *Melanie II - a third-generation software package for analysis of two-dimensional electrophoresis images: II. Algorithms*, Electrophoresis **18** (1997), 2735–2748.

[7] R. E. Banks, M. J. Dunn, D. F. Hochstrasser, J-C. Sanchez, W. Blackstock, D. J. Pappin, and P. J. Selby, *Proteomics: new perspectives, new biomedical opportunities*, Lancet **356** (2000), 1749–1756.

[8] S. Belongie, J. Malik, and J. Puzicha, *Shape matching and object recognition using shape contexts*, IEEE Transactions on Pattern Analysis and Machine Intelligence **24(4)** (2002), 509–522.

[9] E. Benfenati, *Quantitative Structure-Activity Relationships (QSAR) for pesticide regulatory purposes*, Elsevier, 2007, Appendix F.

[10] M. Berkelaar, *Interface to lp_solve v. 5.5 to solve linear/integer programs*, 2008, R package.

[11] P. P. Bernard, D. B. Kireev, J. R. Chrétien, P-L. Fortier, and L. Coppet, *3D model of the acetylcholinesterase catalytic cavity probed by stereospecific organophosphorous inhibitors*, Journal of Molecular Modeling **4** (1998), 323–334.

[12] P. J. Besl and N. D. McKay, *A method for registration of 3-D shapes*, IEEE Transactions on Pattern Analysis and Machine Intelligence **14(2)** (1992), 239–256.

[13] E. Bettens, P. Scheunders, D. Van Dyck, L. Moens, and P. Van Osta, *Computer analysis of two-dimensional electrophoresis gels: A new segmentation and modeling algorithm*, Electrophoresis **18** (1997), 792–798.

[14] T. Bohme Leite, D. Gomes, M. A. Miteva, J. Chomilier, B. O. Villoutreix, and P. Tufféry, *FROG: a FRee Online druG 3D conformation generator*, Nucleic Acids Research **35** (2007), 568–572.

[15] R. F. Boyer, *Concepts in biochemistry*, John Wiley and Sons, 2002.

[16] T. J. Bradford, S. A. Tomlins, X. Wang, and A. M. Chinnaiyan, *Molecular markers of prostate cancer*, Urologic Oncology: Seminars and Original Investigations **24** (2006), 538–551.

[17] C. Bron and J. Kerbosch, *Algorithm 457: Finding all cliques of an undirected graph*, Communications of the ACM **16** (1973), 575–579.

[18] B. D. Bursulaya, M. Totrov, R. Abagyan, and C. L. Brooks III, *Comparative study of several algorithms for flexible ligand docking*, Journal of Computer-Aided Molecular Design **17** (2003), 755–763.

[19] D. A. Case, T. E. Cheatham III, T. Darden, H. Gohlke, R. Luo, K. M. Merz Jr., A. Onufriev, C. Simmerling, B. Wang, and R. J. Woods, *The Amber biomolecular simulation programs*, Journal of Computational Chemistry **26** (2005), 1668–1688.

[20] Y. Z. Chen and C. Y. Ung, *Prediction of potential toxicity and side effect protein targets of a small molecule by a ligand-protein inverse docking approach*, Journal of Molecular Graphics and Modelling **20** (2001), 199–218.

[21] H. Chui and A. Rangarajan, *A new point matching algorithm for non-rigid registration*, Computer Vision and Image Understanding **89** (2003), 114–141.

[22] D. A. Cosgrove, D. M. Bayada, and A. P. Johnson, *A novel method of aligning molecules by local surface shape similarity*, Journal of Computer-Aided Molecular Design **14** (2000), 573–591.

[23] A. D. J. Cross and E. R. Hancock, *Graph matching with a dual-step EM algorithm*, IEEE Transactions on Pattern Analysis and Machine Intelligence **20** (1998), 1236–1253.

[24] P. Cutler, G. Heald, I. R. White, and J. Ruan, *A novel approach to spot detection for two-dimensional gel electrophoresis images using pixel value collection*, Proteomics **3** (2003), 392–401.

[25] J. A. Doorn, C. M. Thompson, R. B. Christner, and R. J. Richardson, *Stereoselective inactivation of torpedo californica acetylcholinesterase by isomalathion: Inhibitory*

reactions with (1R)- and (1S)-isomers proceed by different mechanisms, Chemical Research in Toxicology **16** (2003), 958–965.

[26] A. W. Dowsey, M. J. Dunn, and G-Z. Yang, *The role of bioinformatics in two-dimensional gel electrophoresis*, Proteomics **3** (2003), 1567–1596.

[27] I. L. Dryden, J. D. Hirst, and J. L. Melville, *Statistical analysis of unlabeled point sets: comparing molecules in chemoinformatics*, Biometrics **63** (2007), 237–251.

