Probabilistic Methods for Drug Discovery

DHARMESH MAHENDRAKUMAR MANIYAR

Doctor of Philosophy

ASTON UNIVERSITY

October 2006

ASTON UNIVERSITY

Probabilistic Methods for Drug Discovery

DHARMESH MAHENDRAKUMAR MANIYAR

Doctor of Philosophy (October 2006)

Thesis Summary

The concept of 'chemical space' is of fundamental importance for chemoinformatics and virtual screening research, specially in the early stages of drug discovery. To discover new drugs and to use effectively new chemical tools to understand biology, strategies are required that allow us to systematically explore the 'chemical space' which is huge and high-dimensional since a molecule can be represented by thousands of different descriptors. It is generally thought that high-dimensional representations are too complex for the successful application of many chemoinformatics and virtual screening methods.

This thesis introduces a flexible visual data exploration framework which combines advanced projection algorithms from the machine learning domain with visual representation techniques developed in the information visualisation domain to help a user to explore and understand effectively large multi-dimensional datasets. The advantage of such a framework to other techniques currently available to the domain experts is that the user is directly involved in the data mining process and advanced machine learning algorithms are employeed for better projection. A hierarchical visualisation model guided by a domain expert allows them to obtain an informed segmentation of the input space. Two other components of this thesis exploit properties of these principled probabilistic projection algorithms to develop a guided mixture of local experts algorithm which provides robust prediction and a model to estimate feature saliency simultaneously with the training of a projection algorithm.

Local models are useful since a single global model cannot capture the full variability of a heterogeneous data space such as the chemical space. Probabilistic hierarchical visualisation techniques provide an effective soft segmentation of an input space by a visualisation hierarchy whose leaf nodes represent different regions of the input space. We use this soft segmentation to develop a guided mixture of local experts (GME) algorithm which is appropriate for the heterogeneous datasets found in chemoinformatics problems. Moreover, in this approach the domain experts are more involved in the model development process which is suitable for an intuition and domain knowledge driven task such as drug discovery.

We also derive a generative topographic mapping (GTM) based data visualisation approach which estimates feature saliency simultaneously with the training of a visualisation model. The approach not only provides a better projection by modelling irrelevant features with a separate noise model but also gives feature saliency values which help the user to assess the significance of each feature. As demonstrated in this thesis, this approach has many applications in the drug discovery domain.

The approaches proposed in this thesis are evaluated on real-life datasets from chemoinformatics.

Keywords: Visual data mining, guided mixture of local experts, feature selection, machine learning, information visualisation, chemoinformatics

Acknowledgements

First and foremost I would like to thank my supervisor, Prof. Ian T. Nabney, for his guidance, criticism, support and freedom he gave me in deciding research directions, which probably helped me most in developing my view of scientific research. I thank him for all of the absorbing discussions and valuable feedback whenever needed. His supervision has made my time as a PhD student a very rewarding experience, with many challenges and hard work, but also a lot of fun.

Many thanks to Bruce Williams and Philip Laflin for providing all the required support, for those perfectly organised meetings and visits at *Pfizer*, for answering all sort of my queries and for actively taking part in the project and making the close collaboration possible. I also thank other scientists at Pfizer for their valuable time for useful discussions and feedback.

I thank Pfizer and Overseas Research Students (ORS) Awards for the financial support.

I thank Prof. Saad for his useful suggestion of evaluating projection results using KL-divergence (Section 3.6.1) during my first year viva. The method has been a useful way of evaluating the quality of projections. I thank Dr. Cornford for pointing me to the Bayesian Committee Machine (BCM) and the emulator technology which will be my next course of research.

My work has benefited greatly from discussions with many of the members of the Neural Computing Research Group (NCRG). I very much enjoyed the friendly, collaborative and productive work environment at NCRG. Thank you Prof. Lowe and Prof. Saad for facilitating such a stimulating research environment. A special thank to Vicky Bond for her prompt assistance for different official documents I required from time to time and for her excellent administrative skills.

Many thanks to all my fellow PhD students and 2003 MRes PANN batchmates for making my time in the labs enjoyable and for the useful discussions. A special thank to Rémi Barillec, Dan Woodcock and Stephen d'Aguiar for sharing a great time in the PhD office with me and for their patience during my outrage about a few things, ranging from cricket to visa issues.

Many thanks to all my friends at Aston and all over the world, who stayed connected over the Internet, for helping me get adjusted to the life in England and for listening to my endless comments about the unfavourable local climate. A special thank to LS-G6 flatmates and all Rota members for all the good time we had together.

I am grateful to my parents and my brother for their unconditional love and support. Without them I would not have been here. In spite of being so far, their frequent emails and phone calls have successfully managed to not let me feel alone at any time.

Finally, I would like to thank Ariane, my fiancée, for all her love, support, patience and understanding, and her ability to always put a big simile on my face. You are a gem!

Contents

List of Figures

List of Tables

Declaration

This thesis describes the work carried out between September 2003 and September 2006 in the Neural Computing Research Group at Aston University under the supervision of Prof Ian T. Nabney.

The work reported in this thesis has been composed by myself and has not, nor any similar dissertation, submitted in any previous application for a degree.

Chapter 1

Introduction

The concept of chemical space is of fundamental importance for chemoinformatics[1] research. Chemical space is vast and heterogeneous [Dobson, 2004]. The total number of possible small organic molecules that populate 'chemical space' has been estimated to exceed 10^{60} and each of these molecules can be represented by thousands of different descriptors [Bohacek et al., 1996; Karelson, 2000]. It is generally thought that high-dimensional space representations are too complex for the successful application of many data exploration, compound classification and decision making methods [Godden and Bajorath, 2006]. To discover new drugs and to effectively use new chemical tools to understand biology, strategies are required that allow us to systematically explore 'chemical space'. In the following section we highlight some of the issues in the drug discovery process which has motivated the work presented in this thesis.

1.1 The motivation

Drug discovery continues to be a challenging area. Drug discovery and development is a critical but lengthy and costly process, taking an average of 15 years and US$ 880M to generate a successful medicine [Flanagan et al., 2001]. Currently, quick and cost effective drug discovery is a major focus of competition between pharmaceutical companies. When no ligand[2] for a particular protein is known, a search in chemical space is often undertaken in the hope of identifying compounds that bind to the protein with reasonable affinity. The recent advances in decoding of the human genome sequence, which help in identifying biological targets[3] for disease, have introduced many new biological targets [Sanseau, 2001]. Developing

[1] The combination of chemical synthesis, biological screening, and data-mining approaches used to guide drug discovery and development.

