UNIVERSITY OF SOUTHAMPTON

FACULTY OF PHYSICAL AND APPLIED SCIENCES

School of Electronics and Computer Science

An Artificial Experimenter for Automated Response Characterisation

by

Christopher James Lovell

Thesis for the degree of Doctor of Philosophy

August 2011

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF PHYSICAL AND APPLIED SCIENCES
SCHOOL OF ELECTRONICS AND COMPUTER SCIENCE

Doctor of Philosophy

AN ARTIFICIAL EXPERIMENTER FOR AUTOMATED RESPONSE CHARACTERISATION

by Christopher James Lovell

Biology exhibits information processing capabilities, such as parallel processing and context sensitivity, which go far beyond the capabilities of modern conventional electronic computation. In particular the interactions of proteins such as enzymes are interesting, as they appear to act as efficient biomolecular computers. Harnessing proteins as biomolecular computers is currently not possible, as little is understood about their interactions outside of a physiological context. Understanding these interactions can only occur through experimentation. However, the size and dimensionality of the available experiment parameter spaces far outsize the resources typically available to investigate them, creating a restriction on the knowledge aquisition possible. To address this restriction, new tools are required to enable the development of biomolecular computation.

One such tool is autonomous experimentation, a union of machine learning and computer controlled laboratory equipment within in a closed-loop machine. Both the machine learning and experiment platforms can be designed to address the resource problem. The machine learning element attempts to provide techniques for intelligent experiment selection and effective data analysis that reduce the number of experiments required to learn from. Whilst resource efficient automated experiment platforms, such as lab-on-chip technology, can minimise the volumes of reactants per experiment. Here the machine learning aspect of autonomous experimentation is considered. These machine learning techniques must act as an artificial experimenter, mimicking the processes of successful human experimenters, through developing hypotheses and selecting the experiments to perform. Using this biological domain as motivation, an investigation of learning from a small set of noisy and sometimes erroneous observations is presented. Presented is a principled multiple hypotheses technique motivated from philosophy of science and machine learning for producing potential response characteristics, combined with active learning techniques that provide a robust method for hypothesis separation and a Bayesian surprise method for managing the exploration–exploitation trade-off between new feature discovery and hypothesis disproving. The techniques are validated through a laboratory trial where successful biological characterisation has been shown.

Contents

List of Figures

List of Tables

Nomenclature

x	An experiment parameter
y	An observation
h	A hypothesis
$\hat{h}(x)$	The prediction of a hypothesis for an experiment
$f(x)$	An underlying behaviour
$\hat{f}(x)$	A prediction of an underlying behaviour
ϵ	Gaussian noise applied to an observation
ϕ	Shock noise applied to an observation
$C(h)$	The confidence of a hypothesis
$E(X)$	An error function
$\mathbf{X}$	The set of previous performed experiment parameters
$\mathcal{D}$	The set of previous experiment and observation pairs
$\mathcal{H}$	The working set of hypotheses under consideration
$\mathcal{L}$	A loss function
$\boldsymbol{\beta}$	Parameter values vector
$\hat{\boldsymbol{\beta}}$	Predicted parameter values vector
$\hat{\boldsymbol{\beta}}'$	Transpose of the predicted parameter vector
$\mathbf{x}$	Vector of independent variables (as basis functions in Chapter 3)
$\mathbf{x}_i$	Element i from vector of independent variables
$\mathbf{X}$	Design matrix
$\mathbf{y}$	Vector of dependent variables
$\mathbf{W}$	Weight matrix
λ	Regularisation parameter
$\mathbf{N}$	Smoothing spline design matrix
$\mathbf{\Omega_N}$	Smoothing spline regularisation matrix
ξ	Spline knot
$\mathbf{A}$	Hat matrix
$\mathbf{D}$	Design matrix for thin plate spline
$\mathbf{P}$	Null space for thin plate spline
$\varphi(r)$	Radial basis function
$\begin{bmatrix} x_1 & x_2 \end{bmatrix}$	Vector containing two elements
$\|\mathbf{p} - \mathbf{q}\|$	Euclidean distance between two vectors

$S_{w,\lambda}(f)$	Smoothing spline with specific weight vector and smoothing parameter
N	Number of hypotheses under consideration
h_t	The 'true' hypothesis in the hypothesis separation problem
x^*	Selected experiment parameter
$D(h_i, h_j, x)$	Discrepancy between two hypotheses for an experiment parameter
$A(h_i, h_j)$	Agreement between two hypotheses over all available observations
M	Moles

Declaration of Authorship

I, Christopher James Lovell, declare that the thesis entitled *An Artificial Experimenter for Automated Response Characterisation* and the work presented in the thesis are both my own, and have been generated by me as the result of my own original research. I confirm that:

- this work was done wholly or mainly while in candidature for a research degree at this University;
- where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- where I have consulted the published work of others, this is always clearly attributed;
- where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
- I have acknowledged all main sources of help;
- where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
- parts of this work have been published as: (Lovell et al., 2011), (Lovell et al., 2010a), (Lovell et al., 2010c), (Lovell and Zauner, 2009)

Signed:.....

Date:.....

Acknowledgements

There are many people I would like to thank who have provided support throughout my undertaking of this research.

First I would like to thank my supervisor Klaus-Peter Zauner for steering me away from conventional computer science topics and into a world of biocomputers and slime mould. I am grateful for the guidance, the always open door and the opportunity to undertake work on a new and exciting field of research with a real physical application. I would also like to thank my advisor Steve Gunn for ensuring the project maintained a strong, principled machine learning grounding and for providing support when the maths just would not work. I also thank them both for giving me the opportunity to travel around the world throughout my research.

I would also like to thank Gareth Jones, who undertook his PhD on the hardware portion of this project at the same time as my candidature, for his support, friendship and help in finding the necessary distractions required to complete a thesis. To my Mum and John Warner for all the support and making it possible for me to undertake all my university studies.

Finally I would like to thank Eric Cooke, for simultaneously ensuring I maintained an income, remained well fed and retained my sanity throughout this research. Alvin Ittoo for his friendship and many trips to all-you-can-eat restaurants. Soichiro Tsuda for guidance on how to manage a PhD and for occasionally providing us with a broken slime mould robot to fix and distract us from writing up. Vijay Pakka, who provided guidance on how to get through a PhD. I also thank Hywel Morgan for the use of his lab. Thanks to the anonymous reviewers throughout the thesis, especially those from the Discovery Science community who were extremely supportive and interested in this work. I gratefully acknowledge the financial support from Microsoft Research and the PASCAL2 Network of Excellence.

The reported work was supported in part by a Microsoft Research Faculty Fellowship to Klaus-Peter Zauner.

This work was supported in part by the IST Programme of the European Community, under the PASCAL2 Network of Excellence, IST-2007-216886. This publication only reflects the authors' views.

Chapter 1

Introduction

In order to survive, biological systems have evolved efficient yet complicated information processing capabilities, many of which go far beyond the capabilities of modern conventional electronics (Zauner and Conrad, 2001b). The interactions of such complex biological systems demonstrate effective parallel computing and context sensitivities, which if harnessed could enable a new range of computational mechanisms (Zauner, 2005). Specifically, enzymes appear to act as biomolecular information processors, providing context sensitive pattern recognition that behave differently depending on the chemical environment it is within (Zauner and Conrad, 2001a). The current state of the art in biological computation attempts to manipulate the biology to mimic conventional logic gate based computation (de Silva and Uchiyama, 2007), for example using DNA (Seelig et al., 2006) and enzymes (Zauner and Conrad, 2001a). However, such manipulation by these techniques provide little engineering benefit, as they result in computational mechanisms that are slower and more fragile than their electronic counterparts. Instead, advantages of using biological systems for information processing will occur when new modes of computation are demonstrated that cannot be replicated by current conventional techniques. That is to say, rather than forcing the biology to behave in a prescribed manner to solve a particular problem, the biology should be studied to understand its behaviours, then work towards identifying the problems it can efficiently solve. Such study can only be achieved through physical experimentation to characterise the responses of the interactions of different biological systems. However, experimentally investigating these biological interactions is restricted by available resources, where the size and dimensionality of the potential experiment parameter space will far outsize the resources available. Additionally biochemical experimentation is error prone, meaning that not all experiments will yield observations that are representative of the true behaviours that should be observed. Therefore new engineering tools are required to assist the biochemist to minimise the resources used but maximise the information gained.

The resource limitations can be addressed through two general approaches. One approach is to reduce the number of experiments required to effectively characterise the

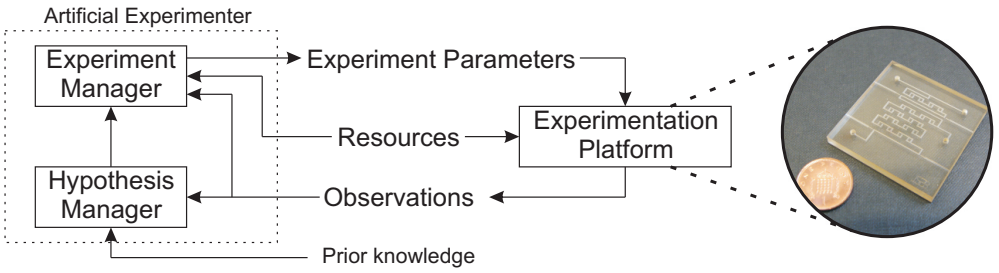


Figure 1.1: Flow of experimentation between an artificial experimenter and an automated experimentation platform. A prototype of a lab-on-chip platform in development for conducting the experiments on is shown.

system under investigation. The other approach is to reduce the physical chemical resources required per experiment. To reduce the number of experiments required to perform effective characterisation, machine learning can be utilised to provide data efficient hypothesis creation and intelligent experiment selection. Whilst resource efficient technologies such as lab-on-chip, can be employed to reduce the resources required per experiment. A machine that combines these two approaches would provide a resource efficient mechanism for experimentation.

A union of resource efficient artificial intelligence techniques and automated laboratory hardware can be made to produce an autonomous experimentation machine. An autonomous experimentation machine can automatically propose hypotheses, determine experiments to perform, and then physically performs those experiments on an automated experimentation platform. Autonomous experimentation is conducted within a closed-loop with no human interaction, as shown diagrammatically in Figure 1.1. Additional to the resource savings the artificial intelligence and hardware platform may bring, such a machine has the additional benefit of allowing the scientist to redirect their time from performing monotonous characterisation experiments, to theorising about the results obtained, or identifying applications for the behaviours discovered. Currently autonomous experimentation is in its infancy, with few examples of autonomous experimentation machines capable of laboratory based discovery appearing in the literature (Żytkow et al., 1990; Du et al., 1999a; Matsumaru et al., 2002; King et al., 2004; Bonowski et al., 2010). In the present thesis, the first issue of developing techniques for reducing the number of experiments required to accurately characterise a phenomena under investigation is considered, through the development of an artificial experimenter. A separate body of work is considering the development of a lab-on-chip experimentation platform, which will allow for experiments to be performed with microscale resource requirements (Jones, 2009).

1.1 Artificial Experimenter

An artificial experimenter is designed to mimic the decisions a human experimenter makes. It will mimic methods for choosing experiments, determining the validity of observations and then determining the response curves for phenomena being investigated from the observations obtained. Whilst autonomous experimentation is still in its infancy, the algorithmic side of the problem, described throughout as the artificial experimenter, is similar to the more established fields of statistical experiment design (Fisher, 1935; Box and Draper, 1987), active learning (MacKay, 1992; Cohn et al., 1996; Settles, 2009), and work in computational scientific discovery (Langley et al., 1987). Like active learning, an artificial experimenter will analyse the data available and determine the next experiment or data point to sample, with the goal being to minimise the number of experiments or samples that are required in the learning process. For data analysis, machine learning techniques are also well suited to modelling characterisation data and hypotheses through regression (Fisher, 1925; Vapnik, 1995; Wahba, 1990; Gunn, 1998; Rasmussen and Williams, 2006). However an artificial experimenter may benefit from looking beyond more mathematically rigid approaches like those often applied in active learning, to consider techniques that are more ad-hoc in mimicking the methods employed by successful human experimenters. These more ad-hoc approaches are more frequently seen in computation scientific discovery methods (Kulkarni and Simon, 1990). For an artificial experimenter to be successful, we consider that a balance between mathematical guarantees and ad-hoc human 'feeling' is required.

The design for an artificial experimenter must consider fundamental factors of physical experimentation. First, as previously discussed, resources will be limited with respect to the experiment parameter spaces being investigated. This means that the artificial experimenter should be able to learn from a small number of observations, perhaps no more than a handful of experiments per parameter dimension. Second, all physical experiments are inherently noisy, where the observations will be distorted by some experimental noise. Finally, the observation for an experiment is never guaranteed to be a valid representation of the phenomena being investigated, particularly in biological experimentation where there is a high level of variability in the systems investigated, meaning that erroneous observations are possible. These factors mean that the goal for an artificial experimenter is to be able to provide an accurate representation of an underlying phenomena under investigation, using a small, noisy and potentially erroneous set of observations.

Currently no techniques in the literature consider learning from small, noisy and potentially erroneous sets of observations. Some active learning techniques consider scenarios where there is no noise (Seung et al., 1992), whilst other experiment selection techniques can struggle with experimental noise (Atkinson and Fedorov, 1975a). The

learning techniques that do consider experimental noise, albeit not erroneous observations, are evaluated based on their ability to learn using a hundred observations in a single parameter dimension (Sugiyama, 2006; Burbidge et al., 2007), or several hundred experiments per parameter dimension (Du and Zhang, 2005), which are far beyond the resource restrictions considered here.

Therefore, there are two key issues of an artificial experimenter, firstly the issue of how to represent and express hypotheses, and secondly how determine the next experiment to perform. With the field in its infancy, generality of both issues is preferential. Efficiency of experimentation in terms of resource cost is also key. Computational efficiency is not as important, so long as algorithms do not become NP-hard, as physically performing experiments is a slow and expensive process in comparison to computation. Postulating the accuracy of observations is also a factor, as observations obtained from experimentation are noisy. Add in the factor that occasionally experimentation yields erroneous observations, particularly when using biomolecular substrates where an enzyme could die rendering observations meaningless, determining accuracy becomes a complex task without performing the expensive process of running many repeat experiments. This then leads questions that an autonomous experimentation system must consider: What do the observations say about the phenomenon being investigated?; Is the next experiment proposed likely to obtain a large amount of useful information?; Are the observations obtained accurate?; If the predictions of the hypotheses do not match the new observation obtained, is it the hypotheses that are wrong or is it the observation that is wrong? The present thesis looks to address the issue of learning from small, noisy and potentially erroneous sets of observations. To do this, part of this work considers methods for creating and managing hypotheses in such a scenario, whilst another looks at developing strategies for choosing a minimal amount of experiments to learn from.

1.1.1 Hypothesis Management

Hypotheses in the computational scientific discovery literature, often require additional domain information so that mechanistic hypotheses can be formed (Lindsay et al., 1993; Kulkarni and Simon, 1990; Valdés-Pérez, 1994; King et al., 2004). However some have limited the amount of domain information required, albeit at the expense of not being able to produce mechanistic hypotheses (Żytkow et al., 1990; Matsumaru et al., 2004). In experimentation with biomolecular substrates, such large amounts of domain information do not exist, so the majority of previous approaches cannot be utilised. For characterisation of response behaviours, statistical machine learning provides techniques of regression that allow for mappings between inputs and responses (Bishop, 2006, Ch. 3), where there are several techniques available for this (Wahba, 1990; Gunn, 1998; Rasmussen and Williams, 2006). Such techniques have been used in autonomous experimentation before, where weighted least squares has been used to find polynomial

models of data (Zembowicz and Żytkow, 1991), albeit with a larger number of observations available.

With a handful of observations, identifying erroneous observations otherwise known as outliers, is difficult due to not having sufficient evidence to confirm the validity of a particular observation. Therefore traditional parameter learning techniques such as cross-validation become unable to tune a single regression calculation so that it ignores outlying observations correctly. Instead, considering multiple hypotheses in parallel, each with a different view of the data available, is a more principled approach to building good representations of the underlying behaviours from limited noisy data (Lovell et al., 2011). Such multiple hypotheses approaches are promoted within scientific philosophy so as to ensure alternate views about the data available are considered (Chamberlin, 1890). Within machine learning ensemble based approaches, such as query by committee (Seung et al., 1992), follow the same idea and consider several models in parallel. However, there is disagreement in how best to apply these approaches and build the sets of hypotheses and there is still much room for improvement (Settles, 2009). When the factors of experimentation described above are taken into consideration, then a principled approach to developing multiple hypotheses can be obtained, by allowing hypotheses to have differing opinions of the validity of their observations (Lovell et al., 2010c).

1.1.2 Experiment Management

The hypotheses can only be developed if a good set of observations are obtained, which identify the features, such as peaks and troughs, of the phenomena under investigation. As hypotheses can never be proved, only disproved (Buck, 1975), the best hypothesis can only be identified from a set of competing hypotheses by obtaining observations that disprove all other alternatives. However, after disproving all other alternatives, there is no guarantee that the hypothesis is an accurate representation of the phenomena, as the experiment selection may have missed features of the phenomena, experimental error may have skewed the observations, or the hypothesis modelling technique may not be able to represent the phenomena being investigated. Assuming hypotheses that accurately represent the behaviour are possible if provided with a suitable training set, experiments must be chosen that balance obtaining observations that explore the experiment parameter space and discover new features of the phenomena, with those experiments that evaluate the hypotheses or the validity of the observations. This type of balance is called the exploration-exploitation trade-off (Auer, 2002). The exploration-exploitation trade-off manages the use of experiments that search from things not known, with experiments to test things that are known. With the process of choosing exploration experiments being trivial, through placing experiments the furthest away from any previously performed experiment, methods for exploitation are slightly more diverse.

Experimental design provides techniques, such as T-optimality (Atkinson and Fedorov, 1975a), for choosing the set of experiments that will optimally differentiate between a set of hypotheses. However, T-optimality suffers in some scenarios, such as when there is experimental error, or when there are well performing hypotheses with similar structure (Atkinson and Fedorov, 1975a). As such these techniques are not suitable for the experimentation scenario considered here, where observations are noisy, and hypotheses will be learnt from a small set of observations meaning that with multiple hypotheses in consideration, there will likely be similar hypotheses amongst them. Alternatively, the variance of hypotheses predictions have been considered (Burbidge et al., 2007), but such techniques suffer if any of the predictions are outliers (Lovell et al., 2011). In this thesis, methods for separating sets of hypotheses are considered, and a more robust approach to experiment selection for separating hypotheses efficiently is presented. Further to this, methods for addressing the exploration-exploitation problem are presented.

1.2 Objectives

The core aim of this thesis is to develop machine learning techniques capable of being implemented as an artificial experimenter, which can be used in response characterisation. The motivation for this system is to provide a tool to aid the characterisation of biomolecular substrates, so that new computational mechanisms can be sought. However, the techniques developed should not be made specific to the problem of enzyme characterisation, rather they should aim to be generalised.

This thesis has the following objectives:

- **Effective hypothesis management.** Resources in experimentation are limited, meaning few observations are available. Of those that are available, obtaining them through physical experimentation will mean that all will be noisy and potentially erroneous. A method for developing accurate hypotheses with a very small, noisy and potentially erroneous training set is required.
- **Efficient experiment placement.** With limited resources, the placement of experiments is critical to being able to develop hypotheses with accurate representations of the underlying phenomena investigated, through both discovering behaviours not yet captured by the hypotheses and evaluating the hypotheses currently under consideration. Techniques for minimising the number of experiments required to achieve this are required.
- **Development and validation of an artificial experimenter.** Combining the hypothesis management component with the experiment placement component, can allow for an artificial experimenter to be developed. This artificial experimenter needs to be evaluated in simulation to confirm that the technique performs

better than alternate techniques, in terms of the number of experiments used and the quality of the hypotheses produced. The artificial experimenter then needs to be evaluated within a laboratory setting with reactants of the target type, to validate its worth in real physical experimentation.

1.3 Research Contributions

The contribution of this thesis is through the investigation and development of techniques for learning with limited numbers of noisy and potentially erroneous training observations, applied to biomolecular response characterisation.

- A multiple hypotheses based approach has been designed that considers the validity of observations with questionable accuracy in parallel, through competing hypotheses.
- Experiment selection techniques designed to differentiate between a set of hypotheses have been evaluated, with a new method proposed that is more robust than currently used methods.
- Methods for addressing the exploration-exploitation trade-off have been considered, so as to develop a method for experiment selection that allows the formation and evaluation of sets of hypotheses, starting with no prior experimental information.
- The hypothesis and experiment management components have been combined to form an artificial experimenter, which has been evaluated within a simulated scenario to evaluate how the techniques compare to existing methods.
- The artificial experimenter has been tested within a laboratory setting through proposing and evaluating hypotheses, along with determining the experiments to perform, in the characterisation of the co-enzyme NADH. These experiments were conducted manually in the laboratory and demonstrated the systems ability to provide a representative hypothesis of the underlying behaviour with few experiments.

These contributions led to the following publications as conference oral presentation:

- Lovell, C. J., Jones, G., Gunn, S. R., and Zauner, K.-P. (2010a). An artificial experimenter for enzymatic response characterisation. In *13th International Conference on Discovery Science*, pages 42–56, Canberra, Australia
  Won the Carl Smith Award for best student paper at DS2010.

- Lovell, C. J., Jones, G., Gunn, S. R., and Zauner, K.-P. (2010c). Characterising enzymes for information processing: Towards an artificial experimenter. In et al., C. S. C., editor, *9th International Conference on Unconventional Computation*, volume 6079, pages 81–92, Tokyo, Japan

- Lovell, C. J., Jones, G., Gunn, S. R., and Zauner, K.-P. (2010b). Autonomous experimentation: Coupling active learning with computer controlled microfluidics (abstract). In *Active Learning and Experimental Design workshop at AISTATS*, Sardinia

Where the abstract for the Active Learning and Experimental Design workshop led to a Journal of Machine Learning Research Workshop and Conference proceedings paper:

- Lovell, C. J., Jones, G., Gunn, S. R., and Zauner, K.-P. (2011). Autonomous experimentation: Active learning for enzyme response characterisation. *JMLR: Workshop and Conference Proceedings*, 16:141–154

An overview of the work carried out was presented to a drug discovery industrial audience as an invited talk:

- Lovell, C. J., Jones, G., and Zauner, K.-P. (2009). Autonomous experimentation: Coupling machine learning with computer controlled microfluidics (abstract). In *European Laboratory and Robotics Interest Group Robotics Workshop, ELRIG Drug Discovery 2009*, Liverpool, UK

Along with the following poster presentations:

- Lovell, C. J. and Zauner, K.-P. (2009). Towards algorithms for autonomous experimentation (extended abstract). In *Eighth International Conference on Information Processing in Cells and Tissues (IPCAT 2009)*, pages 150–152, Ascona, Switzerland

- Lovell, C. J. and Zauner, K.-P. (2008a). Autonomous experimentation: Methods for characterising molecular computing substrates (poster). In *SemiBiotic Systems Conference*, Malta

- Lovell, C. J. and Zauner, K.-P. (2008b). Autonomous experimentation: Methods for characterising molecular computing substrates (poster). In *3rd Microsoft Research Summer School*, Cambridge, UK

Additionally, contributions to work discussing the development of the lab-on-chip device for autonomous experimentation:

- Jones, G., Lovell, C. J., Morgan, H., and Zauner, K.-P. (2011). Organising chemical reaction networks in space and time with microfluidics. *International Journal of Nanotechnology and Molecular Computation (IJNMC)*, 3(1):35–56

- Jones, G., Lovell, C. J., Morgan, H., and Zauner, K.-P. (2010b). Organising chemical reaction networks in space and time with microfluidics. In *1st International Workshop on Computing with Spatio-Temporal Dynamics*, Tokyo, Japan

- Jones, G., Lovell, C. J., Morgan, H., and Zauner, K.-P. (2010a). Characterising enzymes for information processing: Microfluidics for autonomous experimentation (abstract). In *9th International Conference on Unconventional Computation*, page 191, Tokyo, Japan

Finally, other work on exploiting the computational properties of biological systems contributed to during candidacy:

- Gough, J., Jones, G., Lovell, C. J., Macey, P., Morgan, H., Revilla, F., Spanton, R., Tsuda, S., and Zauner, K.-P. (2009). Integration of cellular biological structures into robotic systems. *Acta Futura*, 3:43–49

1.4 Outline of Thesis

The structure of the thesis is as such:

Chapter 2 (Computational Scientific Discovery) discusses the background of computational methods employed in discovery and experimentation based problems. This chapter first considers what experimentation is through the review of philosophy of science ideas. The chapter then continues to review the body of work described in the computational scientific discovery and active learning fields.

Chapter 3 (Managing Multiple Hypotheses) discusses the practical issues of developing hypotheses for response characterisation. The technique used to generate hypotheses, then refine them so as to perform targeted learning from the information available, is discussed, using ideas first published in (Lovell et al., 2010c). Additionally the problems of utilising a single hypothesis in a domain where there are few, noisy and potentially erroneous observations, are highlighted.

Chapter 4 (Separating Sets of Hypotheses) discusses a theoretical problem of choosing the minimal number of experiments required to take a set of competing hypotheses and disprove all but the true underlying hypothesis. This chapter considers a number of different active learning techniques and evaluates them on a generalised toy problem, first published in (Lovell et al., 2010a). Some of the techniques tested are existing

within the active learning literature for regression separation or have been converted from similar classification problems. Whilst others techniques have been adapted from existing machine learning techniques not initially designed for this purpose, or are new techniques.

Chapter 5 (Design for an Artificial Experimenter) discusses the combination of the hypothesis management and experiment selection strategies considered up to this point, to form an artificial experimenter that can guide experimentation, as presented in (Lovell et al., 2011, 2010a). This chapter also considers how the exploration-exploitation trade-off can be addressed, to allow for experiments to explore the space in order to allow the development of accurate hypotheses, whilst also allowing for experiments to evaluate the hypotheses using the methods of hypothesis separation considered in Chapter 4.

Chapter 6 (Evaluating in Simulated Scenarios) discusses the evaluation of the artificial experimenter within a simulation. The simulation allows for the true underlying behaviour to be known, so that the quality of the hypotheses can be judged and compared.

Chapter 7 (Evaluating in Laboratory Scenarios) discusses the evaluation of the artificial experimenter within a laboratory trial, first published in (Lovell et al., 2010a). The artificial experimenter guides manually performed experiments to characterise the co-enzyme NADH.

Chapter 8 (Conclusions) discusses the conclusions and further work from this thesis.

Chapter 2

Computational Scientific Discovery

Computational systems provide excellent mechanisms for identifying patterns within data (Vapnik, 1995; Bishop, 2006). A computational system has the potential to outperform a human in pattern recognition, due to the computer being able to handle far larger and more complex data sets. However computers are not as adept at discovering new knowledge (Żytkow, 1993). Discovery requires observing previously unseen behaviours and then hypothesising either the causes for the observations or inferences that can be made about future observations. A human expert in the domain the discovery is occurring in can more easily outperform a computational system in discovery, as humans will often be able to draw upon a wider range of knowledge or heuristics than a computational system can be programmed with (Kulkarni and Simon, 1990). To investigate the discovery abilities of computational systems, the field of computational scientific discovery was developed (Langley et al., 1987; Darden, 1997; Gaber, 2010).

Computational scientific discovery is based upon understanding how successful scientists achieved discoveries and in particular the role of experimentation in discovery (Langley et al., 1987). The field maintains a close link with the philosophy of science (Williamson, 2010). Drawing on work from the philosophy of science allowed for the creation of abstract processes for discovery that were based on techniques employed by successful scientists (Żytkow and Simon, 1986; Gooding, 1990). These abstract processes enabled some early construction of computational discovery systems (Kulkarni and Simon, 1990). Early computational scientific discovery worked to understand how computational systems could aid the scientist in discovery, but a grander aim was whether a computational system could discover new knowledge (Żytkow, 1993).

An early implementation of such a system was DENDRAL, which built an expert system for scientific reasoning, with one aspect of the project being the development of algorithms for aiding structure elucidation in organic chemistry (Lindsay et al., 1993).

Parts of this project were successfully used within laboratories external to the original research. The problem of discovery from large amounts of provided experimental data is an area still receiving considerable focus (Hey et al., 2009; Evans and Rzhetsky, 2010), where ontological representations of scientific data has also received significant attention (Soldatova et al., 2006; Soldatova and King, 2006; Bundy, 2008; McGuinness et al., 2008). The purpose for such systems is to take a large body of known information and discover relations or concepts within the data that are not known. Several other techniques have investigated domains where prior information exists and where hypotheses can be formed within a more logical framework, often because the observations or parameter domains are discrete and limited (Langley et al., 1987; Fischer and Żytkow, 1990; Valdés-Pérez, 1990; Żytkow and Fischer, 1991; Karp, 1993; King et al., 2004). Examples of these systems may be to determine reaction pathways (Valdés-Pérez, 1990), or identifying hidden structures of the system being investigated (Fischer and Żytkow, 1990). In these systems techniques that can exploit the structural, associative or logical information available can be employed, such as decision trees or inductive logic programming (Hoffman and Mahidadia, 2010). These domains allow prior knowledge to be used to automatically determine a large number of more mechanistic hypotheses, which can each be evaluated experimentally to determine whether or not the hypothesis is valid.

However such logical frameworks do not fit all scientific discovery problems. Additionally scientific discovery often has the problem of actually obtaining the experimental data in the first place. More often than not the parameter spaces available are very large, the number of experiments available is small, and performing experiments is slow. Therefore, there is what can be described as a data acquisition bottleneck (Żytkow et al., 1990) or knowledge acquisition bottleneck (Hoffman and Thakar, 1991). To address this bottleneck, proposals have been made to try and automate as much of the data acquisition process as possible, by combining automated hardware with data analysis techniques that can automatically choose experiments to perform (Rechenberg, 1965; Żytkow et al., 1990; Du and Lindsey, 2002; Matsumaru et al., 2002; King et al., 2004). This produced a new area of computational scientific discovery, as techniques were now needed that not only extracted discoveries from the available data, but could also guide the collection of the experimental data and be proactive within the discovery process.

There are many names describing these closed-loop experimentation systems in the literature. Such names as *automatic research machines* (Rechenberg, 1965), *machine discovery* (Kulkarni and Simon, 1990), *automated discovery* (Żytkow et al., 1990), *robot-scientist* (Żytkow, 1997; King et al., 2004), *onboard science* (Stolorz and Cheeseman, 1998), *autonomous experimentation* (Plouvier et al., 1992; Du and Lindsey, 2002; Matsumaru et al., 2004; King, 2006), *artificial scientist* (Muggleton and Zauner, 2006), *computational scientific discovery* (Langley et al., 1987; Darden, 1997), and *automated science* (Waltz and Buchanan, 2009), have all described, with varying levels of implementation, similar styles of experimentation system. That is, a system that closes the

loop between automatically evaluating and modelling experimental observations, deciding on experiments to perform, and then physically performing those experiments. Added to these, there is the field of active learning, a machine learning technique that autonomously chooses the data it will learn from (MacKay, 1992; Cohn et al., 1996; Settles, 2009).

This notion of autonomous experimentation, where a machine can design experiments, have them performed by an automated experiment platform, then propose hypotheses to represent the data obtained, need not be thought of as working towards a replacement for the human scientist in discovery. In many cases this cannot be the case, as many instincts or abilities used by human scientists have not been captured computationally and may never be. Additionally the knowledge required to make certain discoveries, such as mechanistic ones, can only be achieved with a vast body of domain information, which may be unrealistic to encode or link together for a computational system to use. Therefore, the most common usage for such autonomous experimentation machines will be to assist the scientist, rather than replace them. For example in aiding the acquisition of data in an intelligent manner, so as to conserve resources and provide initial hypotheses for the data.

Early work in automating experimentation proposed restrictions and types of experimentation machines. Primarily was the assumption that no one machine would be able to perform any general experimentation and that machines would be limited to a particular class of problems (Rechenberg, 1965). The author proposed three categories of autonomous experimentation systems, however some of their original meaning may have been lost through the translation of the proposal. The translation states the categories as systems that learn: what happens; why something happens; and how something happens (Rechenberg, 1965). We now restate those categories, but include a fourth category to add clarity to the originally proposed categories. The four categories are: machines that can probe a phenomenon to discover what happens; machines that can characterise behaviours; machines that can discover the mechanisms behind the behaviours observed; and machines that can discover how to make a behaviour occur. To illustrate these four categories further, the first category poses the question: *"What happens when we do X?"*, the second asks: *"What characteristics of Y can we learn by performing X?"*, the third asks: *"Why does doing X give us Y?"*, and the fourth asks: *"How do we make Y?"*.

These four categories still categorise computational experimentation systems today, with examples falling into one or more of those categories. Examples of the first category are approaches that sample observations without trying to understand the behaviours, such as laboratory automation equipment or satellite and space robotics (Stolorz and Cheeseman, 1998) and simple sensor networks (Grismer, 1992). Examples from the second category exist in computation scientific discovery and often require a minimal amount of *a priori* information (Żytkow et al., 1990; Matsumaru et al., 2002; Lovell

et al., 2010a). The third example contains many of the approaches from computational scientific discovery and mostly require large amounts of *a priori* information (Kulkarni and Simon, 1990; Lindsay et al., 1993; Valdés-Pérez, 1994; King et al., 2004; Schmidt and Lipson, 2009). Examples of the fourth objective are less frequent in computational scientific discovery literature, however examples exist that try to optimise some reaction condition (Matsumoto et al., 2002b), and examples exist in the drug discovery community (Warmuth et al., 2003).

In this chapter the background of computational scientific discovery is discussed. First the fundamentals of experimentation will be considered by reflecting on perspectives from the philosophy of science, to develop a more mathematical framework for experimentation. Then we will review techniques from computational scientific discovery and active learning for their suitability in laboratory discovery and their ability to learn from small numbers of noisy and erroneous observations.

2.1 Defining Response Characterisation

To begin, we outline the fundamentals of experimentation, along with the terminology that will be used, from the view of response characterisation.

2.1.1 Experiments and Observations

An experiment is the combination of actions to perform, resources to use and methods for obtaining observations. The actions, resources or observational methods can have adjustable parameters that control them. For example an action may be to mix a solution with an adjustable parameter of how long to mix for, or resources may have parameters to control the quantity, and an observational method such as spectroscopy may have a parameter to control the wavelength used. In practise many of the parameters will remain constant, for example through recognised and tested protocols. The parameters of interest are the ones that are changeable.

Throughout the thesis we will refer to an experiment as the set of these changeable parameters. For simplicity we will from now on refer to the changeable parameters as the parameters and ignore the constant parameters from our experiment definitions. Throughout the thesis an experiment with only a single parameter will be represented as x, whilst an experiment with several parameters will be referred to through the vector $\mathbf{x}$. The experiment parameters reside within a parameter space, which we will refer to as an experiment parameter space, with each parameter representing one-dimension of this space. In some cases, such as space exploration or theoretical problems, the range of each parameter may be infinite. However, in physical experimentation there will normally be limits on the parameters. For example if one parameter was the amount

of a soluble reactant to be placed in a liquid, then there will be a maximum amount of the soluble compound that can dissolve in the solution, after which saturation will occur and additional amounts of the solute will no alter the reaction, and so would not be worth performing. Additionally, the parameter values of possible physical experiments may not be continuous, instead they will likely be discretized by some level of precision that the experimental apparatus is capable of working with.

When an experiment is performed, an interaction with the system under investigation will be carried out and an observational measurement will be obtained. In this thesis we concentrate on one dimensional observations, which will be represented as the singular value y. Although it would be quite possible to have a multi-dimensional observation with different dimensions for each observation taken in the experiment.

When an experiment is performed it will interact with the system under investigation. In experimental characterisation, performing an experiment will yield some response, or observation. Within this framework, a system under investigation can be said to exhibit some behaviour. Where a behaviour could be thought of as a mapping between experiment values and observations. A behaviour may have different features, such as peaks, troughs or areas of linearity in the observational values across the experiment parameter space.

We can therefore consider an initial simple formulation for describing performing an experiment to investigate some system or behaviour f:

$$y = f(x) \tag{2.1}$$

However, physical experimentation is inherently noisy, where a physical experimenter would not expect to obtain the same observational value twice for the same experiment. This noise can be caused by several reasons, for example the interactions of many systems, such as the mixing and reaction of reactants in a solution, will often be too complex for the exact reaction to occur twice. Also instrumental inaccuracies will prevent the experiment parameters from being exactly what were requested and the observational measurements will have some error. These reasons can be abstractly considered as noise parameters on the observations and experiment parameters, that deviate the observations and experiments away from their true or requested values. These parameters are referred throughout as ϵ for noise on the observations and δ for noise on the experiment parameters, where the dimensionality of the experiment parameter will be the same as the noise parameter δ, although the notation will not be distinguished between.

Whilst ϵ and δ noise will be present in the majority of experiments, there is a further type of noise that may occur in only a few experiments but will create a greater divergence between the true response and the observation. This noise is caused by failures of the experiment, such as reactant contamination, which are undetectable and cause the experiment to provide observations that are unrepresentative of the true underlying

behaviour. This additional corruption of the observational values is represented by ϕ, and is not expected to have alter the responses of all experiments, unless there is systematic error altering all observations by a constant amount. In the situation where there was such systematic error, the results for the experiments would likely be discarded as they were too error prone.

By including these noise considerations, the function for obtaining an observation can be updated to:

$$y = f(x + \delta) + \epsilon + \phi \tag{2.2}$$

In the characterisation experiments considered here, each experiment is performed independently of other experiments using fresh batches of reactants. However, some experimental systems will be experiment order dependent, where performing one experiment will change the system under investigation, so perhaps altering the observations obtained from subsequent experiments. In such dependent systems, the above view of experiment characterisation could not be applied.

The philosophy of science view of a good experiment, is one that is able to provide new information and to differentiate between competing views of the behaviours being investigated (Franklin, 1981). In the next section we consider how views of the behaviours being investigated can be made through using hypotheses.

2.1.2 Hypotheses

A hypothesis represents one view about the system being investigated. For example, in response characterisation a hypothesis may provide a representation of the probable response curve. Alternatively in the case of erroneous observations, the hypothesis may hypothesise about the validity of observations by declaring which observations are valid and which are erroneous. Regardless of its purpose, a hypothesis should be able to provide a prediction about the system being investigated and it must also be falsifiable (Buck, 1975). The predictions of a hypothesis can be used to determine the accuracy of the hypothesis, where a hypothesis can evaluate itself against the experimental data available to determine a quality measure, or confidence (Gooding and Addis, 1999).

A hypothesis can never be proved (Buck, 1975), but with sufficient deviation between experimental observations and the predictions of a hypothesis, a hypothesis can be rejected. As such, information gain from experimentation comes through the falsification of hypotheses. To explore this, take for example a hypothesis that is currently well supported by the experimental evidence available. Obtaining further observations that agree with the hypothesis will increase the perceived confidence of the hypothesis, yet it will provide little information that was not already known. However if we take the same hypothesis and obtain repeated observations that significantly differ from the prediction

of the hypothesis, then new information has been collected that can be used to build a more representative hypothesis.

Philosophers of science argue that experimentation can benefit from contemplating multiple hypotheses simultaneously (Chamberlin, 1890; Platt, 1964). The use of multiple hypotheses allows for several different explanations of the data available to be considered and developed in parallel without prejudice. There are several reasons for the prescription of using multiple hypotheses, one of which being that human experimenters if only using a single hypothesis, can become attached to the hypothesis and become unwilling to let it be disproved (Chamberlin, 1890). A second reason, more relevant to the design of an artificial experimenter, is that with a single hypothesis other alternate, yet well performing views of the behaviours will be ignored, leading to experimentation that can be described as incomplete (Chamberlin, 1890). The use of multiple hypotheses also has a potential benefit for a computational system, as a computational system can contemplate many more hypotheses simultaneously than a human. Identified in the prescription of multiple hypotheses, a human experimenter can only verbally express a single line of thought at a time, making the conceptualisation of multiple hypotheses difficult (Chamberlin, 1890). Where as a computational system will have no such problem in considering many in parallel. By being able to consider many different hypotheses in parallel, perhaps several thousand or hundred thousand hypotheses, a computational system will be able to approach the discovery process in different ways to a human scientist, which may provide the computational systems with advantages or new modes for discovery (Giza, 2002).

In the following a hypothesis is represented as h. The prediction of a hypothesis for an experiment parameter is noted as $\hat{h}(x)$. A hypotheses falsifiability will occur when $\hat{h}(x) \neq f(x)$. The confidence of a hypothesis based on the available experimental information and prior knowledge is represented as a function $C(h)$.

2.1.3 Goal for Experimental Response Characterisation

To identify the goal for experimental characterisation, take an initial assumption that there are infinite time and infinite resources to explore the experiment parameter space. With enough observations, a reasonable goal would be to obtain a hypothesis with predictions equal to the true underlying behaviours exhibited by the system, written mathematically as:

$$\forall x, \hat{h}(x) \equiv f(x) \tag{2.3}$$

Notice that the underlying behaviours without the noise models on the independent and dependent variables is required, as we are not interesting in representing the noise model within the hypothesis, rather we want the hypothesis to filter out the noise. However in physical experimentation infinite time and resources, or at least resources that could

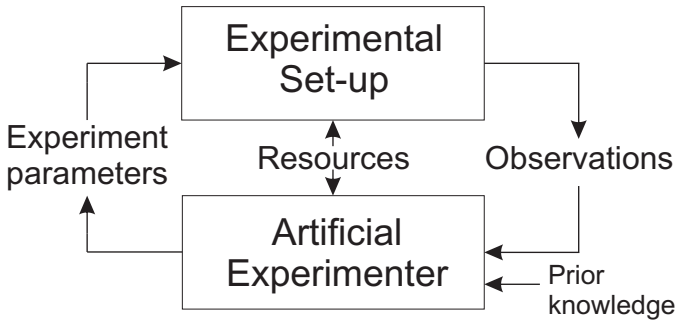


Figure 2.1: Abstract overview of experimentation. An artificial experimenter determines the experiments to perform, based on the observations from previously performed experiments and any available prior information about the domain being investigated. These requested experiments are performed by a experimental set-up, that can physically perform the experiments, returning an observation. Both the artificial experimenter and the experimental set-up are restricted by the resources available.

be used to exhaustively and accurately search the experiment parameter space, are not possible. Therefore resource usage becomes a limiting factor for experimentation, such that the goal is to obtain that hypothesis that matches the underlying behaviour with the minimal amount of resources used, which can be represented through the following error function for characterisation:

$$E(X) = \min \int \left(\hat{h}(x) - f(x)\right)^2 dx + R(X) \tag{2.4}$$

where X represents the set of experiments performed, $R(X)$ is a cost function for the resources used on those experiments, and the integral is across the range of experiment parameters that could be performed.

2.1.4 Abstract View for Experimental Characterisation

An abstracted view of experimentation is provided in figure 2.1, showing the relationship between the experimenter and the hardware device. A closed loop exists between the experimenter that contemplates the experimentation and the experimental hardware. Experiment parameters are passed to the experimental set-up and observations that have some uncertainty in their precision and accuracy are returned. These observations, along with any additional *a priori* knowledge, are used to determine the next experiment to perform. The limiting factor on this loop are the resources available.

The artificial experimenter is comprised of two components. One that manages the proposal and evaluation of the hypotheses, and the other component determines the experiments to perform. These components are referred to as the *hypothesis manager*

and the *experiment manager*. In the following sections existing techniques for automated discovery and automated experimental selection are reviewed.

2.2 Related Work

In this section we review computation scientific discovery style systems that have been applied to physical laboratory problems. In the following section we consider the machine learning perspective.

2.2.1 KEKADA

The KEKADA system, being one of the earliest examples of a computational experimentation system, worked to model the heuristics used by Hans Krebs to discover the urea cycle (Kulkarni and Simon, 1988). The aim was to see if a computational system could mimic and rediscover knowledge obtained in real experimentation. Key to the KEKADA approach was the investigation of surprising behaviours (Kulkarni and Simon, 1990), where a surprising behaviour is one where there is substantial difference between the observation for an experiment and the prediction of the result of that experiment. The KEKADA system utilised strategies inspired by successful human scientists for developing hypotheses and formulating experiments to test those hypotheses. Experiments were then carried out by a human experimenter, with the observations being fed back into the computer by an operator.

Some domain specific prior information was required, encoded in a machine readable form (Kulkarni and Simon, 1990). This knowledge allowed for hypotheses to be developed that would consider specific processes between independent and dependent variables. Additionally the prior domain information allowed the hypotheses to consider pathways of a reaction containing several sub-processes, or abstract hypotheses dealing with general concepts such as whether a particular reactant is a catalyst.

The technique reacted to obtaining a surprising observation, through five competing strategies that would try to learn more about the surprising observation (Kulkarni and Simon, 1990). One strategy looked to identify the independent parameter causing the surprise by systematically omitting one of the independent variables from repeat experiments to see if the surprising observation still occurred. Another strategy would look to magnify the surprise by modifying the independent variables. A third strategy considered if there was an error in a sub-process of the reaction, which led to the development of the surprising observation. A fourth strategy used the domain knowledge to find the scope of the surprise, for instance if an independent parameter was an amino acid, this strategy would consider that all amino acids would cause the surprise. The fifth strategy

considered similarities between the surprising observation and other existing behaviours, to see if the behaviour had been understood elsewhere.

If any of the above strategies provided an experiment that produced a surprising observation, the strategy was repeated to further understand that surprising observation. After investigating the observation further, the experimentation algorithm then looked again to find surprising behaviours.

The KEKADA system was shown to provide a good model of the heuristics used by Hans Krebs (Kulkarni and Simon, 1988). However, the work found that in comparison with human experimenters, KEKADA was limited by the number of heuristics it could employ, with human experimenters able to employ far more (Kulkarni and Simon, 1990). Improvements to the system were suggested through increasing the heuristics and increasing the domain information available, although the authors were cautious to not make the system too domain dependent (Kulkarni and Simon, 1990).

2.2.2 Experimentally Discovering Equations of Observable Behaviours

An early implementation of an autonomous experimentation system utilised the FAHRENHEIT discovery system to analyse observations, plan experiments and then perform those experiments (Żytkow et al., 1990; Żytkow and Zhu, 1991). Based in part on the BACON system (Langley et al., 1987), the FAHRENHEIT system increased the complexity of the behaviours that could be discovered and represented and was evaluated in laboratory scenarios (Żytkow and Zhu, 1991). One implementation of FAHRENHEIT used electrochemistry as the domain to work within, using differential pulse voltametry to measure concentration of ions (Żytkow et al., 1990). Sequences of experiments were chosen to perform, with the observations from those experiments being analysed before a new sequence of experiments were performed (Żytkow, 1997).

The goal of the FAHRENHEIT system was to find an empirical theory that could describe an observed behaviour within a parameter space, then explore the space to find the limits of that behaviour (Żytkow, 1997). The empirical theories looked for mappings between a single independent variable and a single dependent variable (Żytkow, 1997). To perform this, model fitters were developed using weighted least squares to provide polynomial fits to the data (Zembowicz and Żytkow, 1991). Weightings were applied in terms of the accuracy of the observations, such that more weighting was given to those observations that were believed to have a higher accuracy (Zembowicz and Żytkow, 1991). One of the key heuristics for the FAHRENHEIT system was to test the reproducibility of observations and in turn to identify the accuracy of the observations (Żytkow et al., 1990), making it one of the first implementations to actively consider experimental noise. However, repeatability and error was considered in terms of understanding the error related to a particular measurement method (Żytkow et al.,

1992). This meant that independent experimental errors, caused by random failings in the experiment, such as reactant failure, were not considered.

Whilst the FAHRENHEIT approach looked to model regularities in behaviour similar to BACON, the importance of finding points of special interest such as maximum and minimum points and areas of discontinuity were considered (Żytkow et al., 1990; Żytkow and Zhu, 1991). The technique would analyse the available data and identify boundaries between any regions of regular behaviour. Separate models would then be produced for the behaviours either side of the boundary (Żytkow, 1997). This is a process similar to local linear modelling techniques, such as LOLIMOT (Nelles, 2000), where producing separate local models, allowed for simpler individual models to be produced. It is also similar to the work by Hall and Molchanov (2003), who also consider boundary identification in regression by estimating and evaluating the boundary positions within in subsets of the parameter space. However these techniques require a large number of observations to learn from and are not designed to handle erroneous observations. Multiple hypotheses received some consideration in revised versions of FAHRENHEIT, by allowing different explanations of the regularities to be made by different hypotheses (Huang and Żytkow, 1997). Hypotheses were rejected if they did not agree with at least two thirds of the available data, to allow for some observations to be erroneous. The technique gave preference to simpler models when there were hypotheses that could model the behaviour similarly well. However this technique required a larger number of observations.

2.2.3 An Automated Chemistry Workstation

A project that looked at automating a microscale chemistry workstation, led to the design of a system that combined traditional experimental design techniques, such as factorial designs, with adaptive experiment selection protocols (Du et al., 1999a). The goal for the techniques developed was to first demonstrate fully autonomous experimentation could occur and second to develop a system capable of investigating a parameter space to obtain the best yields of product. In terms of the terminology set, we can describe maximum yield as finding the parameter x that maximises $f(x)$. The first goal was addressed by developing software capable of interacting with the automated hardware and managing resources through effective scheduling strategies (Corkan and Lindsey, 1992). Of more interest to the present thesis are the techniques used to address the second problem, of actively investigating a parameter space to achieve the desired goal. However, unlike other systems of discovery considered here that look to discover or understand behaviours, this system concentrated on finding the set of parameters that will give the single best output.