[28] I. L. Dryden and K. V. Mardia, *Statistical shape analysis*, John Wiley and Sons, 1998.

[29] I. L. Dryden and G. Walker, *Highly resistance regression and object matching*, Biometrics **55** (1999), 820–825.

[30] J. El Yazal, S. N. Rao, A. Mehl, and W. Slikker Jr., *Prediction of organophosphorus acetylcholinesterase inhibition using three-dimensional Quantitative Structure-Activity Relationship (3D-QSAR) methods*, Toxicological Sciences **63** (2001), 223–232.

[31] E. Fischer, *Einfluss der configuration auf die wirkung der enzyme*, Berichte der deutschen chemischen Gesellschaft **27** (1894), 2985–2993.

[32] Proteome Informatics Group from the Swiss Institute of Bioinformatics, *Melanie 7.0 DIGE*, 2009, http://www.expasy.ch/melanie/.

[33] T. R. Fukuto, *Mechanism of action of organophosphorus and carbamate insecticides*, Environmental Health Perspectives **87** (1990), 245–254.

[34] C. A. Glasbey and K. V. Mardia, *A review of image warping methods*, Journal of Applied Statistics **25** (1998), 155–171.

[35] Molecular Networks GmbH, *Corina*, 2000, Erlangen, Germany.

[36] N. D. Gold, *Computational approaches to similarity searching in a functional site database for protein function prediction*, Ph.D. thesis, University of Leeds, School of Biochemistry and Microbiology, 2003.

[37] A. C. Good and W. G. Richards, *Explicit calculation of 3D molecular similarity*, Perspectives in Drug Discovery and Design **9/10/11** (1998), 321–338.

[38] A. Görg, W. Weiss, and M. J. Dunn, *Current two-dimensional electrophoresis technology for proteomics*, Proteomics **4** (2004), 3665–3685.

[39] P. J. Green and K. V. Mardia, *Bayesian alignment using hierarchical models, with applications in protein bioinformatics*, Biometrika **93** (2006), 235–254.

[40] W. Halle and E. Göres, *[Prediction of LD50 values by cell culture]*, Pharmazie **42** (1987), 245–248.

[41] I. Halperin, B. Ma, H. Wolfson, and R. Nussinov, *Principles of docking: An overview of search algorithms and a guide to scoring functions*, PROTEINS: Structure, Function and Genetics **47** (2002), 409–443.

[42] G. W. Horgan, A. M. Creasey, and B. Fenton, *Superimposing two-dimensional gels to study genetic variation in malaria parasites*, Electrophoresis **13** (1992), 871–875.

[43] R. Huey, G. M. Morris, A. J. Olson, and D. S. Goodsell, *A semiempirical free energy force field with charge-based desolvation*, Journal of Computational Chemistry **28** (2007), 1145–1152.

[44] European Bioinformatics Institute, *Acetylcholine SMILES formula*, 2008.

[45] J. T. Kent, K. V. Mardia, and C. C. Taylor, *Matching problems for unlabelled configurations*, 2004, LASR 2004 Proceedings. Leeds University Press, pp33-36.

[46] J. Klose, *Protein mapping by combined isoelectric focusing and electrophoresis of mouse tissues. A novel approach to testing for induced point mutations in mammals*, Humangenetik **26** (1975), 231–243.

[47] D. E. Koshland, *Application of a theory of enzyme specificity to protein synthesis*, Proceedings of the National Academy of Sciences, USA **44** (1958), 98–104.

[48] P. F. Lemkin, *Comparing two-dimensional electrophoretic gel images across the internet*, From genome to proteome: 2nd Sienna 2D electrophoresis meeting, 1996.

[49] P. F. Lemkin and L. E. Lipkin, *GELLAB: a computer system for 2D gel electrophoresis analysis. I. Segmentation of spots and system preliminaries*, Computers and Biomedical research **14** (1981), 272–297.

[50] P. F. Lemkin and L. E. Lipkin, *GELLAB: a computer system for 2D gel electrophoresis analysis. II. Pairing spots*, Computers and Biomedical research **14** (1981), 355–380.

[51] P. F. Lemkin and L. E. Lipkin, *GELLAB: a computer system for 2D gel electrophoresis analysis. III. Multiple two-dimensional gel analysis*, Computers and Biomedical research **14** (1981), 407–446.

[52] P. F. Lemkin, J. M. Myrick, Y Lakshmanan, M. J. Shue, J. L. Patrick, P. V. Hornbeck, G. C. Thornwal, and A. W. Partin, *Exploratory data analysis groupware for qualitative and quantitative electrophoretic gel analysis over the internet-Webgel*, Electrophoresis **20** (1999), 3492–3507.