[2] A molecule (compound), or a molecular group that binds to another chemical entity.

[3] A biological target is an enzyme, receptor or other protein that can be modified by an external stimulus.

drugs for these new targets as quickly as possible is the current research thrust for major pharmaceutical companies.

The dominant technique for the identification of new 'leads'[4] in drug discovery is the physical screening (high-throughput screening – HTS) of large libraries of chemicals against a biological target [Fox et al., 1999]. Despite of its wide use, high attrition rates in the later stages of drug discovery have raised questions about the viability of the high-throughput screening paradigm on its own [Lahana, 1999; Handon, 2002; Englebienne, 2005]. It is being recognised that increasing the quality of screening libraries and better triage and understanding using HTS results, rather than their quantity, is likely to be an important determinant for the identification of active compounds that have a chance to make it through the drug discovery pipeline [Bajorath, 2001; Handon, 2002; McGovern et al., 2002; Gribbon and Sewing, 2005]. If the results are only triaged by potency, which was until recently the practice in pharmaceutical industry, then heavier compounds will be selected as hits[5] to follow up. Heavier hits will lead to heavier drugs which lead to absorption/half-life problems in the later stages [Brennan, 2000]. Biological activity data of chemical compounds collected using technologies such as HTS and the availability of detailed physicochemical properties of chemical compounds could be used to mine useful information [Good et al., 2000]. One of our main aims in this research is to provide an effective data exploration tool which helps screening scientists (e.g. biologists, chemists) to understand large datasets better and take effective decisions in an informed way.

Given the vast size of organic chemical space drug discovery cannot be reduced to a simple 'synthesise-and-test' lottery. Even though technologies such as HTS can expose the target to a large number of chemical compounds, it is financially and time-wise impossible to screen all the compounds in a library for a target. A complementary approach, known as virtual screening, has become popular: this means to screen computationally large number of compounds from the chemical space to find compounds that complement targets of known structure, and experimentally test those that are predicted to bind well [Clark and Pickett, 2000; Kitchen et al., 2004; Shoichet, 2004]. Although these two approaches to ligand discovery are distinct, they can be used together to enhance the chances of finding an active compound [Bajorath, 2002a]. In particular, within the pharmaceutical industry, the use of computational models as a 'filter' to select compounds for chemical synthesis from very large virtual libraries for

[4] A representative of a compound series with sufficient potential (as measured by potency, selectivity, pharmacokinetics, physicochemical properties, absence of toxicity and novelty) to progress to a full drug development programme.

[5] Library component whose activity exceeds a predefined, statistically relevant threshold.

experimental screening (HTS) and as a tool for understanding complex datasets have become increasingly common [Gasteiger, 2003; Lahana, 2004; Chen, 2006]. Data mining and prediction techniques are used to find the hidden gems in the HTS hits and in the chemical libraries: maybe not showing enough potency (because of noisy HTS operation), but very attractive otherwise. One of the major challenges in this area is to build effective prediction models which can cope with the diversity in chemical space [Bajorath, 2002b]. We address this issue by developing a mixture of local experts model which works locally in the data space and also involves the domain expert in the model development process.

Typical datasets found in drug discovery involve many descriptors. The question of which descriptors are important is always of a significant interest. Screening scientists would like to know the significance of these descriptors during the modelling process so that compounds can be optimised by concentrating on the descriptors which are more significant for a particular behaviour. In this thesis, we introduce an approach to estimate significance of the descriptors during the training of a data visualisation model.

In the next section, we briefly summarise the major contributions of this thesis.

1.2 Overview of this thesis

In this thesis we introduce a flexible visual data exploration framework for effective data mining (in chapter 3), new mixture of local experts approaches (in chapter 4) and an algorithm which estimates feature saliency during the training of a data visualisation model (in chapter 5). The following is a brief introduction to each of these topics; further details are in the corresponding chapters.

1.2.1 An integrated visual data exploration framework

The exploration of heterogeneous information spaces requires suitable mining algorithms as well as effective visual interfaces. Most existing systems, e.g. SpotFire, Tripos, concentrate on information visualisation and interaction techniques using basic statistical and machine learning algorithms such as PCA, factor analysis and multi-dimensional scaling. Though visual and interaction methods have been helpful, for improved understanding of a large high-dimensional dataset, an effective projection onto a lower-dimension (2D or 3D) manifold is required. In this chapter we introduce a flexible visual data exploration framework which combines advanced projection algorithms developed in the machine learning domain and visual representation techniques developed in the information visualisation domain. The advantage of such a framework is that the user is directly involved in the data mining process. We integrate principled projection methods, such as the generative topographic mapping (GTM) and hierarchical GTM (HGTM), with powerful visual techniques, such as magnification factors, directional curvatures, parallel coordinates and billboarding, to provide a visual data exploration framework. Results on a real-life chemoinformatics dataset using GTM are promising and have been analytically compared with the results from the traditional projection methods. It is also shown that the HGTM algorithm provides additional insight and good segmentation for large datasets. The computational complexity of these algorithms is also discussed to demonstrate their suitability for the visual data exploration framework (chapter 3).

1.2.2 Guided mixture of local experts

A single global predictive model cannot capture the full variability of a data space such as chemical space since the mapping in different regions of the data space may vary. Probabilistic hierarchical visualisation techniques can provide an effective soft segmentation of an input space by a visualisation hierarchy whose leaf nodes represent different regions of the input space. We use this soft segmentation to develop new guided mixture of local experts approaches which are appropriate for a heterogeneous dataset such as found in chemoinformatics problems. Moreover, in this approach the domain experts are directly involved in the model development process which is suitable for an intuition and domain knowledge driven task such as drug discovery. The performance of the algorithms on two different real-world datasets from chemoinformatics is benchmarked against conventional local models and popular global models (chapter 4).