Having the requirement of discovering the optimal output from a particular parameter space, allowed the problem to be tackled through an evolutionary algorithm. As

part of the experiment selection strategy, the simplex evolutionary optimisation algorithm (Spendley et al., 1962), was employed to optimise the identification of the 'best' result through adapting the selection of experiments in response to the observations obtained (Du and Lindsey, 2002). The simplex method guides experimentation by moving away from the experiments where the 'worst' results were found, and moving towards the experiments where the 'best' results were found (Matsumoto et al., 2002a). The simplex algorithm was refined to allow for there to be parallel searches, allowing different areas of the parameter space to be investigated to try and avoid hitting local maxima (Matsumoto et al., 2002a; Du and Lindsey, 2002). Additionally, changes were made so that only the experiments that produced the best yields were used to evolve the next set of parameters (Du et al., 1999b). When the adaptive strategy was combined with the batch based factorial design strategy, it allowed for experiments to be performed that investigated the parameter space then went to further understand where the optimal yields could be obtained.

The techniques employed, addressed the wastage of experiment resources in several ways. If a factorial design was being used to generate the experiments, then there were monitors in place to stop experimentation if the observations obtained were of no interest (Kuo et al., 1999; Du et al., 1999c). Additionally solutions were developed to address the problem of the adaptive experiment selection algorithm wasting experiments, by using a two-tiered strategy that would first place experiments in a breadth-first manner to explore the parameter space, then explore at more depth only if a potentially interesting response was obtained (Matsumoto et al., 2002b).

Overall the techniques are well designed for exploring a parameter space where experimental error is low and the purpose is to find the single best response possible. The technique is not well suited when errors occur as it is likely the refinements made to the simplex algorithm would direct experiments away from an erroneous experiment that produced little yield, even if it would provide a high yield if performed successfully. As the simplex algorithm biases placing experiments in locations of the parameter space where a good observation was previously found, the technique is not well designed for discovering different features in a behaviour where different features may be remote from each other in the parameter space. To bypass this problem with evolutionary algorithms, a different approach called scouting used a dynamic utility for the evolutionary algorithm to search on, which was based on how interesting each observation was rather than how much product each experiment would yield (Pfaffmann and Zauner, 2001). This technique is discussed next.

2.2.4 Scouting

Enzyme reactivity characterisation has been considered before by an autonomous experimentation system, using a technique called scouting (Matsumaru et al., 2002, 2004).

The scouting algorithm (Pfaffmann and Zauner, 2001), is an evolutionary algorithm that focuses experiment placement in areas of the experiment parameter space where unexpected observations were previously obtained.

The technique utilised a simple single hypothesis system based upon a cubic equation, calculated within a subset of the parameter space. Predictions about the outcome of a particular observation were made by calculating a distance weighted average of the observations for the k-nearest neighbouring experiments that had been performed previously. This would make the technique susceptible to false predictions if an erroneous observation was present.

Unlike many evolutionary algorithms that may try to optimise to the best output, the scouting technique tied the fitness value to the amount of surprise each experiment obtained. Surprise was measured by how much the observations differed from the predicted value for that experiment, where if the prediction and the observation were similar then the experiment that obtained the observation was not surprising. This dynamic notion of surprise allowed the technique to identify different features of a behaviour it was investigating (Matsumaru et al., 2004). When an experiment was performed that first discovered a new feature not captured by the hypothesis, the observation for that hypothesis would be surprising. This would then promote further experiments to be performed in that region of the parameter space, allowing the feature to be more fully understood. Then when the feature had been fully characterised, the predictions would become similar to the observations, so experiments in that region would then become unsurprising. With the observations becoming unsurprising, the evolutionary algorithm would then automatically move away from that area of the parameter space to regions elsewhere in the parameter space where the observations obtained were more surprising. This technique would also be able to identify erroneous observations, where if an observation was erroneous and different to the hypothesis, further experiments would be performed in that region and would identify the observation was an outlier. However, due to the limited modelling technique that would consider all observations, the number of experiments required to discount an erroneous observation would likely be high.

Whilst this approach was able to be connected to automated laboratory hardware to conduct autonomous experimentation (Matsumaru et al., 2002), the technique had drawbacks. As the technique only considered a single hypothesis, any erroneous observations would impact future predictions and had no way of being ignored if those observations were selected to form the prediction. Additionally, the technique still required a large number of experiments to be performed, with 120 experiments in a 2-dimensional experiment parameter space being reported (Matsumaru et al., 2002).

2.2.5 Logical Inference Based Systems

The use of large collections of domain information has been central to several computation scientific discovery systems. The DENDRAL expert system was an expert system for discovery (Lindsay et al., 1993), which allowed for mechanistic hypotheses to be produced (Buchanan et al., 1969). Whilst the KEKADA system also promoted the use large amounts of domain information for the same reason (Kulkarni and Simon, 1990). The more recent approach by King et al. (2004) also uses a large amount of prior information, represented within a logical framework, combined with active learning techniques and a robotic platform, which choose and perform experiments that will help fill in the gaps of the information available.

The self-described *Robot Scientist* technique described in (Bryant et al., 2001; Whelan and King, 2004; King et al., 2004, 2009), performs discovery through abductive inference, where a body of known information is encapsulated within Prolog programs, which are used to make logical inferences through ASE-Progol. The system generates a number of hypotheses, which predict the open reading frames that code for an enzyme with particular reactants and products. The ASE-Progol is used to select the cheapest set of experiments required to disprove all but the true hypothesis, in the set of hypotheses considered. The technique is formed around the problem of finding the smallest decision tree (King et al., 2004).

The reported ASE-Progol approach is shown to out perform a random experiment selection strategy and a naive approach that selects the cheapest experiment to perform at each step. A comparison of the three strategies found that to provide a hypothesis with an accuracy of around 88% in simulation, the cost of resources of the ASE-Progol approach was reduced by five orders of magnitude in comparisson to a random search (Bryant et al., 2001). In actual performance, the difference in cost between the random and expected cost strategies is minimal until the accuracy of the hypotheses increases above 65%. After which, the benefit is still small, with £100 providing an accuracy of 74% for random as opposed to 77% for ASE-Progol (King et al., 2004). The benefit appears that the ASE-Progol approach is able to attain a higher accuracy than the random approach, which appears to be limited to 74%.

The use of large amounts of prior information can enable more mechanistic hypotheses to be considered. However, it is crucial the provided data is reliable and complete, otherwise the hypotheses the system develops will be unreliable. In its current form, the system does not evaluate the prior information it is given. Therefore all prior information is considered valid. Ensuring such accuracy, especially in the biological domain is difficult. In the later work (King et al., 2009), the authors reported problems with hypotheses being developed on incomplete prior knowledge, leading to gaps and errors within the hypotheses that could not be fixed or detected by their system automatically. A better approach to this technique would be to incorporate a system of belief in the

prior information, so that hypotheses can be developed that ignore some prior knowledge and question its validity. However, such a system would not likely perform well within the strict logical framework that is currently employed. Additionally, with such a large body of prior domain information being provided, evaluating the prior information itself is impractical due to resource constraints.

Additionally, this technique has further deficiencies caused by an early assumption, which prevents it from being used in a broader framework of experimentation. The assumption is made that the system will have access to a *true* hypothesis that represents what is being investigated. In the general case, a true hypothesis cannot be obtained and uncertainty about its validity will always exist. However, if the parameter space and the scope of the hypotheses are restricted, then it is possible to create a hypothesis mechanism where a finite number of potential hypotheses could occur. In the work described by King et al. (2004), the domain of information has been restricted to a heavily limited view of functional genomics, where a large number of facts about enzyme catalysis is available. As such, the amount of prior information will always need to be large, in order for hypotheses that provide a good representation to be formed. This means that any new information discovered by such a system will likely be dwarfed by the size of the data being provided. In following work ontologies of science are proposed that try to provide a massive body of information, which could be navigated by such an abductive logic based system (Soldatova et al., 2006; Soldatova and King, 2006). However, the authors have also declared that such ontologies, especially in biology, are often poorly maintained, meaning a new ontology would be required to be written before their use could be evaluated (Soldatova and King, 2005).

The issue of cost questions the viability and generality of this system. Removing the cost of the laboratory hardware, the system requires a large body of domain information, which will be expensive to provide, first experimentally, and secondly to provide it in a machine readable form. Secondly the system is evaluated with experiment cost being considered on a logarithmic scale (King et al., 2004). Therefore any claims of cost saving through using the automated system and ASE-Progol algorithms, are weakened by the large initial set up cost, which will have to be repeated if the computation system were to be retasked to work within a different domain. This initial cost will most likely prevent the techniques presented from being utilised in alternate domains.

2.2.6 Gaussian Process and Minimum Distance Based Automated Enzyme Assay

Another body of work investigated the automatic characterisation of enzymatic activity through autonomous experimentation (Bonowski et al., 2010). The technique combined off-the-shelf laboratory hardware, with a standard regression technique and a simple experiment selection algorithm. Using Gaussian process, a model of the response surface

is predicted from all observations. A batch of experiments are initially chosen to fill the parameter space being investigated, where there is a minimum distance requirement between experiment parameters. Subsequent batches place experiments where there is the greatest uncertainty in the model, subject to a minimum distance requirement between experiments, which reduces arbitrarily over time. The number of experiments used is large, with 96 experiments being used within a 2-dimensional parameter space.

The requirement for there to be a minimum distance between experiments is designed to ensure that experiments cover the parameter space. However, the minimum distance prevents experiments from being performed that evaluate the validity of the previous experiments. In biological experimentation, especially with fragile reactants like enzymes, erroneous observations may occur and experimental noise may be substantial. Therefore whilst there is a benefit in promoting exploration within the parameter space, there will be a drawback in preventing experiments that test the observations of previous experiments.

Whilst this technique is within the same domain of work that this thesis addresses, the actual technique fails to address the key problems in biological response characterisation. The work mentions the problem of cost and large parameter spaces. However, the cost issue is only addressed through automating the data acquisition stage. No consideration is given to reducing the number of experiments required to perform the characterisation, nor is there any consideration for errors in the experimentation, outside of the Gaussian noise that can be handled automatically by the Gaussian process. Finally the experiment selection technique appears to be little more than random placement, with an exception that experiments will be placed apart from each other in the parameter space. Whilst the authors claim their technique performs better than random search, the technique provides little new insight into discovery methods. The process of reducing the minimum required distance between experiments is similar to the grid search technique used in the automated chemistry workstation project, where the experiments would be chosen from a grid that would become finer over time (Dixon et al., 2002).

2.3 Experimental Design and Active Learning

Design of experiments, or experimental design, is a statistical technique for selecting a set of data that will provide information to build models or hypotheses with some defined mathematical guarantees (Fisher, 1935; Box and Draper, 1987; Box et al., 2005). For example, an optimal design may allow for a hypothesis to be developed without bias. Such experimental design has been used extensively within response characterisation (Myers et al., 2009). Experimental designs are often closely tied to linear systems, with the covariance matrix of the linear system being used to represent response, also forming part of the experiment design protocol.

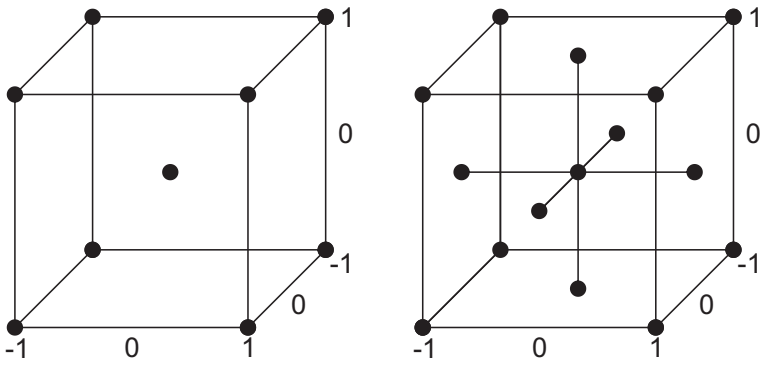


Figure 2.2: Diagram of placement of experiments in a factorial design for 3 parameters. Experiments are placed at the position of dots, where the parameter space is coded between -1 and 1 to represent the minimum and maximum values available. In (a) a cube design with centre point is shown, where experiments are placed at the extreme values permitted. In (b) a star design is shown, where experiments are also placed in the centre of the parameter values permitted.

Experimental design techniques will often apply particular designs for the placement of experiments to be used. For example, a factorial design will place experiments within the parameter space to test all factors, or experiment parameter dimensions (Box and Draper, 1987; Myers et al., 2009). An example of a factorial design is given in Figure 2.2. The number of unique experiment parameter combinations will be small, often only the combinations of the maximum parameter values and sometimes also the middle value of those parameters. But the number of repeat experiments may be high. Some designs become inefficient as the dimensionality of the parameter space increases, as the designs strive to provide the required statistical guarantees.

Whilst experiment design techniques may be classed as optimal for some particular purpose, many lack the ability to adapt with the observations obtained, meaning that resources can be wasted performing experiments that are providing no new information. In many experimental design packages, the level of factorial design required will be selected along with the number of experiments allowed. The package will then return a set of optimal experiments that should be performed in batch. Further experiments can be requested after those experiments have been performed, but the choice of the next set of experiments will only be based on the independent parameters, or experiment parameters, and not the dependent parameters, the observations. Sequential design of experiments is a sub-field of experiment design that tries to address part of this problem (Chernoff, 1959). Sequential design of experiments techniques will produce a set of experiments to be performed, whilst also providing a method of monitoring the observations returned to determine when to stop performing experiments (Wald, 1947). If observations appear to stop providing new information, then experimentation is ended. Whilst this attempts to reduce resource wastage, it does not consider the situation where

only part of the design space is returning uninteresting observations. Neither does it identify regions providing interesting observations and focus experimentation there.

Whilst experimental design has benefits of providing statistical guarantees, it is inefficient in resource usage and unable to react to the observations obtained. A newer field of active learning attempts to maintain a framework for statistical guarantees, whilst also addressing the issue of using the observations obtained to adapt the choice of experiments. In the next section we consider active learning.

2.3.1 Active Learning

Active learning is a closed-loop approach where the learner sequentially decides about the selection of data to be added to the training set (Thrun, 1995; Cohn et al., 1996; Settles, 2009). Opposed to traditional design of experiments or passive learning, where the selection of the training set is made independent of the observations or labels obtained, active learning has been shown to provide significant performance improvements (Castro et al., 2005). Active learning techniques often discuss learning as a classification problem, where the learner is able to obtain the labels for specific instances, but obtaining those labels is expensive and learning should be achieved with the minimum cost. Unlike computational scientific discovery style approaches, which are often application focused, active learning is based within mathematical frameworks, which can allow for the formulation of mathematical proofs and guarantees about the performance of the developed techniques. This is similar to the more established field of experimental design, which provides techniques for choosing experiments that will fulfil different optimality requirements. It is interesting that experimental design methods often reside within a regression framework, often using linear regression as a base for the techniques, whilst active learning often resides in classification. However the use of classification in active learning and regression in experimental design is largely unimportant, as the methods used in both can be considered in more general problems.

With active learning taking a more mathematical view of discovery, some of the techniques developed overlook important issues in experimental scenarios. For instance, the accuracy of labels is often guaranteed or the learning problem is one of binary classification (Seung et al., 1992; Settles, 2009) or where only binary responses are returned to learn from (Kulkarni et al., 1993). Whilst some experimental design techniques have also suffered when responses are noisy (Atkinson and Fedorov, 1975a). However active learning techniques have been successfully applied to physical problems, such as drug discovery (Warmuth et al., 2003), but also reside within other commonly considered machine learning problems, such as text classification (Lewis and Gale, 1994b) or natural language processing (Settles and Craven, 2008). Of those active learning techniques considering regression, there are requirements upon them not suitable for the current

problem, such as knowing the model class (Sung and Niyogi, 1995), and no consideration for erroneous observations (Castro et al., 2005; Sugiyama, 2006; Burbidge et al., 2007).

In the following section, active learning and experimental design techniques will be considered, with a particular focus on those techniques that could be applied to characterisation based experimentation.

2.3.2 Query by Committee

In query by committee, learning is considered through building an ensemble of different hypotheses (Seung et al., 1992; Freund et al., 1997). Experiment selection uses the entire set of hypotheses, by choosing the the experiment where the prediction of the hypotheses are in maximal disagreement (Seung et al., 1992). In classification the disagreement can be identified by having all hypotheses vote on the predicted label to be applied to the proposed experiment, where the experiment is chosen where there is the most disagreement in the predicted labels. Query by committee is of interest to computational scientific discovery style machines, as it closely resembles the philosophy of science view that multiple hypotheses should be considered in parallel (Chamberlin, 1890). However, the technique is currently often applied in situations with little experimental equivalent, such as binary classification with no noise on the observations (Seung et al., 1992).

A problem with the technique is there is no agreed upon methods for proposing the hypotheses that form the set within the literature (Settles, 2009). The disagreement exists within the three components that make up a query by committee approach. First an ensemble of hypotheses must be constructed, second a method for separating hypotheses is required, and finally a method for representing the set of hypotheses to a human or other machine learning interface, which may be unable to understand a set of competing ideas.

2.3.2.1 Separation of Regression Hypotheses

With an ensemble of hypotheses, experiments need to be used to identify which of the hypotheses are the best representations of the behaviours under investigation. An ensemble of hypotheses therefore, allows for some simpler methods of active experiment selection. For example placing experiments where two models differ (Atkinson and Fedorov, 1975b; Atlas et al., 1989), or performing strategies for separating sets of competing hypotheses (Atkinson and Fedorov, 1975a; Sugiyama and Rubens, 2008).

The separation of sets of hypotheses has been considered in experimental design literature through the concept of T-optimality (Atkinson and Fedorov, 1975a,b). Here the experiment that produces the most equal split of agreeing and disagreeing hypotheses, is chosen as the experiment to perform. However, this technique relies on there being

a hypothesis in the set under consideration that provides a good representation of the underlying phenomena. Additionally, the authors state that the technique can struggle when there is noise on the observations, or if the hypotheses are similar to each other (Atkinson and Fedorov, 1975a).

Alternatively, a variance based approach has been used to separate sets of hypotheses (Burbidge et al., 2007). In this method, experiments are placed where the variance of the predictions of the hypotheses is greatest. This technique removes the need for biasing experiment selection based on what is currently believed to be the most confident hypothesis, which could be incorrect, especially in the early stages of experimentation where there are few observations available. However, this approach can be biased by hypotheses with outlying predictions, which can lead to situations where experiments are chosen that in the worst case will only discriminate against one hypothesis in a set of competing hypotheses (Lovell et al., 2010a).

Outside of techniques developed specifically for regression based hypotheses, active learning considers many techniques that have been developed within a classification framework. Such techniques are reconsidered within a regression framework, along with a further discussion of separating regression based hypotheses in Chapter 4.

2.3.3 Minimising Variance

Learning by choosing experiments where the variance of model predictions is largest, has been considered in several scenarios (Atlas et al., 1989; Krogh and Vedelsby, 1995; Cohn et al., 1996; RayChaudhuri and Hamey, 1995; Burbidge et al., 2007). The idea of this technique is to reduce the variance of model predictors, to allow for a more accurate model to be produced. However in some cases the number of models compared have been small, for example only two models trained on the same data are considered by Atlas et al. (1989).

Variance reduction has been considered in active learning with an ensemble of regression models formed through query-by-committee (Burbidge et al., 2007). In this example, a committee of five different regression models is maintained, trained by using different subsets of the available data, so as to have different views of the parameter space. The models are then examined using a set of possible unlabelled examples, where the example with the highest variance in predictions by the regression models is chosen to be the example to obtain a label for. The benefits of this technique are claimed to be to reduce overfitting through minimising the variance in the learner (Burbidge et al., 2007). However, there were several problems with this technique. First the small number of models in consideration will have led to a poor range of models being proposed. Second a large number of experiments were required, with several hundred observations in a 1-parameter problem considered.

Whilst the variance strategy appears a sensible method for placing experiments where models disagree the most, so as to allow for a more accurate representation of the behaviour to be developed, the strategy is not robust and can lead to poor performance under some circumstances (Lovell et al., 2010a). In Chapter 4 we present situations where the variance strategy fails and provide a more robust alternative method for active learning.

2.3.4 Confidence Maps and Uncertainty Sampling

The design for a competence map (Thrun and Möller, 1992; Thrun, 1995), begins with the assumption that the most useful information can be gained by performing experiments where the predicted response is most different to the true response:

$$e = \max_{\mathbf{x} \in \mathbf{X}} \left(f(x) - \hat{f}(x) \right)^2 \tag{2.5}$$

As the true response model $f(x)$ is not known, the competence map tries to estimate the difference between the true and predicted responses across the parameter space. Experiments are then chosen where the competence map believes the difference between the predicted and actual to be greatest. The technique is similar to choosing experiments where the error bar of the learner is greatest, which should also provide a large amount of information (MacKay, 1992), or uncertainty sampling where experiments are performed in regions of the parameter space where the hypotheses are most unsure (Lewis and Gale, 1994a).

However the problem with the technique is that two levels of modelling are required, first the representation of the hypothesis and then the representation how well the model is performing. Without sufficiently large amounts of data that can be used to verify the competence map, the prediction of model accuracy will be weak and likely unuseful. At best a technique like monitoring the size of the error bars could be employed when only small amounts of data is available. However those error bars will be artificially large at the edges of the space being explored and will also be large in areas where no data has been obtained. For experimentation with limited resources accurately predicting where the model is weak will most likely not be possible. So a competence map style approach would not be suitable, although having an awareness of the uncertainty of a particular region of the parameter space would provide useful information that could be used in part of an experiment selection strategy.

2.3.5 Investigating Unclassified Observations

An observation y can be tested against a hypothesis, to see if for some level of error the observation is consistent with the prediction of the hypothesis. If none of the hypotheses

in the set of hypotheses under consideration predict y within some minimum acceptable level of error, then that observation can be determined as being unclassified by the hypotheses. That is to say the observation is not consistent with any of the hypotheses. There could be two reasons for this happening, firstly that y belongs to a behaviour not modelled by any of the hypotheses, or that the observation is incorrect. This issue provides a measure of uncertainty (Cohn et al., 1994), although the cited work only considers the case of the hypotheses being incorrect. Performing experiments near where such an unclassified observation has been identified, information gain can be expected from updating the hypotheses (Cohn et al., 1994), or by identifying the observation as potentially erroneous.

2.3.6 Order of Experiment Selection

The order experiments are performed in may change the state of the system being investigated, meaning that it behaves differently. Such learning can either be order sensitive, where the order that actions are performed will change the state of the phenomenon being investigated, or order free, where actions do not change the phenomenon (Thrun, 1995). In experimentation this will depend on the experimental set-up and the domain of experiments being performed. The KEKADA approach for instance tries to learn reaction pathways, leading to the dependent variables of one experiment being used as an independent variable in another (Kulkarni and Simon, 1990), which will be order dependent. Whilst in the scouting approach, each experiment samples from the experimental parameter space and returns an observation, where experiments are independent of each other (Matsumaru et al., 2004), therefore being order free. An order dependent system will require additional prior information, otherwise some experimental steps taken may cause the system to change and observations to be unrepresentative of the experiment intended.

2.4 Role of Exploration and Exploitation in Discovery

Scientific discovery occurs through the disproof of hypotheses. Some discoveries will bring about paradigm shifts in the field of research (Kuhn, 1962), for example the discovery the world was not flat, or the discovery of DNA. However, most discoveries however will be smaller, bringing a small amount of additional knowledge. Often the larger discoveries are made by accident. Where as smaller discoveries are often built upon a larger foundation of background knowledge. Whilst these are grander thoughts about scientific research, they can be translated to a fit within smaller subsets of experimentation, where a parameter space can either be explored to find new behaviours not yet considered, or investigated through exploiting the available data and current hypotheses to strengthen the understanding. The exploration vs. exploitation problem is

then the decision about when and how much investment should be made into exploring the system, which may yield new discoveries but at high risk, with respect to resources being spent on exploitation to strengthen the hypotheses made (March, 1991).

Successful scientific experimentation requires a suitable management of the exploration vs. exploitation problem. A trade-off is required between performing experiments that explore the parameter space to search for new features or behaviours not yet known about, with performing experiments that exploit the knowledge currently available to refine the hypotheses into more accurate representations of what they are representing. Performing only one of these may result in a failure to provide a useful body of information. Too much exploration will reduce the amount of resources available to analyse and refine hypotheses. Where as too much exploitation may lead to other more interesting features of the behaviour being missed, as fewer resources would be available to explore the parameter space and discover those other features.

The exploration vs. exploitation trade-off, although not always specifically mentioned, has been addressed by many of the computational scientific discovery systems in the literature. The KEKADA system searched the parameter space until a surprising observation was discovered, at which point the strategies changed to investigate the behaviour near that observation, exploiting the information held within its hypotheses and experiment selection strategies (Kulkarni and Simon, 1990). The FAHRENHEIT system searched the space, then when a irregularity was discovered, for example a discontinuity, the extent of that discontinuity was investigated (Żytkow, 1997). The automated chemistry workstation project employed as part of a its strategy a grid search (Dixon et al., 2002), where experiments were placed spread out across the parameter space to provide exploration, with successive experiments reducing width of the grid squares and eventually experiments following the simplex based experiment selection that exploited the information known about the system (Du and Lindsey, 2002). The scouting algorithm continually addressed the problem, where it would explore the parameter space until an unexpected behaviour was found, at which point it exploited the information it had to perform further experiments that characterised the behaviour, switching back to exploration when the behaviour was no longer unexpected (Pfaffmann and Zauner, 2001). The robot scientist approach however, is purely an exploitation technique once experimentation has occurred, with all experiments being chosen to separate the hypotheses (King et al., 2004), and any exploration only occurring by the user that determined the prior knowledge to provide that would in turn shape where in the parameter space the system would explore. The approach by Bonowski et al. (2010) is initially explorative through experiments being placed through a space fitting algorithm, with later experiments being both exploitative through having a preference for experiments to be placed where the uncertainty is greatest, but also explorative through requiring experiments to fulfil some minimum distance requirement.

The exploration vs. exploitation problem exists within several fields of machine learning research as well, which will be discussed in the following sections.

2.4.1 Multi-armed Bandit Problems

The multi-armed bandit problem, is a toy problem built for investigating the problem of sequential design of experiments (Robbins, 1952; Berry and Fristedt, 1985; Auer et al., 1998). The problem consists of there being a number of levers that can be pulled, each returning some form of reward. In some cases the reward that can be obtained by pulling the lever is known, and in other cases it is not known. The goal is to achieve as much reward as possible with a certain number of pulls. This problem lends itself to investigating the exploration–exploitation trade-off. By exploiting the information available, levers with known reward can be pulled to obtain their reward. Whilst exploring the system by pulling the levers with unknown reward, may result in obtaining a reward greater than those known about. Various solutions have been proposed to manage the exploration vs. exploitation in this problem.

Whilst there is a large body of work investigating the exploration–exploitation trade-off in multi-armed bandit problems, we must be aware of the differences between multi-armed bandit problems and experimentation, which may prevent some techniques being employed. In multi-armed bandit problems, levers are often pulled multiple times to build a model of the potential reward. This issue has several problems. First there is the problem of cost and limited resources, where few multi-armed bandit scenarios take the cost of pulling the lever, or performing an experiment, into consideration (Tran-Thanh et al., 2010). Second there is the problem that in experimentation, performing the same experiment several times will not give an indication of reward, but rather an indication of the true underlying behaviour being represented at that part of the parameter space. Therefore in the following, only general concepts of dealing with the exploration–exploitation trade-off will be considered.

2.4.2 Random Transition

One technique for managing the exploration vs. exploitation trade-off is to randomly choose one them. Often there is a weighting applied to the decision, so that there is a bias to either exploration or exploitation, with the weight being defined in the literature as ϵ. Sometimes the bias between exploration and exploitation is predefined and constant throughout, for example the ϵ-greedy strategy, where most of the time exploitation will occur (Sutton and Barto, 1998). Alternatively the bias can be made adaptive, for example reducing the likelihood of exploration occurring over time, where the intention is to spend more resources in the later stages maximising reward through exploitation (Kaelbling et al., 1996). Other techniques adapt the bias based on the

performance of the previous experiment (Osugi et al., 2005; Tokic, 2010). Another technique known as an ϵ-first strategy, will perform a predefined number of exploratory experiments first, with all subsequent observations being exploitation experiments (Tran-Thanh et al., 2010). These techniques, whilst demonstrating advancements over simply only exploring or only exploiting, may benefit from taking into further consideration the data obtained.

2.4.3 Combining Exploration and Exploitation Scores

One method to address the exploration-exploitation trade-off is to first determine how well a proposed experiment will explore and exploit the space through some scoring metric for each. With this information the decision about where to perform the next experiment may be based on some function of these scores, where the best experiment would be able to explore and exploit the parameter space. This function may be a linear combination of an an experiment's ability to explore and exploit (Thrun, 1992, 1995), for example:

$$\text{overall-score}(x) = \text{exploration-score}(x) + \text{exploitation-score}(x)$$

where there is some appropriate metric for determining how well the experiment would explore or exploit the parameter space, for example distance from previously performed experiments and uncertainty in the model at that part of the parameter space, and the experiment is chosen that maximises the function. This function of exploration and exploitation may be weighted by a constant factor (Lovell et al., 2010c) or dynamic (Thrun and Möller, 1992; Cebron and Berthold, 2009), in terms of preference of exploration or exploitation. Although the methods for swapping preference between exploration and exploitation are often arbitrarily chosen. However, such a linear combination of exploration and exploitation scores may cause choices to be made that neither explore or exploit, meaning that dynamic systems that alter between exploration or exploitation specific experiments could provide a better alternative (Thrun, 1992).

2.4.4 Confidence bounds

The confidence bounds of a learner have been successfully used to automatically handle the exploration–exploitation trade-off, using the multi-armed bandit problem as a domain (Agrawal, 1995; Auer, 2002). The technique uses two parameters, which can be learnt from previous lever pulls. The first parameter is the expected reward for each lever, which can be learnt by successive pulls to a lever to build a model of the reward distribution. The second parameter is a measure of uncertainty in each prediction of expected reward. This second parameter is chosen so that the true reward sits somewhere

in the range of the expected reward, plus or minus the uncertainty. At each iteration, the lever with the highest sum of the two parameters is pulled.

Exploration and exploitation occurs automatically, through the two parameters changing over time (Auer, 2002). When learning begins, the uncertainty of any expected rewards will be large and will be the dominant factor in choosing levers to pull. When the uncertainty of the prediction for a lever is large, then by pulling that lever an exploration experiment will occur, as the learner does not know the reward they will receive. Over time, the expected rewards will become known and the uncertainty about those predictions will decrease. This means that the measure of uncertainty will become less dominant in determining which lever to pull, replaced with the expected reward becoming more dominant. When the expected reward is large for a lever, then by pulling that lever an exploitation experiment will occur, as the learner knows the reward they will likely receive. Therefore, when the system begins the experiments will be predominantly explorative, becoming less over time until they are mostly exploitative.

The main problem with this technique is the large number of experiments required to learn the reliability of each expected reward. With erroneous observations, predicting the outcome and noise for experiment parameters with this technique could become very inefficient.

2.4.5 Variance and Undersampling Regions

Some multi-armed bandit strategies, like the confidence bounds strategy mentioned previously (Auer, 2002), require an estimate of the error in the prediction term. Other active learning scenarios, like the one presented in (Burbidge et al., 2007), use the disagreement between an ensemble of predictions. Both of these cases require a variance or variance like term, to measure uncertainty within the learner. The active learning strategies then use this variance term in some manner to choose experiments that exploit the information available, usually through placement of experiments where the term is maximal. However if the variance term is underestimated, through undersampling the region of the parameter space it is predicting, then exploitation experiments may be sub-optimal in obtaining the maximum reward.

Antos et al. (2008) consider this problem of undersampling and propose adding information about the sampling that has occurred throughout the parameter space to guide exploration and exploitation decisions. The technique has a preference for performing experiments where the uncertainty, or variance, is maximal. But will perform exploration experiments in regions that have been undersampled.

Investigating undersampled regions of the parameter space is important, as it ensures exploration occurs. However the technique employed in (Antos et al., 2008), first raises the problem of deciding what qualifies a region being undersampled. The work proposes

an arbitrary proportional requirement in the number of pulls a lever has received, compared to the total number of level pulls performed. When the proportion requirement is not fulfilled, the lever is deemed undersampled and it is pulled. However the technique used for this is also designed to address another problem, where the error may not only be incorrectly specified due to undersampling, but also incorrect due to changing noise on the dependent parameter. This means that over time all levers are re-examined, to validate the predictions made are still accurate. In order for such a technique to be beneficial, it appears that a large number of experiments would be required, to allow the continued evaluation to occur.

2.4.6 Two Stage Exploration–Exploitation

Castro et al. (2005) discuss the benefits of active learning over passive sampling, for increasing the learning rate for number of experiments used. The active learning technique they employ is a two-stage strategy, with rigid exploration and exploitation phases. The first stage is exploration, with half of the experiments being placed uniformly across the parameter space. After this stage an initial model of the behaviour being investigated is constructed. From this model, interesting regions are identified, for example discontinuities or the boundary between two distinct piecewise regions. The second stage uses the remaining experiments to exploit the information in the model, by splitting experiments evenly between all interesting regions. The experiments allocated to each region are placed uniformly across the region of interest in batch. After this stage of experimentation, a second model is produced, with higher resolution than the first.

This strategy is shown to outperform passive strategies, where experiments are placed randomly (Castro et al., 2005). However, it appears that the technique does not fully realise the potential for active learning. Whilst active learning does occur, through the selection of experiments in the second phase that are dependent on the observations obtained during the first phase, this second set of experiments is still chosen and performed in batch. Therefore, if any errors occurred during the first phase of experimentation, or if the second phase subsequently discovers new more interesting behaviours, the strategy will not be able to adapt and save resources in the first instance, or allow extra resources for investigating the behaviours of the second. This means the technique employed by Castro et al. (2005), appears to fit poorly within active learning from noisy and unreliable observations.

2.5 Discussion

Computational scientific discovery began by trying to understand and mimic the process of a human experimenter (Kulkarni and Simon, 1988). Many of the approaches

assume abundant data sources, either in terms of experimentally obtained data (Kulkarni and Simon, 1990; Żytkow et al., 1990) or in terms of background prior information available (King et al., 2004). Expert systems have been developed for scientific inference (Lindsay et al., 1993), and proposals have been made that ontologies of science should be produced to aid computational scientific discovery (Soldatova and King, 2006; Soldatova et al., 2006; Bundy, 2008), even though existing biological ontologies have been criticised for their poor maintenance (Soldatova and King, 2005). Using large amounts of prior information has benefits, it allows for mechanistic hypotheses to be produced and can prevent resources being spent on rediscovering known phenomena. However, relying on large amounts of prior information has potential problems. The first is that if the discovery systems is overly guided by the prior information, the hypotheses developed will be restricted and will fail to explore outside of the given preconceptions. If the prior knowledge is wrong, then hypotheses developed will also have errors, which was demonstrated by hypotheses being developed by an autonomous experimentation that were wrong due to an incompleteness in the prior information (King et al., 2009).

A lot of physical experimentation however, is resource limited and in some circumstances background information does not exist. Take for example Mars exploration rovers that need to detect interesting features then perform experiments using a minimal amount of resources (Castano et al., 2007). An argument has been made that one of the next steps for computational discovery is to consider situations where very small amounts of experimental data exist (Langley, 2002). Whilst another argument is that techniques should include general purpose concepts (Lindsey, 1992). In designing such systems, we can draw on pattern recognition abilities of machine learning systems, whilst also considering heuristics performed by human experimenters.

Of particular relevance to autonomous experimentation, is active learning. Active learning and autonomous experimentation share many parallels. Both can be implemented as closed-loop approaches that have to select from where the next data point is obtained from. However many active learning techniques are more focussed on achieving mathematical guarantees, rather than addressing problems faced in experimentation, such as very limited resources and erroneous observations. Even systems designed for interacting with real world environments have ignored noise (Shen, 1994).

Active learning considers a trade-off between exploration and exploitation. Of the previous examples of autonomous experimentation systems given in Section 2.2, the KEKADA, FAHRENHEIT and scouting approaches address this issue. KEKADA explores the parameter space, then when a surprising observation is found it looks to investigate that surprise (Kulkarni and Simon, 1990). When the FAHRENHEIT system finds behavioural regularities in the parameter space, it seeks to find the scope of those regularities (Żytkow et al., 1990). The scouting approach has a dynamic measure of surprise, which means that when a surprising observation is found, experimentation looks to understand this surprise and once enough information has been found to make the

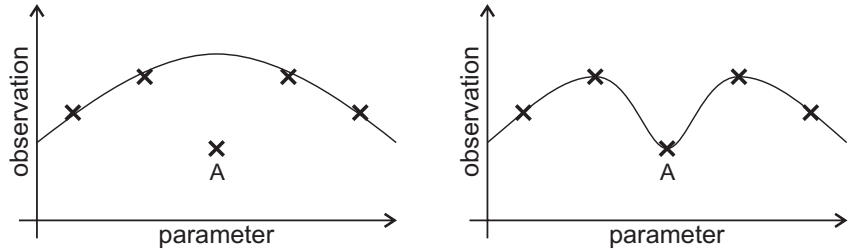
observation no longer surprising, experimentation continues to explore the parameter space (Pfaffmann and Zauner, 2001; Matsumaru et al., 2002). Any future systems must also consider this trade-off.

Active learning and autonomous experimentation can benefit from each other. Active learning provides the foundations for efficient data selection, combined with pattern recognition approaches in machine learning for data analysis, whilst autonomous experimentation provides an important problem to be addressed. This idea of automating the discovery process requiring ideas to be used from several disciplines is not new (Żytkow, 2000), however it still occurs that the fields try to remain separate.

Chapter 3

Managing Multiple Hypotheses

A requirement for an artificial experimenter is the ability to derive models or hypotheses from the experimental observations obtained. These representations of the data, which will be referred to as hypotheses, need to be accurate representations of the underlying behaviours under investigation. Additionally, a successful artificial experimenter will achieve accurate hypotheses with as few experiments as possible, so as to reduce the resources spent. However, biological experimentation complicates this problem by providing observations with noisy responses, along with erroneous observations that are not representative of the behaviours being studied. Whilst the validity of all observations could be determined through repeat experiments, doing so will reduce the resources available for investigating and identifying uncharacterised behaviours. Therefore a hypothesis manager should employ computational methods to handle such uncertainty, built with the view that computation is cheap compared to the cost of experimentation, meaning that computational complexity is unimportant, so long as a solution is feasible. Many previous hypothesis management techniques in automated scientific discovery have been more mechanistic in nature, utilising a large body of domain information to produce often an exhaustive set of mechanistic hypotheses within a restricted domain (Valdés-Pérez, 1994; King et al., 2004). However as such large bodies of prior information are not available in the present problem and mechanistic hypotheses are not required, we instead base the hypothesis manager on the development of regression models of the data. In the following, methods for developing regression based hypotheses capable of characterising response behaviours and handling the above described situations are discussed. Throughout a hypothesis is based around a regression function, as will be discussed later.



(a) (b)

Figure 3.1: Illustration of observation validity problem. The majority of observations (crosses) appear to form an arc. However observation A fails to follow this trend. This could be either because A is erroneous and a true outlier as shown in (a), or it could be that A is from a feature of the behaviour not yet characterised making (b) a better view of the data.

3.1 Uncertainty in the Observations

Determining the validity of the observations obtained, is a key problem for a hypothesis manager. In physical experimentation, no observation will be the exact value of the true underlying behaviour. For example experimental errors, imprecise measurement values and differences in the physical reaction, will all have an impact on the value of the observation obtained. This means that running the same experiment twice will give different observation values. In most cases this impact is small and can be comparable to additive Gaussian noise on the observations, or even on the experiment parameters themselves due to imprecise measurements meaning the actual experiment performed was not exactly the one requested. Such small fluctuations can normally be handled by suitable regression techniques, like the smoothing spline or Gaussian process. However in physical experimentation, and in particular biochemical experimentation where the reactants are fragile, additional uncertainty comes through the problem of erroneous observations.

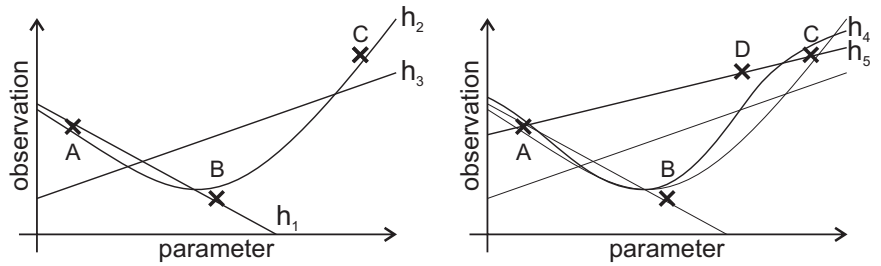
Erroneous observations add a different type of noise to the problem. First unlike the experimental noise discussed previously, it is not present in all experiments. Second the difference between the true underlying behaviour and the actual observation obtained may be much larger. As such, erroneous observations can be considered as shock noise that provides an observation unrepresentative of the true underlying behaviour. Essentially an erroneous observation can be thought of as an outlying data point. However, with only few data points available it is not possible to accurately determine whether an observation that appears to disagree with the current data points and hypotheses is in fact erroneous. Rather it could be that the current view of the data is incorrect and that the observation itself is a valid representation of the true underlying behaviour being investigated.

To illustrate the problem of observation validity, consider the case in Figure 3.1. Here there are five data points, of which the four unlabelled observations appear to form a smooth arc. However, the fifth observation, labelled A, disagrees with this appearance. As erroneous observations are expected from physical experimentation, it could be that this observation is a true outlier, and as such can be ignored. However, it could also be that the apparent outlying observation is actually from a feature of the behaviour not yet characterised. With only a few data points available, no decision can realistically be made about the validity of such observations with any degree of accuracy. Confirmation of observation validity can only be made after further experimental evidence is provided. Therefore we consider an approach where the decisions about observation validity is postponed until further experimental evidence is available. This can be achieved through utilising a multiple hypotheses approach.

3.2 Multiple Hypotheses

The idea of considering a set of different hypotheses in parallel, each with a different view about the behaviour being investigated, is a technique employed by successful experimenters and promoted in philosophy of science literature (Chamberlin, 1890). If managed well, multiple hypotheses allow for different views of the data to be considered and can prevent ideas from being overlooked (Chamberlin, 1890).

In machine learning the same idea exists within ensemble methods and in particular query by committee (Seung et al., 1992; Freund et al., 1997). Here the learning occurs across the committee members, or hypotheses, which can be used to determine experiments to be performed. However, there are no agreed upon general methods for creating the hypotheses within these committees (Settles, 2009). Some previous approaches in the literature produce hypotheses from random subsets of the observations available (Freund et al., 1997; Abe and Mamitsuka, 1998). However a purely random technique of hypothesis proposal will ignore any information available from the data that could be used to determine observation accuracy. Therefore a more principled approach, motivated from the problem and influenced by methods employed by successful experimenters, is considered here. The technique presented is similar to boosting (Shapire, 1990) and bagging (Breiman, 1996), in terms of wanting to increase the diversity in the hypotheses to allow for different views of the data. However as not enough experimental information will be available to classify the validity of all observations with any degree of confidence, the more principled element of the new technique is to ensure the creation of hypotheses that actively question the validity of observations. The validity of hypotheses is examined by having different hypotheses being considered in parallel, which have contrasting views of potentially erroneous observations.



(a) (b)

Figure 3.2: Illustration of using multiple hypotheses to address observation validity problem. Initially 3 observations (crosses) are available. Hypotheses (lines) are proposed with differing views about the validity of the observations. In (a), h_1 questions the validity of C, whilst h_2 and h_3 consider all observations valid but assume different experimental noise levels. Alternate hypotheses questioning validity of A and B could be proposed, but not shown here. In (b), the observation D is obtained which appears to confirm the validity of B and reject h_1. New hypothesis h_4 considers all hypotheses valid, whilst h_5 declares observation B to be erroneous.

To illustrate this, consider the situation presented in Figure 3.2, where observations are labelled alphabetically in the order obtained. After the first two observations are obtained, hypothesis h_1 appears as a reasonable hypothesis. On obtaining observation C however, a potential flaw in this hypothesis is found, suggesting that the hypothesis is erroneous, or with the expectation of erroneous observations, the observation itself could be erroneous. Continuing with the acquisition of observation D, the validity of observation C is now more likely, however observation B is now of questionable validity. As can be seen, maintaining a set of different possible hypotheses helps address this problem, where the hypotheses maintain different views about the validity of the observations and different response predictions. In the next section a method for forming these multiple hypotheses is discussed.

3.3 Defining a Hypothesis

Throughout, a hypothesis is represented as the 1-dimensional smoothing spline, but could be replaced by alternate regression techniques. A smoothing spline is a piecewise cubic spline regression technique that can be placed within a Bayesian framework, calculated as (Wahba, 1990):

$$S_{w,\lambda}(f) = \sum_{i=1}^{n} w_i \left(y_i - f\left(x_i\right)\right)^2 + \lambda \int_a^b \left(f''(x)\right)^2 dx \tag{3.1}$$

where experiment parameter and observation pairs x_i and y_i are used to train a regression fit of the data. A weighting parameter, w_i, can be applied each x_i, y_i pair, and the

hyperparameter λ controls the amount of regularisation, with b and a being the maximum and minimum of the x_i values respectively. Further discussion of the regression techniques used are provided in Appendix A.

The combination of training observations, regularisation and weightings form a hypothesis, which is the minimiser of the smoothing spline function for a particular w and λ:

$$h = \min S_{w,\lambda}(f) \tag{3.2}$$

The method for choosing these parameters is discussed in the following section.

3.4 Building Multiple Hypotheses

To allow for a more principled approach to creating an ensemble of hypotheses, we must consider what is required. Firstly there are differences in opinion of the shape of the response curves, and to what extent they are linear or nonlinear. This can be achieved through using different regularisation parameters, where a range of parameters can be considered to allow for fits that may provide a linear fit or a near point-to-point fit. The range of regularisation values should ensure parameters exist that will not over or underfit the data.

Building hypotheses based on subsets of the available data allows for a number of observations to be available for unbiased evaluation of the hypothesis. With the number of observations available being small, the amount of observations available to do this will be restricted. Evaluation techniques are discussed further later.

Most importantly in an experimental domain where erroneous observations are possible, is the ability for hypotheses to disregard certain observations. To achieve different views of observations with questionable validity, observations can be weighted differently in the regression calculation. In Zembowicz and Żytkow (1991) and Christensen et al. (2003), where the accuracy of observations could be determined from the data available, deliberate weighting of observations has been applied to obtain better predictions of the underlying behaviours. But in the present problem, obtaining accuracy information is restricted by resources. Instead, as multiple hypotheses allows different views about the validity of the observations to be considered in parallel, then different weightings can be considered in parallel. For example one hypothesis may consider the observation valid and another hypothesis may consider it erroneous, with all other parameters remaining the same. This allows any decisions about observation validity to be postponed until sufficient evidence is available.

3.4.1 Weighting Observations

Applying different weights to the observations allows for different fits of the data to be achieved, as shown in the previous chapter. By applying a high weighting to an observation, the resulting regression fit of the data will be pulled closer to that observation, up to the point where the regression is forced to pass through that data point. Whilst giving an observation a zero weighting will remove it from consideration in the regression calculation, meaning that if the observation is outlying from the other observations the result of the regression will not pass near the observation. This characteristic of observation weighting can be exploited within a hypothesis framework.

Take the example considered in Figure 3.3, where the original example from Figure 3.2 is shown alongside a potential hypothesis in Figure 3.3a that considers all observations equally likely. If observation A is identified as being in disagreement with the hypothesis, then it could suggest the hypothesis is wrong or the observation is erroneous. As there is insufficient data available to decide, new hypotheses that consider the observation to be erroneous or accurate can be formed to allow both views to be considered in parallel. If the observation is erroneous, then we require the hypotheses to ignore the observations. However, if the observation is representative of the underlying behaviour and it is highlighting a failure in the current hypothesis, then we would require new hypotheses to be created that provide response curves that move closer to the observation. Therefore observation weighting can be applied within a hypothesis to allow it to state its belief about the validity of the observations, where a zero weighting states the observation is believed to be erroneous and a high weighting states the observation is believed to be accurate.

The value of the weighting of each observation could be determined through the hypotheses confidence in each observation and be chosen from a continuous variable. For the sake of simplicity, the current design for hypotheses within the artificial experimenter presented choose observation weightings from one of three values. By default all observations are set to 1 when a new hypothesis is created. Observations are then set to 0 if the observation is believed to be erroneous by the hypothesis and arbitrarily 100 if the observation is believed to be accurate. The value of 100 will cause the regression prediction to pass through or near the weighted observations in most cases.

With a method of presenting hypotheses that can be built in terms of whether the hypotheses believe the validity of observations, the next stage is to identify the observations of questionable validity.