[53] Non linear Dynamics Ltd., *Progenesis*, 2001, www.perkinelmer.com/proteomics.

[54] Bio-Rad Laboratories (UK) Ltd, *Pdquest 2-d analysis software*, 2009, http://www.selectscience.net/products/pdquest-2-d-analysis-software/?prodID=9997.

[55] J. Massoulié, L. Pezzementi, S. Bon, E. Krejci, and F. M. Vallette, *Molecular and cellular biology of cholinesterases*, Progress in Neurobiology **41** (1993), no. 1, 31–91.

[56] M. Y. Mizutani and A. Itai, *Efficient method for high-throughput virtual screening based on flexible docking: Discovery of novel acetylcholinesterase inhibitors*, Journal of Medical Chemistry **47** (2004), 4818–4828.

[57] G. M. Morris, D. S. Goodsell, R. S. Halliday, R. Huey, W. E. Hart, R. K. Belew, and A. J. Olson, *Automated docking using a Lamarckian genetic algorithm and an empirical binding free energy function*, Journal of Computational Chemistry **19** (1998), 1639–1662.

[58] R. J. Morris, R. J. Najmanovich, A. Kahraman, and J. M. Thornton, *Real spherical harmonic expansion coefficients as 3D shape descriptors for protein binding pocket and ligand comparisons*, Bioinformatics **21** (2005), 2347–2355.

[59] J. C. Nishihara and K. M. Champion, *Quantitative evaluation of proteins in one- and two-dimensional polyacrylamide gels using a fluorescent stain*, Electrophoresis **23** (2002), 2203–2215.

[60] M. North, *Principles and applications of stereochemistry*, CRC Press, 1998, pg 192.

[61] P. H. O'Farrell, *High resolution two-dimensional electrophoresis of proteins*, Journal of Biological Chemistry **250** (1975), 4007–4021.

[62] K-P. Pleissner, F. Hoffmann, K. Kriegel, C. Wenk, S. Wegner, A. Salström, H. Oswald, H. Alt, and E. Fleck, *New algorithmic approaches to protein spot detection and pattern matching in two-dimensional electrophoresis gel databases*, Electrophoresis **20** (1999), 755–765.

[63] F. A. Potra, X. Liu, F. Seillier-Moiseiwitsch, A. Roy, Y. Hang, M. R. Marten, B. Raman, and C. Whisnant, *Protein image alignment via piecewise affine transformations*, Journal of Computational Biology **13** (2006), 614–630.

[64] B. Raman, A. Cheung, and M. R. Marten, *Quantitative comparison and evaluation of two commercially available, two-dimensional electrophoresis image analysis software packages, Z3 and Melanie*, Electrophoresis **23** (2002), 2194–2202.

[65] M.L. Raves, M. Harel, Y.-P. Pang, I. Silman, A.P. Kozikowski, and J.L. Sussman, *Structure of acetylcholinesterase complexed with the nootropic alkaloid, (-)-huperzine A*, Natural Structural Biology **4** (1997), 57–63.

[66] M. Recanatini, A. Cavalli, and C. Hansch, *A comparitive QSAR analysis of acetylcholinesterase inhibitors currently studied for the treatment of Alzheimerś disease*, Chemico-Biological Interactions **105** (1997), 199–228.

[67] R. J. Richardson, T. B. Moore, U. S. Kayyali, J. H. Fowke, and J. C. Randall, *Inhibition of hen brain acetylcholinesterase and neurotoxic esterase by chlorpyrifos in vivo and kinetics of inhibition of chlorpyrifos oxon in vitro: Application to assessment of neuropathic risk*, Fundamental and Applied Toxicology **20** (1993), 273–279.

[68] N. J. Richmond, P. Willett, and R. D. Clark, *Alignment of three-dimensional molecules using an image recognition algorithm*, Journal of Molecular Graphics and Modelling **23** (2004), 199–209.

[69] M. Rogers, J. Graham, and R. P. Tonge, *Automatic construction of statistical shape models for protein spot analysis in electrophoresis gels*, Proceedings of the British Machine Vision Conference (2003), 369–378.

[70] K. Rohr, P. Cathier, and S. Wörz, *Elastic registration of electrophoresis images using intensity information and point landmarks*, Pattern Recognition **37** (2004), no. 5, 1035 – 1048.

[71] D. G. Rowlands, A. Flook, P. I. Payne, A. van Hoff, T. Niblett, and S. McKee, *GESA - a two-dimensional processing system using knowledge base techniques*, Electrophoresis **9** (1988), 820–830.

[72] M. A. Rubin, M. Zhou, S. M. Dhanasekaran, S. Varambally, T. R. Barrette, M. G. Sanda, K. J. Pienta, D. Ghosh, and A. M. Chinnaiyan, *α-methylacyl coenzyme a racemase as a tissue biomarker for prostate cancer*, Journal of American Medical Association **287** (2002), 1662–1670.