1.2.3 Data visualisation with simultaneous feature selection

Data visualisation algorithms and feature selection techniques are both widely used in chemoinformatics/bioinformatics but as distinct analytical approaches. We derive a generative topographic mapping based data visualisation approach which estimates feature saliency simultaneously with the training of the visualisation model. Such saliency measures the importance of a feature on the definition of the cluster structure yielded by a data visualisation model. The approach not only provides a better projection by modelling irrelevant features with a separate noise model but also gives feature saliency values which help the user to assess the significance of each feature. We compare the quality of projection obtained using the new approach with the projections from traditional GTM and self-organising map (SOM) algorithms. The results obtained on a synthetic and a real-life chemoinformatics dataset demonstrate that the proposed approach successfully identifies feature significance and provides coherent (compact) projections (chapter 5).

1.3 Publications based on work in this thesis

This thesis gathers and complements the material in following publications. The chapter numbers given refer to the chapter of this thesis where the main content of the corresponding paper can be found.

- [Maniyar et al., 2006]: D. M. Maniyar, I. T. Nabney, B. S. Williams, and A. Sewing. Data visualization during the early stages of drug discovery. *Journal of Chemical Information and Modelling*, 46(4):1806–1818, 2006 (Chapter 3).
- [Maniyar and Nabney, 2006c]: D. M. Maniyar and I. T. Nabney. Visual data mining using principled projection algorithms and information visualization techniques. *In Proceeding of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* pp 643–648, 2006 (Chapter 3).
- [Maniyar and Nabney, 2005]: D. M. Maniyar and I. T. Nabney. Guiding local regression using visualisation. *In Deterministic and Statistical Methods in Machine Learning, LNAI, Springer-Verlag*, 3635:98–109, 2005 (Chapter 4).
- [Maniyar and Nabney, 2006a]: D. M. Maniyar and I. T. Nabney. Data Visualization with Simultaneous Feature Selection, *In Proceeding of the IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology.* pp 156–163, 2006 (Chapter 5). [It won 'the best student paper award']
- D. M. Maniyar and I. T. Nabney. Exploiting Probabilistic Visualization Algorithms, *IEEE Transactions on Knowledge and Data Engineering.* In submission, (chapter 4 and chapter 5).

A longer version of the paper [Maniyar and Nabney, 2006c] was invited to be published in:

- [Maniyar and Nabney, 2006b]: D. M. Maniyar and I. T. Nabney. Visual Data Mining: Integrating Machine Learning with Information Visualization. *In Proceeding of the 7th International Workshop on Multimedia Data Mining.* pp 63–72, 2006 (chapter 3).

1.4 Notation and conventions

In the mathematical notation, the convention will be that an italic typeface indicates scalar values, e.g. t_{nd}, x, while bold typeface indicates vectors and matrices, the former using lower case symbols, e.g. $\mathbf{t}_n, \mathbf{x}$, and the latter using upper case symbols, e.g. $\mathbf{T}, \mathbf{X}$. Exceptions to this convention do appear, but they will be explicitly pointed out. The symbols used for the most commonly occurring quantities in this thesis are listed in Table 1.1.

N	number of data points
n	data label
D	number of data dimensions (features)
d	feature label
L	number of latent (projection) dimensions
l	latent dimension label
M	number of components in a mixture model
$\mathbf{T}$	dataset stored as a $N \times D$ matrix
$\mathbf{I}$	an identity matrix
$\mathbf{W}^\top$	transpose of matrix $\mathbf{W}$
P	probability mass function
p	probability density function
$\mathcal{L}$	likelihood

Table 1.1: Symbols used.

1.5 Thesis structure

The thesis is organised as follows:

Chapter 2: We provide an introduction to the drug discovery process and major phases during the early stages of pharmaceutical research. Then we review current technologies and challenges in experimental and virtual screening in pharmaceutical research. The chapter ends with a discussion on the challenges we aim to address in this thesis.

Chapter 3: Here the importance of data visualisation and exploration is first discussed. We then describe selected projection and information visualisation techniques. Then the visual data exploration framework is introduced with a description of the software we have developed. Finally the results of different projection algorithms using the visual data exploration framework are discussed and concluded.

Chapter 4: In this chapter we introduce the guided mixture of local experts algorithm. The requirement of a local approach is motivated and related work is discussed. A few popular global and local models are also briefly introduced. Later in the chapter, the results of our approach with the results of other global and local regression algorithms are compared. The chapter finishes with a discussion and conclusion.

Chapter 5: First we motivate the reader about why treating data visualisation and feature selection simultaneously is not only a synergistic but also a logical step forward. Then we introduce the novel approach for estimating feature saliency during the training of a probabilistic data visualisation algorithm. The results on a synthetic and a real-life dataset from chemoinformatics are presented and discussed thoroughly.

Chapter 6: First the work presented in this thesis is summarised and final conclusions are drawn. Then we discuss the directions for the future research work.

Appendix A: The derivation of the M-step of the EM algorithm for GTM-FS is presented here.

Chapter 2

Screening in Drug Discovery

Discovering a new drug is a complex and lengthy task. Although, historically the discovery of novel drugs has been led by biology, chemistry and pharmacology, recent advances in computational methods and the availability of large datasets have made extensive data analysis and computational modelling an increasingly important part of the early pharmaceutical research. Currently, A report on *in silico*[1] technology estimated that by 2006 ~10% of pharmaceutical R&D expenditure would be on computer simulation and modelling, a figure set to rise to 20% by 2016 [Anderson, 2002]. It seems clear that as a whole, the pharmaceutical R&D landscape will change further. This chapter provides a brief introduction to the general process of the early stages of pharmaceutical research and various molecule-screening methods.

2.1 The drug discovery process

The modern drug discovery process (see Figure 2.1) can be abstractly divided into 2 phases: early phase and late phase. Target and lead discovery are the main components of early pharmaceutical research. Most of chemoinformatics applications, and also this thesis, are focused on the computational methods used during lead discovery and optimisation.

Drug-discovery programs typically start with the identification of suitable drug target (proteins such as receptors, enzymes and ion channels) which is the causes for a disease. During the stepwise process of target validation, a sufficient level of 'confidence' has to be established that the target is of relevance to the disease under study and that modulation of the target will lead to effective disease treatment. Once the target has been validated, its modulators are identified. Such modulators can be agonists or antagonists in the case of

[1] Refers to modelling research conducted using computers in conjunction with informatics capabilities.