3.4.2 Identifying Erroneous Observations

Ideally a complete set of hypotheses could be created that question the validity of all observations in all combinations. However the computational complexity to do this is

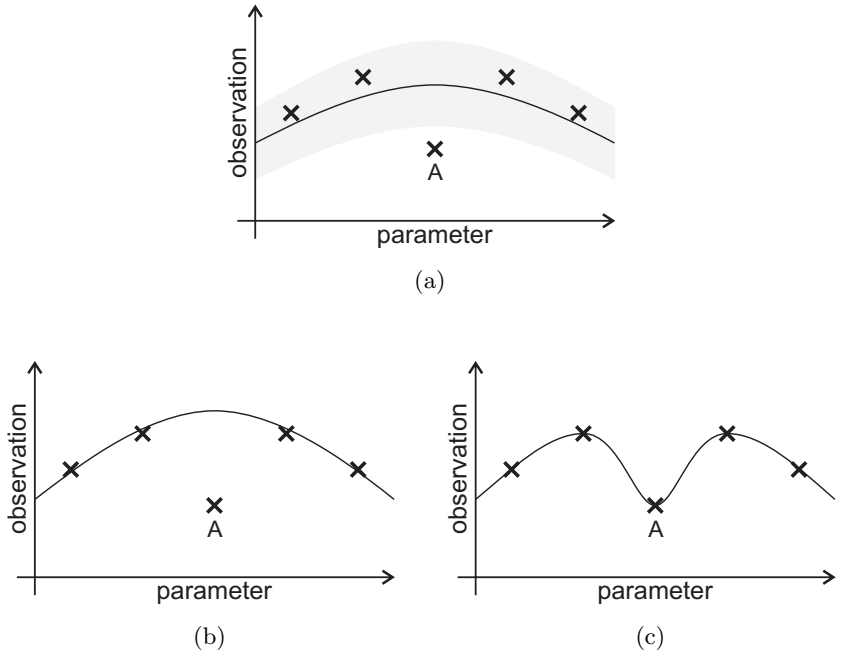


Figure 3.3: Effect of weighting observations in a hypothesis. In (a) all observations are considered with equal weighting with example error bar highlighted grey, in (b) observation A is considered erroneous with a zero weighting and in (c) observation C is considered valid with a high weighting.

currently too great, even when only a handful of observations are available to train from. Instead methods are required to identify where a hypothesis and an observation are in disagreement, as such disagreement is either indicating an erroneous observation or suggesting that the hypothesis is invalid. One method of doing this is to inspect the error bar value from the smoothing spline component of the hypothesis.

As discussed in Chapter A, the smoothing spline can provide a predictive error bar across the regression. This error bar value can be used as an active threshold for determining whether an observation is in agreement or disagreement with a hypothesis, as shown in Figure 3.3, where observation A falls outside the error bar in the initial hypothesis shown in (a). Observations that fall outside of the error bar can be regarded as in disagreement with the hypothesis and those inside are in agreement.

This technique works best when the hypotheses are confident in their predictions. When there is a higher amount of uncertainty in the regression fit the error bars will be larger, engorging all observations. Larger error bars are more likely to occur when regularisation is higher and a more linear fit is not representative of the data available. By having several regularisation values available, chosen to promote fits ranging from under to over fitting of the data, such occurrences should be infrequent with a large enough set of hypotheses.

With a method for detecting potentially erroneous observations, next we consider what happens when a disagreement between hypothesis and observation is detected.

3.4.3 Refining Existing Hypotheses

When an observation and a hypothesis are in disagreement, it could be either that the observation is erroneous or the hypothesis is erroneous. To allow for either case, the parameters of the hypothesis can be taken and used to form two new refined hypotheses, one that considers the observation to be erroneous and another that considers the observation to be accurate, with all three hypotheses being kept in consideration until further evidence is available to disprove them. This refinement allows the weighting parameters to be learnt.

The process of hypothesis refinement occurs as follows. All parameters from the original hypothesis are taken and copied into two new hypotheses. The parameters of these new hypotheses are then altered to allow for the different views in the validity of the observation of questionable validity. One hypothesis will set the weight of the observation to be high, with the other setting the weight of the observation to be zero. If the observation was not within the original hypotheses training set, then the observation is added.

Importantly all hypotheses are maintained within a working set of hypotheses under consideration, until further evidence is available to disprove them. This allows for the decision about the validity of the observations to be postponed until a later stage of experimentation. Next we consider how hypotheses are evaluated with the observations available.

3.4.4 Evaluating the Confidence of the Hypotheses

In many machine learning problems, the available observations may split into two sets, one to train the learner with and the other for testing and evaluation of the learner. This approach is often used when there are large numbers of observations available, as it allows for learning and evaluation to occur on separate data to test the generalisation of the learner (Bishop, 2006). However, in the present problem, there are arguments against this approach. First as only small numbers of observations available, splitting the data will further reduce the amount of data the learner can train from and evaluation will only occur from one or two observations. Evaluating on a small number of observations is further problematic if one of those observations is erroneous, as the evaluation will be invalid and could lead to a hypothesis that well represents the underlying behaviour being rejected from consideration.

Alternatively hypotheses could be evaluated solely on the observations obtained after they were created. This allows for hypotheses to be created using all of the available data to potentially allow for a better representation of the underlying behaviour at the time of creation, whilst also allowing a fresh set of observations to evaluate the hypothesis from the experiments performed after its creation. However, there is again the drawback of the test set being small, especially soon after a hypothesis is created. Additionally, there may become a bias towards hypotheses depending on how many observations they are evaluated with. For example a hypothesis with only one well fitting test observation may artificially appear to perform better than a hypothesis evaluated with several reasonably well fitting observations.

Instead we evaluate hypotheses against all of the available observations, with one caveat that hypotheses are created using a subset of the available observations. By requiring hypotheses to be built on a subset of the available data, it ensures that some observations will be available to test the hypothesis that have not been trained with. With further experimentation, the number of observations used to evaluate that have not been trained with will also grow. This then allows a squared loss error function to be applied to evaluate the hypotheses using all available observations. However, further consideration to the effect of erroneous observations should be made in future work to try and improve the performance of an evaluation metric in this setting.

3.4.5 Representing the Hypotheses to the User

Whilst using multiple hypotheses allows for different views to be considered throughout experimentation, at the end of experimentation a usable view of the experiment parameter space under investigation should be returned. A disadvantage of multiple hypotheses, is that in our mind humans can struggle to visualise several different views about the same behaviour being investigated (Chamberlin, 1890), so returning the whole set of hypotheses under consideration by the artificial experimenter is not desirable. However, that is not to say that the alternate hypotheses should be thrown away, as depending on the amount of further investigation required into the behaviour, those workings of the artificial experimenter may be valuable.

Instead we are faced with a problem of how to capture the information held in the hypotheses, to return it to the user. One approach may be to merge the predictions of the hypotheses, to arrive at a representation that is a mean or weighted mean of all of the hypotheses predictions. However, when hypotheses have differing views about the validity of observations, taking the mean of those hypotheses will most likely remove the benefit of having the different opinions about the validity of the data and revert to an outcome similar to a single hypothesis approach that averages through all the data.

Alternatively the most confident hypothesis may be returned as the current representation of the underlying behaviour under investigation. This has the benefit of providing a single view of the underlying behaviour that is believed to be best representative of the data. However if parts of the hypothesis are not well tested then errors may exist within it that have been addressed in alternate hypotheses. Therefore a selection of the most confident hypotheses could be returned, providing a set that was small enough to manage but large enough to cover different possibilities. In returning a set of possible hypotheses, techniques for identifying structural differences in the responses predicted could be used so that only confident yet structurally different hypotheses are returned.

In the evaluations considered later, for simplicity we evaluate the technique using the most confident hypothesis. However, in laboratory usage where a real user is provided with the final output from the artificial experimenter, returning the most confident yet different hypotheses would be more useful for the human scientist.

3.4.6 Process of Hypothesis Management

Given an initial set of observations, for example two observations in the 1-dimensional case to allow sufficient data for simple hypotheses to be created, the following procedure for hypothesis management is performed after subsequent experiments. Using all observations, a set of new hypotheses are created using random subsets of the available observations along with randomly selected smoothing parameters. These new hypotheses, proposed randomly, allow for different initial views of the parameter space. The smoothing parameter for each hypothesis is chosen from a set of possible parameters ($\lambda \in \{10, 50, 100, 150, 500, 1000\}$) that allow for a range of different fits of the data, further promoting different initial views of the behaviour being investigated. All of these new hypotheses are added to the set of hypotheses maintained from the previous round of experimentation, to form what will be called the working set of hypotheses.

Next the hypotheses go through the process of refinement, by seeing if any hypotheses and observations disagree. To do this, all observations are compared against all of the working hypotheses. Using the error bar of the hypothesis as an active threshold for agreement, as discussed in Section 3.4.3, any observation outside the 95% confidence interval of the prediction of a hypothesis is regarded as being in disagreement with that hypothesis, referred to here as h_{original}. As an observation in disagreement could either be erroneous or showing an error in the hypothesis, h_{original} is refined into two new hypotheses. One of these new hypotheses will declare the observation to be valid, h_{valid}, with the other declaring the observation to be erroneous, $h_{\text{erroneous}}$. Both h_{valid} and $h_{\text{erroneous}}$ are based upon h_{original}, which is left unchanged in the working set of hypotheses. The two new refined hypotheses are altered from h_{original} by including the suspect observation in their training observations with different weightings, where h_{valid} will give the observation a high weighting (currently set arbitrarily at 100), whilst

$h_{\text{erroneous}}$ will give the observation a weighting of 0. Both h_{valid} and $h_{\text{erroneous}}$ are added to the working set of hypotheses.

After all combinations of hypotheses and observations have been compared to test their agreement and refinements have been made, the working set of hypotheses are evaluated. This evaluation is performed using a squared loss error function to compare hypotheses against all available observations. In the simulated and laboratory evaluations presented in the following chapters, this error function is:

$$C(h) = \frac{1}{N} \sum_{n=1}^{N} \exp \left(\frac{-\left(\hat{h}(x_n) - y_n\right)^2}{2\sigma^2} \right) \tag{3.3}$$

where $\hat{h}(x_n)$ is the hypotheses prediction for experiment parameter x_n, with y_n being the real experimental observation for parameter x_n, σ is chosen a priori (currently 1.96), and N is the number of observations. The result of the evaluation is represented as the confidence of the hypothesis, $C(h)$, which provides a value 0 to 1.

Finally the hypotheses are ranked by confidence, with any duplicate hypotheses removed from the set. For computational efficiency, the number of working hypotheses considered in parallel can be reduced. Removing the hypotheses that perform the worst in the evaluation stage, ensures that whilst the number of hypotheses considered in parallel remains large, it does not become computationally infeasible to inspect in the experiment selection stage or subsequent iterations of this hypothesis management procedure. In the trials presented in the evaluation chapters, 200 new random hypotheses are created in each iteration, and the best 20% of all hypotheses under consideration are maintained into the next round of experimentation. From initial trials it appears that so long as the number of new hypotheses created is large, the number of hypotheses retained after each experiment can be altered as required for performance.

3.5 Comparison to Single Hypothesis Approaches

To demonstrate the characteristics of the multiple hypotheses technique and to compare it with standard single model methods, a comparison is made here over three different scenarios. Further evaluation and comparison of the multiple hypotheses technique to a single hypothesis approach are given later in Chapter 6. In each scenario presented here, ten observations are chosen equally spaced across the parameter space. Two methods for parameter learning are selected for the single hypothesis approach, leave-one-out cross validation and bootstrapping. The same smoothing parameters are available in both. In the bootstrapping parameter learning, 200 iterations of randomly selecting training and test sets are made to compare the smoothing parameters. In the multiple

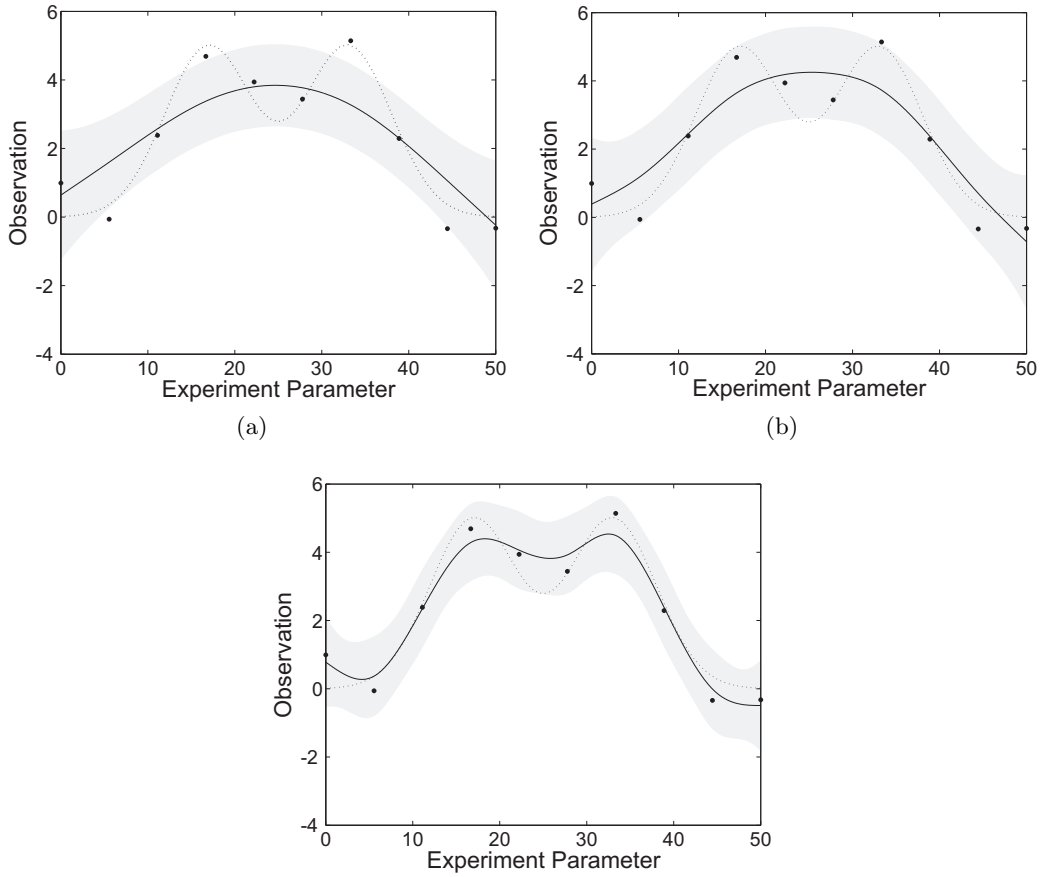


Figure 3.4: Comparison of single and multiple hypotheses techniques with non-monotonic underlying behaviour. In all the hypothesis is shown as the solid line with shaded error bars, observations are dots and the underlying behaviour used is shown as the dashed line. In (a) leave-one-out cross validation is used to learn the smoothing parameter for a single hypothesis. In (b) bootstrapping is used to learn the smoothing parameter for a single hypothesis. In (c) the most confident hypothesis of the multiple hypotheses technique is shown.

hypotheses technique, the most confident hypothesis under consideration using $C(h)$ stated in Equation 3.3 is used to evaluate the hypotheses.

In the first scenario, a nonmonotonic underlying behaviour is used with two peaks, where observations near those peaks may appear as noise or erroneous observations. As shown in Figure 3.4, both single hypothesis methods underfit the underlying behaviour, failing to characterise the two peaks in the behaviour. Whilst the multiple hypotheses approach identifies that there are two peaks, without applying a high weighting to any observations in the hypothesis shown, although it does not match the amplitude of the peaks and trough in that region. However, with the data available matching the amplitude would require overfitting the training data available. Additionally for the problem of enzyme characterisation, identifying any notable features of a behaviour is a suitable solution. If the exact details of those features are required, then additional experimentation can be

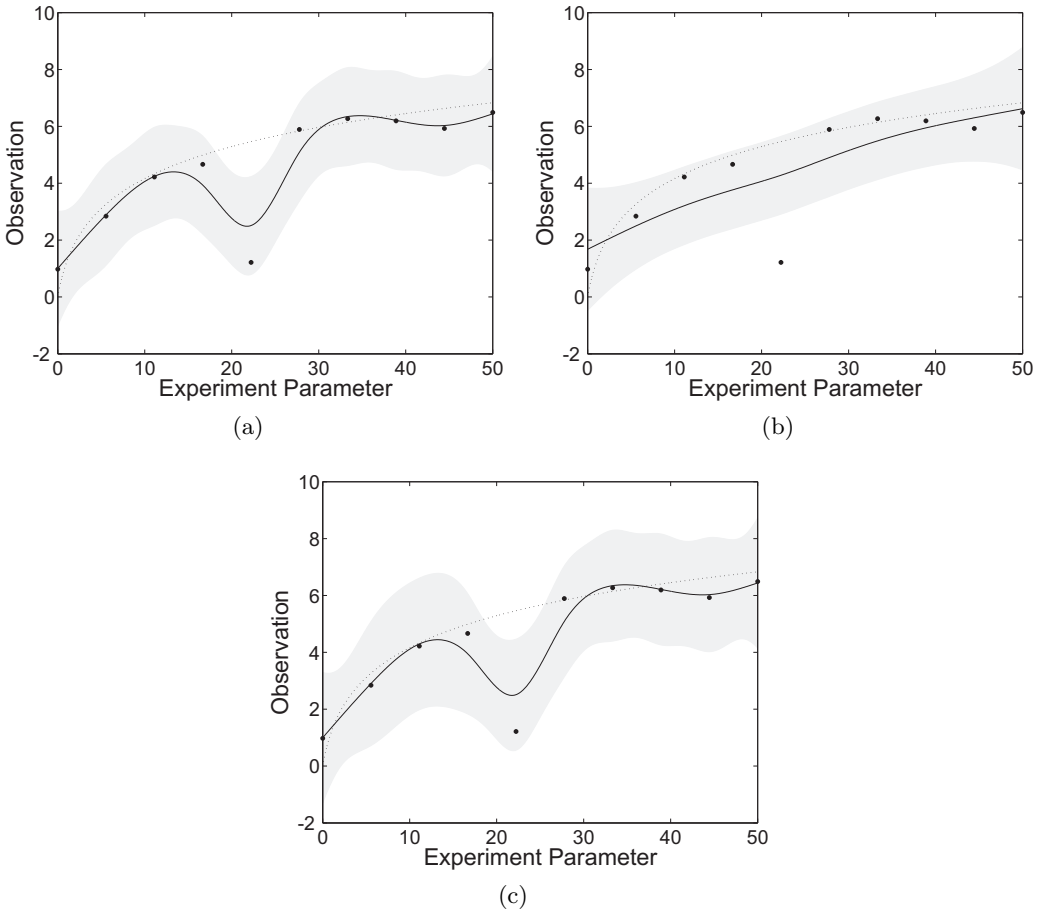


Figure 3.5: Comparison of single and multiple hypotheses techniques with an erroneous observation. In all the hypothesis is shown as the solid line with shaded error bars, observations are dots and the underlying behaviour used is shown as the dashed line. In (a) leave-one-out cross validation is used to learn the smoothing parameter for a single hypothesis. In (b) bootstrapping is used to learn the smoothing parameter for a single hypothesis. In (c) the most confident hypothesis of the multiple hypotheses technique is shown.

performed later. This would mean that the single hypothesis techniques in this situation would not be suitable for quick enzyme characterisation.

In the second scenario, a simpler underlying behaviour is used, but one of the observations is erroneous. A simpler behaviour is used so that it is only the effect of the erroneous observation that should cause differences between the hypotheses. As shown in Figure 3.5, it is the bootstrapping single hypothesis that best ignores the erroneous observation, with the cross validation and the multiple hypotheses approaches providing the same hypothesis. In this instance the multiple hypotheses technique appears to fail to detect the erroneous observation, however as the method is able to considers observations as being erroneous and valid, an alternate hypothesis will exist that states that observation is erroneous. As only the single most confident hypothesis is shown and confidence is determined by a squared loss against all observations, then in this situation

the hypothesis that passes closer to all observations will be held in higher regard than those that do not. If more experimental evidence was obtained around the erroneous observation, then the observation should be ignored. Interestingly the bootstrapping approach is able to identify the erroneous observation, but the cross validation does not. However the hypothesis generated by the bootstrapping technique highlights one of the problems with the single hypothesis technique. For if the erroneous observation was in fact a true representation, the bootstrapping hypothesis would not detect it, whilst the multiple hypotheses technique would. Where as if the observation were erroneous, the bootstrapping technique would declare it to be erroneous first, however with further experiments the multiple hypotheses technique would also identify the error. This means there is a trade-off between having the hypotheses discover new features by occasionally overfitting the data and being too conservative in the identification of erroneous observations.

To confirm that with a small number of additional observations the multiple hypotheses technique will spot the error, in the third scenario, the second scenario is repeated but with the erroneous observation being tested with a further two experiments performed nearby in the experiment parameter space. As shown in Figure 3.6, the multiple hypotheses technique now correctly identifies the erroneous observation, and produces a hypothesis with small error bars that closely matches the underlying behaviour. The leave-one-out cross validation technique is still influenced by the erroneous observation, creating a highly uncertain hypothesis. The cross validation technique is interesting in its ability to average through the important observations in the first scenario, but be influenced more by the erroneous observation in the second and third scenarios. Whilst the bootstrapping technique provides a hypothesis similar to the one in the previous scenario, ignoring the erroneous observation.

With a single outlying observation, the multiple hypotheses approach may be more likely to overfit to the data and produce a hypothesis that passes near that observation. If the observation is erroneous, this will only be a problem if the validity of the observation is not considered by any alternate hypotheses. With the alternate hypotheses, the experiment manager should trigger experiments to evaluate that observation, allowing more accurate hypotheses to be created in the future. The single hypothesis approach will likely ignore that observation by averaging across all observations. However, this means that if the outlying observation was actually valid, the single hypothesis approach would not spot this and would only discover it if further experiments were placed in that location by luck.

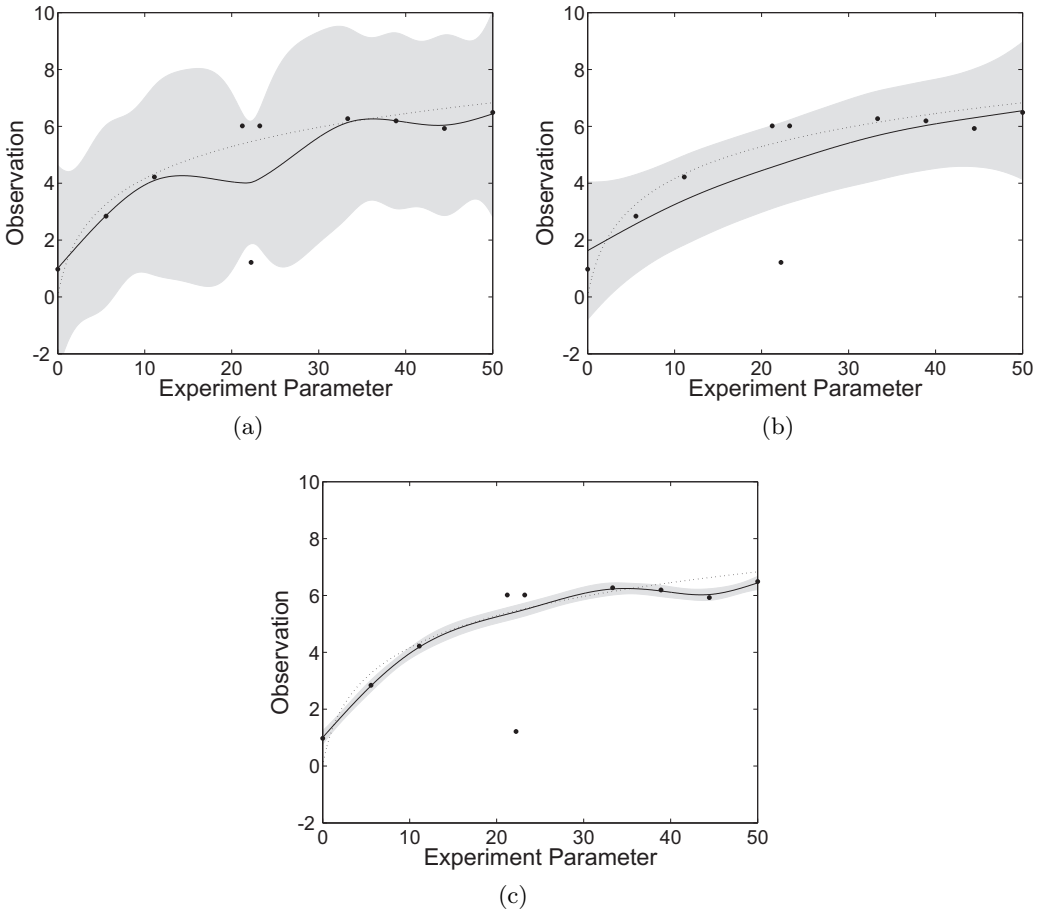


Figure 3.6: Comparison of single and multiple hypotheses techniques with an erroneous observation that has been further examined. In all the hypothesis is shown as the solid line with shaded error bars, observations are dots and the underlying behaviour used is shown as the dashed line. In (a) leave-one-out cross validation is used to learn the smoothing parameter for a single hypothesis. In (b) bootstrapping is used to learn the smoothing parameter for a single hypothesis. In (c) the most confident hypothesis of the multiple hypotheses technique is shown.

3.6 Conclusions

A key problem for a hypothesis manager, is how to handle uncertainty in the form of erroneous observations. By accepting all observations as valid, errors can mislead the development of hypotheses. Determining the validity of observations is impeded by the limited resources, which prevent repeat experiments. In this situation, maintaining a single hypothesis appears inefficient in obtaining an accurate representation of the underlying behaviour. Alternatively, we can consider using multiple hypotheses that maintain different views of the validity of the observations in parallel. Whilst many multiple hypotheses based approaches produce hypotheses using random subsets of the data (Freund et al., 1997; Abe and Mamitsuka, 1998), we believe a more structured

approach can be applied to deal with the uncertainty about the validity of the observations. That is, where an observation appears erroneous, separate hypotheses can be used in parallel that consider the observation as erroneous or valid, with further experimentation providing the evidence to differentiate between the hypotheses (Lovell et al., 2010c, 2011, 2010a).

In review, the hypothesis manager maintains an expanding ensemble of working hypotheses throughout the experimentation conducted. Learning is achieved in a method similar to boosting, with the main difference being that the different hypotheses are maintained throughout the learning process. The hypotheses have a parameter to control the amount of regularisation applied in the smoothing spline calculation, along with parameters controlling the weightings applied to the observations. The regularisation parameter, λ, is applied randomly to the new hypotheses created after each observation is obtained, so as to provide different initial views of the behaviour being investigated. Whilst the weighting parameters of hypotheses are refined in cases where observations contradict with the predictions of the hypotheses. These different weightings allow for hypotheses to have different views about the validity of the observations. Maintaining a set of different hypotheses is desirable when there are few, potentially erroneous observations, as there will not be enough information to accurately identify erroneous observations. Additionally, it allows for the different hypotheses to be utilised by an active learning experiment selection technique, which can look to differentiate between the hypotheses, in a manner similar to that conducted by scientists. This active learning is discussed next.

Chapter 4

Separating Sets of Hypotheses

Many active learning tasks reside within a domain whereby obtaining of labels is expensive yet reliable (Freund et al., 1997; Settles, 2009). However, physical experimentation rarely affords reliable observations. Take for example biochemical experimentation, where each experiment will have some error. This error may be caused for example by inaccuracies of the measurements, or due to a lack of control over experiment variables such as enzymes varying from batch to batch. Managing this uncertainty is further impeded by limited resources, allowing for only a handful of experiments per parameter dimension. Therefore, in physical experimentation the learner must learn from a small, noisy and potentially erroneous set of observations.

The combination of erroneous observations and a lack of resources to test the validity of all observations, leads to uncertainty in the learner. In particular, when the learner is presented with an observation that does not agree with a hypothesis under consideration, the learner faces a dilemma of whether it is the observation that is erroneous or whether it is the hypothesis that is incorrect. As shown in the previous chapter, one approach to address this issue is to use multiple hypotheses, whereby questionable observations are classified as both valid and erroneous in parallel through different hypotheses. Decisions about the validity of these hypotheses can then be postponed until further experimental evidence is available. However, through the course of experimentation this can lead to the development of many different hypotheses that are similarly effective in representing the available observations. In this situation active learning can be applied to select experiments that will differentiate between the hypotheses the most, in order to identify the optimal hypothesis efficiently.

In this chapter we consider the effective separation of hypotheses through active experiment selection. To do this we consider an abstract toy problem, where the goal is to identify the hypothesis that best represents the behaviour with the smallest number of experiments. In this chapter hypotheses and their creation are kept abstract, to provide a simple toy problem that can be addressed by the active learning techniques presented.

4.1 Formulation of Problem

In this toy problem, a pre-defined set of hypotheses are investigated by choosing experiments from a simulated underlying behaviour, to identify the hypothesis that best represents the underlying behaviour. To model the underlying behaviour, one of the hypotheses is used to provide observations with additive Gaussian noise for requested experiments. Throughout this chapter the hypothesis that represents the underlying behaviour will be referred to as the true hypothesis. The difficulty of the problem can be changed by altering the similarity between the hypotheses, where the more similar the hypotheses are, the harder it will become to separate them. The method used to create these hypotheses and to evaluate the performance of the techniques are given next.

4.1.1 Toy Hypothesis Formation

As hypotheses in a real experimental setting will have been built from the same sets of observations, then in most cases the hypotheses will be structurally similar. However, there will also be some significant differences where hypotheses take different views about the accuracy of observations. To create a set of hypotheses that have these characteristics we start out with a large set of data points, equally distributed over the x parameter from some function modelling a phenomena, where the y values are altered by large Gaussian noise $N(0, 2^2)$. Hypotheses, each represented as a smoothing spline (Wahba, 1983), are created using randomly selected subsets of these data points, with differing regularisation parameters. Note we are not creating hypotheses as stated in the previous chapter, we are simply creating an abstract set of similar hypotheses. This set of data points is then discarded, leaving a set of hypotheses. This set of hypotheses have similar characteristics to those that can result from the hypothesis manager of an autonomous experimentation system, by giving different perspectives of the same data. The hypothesis that is most similar to all other hypotheses in a least squares manner, is selected to represent what we will refer to as the true hypothesis in that set of hypotheses. The most similar hypothesis is chosen to make the problem harder. The true hypothesis will be used to provide the observations, altered by Gaussian noise $N(0, 0.5^2)$, for experiments requested by the active learning algorithms. The goal for the active learning techniques is to identify the true hypothesis.

4.1.2 Evaluation Method

From a set of hypotheses, the active learning techniques must obtain a set of observations that can be used to clearly differentiate the best, or in this toy case true, hypothesis. To measure whether this has been achieved, we will assume each hypothesis has a confidence

that states how well the hypothesis believes it represents the underlying behaviour. The first goal is to achieve a situation where the hypothesis with the highest confidence is also the true underlying hypothesis. Next we are interested in how quickly this first goal can be achieved by the active learning techniques. Finally when the first goal has been achieved, we are also interested in the difference in confidence between the most confident hypothesis and the next most confident hypothesis. This final goal is important for translation to a real system, as by maximising this difference we give the end user the greatest assurance that they can believe the most confident hypothesis is the best hypothesis available, assuming there was a wide range of alternate hypotheses.

To evaluate the techniques we use the following evaluation function:

$$E(\mathcal{H}) = \max\left(C(h_t) - \max_{h \in \mathcal{H}, h \neq h_t} C(h)\right) \tag{4.1}$$

where $\mathcal{H}$ is the working set of hypotheses under consideration, $C(h)$ is a function returning the confidence of a hypothesis, and h_t is the true hypothesis. When this function is positive, it means that the most confident hypothesis is the hypothesis that was selected to be true, h_t, so the best available hypothesis has been chosen. Positive values of this function indicate the amount of evidence that has been obtained to distinguish the best available hypothesis from the next alternate hypotheses. Larger values of this function would give someone who did not know the underlying behaviour more confidence that hypothesis with the highest confidence is actually the best available hypothesis under consideration, as the next best alternate would have a lower confidence.

4.2 Active Learning Techniques

Design of experiments and sequential learning have considered the problem of actively selecting observations to differentiate between hypotheses. In particular T-optimal designs have been proposed, however they require the selection of the most likely hypothesis and can perform poorly if that hypothesis is similar to other hypotheses under consideration (Atkinson and Fedorov, 1975a). It is likely that in the experimentation considered here the set of hypotheses will be similar, as they are generated based on the same small set of observations.

There exist active learning techniques for separating committees of hypotheses, however the majority consider the problem of classification often where there is a known set of potential labels (Settles, 2009). Additionally, those techniques often rely on hypotheses being rejected on the grounds of a single disagreeing observation, which is not suitable in a situation where erroneous observations are possible. However, those techniques can be adapted by replacing the binary agreement with a continuous measure of confidence. This confidence can then be used to weight the contribution of the hypothesis, so that

as hypotheses have more observations disagreeing with them their confidence lowers and in turn they are given less importance when choosing the hypotheses to separate. In the following we make this conversion, but before this we now introduce some common terminology used throughout.

In the following we represent an experiment parameter as x, with associated observation (or label) y. A set of hypotheses $\mathcal{H}$, of size $|\mathcal{H}|$, with individual members referenced as h_i, where the prediction of a hypothesis is written as $\hat{h}_i(x)$. We measure the belief of a hypothesis that performing x will lead to y as $P_{h_i}(y|x)$, represented as a Gaussian function:

$$P_{h_i}(y|x) = \exp\left(\frac{-\left(\hat{h}_i(x) - y\right)^2}{2\sigma^2}\right) \tag{4.2}$$

where σ^2 can either be set as a constant across all hypotheses or derived from the uncertainty of a particular hypothesis for that experiment parameter. The confidence of a hypothesis is noted as $C(h_i)$, which we leave abstract for the moment.

Finally, the majority of existing techniques consider the problem of classification using a discrete set of known labels. However, discrete labels y_i are not known before experimentation begins in the regression scenario considered here. Instead the predictions of the hypotheses for x can be used to provide the different labels for each parameter value, where $y_i = \hat{h}_i(x)$.

4.2.1 Variance

An initial starting point to determine the difference between a set of regression based hypotheses is variance, which has been tested with some limited success (Burbidge et al., 2007). Here we consider a weighted variance metric, where the views of the hypotheses are weighted by their confidences:

$$x^*_{\text{Var}} = \arg\max_x \; k \sum_{i=1}^{|\mathcal{H}|} C(h_i) \left(\hat{h}_i(x) - \mu^*\right)^2 \tag{4.3}$$

where

$$\mu^* = \frac{1}{C} \sum_{i=1}^{|\mathcal{H}|} C(h_i)\hat{h}_i(x) \tag{4.4}$$

and k is normalising constant for weighted variance and C is the sum of all hypotheses confidences.

4.2.2 Vote Entropy

Engelson and Dagan (1996) use an entropy based measure, whereby hypotheses vote on whether they believe y is the expected response for x:

$$x^*_{\text{VE}} = \arg\max_x -\frac{1}{\log|\mathcal{H}|}\sum_{i=1}^{|\mathcal{H}|}\frac{V(y_i,x)}{|\mathcal{H}|}\log\frac{V(y_i,x)}{|\mathcal{H}|} \tag{4.5}$$

where V is the number of hypotheses that believe label y_i is correct for parameter x. In its original form the value of V is binary, however the approach has been adapted to allow for a weighted vote entropy (Olsson, 2009):

$$x^*_{\text{WVE}} = \arg\max_x -\frac{1}{\log w}\sum_{i=1}^{|\mathcal{H}|}\frac{W(y_i,x)}{w}\log\frac{W(y_i,x)}{w} \tag{4.6}$$

where w is the sum of all weights and W is the sum of weights for those hypotheses that agree. In this revised form, we are able to replace the binary votes with the probability term (4.2). Additionally to allow for hypotheses with different confidences, we consider the weighting function to be:

$$W(y,x) = \sum_{i=1}^{h} C(h_i)P_{h_i}(y|x) \tag{4.7}$$

4.2.3 McCallum KL Divergence

McCallum and Nigam (1998) propose a discrepancy method for committee situations where there are discrete known labels, using a KL-divergence approach:

$$x^*_{\text{KLM}} = \arg\max_x \frac{1}{|\mathcal{H}|}\sum_{i=1}^{|\mathcal{H}|}\sum_{j=1}^{|\mathcal{H}|}P_{h_i}(y_j|x)\log\frac{P_{h_i}(y_j|x)}{P_{\mathcal{H}}(y_j|x)} \tag{4.8}$$

where

$$P_{\mathcal{H}}(y_j|x) = \frac{1}{|\mathcal{H}|}\sum_{k=1}^{|\mathcal{H}|}P_{h_k}(y_j|x) \tag{4.9}$$

which is the consensus probability between all hypotheses that the observation y_j will be obtained within some margin of error when experiment x is performed. This discrepancy measure determines the most interesting experiment available as the experiment that causes the largest mean difference between the individual hypotheses and the consensus over the observation distributions.

In the current form this approach requires hypotheses that do not match the observations to be removed. However, if $P_{h_i}(y|x)$ is multiplied by the confidence of the hypothesis,

$C(h_i)$, and the normalising term $\frac{1}{|\mathcal{H}|}$ in (4.8) and (4.9) is replaced with the inverse of sum of the confidences, $\frac{1}{C}$, the impact a hypothesis has on the decision process can be scaled by its confidence.

4.2.4 Bayesian Surprise

The Kullback-Leibler divergence (Kullback and Leibler, 1951) has also been adapted to provide a metric for Bayesian surprise by integrating over the difference between the posterior and prior probabilities (Itti and Baldi, 2009). This technique has been applied previously in active learning to a two-class classification problem (Danziger et al., 2007). Here we consider the prior probability as the mean over (4.2) for the set of all previously performed experiments X with observations Y:

$$P_{h_i}(Y|X) = \frac{1}{n}\sum_{j=1}^{n} P_{h_i}(y_j|x_j) \tag{4.10}$$

Whilst the predicted posterior probability also takes into consideration what the new probability of the hypothesis would be if a particular experiment x_p was performed that resulted in a specific y_p:

$$P_{h_i}(Y, y_p|X, x_p) = \frac{1}{n+1}\left(nP_{h_i}(Y|X) + P_{h_i}(y_p|x_p)\right) \tag{4.11}$$

Using these distributions, we consider all predicted observations to determine a surprise term:

$$x^*_{\text{surprise}} = \arg\min_x \frac{1}{|\mathcal{H}|}\sum_{i=1}^{|\mathcal{H}|}\sum_{j=1}^{|\mathcal{H}|} K\left(h_i, y_j\right) \tag{4.12}$$

where K is the KL divergence to provide Bayesian surprise (Itti and Baldi, 2009):

$$K\left(h_i, y_j\right) = P_{h_i}\left(Y, y_j|X, x\right)\log\frac{P_{h_i}\left(Y, y_j|X, x\right)}{P_{h_i}\left(Y|X\right)} \tag{4.13}$$

Importantly for use in the active learning technique described here, we have chosen to look for the experiment with the minimum of the surprise equation, where as its original usage looked for the maximum. In the problem it is applied to here, looking for the minimum of the surprise equation should find the experiment that weakens all hypotheses. If we were to instead look for the maximum value of the surprise equation, we would be looking for the experiment that gives a surprising improvement to all hypotheses, which by definition will limit the discrepancy between the hypotheses. It can be shown using the framework presented here that using the minimum KL-divergence value results in a better performing discrepancy technique than using the maximum KL-divergence.

4.3 New Active Learning Techniques

This section details new techniques considered for separating regression based hypotheses.

4.3.1 A New Method Using Surprise: Surprise–Explore

Additionally we can alter the surprise equation so that only experiments are performed that provide an overall weakening of the hypotheses. If hypotheses are likely to be weakened by performing a particular experiment, then the KL-divergence will be negative. Therefore, when the sum of the KL-divergences is positive an alternate strategy can be performed. Adding an exploration strategy to complement the exploitation provided by the KL-divergence techniques could be used to address the exploration-exploitation trade-off (Auer, 2002). This will be of value in particular in a situation where the experimentation had to also build the set of possible hypotheses as well as discriminate among them. Using the following exploration strategy that maximises the distance between the potential parameter x and parameters x' where experiments were performed previously:

$$x^*_{\text{explore}} = \arg\max_x \ \min \ |x - x'| \tag{4.14}$$

we can form the following metric combining Bayesian surprise and exploration:

$$x^*_{\text{surprise-explore}} = \begin{cases} x^*_{\text{surprise}} & \text{if surprise} < 0, \\ x^*_{\text{explore}} & otherwise \end{cases} \tag{4.15}$$

4.3.2 Maximum Discrepancy

The goal here is to find an experiment that separates the hypotheses. This can be thought of as choosing the experiment parameter that maximises the difference in predictions between all hypotheses under consideration. Mathematically we consider maximising the integration of the differences between all of the hypotheses, over all possible experiment outcomes:

$$A = \sum_{i=1}^{|\mathcal{H}|} \sum_{j=1}^{|\mathcal{H}|} \int (h_i - h_j)^2 \, dy_t \tag{4.16}$$

where the likelihood function $P_h(y|x)$ can be used to determine the differences in the hypotheses:

$$A = \sum_{i=1}^{|\mathcal{H}|} \sum_{j=1}^{|\mathcal{H}|} \int \left(P_{h_i}(y|x) - P_{h_j}(y|x)\right)^2 dy \tag{4.17}$$

then as $P_{h_i}(y|x)$ is a Gaussian distribution, and distinct y can be taken from the predictions of the hypotheses, we can formulate a discrepancy measure:

$$x^*_{\text{d-incomplete}} = \arg\max_x \sum_{i=1}^{|\mathcal{H}|} \sum_{i=j}^{|\mathcal{H}|} 1 - P_{h_i}\left(\hat{h}_j(x)|x\right) \tag{4.18}$$

where we look for the experiment parameter where the hypotheses disagree the most. In this instance the derivation matches the assumption that labels y_j can be obtained from the different hypotheses predictions, allowing for $P_{h_i}\left(\hat{h}_j(x)|x\right)$ to be replaced with $P_{h_i}\left(y_j|x\right)$ to be consistent with the previous notation. Next we require a method of using the prior information. On subsequent runs, we want to find the discrepancy within the sets of currently agreeing hypotheses, whilst also taking into consideration how well those hypotheses fit the current observations, $\mathcal{D}$. We therefore multiply the disagreement term $1 - P_{h_i}\left(y_j|x\right)$ by $P(h_i, h_j|\mathcal{D})$ defined as:

$$P(h_i, h_j|\mathcal{D}) = C(h_i)C(h_j)P(h_i|h_j) \tag{4.19}$$

where $P(h_i|h_j)$ is a term representing the agreement between h_i and h_j for the previous observations:

$$P(h_i|h_j) = \prod_{k=1}^{N} \exp\left(\frac{-\left(\hat{h_i}(x_k) - \hat{h_j}(x_k)\right)^2}{2\sigma^2}\right) \tag{4.20}$$

Combined this provides the discrepancy equation:

$$x^*_{\text{discrepancy}} = \arg\max_x \sum_{i=1}^{|\mathcal{H}|} \sum_{i=j}^{|\mathcal{H}|} \left(1 - P_{h_i}\right) P(h_i, h_j|\mathcal{D}) \tag{4.21}$$

4.4 Results

Using the hypothesis creation protocol described in section 4.1.1, we consider the effectiveness of 8 discrepancy methods to identify the true hypothesis. These active learning techniques are: random; exploration (4.14); variance (4.3); McCallum KL-divergence (4.8); weighted vote entropy (4.6); surprise (4.12); surprise-explore (4.15); and maximum discrepancy (4.21). We look to find disagreement between sets of 20 hypotheses. A representative example of one of these sets can be seen in figure 4.1. Additionally we use the following function to obtain the confidence of a hypothesis:

$$C(h) = \frac{1}{n}\sum_{i=1}^{n} \exp\left(\frac{-\left(\hat{h}(x_i) - y_i\right)^2}{2\sigma^2}\right) \tag{4.22}$$

where x_i and y_i are the n previously obtained observations and σ^2 is constant throughout.

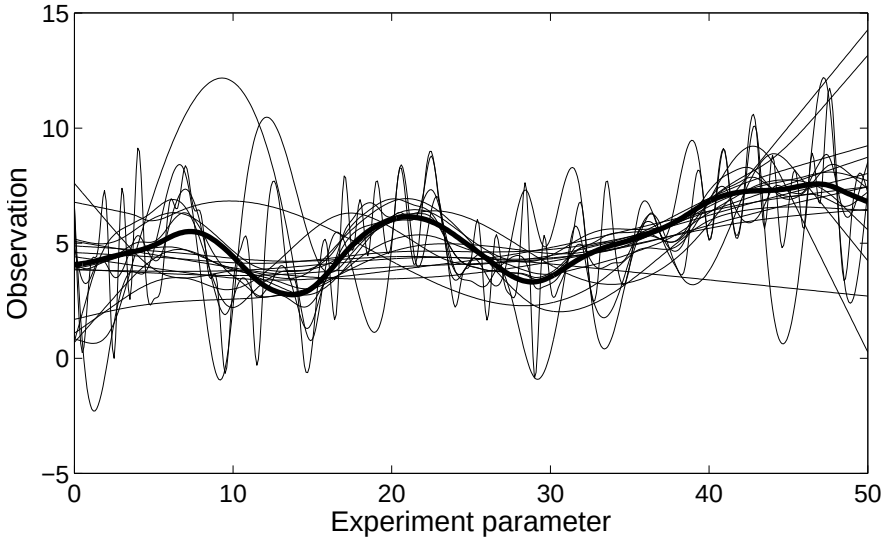


Figure 4.1: Example of a corpus of 20 hypotheses. The hypothesis selected to provide observations for the simulation (the 'true' hypothesis) is shown in bold.

In figure 4.2 the effectiveness of the different search strategies are shown. In (a) the outcome for a set of hypotheses similar to those shown in figure 4.1 are shown. Here the maximum discrepancy approach identifies the true hypothesis the quickest, see table 4.1 for comparison. In (b) the variability between the hypotheses is increased so that hypotheses are less similar to each other. As expected the number of experiments required to identify the true hypothesis is reduced for all techniques. However, for this increased variability the maximum discrepancy approach also demonstrates an advantage over the other approaches, as the difference in confidence between the true hypothesis and the next best hypothesis is larger than the other approaches. In (c) the hypotheses are more similar to each other compared to those in (a). Again as expected, all of the approaches require more experiments before the true hypothesis becomes the most confident hypothesis. In this instance though the variance and maximum discrepancy approaches achieve this quicker than the other approaches, requiring 4 less experiments to the next best approach and 11 less compared to a random search. Next we consider the effect of incorrectly estimating the noise parameter σ^2.

4.4.1 Predicting Noise

In the above results we consider σ^2 in both the confidence function and in the 8 active learning techniques to be constant and equal to the noise applied to the experiments. In reality σ^2 will not be known and will also be estimated from the observations. The error bars provided by regression techniques is one method of obtaining these predictions. As the predictions for the error may be inaccurate, we consider the effect of over and under estimating the noise parameter, shown in figure 4.3, where the original sets of hypotheses used in figure 4.2a are again used but where the predicted σ^2 is double (figure 4.3a) and

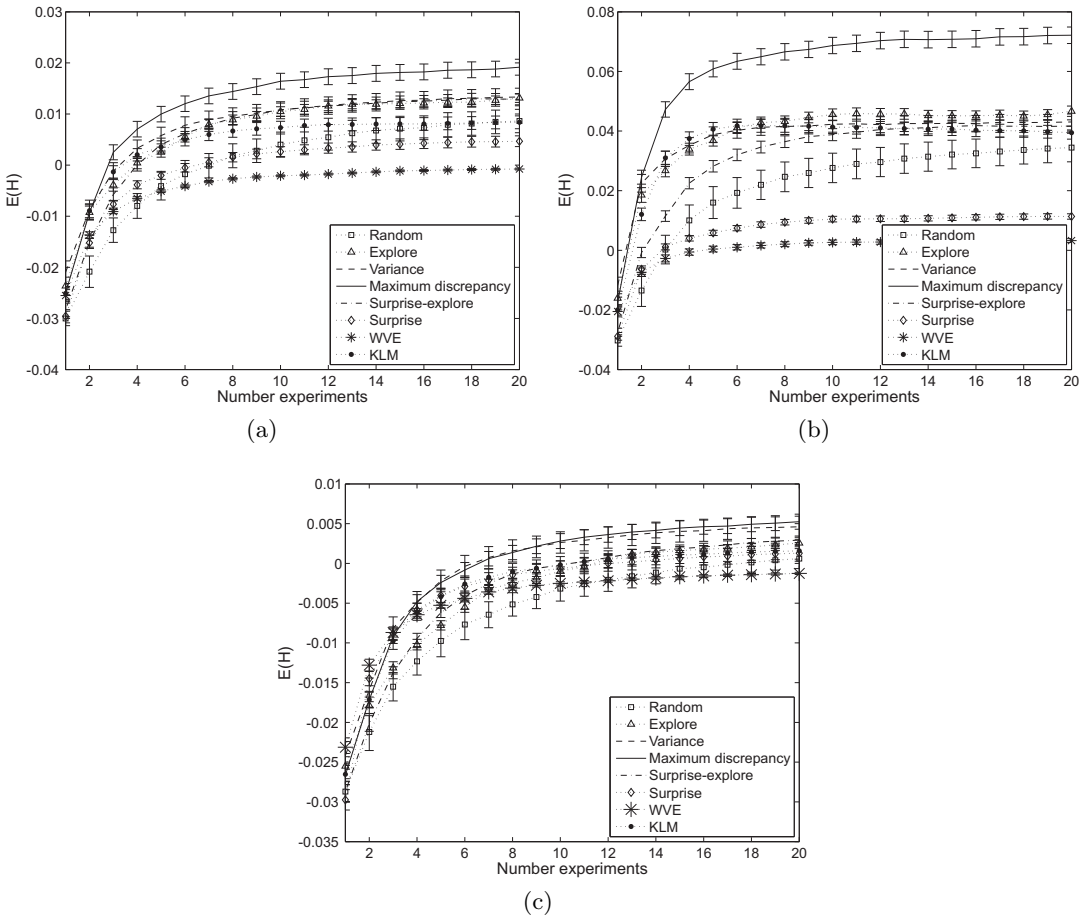


Figure 4.2: Effectiveness of selection strategies for sets of hypotheses with differing variability. In (a) the base line case is shown, in (b) there is more variability between the hypotheses (using $N(0, 4)$ on the initial data point creation) and in (c) less variability between the hypotheses (using $N(0, 1)$ on the initial data point creation). Mean shown for 1000 trials.

half (figure 4.3b) the actual noise value used when generating observations. For the maximum discrepancy and variance approaches, the number of experiments required to identify the true hypothesis remains the same. However, when under estimating the noise, the maximum discrepancy approach is able to produce a slightly larger difference in confidence between the true hypothesis and the next best hypothesis.