[73] Z. Smilansky, *Auomatic registration for images of two-dimensional protein gels*, Electrophoresis **22** (2001), 1616–1626.

[74] S. F. Sousa, P. A. Fernandes, and M. J. Ramos, *Protein-ligand docking: Current status and future challenges*, PROTEINS: Structure, Function and Bioinformatics **65** (2006), 15–26.

[75] O. Sperandio, M. A. Miteva, F. Delfaud, and B. O. Villoutreix, *Receptor-based computational screening of compound databases: The main docking-scoring engines*, Current Protein and Peptide Science **7** (2006), 1–25.

[76] P. Tarroux, P. Vincens, and T. Rabilloud, *HERMeS: A second generation approach to the automatic analysis of two-dimensional electrophoresis gels. Part V: Data analysis*, Electrophoresis **8** (1987), 187–199.

[77] C. C. Taylor, K. V. Mardia, and J. T. Kent, *Matching unlabelled configurations using the EM algorithm*, 2003, LASR 2003 Proceedings. Leeds University Press, pp19-21.

[78] I. V. Tetko, J. Gasteiger, R. Todeschini, A. Mauri, D. Livingstone, P. Ertl, V. A. Palyulin, E. V. Radchenko, N. S. Zefirov, A. S. Makarenko, V. Y. Tanchuk, and V. V.

Prokopenko, *Virtual computational chemistry laboratory - design and description*, Journal of Computer-Aided Molecular Design **19** (2005), 453–463.

[79] C. M. Thompson and R. J. Richardson, *Pesticide toxicology and international regulation (current toxicology series)*, John Wiley & Sons Ltd., 2005, pgs 89-127.

[80] M. Ünlü, M. E. Morgan, and J. S. Minden, *DIfference Gel Electrophoresis: A single gel method for detecting changes in protein extracts*, Electrophoresis **18** (1997), 2071–2077.

[81] W. Van Belle, G. Sjøholt, N. Ånensen, K-A. Høgda, and B. T. Gjertsen, *Adaptive contrast enhancement of two-dimensional electrophoretic protein gel images facilitates visualization, orientation and alignment*, Electrophoresis **27** (2006), 4086–4095.

[82] P. Vincens, *HERMeS: A second generation approach to the automatic analysis of two-dimensional electrophoresis gels. Part II: Spot detection and integration*, Electrophoresis **7** (1986), 357–367.

[83] P. Vincens, N. Paris, J-L. Pujol, C. Gaboriaud, T. Rabilloud, J-L. Pennetier, P. Matherat, and P. Tarroux, *HERMeS: A second generation approach to the automatic analysis of two-dimensional electrophoresis gels. Part I: Data acquisition*, Electrophoresis **7** (1986), 347–356.

[84] P. Vincens and P. Tarroux, *HERMeS: A second generation approach to the automatic analysis of two-dimensional electrophoresis gels. Part III: Spot list matching*, Electrophoresis **8** (1987), 100–107.

[85] P. Vincens and P. Tarroux, *HERMeS: A second generation approach to the automatic analysis of two-dimensional electrophoresis gels. Part IV: Data base organization and management*, Electrophoresis **8** (1987), 173–186.

[86] A. K. Walker, G. Rymar, and P. C. Andrews, *Mass spectrometric imaging of immobilized pH gradient gels and creation of 'virtual' two-dimensional gels*, Electrophoresis **22** (2001), 933–945.

[87] G. Walker, *Robust, non-parametric and automatic methods for matching spatial point patterns*, Ph.D. thesis, University of Leeds, Department of Statistics, 1999.

[88] J. Wang, R. M. Wolf, J. W. Caldwell, P. A. Kollman, and D. A. Case, *Development and testing of a general AMBER force field*, Journal of Computational Chemistry **25** (2004), 1157–1174.

[89] A. M. Wheelock and A. R. Buckpitt, *Software-induced variance in two-dimensional gel electrophoresis image analysis*, Electrophoresis **26(23)** (2005), 4508–4520.

[90] M. R. Wilkins, K. L. Williams, R. D. Appel, and D. F. Hochstrasser, *Proteome research: New frontiers in functional genomics*, Springer, 1997, pgs 187-220.

[91] J. Zhao, B. Wang, Z. Dai, X. Wang, L. Kong, and L. Wang, *3D-Quantitative Structure-Activity Relationship study of organophosphate compounds*, Chinese Science Bulletin **49** (2004), 240–245.

[92] M. J. Zvelebil and J. O. Baum, *Understanding bioinformatics*, Garland Science, 2008, pgs 613-624.