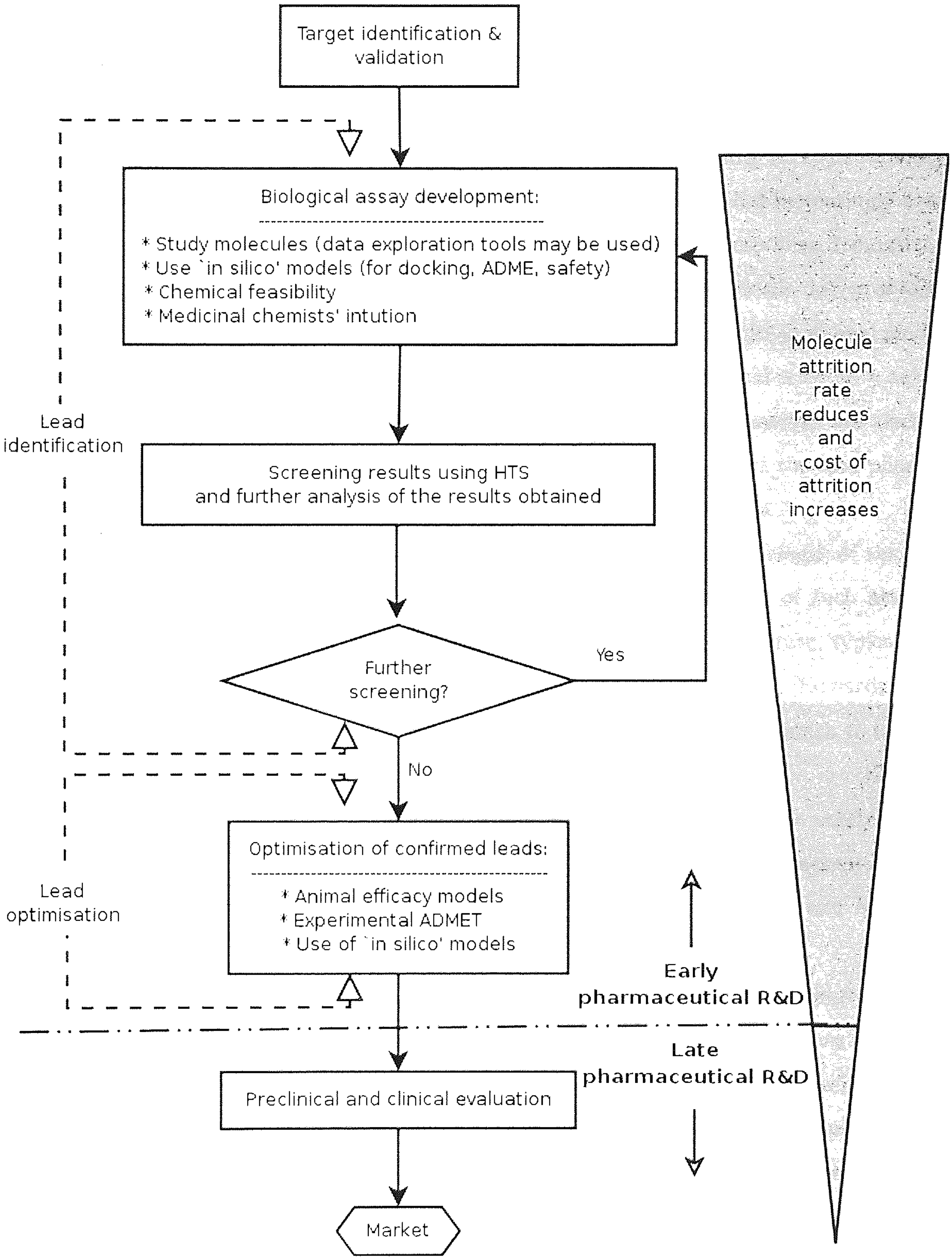

Target identification & validation
Biological assay development:
* Study molecules (data exploration tools may be used)
* Use `in silico' models (for docking, ADME, safety)
* Chemical feasibility
* Medicinal chemists' intution
Lead identification
Screening results using HTS and further analysis of the results obtained
Further screening?
Yes
No
Optimisation of confirmed leads:
* Animal efficacy models
* Experimental ADMET
* Use of `in silico' models
Lead optimisation
Early pharmaceutical R&D
Late pharmaceutical R&D
Preclinical and clinical evaluation
Market
Molecule attrition rate reduces and cost of attrition increases

Figure 2.1: The process of pharmaceutical research, often referred to as the 'drug discovery process', can roughly be divided into an early and late phase. The early phase is mainly represented by target and lead discovery, whereas the later phase deals mainly with clinical evaluation and development.

receptors, activators or inhibitors of enzymes, and openers or blockers of ion channels.

Once the target are understood in detail and modulators are identified, the lead identification step starts with the design and development of a suitable biological assay to monitor the target under study. Various *in vitro*[2] and *in silico* methods are used during such biological assay design and development. Subsequently, high-throughput screening (HTS) technology is used to expose the target to a large number of chemical compounds that increasingly come from high-speed parallel and combinatorial synthesis[3]. Active compounds that demonstrate dose-dependent target modulation are called 'lead' compounds when a certain degree of selectivity[4] for the target under study can be shown and the first positive results in animal models are obtained. Such lead compounds are optimised in terms of potency[5] and selectivity as well as physicochemical properties, and their pharmacokinetic[6] and safety features are assessed before they can become candidates for drug development. At this point the late phase of drug discovery process starts, this is dominated by extensive clinical trials.

As shown in Figure 2.1, it is important to note that during the later stages of the drug discovery process, the compound attrition rate decreases while the cost of such attrition increases. The drug discovery process is characterised by a high attrition rate; typically up to 76% between target identification and an investigational new drug filing [Edwards et al., 2002] and ~90% by the end of clinical trials [Dimasi, 2001]. A rational approach to increase the efficiency and reduce the cost of pharmaceutical R&D is to reduce the attrition rate in the costly downstream stages by increasing the attrition rate in the less costly, earlier stages of the process – a 'fail early, fail cheap' strategy that has been widely accepted in the pharmaceutical industry [Smith and Waterbeemd, 1999; Lin et al., 2003; Yu and Adedoyin, 2003].