4.4.2 Weakness of Variance Strategy

The results in figure 4.2 indicate that when the set of hypotheses are similar, the variance and maximum discrepancy approaches perform almost identically, but when the hypotheses are more varied the maximum discrepancy approach is able to obtain more evidence to support the true hypothesis than the variance approach. When the hypotheses are more varied, there will be greater differences between hypotheses predictions. In some cases there will be situations where a single hypothesis may have a prediction

Table 4.1: Number of experiments until the hypothesis with the highest confidence is the true hypothesis. Table corresponds to zero crossing on fig 4.2 and 4.3. Key: R – random, E – explore, V – variance, D – maximum discrepancy, Se – surprise-explore, S – surprise, WVE – weighted vote entropy, KLM – McCallum KL-divergence.

	Strategy							
Fig	R	E	V	D	Se	S	WVE	KLM
4.2a	8	4	4	**3**	4	7	-	4
4.2b	3	**2**	**2**	**2**	3	3	5	**2**
4.2c	18	12	**7**	**7**	11	13	-	11
4.3a	8	4	**3**	**3**	5	6	-	4
4.3b	8	4	**3**	**3**	5	9	-	4

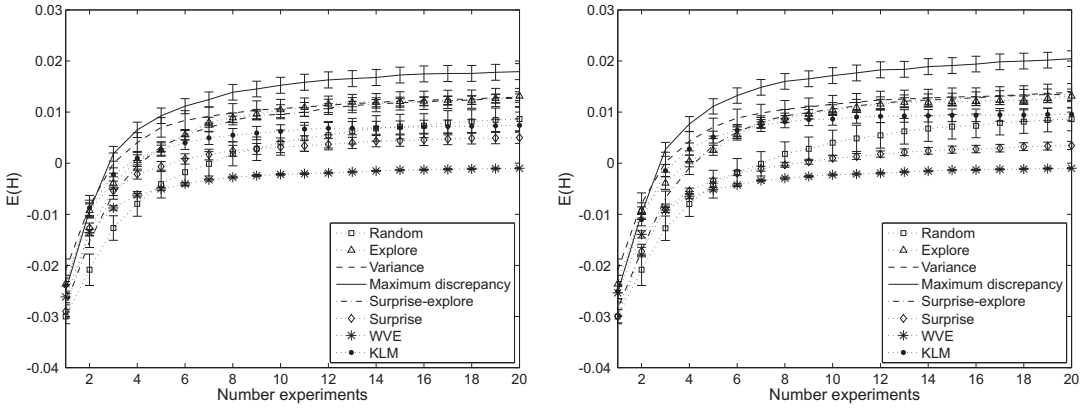


Figure 4.3: Effect of over and under estimating the noise. In (a) the noise prediction is twice the real noise, and in (b) the noise prediction is half the real noise. Mean shown for 1000 trials.

significantly different to the other hypotheses, essentially an outlying hypothesis. Variance measures can be artificially increased by the presence of a single outlying value. Subsequently, we can infer that the variance active learning technique can be misled by an outlying hypothesis into performing an experiment that will only differentiate between the outlying hypothesis and the rest of the hypotheses. It is plausible that it is this behaviour that causes the variance approach to perform worse than the maximum discrepancy approach as the variability of the hypotheses increases.

To demonstrate this problem, consider figure 4.4. Here there are 4 hypotheses with differing views about the expected outcomes for the parameter space. The locations where the variance and maximum discrepancy approaches would select the experiment they expect to provide the most discrepancy between hypotheses is shown. Here the variance approach selects a location where 3 hypotheses give similar predictions and 1 hypothesis has a different prediction. In this instance the variance approach will

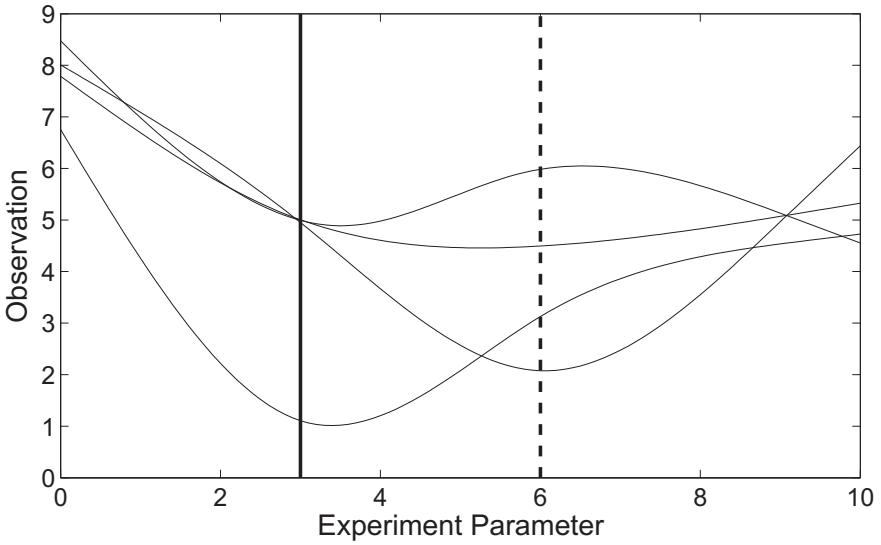


Figure 4.4: Location of experiments selected to maximise discrepancy between hypotheses for the variance and maximum discrepancy. Solid bold vertical line is the experiment parameter the variance approach chooses. Dashed bold vertical line is the experiment parameter the maximum discrepancy approach chooses. The curves show the predictions of the hypotheses across the parameter space.

at best provide evidence against 3 hypotheses and at worst 1. Whilst the maximum discrepancy approach selects the region where all of the hypotheses are the least similar to one another. The maximum discrepancy approach will therefore at best provide evidence against 3 of the hypotheses, and at worst 2 hypotheses in the situation where the observation is in between 2 predictions.

4.5 Conclusions

Presented here is an approach for finding disagreement between a set of regression based hypotheses that is more robust than existing techniques. We have demonstrated that whilst in many cases a maximum variance strategy and the maximum discrepancy strategy proposed provide similar effectiveness in discriminating between hypotheses, there are situations where the maximum discrepancy technique is more robust than a maximum variance method.

When the hypotheses are similar, both the variance and maximum discrepancy approaches work similarly well. However, when there is more discrepancy between the hypotheses, the variance approach can become stuck investigating the difference between a single hypothesis with a prediction that is an outlier with respect to the alternate hypotheses. In the worst case, experiments testing such occurrences could lead to only one hypothesis being disproved. Instead we would prefer a method that will always separate as many hypotheses as possible per experiment and does not suffer such a problem.

The maximum discrepancy method for separating hypotheses can therefore be utilised as part of an experiment selection strategy. However, as this technique only considers the separation of hypotheses, it will only work if a hypothesis that well represents the underlying behaviour exists within the set of hypotheses under consideration. As such the techniques considered here can be thought of as being exploitation strategies, which take the information contained in the hypotheses and seek to build on our knowledge by disproving alternate hypotheses. What these techniques lack is an ability to choose experiments that explore the space to discover new behaviours not yet characterised the hypotheses. In the next chapter we consider the combination of the hypothesis management and experiment selection techniques to form an artificial experimenter. We address this issue of creating an experiment selection technique that uses the information available to it to both explore the parameter space and find new behaviours, whilst also separating the hypotheses to identify the most suitable representation of the underlying behaviour.

Chapter 5

Design for an Artificial Experimenter

The artificial experimenter combines several components. It has a hypothesis manager, which is able to analyse a set of experimental observations, propose differing hypotheses and evaluate them based on the available information. These hypotheses, along with the observations themselves, are used by another component, the experiment manager. The experiment manager decides on the experiments to perform. So far we have discussed the development of an ensemble based hypothesis management system, along with techniques for choosing experiments to effectively differentiate between those hypotheses. In this chapter we will finalise the design for the artificial experimenter, by discussing how these components can be used to provide an effective automatic discovery system. To do this we must further consider the experiment selection strategies and begin by considering the different purposes for performing experiments.

5.1 Experiment Types

Experiments are performed to discover new information and to strengthen the experimenters understanding of the behaviour under investigation. Two types of experiment exist. There are those experiments that explore the parameter space for new features of the behaviour that have not yet been discovered, like a peak, trough or linear region, which can lead to a radically changed view of the behaviour. Then there are those experiments that exploit the information held within the hypotheses, by examining the differences between those hypotheses. This exploitation aims to strengthen the understanding about a particular feature of the behaviour that has already been discovered, and to test the validity of those hypotheses. Both types of experiments are important. The first tries to ensure that all important features are discovered. Whilst the second type ensures the features that are reported are accurate. An experimenter will often have

to balance how much exploration to do, which may waste resources if all the features have been discovered, against how much exploitation to perform.

In this section, first we will formalise these two types of experiments, as an exploration experiment and a exploitation experiment. Then we will consider the risk versus reward for these experiment types and the exploration vs. exploitation problem.

5.1.1 Exploration Experiment

Exploration experiments may be thought of experiment parameters that are randomly selected. However, with small numbers of experiments a random strategy is not best considered, as it may place experiments near previously performed experiments, where the behaviour of the system may be well known. To counter the problem of placing an experiment in the parameter space near where one has been performed previously, an exploration experiment is chosen here to be the experiment that is maximally away from any other previously performed experiment in the experiment parameter space. The exploration score for an experiment is defined as:

$$E(x) = \min_{p \in X} \|x - p\| \tag{5.1}$$

where X is the set of previously performed experiments.

5.1.2 Exploitation Experiment

An exploitation experiment will seek to differentiate between the hypotheses. Exploitation experiments will therefore utilise the most successful hypothesis separation technique from Chapter 4, the maximum discrepancy approach. By using the confidence of the hypotheses, the technique will seek to differentiate between currently well performing hypotheses that have not yet been separated. For clarity, this function is restated here:

$$D(x) = \sum_{i=1}^{|\mathcal{H}|} \sum_{j=1}^{|\mathcal{H}|} C(h_i) C(h_j) A(h_i, h_j) D(h_i, h_j, x) \tag{5.2}$$

where $\mathcal{H}$ is the working set of hypotheses, whilst $C(h_i)$ and $C(h_j)$ are the confidences of the hypotheses using the available observations:

$$C(h) = \frac{1}{n} \sum_{i=1}^{n} \exp\left(\frac{-\left(\hat{h}(x_i) - y_i\right)^2}{2\sigma^2} \right) \tag{5.3}$$

where σ^2 is a constant 1.96. This confidence function will provide a confidence score between 0 and 1. Any observations that are within the spread of the Gaussian function will increase the confidence of the hypothesis, whilst observations outside of the spread

will not. This means that the penalty a hypothesis will receive for having outlying observations that do not agree with the prediction is limited. If a standard mean squared error function were used, then hypotheses would receive greater penalty for missing observations the further the observation was from the prediction. Meaning that if an erroneous observation was obtained that was extremely different from the true underlying behaviour, the penalty applied to those hypotheses that ignored the observation, may prevent those hypotheses from having a high enough confidence to be considered as accurate representations of the behaviour.

The function $A(h_i, h_j)$ is the agreement between two hypotheses for the observations previously obtained:

$$A(h_i, h_j) = \prod_{k=1}^{N} \exp\left(\frac{-\left(\hat{h}_i(x_k) - \hat{h}_j(x_k)\right)^2}{2\sigma^2_{i,x_k}} \right) \tag{5.4}$$

Finally, $D(h_i, h_j, x)$ is the discrepancy between the two hypotheses for the potential experiment under consideration:

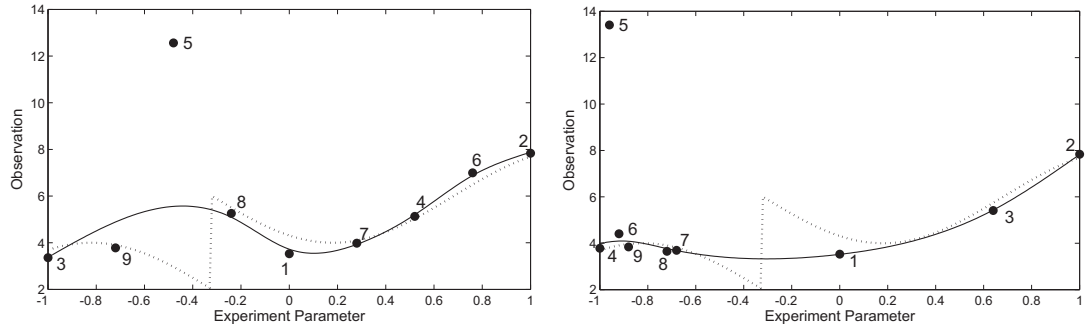
$$D(h_i, h_j, x) = 1 - \exp\left(\frac{-\left(\hat{h}_i(x) - \hat{h}_j(x)\right)^2}{2\sigma^2_{i,x}} \right) \tag{5.5}$$

where in each case $\hat{h}_i(x)$ is the prediction of a hypothesis for the experiment x, and $\sigma^2_{i,x}$ is the error bar value of the hypothesis for the experiment.

5.1.3 Risk versus Reward

To examine the risk versus reward, let us consider a goal for experimentation. As hypotheses can only ever be disproved (Chamberlin, 1890), an experiment may be evaluated based on the amount to which it reduces the confidence in the hypotheses under consideration. A greater reward will be obtained from disproving a hypothesis that currently appears to be good representation of the behaviour than further disproving a hypothesis already believed to be a poor representation. By disproving a hypothesis that is viewed to be good, the experiment will have found some previously unknown feature of the behaviour that the hypothesis failed to predict. This discovery should yield new hypotheses that provide a better representation of the underlying phenomena.

Experiments that explore the parameter space provide a high risk yet potentially high reward strategy. If an exploration experiment obtains an experiment not yet characterised by any of the hypotheses, then that discovery will lead to a significant change in the way the behaviour is viewed. However, there is no guarantee that a new, ground breaking discovery is there to be made, or that the parameter space is small enough



(a) (b)

Figure 5.1: Illustration of exploration and exploitation problems. In both the true underlying behaviour is shown as the dotted line, hypothesis as the solid line and observations as dots numbered in order performed. In (a) an exploration strategy only is used, which does not examine the erroneous observation 5 and fails to identify the discontinuity feature. In (b) 3 initially equally spaced experiments are performed with the remaining purely exploitation experiments, which causes an over investigation of the erroneous observation and the discontinuity feature missed.

for it to be reasonably discovered with the resources available. This means that it is possible to waste experiments searching features that either do not exist or are unlikely to be found. Instead these resources could have been used to either strengthen the case for a particular hypothesis or search a different parameter space.

Experiments that exploit the hypotheses provide a low risk and low reward strategy. These experiments are designed to identify the differences between hypotheses, so are guaranteed to obtain a result that will lower the confidence in some hypotheses. However, in most cases it is likely these differences will be small. The reward can be increased if experiments are chosen to examine the differences between good, highly confident hypotheses. Whilst examining differences between poor hypotheses or between a poor and a good hypothesis will increase the risk and likely lower the average reward, however such experiments may be lucky and provide an unexpected observation that strengthens the weaker hypothesis and weakens the more confident hypothesis. Relying too much on exploitation may cause key features of the behaviour to be missed.

Managing this risk versus reward is an important part of experimentation. A more risky exploration strategy may bring greater reward, but the only guarantee that strategy provides is increased cost. Additionally, too much exploration will not ensure observation validity, as shown in Figure 5.1a. Whilst the less risky strategy may provide hypotheses with greater confidence in the areas of the parameter space examined, but may miss important features of the behaviour that could have greater use, as shown in Figure 5.1b.

The artificial experimenter must consider this problem and a trade-off between experiments that explore the parameter space and those that exploit the hypotheses must be

made. In the following section we discuss this trade-off.

5.2 Managing Exploration-Exploitation Trade-off

The exploration-exploitation trade-off is a common problem in machine learning, where it has received a large amount of attention in multi-armed bandit problems (Auer, 2002; Auer et al., 2002; Antos et al., 2008). Some approaches may switch between an explicit exploration or exploitation strategy, for example the ϵ based strategies considered in multi-armed bandit problems. These switches may be pre-programmed, for instance the ϵ-first strategy that performs a batch of exploration experiments to start (Tran-Thanh et al., 2010). Alternatively the switches may be adaptive over time, such as the ϵ decreasing strategies, where there are random switches between exploration and exploitation, but the probability of an exploration or exploitation experiment alters over time (Sutton and Barto, 1998; Tokic, 2010), or other greedy approaches (Auer et al., 2002). As the spirit of autonomous experimentation is to be a dynamic technique opposed to the static design of experiments, we would prefer this trade-off to also be adaptive.

As with the rest of the artificial experimenter design, we will look to seek ideas from scientific discovery principles when addressing this trade-off. In the following strategies discussed, all will follow an ϵ-first style strategy of performing a small number of exploratory experiments to begin. The ϵ-first strategy is performed as no prior information exists and it will allow an initial set of hypotheses to be formed.

5.2.1 Exploitation Peaks

The exploitation function of $D(x)$ in Equation 5.2, will give a maximal value where the hypotheses most disagree. Performing these experiments when there are good hypotheses in consideration, will identify the hypothesis that most suitably describes the underlying behaviour. However these exploitation experiments will likely focus on particular areas of the parameter space and may place experiments close to each other in the parameter space. This will mean that little exploration will occur and unidentified behaviours may be missed, or only a small number of the differences between hypotheses are examined. As performing purely exploration experiments is risky, as resources may be wasted, one solution to the exploration-exploitation trade-off is to find a method of allowing experiments to placed across the parameter space in an exploratory manner, whilst also fulfilling some exploitation requirement to allow for some guaranteed reward for performing the experiment.

If we consider the function $D(x)$ across the parameter space x, we may expect to see local maxima, or peaks, in the function, as shown in Figure 5.2. In contrast to selecting

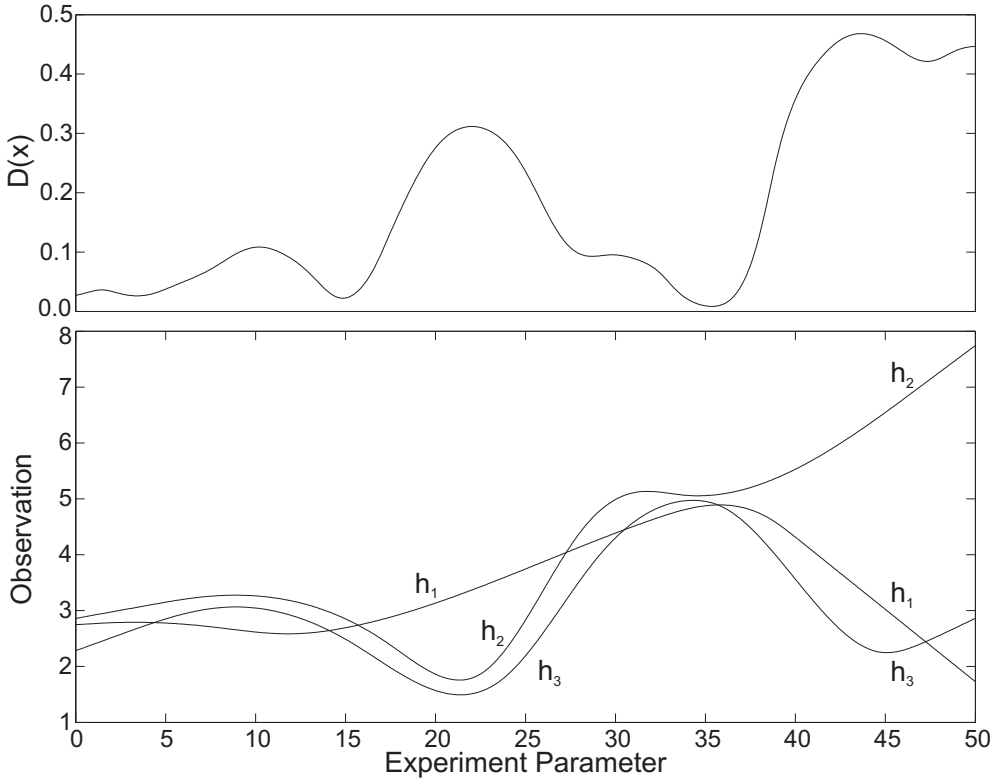


Figure 5.2: Illustration of discrepancy equation across the parameter space. For the hypotheses shown (lower figure), there are distinct peaks in the associated $D(x)$ value for the discrepancy equation (upper figure). These peaks occur in different regions of the parameter space and indicate different reasons for the hypotheses to differ from each other.

the absolute highest values of $D(x)$, these peaks will exist across the parameter space and may be able to identify different features in the behaviour where the hypotheses disagree. Therefore, by placing experiments at these peaks, three benefits will occur. First, there will be a guaranteed information gain through identifying a difference between the hypotheses. Second, different differences between hypotheses will be examined. Finally, experiments will be placed across the parameter space allowing for some additional exploration.

This technique is similar to that described by Castro et al. (2005). In both cases a number of initial exploratory experiments are performed across the parameter space, albeit with Castro et al. (2005) using half of the available resources for this task. Next both investigate interesting areas of the parameter space in parallel. However, the technique employed by Castro et al. (2005) uses the remaining half of available experiments, split between all interesting regions identified after the initial model is built from the exploration experiments. This means that only two models are produced, a lower resolution initial model used to identify interesting behaviours, and a second model with higher resolution in those areas of perceived interest. Conversely the new method presented in this section will repeat the investigation of interesting behaviours, allowing the system

to adapt to identify new interesting behaviours in the parameter space, whilst dismissing erroneously identified interesting behaviours quickly. In the solution proposed by Castro et al. (2005), if an interesting region is identified by an erroneous experiment, a large number of experiments will be wasted through the batch selection of experiments chosen to examine that behaviour, leading to a reduction in the effectiveness of the active learning technique. Where as the present new solution is fully adaptive to the observations obtained, allowing errors to be spotted quickly.

The process for the exploitation peaks experiment selection technique is as follows. Starting with the initial observations and hypotheses, a set of experiments to perform are chosen as those at the peaks of $D(x)$, where experiments are not repeated. Those experiments are then chosen in order of their $D(x)$ value, from largest to smallest, so that if resources are depleted, then the experiments that are likely to differentiate between the hypotheses the most, will have been performed. After each experiment is conducted, new hypotheses are created, but the next set of experiments to perform are only chosen once the current set of experiments have all been performed. This process continues until the maximum allowed number of experiments determined by the user have been performed.

5.2.2 Surprise Based Exploration-Exploitation Switching

Investigating surprising observations, defined as those observations that disagree with a well performing hypothesis, has been highlighted as a technique utilised by successful human experimenters and has also been considered in previous computational scientific discovery techniques (Kulkarni and Simon, 1990; Matsumaru et al., 2002). A surprising observation either highlights a failure in the hypothesis or an erroneous observation. If the observation is highlighting a failure of a hypothesis, especially an otherwise well performing hypothesis with a high prior confidence, then additional experiments should be performed to further investigate the behaviour where that observation was found, to allow the development of improved hypotheses. As such we consider the use of surprise to manage the exploration-exploitation trade-off, where obtaining surprising observations will lead to more exploitation experiments, and unsurprising observations lead to exploration experiments.

A Bayesian formulation for surprise has been considered previously in the literature, where a Kullback-Leibler divergence is used to identify surprising improvements to the models being formed (Itti and Baldi, 2009):

$$S = \int_{\mathcal{H}} P(h|D) \log \frac{P(h|D)}{P(h)} dH \tag{5.6}$$

where $P(h)$ is the prior probability of the hypothesis and $P(h|D)$ is the posterior given some observation D.

In their work, the authors conducted tests to compare human visual sensory reflexes with the Bayesian surprise (Itti and Baldi, 2009). Given a video of a dynamic natural scene, eye movements were recorded to determine where the test subject's attention was drawn to throughout. The Bayesian surprise metric was applied across the same videos, with the results showing that 72% of the human saccades were classed as being more surprising than average by the Bayesian surprise metric (Itti and Baldi, 2009). This shows the methods ability to describe visual surprise, which could be translated into a measure of surprise for experimentation, with experiments being chosen that maximise the expected surprise.

This formalism of surprise, could be utilised within an artificial experimenter. In their formulation, the surprise in Itti and Baldi (2009) is scaled by higher posterior probabilities, or confidence, giving preferences to surprising improvements in the posterior probability. However, in experimentation where hypotheses must be disproved, a surprising observation would be one that lowered the confidence of a previously highly confident hypothesis. Whilst looking for reductions in posterior probability may appear counter-intuitive, it is important to remember that successful refinement of those hypotheses will result in new hypotheses with higher confidences. Therefore, we interchange the prior and posterior terms to rework the Bayesian surprise function to be:

$$S = \sum_i C(h_i) \log \frac{C(h_i)}{C'(h_i)} \tag{5.7}$$

where $C(h)$ is the prior confidence of h before the experiment is performed, and $C'(h)$ is the posterior confidence of h after the experiment has been performed and includes the new experiment-observation pair in the evaluation. $C'(h)$ is calculated across all hypotheses under consideration before any new hypotheses are added. The function in Equation 5.3, is used to calculate the confidence.

Positive values of S states that the observation was surprising, as the overall confidence of the hypotheses have been reduced. Whilst a negative value states the observation was not surprising, as the overall confidence has increased. The result of S can therefore be used to control the switching between exploration and exploitation experiments, where a positive value will dictate that the next experiment will be exploitative, so as to allow investigation of the surprising observation. Whilst a negative value of S will lead to an exploration experiment next, to search for new surprising features of the behaviour.

The procedure for this experiment selection technique is as follows. The prior confidence of the current set of hypotheses before the experiment is performed, is compared with the posterior confidence of those same hypotheses after the experiment is performed, using the surprise function of Equation 5.7. If $S > 0$ then an exploitation experiment, the maximum of the discrepancy equation $D(x)$, will be performed on the next iteration. Otherwise an exploration experiment will be performed, which is the experiment that maximises $E(x)$. After S has been calculated, the hypothesis manager will go through

the process of creating new hypotheses. This process of evaluating experiments using surprise to choose the next experiment type, is continued until the maximum number of experiments allowed has been performed.

5.3 Discussion

By combining the hypothesis manager discussed in Chapter 3 with the experiment selection techniques discussed here, an artificial experimenter can be built. The artificial experimenter will begin by applying any prior knowledge. In this case no prior data is given, although prior assumptions about the types of behaviour and likely noise characteristics have been taken into consideration in creating the hypotheses and the hypothesis manager themselves.

Next a number of initial experiments must be performed. These initial experiments will allow the hypothesis manager to propose a starting set of hypotheses. When the artificial experimenter first starts there will be no hypotheses in consideration, meaning this initial set of hypotheses cannot be chosen through active learning and can be performed in batch. A sensible approach for placing these initial experiments is equidistantly across the parameter space. Equidistant placement of experiments should prevent the initial experiments from biasing one area of the parameter space, by ensuring an even distribution of experiments across the parameter space.

With a set of initial observations, the hypothesis manager can develop hypotheses about the available data using the techniques outline in Chapter 3. The experiment manager will then select the next experiment to perform using one of the techniques proposed in this chapter. In choosing the next experiment, the experiment manager will use the information contained within the hypotheses and the information about where the observations were previously positioned, to try and address the exploration–exploitation trade-off.

Once the experiment manager has chosen the next experiment to perform, the experiment will be conducted either manually, or the experiment could be performed automatically by combining it to automated laboratory hardware. The observation from that experiment will then be returned back to the artificial experimenter. With the design using a smoothing spline to represent each hypothesis in the 1-dimensional problem, the observation will be a single numerical value. Although the design could be updated to consider observations to be multi-parameter values. After the observation is returned to the artificial experimenter, the hypothesis manager will update the hypotheses under consideration and then the experiment manager will choose the next experiment to perform. This loop of updating the hypotheses, choosing the experiment and performing the experiment is repeated until a termination condition is reached. In this design the

termination condition is a pre-assigned maximum number of experiments that can be performed.

In the next sections we test this design for the artificial experimenter and evaluate the performance of the different experiment manager strategies.

Chapter 6

Evaluating in Simulated Scenarios

6.1 Problem Formulation

The biological domains of interest currently do not have significant documented behaviours that can be used to validate the techniques proposed. Therefore to evaluate the approaches presented, we consider a generalised problem that closely matches the target problem domain. First we assume that the true underlying behaviour exhibited by the biological system under investigation can be modelled by some function $f(x)$. The goal for the system is to build a function $g(x)$, which matches the response of $f(x)$. However, the responses from queries to $f(x)$ can be distorted by experiment measurement errors, causing noise to be applied to the responses (through ϵ). Additionally, the lack of control of the biological materials, also present distortions to the responses of $f(x)$. In enzyme experimentation, the reactants can undergo some undetectable physical or chemical change, which leads to experiments with those reactants yielding erroneous observations, unrepresentative of the true underlying behaviour. We model such instances as shock noise (through ϕ), which applies a large offset to the response variable. Whilst ϵ can occur on every experiment, ϕ will only be non-zero for a small proportion of experiments. We do not consider the case where ϕ occurs for a large number of experiments, as if this were the case all the results from such experimentation would be disregarded from consideration anyway for being too unreliable. We therefore represent a response characterisation experiment as:

$$y = f(x) + \epsilon + \phi \tag{6.1}$$

where parameter x and response y can be replaced with vectors to allow for multiple dimensions in both. The above equation differs from the one presented in Section 2.1, as the experiment parameter noise value δ has been removed. This parameter can be removed here for simplicity of presenting the results, because the underlying functions to

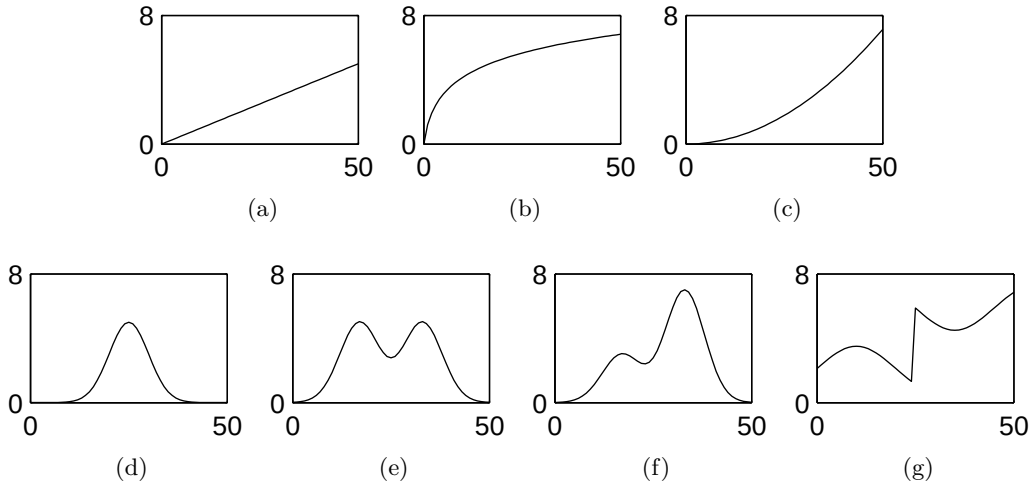


Figure 6.1: Underlying behaviours motivated from possible enzyme experiment responses. Experiment parameters on all x axes and observations on y.

be used have low variability in response within windows of the parameter space, meaning that the noise ϵ will perform a similar response shift to how δ would alter the values.

6.1.1 Underlying Behaviours

Whilst models of existing behaviours do not currently exist for the domain of interest, we can define some properties of those behaviours that may be expected or would be potentially useful for engineering with these biomolecules. In Figure 6.1, a range of underlying behaviours, f_a,...f_g, are presented, and described mathematically in Table 6.1. These behaviours test, in figure order (a–g): (a) linear response, (b) non-linear response, (c) power law, (d) single peak, (e) two peaks, (f) two peaks where one peak is dominant over the other, (g) discontinuity between two distinct behaviours. Behaviours (a-c) are motivated from expectations that behaviours are often described in terms of linear systems or power laws, where (b) is similar to Michaelis-Menton kinetics (Nelson and Cox, 2008) and (c) is similar to responses where there is a presence of cooperativity between substrates and enzymes (Tipton, 2002). Behaviours (d-g) are motivated from the belief that expected behaviours in the domain being investigated may be nonmonotonic and could also include a phase change between distinct behaviours (Zauner and Conrad, 2001a). We next discuss the implementation issues of the computational side of autonomous experimentation.

6.2 Method

Simulated experiments were conducted using the behaviours described in Section 6.1.1. All observations had additional Gaussian noise $\epsilon = N(0, 0.5^2)$. Experiments were

Figure 6.1	Function
a	$f_a(x) = \frac{x}{10}$
b	$f_b(x) = 4\log(x+1)$
c	$f_c(x) = \frac{x^2}{350}$
d	$f_d(x) = 5\exp\left(-\frac{(x-25)^2}{50}\right)$
e	$f_e(x) = 5\exp\left(-\frac{(x-17)^2}{50}\right) + 5\exp\left(-\frac{(x-33)^2}{50}\right)$
f	$f_f(x) = 3\exp\left(-\frac{(x-17)^2}{50}\right) + 6\exp\left(-\frac{(x-33)^2}{50}\right)$
g	$f_g(x) = \begin{cases} 3.5\exp\left(-\frac{(x-10)^2}{200}\right) & \text{if } x < 25, \\ 8 - 3.5\exp\left(-\frac{(x-35)^2}{200}\right) & \text{otherwise} \end{cases}$

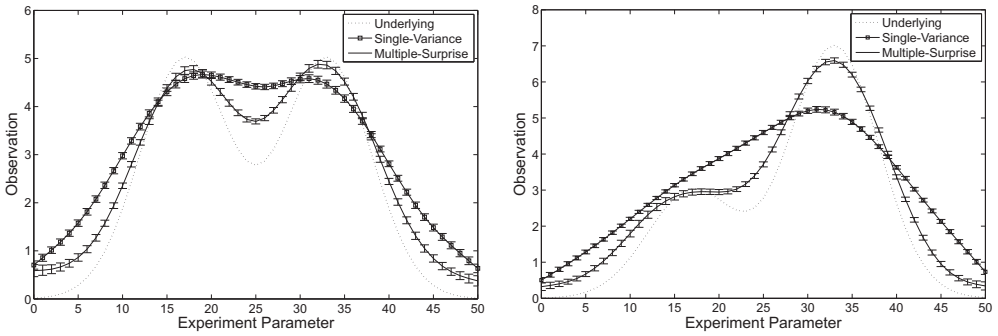
Table 6.1: Functions for the underlying phenomena shown in Figure 6.1.

bounded between zero and 50. To make the active experiment selection more tractable, the available experiment parameter settings were discretized evenly over this parameter space to provide a choice of 51 different experiments. Initially 5 exploration experiments were performed that were equidistant to one another in the parameter space. One of the initial experiments in each trial had random shock noise $\phi = N(3, 1)$ applied to it. After the exploration experiments were performed, hypotheses were created and active experiment selection began. The evaluation of the technique occurred over 15 actively selected experiments, where 3 of those experiments produced erroneous observations.

To contrast the multiple hypotheses technique, a single hypothesis approach was also tested. The single hypothesis used a smoothing spline created through cross-validation to determine the smoothing parameter, where all available observations were used. The single hypothesis had the same set of smoothing parameters that the multiple hypotheses technique had available to it ($\lambda \in \{10, 50, 100, 150, 500, 1000\}$). In the single hypothesis approach, experiments were selected through two methods. The two methods tested in the single hypothesis approach were: selecting experiments at the maximum variance of the smoothing spline; and random selection. Experiments were selected in the multiple hypotheses approaches by: random selection; choosing the experiment with the maximum discrepancy $D(x)$, from Equation 5.2; choosing experiments at the peaks of the discrepancy equation $D(x)$; and using the surprise technique proposed in Section 5.2.2.

To evaluate, the most confident hypothesis from each trial was used to determine whether the techniques could repeatedly provide a good representation of the underlying behaviour being investigated. This was calculated as the mean squared error between the most confident hypothesis and the true underlying behaviour:

$$E = \frac{1}{N}\sum_{n=1}^{N}\left(\hat{b}(x_n) - f(x_n)\right)^2 \tag{6.2}$$



(a) (b)

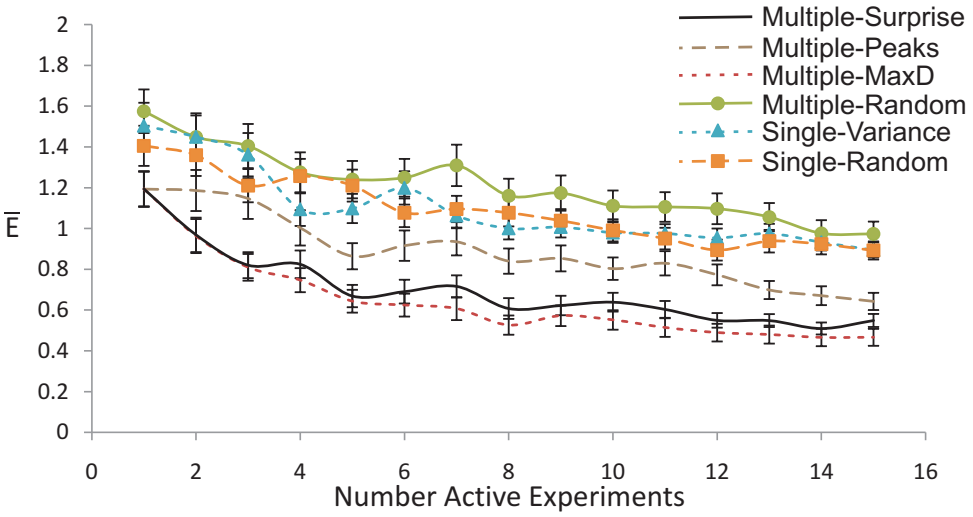
Figure 6.2: Comparison between the true underlying behaviour and the mean of the most confident hypotheses predictions for all trials, using the single hypothesis approach using predication variance experiment selection, and the multiple hypotheses approach using surprise exploration-exploitation switching for experiment selection. Shown using behaviour f_e (a) and f_f (b).

where $\hat{b}(x_n)$ is the prediction of the most confident hypothesis tested over N equally spaced experiment parameters x_n. The results over 100 trials for the 7 different underlying behaviours are shown in Figure 6.3.

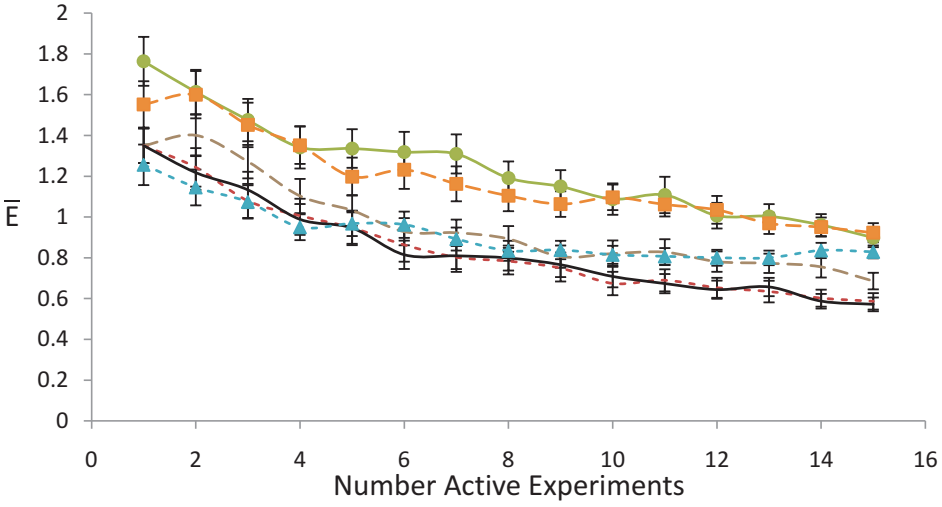
6.3 Results

Throughout, the single hypothesis techniques perform poorly in comparison to the multiple hypotheses techniques. Poor performance is due to the single hypothesis generally averaging through all of the data, which can result in features of the behaviours being missed, especially in the more complex nonmonotonic behaviours (d–g), as shown in Figure 6.2. In the monotonic cases, the difference in performance between the single and multiple hypotheses techniques comes from the single hypothesis averaging through all observations, including the erroneous ones, which allows the erroneous observations to affect the predicted responses, making the hypothesis less accurate.

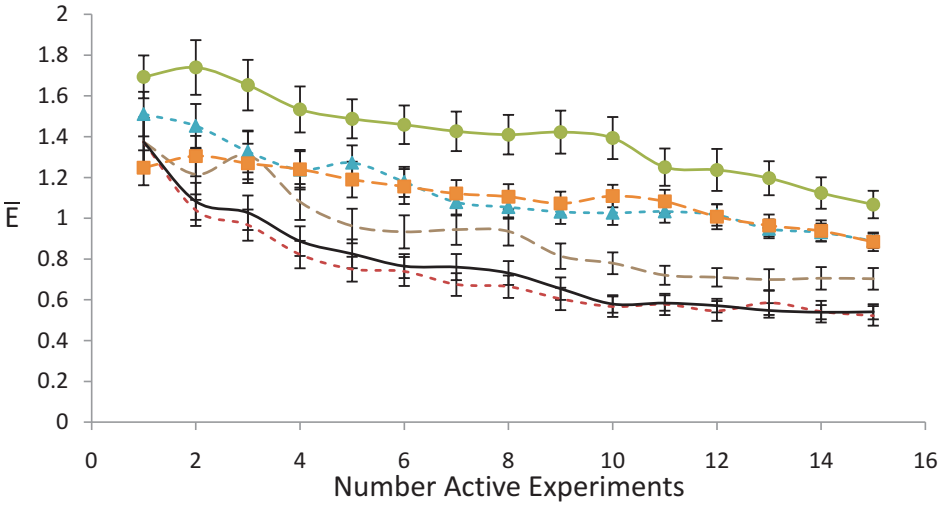
The multiple hypotheses techniques generally outperform the single hypothesis methods, however the extent of which is dependent on the active learning technique employed. The random strategy performs poorly in the monotonic behaviours (a–c), as experiments are not performed specifically to evaluate the accuracy of observations, which allows for the hypotheses to be misled by the erroneous observations. Whilst this is still an issue in the nonmonotonic behaviours (d–g), the random strategy will generally explore the parameter space more, so identifying the different features of the behaviour being investigated. This leads the random strategy with multiple hypotheses to have a lower error than the single hypothesis techniques and occasionally similar to the other multiple hypotheses techniques. The maximum discrepancy technique (MaxD) performs well in the simpler monotonic behaviours, as most of the differences between hypotheses will



(a)

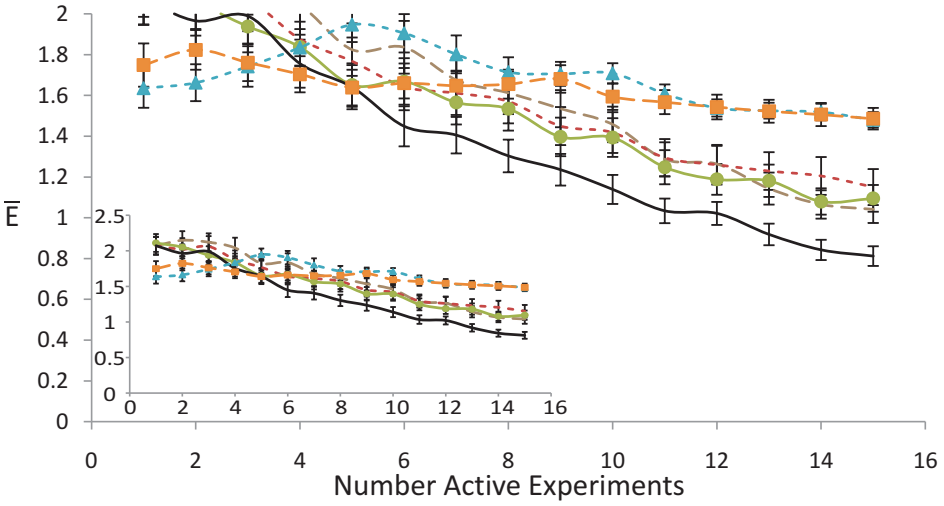


(b)

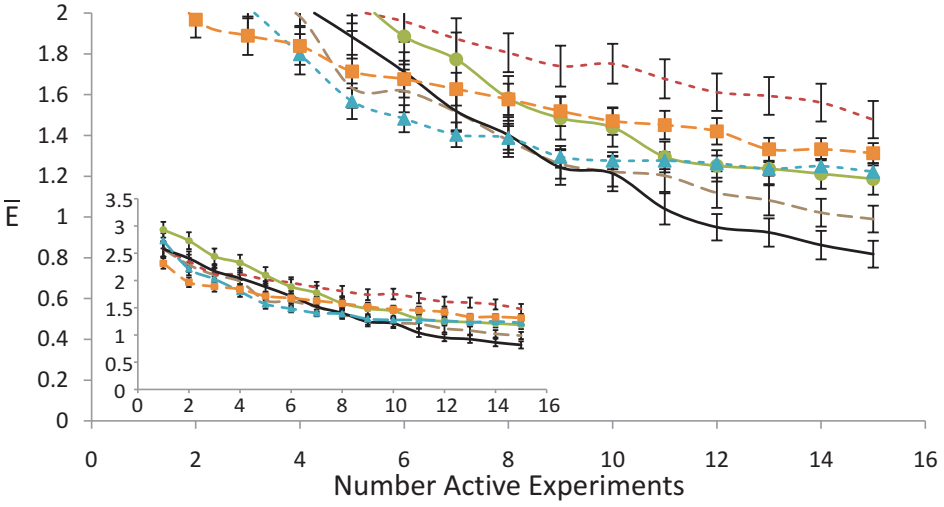


(c)

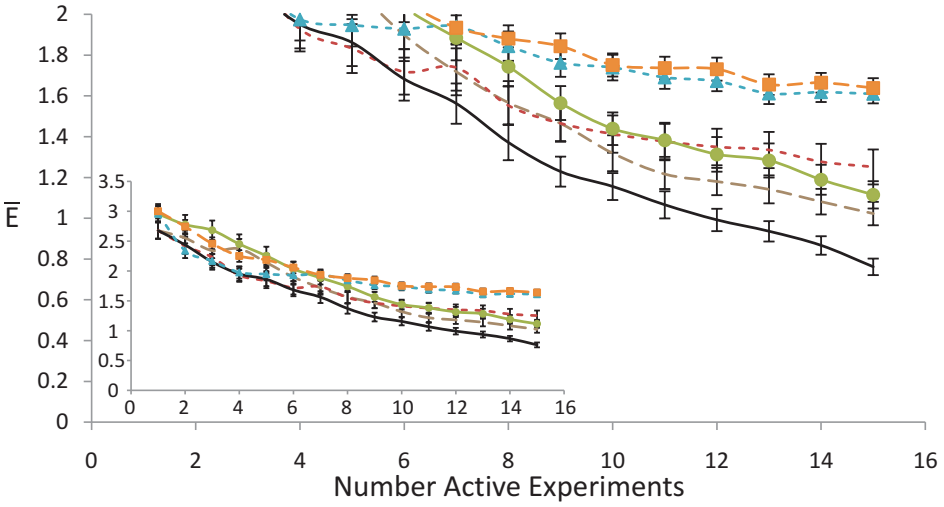
Figure 6.3



(d)

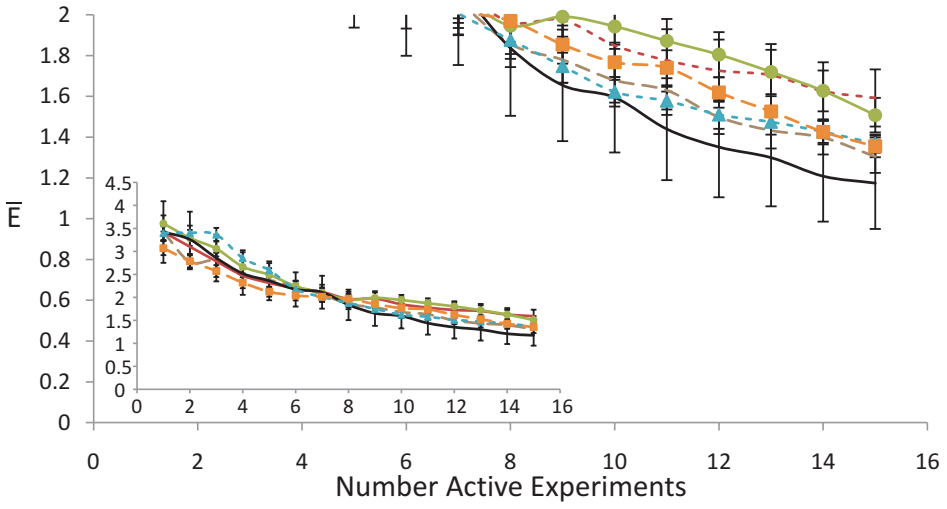


(e)



(f)

Figure 6.3



(g)

Figure 6.3: Performance of active learning and hypothesis management techniques for 1-dimensional problem. Shown is a comparison of error between the most confident hypothesis and the true underlying behaviour, over the number of actively chosen experiments, where 20% of the observations are erroneous, for 100 iterations. Shown in (a–g) are the corresponding results for the 7 behaviours shown in Figure 6.1.

be caused by erroneous observations, which the technique will investigate and be able to produce an accurate representation of the behaviour. In the monotonic behaviours however, the technique may miss some of their features, where its success in identifying the features is dependent on the initial exploratory experiments, as it will perform no exploration on its own and may become stuck investigating the same feature repeatedly. Using the peaks of the discrepancy equation provides more exploration of the parameter space than choosing just the maximum of the equation, allowing for lower error values in the nonmonotonic behaviours. However, in the monotonic behaviours the strategy may spend more experiments investigating small differences between the hypotheses than investigating erroneous observations, meaning that the resultant hypotheses are not as accurate as using the maximum discrepancy for these behaviours. The surprise technique performs consistently well for all behaviours tested. The surprise technique is able to evaluate the accuracy of the observations and suitability of the hypotheses through exploitation experiments, whilst performing a small number of additional exploratory experiments to further investigate the parameter space.