A successful drug is a combination of biological activity and drug-like properties [Selick et al., 2002]. A number of properties of small molecules are important for their use as a building block or potential drug, in addition to their ability to bind potently and specifically to a particular protein target. Such 'drug-like' properties include their ability to cross biological

[2] Biological study which is carried out in an artificial environment (in the laboratory) outside a living organism.

[3] Using a combinatorial process to prepare sets of compounds from sets of building blocks combined in many different ways.

[4] The word selectivity describes a drug's ability to affect a particular cell population in preference to others.

[5] An expression of the activity of a drug, in terms of the concentration or amount needed to produce a defined effect.

[6] Process of the uptake of drugs by the body, the bio-transformation they undergo, the distribution of the drugs and their metabolites in the tissues, and the elimination of the drugs and their metabolites from the body. Both the amounts and the concentrations of the drugs and their metabolism are studied [taken from IUPAC Compendium].

membranes, their chemical stability, and their solubility in water and dimethyl sulphoxide (a common organic solvent). Until recently, major proportion (~40%) of development compounds failed to reach the market due to poor 'drug-like' properties [Venkatesh and Lipper, 2000]. There has been much interest in the pharmaceutical industry in engineering 'drug-like' properties and discarding candidate compounds that are unlikely to be effective drugs, even before they are synthesised [Waterbeemd et al., 2001; Kerns and Di, 2003]. Now property screening in parallel with activity screening, which allows medicinal chemists to optimise biological activities as well as drug-like properties, is widely accepted and implemented in pharmaceutical companies.

Although most of the processes of early pharmaceutical research rely predominantly on experimental work in the laboratory, use of *in silico* methods, particularly to evaluate 'drug-like' properties, has become increasingly important to speed up the drug discovery process and further decrease late-stage attrition [Anderson, 2002].

In the next two sections we describe two distinct but complementary approaches for screening: experimental screening and virtual screening.

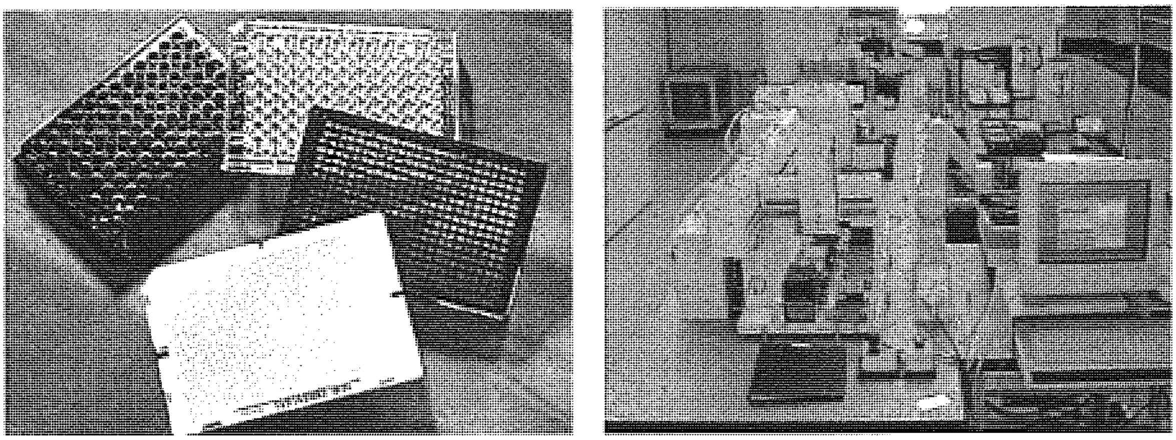
2.2 Experimental screening

Advanced experimental, *in vitro*, technologies are used during the "Lead identification" and "Lead optimisation" stages to find molecules that bind the target and possess 'drug-like' properties. Experimental screening efforts accounts for a substantial part of the total research and development expenditures of the pharmaceutical industry [Handon, 2002]. According to their functions, experimental screening technologies can broadly categorised into two categories: evaluating biological activity and evaluating drug-likeness.

2.2.1 Evaluating biological response

Currently, physical screening of large libraries of chemicals against a biological target (high-throughput screening – HTS) is the dominant technique for the identification of new lead compounds in drug discovery. In a high-throughput screen, many different molecules are evaluated in the same biological test for their effect on a protein or cellular process. The term 'screen' is used to indicate that many different chemicals are tested but only a small number of them are expected to be active. The term 'high-throughput' is used to indicate that many chemicals are put through this process in a short period of time, typically 10^6 chemical compounds a day [Roberts, 2000]. Because of this capability, HTS technologies have become an important part of the drug discovery process.

HTS is highly automated by the use of sophisticated laboratory equipments, robotics and chemical analysis techniques. Figure 2.2 displays an example of different kinds of microplates[7] used during an assay and a robot in action during HTS. A good review of HTS technology and advances in the field is available in [Liu et al., 2004].



(a) 96, 384 and 1536 well microplates

(b) A robot in action during HTS

Figure 2.2: High-throughput screening.

The selection of compounds to screen (biological assay) during an HTS campaign is based on understanding of drugs for similar targets, experience of the screening scientists (medicinal chemists, biologists, etc.) and computational methods (as discussed in Section 2.3). Understanding the large amount of information available is critical to good assay development. Effective assay development continues to present major bottlenecks for biological screening [Bajorath, 2001; Fox et al., 2002]. In addition, it is also being recognised that increasing the quality of screening libraries and HTS assays, rather than their size, is likely to be an important determinant for the identification of active compounds that have a chance to make it through the drug discovery pipeline [Fox et al., 1999; Handon, 2002]. Chemoinformatics tools are increasingly used to handle the vast amounts of data from HTS [Oprea et al., 2003] and to bring rigour to the process of looking for genuine leads. How *in silico* methods are used to develop effective assays is briefly discussed in Section 2.3.1.

[7] A standardised plastic tray with "wells," or depressions, for holding small quantities of material.

2.2.2 Evaluating drug-likeness

Nearly a decade ago, an analysis, see Figure 2.3, of the main reasons for attrition in drug development showed half of all failures were attributed to poor drug-likeness (pharmacokinetics: 39% and animal toxicity: 11%) [Kennedy, 1997]. This analysis clearly indicated that it is critical to focus on evaluating 'drug-likeness' properties as early as possible in the drug-discovery process.