Over the 100 trials, the surprise technique used few exploration experiments per trial. The surprise technique used an average of 5 exploration experiments in the monotonic cases, normally in the latter stages of experimentation, and 4 exploration experiments in middle to latter stages for the nonmonotonic cases. As the hypotheses quickly produce a good representation of the underlying phenomena in the monotonic cases, additional exploratory experiments are performed, as the observations obtained are not surprising

to the hypotheses. If we allow the multiple peaks technique to have an additional 5 initial exploratory experiments but with 5 fewer actively chosen experiments, we find that it has a similar performance to the surprise method with only the 5 initial exploratory experiments, except for a significant improvement in predicting f_g by the multiple peaks technique. However, this is due to the initial 10 exploratory experiments covering all features of the behaviour. The surprise technique is therefore more preferable than the multiple peaks technique, as it has a lower initial exploratory experiment requirement. The surprise technique instead can decide for itself whether additional exploration is required. Additionally the technique could be adapted to further reduce resource usage, by automatically terminating experimentation after performing several unsurprising actively selected exploration experiments.

6.3.1 Statistical Significance

To further analyse the results, a two-tailed t-test with $\alpha = 0.05$ is used to determine if the results are statistically significant at the 95% confidence interval. For behaviours f_a, f_b and f_c, the results obtained by the multiple hypotheses technique using the surprise technique are clearly not significant compared to the multiple hypotheses technique that uses the maximum of the discrepancy equation. For the remaining behaviours, the surprise technique provides significant improvements over most other techniques. In Table 6.2, the surprise based experiment selection technique is compared to the random and multiple peaks experiment selection techniques, where multiple hypotheses have been used in all cases.

6.4 Two Dimensional Evaluation

In this section we investigate the performance of the best performing techniques from the 1-dimensional parameter space case, with 2-dimensional parameter space problems. The techniques tested are the multiple hypotheses approaches with random, discrepancy peaks and surprise based methods for experiment selection. The multiple hypotheses rules and experiment selection techniques are kept the same as previous, but for performance purposes 40 new hypotheses are created in each iteration, with the best 100 hypotheses being kept in each iteration. This time hypotheses are represented using a thin plate spline, as defined in Chapter A, with a choice of smoothing parameters ($\lambda \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$). All independent parameters are coded between 0 and 1, from behaviours with uncoded x_1 and x_2 parameters ranging between 0 and 50.

The underlying behaviours used are presented in Figure 6.4, where (a) provides a single feature, (b) a behaviour where only the x_2 factor provides a role in determining the response, and (c) a behaviour with two peaks and a trough. In each case the behaviours

Function	Technique	Active experiments with significant result
f_a	Random	all
	Peaks	3, 5–14
f_b	Random	all
	Peaks	12, 14, 15
f_c	Random	all
	Peaks	8, 10–15
f_d	Random	10, 11, 13–15
	Peaks	6, 8–15
f_e	Random	11–15
	Peaks	none
f_f	Random	3–15
	Peaks	4, 9, 12–15
f_g	Random	11–14
	Peaks	none

Table 6.2: Identification of statistically significant results in the 1-dimensional case. The results where there are significant differences between the multiple hypotheses surprised based experiment selection technique and the multiple hypotheses random experiment selection and multiple peaks experiment selection are shown. In all cases, a significant difference indicates that the surprise technique provides an improvement over the alternate technique. There are no cases of the surprise technique performing significantly worse.

are scaled between 0 and 8 on the dependent variable, so that the noise parameters ϵ and ϕ can remain the same for both the 1 and 2-dimensional cases. The underlying behaviours representing figures 6.4a, 6.4b and 6.4c respectively, are defined by:

$$f_1(x_1, x_2) = 8 \exp\left(\frac{-(x_1 - 25)^2 - (x_2 - 25)^2}{100}\right) \tag{6.3}$$

$$f_2(x_1, x_2) = 4\left(\sin\left(\frac{\frac{\pi}{2}(x_1 - 25)}{25}\right) + 1\right) \tag{6.4}$$

$$\begin{aligned} f_3(x_1, x_2) = 1.0 &+ 7 \exp\left(\frac{-(x_1 - 8)^2 - (x_2 - 8)^2}{160}\right) \\ &+ 3 \exp\left(\frac{-(x_1 - 33)^2 - (x_2 - 17)^2}{100}\right) \\ &- \exp\left(\frac{-(x_1 - 19)^2 - (x_2 - 36)^2}{140}\right) \end{aligned} \tag{6.5}$$

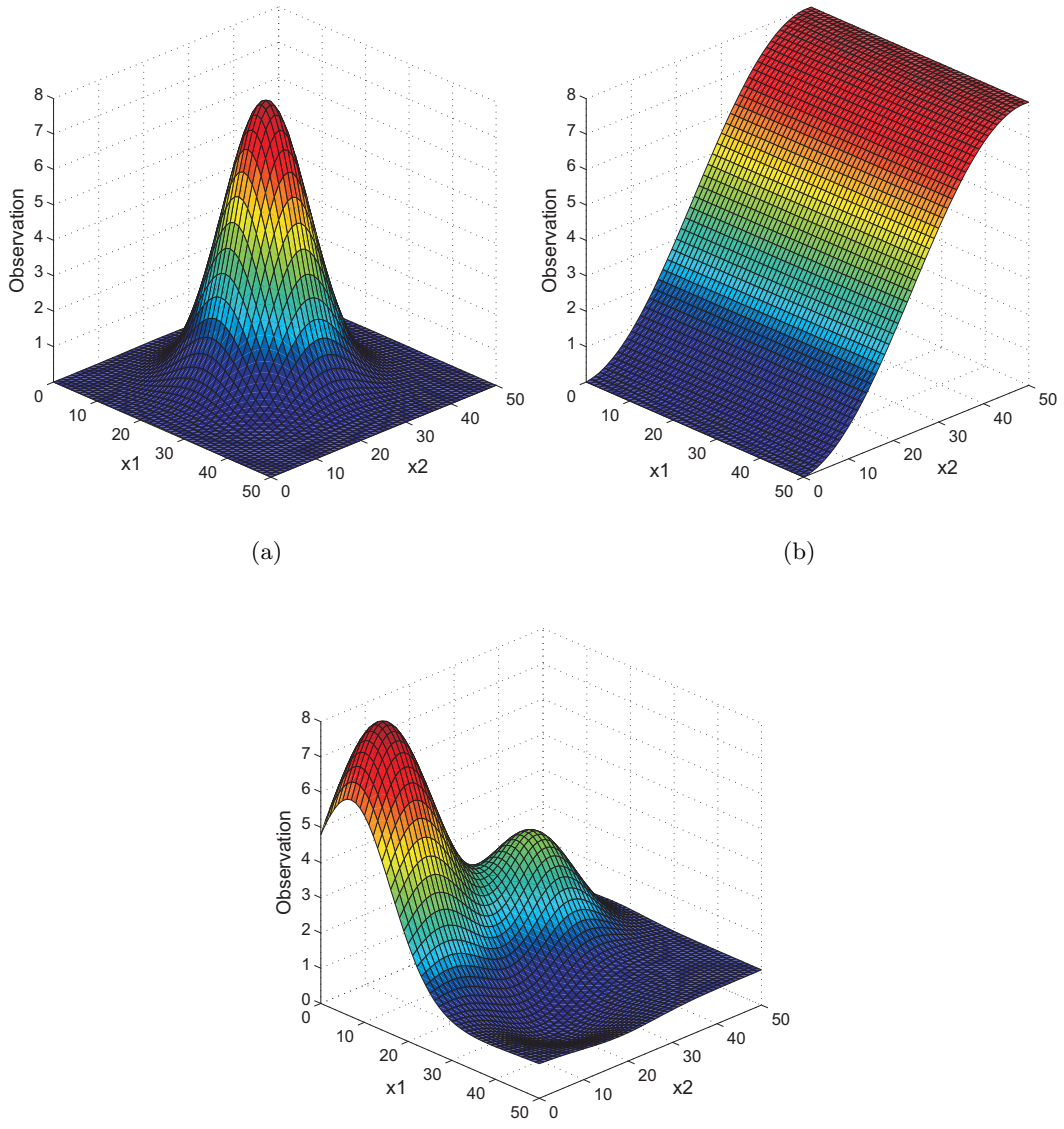


Figure 6.4: Underlying behaviours used for 2-dimensional trial. In (a) a single constant feature is present, corresponding to f_1. In (b) the observations depend only on the value of the x_2 factor, corresponding to f_2.. In (c) multiple features are present, corresponding to f_3.

6.4.1 Method

The surprise, discrepancy peaks and random experiment selection techniques were used, each replacing the single x used in the 1-dimensional case with a parameter vector containing x_1 and x_2. In each trial, 5 initial experiments were performed, which were equally spaced around the parameter space ([0,0], [1,0], [0,1], [1,1] and [0.5,0.5] in coded values). A further 25 actively chosen experiments were performed, where 3 of the experiments were erroneous. As in the 1-dimensional case, Gaussian noise was added to

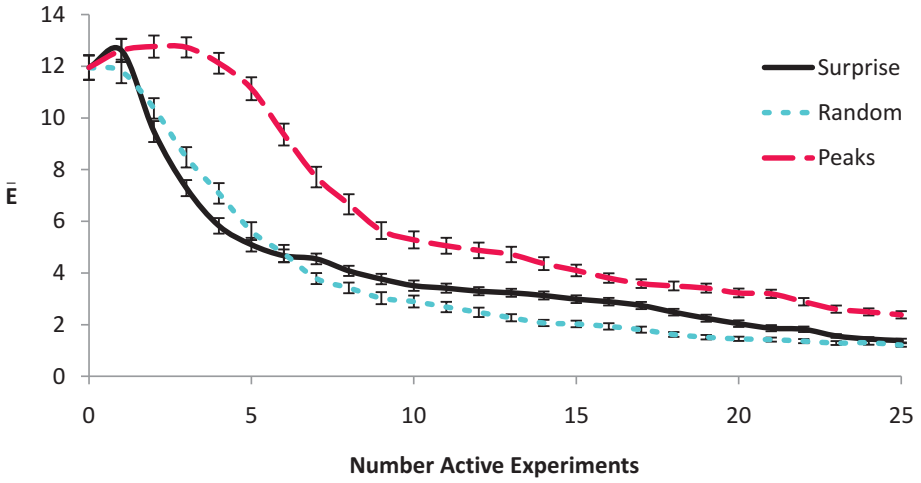
all observations with $\epsilon = N(0, 0.5^2)$ and the noise applied to an erroneous observation was $\phi = N(3, 1)$. The techniques were again evaluated by comparing a mean squared-error between the most confident hypothesis and the true underlying behaviour, following Equation 6.2.

6.4.2 Results

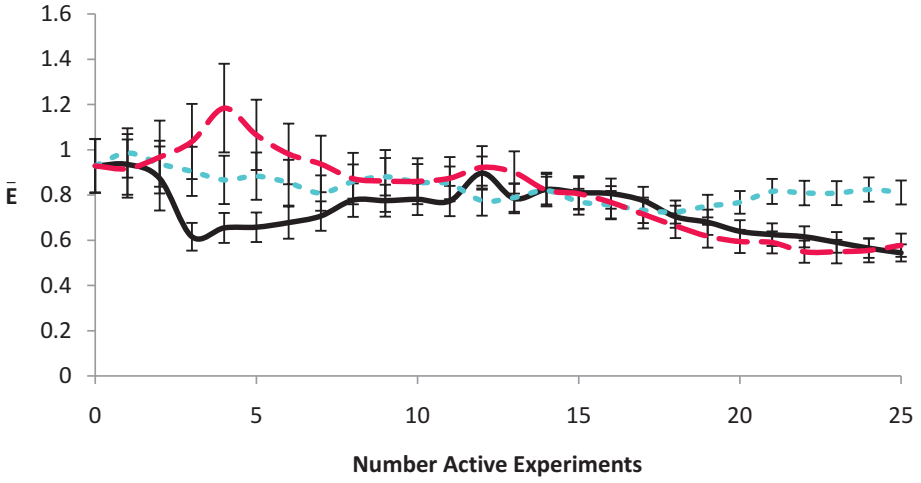
In the 2-dimensional problem, the results show there is less difference between the surprise and random experiment selection techniques than in the 1-dimensional case, whilst the discrepancy peaks technique again generally performs the worst, as shown in Figure 6.5. However, overall it appears that the surprise technique is still a more robust technique than the others considered, with the technique providing significant improvements over a random strategy in two of the three underlying behaviours.

For the single feature behaviour, f_1, the random technique outperforms the surprise technique between the 7th and 23rd active experiments, as shown in Figure 6.5a. During these experiments, the surprise technique spends more time investigating smaller differences between the hypotheses, causing a greater amount of exploitation early on than is perhaps necessary. Whilst the random technique is able to explore the parameter space more early on, allowing it to form a better general understanding of the behaviour quicker than the surprise technique. However, the random technique can still suffer here if it samples a region only once and obtains an erroneous observation there, which can cause it to include an additional feature in the behaviour not present in the underlying behaviour, as shown in Figure 6.6b. The discrepancy peaks technique performs poor throughout, by also over exploiting the information obtained rather than exploring. Over exploiting the information causes the discrepancy peaks technique to continually investigate small differences between the hypotheses, caused by the Gaussian noise applied to each observation. In Figure 6.6, a comparison of the most confident hypotheses after 25 active experiments is shown. In each case the error between the hypothesis and the true underlying behaviour is representative of the mean error shown in Figure 6.5a.

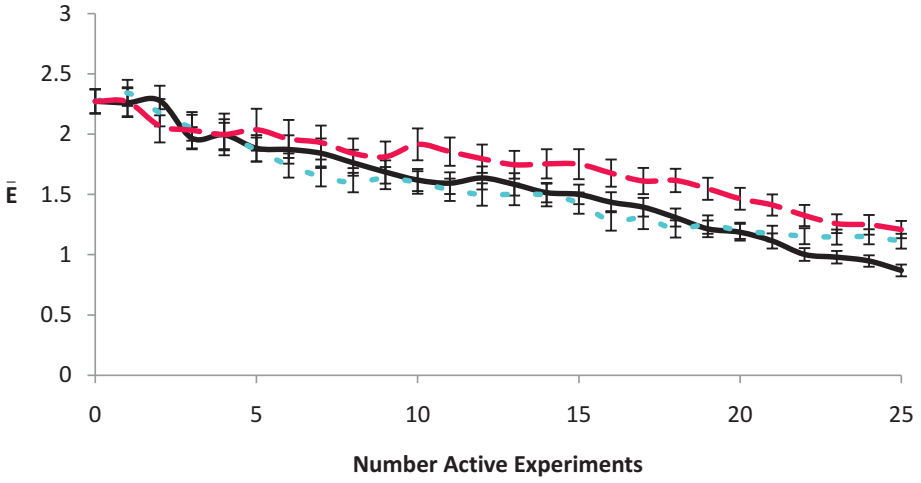
For the single factor behaviour, f_2, the initial 5 data points provide all of the techniques a good starting point. If those initial observations were given without error, then using only those observations the multiple hypotheses technique would be able to provide hypotheses that represent the underlying behaviour well, due to the simplicity of the behaviour. This means that this behaviour largely tests the ability of the experiment selection techniques to deal with erroneous observations in this 2-dimensional parameter space. The random technique fails to improve the performance of the most confident hypothesis throughout the 25 actively chosen experiments. In part this is caused by the technique not investigating erroneous observations, so any improvement made by understanding the simple behaviour is lost by the erroneous observations. The surprise technique performs well early on, as it is able to investigate erroneous observations.



(a)



(b)



(c)

Figure 6.5: Performance of active learning and hypothesis management techniques for 2-dimensional problem using a multiple hypotheses representation of the behaviours. Error bars shown are the standard error over 100 trials.

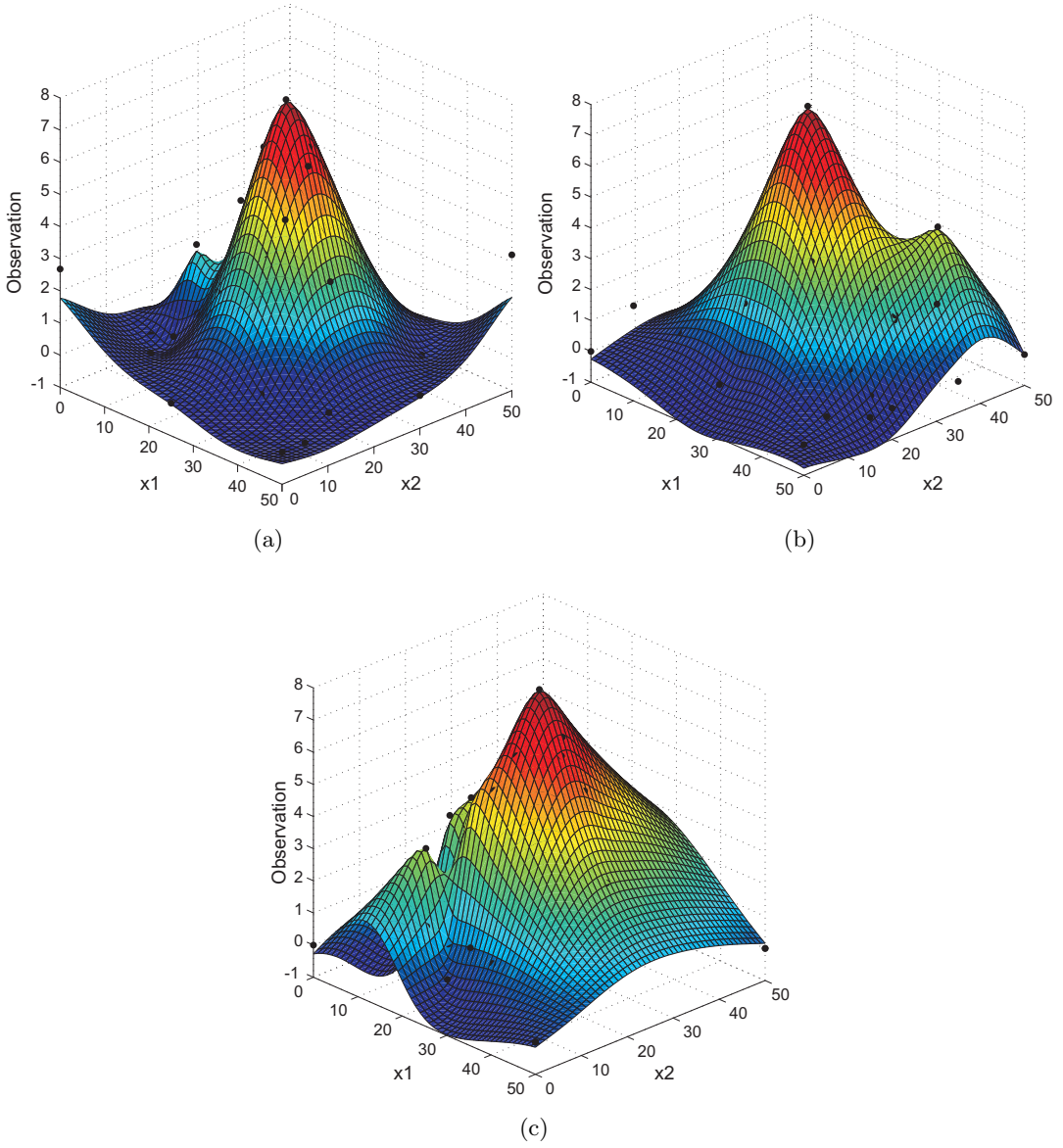


Figure 6.6: Representative illustration of the most confident hypotheses created for different experiment selection strategies on behaviour f_1. In (a) the result using the surprise experiment selection is shown, in (b) the random selection strategy is used and in (c) the discrepancy peaks strategy is used. In each case the error between the hypothesis shown and the underlying behaviour is representative of the mean error over 100 trials.

However the mean error increases again between 7 and 15 experiments because the technique over investigates some observations, causing the hypotheses to overfit some of the noise. The multiple peaks technique also suffers the problem of over sampling a region, causing lots of hypotheses with differences of opinion in a small area, which leads to hypotheses overfitting the noisy observations in those areas. However, over time the discrepancy peaks technique lowers the error to slightly below the surprise error in the latter stages of experimentation. By 25 experiments the performance of both techniques

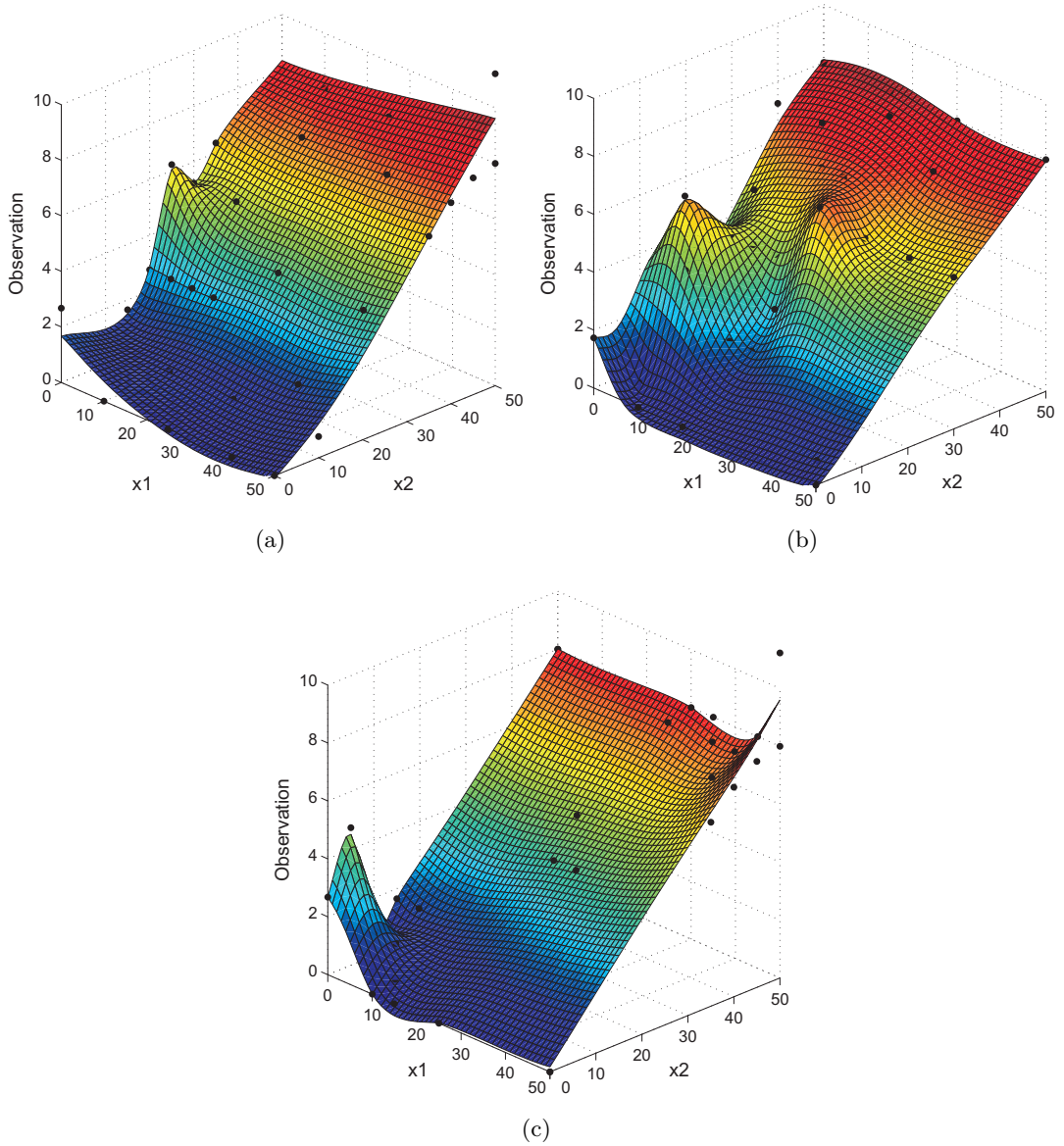


Figure 6.7: Representative illustration of the most confident hypotheses created for different experiment selection strategies on behaviour f_2. In (a) the result using the surprise experiment selection is shown, in (b) the random selection strategy is used and in (c) the discrepancy peaks strategy is used. In each case the error between the hypothesis shown and the underlying behaviour is representative of the mean error over 100 trials.

are nearly equal. In Figure 6.7, a comparison of the most confident hypotheses after 25 active experiments is shown. In each case the error between the hypothesis and the true underlying behaviour is representative of the mean error shown in Figure 6.5b.

For the behaviour with multiple features, f_3, the surprise and random techniques reduce the error at a similar rate for the first 18 active experiments. However after 18 active experiments the error for the random technique levels out, whilst it continues to reduce for the surprise technique. In the random technique the experiments are spread out

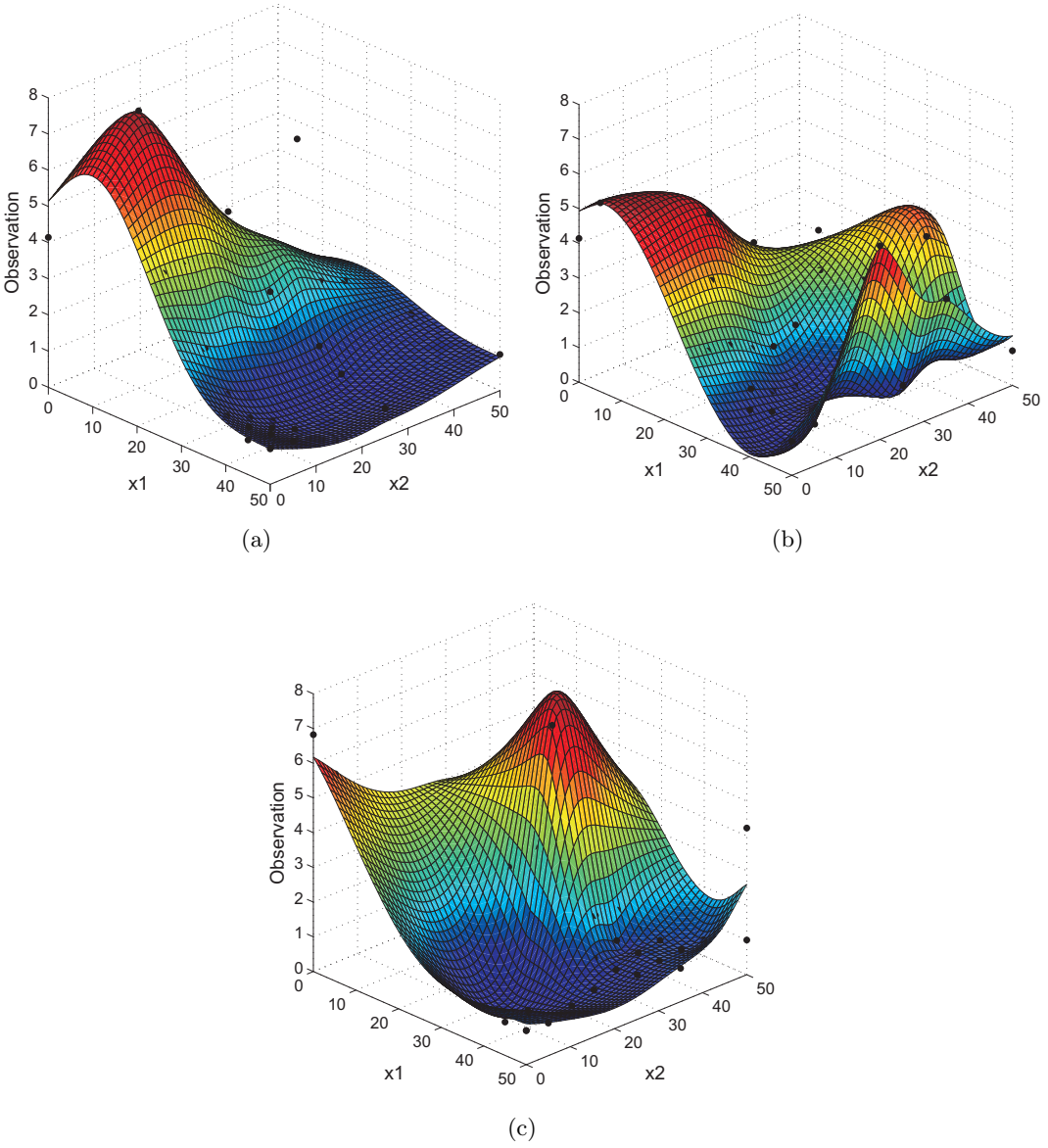


Figure 6.8: Representative illustration of the most confident hypotheses created for different experiment selection strategies on behaviour f_3. In (a) the result using the surprise experiment selection is shown, in (b) the random selection strategy is used and in (c) the discrepancy peaks strategy is used. In each case the error between the hypothesis shown and the underlying behaviour is representative of the mean error over 100 trials.

across the parameter space, allowing for the different features to be identified quickly, albeit at a low resolution. However, as the experiments are not directed, increasing the understanding of any particular behaviour is by chance and potentially erroneous observations are ignored. These two factors prevent the error from reducing further later on in experimentation when compared to the surprise technique. The surprise technique through performing exploit experiments, performs more experiments near the features

it discovers, causing better representations of the behaviour to be formed. Additionally, the technique is able to investigate and identify erroneous observations, whilst also performing experiments to further search the parameter space. The discrepancy peaks technique performs worse than the other two techniques, because it over exploits and becomes focussed in particular regions of the parameter space where the first unexpected behaviours were obtained. Unlike the other two behaviours where all techniques provided a somewhat representative representation of the underlying behaviour, in this example the surprise based technique is the only technique to provide a good representation of the underlying behaviour, as shown in Figure 6.8, where hypotheses are representative of the final mean error.

6.4.3 Statistical Significance

Like the 1-dimensional case, the results for the 2-dimensional parameter space have been analysed using a two-tailed t-test with $\alpha = 0.05$ to determine if the results are significant at the 95% confidence interval. The surprise technique provided significant improvements over a random selection strategy for behaviours f_2 and f_3, although only in the latter stages of experimentation. These improvements are in part due to the surprise technique being able to better identify erroneous observations than the random technique. Additionally the surprise technique is able to investigate the new features it discovers further, which allows it to provide a better representation of the more complex behaviour f_3. The random strategy performs significantly better than the surprise technique for the majority of the experimentation performed using behaviour f_1. This is due to the random technique being able to explore the parameter space more, where the surprise technique spends some additional time investigating small differences between the hypotheses that only provide small benefits for developing a representation of the behaviour. The discrepancy peaks technique performs significantly worse than the surprise technique for behaviours f_1 and f_3 by the end of the experimentation performed. However, the technique performs similarly well compared to the surprise technique for the behaviour f_2, due to little exploration being required, which the technique is poor at providing. In Table 6.3, the surprise based experiment selection technique is compared to the random and multiple peaks techniques, where multiple hypotheses have been used in all cases.

6.5 Conclusions

Presented here are simulated evaluations of artificial experimenter designs, using underlying functions influenced by potential behaviours that could be exhibited by the biological systems of the target domain.

Function	Technique	Active experiments with significant result
f_1	Random	3, 4, (7–23 surprise is significantly worse)
	Peaks	2–25
f_2	Random	3, 21–25
	Peaks	3–6
f_3	Random	23–25
	Peaks	18–25

Table 6.3: Identification of statistically significant results in the 2-dimensional case. In each case a comparison is made between the surprise technique and the one stated. All statements show where there is a significant difference between the results. In all cases, except where stated, the surprise technique provides a significant improvement over the alternate technique.

A comparison between single and multiple hypotheses methods have been made, where the multiple hypotheses technique has been shown to consistently outperform the single hypothesis technique. Our belief is that the uncertainty that exists within this problem, is best dealt with through a multiple hypotheses approach. In such an approach, decisions about the validity of observations can be delayed until more experimental evidence is available, through competing hypotheses with different views about the validity of the observations. These multiple hypotheses can be used for effective response characterisation when coupled to an active learning technique, which will outperform a single hypothesis based approach.

The effectiveness of the multiple hypotheses technique depends on the active learning technique it is applied to, where a random selection strategy can perform poorly. In the 1-dimensional case, a technique based upon Bayesian surprise has been shown to repeatedly outperform all other techniques. This surprise technique is successful because it provides an effective management of the exploration vs. exploitation trade-off, by exploring when observations are obtained that the hypotheses could already predict and exploiting when the observations conflict with the hypotheses. The effective management of the trade-off is in contrast to the random technique that uses the majority of its resources exploring the parameter space, causing it to not obtain data that can be used to refine hypotheses in regions where unexpected behaviours have been obtained. The discrepancy peaks technique can perform poorly by over exploiting the parameter space, causing it to miss features of the behaviour being investigated.

The maximum discrepancy peaks technique considered in the previous chapter rarely outperform the alternate techniques. In this technique all experiments are selected to provide some exploration and exploitation. However it appears that the trade-off it provides is insufficient for the particular problem, likely because all experiments are a compromise between explore and exploit.

Expanding to higher dimensions, the surprise technique can still provide a significant improvement over the alternate techniques by the end of experimentation with 25 active

experiments in some cases. However the degree of benefit is less than in the 1-dimensional case. A limitation of the surprise technique in the higher dimension examples tested, is that it performs little exploration early on. As the thin plate splines considered in parallel have larger variations with noisy data than the 1-dimensional data, the surprise technique is more inclined to exploit especially early on, where much of the data is surprising. This can lead to some of the early observations being focussed within a particular area. However, unlike the discrepancy peaks technique that can also focus the placement of experiments within a small area, the surprise technique does adapt over time and will explore the parameter space, leading to it providing more accurate hypotheses than the other techniques later on in experimentation in most cases. The surprise technique could benefit from additional initial exploration, however care would have to be taken to ensure that this exploration does not bias particular regions of the parameter space.

A potential weakness in the results presented here is the way that they are expressed. Whilst the error term used is a standard squared error between the most confident hypothesis and the true hypothesis, making a comparison between techniques based on this value can be difficult, especially when the error value is large, as the hypotheses being compared may not be representative of the underlying behaviour. This can be seen in the 2-dimensional results, particularly with behaviour f_3 where the surprise technique ends with a mean error of 0.87 and the random technique ends with 1.11, but the representative hypotheses for these error rates shown in Figure 6.8 clearly show a big difference between the representativeness of the hypotheses. Therefore, whilst the evaluation technique provides a reasonably good indication of learning performance and the rate of error reduction indicates which techniques should provide a representative hypothesis first, when the error rate is large, typically greater than 1 in the results here, making inferences about the best technique on error rate a time t alone can potentially lead to trying to choose the best technique where no technique is capable of providing a representative hypothesis. Further work could be spent trying to identify more suitable quantitative evaluation techniques, however the current evaluation technique may still be preferable for simplicity, so long as a qualitative check of the hypotheses is made to understand what the error threshold is between unrepresentative hypotheses and representative ones.

In the next chapter the use of these techniques within a laboratory setting is presented.

Chapter 7

Evaluating in Laboratory Scenarios

Further to the simulated evaluation, the artificial experimenter has been tested within a laboratory setting to validate the techniques can work with a real physical environment. In this chapter we will evaluate a 1-dimensional experiment parameter space in a laboratory setting. The 1-dimensional parameter space evaluation characterises the optical absorbance properties of the co-enzyme NADH, where a theoretical prediction of the response curve can be used to compare to. The artificial experimenter guides a human experimenter by telling them the experiments to perform. The human experimenter performs the requested experiments, returning the observational values to the artificial experimenter, which then analyses them and determines the next experiment to perform.

7.1 Characterisation of NADH Response

The co-enzyme NADH is commonly used for monitoring enzymatic catalytic activity, making it a component of enzyme activity characterisation (Matsumaru et al., 2002). On its own, the optical absorbance profile of NADH can be characterised, with the benefit of being comparable to a theoretical value obtained from the Beer-Lambert law (Nelson and Cox, 2008). However the optical absorbance profile is known to be non-linear, so the resulting observations will not be entirely comparable to the theoretical model. This makes the characterisation of NADH an attractive evaluation for the artificial experimenter, as there is an element that can be compared to a theoretical value, whilst there is also a non-linear component that will test the flexibility of the technique with realistic behaviours, which is further improved with real experimental noise.

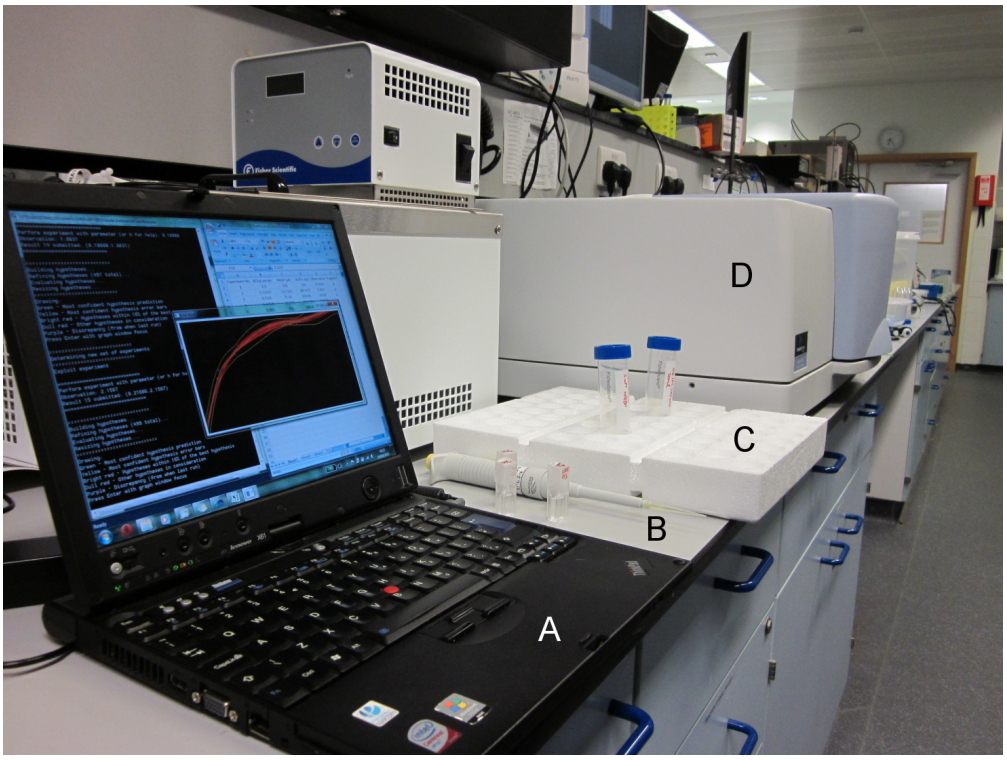


Figure 7.1: Photo of laboratory set-up. A laptop running the artificial experimenter software (A) requests experiments to be performed by providing the concentrations of NADH to test. The required concentration of NADH is mixed in cuvettes (B) using stock solutions of 5mM NADH and 10 mM Tris buffer (C). After mixing by inversion, samples are placed in the spectrophotometer (D), which gives an absorption reading for a wavelength of 340 nm. This reading is given back to the artificial experimenter as the observation for that experiment.

7.1.1 Materials and Methods

The experiment parameter space for the artificial experimenter was the concentration of NADH. Experiments could be chosen from a predetermined range of concentrations, from 0.001–1.5 mM. The parameter space was coded to the parameter space used in the simulated evaluation, allowing for same set of smoothing parameters to be used ($\lambda = \{10, 50, 100, 150, 500, 1000\}$).

Prior to any experiments being performed, a stock solution of 5 mM NADH and a 10 mM Tris buffer at pH 8.5 were prepared. Additionally, a PerkinElmer Lambda 650 UV-Vis Spectrophotometer was calibrated to provide optical absorbance readings at 340 nm. The photometric range of the spectrophotometer was 6 A (PerkinElmer, 2004).

The artificial experimenter requested an initial 5 experiments, chosen equidistant across the parameter space. These and all subsequent experiments, were carried out by the human experimenter, using the apparatus shown in Figure 7.1. The procedure for conducting the experiments requested was as follows. The artificial experimenter would

state the concentration of NADH to be tested. The human experimenter created the requested NADH dilution by mixing volumes taken from the stock solution and the buffer in a cuvette. The cuvette was placed in the spectrophotometer and a measurement of optical absorbance was recorded at 340 nm. The absorbance measurement was taken as the observation for that particular experiment and was submitted to the artificial experimenter by the human experimenter. The artificial experimenter would then generate a new set of hypotheses, in the same manner used in the simulated evaluation. The artificial experimenter then presented a graph of the observations, along with the current best hypothesis, the alternate hypotheses and the discrepancy amongst them. Using the experiment selection algorithm selected for the evaluation, the artificial experimenter actively chose the next experiment to perform. A total of 10 actively selected experiments were allowed, bringing the total to 15 experiments performed for the evaluation. The maximum discrepancy peaks and surprise techniques were used in separate trials to perform the experiment selection.

7.1.1.1 Beer-Lambert Law

The Beer-Lambert law allows predictions to be made about the absorption of light through a material it is travelling through. In this case the light is from the spectrophotometer, providing a UV-light with a wavelength of 340 nm, whilst the material is the NADH being characterised.

The law takes the form of the equation:

$$A = \epsilon c l \tag{7.1}$$

where A is the absorbance, ϵ is the extinction coefficient for a particular wavelength of the material, c is the concentration of the material, and l is the path length, or distance the light travels through the material. For NADH, the extinction coefficient is obtainable from the manufacturer data sheet, where $\epsilon = 6.22\ \mathrm{M}^{-1}\mathrm{cm}^{-1}$ for a wavelength of 340 nm (Sigma-Aldrich, 2010). The path length is 1 cm, and the concentrations will range across the experiment parameter space selected. The absorbance is a dimensionless quantity, so has no units.

7.1.2 Results

In the following we discuss the results from a laboratory trial using the maximum discrepancy peaks experiment selection method and the surprise based explore-exploit switching strategy. In each we discuss the predicted characterisation made by the most confident hypothesis after each experiment.

7.1.3 Maximum Discrepancy Peaks

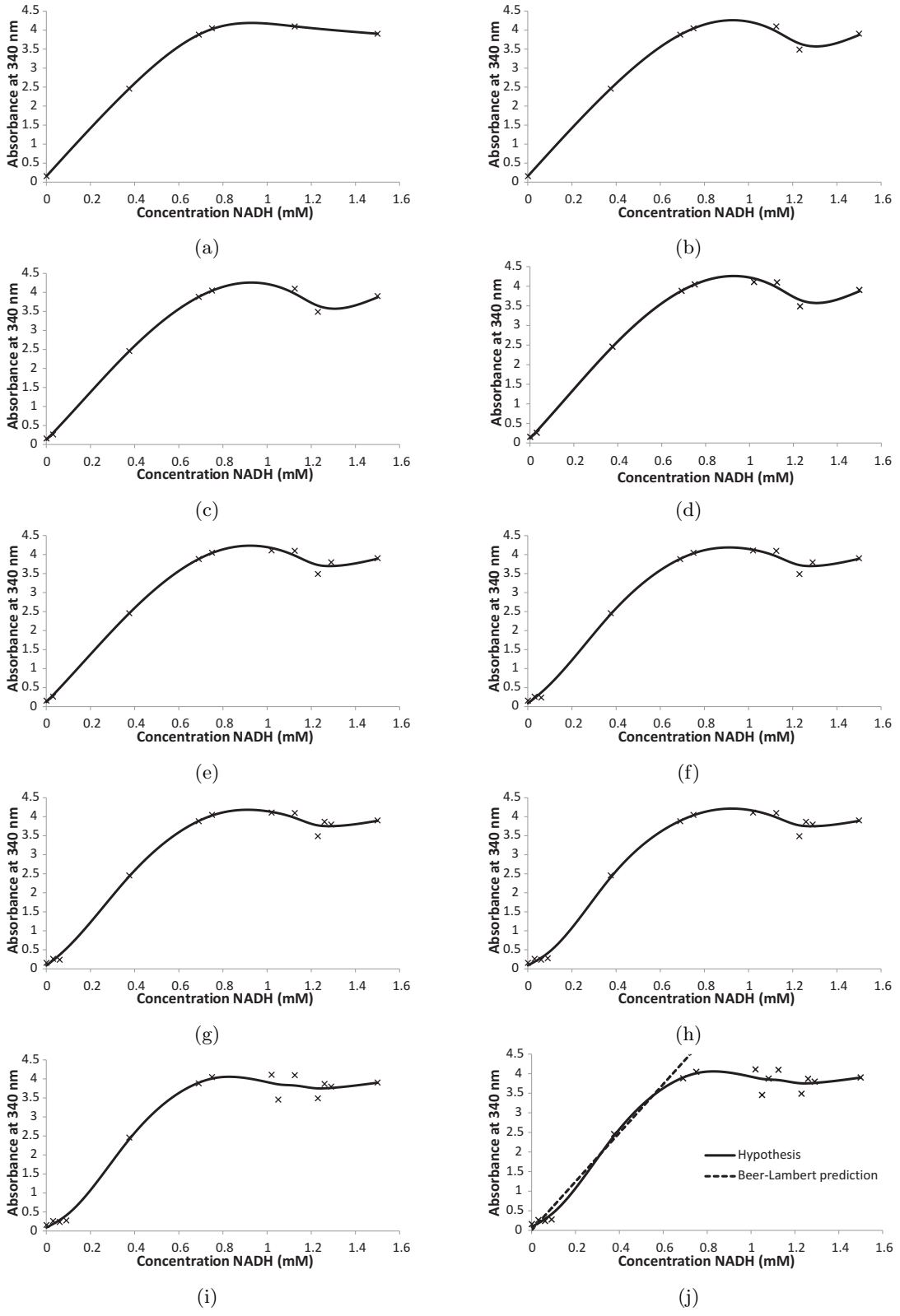


Figure 7.2: Most confident hypothesis over 10 actively chosen experiments for the maximum discrepancy peaks experiment selection technique.

In Figure 7.2, the most confident hypothesis after each actively selected experiment was performed are shown, where the maximum discrepancy peaks method has been used to perform experiment selection. After the initial exploratory experiments, the artificial experimenter identifies the key feature that there is an increase in absorbance between 0.001 and 0.75 mM, that then begins to level out, as shown in (a). This first actively chosen experiment, placed roughly where the increase in absorbance ends at 0.69 mM, provides an observation that agrees with the initial trend found in the data.

The second active experiment at 1.23 mM, provides an observation lower than the initial prediction. This observation causes the artificial experimenter to consider the possibility the behaviour of the absorbance is to decrease, rather than level out as concentration increases. The majority of the remaining experiments then investigate whether the absorbance levels out or if it drops again in this region, except for a few experiments that are placed in the minimum experiment parameter region. The small number of experiments placed in the minimum of the parameter region are caused by the discrepancy equation having a small peak at the minimum of the experiment parameter space. The experiments that test the right hand region where the absorbance appears to lower again over increased concentration, obtain observations that are more noisy, causing repeated experiments to be performed. In this right hand region, the most confident hypothesis detects that the observations are noisy and produces a hypothesis that is roughly linear and flat through that region.

After the 10 actively chosen experiments have been performed (15 experiments in total including the initial 5 exploratory experiments), we can see that the most confident hypothesis matches the Beer-Lambert law rate of change in absorbance prediction, using the indicated coefficient of 6.22 $M^{-1}cm^{-1}$ at a wavelength of 340 nm (Siegel et al., 1959), as shown in Figure 7.2j. The hypothesis then goes through a non-linear region to predict that there is a relatively flat region in the higher concentrations, which is what would be expected.

7.1.4 Surprise Explore-Exploit Switching

In Figure 7.3, the most confident hypothesis after each actively selected experiment was performed are shown, where the surprise explore-exploit switching technique has been used to perform experiment selection. With the first actively chosen experiment, an exploitation experiment was performed to investigate the non-linear component of the behaviour, as shown in (a). As this observation agrees with the more confident hypotheses and the rest of the data indicates near linear regions either side of the behaviour, the experiment selection strategy performs an exploration experiment to look for uncharacterised behaviours. This exploration experiment, shown in (b), obtains an experiment with an absorbance slightly higher than the predicted values from the

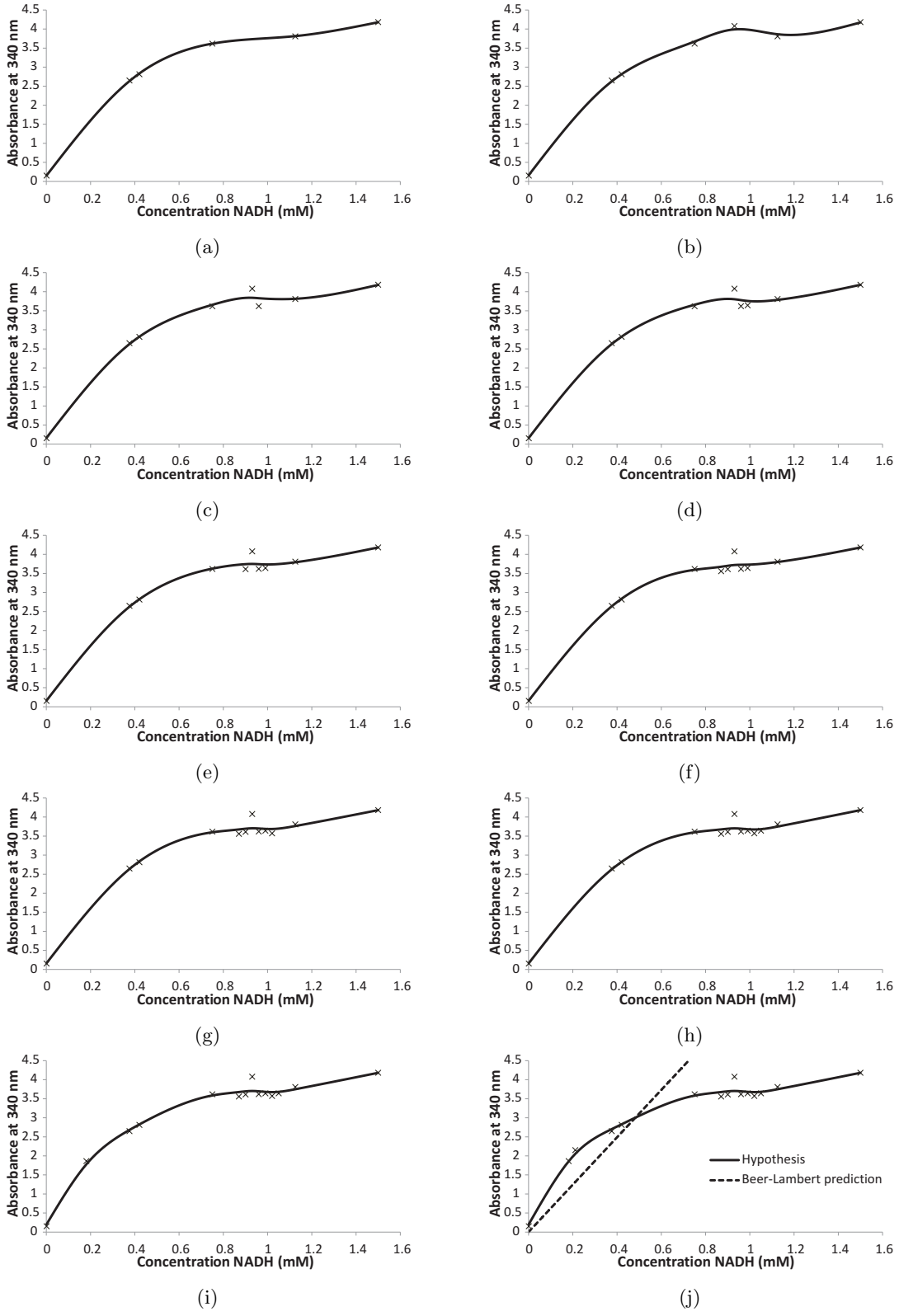


Figure 7.3: Most confident hypothesis over 10 actively chosen experiments for the surprise explore-exploit switching experiment selection technique.

Active Experiment No.	Explore or Exploit
1	Exploit
2	Explore
3	Exploit
4	Exploit
5	Exploit
6	Exploit
7	Exploit
8	Exploit
9	Explore
10	Exploit

Table 7.1: Listing of whether surprise explore-exploit switching technique chose an exploration or exploitation experiment to perform.

hypotheses, which makes the observation surprising. The next 6 experiments (c–h) continue to investigate that area of the parameter space.

Once the observations in the region investigated by the 2nd–7th actively chosen experiments again become unsurprising, the experiment selection technique performs another exploration experiment (i), that investigates the lower concentrations. This exploration experiment obtains an observation that is higher than the current linear view of the region, causing an exploitation experiment to be performed in the next iteration. The observation obtained by this final experiment (j) matches the previous experiment, causing the most confident hypotheses to have a higher rate of change than the Beer-Lambert prediction. Where as earlier hypotheses had a prediction that followed much closer to the Beer-Lambert law for the initial increase in absorbance over concentration of NADH. The final two observations are likely to be erroneous, but because they agree with each other the artificial experimenter has no reason to declare them erroneous, meaning that the final hypothesis is slightly worse than the hypothesis formed by the discrepancy peaks experiment selection strategy. In Table 7.1, a listing of whether each experiment was an exploration or exploitation experiment is given.

7.1.5 Discussion

In a comparison of the two strategies, both managed to choose experiments that examined key aspects of the behaviour under investigation, leading to good hypotheses being generated. However the maximum discrepancy peaks strategy wasted a number of experiments repeatedly investigating a feature of the behaviour that did not provide new information. Both strategies first investigated the non-linear component of the behaviour and found that the change in behaviour was valid. Both strategies also found a noisy region of the parameter space between 0.8 and 1.2 mM and spent a number of observations understanding the behaviour observed there. However, whilst the surprise explore-exploit strategy performed experiments in that region before moving

on to explore a new region of the parameter space to test, the maximum discrepancy peaks technique repeatedly investigated this region. These repeated experiments by the maximum discrepancy peaks strategy can be seen as a failure in the strategy, as the observations obtained were similar and added little to the final hypothesis. Where as the surprise explore-exploit strategy used its final two experiments to first further explore the parameter space, which produced a surprising observation, leading it to perform a further experiment to investigate why that experiment was surprising.

Despite the above criticism, the maximum peaks discrepancy strategy was able to produce a hypothesis that represented the Beer-Lambert theoretical better value than the surprise method after the 15 experiments had been performed. In the part of the behaviour where the absorbance increases with increased concentration, the maximum peaks discrepancy strategy is nearly identical to the theoretical value. Whilst the surprise based explore-exploit switching technique provides an apparent two phase increase in absorbance over concentration. However, the inaccuracies of the surprise based explore-exploit switching technique can be identified as being caused by experimental observations that do not match the theoretical value. With limited resources and noisy potentially erroneous observations, such occurrences will occur, and if they occur when few resources are available to evaluate them, such errors will occur. Importantly, the surprise based explore-exploit switching technique did perform an additional experiment to test the first experiment near the 0.2 mM concentration, which obtained an observation that appears to confirm the first observation. The difference between the surprise explore-exploit switching most confident hypothesis and the theoretical value, are therefore caused by the complexity of working with physical systems, which do not always match the theoretical values, even when repeat experiments are performed.

Chapter 8

Conclusions

Automating the methods and processes used by scientists to successfully navigate and understand a parameter space through laboratory characterisation, poses interesting challenges for machine learning. With resources being a limiting factor on what can be learnt, techniques are required that learn with only a small number of observations. Additionally, the unpredictability of physical experimentation, particularly in biological experimentation, mean that the observations are not only noisy but may also misrepresent the behaviours actually being exhibited by the system under investigation. These limited resources mean that standard techniques of performing the same experiment several times to get a better representation of the underlying behaviour, are not particularly suitable, as they will direct resources away from exploring the system under investigation. Instead techniques are needed that take into consideration the questionable validity of observations, albeit with a minimal amount of supporting evidence to show either validity or invalidity.

This thesis has presented an artificial experimenter that is capable of learning from limited and potentially erroneous observations, using a principled multiple hypotheses approach, combined with a method for data selection that uses a surprise based metric to control the exploration vs. exploitation trade-off. It is important to note that the techniques developed here are not designed to return highly in-depth and accurate hypotheses, as the number of experiments allowed are too few. Rather they are designed to return a general overview of the system being investigated, with key features of that behaviour along with erroneous observations identified. For the problem domain being used to motivate the work, the characterisation and identification of biomolecular substrates that may have properties that could be exploited for biomolecular computation, this the type of output is exactly what is required. It allows a cheap investigation of a parameter space, identify interesting features, such as peaks, troughs or areas of discontinuity, and returns hypotheses that a human scientist can use to determine if it is worth while using further resources to get a deeper understanding of the parameter space.

This thesis has highlighted and addressed common problems faced in experimentation, which have not been appropriately addressed in the related knowledge discovery domains of computational scientific discovery and active learning. First there is the problem that data is extremely expensive to obtain experimentally, meaning that the number of observations available will be very small. Many computational scientific discovery systems do not actively seek to minimise the number of experiments performed, as the goal for these systems is to demonstrate the principle that computational systems can discover new knowledge. Active learning is based upon the notion of learning from the minimal set of observations, however many problems currently addressed by active learning allow for far larger sets of training data than would be available from resource restricted physical experimentation. The second problem is the notion of erroneous observations, caused by some failure in the experiment. Again many computational scientific discovery and active learning systems do not consider this problem. In some cases previous techniques assume either there is no noise in the problem. With physical experimentation, erroneous observations is a fundamental problem. If a computational system assumes all data to be valid within some margin or error caused by Gaussian noise or similar, then a single erroneous observation will corrupt the hypotheses formed and result in a poor understanding of the behaviour being investigated. Systems that took this approach of assuming all data to be valid would not be accepted for laboratory use by any experimenter. As such, these knowledge discovery domains have much scope for development and there are many real-world problems that they could be applied to. By further addressing the key problems faced by laboratory experimenters, robotic laboratory equipment can feasibly evolve from the mechanical automated systems that can automatically perform requested experiments, to fully autonomous discovery machines that can also decide on the experiments to perform and guide the discovery process.

The contributions of this thesis fall in three parts. First that utilising multiple hypotheses allows for small sets of observations with questionable validity to be learnt from more effectively than standard single hypothesis techniques. Second, that using the variance of hypotheses predictions to select experiments that differentiating between them is suboptimal and an alternate technique has been shown to be more robust. Finally, that a Bayesian formulation of surprise can be used to effectively manage the exploration vs. exploitation trade-off for experiment selection. These are summarised in more detail in the following section.

8.1 Summary of Work

Using multiple hypotheses in scientific thinking, is a technique promoted by philosophers of science, as discussed in Chapter 2. Multiple hypotheses allow for different views about the available data to be made in parallel. For a human scientist, using multiple hypotheses is promoted so as to avoid prejudice, attachment to a particularly long held

belief, and to expand thoughts about what may be occurring. In a computational system these reasons are also valid, except for a notion of attachment. In this thesis a multiple hypotheses technique has been employed using a principled process of proposing and refining hypotheses. To avoid prejudice, the hypothesis management system developed allows hypotheses to question the validity of observations and will generate hypotheses with alternate views about observation validity in circumstances where it is believed the validity may be questionable. To expand the views being considered, the hypothesis manager will first consider multiple hypotheses in parallel, but will also create a number of new hypotheses with random parameters after each observation. The hypothesis manager, using an ensemble of different hypotheses, is then able to more effectively learn from small sets of observations that are potentially erroneous than standard single hypothesis techniques. The principled approach of hypothesis creation presented, may also provide a more general solution to designing models within query-by-committee style ensembles.

Separating sets of regression based hypotheses has had much consideration especially from design of experiment style approaches. However, those techniques have limitations with noisy observations and similarity between hypotheses (Atkinson and Fedorov, 1975a). With physical experimentation providing noisy observations and the hypothesis manager likely to consider a number of similar hypotheses, theses techniques are not suitable. There are many active learning techniques designed for classification, which can be converted for use in separating sets of hypotheses. Although, as shown in Chapter 4, none of these techniques outperform a strategy of selecting where the variance of the hypotheses predictions is greatest. However, the variance strategy itself can be misled by outlying hypotheses, making it possible for the technique to make sub-optimal choices in terms of the number of hypotheses that could be disproved per single experiment. Instead a new method for hypothesis separation that compares the prediction of each hypothesis to all other hypotheses, has been demonstrated to separate a set of hypotheses more effectively and robustly than selecting where the variance of predictions is maximal.

Managing the exploration vs. exploitation problem, is a key problem for an artificial experimenter. Without enough exploration, not all of the parameter space will be explored and features of the behaviours being investigated may be missed. Whilst not enough exploitation will mean that erroneous observations may be left undetected and the hypotheses may not provide enough detail for the features of the behaviour. The multi-armed bandit problem is often used to consider the exploration vs. exploitation problem. However, the multi-armed bandit problem looses some conversion to experimentation, as each of the experiments, or lever pulls, that can be made in the multi-armed bandit problem are independent of each other. Whilst in experimentation, it would be expected that the behaviour being investigated has some continual pattern or order across the parameter space. Generally techniques for addressing this problem

will either combine exploration and exploitation scores and choose the best all round experiment, or they will swap between dedicated exploration or exploitation promoting experiments, with the transition between the two often being random. Combining exploration and exploitation scores can lead to experiments that do neither very well, and randomly swapping between exploration or exploitation seems to throw away some of the information available to help make the decision. Therefore, a technique that considers more of the available information has been proposed here. In the background, several techniques are identified that discuss using a notion of surprise, and that finding surprising or unexpected observations is a good thing, as it demonstrates where the current understanding of the behaviour being investigated is weak. Using an existing definition of Bayesian surprise, we have been able to produce a method for managing the exploration vs exploitation problem in characterisation experiment selection, which out performs alternate techniques. This Bayesian surprise method has the additional advantage of being parameter free. An interesting future question would be to investigate the generality of this Bayesian surprise in discovery problems.

The techniques in this thesis have been evaluated with several problems. To test hypothesis separation, an abstract frame work was built and used to provide a toy problem to solve. Simulated behaviours have been used, to allow comparisons to be made between different hypothesis management and experiment selection techniques. Finally it has been validated through laboratory trials, where the techniques developed have been shown to characterise a 1-dimensional biological response behaviour, matching the theoretical expected response, whilst also identifying a valid non-linear component. In the next section thoughts about future work are presented.

8.2 Relation to Early Active Learning Work

Early work in active learning considers general solutions for determining the experiments to be performed, or examples to be labelled (MacKay, 1992). This work uses predicted information gain to determine the best data to select to either produce a good single model of the data, or to choose data that will differentiate between multiple models. In both cases an assumption is made that representative models of the system being investigated are under consideration. Without these, both techniques are stated to potentially fail in determining the most suitable sequence of experiments to learn from. When there is very limited data available to build the models from, then it is not guaranteed that such models will exist after n experiments in the general case. We now examine how these techniques can fail.

In the single hypothesis approach considered in (MacKay, 1992), experiments are chosen to be performed where the error bar for the model is largest, whilst in the multiple hypotheses version experiments are chosen where there is the most disagreement. The

information measure provides a disagreement function for multiple hypotheses that is similar to the discrepancy measure presented in this thesis, albeit with the discrepancy measure being more efficient to calculate. In both cases presented in (MacKay, 1992), the techniques can be considered to be exploitation only experiment selection techniques. This is because the assumption is made that the hypotheses under consideration or the model being fit, will provide a good representation of the true underlying behaviour. MacKay states that if that assumption is not true, then the techniques presented may fail to provide a good solution to the experiment selection problem. In the problem considered here, the combination of limited experimental data and erroneous observations, mean that accurate hypotheses are not guaranteed in the general case and are indeed unlikely during the majority of experimentation. Additionally, as the goal of the problem is to produce accurate models with as few experiments as possible, we would not want to continue much more experimentation once a good representative hypothesis had been produced, other than to identify it and validate it. Instead the majority of experiments in the present problem will instead be used to produce that representative hypothesis.

To expand on this, we consider further the difference between the the work in (MacKay, 1992) and the work presented in this thesis. The techniques employed in (MacKay, 1992) are designed to minimise the prediction variance of the hypothesis over number of experiments, in order reduce the error between prediction and actual. However, if an incorrect assumption is made, for example by being misled by an erroneous observation, such that the prediction variance is low for a region of the model that does not accurately represent the actual underlying behaviour, then such a technique may not converge on a model that accurately represents the underlying behaviour. Or that such convergence may occur after an extremely large number of experiments. In contrast, the technique presented in this thesis will select experiments that discriminate between hypotheses to reduce the prediction variance of the hypotheses, whilst also periodically expanding the prediction variance through the hypothesis manager producing new random hypotheses with different views of the data and exploration experiments. The expansion of the prediction variance is to try and increase the chance that any erroneous data is identified, by preventing the variance in the hypotheses predictions from becoming small based on an erroneous observation. As hypotheses are evaluated based on the experimental evidence available, with their input into the experiment selection decision weighted by this confidence, we do not waste resources investigating any artificial increases in prediction variance where consistent experimental evidence has already been obtained, as the hypotheses causing the increase will be ignored. Where there is limited data available to learn from and the reliability of the data is not known, this mix of expanding and reducing the prediction variance can lead to a faster convergence on a more accurate representation of the underlying behaviour than the variance reduction techniques presented in MacKay (1992). This can be seen in Figure 6.3a–6.3g by comparing the single

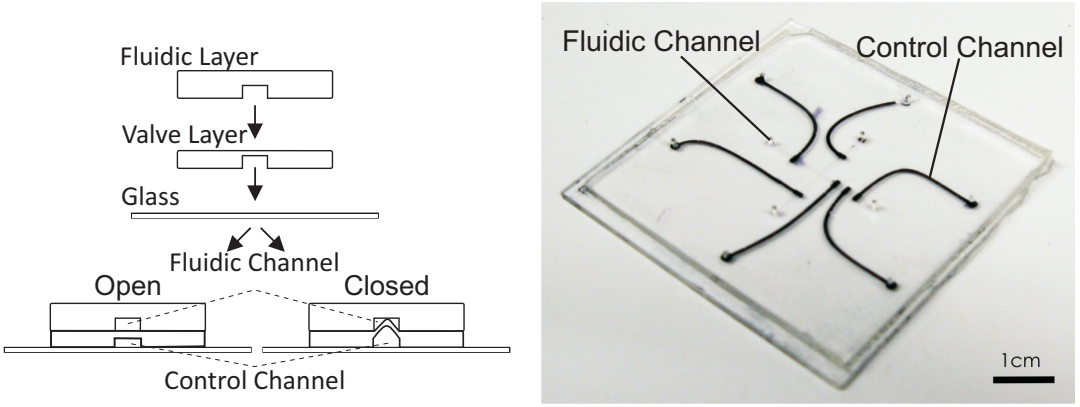
hypothesis with variance experiment selection and multiple hypotheses with the maximum discrepancy experiment selection, where both techniques are prediction variance reduction strategies.

8.3 Future work

The field of active learning is an important emerging field. The potential usage of active learning is wide ranging, from guiding physical experimentation like the problem addressed in this thesis, to learning from extremely large and complex data sets such as those found in systems biology. Like advancements in robotics allowed for a rapid increase in automated machines such as laboratory robotics and exploratory satellites, applied active learning could provide a similar rapid uptake in fully autonomising those machines. However many current techniques in active learning neglect or underestimate the real-world problems faced in the discovery systems they are building tools for. For active learning to become more prevalent and have real application usage, it will need to address more of the problems faced. For example, the number of labels that can be sought must be much lower than found in many current systems, as the resources typically available in physical experimentation will be very small. Noise must be considered, as any sensed variables that have come from a physical interaction with an environment will be inherently noisy. Erroneous observations or misclassifications must also be handled, as physical experimentation will have hidden complexities that can cause unidentifiable mistakes to occur, whilst manual labelling of examples by a human could contain misclassifications. Whilst these more general concerns need to be addressed, we present more concrete problems that could be addressed through future work next.

8.3.1 Autonomous Experimentation with a Lab-on-Chip Platform

The artificial experimenter developed here is one component of the larger concept of autonomous experimentation. Autonomous experimentation is the union of machine learning techniques that build hypotheses and select experiments, with automated laboratory hardware that can perform requested experiments. Whilst laboratory automation is a highly commercial industry, new technologies are in development that could allow for automated microscale experimentation (Jones et al., 2010a). These lab-on-chip platforms utilise microfluidic technology that allows for experiments to be performed using small volumes of chemical resources. This technology will enable the aim of reducing resource costs in experimentation to be tackled from both the algorithmic side, by reducing the number of experiments required to learn and from the physical side, by reducing the cost per experiment. Additionally, fully autonomous experimentation would provide a tool for scientists to allow them to divert their time from initial monotonous characterisation experiments, to focussing on understanding the information obtained.



(a) (b)

Figure 8.1: Microfluidic chip layered design (left) and photo of prototype chip (right). Reactants flow in channels between the fluidic and valve layers, whilst control channels exist between the valve and glass layers. Pressure on the control channels control whether fluidic channels are open or closed, to allow reactants to pass. On-chip absorbance measurement will allow for all experimentation to take place on chip.

Lab-on-chip technology, as shown in Figure 8.1, lends itself to use within automated systems. The majority of the system can be controlled using a series of in-built valves, that can block or allow the flow of liquids through the device. These valves can be operated using a computer controlled electrohydraulic interface (Jones et al., 2010a), which can be scripted to provide functionality such as flow control or pumping through a peristaltic mechanism. Additionally, experimental observations can be made through low cost on-chip implementations of standard laboratory measurement tools, for example spectrophotometers, with the observations fed directly back into a computer controller.

The union of the designed artificial experimenter with an automated lab-on-chip platform, would allow for low-cost general purpose characterisation of liquids. Whilst the motivation for this project was the automatic characterisation of biomolecular substrates, a wide range of potential domains could be applied. Additionally, the tool could also be used within machine learning research, as a cheap method for obtaining real physical data, rather than relying on large cross-disciplinary research projects to obtain samples of data, or on simulated data.

8.3.2 Extending to Further Dimensions

The natural progression for this work is to consider the problem in higher dimensional parameter spaces. Doing so will make the techniques more attractive for use in physical experimentation, where this problem is faced continually. However, by increasing the dimensionality further the problem will face a curse of dimensionality. Whilst in this thesis we have been able to build representative hypotheses with a very small number

of observations, an interesting question will be to discover the increasing scalability of the techniques proposed.

When moving from 1 to 2-dimensional parameter spaces, the surprise based active learning technique had a lesser benefit when compared to a random experiment selection strategy than it did in the 1-dimensional problem. Further investigation will be required to improve our confidence that the surprise active learning technique will provide a performance improvement over passive random sampling in higher dimensions. To make this investigation we would want to first evaluate the technique based on a larger number of underlying models. In the results taken, the random strategy outperformed the surprise technique only on the behaviour that was represented by a single 2-dimensional Gaussian peak, with the surprise technique appearing to perform far better in comparison when the behaviour was more complex. In the case of the Gaussian peak behaviour, it may be that due to the thin plate spline being able to represent that particular shape more easily through the prior used with a single hyper-parameter controlling smoothness in all parameter dimensions, that a random distribution of data points will produce thin plate splines that perform better than biased distributions created by an active selection strategy. Secondly the thin plate spline itself may not be the best technique for hypothesis representation in higher dimensions. In 1-dimension, the smoothing spline can be smoothed more with noisy data, whilst the thin-plate spline has a rapid jump from localised roughness near noisy observations to entirely smooth across the entire parameter space. The localised roughness near noisy observations can cause hypotheses to have different predictions for the data in that region, leading to the problem that as more noisy data is obtained, the predictions become rougher in that region resulting in more disagreement and more surprising observations. An updated prior may allow for more localised smoothing, whilst preventing the global shape of the fit from becoming too smooth leading to the data being underfit, which may prevent such scenarios occurring and allowing agreement between hypotheses to occur quicker and for more exploration to occur later on in experimentation.

Another avenue of investigation would be to separate the resources used in the initial exploration from the resources used during the active experiment selection phase of the artificial experimenter. By doing this we may find that the greatest dependency on the resources during the active experiment selection phase will be from the complexity of the underlying behaviour, not the complexity of the parameter space. In the methods proposed here, a number of initial exploratory experiments are performed to provide observations that can be used to develop the first set of hypotheses. These experiments are placed at the maximal regions of the parameter dimensions with others placed centrally or evenly spaced within the parameter space. Consequently, as the dimensionality increases then the number of initial exploratory experiments will also increase. In the 1-dimensional parameter space more experiments per parameter dimension were afforded to the exploration phase than the 2-dimensional phase. The reason for this was that

initial experiment placement were designed to be maximally separated from each other with a central experiment, whilst also providing a reflection symmetry along either dimension. This design was chosen to avoid biasing one dimension of the parameter space with a greater number of different experiments than another. In the 2-dimensional parameter space the minimum number of experiments to achieve this was taken, as the next suitable designs would require 9 or 13 experiments, more than doubling the exploration resource. However if enough initial exploration experiments are performed to get a good initial understanding of the behaviour, the remaining experiments may only need to examine potential erroneous observations or features not fully represented by the hypotheses. An interesting question would be to investigate a similar behaviour across an increasingly large parameter space, using a standardised set of rules to choose the initial exploration experiments, to see whether the number of active learning experiments required altered. For example a behaviour with a single peak, such as one based on a single Gaussian distribution, could be implemented across different dimensional spaces. The exploration experiments could then be chosen in each case to reside at the maximum and minimum of each parameter dimension and the central position, to ensure a standardised initial starting representation. A comparison of the number of subsequent actively chosen experiments required to achieve similar accuracy, could be made between the different dimensions. If the complexity of the underlying behaviour is the controlling factor on the number of active learning experiments required, then they should remain similar for the same underlying behaviour types.

8.3.3 Autonomous Robotic Exploration

The use of robotics to explore locations that are remote and inaccessible to human exploration, such as space or deep sea exploration, shares many similarities with artificial experimenters and laboratory autonomous experimentation. In these instances, communication between robot and human controller is restricted, either by a time delay caused by the distance separating them, or blocked completely for example by by the lack of a clear line of sight for satellite communication or by radio signals being blocked by the deep sea conditions. Without direct human control, these robots require artificial intelligences to control the robots autonomously, in order to allow them to self-determine interesting artefacts of the surroundings it is exploring. Currently such systems are in the early stages of development, and have received attention by NASA and the European Space Agency for space exploration (Stolorz and Cheeseman, 1998; Sherwood et al., 2006; Castano et al., 2007; Woods et al., 2008) and by the Monterey Bay Aquarium Research Institute for deep sea exploration (McGann et al., 2008; Rajan et al., 2009).

Robotic exploration shares a similar fundamental problem to autonomous experimentation. This is the autonomous discovery in situations where there a limited number of

physical experiments that can be performed, where the observations returned from those experiments may be noisy. In addition robotic exploration could encounter situations where the physical environment being investigated changes, either as a result of actions performed by the robotic explorer, for example lifting a rock, or from an external event, for example a rapid gaseous release or eruption causing the environment to change. However the entire goal of the system will depend on the limiting resource constraint.

In the laboratory based artificial experimenter, experiments can be placed around the entire parameter space without significant cost difference. Whilst some experiments may require a higher concentration of chemicals and so cost slightly more, the difference in cost between the most and least resource intensive experiment is likely to be small. This means that experiments can be placed in any region of the parameter space at any time, and that the region of the parameter space being investigated can change frequently without a resource concern. However, in robotic exploration there will be a cost for physically moving from one location to another. If transport resources becomes the dominant resource constraint, then the discovery problem changes from the more open situation in laboratory based experimentation where experiments can be placed anywhere, to one of determining when to stop investigating one area and move onto the next, where it is unlikely that any region of the parameter space will be returned too once the robot has left. A consideration for robotic exploration is therefore to develop robotic platforms where the transport cost is not significant to other resource concerns. The use of solar power, or energy harvesting devices (Beeby et al., 2007), may enable such cost to be reduced, allowing resource focus to be placed on the cost per experiment.

If transport costs are not limited or are sufficiently large, but there is a different resource constraint, autonomous robotic exploration may benefit from artificial experimenter techniques. Alternate limiting resource constraints would for in the case of space exploration be the communications bandwidth, where only a limited number of messages can be transmitted between the robot and the human scientists, and the robot must choose the data that is to be returned. Another example resource constraint may be chemical constraints, for example a robotic platform that wanted to examine physical properties of its surroundings using chemical analysis, where only a limited number of experiments can be performed before the chemicals need to be refilled. Such resource constraints, fit closely to the constraints placed on laboratory artificial experimenters, potentially allowing for more generalised techniques to be common between both laboratory discovery and robotic exploration.

8.3.4 Medical Diagnosis and Active Information Triage

Medical diagnosis is an area that could benefit in several ways from artificial experimenter style techniques or applied active learning. One way is in terms of reducing the number of medical tests a single patient must undergo to get a diagnosis. Whilst

another way is to reduce the time it takes to analyse large complex data sets obtained from certain medical tests, for example body scanners or microscope analysis.

Patients could benefit from techniques to reduce the number of medical tests they must endure. This problem has previously received some attention by Yu et al. (2009). In this scenario, the parallels to an artificial experimenter are clear, as the number of tests that can be performed per patient are to be minimised. Additionally, the system should also acknowledge that some medical tests will return an invalid response. However, handling erroneous observations in this medical diagnosis setting differs from that considered by the laboratory based artificial experimenter, in that the suite of available medical tests will have been performed numerous times on many different patients, allowing for a reasonable expected prior error rate to be determined. This prior information can be used to determine posterior confidences of any hypothesised diagnoses. This has a benefit if a multiple hypotheses technique is used, in that any diagnoses that ignore particular trials due to suspected error, can have their decision to ignore the trial become part of their posterior confidence. The confidence of each diagnosis can therefore be more sophisticated than in the artificial experimenter, where the current technique considers evaluation based on a least squares evaluation between data and prediction. Where as with added information about the reliability of trials, possible diagnoses that ignore unreliable trials will not be impeded and lose preference to diagnoses that accept all trials and overfit the data.

Another way active learning could be applied to medical diagnosis, is to help reducing the time taken to analyse the data obtained from a particular medical trial. Take for example large complex data sets, such as NMR analysis or analysing a cell culture, that have parameter spaces with high dimensionality, are complete and can be fully examined. However, because of the complexity of these data sets, there is a cost penalty in terms of the amount of time that is taken to analyse each sample, either by hand or by machine. The conventional approach would be to develop hardware or software capable of either analysing the data in a brute force manner or optimising particular methods for analysis, or to hire more analysts. Active learning could enable further time reductions through a form of information triaging, where the areas of the available data that appear most interesting or unexpected are identified with only a small number of potentially time consuming tests on the dataset. These more interesting regions are then focussed on with a more detailed analysis later, with the data to be examined sorted in order of interest. For example, the active learning techniques could be developed to quickly identify irregularities or inconsistencies within the data obtained from a medical test, which are highlighted for further analysis by a specialist. Those trials that suggest the most significant chance of showing an irregularity, for example cancerous tissue, could be highlighted for preferential analysis by a specialist, or analysed using more complex and slower machine learning techniques, allowing for faster diagnoses of time critical illness by prioritising the patients most likely to have a positive diagnosis. Through information

triage, large datasets could be analysed quickly, with key features highlighted, that can be examined using more time costly techniques.

8.3.5 Laboratory Classification Systems

The use of active learning techniques for laboratory assistance need not be limited to an artificial experimenter style problem. Another problem often faced is classification, for example cell sorting, which could benefit from active learning style techniques. Let us consider the problem of cell sorting. We begin with a large collection of different cells, for which we have no prior information about how to classify them. What we do have are a range of laboratory methods that can be used to characterise them, for example impedance measurements or optical measurements. We also have an expert who can analyse the data for a particular cell and provide a label. The goal is to separate the cells into their different groups.

One approach to this problem may be to perform a semi-supervised learning approach to the problem, by getting the expert to label a number of samples at the start. These labels will then be used to aid an unsupervised learner that will attempt to classify the cells. However such a system relies on either limited overlap between classifiers in the features space, or a set of labelled examples that addresses any such conflicts. As providing the labels would be expensive in terms of the experts time, we can assume that the number of labelled examples is only a few percent of the total number of examples. Therefore it is possible that first not all the different labels would have been seen and second that any conflicts in the feature space had not been resolved.

Such a problem lends itself to active learning, where the active learning technique will monitor the unsupervised learning process and request further labels in situations where there is a conflict in the feature space, or where the confidence in a particular label is low. This active learner could begin with a smaller number of labelled examples and only request further labels that are important to the classification. This has the further benefit of only requesting information that is required, in that the user does not need to be concerned over how much information should be provided at the start. With an active learning technique that only asks for the information that is required, it will prevent the lab users wasting their time or resources providing far more information than is required to perform the classification.

8.3.6 Re-factoring to a Multi-Armed Bandit Style Problem

To further investigate the exploration vs. exploitation trade-off, future research may benefit from developing more abstract problems that have a greater accessibility from the machine learning field. The multi-armed bandit problem is often used for investigating methods for managing the exploration vs. exploitation trade-off, however it lacks

some of the problems faced in physical experimentation. In the multi-armed bandit problem the levers have independent rewards, the observations are noisy and the goal is to identify the levers that will maximise reward over time. For a translation to physical experimentation, the levers would need to have a dependency between them on their rewards, such that neighbouring levers would be expected to return similar rewards and follow any trends of increasing or decreasing reward. Currently few techniques consider dependency between the machines (Pandey et al., 2007). The noise would remain, however the machines would also need to occasionally malfunction through returning rewards different to their normal programmed reward. The goal would need to be altered from finding the best single response, to maximising the accuracy of predictions for each machine. Finally the resources available would need to be far more restricted than they are considered in most multi-armed bandit problems. In this format the problem could be presented as a machine learning challenge, to characterise a behaviour with erroneous observations using the smallest number of experiments possible, but within an understood multi-armed bandit framework.

Appendix A

Representing Response Behaviours

The data obtained from response characterisation needs to be represented within a suitable framework. Unlike existing techniques for computerised hypotheses, the characterisation need not work to understand the mechanisms for how or why behaviours occur, like those found in King et al. (2004) that learns the bindings between the open reading frames and enzymatic activity, or the logistic regression used in Schmidt and Lipson (2009) to generate possible underlying laws of physically observed behaviours. Instead only the shape of the response curves or surfaces are required, with the underlying reasons or possible laws being unimportant. In this chapter a suitable regression technique for response characterisation is considered. This technique will form the basis of the hypotheses used within the artificial experimenter.

A.1 Introduction

Regression techniques look to find patterns in data sets to find possible representations of mappings between independent and dependent variables. Simple methods such as polynomial regression can provide good solutions and are often used in design of experiments settings (Box and Draper, 1987), however they can sometimes lack the ability to model more complex interactions. For instance a quadratic would be unable to fit a sine wave over several π, yet a higher polynomial term would increase the chance of overfitting the data and providing a poor fit.

The smoothing spline is a regularised regression technique that fits a piecewise cubic polynomial that has the form of a natural cubic spline (Schoenberg, 1964; Reinsch, 1967; Wahba, 1975; de Boor, 1978; Wahba, 1990), where the knots of the spline can also be weighted (Salkauskas, 1974; Bos and Salkauskas, 1987; de Boor, 2001; Davies and

Meise, 2008). The resulting equation therefore consists of several additive cubic terms. Regularisation is applied to minimise the bending energy of the regression output, which is represented as the second derivative of the function. This allows for a flexible approach that can provide a regression solution in the range of a straight line fit of the data to a point-to-point linear interpolation of the data points. The smoothing spline is in a similar vein to both Gaussian process regression (Rasmussen and Williams, 2006) and support vector machine regression (Gunn, 1998; Smola and Schölkopf, 2004). Whilst we will only consider the smoothing spline here, the relationships between these three regression techniques are explored in (Seeger, 2002).

There exists several mathematical ways for calculating the smoothing spline (Wahba, 1990; Eubank, 2004). Here we look to clarify and simplify the calculations through providing a full explanation of the implementation of smoothing spline regression using a linear least squares approach. We begin by starting with simple polynomial regression, then extend to smoothing spline regression and finally to a form allowing multidimensional independent variables. The reader may also be interested in publications by Wahba (Wahba, 1990) and the *A Simple Smoothing Spline* set of papers by Eubank, in particular the third paper (Eubank, 2004) that provides the framework for the linear least squares approach as described here. Additionally, we provide Matlab and Octave implementations of the regression algorithms described here along with worked examples to clarify aspects of the calculations. The implementations and worked examples can be found in Appendix B and Appendix C.

A.2 Regression Techniques

Regression techniques provide a method of calculating likely associations between independent and dependent variables, by reducing the expected loss (Bishop, 2006). Many loss functions exist, however often a quadratic least squares loss function is employed, minimising the difference between the dependent variables y and the predictions of those values $\hat{f}(x)$:

$$\mathcal{L}_{\text{quad}} = \min \sum_{i=1}^{n} (y_i - \hat{f}(x_i))^2 \tag{A.1}$$

The quadratic least squares loss function will be used throughout. Additionally, all of the regression described here is linear regression. Confusingly, the terms linear and non-linear refer to a statisticians view of linearity and say nothing about the shape of the line fitted to the data. That is to say that we can fit a polynomial or a logarithmic term to some data and that will still be linear regression. Therefore linear refers to whether there is a linear separation between the parameters to be learnt and the factors of the model being applied. The following section will provide more clarity on the relationship

between these parameters and factors. For future reference not necessary for current understanding, a regression calculation can be classed as linear if it satisfies the form given in (A.5)

A.2.1 Least Squares Regression

Take the case that we have a set of n independent variables, $x_i \in x_1, ..., x_n$, and corresponding dependent variables, $y_i \in y_1, ...y_n$, for which we want to determine any correlation in a linear space using a least squares loss function. Taking a function, for example, $f(x) = ax^2 + bx + c$, the goal is to identify the parameters a, b and c that best describe the relation between the independent and dependent variables. The true parameters can be vectorised in $\boldsymbol{\beta}$, such that the function is now described as:

$$f(x) = \boldsymbol{\beta}_1 + \boldsymbol{\beta}_2 x + \boldsymbol{\beta}_3 x^2 \tag{A.2}$$

Further to this, an arbitrary polynomial of degree p can be applied with a function in the form:

$$f(x) = \boldsymbol{\beta}_1 + \boldsymbol{\beta}_2 x + \boldsymbol{\beta}_3 x^2 + \cdots + \boldsymbol{\beta}_{p+1} x^p \tag{A.3}$$

The required form can be described by a basis function represented in a vector $\mathbf{x}$, which corresponds to the transposed parameter vector $\hat{\boldsymbol{\beta}}'$, which are the regression estimators for $\boldsymbol{\beta}$, as such:

$$\begin{array}{rccccccc} \mathbf{x} = & [& 1 & x & x^2 & \dots & x^p &] \\ \hat{\boldsymbol{\beta}}' = & [& \hat{\boldsymbol{\beta}}_1 & \hat{\boldsymbol{\beta}}_2 & \hat{\boldsymbol{\beta}}_3 & \dots & \hat{\boldsymbol{\beta}}_{p+1} &] \end{array} \tag{A.4}$$

Using this basis vector, $\mathbf{x}$, and the parameter vector $\hat{\boldsymbol{\beta}}$, the function can be described as a vector calculation:

$$f(x) = \mathbf{x}\hat{\boldsymbol{\beta}} \tag{A.5}$$

Subsequently, the loss function in (A.1) can be used to form the following error function, which measures the difference between the real and estimated values of y:

$$E(x) = \min_{\hat{\boldsymbol{\beta}}} \sum_{i=1}^{n} \left(y_i - \mathbf{x}_i \hat{\boldsymbol{\beta}} \right)^2 \tag{A.6}$$

From this vectorised form, we can develop the beginning of a matrix equation that can be used to solve the regression. The matrix $\mathbf{X}$ is formed such that each row contains

the basis functions, $\mathbf{x}_i$, of a particular independent variable, x_i. The column vector $\mathbf{y}$ contains the respective dependent variables, as shown:

$$\mathbf{X} = \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^p \\ 1 & x_2 & x_2^2 & \cdots & x_2^p \\ \vdots & \vdots & \cdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^p \end{bmatrix}; \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \tag{A.7}$$

The task for regression is to learn the parameters $\hat{\boldsymbol{\beta}}$ that best provide a mapping between the independent and dependent variables. The error function chosen will provide this evaluation metric. The basis function matrix $\mathbf{X}$ (sometimes referred to as the model matrix or design matrix) and parameter vector $\hat{\boldsymbol{\beta}}$, can be multiplied together to form a prediction of the dependent variable, $\hat{y}_i$, for each independent variable $\hat{x}_i$:

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}} \tag{A.8}$$

We can therefore evaluate how well the parameters in $\hat{\boldsymbol{\beta}}$ provide the mapping between independent and dependent variable through the subtraction $\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}$. This enables the error function to be redefined in matrix form:

$$E(x) = \min_{\hat{\boldsymbol{\beta}}} \left(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}\right)' \left(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}\right) \tag{A.9}$$

Which leads to a minimisation that provides the values for $\hat{\boldsymbol{\beta}}$ (Bishop, 2006, p.142):

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{y} \tag{A.10}$$

The vector $\hat{\boldsymbol{\beta}}$ is returned in the same form as the basis function. Therefore, the predicted response values for any x is calculated as:

$$\hat{y} = \hat{\boldsymbol{\beta}}_1 + \hat{\boldsymbol{\beta}}_2 x + \hat{\boldsymbol{\beta}}_3 x^2 + \cdots + \hat{\boldsymbol{\beta}}_{p+1} x^p \tag{A.11}$$

A.2.1.1 Weighted Least Squares Regression

In some circumstances it may be beneficial to give some data points a higher influence over the result of the regression. For example, some data may have a higher confidence of being correct than other data and it might be beneficial to give preference to data that is of higher confidence. This can be achieved through applying weightings. Weightings are applied to each data point through the use of the matrix $\mathbf{W}$:

$$\mathbf{W} = \begin{bmatrix} w_1 & 0 & \cdots & 0 \\ 0 & w_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & w_n \end{bmatrix} \tag{A.12}$$

The matrix calculation is then updated to include the weights, leading to $\hat{\boldsymbol{\beta}}$ being calculated as:

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}\mathbf{y} \tag{A.13}$$

A.2.2 Regularised Least Squares Regression

Regularisation allows for a prior to be applied to the regression calculation. The prior allows for selection of solutions that fit the data in a way perceived to be more preferential with respect to the prior. Consequently, the prior allows for Bayesian methods of regression to be performed.

In practical terms, better fits of the data may only be achieved by increasing the complexity of the regression method being used. For example, in linear regression the complexity would be increased by increasing the number of polynomial terms. However, increasing the complexity too much will lead to overfitting. By applying the correct regularisation, the complexity can be increased, but with the reassurance that the regularisation will penalise overfitting. A different view is that by using regularisation, far more complexity can be given to the regression method, however the regularisation will only use the amount of complexity required to find a fit that best suits the data.

The degree that regularisation impacts on the result of the regression is controlled through a scaling hyperparameter, normally denoted as λ. However it should be noted that it is possible to have more than one regularisation parameter. To demonstrate regularisation, we first consider the case of ridge regression.

A.2.2.1 Ridge Regression

Tikhonov regression, otherwise known as ridge regression (Hoerl and Kennard, 1970), is a regularised form of linear regression, solved through the minimisation of the following error function (Bishop, 2006, p.10):

$$E(\mathbf{x}) = min \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \left\| w \right\|^2 \tag{A.14}$$

Here regularisation is captured in terms of w and is often applied as an identity matrix. The hyperparameter λ controls how much regularisation is used. The matrix calculation for $\hat{\boldsymbol{\beta}}$ using ridge regression is then:

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}'\mathbf{y} \tag{A.15}$$

The matrix $\lambda\mathbf{I}$ provides the regularisation. This increases the value of the diagonal elements of the $\mathbf{X}'\mathbf{X}$ matrix. These diagonal elements are strongly representative of the dependence on each factor, where in an orthogonally designed basis matrix only the diagonal elements will be non-zero, except for the first row and column that describes the constant translation value. Through the inversion of the resultant matrix including the regularisation, the importance of the regularised factors is decreased.

Following the generally regarded definition for ridge regression in (A.15) verbatim, may be misleading as when the regularisation hyperparameter is increased the line is drawn to the origin, as seen in Figure A.1a. With the hyperparameter at is maximum, the regression will produce $y = 0$ in the 1-dimensional case. Instead by setting the first and second diagonal elements of the identity matrix to zero, only the impact of the polynomial factors $p > 1$ are changed. This leads to the result of the regression being a linear least squares fit through the data when the regularisation is at its maximum, as shown in Figure A.1d. This idea of regularising only part of the model can be extended to the more general concept of semiparametric modelling (Ruppert et al., 2003).

Weightings can be applied to ridge regression as they are in linear regression. The following matrix calculation provides the weighted form of ridge regression:

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{W}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^{\mathsf{T}}\mathbf{W}\mathbf{y} \tag{A.16}$$

Polynomial regression as described, can be performed with a low order polynomial successfully. With higher order polynomials, additional artefacts not representative of the training data can be observed. Additionally, such techniques do not perform well with periodic functions such as sine waves. We therefore use linear and ridge regression as simple examples to bridge the gap in explanation to a more sophisticated regression technique, the smoothing spline.

A.3 Smoothing Splines

Regression using polynomial basis functions can provide methods for identifying simple behaviours. With more complex behaviours, such regression techniques will require

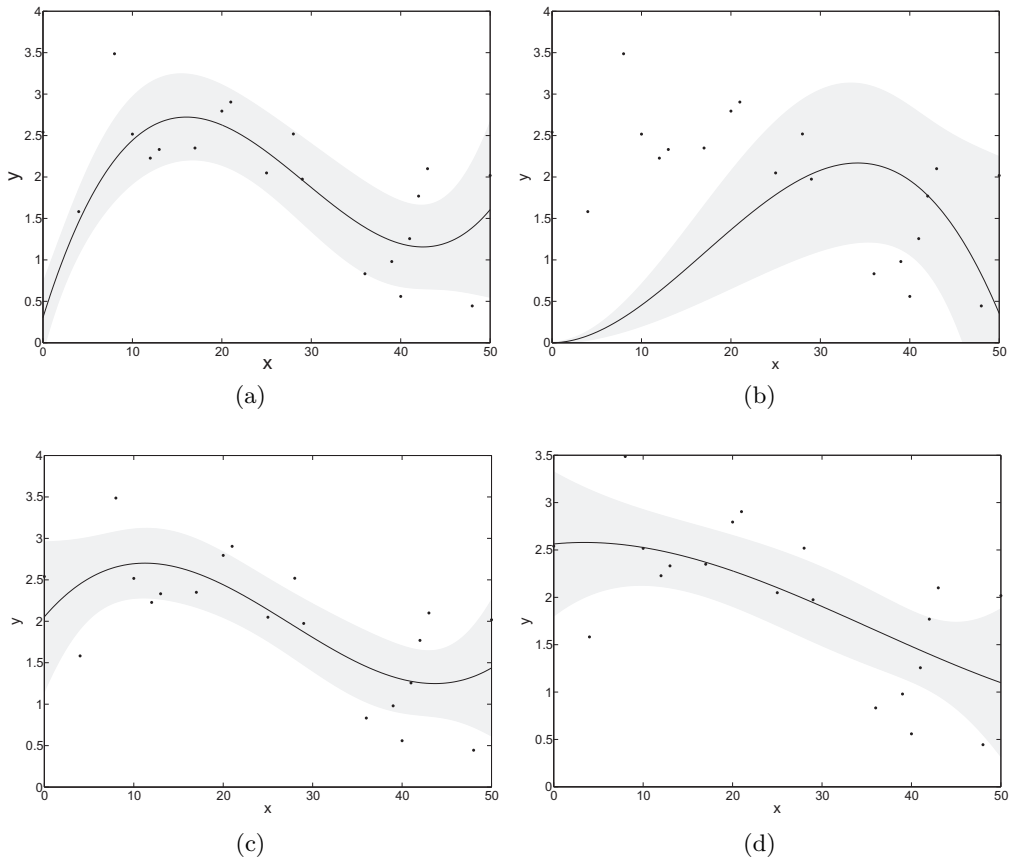


Figure A.1: Effect of changing the hyperparameter in ridge regression. In (a and b) the full identity matrix is used, in (c and d) an identity matrix with the first two diagonal elements corresponding to the intercept and the linear factor are set to zero. The hyperparameter λ in (a and c) is 10, (b and d) is 100,000. The degree of polynomial used is 3. At values below 0.1 the result of the regression is similar and not shown. The shaded grey areas show the approximate 95% confidence intervals for the regression.

higher polynomials, which will increase the complexity of the regression calculation and also increase the risk of overfitting.

The smoothing spline method uses regularised continuous piecewise cubic models (Eubank, 1994). The regularisation is performed using the second order derivative of the spline, such that roughness in the regression function is penalised to obtain a smoother fit of the data. A single hyperparameter, λ, is used to control the amount of regularisation that is applied to the smoothing spline.