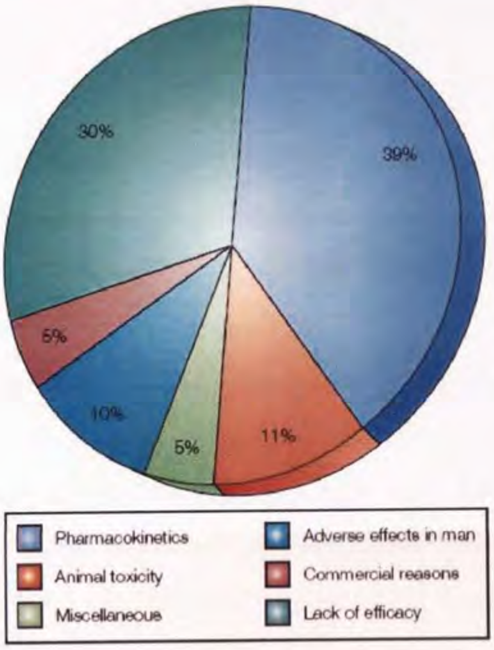


Figure 2.3: An analysis of the main reasons for attrition in drug development (Figure adapted from [Waterbeemd and Gifford, 2003].)

Based on such analysis, Lipinski and others rediverted drug discovery back to the principles of medicinal chemistry as key to reducing attrition in late stage [Lipinski et al., 1997]. More recently, the need to go further in defining what makes a good lead has been recognised with the concept of drug-likeness. Drug-likeness implies cut-off values in the physicochemical profile of chemical libraries such that they have reduced complexity (e.g. MW below 400) and other more restricted properties.

To evaluate drug-likeness effectively, we require data/information about phisicochemical, pharmacokinetic (ADME)[8]) and safety (eg. Toxicity – Tox) properties of compounds. The goal of experimental, *in vitro*, assays is to provide, with reasonable accuracy, a preliminary prediction of the *in vivo*[9] behaviour of a compound to assess its potential to become a drug

[8] Characteristics of a substance in terms of Absorption, Distribution, Metabolism, Excretion.

[9] Biological study which takes places within a living biological organism.

(drug-likeness) [Bachmann and Ghosh, 2001]. In recent years, combinatorial chemistry and high-throughput screening have significantly increased the number of compounds for which early data on ADME-Tox are needed, which has in turn driven the development of a variety of medium and high-throughput *in vitro* ADME-Tox screens [Kerns and Di, 2003]. Protocol simplification and a widespread use of microtiter plate formats have made it possible for many assays to be automated using robotic systems. Detailed description of such laboratory techniques is out of the scope of this thesis; a good review is available in [Yu and Adedoyin, 2003].

2.3 Virtual screening

Screening compounds with the use of computational methods is broadly known as virtual, *in silico*, screening. Virtual screening of compound databases is currently one of the most popular chemoinformatics applications in pharmaceutical research. It has been argued that virtual screening techniques complementing experimental screening, particularly HTS, can make the drug discovery process more efficient [Bajorath, 2002a; Oprea and Matter, 2004]. In this section we provide an overview of popular virtual screening techniques and discuss them in the context of the experimental screening techniques described in the previous section. According to their functions, virtual screening methods can be broadly divided in two categories: library design and predicting drug-likeness.

2.3.1 Library and assay design

In silico, techniques are widely used in compound library and assay design to reduce the number of compounds to be tested, and two basic applications can be distinguished: diversity and structure-based design.

Diversity-based library design

When no structural information about the target is available, the assay should be designed to test a diverse set of molecules. Diversity design aims to select a smaller sub-library from a larger compound library in such a way that the full range of chemical diversity is best represented [Gorse and Lahana, 2000]. The different computational methods for compound selection are mainly based on compound similarity clustering, grid-like partitioning of chemical space or the application of genetic algorithms [van Drie and Lajiness, 1998]. The results of such *in silico* diversity selections are smaller sub-libraries of manageable size with a high degree of chemical diversity that are then subjected to HTS *in vitro*.

Structure-based library design

Structure-based library design is biased by structural requirements for activity on a particular target and needs prior information of the target structure (e.g. using X-ray crystallography or nuclear magnetic resonance). The goal is to select from existing compound libraries or to design compounds with three-dimensional complementarity (i.e. shape, size and physico-chemical properties) to the target-binding site. In the latter case, new approaches can directly guide the design of virtual combinatorial libraries, which are first screened *in silico* for target complementarity, thus reducing the number of compounds that will have to be synthesised and tested *in vitro*. It can be expected that the hit-rate (rate of compounds found to be active on the target under study in a dose-dependent manner) of such focused libraries will be higher than that of diversity screening. Based on virtual screening, it is relatively easy to exclude parts of libraries from further consideration that are clearly not compatible with a targeted binding site or chemically too distinct from known actives. A good review of some of the popular structure-based screening (QSAR – quantitative structure activity relationships) techniques is given in [Bajorath, 2002b; Lyne, 2002; Balen et al., 2004]. Programs such as AutoDock, DOCK, FlexX, FRED, GOLD and Glide can be used to examine millions of compounds for their propensity to interact with the target protein, and the relative fit of each candidate scored [Ewing et al., 2001; Osterberg et al., 2002; Kramer et al., 1999; Halgren et al., 2004].

Both diversity and structure-based screening can be performed in an iterative manner. In this case, the results of *in vitro* HTS are analysed *in silico* to derive rules that can be used for the rational selection of further molecules to be tested *in vitro*.

Although *in silico* library design is a useful emerging technology, current success rates are low because it is difficult to predict how small molecules will interact with a protein; there is flexibility in the torsion angles in both the protein and small molecule, causing uncertainty regarding the three-dimensional structure of both. Improvements in the predictive accuracy of such programs will affect virtual screening, and so the discovery of novel protein ligands [Stockwell, 2004].

2.3.2 Predicting drug-likeness

As stated earlier, poor pharmacokinetics and toxicity are significant causes of costly late-stage failures in drug development, it has become widely appreciated that these properties should be considered as early as possible in the drug discovery process [Waterbeemd and Gifford, 2003]. In addition to their use in the 'Lead optimisation' stage, the computational techniques described here can be used early on to select a subset of compounds for screening or to guide combinatorial library design.