Following (Wahba, 1978), a smoothing spline is defined by minimising the function:

$$E(\mathbf{x}) = \min \sum_{i=1}^{n} \left(y_i - f\left(x_i\right)\right)^2 + \lambda \int_a^b \left(f''(x)\right)^2 dx \tag{A.17}$$

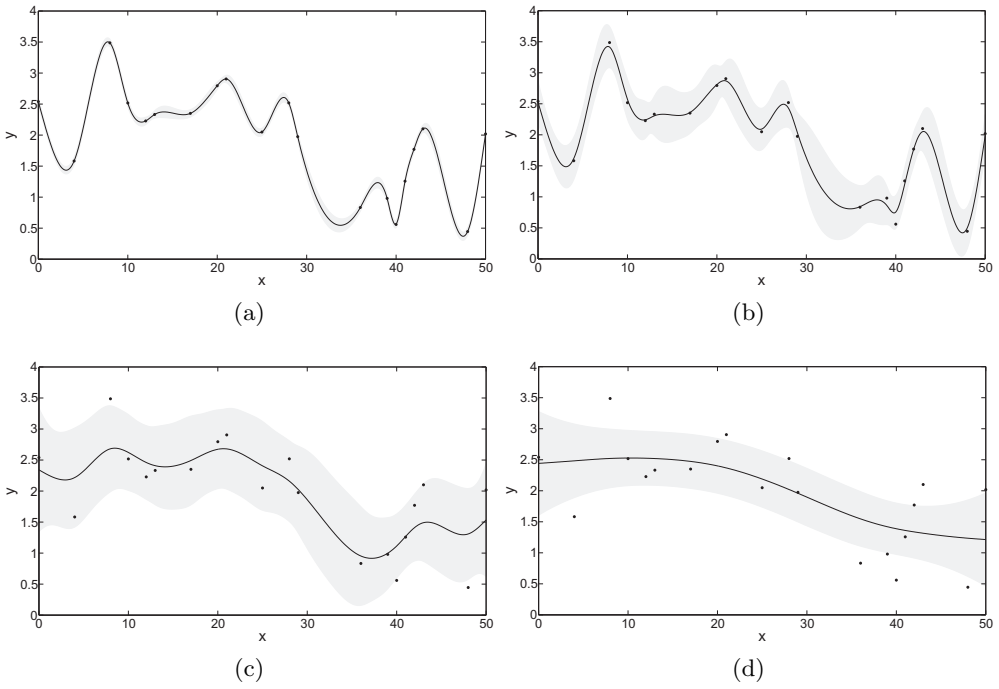


Figure A.2: Effect of changing the hyperparameter in smoothing splines. The hyperparameter λ in (a) is 0.001, (b) is 0.1, (c) is 10 and (d) is 1000. Unlike the ridge regression case, here a λ of 0.001 gives a different spline to that of 0.1. The shaded grey areas show the approximate 95% confidence intervals for the regression.

where $\lambda \geq 0$, (x_i, y_i) are pairs of training data, which are ordered by the x value such that $a \leq x_1 \leq \cdots \leq x_n \leq b$. The smoothing spline places a knot on each training point, such that the knots are identified by the ordered x training values as $\xi_1 \ldots \xi_K$, for $K = n$ knots. When $\lambda = 0$, the result is the smoothest spline that passes through all of the training data, i.e. near linear interpolation between the points. With increasing λ, the spline becomes smoother, as shown in Figure A.2. When $\lambda = \infty$, the result is the smoothest fit of the data, i.e. a straight line fit.

A.3.1 Matrix Calculation

Using the same technique as shown in Section A.2.1, the error function in (A.17) can be solved through the matrix calculation (Eubank, 2004):

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{N}'\mathbf{N} + \lambda\boldsymbol{\Omega}_{\mathbf{N}}\right)^{-1}\mathbf{N}'\mathbf{y} \tag{A.18}$$

This matrix calculation is identical to the ridge regression, except that there is a different basis matrix, now $\mathbf{N}$, and a different regulariser, $\boldsymbol{\Omega}_{\mathbf{N}}$. The basis matrix is now calculated from a piecewise natural cubic spline basis function. Rows of $\mathbf{N}$ have the form:

$$N = \begin{bmatrix} 1 & x & N_{1+2}(x) & \cdots & N_{k+2}(x) \end{bmatrix} \tag{A.19}$$

where k is the knot number and N are cubic terms calculated for all knots using the following basis functions: [1]:

$$\begin{aligned} N_1(x) &= 1, \\ N_2(x) &= x, \\ N_{k+2}(x) &= h_k(x) - h_{K-1}(x), k = 1, ..., K-2, \\ h_k(x) &= \frac{(x-\xi_k)_+^3 - (x-\xi_K)_+^3}{\xi_K - \xi_k} \end{aligned} \tag{A.20}$$

Using the basis functions in (A.20), the matrix $\mathbf{N}$ is built from the basis functions using the equation:

$$\{\mathbf{N}\}_{ij} = N_j(x_i) \tag{A.21}$$

This provides a general matrix form, where the subscript values of N are split into the form $k+2$, to show the values used in the basis function h:

$$\mathbf{N} = \begin{bmatrix} 1 & x_1 & N_{1+2}(x_1) & \dots & N_{(K-2)+2}(x_1) \\ 1 & x_2 & N_{1+2}(x_2) & \dots & N_{(K-2)+2}(x_2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & N_{1+2}(x_n) & \dots & N_{(K-2)+2}(x_n) \end{bmatrix} \tag{A.22}$$

Also using the basis functions, the regularisation matrix $\mathbf{\Omega_N}$ is calculated through (Eubank, 2004):

$$\{\mathbf{\Omega_N}\}_{ij} = \int_a^b N_i''(x) N_j''(x) dx \tag{A.23}$$

With the second order derivative of $N_1(x)$ and $N_2(x)$ being zero, the first 2 rows and columns of the matrix $\mathbf{\Omega_N}$ are also zero. When $k > 0$, the second order derivative of the basis functions can be found. To begin, the second order derivative of h becomes:

[1] Note that the function h may be referred to as d in other literature. For clarity of expressing the derivative in (A.24), h has been used.

$$\begin{aligned}\frac{d^2}{dx^2}h_k(x) &= \frac{d}{dx}\left(\frac{3\left(x-\xi_k\right)_+^2 - 3\left(x-\xi_K\right)_+^2}{\xi_K-\xi_k}\right) \\ &= \frac{6\left(x-\xi_k\right)_+ - 6\left(x-\xi_K\right)_+}{\xi_K-\xi_k}\end{aligned} \tag{A.24}$$

Leading to the second order derivative of $N_{k+2}(x)$ as:

$$N''_{k+2}(x) = \frac{6\left(x-\xi_k\right)_+}{\xi_K-\xi_k} - \frac{6\left(x-\xi_K\right)_+}{\xi_K-\xi_k} - \frac{6\left(x-\xi_{K-1}\right)_+}{\xi_K-\xi_{K-1}} + \frac{6\left(x-\xi_K\right)_+}{\xi_K-\xi_{K-1}} \tag{A.25}$$

Now if we represent the basis functions $N(x)$ in terms of i and j, $N''_i(x)N''_j(x)$ contains 16 terms derived from (A.25) to be integrated. However, the calculation is simplified due to the positive requirements of $(x-\xi)_+$. When $x \leq \xi$, the positive requirement will not be fulfilled leading to $(x-\xi)_+ = 0$. Consequently, the product of $N''_{i+2}(x)N''_{j+2}(x)$ and therefore the result of the integration is also 0. Therefore, we only need to consider the cases where $x > \xi$. With ξ_K being the maximum value of the training points, the following holds true for all training points: $x \leq \xi_K$. Therefore, the terms containing $(x-\xi_K)$ will evaluate to 0 and can be ignored. This leaves the product of $N''_{i+2}N''_{j+2}$ as:

$$N''_{i+2}N''_{j+2} = \left(\frac{6\left(x-\xi_i\right)_+}{\xi_K-\xi_i} - \frac{6\left(x-\xi_{K-1}\right)_+}{\xi_K-\xi_{K-1}}\right)\left(\frac{6\left(x-\xi_j\right)_+}{\xi_K-\xi_j} - \frac{6\left(x-\xi_{K-1}\right)_+}{\xi_K-\xi_{K-1}}\right) \tag{A.26}$$

The integration can be summed over the terms of the product of $N''_i(x)$ and $N''_j(x)$, allowing us to consider each term separately. Therefore, using the terms in (A.26) in terms of both i and j as an example, we form the following product to be integrated:

$$\begin{aligned}\int_a^b N''_{i+2}(x)N''_{j+2}(x)dx &= \int_{a_1}^{b_1}\left(\frac{6\left(x-\xi_i\right)_+}{\xi_K-\xi_i}\right)\left(\frac{6\left(x-\xi_j\right)_+}{\xi_K-\xi_j}\right)dx \\ &\quad - \int_{a_2}^{b_2}\left(\frac{6\left(x-\xi_i\right)_+}{\xi_K-\xi_i}\right)\left(\frac{6\left(x-\xi_{K-1}\right)_+}{\xi_K-\xi_{K-1}}\right)dx \\ &\quad - \int_{a_3}^{b_3}\left(\frac{6\left(x-\xi_{K-1}\right)_+}{\xi_K-\xi_{K-1}}\right)\left(\frac{6\left(x-\xi_j\right)_+}{\xi_K-\xi_j}\right)dx \\ &\quad + \int_{a_4}^{b_4}\left(\frac{6\left(x-\xi_{K-1}\right)_+}{\xi_K-\xi_{K-1}}\right)\left(\frac{6\left(x-\xi_{K-1}\right)_+}{\xi_K-\xi_{K-1}}\right)dx\end{aligned} \tag{A.27}$$

In each of the above terms the integration values a and b may be different. To solve, we can first clean up the problem by placing the scaling terms involved outside of the

integration, and finally make a general case for the integration of any component of the product of $N''_{i+2}(x)N''_{j+2}(x)$. To illustrate we consider only the first term from (A.27):

$$\begin{aligned}\int_a^b N''_{i+2}(x)N''_{j+2}(x)dx &= \int_a^b \left(\frac{6\,(x-\xi_i)_+}{\xi_K-\xi_i}\right)\left(\frac{6\,(x-\xi_j)_+}{\xi_K-\xi_j}\right)dx+\ldots \\ &= \frac{36}{(\xi_K-\xi_i)\,(\xi_K-\xi_j)}\int_a^b (x-\xi_i)_+\,(x-\xi_j)_+\,dx+\ldots \\ &= c\int_a^b (x-\alpha)_+\,(x-\beta)_+\,dx+\ldots \end{aligned} \tag{A.28}$$

where α and β represent knot values. However, as a and b are unknown in the general case, further work is required to allow the multiplication and subsequent integration in (A.28) to be generally defined. To allow for the general calculation, we need to alter the integration parameters a and b, so that the solution will be either positive or 0 for $(x-\xi_k)_+$.

First we make two assumptions, first that $b > a$ and second that $\alpha < \beta$, which are arranged such that:

$$a \le \alpha < \beta \le b$$

This allows us to integrate across the parameters, where $g(x)$ represents here the term $(x-\alpha)_+\,(x-\beta)_+$:

$$\int_a^\alpha g(x)dx + \int_\alpha^\beta g(x)dx + \int_\beta^b g(x)dx \tag{A.29}$$

Such a representation allows for further simplification of the calculation, because:

$$\int_a^\alpha (x-\alpha)_+\,(x-\beta)_+\,dx = 0 \tag{A.30}$$

as $a \le x \le \alpha$, therefore $(x-\alpha)_+ = 0$, also:

$$\int_\alpha^\beta (x-\alpha)_+\,(x-\beta)_+\,dx = 0 \tag{A.31}$$

as $\alpha \le x \le \beta$, therefore $(x-\beta)_+ = 0$. Finally,

$$\int_\beta^b (x-\alpha)_+\,(x-\beta)_+\,dx \ge 0 \tag{A.32}$$

as $\beta \le x \le b$, therefore $(x-\alpha)_+ > 0$ and $(x-\beta)_+ \ge 0$, meaning that we only need to consider this set of integration parameters and can remove the positive requirements as the results will be positive in all circumstances:

$$\int_a^b (x-\alpha)_+ (x-\beta)_+ \, dx = \int_\beta^b (x-\alpha)(x-\beta)\, dx \tag{A.33}$$

The integration can finally be calculated as:

$$\begin{aligned} \int_\beta^b (x-\alpha)(x-\beta)\, dx &= \int_\beta^b \left(x^2 - \alpha x - \beta x + \alpha\beta\right) dx \\ &= \left[\frac{1}{3}x^3 - \frac{1}{2}(\alpha+\beta)x^2 + \alpha\beta x\right]_\beta^b \end{aligned} \tag{A.34}$$

Therefore, by considering only the integrations where the maximum interval value is the largest knot, ξ_K, a tractable solution to (A.28) can be found:

$$\begin{aligned} \int_a^b N''_{i+2}(x)N''_{j+2}(x)dx &= \int_a^b \left(\frac{6(x-\xi_i)_+}{\xi_K - \xi_i}\right)\left(\frac{6(x-\xi_j)_+}{\xi_K - \xi_j}\right) dx + \dots \\ &= \frac{36}{(\xi_K-\xi_i)(\xi_K-\xi_j)} \int_{\xi_k}^{\xi_K} (x-\xi_i)(x-\xi_j)\, dx + \dots \\ &= \frac{36}{(\xi_K-\xi_i)(\xi_K-\xi_j)} \left[\frac{1}{3}x^3 - \frac{1}{2}(\xi_i+\xi_j)x^2 + \xi_i\xi_j x\right]_{\xi_k}^{\xi_K} \\ &\quad - \frac{36}{(\xi_K-\xi_i)(\xi_K-\xi_{K-1})} \left[\frac{1}{3}x^3 - \frac{1}{2}(\xi_i+\xi_{K-1})x^2 + \xi_i\xi_{K-1}x\right]_{\xi_{K-1}}^{\xi_K} \\ &\quad - \frac{36}{(\xi_K-\xi_{K-1})(\xi_K-\xi_j)} \left[\frac{1}{3}x^3 - \frac{1}{2}(\xi_{K-1}+\xi_j)x^2 + \xi_{K-1}\xi_j x\right]_{\xi_{K-1}}^{\xi_K} \\ &\quad + \frac{36}{(\xi_K-\xi_{K-1})(\xi_K-\xi_{K-1})} \left[\frac{1}{3}x^3 - \xi_{K-1}x^2 + 2\xi_{K-1}x\right]_{\xi_{K-1}}^{\xi_K} \end{aligned} \tag{A.35}$$

where the integration parameter ξ_k, is the maximum of ξ_i and ξ_j. Repeating this procedure for other valid terms of $N''_{i+2}(x)N''_{j+2}(x)$ provides the regularisation matrix $\mathbf{\Omega_N}$:

$$\mathbf{\Omega_N} = \begin{bmatrix} 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & \int N''_{1+2}(x)N''_{1+2}(x)dx & \int N''_{1+2}(x)N''_{2+2}(x)dx & \dots & \int N''_{1+2}(x)N''_{(K-2)+2}(x)dx \\ 0 & 0 & \int N''_{2+2}(x)N''_{1+2}(x)dx & \int N''_{2+2}(x)N''_{2+2}(x)dx & \dots & \int N''_{2+2}(x)N''_{(K-2)+2}(x)dx \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \int N''_{(K-2)+2}(x)N''_{1+2}(x)dx & \int N''_{(K-2)+2}(x)N''_{2+2}(x)dx & \dots & \int N''_{(K-2)+2}(x)N''_{(K-2)+2}(x)dx \end{bmatrix} \tag{A.36}$$

Using (A.18), the solution to $\hat{\boldsymbol{\beta}}$ can now be calculated. Subsequently, predictions for independent parameters can be calculated through:

$$\hat{f}_\lambda(x) = \sum_{j=1}^{N} N_j(x)\hat{\boldsymbol{\beta}}_j \tag{A.37}$$

With $\hat{f}_\lambda$ providing the mean, the error bars for the spline can be obtained using the following method (Wahba, 1983):

$$\hat{\sigma} = \hat{f}_\lambda(x_i) \pm z_{\alpha/2}\sigma\sqrt{\mathbf{A}_{ii}} \tag{A.38}$$

where $z_{\alpha/2}$ is the confidence value for the error bar, σ is variance:

$$\sigma^2 = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n - \mathrm{tr}(\mathbf{A})} \tag{A.39}$$

and $\mathbf{A}_{ii}$ are the diagonal elements of the hat matrix $\mathbf{A}$, which relates the training to the predicted points, as given by:

$$\mathbf{A} = \mathbf{N}\left(\mathbf{N}'\mathbf{N} + \lambda\boldsymbol{\Omega}_\mathbf{N}\right)^{-1}\mathbf{N}' \tag{A.40}$$

With the above giving the error bars for each training point, the following can be used to interpolate and extrapolate the error bars, where $\mathbf{z}$ is a column vector for the basis functions of x_i:

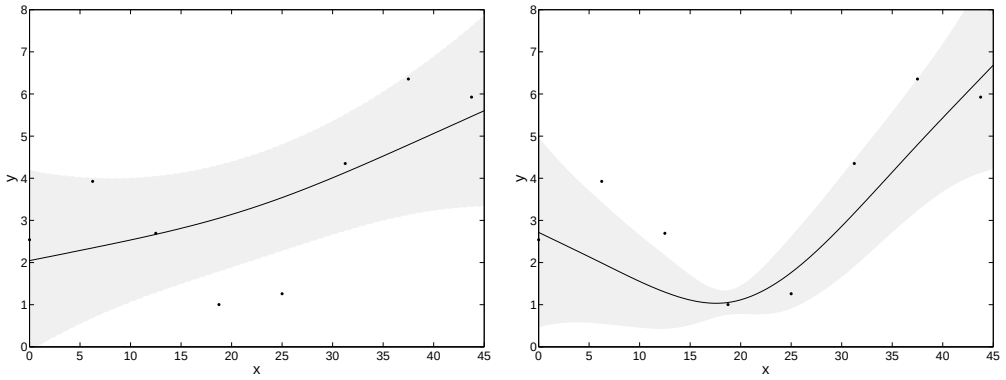
$$\hat{\sigma} = \hat{f}_\lambda(x_i) \pm z_{\alpha/2}\sigma\sqrt{\mathbf{z}'\left(\mathbf{N}'\mathbf{N} + \lambda\boldsymbol{\Omega}_\mathbf{N}\right)^{-1}\mathbf{z}} \tag{A.41}$$

A.3.2 Weighted Smoothing Spline

With the smoothing spline calculation placed in a matrix form similar to least squares regression, techniques commonly used to extend least squares regression can be easily applied to the smoothing spline equation. For example, using the weight matrix given in (A.12), a weighted smoothing spline can be calculated as:

$$\hat{\boldsymbol{\beta}} = (\mathbf{N}'\mathbf{W}\mathbf{N} + \lambda\boldsymbol{\Omega}_\mathbf{N})^{-1}\mathbf{N}'\mathbf{W}\mathbf{y} \tag{A.42}$$

where (A.37) can again be used to provide predictions for independent variables. An example of the effect of weighting observations in a smoothing spline are given in Figure A.3. With the weighted smoothing spline, the hat matrix becomes:



(a) (b)

Figure A.3: Effect of weighting training points on a smoothing spline. In both cases the smoothing parameter is chosen through bootstrapping. The dots show the training points. The solid line shows the predicted observations, with the error bars shown as the enclosing region with dashed lines. In (A) the initial spline is shown as being a near straight fit. In (B) the data point with experiment parameter 18 is weighted. The shaded grey areas show the approximate 95% confidence intervals for the regression.

$$\mathbf{A} = \mathbf{N}\left(\mathbf{N}'\mathbf{W}\mathbf{N} + \lambda\mathbf{\Omega}_{\mathbf{N}}\right)^{-1}\mathbf{N}' \tag{A.43}$$

and the error bar calculation becomes:

$$\hat{\sigma} = \hat{f}_{\lambda}(x_i) \pm z_{\alpha/2}\sigma\sqrt{\mathbf{z}'\left(\mathbf{N}'\mathbf{W}\mathbf{N} + \lambda\mathbf{\Omega}_{\mathbf{N}}\right)^{-1}\mathbf{z}} \tag{A.44}$$

A.4 Regression in Higher Dimensional Input Spaces

Few problems lend themselves to be described in a one-dimensional parameter space, therefore requiring higher dimensional solutions. Importantly, here we only consider increasing the dimensionality of the independent variables. For ease of explanation we again first consider the case of linear regression and how it can be extended to higher dimensions. We then move to extending the smoothing spline into its 2-dimensional form, the thin plate spline and then consider arbitrary dimensions.

A.4.1 Data Representation

Now we consider the case that we have more than one independent parameter, which are associated to a single dependent variable. We use the notation x_1 to represent the first

parameter and x_2 to represent the second, such that the representation of independent to dependent variables becomes:

$$f(x_1, x_2) = y \tag{A.45}$$

There is however no standard notation of independent parameters in multiple dimensions, with some approaches using x and y to represent the parameters and z to represent the dependent variable. The notation used within simply allows far higher dimensional spaces to be represented without having to find letters to represent each parameter.

Previously the independent parameter was represented as a vector. We now represent the independent parameters as an $m \times n$ matrix, where n is the number of training points and m is the number of independent parameters. To achieve this, each independent parameter is represented as a column vector $\mathbf{x}_{1\ldots m}$, with the rows representing a particular setting for that parameter. These column vectors are placed within an independent parameter matrix $\mathbf{x}$:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \mathbf{x}_3 & \cdots & \mathbf{x}_m \end{bmatrix} \tag{A.46}$$

When the independent parameters $\mathbf{x}_{1\ldots m}$ are represented in their full column vectors, the matrix $\mathbf{x}$ can be used such that $\mathbf{x}_{ij}$ provides element i within the independent parameter j. Additionally, the rows of the fully expressed matrix $\mathbf{x}$ match the rows of the dependent variable vector $\mathbf{y}$. Throughout we will refer to the column vector form of the independent parameters and only refer to the matrix $\mathbf{x}$ to identify to particular elements of the independent parameters.

A.4.2 Multi-dimensional Linear Regression

The conversion from single to multi-dimensional equations is simple, add more factors. Take the case of a 2 variable equation using of polynomial of order 2. The linear equation is simply the sum of the factors of the first variable, the second variable and the cross products between those variables:

$$\hat{y} = \hat{\boldsymbol{\beta}}_1 + \hat{\boldsymbol{\beta}}_2 x_1 + \hat{\boldsymbol{\beta}}_3 x_2 + \hat{\boldsymbol{\beta}}_4 x_1^2 + \hat{\boldsymbol{\beta}}_5 x_2^2 + \hat{\boldsymbol{\beta}}_6 x_1 x_2 \tag{A.47}$$

This extends to the least squares solution, where all that is required is an extension of the factors in the basis matrix:

$$\mathbf{X} = \begin{bmatrix} 1 & \mathbf{x}_{11} & \mathbf{x}_{12} & \mathbf{x}_{11}^2 & \mathbf{x}_{12}^2 & \mathbf{x}_{11}\mathbf{x}_{12} \\ 1 & \mathbf{x}_{21} & \mathbf{x}_{22} & \mathbf{x}_{21}^2 & \mathbf{x}_{22}^2 & \mathbf{x}_{21}\mathbf{x}_{22} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \mathbf{x}_{n1} & \mathbf{x}_{n2} & \mathbf{x}_{n1}^2 & \mathbf{x}_{n2}^2 & \mathbf{x}_{n1}\mathbf{x}_{n2} \end{bmatrix}; \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \tag{A.48}$$

The solution for $\hat{\boldsymbol{\beta}}$ is then calculated in the same way as 1-dimensional linear regression, using the matrix calculation given in (A.10). Weightings can be applied in the same manner as the 1-dimensional case, as can regularisation through a multi-dimensional ridge regression.

Further dimensions or an increased order polynomial can be achieved by determining the factors required and placing them in the matrix form. The order of terms in the matrix $\mathbf{X}$ is unimportant, so long as the order of the factors in $\mathbf{X}$ is the same as the representation used to get from $\hat{\boldsymbol{\beta}}$ to a prediction of the dependent variable. Additionally, factors can be disabled by setting the appropriate column in $\mathbf{X}$ to zero, which can be of use in determining the interaction of parameters.

A.5 Thin Plate Spline

A 2-dimensional smoothing spline, known as a thin-plate spline, as it resembles shaping a thin plate of metal around a surface, can be calculated through the following minimisation (Wahba, 1983; Hastie et al., 2009):

$$\min \sum_{i,j}^{n} (y - f(x_1, x_2))^2 + \lambda \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left(f''(x_1, x_1)^2 + 2f''(x_1, x_2)^2 + f''(x_2, x_2)^2 \right) dx_1 dx_2 \tag{A.49}$$

It would appear at first inspection that it might be possible to use the same approach as multi-dimensional linear regression, by adding the additional factors for the second parameter and the cross product between the two parameters. The regularisation could then be split into parts concerning the terms containing only x_1, terms containing only x_1 and x_2, and the terms containing only x_2. However, the spatial requirements of the smoothing spline basis functions do not match the general spatial configuration of data points in a 2-dimensional system.

Take the basis function concerning $N_{k+2}(x)$ in (A.20), which relies on the ordering of the knots. This ordering can be achieved in a 1-dimensional case, however when we extend to even 2-dimensions this ordering is not guaranteed to be the same for both the first and second parameter, as shown in Figure A.4. The loss of the ordering would cause the

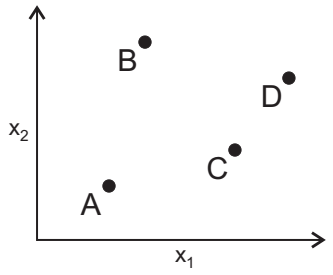


Figure A.4: Ordering of training points in a 2-dimensional system. On the x_1 axis the ordering will be A, B, C, D, whilst on the x_2 axis the ordering will be A, C, D, B.

algorithm to fail. Instead, a spline based radial basis function can be used to extend to higher dimensional spaces.

The use of a radial basis function means that the least squares format used throughout cannot be directly applied to solving a thin plate spline, as it will not correctly handle the null space. Instead a linear system including a QR decomposition is used to split the affine and non-affine components of the calculation (Bookstein, 1989; Wahba, 1990):

$$\begin{bmatrix} \mathbf{D} + \lambda \mathbf{I} & \mathbf{P} \\ \mathbf{P}' & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix} = \begin{bmatrix} \mathbf{y} \\ 0 \end{bmatrix} \tag{A.50}$$

where $\mathbf{u}$ and $\mathbf{v}$ are parameters to be learnt and can be thought of as $\hat{\boldsymbol{\beta}}$ split in two. The matrix $\mathbf{D}$ is the design matrix composed of radial basis functions, where each element in the matrix is calculated from the Euclidean distance between two training points:

$$\mathbf{D} = \begin{bmatrix} \varphi\left(\left\|\begin{bmatrix} \mathbf{x}_{11} & \mathbf{x}_{12} \end{bmatrix} - \begin{bmatrix} \mathbf{x}_{11} & \mathbf{x}_{12} \end{bmatrix}\right\|\right) & \cdots & \varphi\left(\left\|\begin{bmatrix} \mathbf{x}_{11} & \mathbf{x}_{12} \end{bmatrix} - \begin{bmatrix} \mathbf{x}_{n2} & \mathbf{x}_{n2} \end{bmatrix}\right\|\right) \\ \varphi\left(\left\|\begin{bmatrix} \mathbf{x}_{21} & \mathbf{x}_{22} \end{bmatrix} - \begin{bmatrix} \mathbf{x}_{11} & \mathbf{x}_{12} \end{bmatrix}\right\|\right) & \cdots & \varphi\left(\left\|\begin{bmatrix} \mathbf{x}_{21} & \mathbf{x}_{22} \end{bmatrix} - \begin{bmatrix} \mathbf{x}_{n2} & \mathbf{x}_{n2} \end{bmatrix}\right\|\right) \\ \vdots & \ddots & \vdots \\ \varphi\left(\left\|\begin{bmatrix} \mathbf{x}_{n1} & \mathbf{x}_{n2} \end{bmatrix} - \begin{bmatrix} \mathbf{x}_{11} & \mathbf{x}_{12} \end{bmatrix}\right\|\right) & \cdots & \varphi\left(\left\|\begin{bmatrix} \mathbf{x}_{n1} & \mathbf{x}_{n2} \end{bmatrix} - \begin{bmatrix} \mathbf{x}_{n2} & \mathbf{x}_{n2} \end{bmatrix}\right\|\right) \end{bmatrix} \tag{A.51}$$

using the radial basis function $\varphi(r)$ for the thin plate spline, which in 2-dimensions is given as (Bookstein, 1989):

$$\varphi(r) = \begin{cases} r^2 \log(r) & r > 0, \\ 0 & \text{otherwise} \end{cases} \tag{A.52}$$

Notice the spatial requirements are considered by the radial basis function, with the Euclidean distance in (A.52) between the vectors containing the prediction parameters

x_1 and x_2 and the training parameters $\mathbf{x}_{i1}$ and $\mathbf{x}_{i2}$. For reference, the Euclidean distance between two vectors $\mathbf{p}$ and $\mathbf{q}$ is calculated by:

$$\|\mathbf{p}-\mathbf{q}\| = \sqrt{(\mathbf{p}-\mathbf{q})\cdot(\mathbf{p}-\mathbf{q})} \tag{A.53}$$

Alternatively, the $n \times n$ design matrix can be mathematically described as:

$$\mathbf{D}_{ij} = \varphi\left(\left\|\begin{bmatrix}\mathbf{x}_{i1} & \mathbf{x}_{i2}\end{bmatrix} - \begin{bmatrix}\mathbf{x}_{j2} & \mathbf{x}_{j2}\end{bmatrix}\right\|\right) \tag{A.54}$$

Looking back to the thin plate spline equation in (A.50), the design matrix faces regularisation through the addition of the identity matrix scaled by the λ hyperparameter. The null space is considered by the matrix $\mathbf{P}$, where:

$$\mathbf{P} = \begin{bmatrix} 1 & x_{11} & x_{12} \\ 1 & x_{21} & x_{22} \\ \vdots & \vdots & \vdots \\ 1 & x_{n1} & x_{n2} \end{bmatrix} \tag{A.55}$$

The vectors $\mathbf{u}$ and $\mathbf{v}$ are calculating through solving the linear system (A.50) as such:

$$\hat{\boldsymbol{\beta}} = \begin{bmatrix} \mathbf{D}+\lambda\mathbf{I} & \mathbf{P} \\ \mathbf{P}' & 0 \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{y} \\ 0 \end{bmatrix} \tag{A.56}$$

To give the parameters $\mathbf{u}$ and $\mathbf{v}$ as:

$$\mathbf{u} = \begin{bmatrix} \hat{\boldsymbol{\beta}}_1 \\ \vdots \\ \hat{\boldsymbol{\beta}}_n \end{bmatrix}; \quad \mathbf{v} = \begin{bmatrix} \hat{\boldsymbol{\beta}}_{n+1} \\ \vdots \\ \hat{\boldsymbol{\beta}}_{|\hat{\boldsymbol{\beta}}|} \end{bmatrix} \tag{A.57}$$

where n is the number of training data points. Finally, to obtain predicted values from the thin plate spline for some point (x_1, x_2), a matrix calculation is performed using a column vector $\mathbf{z}$ that contains the parameters (x_1, x_2) to be predicted from, in terms of the basis functions:

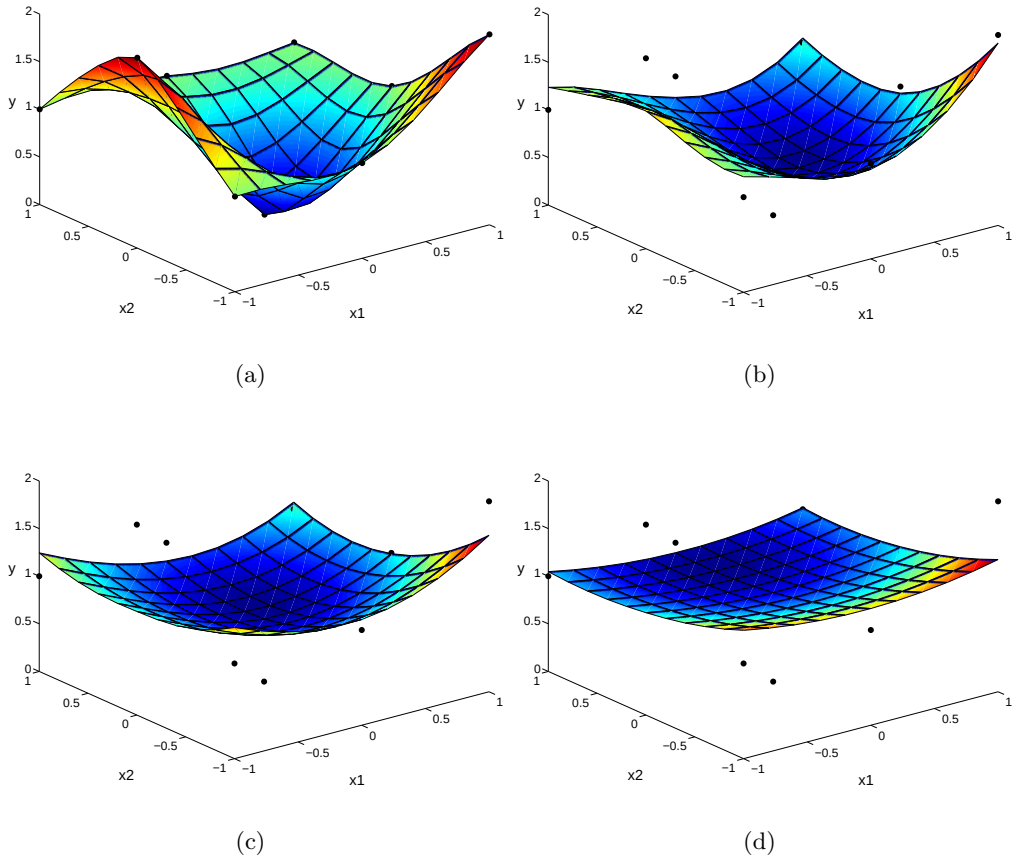


Figure A.5: Example of a thin-plate spline. The smoothing parameters λ are (A) 0, (B) 1, (C) 10 and (D) 100. Training points sit in a square around the extremes of the axes and at the centre, with $x_1 = \{1, 1, -1, -1, 0, 1, 0, -1, 0\}$, $x_2 = \{1, -1, 1, -1, 0, 0, 1, 0, -1\}$ and $y = \{1, 2, 1, 1, 0, 1, 1, 2, 1\}$.

$$\mathbf{z} = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ \varphi\left(\left\|\begin{bmatrix} x_1 & x_2 \end{bmatrix} - \begin{bmatrix} \mathbf{x}_{11} & \mathbf{x}_{12} \end{bmatrix}\right\|\right) \\ \vdots \\ \varphi\left(\left\|\begin{bmatrix} x_1 & x_2 \end{bmatrix} - \begin{bmatrix} \mathbf{x}_{n1} & \mathbf{x}_{n2} \end{bmatrix}\right\|\right) \end{bmatrix} \tag{A.58}$$

allowing for the prediction to be calculated as:

$$\hat{f}(x_1, x_2) = \mathbf{z}' \begin{bmatrix} \mathbf{v} \\ \mathbf{u} \end{bmatrix} \tag{A.59}$$

An example of a thin plate spline and the effect of altering λ is shown in Figure A.5.

A.5.1 Weighted Thin Plate Spline

Weightings are applied to the diagonal elements of the design matrix and can be thought of as altering the regularisation applied to particular training points. The weighting matrix $\mathbf{W}$ is the same as that used previously in (A.12), albeit now inversed:

$$\begin{bmatrix} \mathbf{D} + \lambda \mathbf{I}\mathbf{W}^{-1} & \mathbf{P} \\ \mathbf{P}' & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix} = \begin{bmatrix} \mathbf{y} \\ 0 \end{bmatrix} \tag{A.60}$$

This ensures that highly weighted data points will receive less regularisation, causing the regression to be pulled closer to the data point. However, this only works where diagonal elements of $\mathbf{W}$ are greater than 0.

A.5.2 Error Bars

The formulation of error bars for both smoothing spline and thin plate spline techniques often differ in method in the literature, with many approaches describing themselves to be similar to another. Here we consider a method for formulating the error bars for the thin plate spline, which is similar to the method used in the smoothing spline. Using the column vector $\mathbf{z}$ in (A.58), and placing the null space and basis function matrices into a form similar to ordinary least squares, the following calculation can be used to provide an estimate of the error bars:

$$\hat{\sigma}^2 = \hat{f}(x_1, x_2) \pm z_{\alpha/2}\sigma \sqrt{\mathbf{z}' \left(\begin{bmatrix} \mathbf{P} & \mathbf{D} + \mathbf{I}\lambda \end{bmatrix}' \mathbf{W} \begin{bmatrix} \mathbf{P} & \mathbf{D} + \mathbf{I}\lambda \end{bmatrix} \right)^{-1} \mathbf{z}} \tag{A.61}$$

A.5.3 Extending to Higher Dimensions

The radial basis function allows for generalisation to higher dimensional spaces, through the use of distance between two points in a vector space. In the general case the radial basis function using some kernel φ this can be described as:

$$\varphi\left(\|\mathbf{p} - \mathbf{q}\|\right) \tag{A.62}$$

However, the kernels require updating when dealing with higher dimensional spaces. For spline based regression, the thin-plate spline described previously is a special case of the polyharmonic spline, which has the following derivations for higher dimensional spaces (Madych and Nelson, 1990):

$$\varphi_{\text{ps}}(r) = \begin{cases} r^k & k = 1, 3, 5, ..., \\ r^k log(r) & k = 2, 4, 6, ... \end{cases} \tag{A.63}$$

The size of the design matrix is dependent on the number of training observations only. The additional parameters found in a higher dimensional system are handled by the kernels, which consider the Euclidean distance between two data points. Therefore, the design matrix remains the same as that described for the two-dimensional case in (A.51), save for each data point being represented with more parameters. The matrix $\mathbf{P}$ representing the null space grows to include the additional parameters being considered:

$$\mathbf{P}_{3d} = \begin{bmatrix} 1 & x_{11} & x_{12} & x_{13} \\ 1 & x_{21} & x_{22} & x_{23} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_n 1 & x_n 2 & x_n 3 \end{bmatrix} \tag{A.64}$$

Additionally the basis vector $\mathbf{z}$ is also updated to incorporate the additional parameter:

$$\mathbf{z}_{3d} = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ x_3 \\ \varphi\left(\left\| \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix} - \begin{bmatrix} \mathbf{x}_{11} & \mathbf{x}_{12} & \mathbf{x}_{13} \end{bmatrix} \right\|\right) \\ \vdots \\ \varphi\left(\left\| \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix} - \begin{bmatrix} \mathbf{x}_{n1} & \mathbf{x}_{n2} & \mathbf{x}_{n3} \end{bmatrix} \right\|\right) \end{bmatrix} \tag{A.65}$$

allowing the solution to be calculated using (A.59).

A.6 Choosing the Hyperparameter

The choice of the regularisation hyperparameter can drastically effect the outcome of the regression. Methods for selecting the hyperparameter rely on some evaluation function for the regression. One approach for evaluating the result of regression is to use the sum squared error (SSE), also known as the residual sum of squares (RSS):

$$\text{SSE} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{A.66}$$

Where y_i is the result at data point i from the known data, and $\hat{y}_i$ is the predicted result for the same data point. The minimiser of this evaluator would be seen as the preferred

fit. However, using the SSE has some problems. Firstly, the minimisation of the squared error would promote overfitting. Secondly, the Anscombe data set shows circumstances where the SSE can be the same for significantly different fits of the data (Anscombe, 1973). An alternative evaluation method used to evaluate smoothing splines, also takes into consideration the roughness of the spline used in the regression (Wahba, 1983):

$$E = \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n\left(1 - \frac{\mathrm{tr(A)}}{n}\right)^2} \tag{A.67}$$

With a chosen evaluation method, the next stage is to perform the evaluation of the different hyperparameter values. Regression will often split the available data into training and test sets, where the regression is performed on the training set and evaluated using the test set. The same training set will be used with the different hyperparameter values, to determine the hyperparameter that provides the best evaluation result. However, in autonomous experimentation there will often not be a large enough amount of data to be able to split into training and test sets. In these cases, alternative methods for evaluating regression methods using small amounts of data must be used. In particular we describe leave-one-out cross-validation and bootstrapping.

Both of these methods split the available data points into many different training and test sets, where each training and test set are used to provide evidence to determine the suitability of each of the proposed hyperparameters. In both of these cases we assume that we have a data set $\mathbf{X} = \{x_1, ..., x_n\}$ that contains n data points, and a set of possible hyperparameters $\lambda = \{\lambda_1, ..., \lambda_m\}$.

A.6.1 Leave-One-Out Cross-Validation

Leave-one-out cross-validation takes a single data point x_i from the data set to be the test point and uses the remaining points in the data set as the training point (Bishop, 2006, p. 33). This is repeated for all points $x_{i=1,...,n}$ such that every data point becomes the test point once. Therefore, with n data points there are n associated training and test sets. Using the chosen evaluation function, each training–test pair is then used to evaluate each of the hyperparameters being considered. A mean of the evaluation results is taken for each hyperparameter, with the hyperparameter with the lowest mean being chosen to use in the regression calculation using all of the data points. An outline of the algorithm is given:

Algorithm 1 Leave-one-out Cross Validation

for $i = 1$ to n **do**
  $t_i = x_i$ {Setting the test data}
  $T_i = \mathbf{X} - x_i$ {Setting the training data}
  for $j = 1$ to m **do**
    r $\leftarrow$ new Regression(T_i, λ_j)
    e $\leftarrow$ evaluation(r,t_i)
    $E_j \leftarrow E_j$+e
  end for
end for
$E_j \leftarrow$ mean(E_j)
$h \leftarrow$ min(E)
r $\leftarrow$ new Regression(x, λ_h)

A.6.1.1 Bootstrapping

Bootstrapping creates training sets by choosing random samples of the data from the available data with replacement (Bishop, 2006, p. 23). Training sets will often have repeated data. Available data that is not in the training set, forms the test data set. This process is repeated a number of times to create a number of training–test pairs. The bootstrapping approach allows for many different training and test sets to be created when there is only a small amount of data available. Like cross-validation, the training and test sets are used to evaluate the different hyperparameters, with each hyperparameter setting being evaluated by all the training–test pairings. Again the hyperparameter with the lowest mean error is selected to form the final representation using all available data points. An outline of the algorithm is given:

Algorithm 2 Bootstrapping

for $i = 1$ to l **do**
  $t_i \leftarrow$ random($\mathbf{X}$) {Ensure at least one test item}
  for $j = 1$ to n **do**
    $T_i \leftarrow T_i \cup$ random($\mathbf{X} - t_i$) {Add the training items}
  end for
  $t_i \leftarrow t_i \cup (\mathbf{X} - T_i)$ {Add the remaining data to the test data}
  for $j = 1$ to m **do**
    r $\leftarrow$ new Regression(T_i, λ_j)
    e $\leftarrow$ evaluation(r,t_i)
    $E_j \leftarrow E_j$+e
  end for
end for
$E_j \leftarrow$ mean(E_j)
$h \leftarrow$ min(E)
r $\leftarrow$ new Regression(x, λ_h)

A.7 Final Remarks

Provided here is a tutorial on using smoothing splines and the higher dimensional thin plate spline. The smoothing spline will form the basis of the hypotheses presented in the following chapters for the 1-dimensional cases considered, with the thin plate spline being considered for higher dimensional cases. In Appendix B, Matlab and Octave code is provided to implement the techniques described. In Appendix C are a set of examples showing the matrices that are used in the calculations.

Appendix B

Matlab Implementation

B.1 Ridge Regression

```
function [beta] = RidgeRegression(x,y,varargin)
  p = 2;
  lambda = 1;
  W = eye(length(x));

  % Identify any arguments
  optargin = size(varargin,2)
  if (optargin > 0)
    p = varargin{1};
  end

  if (optargin > 1)
    lambda = varargin{2};
  end

  % Parameter: Weights vector
  if (optargin > 2)
    weights = varargin{3};

    for i=1:length(x)
      W(i,i) = weights(i);
    end
  end

  % Basis functions
  X = ones(length(x),p+1);
  for i=1:p
    X(:,i+1) = power(x,i);
  end

  % Build regularisation matrix leaving first 2 elements 0
  om = eye(p+1);
  om(1,1) = 0;
  om(2,2) = 0;

  % The least squares calculation
  beta = pinv((X' * W * X) + (lambda * om)) * (X' * W * y);
```

```
  % Prediction vectors
  xstar = (min(x):0.1:max(x))';
  ystar = zeros(length(xstar),1);

  % Calculate response predictions
  for i=0:p
      ystar = ystar + (power(xstar,i) * beta(i+1));
  end

  % Constant terms in error bar calculations
  yHat = X * beta;
  SSE = sum(power(yHat - y,2));
  A = X * pinv((X' * W * X) + (lambda * om)) * X';
  var = sqrt(SSE / (length(x) - trace(A)));
  inner = pinv((X' * W * X) + (lambda * om));

  vP = zeros(length(xstar),1);
  vM = zeros(length(xstar),1);

  for j=1:length(xstar)
    z = ones(p+1, 1);
    z(1) = 1;
    for i=1:p
      z(i+1) = power(xstar(j), i);
    end

    err = 1.96 * var * sqrt(z' * inner * z);

    vP(j) = ystar(j) + err;
    vM(j) = ystar(j) - err;
  end

  clf
  % Provides the error bars, fails in some versions of Octave,
  % in which case use the alternative code at the end
  fill([xstar', fliplr(xstar')], [vP', fliplr(vM')],
       [0.94 0.94 0.94], 'EdgeColor', [0.94 0.94 0.94])
  hold on

  % Plot the training points
  plot(x,y, 'x', 'Color', 'k')
  hold on

  % Plot the predictions
  plot(xstar, ystar, 'k')

  % Alternative plot for error bars if 'fill' does not work
  % plot(xstar, vP, 'm');
  % plot(xstar, vM, 'y');
```

B.2 Smoothing Spline

```
function [beta] = SmoothingSpline(x,y,varargin)
  lambda = 1;
  W = eye(length(x));
```

```
% Identify any arguments
optargin = size(varargin,2);

% Parameter: Lambda smoothing value
if (optargin > 0)
  lambda = varargin{1};
end

% Parameter: Weights vector
if (optargin > 1)
  weights = varargin{2};

  for i=1:length(x)
    W(i,i) = weights(i);
  end
end

%Basis functions
N = ones(length(x),length(x));
N(:,2) = x;

for i=1 : length(x)
  for j=1 : length(x) -2
    xx = x(i);
    N(i,j+2) =  ((NonNegative(power(xx - x(j), 3))...
                    - NonNegative(power(xx - x(length(x)), 3)))...
                      / (x(length(x)) - x(j)))...
                  - ((NonNegative(power(xx - x(length(x)-1), 3))...
                    - NonNegative(power(xx - x(length(x)), 3)))...
                      / (x(length(x)) - x(length(x)-1)));
  end
end

% Calculate regularisation matrix
om = zeros(length(x), length(x));
for i=1:length(x)-2
  for j = 1:length(x)-2

      xiK = x(length(x));
      xiKm1 = x(length(x)-1);

      xiI = x(i);
      xiJ = x(j);

      % Eq1
      b = xiK;
      B = max(xiI,xiJ);
      left = ((36) / ((xiK - xiI) * (xiK - xiJ)));
      right1 = ((1/3) * power(b,3))  ...
                - (0.5 * (xiI + xiJ) * power(b,2)) ...
                + (xiI * xiJ * b);
      right2 = ((1/3) * power(B,3))  ...
                - (0.5 * (xiI + xiJ) * power(B,2)) ...
                + (xiI * xiJ * B);

      v = left * (right1 - right2);

      % Eq2
```

```
      B = xiKm1;
      left = ((36) / ((xiK - xiI) * (xiK - xiKm1)));
      right1 = ((1/3) * power(b,3))  ...
                - (0.5 * (xiI + xiKm1) * power(b,2)) ...
                + (xiI * xiKm1 * b);
      right2 = ((1/3) * power(B,3))  ...
                - (0.5 * (xiI + xiKm1) * power(B,2)) ...
                + (xiI * xiKm1 * B);

      v = v - (left * (right1 - right2));

      % Eq3
      left = ((36) / ((xiK - xiKm1) * (xiK - xiJ)));
      right1 = ((1/3) * power(b,3)) ...
                - (0.5 * (xiKm1 + xiJ) * power(b,2)) ...
                + (xiKm1 * xiJ * b);
      right2 = ((1/3) * power(B,3))  ...
                - (0.5 * (xiKm1 + xiJ) * power(B,2)) ...
                + (xiKm1 * xiJ * B);

      v = v - (left * (right1 - right2));

      % Eq4
      left = ((36) / ((xiK - xiKm1) * (xiK - xiKm1)));
      right1 = ((1/3) * power(b,3)) ...
                - (0.5 * (xiKm1 + xiKm1) * power(b,2)) ...
                + (xiKm1 * xiKm1 * b);
      right2 = ((1/3) * power(B,3)) ...
                - (0.5 * (xiKm1 + xiKm1) * power(B,2)) ...
                + (xiKm1 * xiKm1 * B);

      om(i+2,j+2) = v + (left * (right1 - right2));
  end
end

% The least squares calculation
beta = pinv(N' * W * N + (lambda * om)) * N' * W * y;

% Prediction vectors
xstar = (min(x)-1:0.1:max(x)+1)';
ystar = zeros(length(xstar),1);

% Constant terms in error bar calculations
yHat = N * beta;
SSE = sum(power(yHat - y,2));
A = N * pinv(N' * W * N + (lambda * om)) * N';
var = sqrt(SSE / (length(x) - trace(A)));
inner = pinv(N' * W * N + (lambda * om));

vP = zeros(length(xstar),1);
vM = zeros(length(xstar),1);

for i= 1 : length(xstar)
  % Build basis vector for prediction point
  z = ones(length(x),1);
  z(2) = xstar(i);
  for j = 3:length(x)
      z(j) = ((NonNegative(power(xstar(i) - x(j-2),3)) ...
                - NonNegative(power(xstar(i) - x(length(x)),3))) ...
```

```
                    / (x(length(x)) - x(j-2))) ...
                - ((NonNegative(power(xstar(i) - x(length(x)-1),3)) ...
                  - NonNegative(power(xstar(i) - x(length(x)),3))) ...
                    / (x(length(x)) - x(length(x)-1)));
  end

  % Calculate response prediction
  ystar(i) = z' * beta;

  % Error bar calculation
  err = 1.96 * var * sqrt(z' * inner * z);
  vP(i) = ystar(i) + err;
  vM(i) = ystar(i) - err;
end

clf
% Provides the error bars, fails in some versions of Octave,
% in which case use the alternative code at the end
fill([xstar', fliplr(xstar')], [vP', fliplr(vM')], ...
     [0.94 0.94 0.94], 'EdgeColor', [0.94 0.94 0.94])
hold on
plot(x,y,'x')
hold on

plot(xstar, ystar, 'k')

% Alternative plot for error bars if 'fill' does not work
%plot(xstar, vP, 'm');
%plot(xstar, vM, 'y');

% Function for dealing with non-negative requirement
function [valOut] = NonNegative(valIn)
  if valIn > 0
    valOut = valIn;
  else
    valOut = 0;
  end
```

B.3 Thin Plate Spline

```
function [beta] = TPS(x1, x2, y, lambda,varargin)

  % Put all independent and dependent variables in one matrix
  C = [x1 x2 y];

  % Initialise D - basis function matrix
  D = zeros(length(x1));

  % Initialise P - null space matrix
  P = [ones(length(x1),1) x1 x2];

  % Initialise identity matrix
  I = eye(length(x1));

  % Initialise weights and check if there are any user defined weights
  weights = ones(length(x1),1);
```

```
optargin = size(varargin,2);
if (optargin > 0)
  weights = varargin{1};
end

W = zeros(length(x1));
for i=1:length(x1)
  W(i,i) = weights(i);
end

% Inititalise the empty matrix used in the linear system
O = zeros(3);
o = zeros(3,1);

% Build the basis matrix D
for i=1:length(x1)
    for j=1:length(x1)
        D(i,j) = rbf([C(i,1) C(i,2)], [C(j,1) C(j,2)]);
    end
end

% Calculate the left hand matrix, A, of the linear system
A = [D + (lambda * I * pinv(W)) P; P' O];

% Calculate the right hand matrix, b, of the linear system
b = [y; o];

% Calculate the beta matrix
beta = pinv(A) * b;

% Obtain u and v vectors
u = beta(1:length(x1));
v = beta(length(x1)+1: length(beta));

% Calculate performance ratings for "error" bars
trA = trace([P D] * pinv([P D + (lambda * I)]' * W ...
                  * [P D + (lambda * I)] ) * [P D]');

SSE = 0;
for i=1:length(x1)
    SSE = SSE + power(y(i) - predict(x1(i),x2(i),u,v,C), 2);
end

theta = sqrt(SSE / (length(x1) - (trA - 1)));

% End of training calculation
% Determine the x values to draw for around the training values
minX1 = min(x1);
maxX1 = max(x1);
minX2 = min(x2);
maxX2 = max(x2);
% How detailed the resulting surface will be
drawSparsity = 40;
x1Interval = (maxX1-minX1)/drawSparsity;
x2Interval = (maxX2-minX2)/drawSparsity;
% The x values that will be drawn for
[X1,X2] = meshgrid(minX1:x1Interval:maxX1, minX2:x2Interval:maxX2);
% The y responses to be drawn
Y = zeros(length(X1));
```

```
  % The error bar values to be drawn
  Zp = zeros(length(X1));
  Zm = zeros(length(X1));

  % Calculate predictions for each of the x1 and x2 values
  for i=1:length(X1)
    for j=1:length(X1)
      % Perform the prediction for the tested value
      z = getZ(X1(i,j), X2(i,j), u, v, C) ;
      Y(i,j) = z' * [v;u];

      % Calculate the "error" bar
      e = 1.96 * theta * sqrt(z' * pinv([P D + (lambda * I)]' * W ...
                                  * [P D + (lambda * I)]) * z);
      Zp(i,j) = Y(i,j) + e;
      Zm(i,j) = Y(i,j) - e;
    end
  end

  % Draw the surface and optionally the "error" bar
  clf;
  plot3(x1,x2, y, 'x');
  hold on;
  % Uncomment to draw "error" bar
  %surf(X1,X2,Zp);
  surf(X1,X2,Y);
  % Uncomment to draw "error" bar
  %surf(X1,X2,Zm);

end


% Calculates the z vector
function [z] = getZ(x1,x2,u,v,C)
  z = zeros(length(u) + length(v), 1);
  z(1) = 1;
  z(2) = x1;
  z(3) = x2;

  for i=4:length(z)
      z(i) = rbf([C(i-3,1) C(i-3,2)], [x1 x2]);
  end
end

% Calculates the radial basis function for two vectors
function [out] = rbf(p,q)
    r = dist(p,q);
    if r > 0
        out = power(r,2) * log(r);
    else
        out = 0;
    end
end

% Distance between two vectors
function [d] = dist(p,q)
    d = sqrt(dot(p-q,p-q));
end
```

```
% Contained function to form prediction from x values
function [y] = predict(x1,x2,w,a,C)
    y = a(1) + (a(2) * x1) + (a(3) * x2);

    for i=1:length(w)
        y = y + w(i) * rbf([C(i,1) C(i,2)], [x1 x2]);
    end
end
```

Appendix C

Examples

C.1 Linear Regression

Take the independent variable $\mathbf{x}$ and dependent variable $\mathbf{y}$, with corresponding weights $\mathbf{w}$, where all values are ordered based on the value of $\mathbf{x}$. A polynomial of degree 2 will be used.

$$\mathbf{x} = \begin{bmatrix} 6 \\ 7 \\ 8 \\ 9 \end{bmatrix}; \mathbf{y} = \begin{bmatrix} 4 \\ 2 \\ 1 \\ 6 \end{bmatrix}; \mathbf{w} = \begin{bmatrix} 1 \\ 10 \\ 1 \\ 1 \end{bmatrix} \tag{C.1}$$

The basis matrix $\mathbf{X}$ is:

$$\mathbf{X} = \begin{bmatrix} 1 & 6 & 36 \\ 1 & 7 & 49 \\ 1 & 8 & 64 \\ 1 & 9 & 81 \end{bmatrix} \tag{C.2}$$

The weight matrix $\mathbf{W}$ is:

$$\mathbf{W} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{C.3}$$

and the regularisation matrix $\hat{\mathbf{I}}$ is:

$$\hat{\mathbf{I}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{C.4}$$

Following (A.16), with $\lambda = 1$, $\hat{\boldsymbol{\beta}}$ is calculated as:

$$\hat{\boldsymbol{\beta}} = \begin{bmatrix} 69.9545 \\ -18.6018 \\ 1.2691 \end{bmatrix} \tag{C.5}$$

Now we will get the predicted value for $x = 8.4$. The basis vector $\mathbf{z}$ is:

$$\mathbf{z} = \begin{bmatrix} 1.00 \\ 8.40 \\ 8.40^2 \end{bmatrix} = \begin{bmatrix} 1.00 \\ 8.40 \\ 70.56 \end{bmatrix} \tag{C.6}$$

Calculating $\mathbf{z}'\hat{\boldsymbol{\beta}}$, the prediction for $x = 8.4$ is $f(8.4) = 3.2463$ and an error bars calculated as 3.2463 ± 1.2596.

C.2 Smoothing Spline

Take the independent variable $\mathbf{x}$ and dependent variable $\mathbf{y}$, with corresponding weights $\mathbf{w}$, where all values are ordered based on the value of $\mathbf{x}$.

$$\mathbf{x} = \begin{bmatrix} 6 \\ 7 \\ 8 \\ 9 \end{bmatrix}; \mathbf{y} = \begin{bmatrix} 4 \\ 2 \\ 1 \\ 6 \end{bmatrix}; \mathbf{w} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 10 \end{bmatrix} \tag{C.7}$$

The important knot points therefore are $\xi_K = 9$ and $\xi_{K-1} = 8$. The basis matrix $\mathbf{N}$ becomes:

$$\mathbf{N} = \begin{bmatrix} 1 & 6 & N_{1+2}(6) & N_{2+2}(6) \\ 1 & 7 & N_{1+2}(7) & N_{2+2}(7) \\ 1 & 8 & N_{1+2}(8) & N_{2+2}(8) \\ 1 & 9 & N_{1+2}(9) & N_{2+2}(9) \end{bmatrix} \tag{C.8}$$

$$= \begin{bmatrix} 1 & 6 & h_1(6) - h_3(6) & h_2(6) - h_3(6) \\ 1 & 7 & h_1(7) - h_3(7) & h_2(7) - h_3(7) \\ 1 & 8 & h_1(8) - h_3(8) & h_2(8) - h_3(8) \\ 1 & 9 & h_1(9) - h_3(9) & h_2(9) - h_3(9) \end{bmatrix} \tag{C.9}$$

$$= \begin{bmatrix} 1.0000 & 6.0000 & 0.0000 & 0.0000 \\ 1.0000 & 7.0000 & 0.3333 & 0.0000 \\ 1.0000 & 8.0000 & 2.6666 & 0.5000 \\ 1.0000 & 9.0000 & 8.0000 & 3.0000 \end{bmatrix} \tag{C.10}$$

Continuing, the regularisation matrix $\mathbf{\Omega_N}$ is:

$$\mathbf{\Omega_N} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \int N''_{k=1}(x)N''_{k=1}(x)dx & \int N''_{k=1}(x)N''_{k=2}(x)dx \\ 0 & 0 & \int N''_{k=2}(x)N''_{k=1}(x)dx & \int N''_{k=2}(x)N''_{k=1}(x)dx \end{bmatrix} \tag{C.11}$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 16 & 9 \\ 0 & 0 & 9 & 6 \end{bmatrix} \tag{C.12}$$

The weight matrix $\mathbf{W}$ is:

$$\mathbf{W} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix} \tag{C.13}$$

Following (A.42), with $\lambda = 1$, $\hat{\boldsymbol{\beta}}$ is calculated as:

$$\hat{\boldsymbol{\beta}} = \begin{bmatrix} 5.27158 \\ -0.41380 \\ 0.60561 \\ -0.19228 \end{bmatrix} \tag{C.14}$$

Now we will get the predicted value for $x = 7.8$. The basis vector $\mathbf{z}$ is:

$$\mathbf{z} = \begin{bmatrix} 1 \\ 7.8 \\ N_{1+2}(7.8) \\ N_{2+2}(7.8) \end{bmatrix} = \begin{bmatrix} 1 \\ 7.8 \\ h_1(7.8) - h_3(7.8) \\ h_2(7.8) - h_3(7.8) \end{bmatrix} \tag{C.15}$$

$$= \begin{bmatrix} 1 \\ 7.8 \\ \frac{(7.8-6)_+^3 - (7.8-9)_+^3}{9-6} - \frac{(7.8-8)_+^3 - (7.8-9)_+^3}{9-8} \\ \frac{(7.8-7)_+^3 - (7.8-9)_+^3}{9-7} - \frac{(7.8-8)_+^3 - (7.8-9)_+^3}{9-8} \end{bmatrix} = \begin{bmatrix} 1.000 \\ 7.800 \\ 1.944 \\ 0.256 \end{bmatrix} \tag{C.16}$$

Calculating $\mathbf{z}'\hat{\boldsymbol{\beta}}$, the prediction for $x = 7.8$ is $f(7.8) = 3.17202$ and an error bars calculated as 3.17202 ± 1.9746.

C.3 Thin Plate Spline

Take the indepedent parameters $\mathbf{x}_1, \mathbf{x}_2$, the dependent variable $\mathbf{y}$ and corresponding weights $\mathbf{w}$:

$$\mathbf{x}_1 = \begin{bmatrix} 1 \\ 1 \\ 2 \\ 3 \\ 3 \end{bmatrix}; \mathbf{x}_2 = \begin{bmatrix} 1 \\ 3 \\ 2 \\ 1 \\ 3 \end{bmatrix}; \mathbf{y} = \begin{bmatrix} 1 \\ 1 \\ 4 \\ 1 \\ 0 \end{bmatrix}; \mathbf{w} = \begin{bmatrix} 1 \\ 1 \\ 10 \\ 1 \\ 1 \end{bmatrix} \tag{C.17}$$

Using (A.46), the independent parameter matrix $\mathbf{x}$ becomes:

$$\mathbf{P} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 3 \\ 1 & 2 & 2 \\ 1 & 3 & 1 \\ 1 & 3 & 3 \end{bmatrix} \tag{C.18}$$

Continuing, the basis matrix $\mathbf{D}$ becomes:

$$\mathbf{D} = \begin{bmatrix} \varphi\left(\left\|\begin{bmatrix}1\\1\end{bmatrix} - \begin{bmatrix}1\\1\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}1\\1\end{bmatrix} - \begin{bmatrix}1\\3\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}1\\1\end{bmatrix} - \begin{bmatrix}2\\2\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}1\\1\end{bmatrix} - \begin{bmatrix}3\\1\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}1\\1\end{bmatrix} - \begin{bmatrix}3\\3\end{bmatrix}\right\|\right) \\ \varphi\left(\left\|\begin{bmatrix}1\\3\end{bmatrix} - \begin{bmatrix}1\\1\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}1\\3\end{bmatrix} - \begin{bmatrix}1\\3\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}1\\3\end{bmatrix} - \begin{bmatrix}2\\2\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}1\\3\end{bmatrix} - \begin{bmatrix}3\\1\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}1\\3\end{bmatrix} - \begin{bmatrix}3\\3\end{bmatrix}\right\|\right) \\ \varphi\left(\left\|\begin{bmatrix}2\\2\end{bmatrix} - \begin{bmatrix}1\\1\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}2\\2\end{bmatrix} - \begin{bmatrix}1\\3\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}2\\2\end{bmatrix} - \begin{bmatrix}2\\2\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}2\\2\end{bmatrix} - \begin{bmatrix}3\\1\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}2\\2\end{bmatrix} - \begin{bmatrix}3\\3\end{bmatrix}\right\|\right) \\ \varphi\left(\left\|\begin{bmatrix}3\\1\end{bmatrix} - \begin{bmatrix}1\\1\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}3\\1\end{bmatrix} - \begin{bmatrix}1\\3\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}3\\1\end{bmatrix} - \begin{bmatrix}2\\2\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}3\\1\end{bmatrix} - \begin{bmatrix}3\\1\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}3\\1\end{bmatrix} - \begin{bmatrix}3\\3\end{bmatrix}\right\|\right) \\ \varphi\left(\left\|\begin{bmatrix}3\\3\end{bmatrix} - \begin{bmatrix}1\\1\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}3\\3\end{bmatrix} - \begin{bmatrix}1\\3\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}3\\3\end{bmatrix} - \begin{bmatrix}2\\2\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}3\\3\end{bmatrix} - \begin{bmatrix}3\\1\end{bmatrix}\right\|\right) & \varphi\left(\left\|\begin{bmatrix}3\\3\end{bmatrix} - \begin{bmatrix}3\\3\end{bmatrix}\right\|\right) \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 2.77259 & 0.69315 & 2.77259 & 8.31777 \\ 2.77259 & 0 & 0.69315 & 8.31777 & 2.77259 \\ 0.69315 & 0.69315 & 0 & 0.69315 & 0.69315 \\ 2.77259 & 8.31777 & 0.69315 & 0 & 2.77259 \\ 8.31777 & 2.77259 & 0.69315 & 2.77259 & 0 \end{bmatrix} \tag{C.19}$$

The weight matrix $\mathbf{W}$ is:

$$\mathbf{W} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{C.20}$$

and the regularisation matrix $\hat{\mathbf{I}}$ is:

$$\hat{\mathbf{I}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{C.21}$$

Following (A.56), with $\lambda = 1$, $\hat{\boldsymbol{\beta}}$ is calculated to be:

$$\hat{\boldsymbol{\beta}} = \begin{bmatrix} -0.4007 \\ -0.2682 \\ 1.3378 \\ -0.2682 \\ -0.4007 \\ 5.7935 \\ -0.2500 \\ -0.2500 \end{bmatrix} \tag{C.22}$$

Where the vectors **u** and **v** are:

$$\hat{u} = \begin{bmatrix} -0.4007 \\ -0.2682 \\ 1.3378 \\ -0.2682 \\ -0.4007 \end{bmatrix}; \hat{v} = \begin{bmatrix} 5.7935 \\ -0.2500 \\ -0.2500 \end{bmatrix} \tag{C.23}$$

Now we will get the predicted value for (1,1). The basis vector **z** is:

$$\mathbf{z} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ \varphi\left(\left\|\begin{bmatrix}1 & 1\end{bmatrix} - \begin{bmatrix}1 & 1\end{bmatrix}\right\|\right) \\ \varphi\left(\left\|\begin{bmatrix}1 & 1\end{bmatrix} - \begin{bmatrix}1 & 3\end{bmatrix}\right\|\right) \\ \varphi\left(\left\|\begin{bmatrix}1 & 1\end{bmatrix} - \begin{bmatrix}2 & 2\end{bmatrix}\right\|\right) \\ \varphi\left(\left\|\begin{bmatrix}1 & 1\end{bmatrix} - \begin{bmatrix}3 & 1\end{bmatrix}\right\|\right) \\ \varphi\left(\left\|\begin{bmatrix}1 & 1\end{bmatrix} - \begin{bmatrix}3 & 3\end{bmatrix}\right\|\right) \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 2.77259 \\ 0.69315 \\ 2.77259 \\ 8.31777 \end{bmatrix} \tag{C.24}$$

Calculating $\mathbf{z}'\hat{\boldsymbol{\beta}}$, the prediction for $(1, 1)$ is $f(1, 1) = 1.4007$.

References

Abe, N. and Mamitsuka, H. (1998). Query learning strategies using boosting and bagging. In *ICML '98: Proceedings of the Fifteenth International Conference on Machine Learning*, pages 1–9, San Francisco, CA, USA. Morgan Kauffmann.

Agrawal, R. (1995). Sample mean based index policies with $O(\log n)$ regret for the multi-armed bandit problem. *Advances in Applied Probability*, 27:1054–1078.

Anscombe, F. J. (1973). Graphs in statistical analysis. *The American Statistician*, 27(1):17–21.

Antos, A., Grover, V., and Szepesvári, C. (2008). Active learning in multi-armed bandits. In *Proceedings of 19th International Conference on Algorithmic Learning Theory (ALT-08)*, pages 287–302.

Atkinson, A. C. and Fedorov, V. V. (1975a). The design of experiments for discriminating between several models. *Biometrika*, 62(2):289–303.

Atkinson, A. C. and Fedorov, V. V. (1975b). The design of experiments for discriminating between two rival models. *Biometrika*, 62(1):57–70.

Atlas, L., Cohn, D., Ladner, R., El-Sharkawi, M. A., Marks, R. J., Aggoune, M. E., and Park, D. C. (1989). Training connectionist networks with queries and selective sampling. In *Advances in Neural Information Processing Systems 2*, pages 566–573.

Auer, P. (2002). Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3:397–422.

Auer, P., Cesa-Bianchi, N., and Fischer, P. (2002). Finite-time analysis of the multi-armed bandit problem. *Machine Learning*, 47:235–256.

Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. E. (1998). Gambling in a rigged casing: The adversarial multi-armed bandit problem. Technical Report NC2-TR-1998-025, Royal Holloway, University of London.

Beeby, S. P., Torah, R. N., Tudor, M. J., Glynne-Jones, P., O'Donnell, T., Saha, C. R., and Roy, S. (2007). A micro electromagnetic generator for vibration energy harvesting. *J. Micromech. Microeng.*, 17:1257–1265.

Berry, D. A. and Fristedt, B. (1985). *Bandit Problems: Sequential allocation of experiments*. Chapman and Hall.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer, New York, NY.

Bonowski, F., Kitanovic, A., Ruoff, P., Holzwarth, J., Kitanovic, I., Bui, V. N., Lederer, E., and Wölfl, S. (2010). Computer controlled automated assay for comprehensive studies of enzyme kinetic parameters. *PLos ONE*, 5(5):1–10.

Bookstein, F. L. (1989). Principlal warps: Thin-plate splines and decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6):567–585.

Bos, L. and Salkauskas, K. (1987). *Weighted Splines Based on Piecewise Polynomial Weight Functions*, chapter 5, pages 87–98. Society for Industrial and Applied Mathematics.

Box, G. E. P. and Draper, N. R. (1987). *Empirical Model-Building and Response Surfaces*. Wiley.

Box, G. E. P., Hunter, J. S., and Hunter, W. G. (2005). *Statistics for Experimenters: Design, Innovation, and Discovery*. Wiley-Intersecience, 2nd edition.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2):123–140.

Bryant, C. H., Muggleton, S. H., Oliver, S. G., Kell, D. B., Reiser, P., and King, R. D. (2001). Combining inductive logic programming, active learning and robotics to discover the function of genes. *Electronic Transactions in Artificial Intelligence*, 6:1–34.

Buchanan, B. G., Sutherland, G., and Feigenbaum, E. (1969). *Heuristic Dendral: A Program for Generating Explanatory Hypotheses in Organic Chemistry*, volume 4 of *Machine Intelligence*, chapter 12, pages 209–254. American Elsevier, New York.

Buck, C. (1975). Popper's philosophy for epidemiologists. *International Journal of Epidemiology*, 4(3):159–168.

Bundy, A. (2008). Why ontology evolution is essential in modelling scientific discovery. In *2008 AAAI Fall Symposium on Automated Scientific Discovery*, pages 8–9, Menlo Park, CA, USA. The AAAI Press.

Burbidge, R., Rowland, J. J., and King, R. D. (2007). Active learning for regression based on query by committee. In *Intelligent Data Engineering and Automated Learning – IDEAL 2007*, pages 209–218. Springer-Verlag.

Castano, R., Estlin, T., Gaines, D., Chouinard, C., Bornstein, B., Anderson, R. C., Burl, M., Thompson, D., Castano, A., and Judd, M. (2007). Onboard autonomous rover science. In *Proceedings of the 2007 IEEE Aerospace Conference, Big Sky, Montana*, pages 1–13.

Castro, R., Willet, R., and Nowak, R. (2005). Faster rates in regression via active learning. In *Advances in Neural Information Processing Systems 18 (NIPS-05)*.

Cebron, N. and Berthold, M. R. (2009). Active learning for object classification: from exploration to exploitation. *Data Mining and Knowledge Discovery*, 18:283–299.

Chamberlin, T. C. (1890). The method of multiple working hypotheses. *Science (old series)*, 15:92–96. Reprinted in: Science, v. 148, p. 754–759, May 1965.

Chernoff, H. (1959). Sequential design of experiments. *The Annals of Mathematical Statistics*, 30:755–770.

Christensen, S. W., Sinclair, I., and Reed, P. A. S. (2003). Designing committees of models through deliberate weighting of data points. *Journal of Machine Learning Research*, 4:39–66.

Cohn, D., Atlas, L., and Ladner, R. (1994). Improving generalization with active learning. *Machine Learning*, 15:201–221.

Cohn, D. A., Ghahramani, Z., and Jordan, M. I. (1996). Active learning with statistical models. *Journal of Artificial Intelligence Research*, 4:129–145.

Corkan, L. A. and Lindsey, J. S. (1992). Experiment manager software for an automated chemistry workstation, including a scheduler for parallel experimentation. *Chemom. Intell. Lab. Syste.: Lab. Info. Manage.*, 17:47–74.

Danziger, S. A., Zeng, J., Wang, Y., Brachmann, R. K., and Lathrop, R. H. (2007). Choosing where to look next in a mutation sequence space: Active learning of informative p53 cancer rescue mutants. *Bioinformatics*, 23:104–114.

Darden, L. (1997). Recent work in computational scientific discovery. In Shaffo, M. and Langley, P., editors, *Proceedings of the Nineteenth Annual Conference of the Cognitive Science Society*, pages 161–166, Mahwah, New Jersey.

Davies, P. L. and Meise, M. (2008). Approximating data with weighted smoothing splines. *Journal of Nonparametric Statistics*, 20(3):207–228.

de Boor, C. (1978). *A Practical Guide to Splines*. Springer Verlag, New York.

de Boor, C. (2001). Calculation of the smoothing spline with weighted roughness measure. *Mathmatical Models and Methods in Applied Sciences*, 11:33–41.

de Silva, A. P. and Uchiyama, S. (2007). Molecular logic and computing. *Nature Nanotechnology*, 2:399–410.

Dixon, J. M., Du, H., Cork, D. G., and Lindsey, J. S. (2002). An experimental planner for performing successive focused grid searches with an automated chemistry workstation. *Chemometrics and Intelligent Laboratory Systems*, 62:115–128.

Du, H., Corkan, L. A., Yang, K., Kuo, P. Y., and Lindsey, J. S. (1999a). An automated microscale chemistry workstation capable of parallel, adaptive experimentation. *Chemometrics and Intelligent Laboratory Systems*, 48:181–203.

Du, H., Jindal, S., and Lindsey, J. S. (1999b). Implementation of the multidirectional search algorithm on an automated chemistry workstation. a parallel yet adaptive approach for reaction optimization. *Chemometrics and Intelligent Laboratory Systems*, 48:235–256.

Du, H. and Lindsey, J. S. (2002). An approach for parallel and adaptive screening of discrete compounds followed by reaction optimization using an automated chemistry workstation. *Chemometrics and Intelligent Laboratory Systems*, 62:159–170.

Du, H., Shen, W., Kuo, P. Y., and Lindsey, J. S. (1999c). Decision-tree programs for an adaptive automated chemistry workstation. application to catalyst screening experiments. *Chemometrics and Intelligent Laboratory Systems*, 48:205–217.

Du, T. and Zhang, S. (2005). Active learning with ensembles for DOE. In *Current Trends in High Performance Computing and Its Applications*, pages 283–287. Springer Berlin Heidelberg.

Engelson, S. P. and Dagan, I. (1996). Minimizing manual annotation cost in supervised training from corpora. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pages 319–326. Association for Computational Linguistics.

Eubank, R. L. (1994). A simple smoothing spline. *The American Statistician*, 48(2):103–106.

Eubank, R. L. (2004). A simple smoothing spline, III. *Computational Statistics*, 19:227–241.

Evans, J. and Rzhetsky, A. (2010). Machine science. *Science*, 329:399–340.

Fischer, P. and Żytkow, J. M. (1990). Discovering quarks and hidden structure. In *Methodologies for Intelligent Systems 5*.

Fisher, R. A. (1925). *Statistical Methods for Research Workers*. Oliver and Boyd.

Fisher, R. A. (1935). *The Design of Experiments*. Oliver and Boyd.

Franklin, A. D. (1981). What makes a 'good' experiment? *The British Journal for the Philosophy of Science*, 32(4):367–374.

Freund, Y., Seung, H. S., Shamir, E., and Tishby, N. (1997). Selective sampling using the query by committee algorithm. *Machine Learning*, 28:133–168.

Gaber, M. M., editor (2010). *Scientific Data Mining and Knowledge Discovery*. Springer.

Giza, P. (2002). Automated discovery systems and scientific realism. *Minds and Machines*, 12:105–117.

Gooding, D. (1990). *Experiment and the Making of Meaning*. Kluwer Academic Publishers, Dordrecht.

Gooding, D. and Addis, T. R. (1999). A simulation of model-based reasoning about disparate phenomena. In Magnani, L., Nersessian, N. J., and Thagard, P., editors, *Model Based Reasoning in Scientific Discovery*, pages 103–123, New York. Kluwer Academic/Plenum Publishers.

Gough, J., Jones, G., Lovell, C. J., Macey, P., Morgan, H., Revilla, F., Spanton, R., Tsuda, S., and Zauner, K.-P. (2009). Integration of cellular biological structures into robotic systems. *Acta Futura*, 3:43–49.

Grismer, M. E. (1992). Field sensor networks and automated monitoring of soil water sensors. *Soil Science*, 154(6):482–489.

Gunn, S. R. (1998). Support vector machines for classification and regression. Technical report, University of Southampton.

Hall, P. and Molchanov, I. (2003). Sequential methods for design-adaptive estimation of discontinuities in regression curves and surfaces. *The Annals of Statistics*, 31(3):921–941.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2 edition.

Hey, T., Tansley, S., and Tolle, K., editors (2009). *The Fourth Paradigm: Data-Intensive Scientific Discovery*. Microsoft Research, Redmond, Washington.

Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1).

Hoffman, A. and Mahidadia, A. (2010). Machine learning. In *Scientific Data Mining and Knowledge Discovery*, pages 7–52. Springer.

Hoffman, A. G. and Thakar, S. (1991). Acquiring knowledge by efficient query learning. In *International Joint Conference on Artificial Intelligence*, pages 738–788.

Huang, K.-M. and Żytkow, J. M. (1997). Discovering empirical equations from robot-collected data. *Foundations of Intelligent Systems*, pages 287–297.

Itti, L. and Baldi, P. (2009). Bayesian surprise attracts human attention. *Vision Research*, 49:1295–1306.

Jones, G. (2009). Towards an integrated microuidic device for an autonomous experimentation system. Mini-thesis, School of Electronics and Computer Science, University of Southampton.

Jones, G., Lovell, C. J., Morgan, H., and Zauner, K.-P. (2010a). Characterising enzymes for information processing: Microfluidics for autonomous experimentation (abstract). In *9th International Conference on Unconventional Computation*, page 191, Tokyo, Japan.

Jones, G., Lovell, C. J., Morgan, H., and Zauner, K.-P. (2010b). Organising chemical reaction networks in space and time with microfluidics. In *1st International Workshop on Computing with Spatio-Temporal Dynamics*, Tokyo, Japan.

Jones, G., Lovell, C. J., Morgan, H., and Zauner, K.-P. (2011). Organising chemical reaction networks in space and time with microfluidics. *International Journal of Nanotechnology and Molecular Computation (IJNMC)*, 3(1):35–56.

Kaelbling, L. P., Littman, M. L., and Moore, A. W. (1996). Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285.

Karp, P. D. (1993). Design methods for scientific hypothesis formation and their application to molecular biology. *Machine Learning*, 12:89–116.

King, R. D. (2006). Robot scientist: an autonomous platform for systems biology discovery. Technical Report SC-POS-04 09/06, Caliper LifeSciences.

King, R. D., Rowland, J., Oliver, S. G., Young, M., Aubrey, W., Byrne, E., Liakata, M., Markham, M., Pir, P., Soldatova, L. N., Sparkes, A., Whelan, K. E., and Clare, A. (2009). The automation of science. *Science*, 324(5923):85–89.

King, R. D., Whelan, K. E., Jones, F. M., Reiser, P. G. K., Bryant, C. H., Muggleton, S. H., Kell, D. B., and Oliver, S. G. (2004). Functional genomic hypothesis generation and experimentation by a robot scientist. *Nature*, 427:247–252.

Krogh, A. and Vedelsby, J. (1995). Neural network ensembles, cross validation, and active learning. In *Advances in Neural Information Processing Systems 7*, pages 231–238, Cambridge MA. MIT Press.

Kuhn, T. S. (1962). *The Structure of Scientific Revolutions*. University of Chicago Press.

Kulkarni, D. and Simon, H. A. (1988). The processes of scientific discovery: The strategy of experimentation. *Cognitive Science*, 12:139–175.

Kulkarni, D. and Simon, H. A. (1990). Experimentation in machine discovery. In Shrager, J. and Langley, P., editors, *Computational Models of Scientific Discovery and Theory Formation*, pages 255–273. Morgan Kaufmann Publishers, San Mateo, CA.

Kulkarni, S. R., Mitter, S. K., and Tsitsiklis, J. N. (1993). Active learning using arbitrary binary valued queries. *Machine Learning*, 11:23–35.

Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *Annals of Mathematical Statistics*, 22:79–86.

Kuo, P. Y., Du, H., Corkan, L. A., Yang, K., and Lindsey, J. S. (1999). A planning module for performing grid search, factorial design, and related combinatorial studies on an automated chemistry workstation. *Chemometrics and Intelligent Laboratory Systems*, 48:219–234.

Langley, P. (2002). Lessons for the computational discovery of scientific knowledge. In *Proceedings of First International Workshop on Data Mining Lessons Learned*, pages 9–12, Sydney.

Langley, P., Simon, H. A., Bradshaw, G. L., and Żytkow, J. M. (1987). *Scientific Discovery: computational explorations of the creative processes*. MIT Press.

Lewis, D. and Gale, W. (1994a). A sequential algorithm for training text classifiers. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3–12.

Lewis, D. D. and Gale, W. A. (1994b). A sequential algorithm for training text classifiers. In *ACM Conference on Research and Development in Information Retrieval*, pages 3–12.

Lindsay, R. K., Buchanan, B. G., Feigenbaum, E. A., and Lederberg, J. (1993). DENDRAL: a case study of the fist expert system for scientific hypothesis formation. *Artificial Intelligence*, 61:209–261.

Lindsey, J. S. (1992). A retrospective on the automation of laboratory synthetic chemistry. *Chemometrics and Intelligent Laboratory Systems*, 17:15–45.

Lovell, C. J., Jones, G., Gunn, S. R., and Zauner, K.-P. (2010a). An artificial experimenter for enzymatic response characterisation. In *13th International Conference on Discovery Science*, pages 42–56, Canberra, Australia.

Lovell, C. J., Jones, G., Gunn, S. R., and Zauner, K.-P. (2010b). Autonomous experimentation: Coupling active learning with computer controlled microfluidics (abstract). In *Active Learning and Experimental Design workshop at AISTATS*, Sardinia.

Lovell, C. J., Jones, G., Gunn, S. R., and Zauner, K.-P. (2010c). Characterising enzymes for information processing: Towards an artificial experimenter. In et al., C. S. C., editor, *9th International Conference on Unconventional Computation*, volume 6079, pages 81–92, Tokyo, Japan.

Lovell, C. J., Jones, G., Gunn, S. R., and Zauner, K.-P. (2011). Autonomous experimentation: Active learning for enzyme response characterisation. *JMLR: Workshop and Conference Proceedings*, 16:141–154.

Lovell, C. J., Jones, G., and Zauner, K.-P. (2009). Autonomous experimentation: Coupling machine learning with computer controlled microfluidics (abstract). In *European Laboratory and Robotics Interest Group Robotics Workshop, ELRIG Drug Discovery 2009*, Liverpool, UK.

Lovell, C. J. and Zauner, K.-P. (2008a). Autonomous experimentation: Methods for characterising molecular computing substrates (poster). In *SemiBiotic Systems Conference*, Malta.

Lovell, C. J. and Zauner, K.-P. (2008b). Autonomous experimentation: Methods for characterising molecular computing substrates (poster). In *3rd Microsoft Research Summer School*, Cambridge, UK.

Lovell, C. J. and Zauner, K.-P. (2009). Towards algorithms for autonomous experimentation (extended abstract). In *Eighth International Conference on Information Processing in Cells and Tissues (IPCAT 2009)*, pages 150–152, Ascona, Switzerland.

MacKay, D. J. C. (1992). Information–based objective functions for active data selection. *Neural Computation*, 4:589–603.

Madych, W. R. and Nelson, S. A. (1990). Polyharmonic cardinal splines. *J. Approx. Theory*, 60:141–156.

March, J. G. (1991). Exploration and exploitation in organizational learning. *Organization Science*, 2(1):71–87.

Matsumaru, N., Centler, F., Zauner, K.-P., and Dittrich, P. (2004). Self-adaptive scouting—autonomous experimentation for systems biology. In Raidl, G. R. et al., editors, *Applications of Evolutionary Computing: EvoWorkshops 2004*, volume 3005 of *Lecture Notes in Artificial Intelligence*, pages 52–62. Springer, Heidelberg.

Matsumaru, N., Colombano, S., and Zauner, K.-P. (2002). Scouting enzyme behavior. In Fogel, D. B., El-Sharkawi, M. A., Yao, X., Greenwood, G., Iba, H., Marrow, P., and Shackleton, M., editors, *2002 World Congress on Computational Intelligence, May 12-17*, pages CEC 19–24, Honolulu, Hawaii. IEEE, Piscataway, NJ.

Matsumoto, T., Du, H., and Lindsey, J. S. (2002a). A parallel simplex search method for use with an automated chemistry workstation. *Chemometrics and Intelligent Laboratory Systems*, 62:129–147.

Matsumoto, T., Du, H., and Lindsey, J. S. (2002b). A two-tiered strategy for simplex and multidirectional optimization of reactions with an automated chemistry workstation. *Chemometrics and Intelligent Laboratory Systems*, 62:149–158.

McCallum, A. K. and Nigam, K. (1998). Employing em and pool-based active learning for text classification. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 584–591. Morgan Kaufmann.

McGann, C., Py, F., Rajan, K., Ryan, J., and Henthorn, R. (2008). Adaptive control for autonomous underwater vehicles. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*.

McGuinness, D. L., Fox, P., and Brodaric, B., editors (2008). *Semantic Scientific Knowledge Integration: Papers from the 2008 AAAI Spring Symposium*.

Muggleton, S. and Zauner, K.-P. (2006). Artifical scientists. In Emmott, S. and Rison, S., editors, *Towards 2020 Science*, pages 36–37. Microsoft Research, Cambridge.

Myers, R. H., Montgomery, D. C., and Anderson-Cook, C. M. (2009). *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*. John Wiley & Sons, 3rd edition.

Nelles, O. (2000). *Nonlinear System Identification*. Springer Verlag, Heidelberg, Germany.

Nelson, D. L. and Cox, M. M. (2008). *Lehninger Principles of Biochemistry*. W. H. Freeman and Company, New York, USA, 5th edition.

Olsson, F. (2009). A literature survey of active machine learning in the context of natural language processing. Technical report, Swedish Institute of Computer Science.

Osugi, T., Jun, D., and Scott, S. (2005). Balancing exploration and exploitation: A new algorithm for active machine learning. In *Fifth IEEE International Conference on Data Mining (ICDM'05)*.

Pandey, S., Chakrabarti, D., and Agarwal, D. (2007). Multi-armed bandit problems with dependent arms. In *Proceedings of 24th International Conference on Machine Learning*, pages 721–728.

PerkinElmer (2004). *Technical Specifications for the LAMBDA 650 UV/Vis Spectrophotometer*.

Pfaffmann, J. O. and Zauner, K.-P. (2001). Scouting context-sensitive components. In Keymeulen, D., Stoica, A., Lohn, J., and Zebulum, R. S., editors, *The Third NASA/DoD Workshop on Evolvable Hardware—EH-2001*, pages 14–20, Long Beach. IEEE Computer Society, Los Alamitos.

Platt, J. R. (1964). Strong inference, certain systematic methods of scientific thinking may produce much more rapid progress than others. *Science*, 146(3642).

Plouvier, J. C., Corkan, L. A., and Lindsey, J. S. (1992). Experiment planner for strategic experimentation with an automated chemistry workstation. *Chemometrics and Intelligent Laboratory Systems*, 17:75–94.

Rajan, K., Py, F., McGann, C., Ryan, J., O'Reilly, T., Maughan, T., and Roman, B. (2009). Onboard adaptive control of AUVs using autmoated planning and execution. In *International Symposium on Unmanned Untethered Submersible Technology (UUST)*.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, Massachusetts.

RayChaudhuri, T. and Hamey, L. G. C. (1995). Minimisation of data collection by active learning. In *IEEE International Conference on Neural Networks*, volume 3, pages 1338–1341.

Rechenberg, I. (1965). Cybernetic solution path of an experimental problem. Technical report, Royal Aircraft Establishment, Library Translation 1122, Farnborough. Reprinted with commentary in D. B. Fogel, editor, *Evolutionary Computation: The Fossil Record*, pages 297–309, IEEE Press, New York, 1998.

Reinsch, C. H. (1967). Smoothing by spline functions. *Numberische Mathematik*, 19:177–183.

Robbins, H. (1952). Some aspects of the sequential design of experiments. *Bulletin on the American Mathematical Society*, 58(5):527–535.

Ruppert, D., Wand, M. P., and Carroll, R. J. (2003). *Semiparametric Regression*. Cambridge University Press.

Salkauskas, K. (1974). C^1 splines for interpolation of rapidly varying data. *Rocky Mountain Journal of Mathematics*, 14(1):239–250.

Schmidt, M. and Lipson, H. (2009). Distilling free-form natural laws from experimental data. *Science*, 324(5923):81–85.

Schoenberg, I. J. (1964). Spline functions and the problem of graduation. In *Proceedings of the National Academy of Sciences of the United States of America (PNAS)*, volume 52, pages 947–950.

Seeger, M. (2002). Relationships between gaussian processes, support vector machines and smoothing splines. Technical report, University of Edinburgh.

Seelig, G., Soloveichik, D., Zhang, D. Y., and Winfree, E. (2006). Enzyme-free nucleic acid logic circuits. *Science*, 314:1585–1588.

Settles, B. (2009). Active learning literature survey. Technical report, University of Wisconsin-Madison.

Settles, B. and Craven, M. (2008). An analysis of active learning strategies for sequence labelling tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1069–1078.

Seung, H. S., Opper, M., and Sompolinsky, H. (1992). Query by committee. In *Proceedings of the ACM Workshop on Computational Learning Theory*, pages 287–294.

Shapire, R. (1990). The strength of weak learnability. *Machine Learning*, 5(2):197–227.

Shen, W.-M. (1994). Discovery as autonomous learning from the environment. *Mach. Learn.*, 12(1-3):143–165.

Sherwood, R., Chien, S., Tran, D., Davies, A., Castano, R., Rabideau, G., Mandl, D., Szwaczkowski, J., Frye, S., and Shulman, S. (2006). Enhancing science and automating operations using onboard autonomy. In *The 9th International Conference on Space Operations, Rome, Italy.*

Siegel, J. M., Montgomery, G. A., and Bock, R. M. (1959). Ultraviolet absroption spectra of dpn and analogs of dpn. *Archives of Biochemistry and Biophysics*, 82(2):288–299.

Sigma-Aldrich (2010). *β-Nicotinamide adenine dinucleotide, reduced dipotassium salt hydrate Product Information Sheet N4505.*

Smola, A. J. and Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and Computing*, 14:199–222.

Soldatova, L. N., Clare, A., Sparkes, A., and King, R. D. (2006). An ontology for a robot scientist. *Bioinformatics*, 22:464–471.

Soldatova, L. N. and King, R. D. (2005). Are the current ontologies in biology good ontologies? *Nature Biotechnology*, 23:1095–1098.

Soldatova, L. N. and King, R. D. (2006). An ontology of scientific experiments. *Journal of the Royal Society Interface*, 3(11).

Spendley, W., Hext, G. R., and Himsworth, F. R. (1962). Sequential application of simplex designs in optimisation and evolutionary operation. *Technometrics*, 4(4):441–461.

Stolorz, P. and Cheeseman, P. (1998). Onboard science data analysis: applying data mining to science-directed autonomy. *IEEE Intelligent Systems*, 13:62–68.

Sugiyama, M. (2006). Active learning in approximately linear regression based on conditional expectation of generalization error. *Journal of Machine Learning Research*, 7:141–166.

Sugiyama, M. and Rubens, N. (2008). Active learning with model selection in linear regression. In *SIAM International Conference on Data Mining*, pages 518–529.

Sung, K. K. and Niyogi, P. (1995). Active learning for function approximation. In *Advances in Neural Information Processing 7 (NIPS-1995)*.

Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning: An Introduction*. MIT Press.

Thrun, S. (1992). The role of exploration in learning control. In *Handbook for Intelligent Control: Neural, Fuzzy and Adaptive Approaches*. Van Nostrand Reinhold.

Thrun, S. (1995). Exploration in active learning. In Arbib, M., editor, *Handbook of Brain Science and Neural Networks*, pages 381–384.

Thrun, S. and Möller, K. (1992). Active exploration in dynamic environments. In Moody, J., Hanson, S., and Lippmann, R., editors, *Advances in Neural Information Processing Systems 4*. Morgan Kaufmann.

Tipton, K. F. (2002). *Enzyme Assays*, chapter 1, pages 1–44. Practical Approach. Oxford University Press, Oxford, England, 2nd edition.

Tokic, M. (2010). Adaptive ϵ-greedy exploration in reinforcement learning based on value differences. In *33rd Annual German Conference on AI*, pages 203–210.

Tran-Thanh, L., Chapman, A., de Cote, E. M., Rogers, A., and Jennings, N. R. (2010). ϵ-First policies for budget-limited multi-armed bandits. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence (AAAI-10)*, pages 1211–1216.

Valdés-Pérez, R. E. (1990). *Machine Discovery of Chemical Reaction Pathways*. PhD thesis, School of Computer Science, Carnegie Mellon University. CMU-CS-90-191.

Valdés-Pérez, R. E. (1994). Conjecturing hidden entities via simplicity and conservation laws: Machine discovery in chemistry. *Artificial Intelligence*, 65(2):247–280.

Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*. Springer.

Wahba, G. (1975). Smoothing noisy data with spline functions. *Numberische Mathematik*, 24:383–393.

Wahba, G. (1978). Improper priors, spline smoothing and the problem of guarding against model errors in regression. *J. R. Statist. Soc. B*, 40(3):364–372.

Wahba, G. (1983). Bayesian "confidence intervals" for the cross-validated smoothing spline. *J. R. Statist. Soc. B*, 45(1):133–150.

Wahba, G. (1990). *Spline Models for Observational Data*, volume 59 of *CBMS-NSF Regional Conference series in applied mathematics*. Society for Industrial and Applied Mathematics, Philadelphia, PA.

Wald, A. (1947). *Sequential Analysis*. Wiley, New York.

Waltz, D. and Buchanan, B. G. (2009). Automating science. *Science*, 324(5923):43–44.

Warmuth, M. K., Liao, J., Rätsch, G., Mathieson, M., Putta, S., and Lemmen, C. (2003). Active learning with support vector machines in the drug discovery process. *J. Chem. Inf. Comput. Sci*, 43:667–673.

Whelan, K. E. and King, R. D. (2004). Intelligent software for laboratory automation. *Trends in Biotechnology*, 22(9).

Williamson, J. (2010). The philosophy of science and its relation to machine learning. In Gaber, M. M., editor, *Scientific Data Mining and Knowledge Discovery*, pages 77–89. Springer.

Woods, M., Shaw, A., Rendell, P., Barnes, D., Pugh, S., Price, D., Pullan, D., and Long, D. (2008). Developing an autonomous science capability for european mars missions. In *10th Workshop on Advanced Space Technologies for Robotics and Automation (ASTRA)*.

Yu, S., Krishnapuram, B., Rosales, R., and Rao, R. B. (2009). Active sensing. In *Proceedings of the 12th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 639–646.

Zauner, K.-P. (2005). Molecular information technology. *Critical Reviews in Solid State and Material Sciences*, 30(1):33–69.

Zauner, K.-P. and Conrad, M. (2001a). Enzymatic computing. *Biotechnol. Prog.*, 17:553–559.

Zauner, K.-P. and Conrad, M. (2001b). Molecular approach to informal computing. *Soft Computing*, 5:39–44.

Zembowicz, R. and Żytkow, J. M. (1991). Automated discovery of empirical equations from data. In *ISMIS '91: Proceedings of the 6th International Symposium on Methodologies for Intelligent Systems*, pages 429–440.

Żytkow, J. M. (1993). Cognitive autonomy in machine discovery. *Machine Learning*, 12:7–16.

Żytkow, J. M. (1997). Robot-discoverer: artificial intelligent agent who searches for knowledge. In *Proceedings of High Technology Symposium, Yamaguchi*, pages 11–18.

Żytkow, J. M. (2000). Automated discovery: A fusion of multidisciplinary principles. In *Canadian Conference on AI*, pages 443–448.

Żytkow, J. M. and Fischer, P. J. (1991). Constructing models of hidden structure. In *Methodologies for Intelligent Systems 6*, pages 441–449.

Żytkow, J. M. and Simon, H. A. (1986). A theory of historical discovery: The construction of componential models. *Machine Learning*, 1:107–137.

Żytkow, J. M. and Zhu, J. (1991). Application of empirical discovery in knowledge acquisition. In *Machine Learning – European Working Session on Learning*, pages 101–117.

Żytkow, J. M., Zhu, J., and Hussam, A. (1990). Automated discovery in a chemistry laboratory. In *Proceedings of the 8th National Conference on Artificial Intelligence*, pages 889–894, Boston, MA. AAAI Press / MIT Press.

Żytkow, J. M., Zhu, J., and Zembowicz, R. (1992). Operational definition refinement: a discovery process. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 76–81. AAAI Press.