Though in recent years throughput provided by the simplified *in vitro* ADME-Tox assays has increased, throughput capacity remains low in comparison with that of HTS activity assays or combinatorial chemistry, consequently limiting the application of these assays to only a fraction of the compounds evaluated in discovery. The need for increased ADME-Tox throughput to fully meet the demands of discovery has led to renewed and increasing interests in computational, *in silico*, models [Selick et al., 2002; Yu and Adedoyin, 2003; Waterbeemd and Gifford, 2003].

In silico approaches to predict pharmacokinetic parameters (ADME) were pioneered by Lipinski et al. [1997]. By studying the physicochemical properties of more than 2000 drugs from the world drug index (WDI) database, which can be assumed to have entered Phase II human clinical trials (and therefore must possess drug-like properties), the so-called 'rule-of-five' was derived to predict oral bioavailability (intestinal absorption) of a compound. It identifies several key properties that should be considered for small molecules that are intended to be orally administered. These properties are: molecular mass less than 500 daltons; number of hydrogen-bond donors less than 5; number of hydrogen-bond acceptors less than 10; calculated octanol/water partition coefficient (an indication of the ability of a molecule to cross biological membranes) less than 5. In general, such studies point to the most important physicochemical and structural properties characteristic of a good drug in the context of our current knowledge. These properties are then typically used to construct predictive ADME models and form the basis for what has been called property-based design [Ekins et al., 2000; Waterbeemd et al., 2001; Podlogar et al., 2001]. Prediction approaches ranging from simple multiple linear regression (LR) to machine learning techniques are now being applied to the analysis of ADME data [Livingstone and Manallack, 2003; Waterbeemd and Gifford, 2003; Xiao et al., 2005]. Data mining and machine learning methods such as principal component analysis, artificial neural networks (ANN), self-organizing maps (SOM), classification and regression trees (CART), and genetic algorithms (GA), originally developed and used in other fields, are now also successfully being used for 'drug-likeness' prediction [Sadowski, 2000; Schneider, 2000; Hou and Xu, 2002; Bai et al., 2004].

For compounds targeted at the central nervous system, another important aspect is blood–brain barrier (BBB) penetration. On the basis of predictive models described by Abraham et al. [1994], a simple two-variable equation has been devised that allows rapid automated *in silico* screening of (virtual) libraries for compounds with a potential to cross the BBB [Clark and Pickett, 2000]. Predicting the toxicity of compounds, another important aspect, has been reviewed in [Dearden, 2003].

Sufficient high-quality and reliable data are not yet available to develop robust models; thus the predictive ability of the underlying models is limited and needs further development. As a result of the availability of experimental data in the literature, considerable effort has gone into the development of models to predict physicochemical properties relevant to ADME, such as lipophilicity. However, despite its importance, the prediction of pharmacokinetic properties such as clearance, volume of distribution and half-life directly from molecular structure is making slower progress owing to a lack of published data. Similarly, the prediction of various aspects of metabolism and toxicity is also underdeveloped.

2.4 Challenges and opportunities

Though the last decade has seen a lot of development in pharmaceutical research and particularly in *in silico* techniques, there are still many challenges ahead. In this section we highlight the challenges in *in silico* research which we aim to address with the work presented in this thesis.

Data visualisation and exploration: Good understanding of biological activity results and physicochemical properties can help to guide the design of future assays and libraries. Moreover, exploring chemical space and understanding the datasets for the problem at hand are important aspects to avoid being misled by noisy results from experimental and *in silico* methods. For example, one of the problems with the new types of organic compound that are now being explored as drugs is that they may be extremely potent when tested against isolated targets in the laboratory environment, but within the complex cellular milieu, they might interact with cellular components other than the desired target. That is why understanding and exploring screening results with other descriptors is important rather than just relying on an activity measures.

Obtaining such an overview and understanding of large high-dimensional datasets requires effective projection and exploration techniques. The use of powerful visualisation tools, such as SpotFire[10], is common in pharmaceutical research. Though these software are quite useful, the lower-dimensional projection obtained using the traditional projection algorithms provided in them is not satisfactory for large high-dimensional datasets. In this thesis we introduce a visual data exploration framework which identifies appropriate principled projection algorithms to obtain an effective projection of higher-dimensional dataset on to a lower-dimensional space and integrates advanced visual exploration facilities to support domain experts in data exploration.

Effective prediction models: A bottleneck for effective prediction of 'drug-likeness' properties is that good predictive accuracy the predictability of regression models is generally limited to the chemical space that is covered by the compounds in the training set or those fairly close to them. The use of a more diverse set of chemical molecules in model development and computational models which can cope with such diversity should ensure a better predictability and wider applicability [Yu and Adedoyin, 2003]. Most of the prediction models used in drug discovery are global models. A single global model usually fails to model all the diversity in different region of input space. Moreover, many prediction models used in pharmaceutical industry are 'black boxes' for most people so there is a need to involve domain experts (chemist, biologists, screening scientists) in the model development process to increase their confidence in the results. In this thesis we introduce a guided mixture of local experts algorithm which works well with heterogeneous datasets and involves the domain experts in the model development process.

Estimation of descriptor importance: The screening scientists have to consider many descriptors for a successful drug discovery operation. Though, most of the times, the selection of the descriptors is based on their domain knowledge and experience, the question of which descriptors are important has always been of a significant interest in pharmaceutical research. Moreover, good clustering and prediction often depend crucially on the right molecular descriptors for the problem at hand. It would be interesting if a model can predict relevance of the descriptors. In this thesis we present a data visualisation with simultaneous feature significance determination approach.

[10] Spotfire: `http://www.spotfire.com/`

Chapter 3

An Integrated Visual Data Exploration Framework

Multi-dimensional compound optimisation is a new paradigm in the drug discovery process, yielding efficiencies during early stages, and reducing attrition in the later stages of drug development. The success of this strategy relies heavily on understanding multi-dimensional data about compounds and extracting useful information from it. The exploration of heterogeneous information spaces requires suitable mining algorithms as well as effective visual interfaces. In this chapter we introduce a flexible visual data exploration framework which combines advanced projection algorithms from the machine learning domain and visual representation techniques developed in the information visualisation domain. The advantage of such an framework is that the user is directly involved in the data mining process. Results on several real-life case studies using principled probabilistic projection algorithms, such as the generative topographic mapping, are promising and have been analytically compared with the results from the traditional projection methods. It is also shown that a hierarchical visualisation approach, such as the hierarchical GTM algorithm, provides additional value for large datasets. The research is carried out with the domain experts (screening scientists) at Pfizer. The computational complexity of these algorithms is analysed to demonstrate their suitability for the visual data exploration framework where interaction and speed of the algorithms are important parameters.

3.1 Introduction

Today, the data available to tackle many scientific challenges is vast in quantity and diverse in nature. The wide availability of ever-growing data sets from different domains has created a need for effective knowledge discovery and data exploration. An important component of effective data exploration is to obtain "natural" groupings in a large multivariate dataset. In a recent review on "Statistical Challenges in Functional Genomics", Sebastiani et al. [2003], stated "The newly born functional genomic community is in great need of tools for data analysis and visual display of the results". Since it is difficult for a human to visualise data in more than three dimensions, effective visualisation which gives better clustering (grouping) of high-dimensional data onto lower-dimensional space is desirable for an effective data exploration tool. Here, we use the term *visualisation* to mean any method of projecting data into a lower-dimensional space in such a way that the projected data keeps most of the topographic properties (i.e. 'structure') and makes it easier for the users to interpret the data to gain useful information from it. Exploration of complex information spaces is an important research topic in many fields, including computer graphics, data mining, machine learning, and other areas of statistics, as well as database management and data warehousing. In the last decade, many new machine learning techniques have been proposed for effective projection.

Machine learning is often split into three categories: supervised learning, where a data set is split into inputs and outputs and the goal is to relate inputs to associated outputs; reinforcement learning, where typically a reward is associated with achieving a set goal, and unsupervised learning where the objective is to understand the structure of a data set while there might be no specific target for a pattern. One approach to unsupervised learning is to represent the data, $\mathbf{T}$, in some lower dimensional embedded space, $\mathbf{X}$. In a probabilistic model the variables associated with such a space are often known as latent variables. In this chapter our focus will be on those machine learning methods that represent the data in this latent space.

For a complex large high-dimensional dataset, where clear clustering is difficult and grouping of data points is in soft clusters (overlaping clusters) or separate clusters of similar types, using visual aids to explore further can reveal insight that may prove useful in data mining. For data mining to be effective, it is important to include the domain expert in the data exploration process and combine the flexibility, creativity, and general knowledge of the domain expert with automated machine learning algorithms to obtain useful results [Keim, 2002].

The principal purpose of the visual aids developed in the information visualisation domain is to present data in a visual form provided with interactive exploration facilities, allowing the domain expert to get insight into the data, draw conclusions, and understand the structure of the data. Visual representation techniques on their own cannot entirely replace analytic nonvisual mining algorithms to represent a large high-dimensional dataset in a meaningful way. Rather, it is useful to combine multiple methods from different domains for effective data exploration [Hinneburg et al., 1999; Won, 1999; Kreuseler and Schumann, 2002].

Traditionally the research in information visualisation does not involve focus on applying advanced machine learning algorithms to project high-dimensional data on to low-dimension (latent space) effectively. Recently, the core research in visual data mining has focused on combination of visual representation techniques and projection algorithms as well as on integrating the user in the exploration process. Integrating visual and nonvisual methods in order to support a variety of exploration tasks, such as identifying patterns in large unstructured heterogeneous information or identifying clusters or studying data in different clusters in detail etc., requires sophisticated machine learning algorithms, visual methods, and interaction techniques.

Ankerst [2001] classified visual data mining approaches into three categories. Approaches of the first type apply visual methods independently of data mining algorithms. The second type uses visual methods in order to represent patterns and results from mining algorithms graphically. The third type tightly integrates both mining algorithms and visual methods in such a way that intermediate steps of the mining algorithms can be visualised and further guided by the domain expert. This tight integration allows users to control and steer the mining process directly based on the visual feedback they receive. The approach we present here belongs to the third type; we introduce a flexible framework for visual data mining which combines principled projection algorithms developed in the machine learning domain and advanced visual representation techniques to provide feedback and involve the domain expert in the model development process.

A visual data exploration framework requires to be implemented in an easy-to-use interface. Shneiderman's mantra of "Overview first, zoom and filter, details-on-demand" [Shneiderman, 1996] nicely summarises the design philosophy of modern information visualisation systems for better usability. First, the user needs to get an overview of the data. In the second stage, the user identifies interesting patterns and focuses on one or more of them. Finally, to analyse patterns in detail, the user needs to drill down and access details of the data. Visual representation and interaction techniques may be used for all three steps of the data exploration process [Kreuseler and Schumann, 2002]. The interface we developed using our framework follows Shneiderman's mantra to provide an effective user interface. The advantage of such an interface is that the user is directly involved in the data mining process taking advantage of powerful and principled machine learning algorithms.

As reviewed in [Ferreira de Oliveira and Levkowitz, 2003], traditional projection methods such as principle component analysis (PCA) [Bishop, 1995], factor analysis (FA) [Harman, 1967], multi-dimensional scaling (MDS) [Young, 1987], Sammon's mapping [Sammon, 1969] and the self-organizing maps (SOM) [Kohonen, 1995] are already widely used in the knowledge discovery and data mining domain [Hoffman et al., 1997; Wang and Wang, 2002; Koua and Kraak, 2004; Huang et al., 2005]. For many real-life high-dimensional datasets, the generative topographic mapping (GTM) [Bishop et al., 1998], a principled projection algorithm, provides better (i.e. more informative) projections than those obtained from traditional methods, such as PCA, Sammon's mapping, and SOM [Maniyar et al., 2006]. Moreover, since the GTM provides a probabilistic representation of the projection manifold, it is possible to analytically define and calculate (local) geometric properties anywhere on the manifold. For example, we can calculate the local magnification factors [Bishop et al., 1997b], which describe how small regions in the visualisation space are stretched or compressed when mapped to the data space. Note that it is not possible to obtain magnification factors for Sammon's mapping. For PCA, the magnification factors are constant as it is a linear map. For the SOM, the magnification factors can only be approximated [Bishop et al., 1997c]. It is also possible in the GTM to calculate analytically the local directional curvatures of the projection manifold to provide the user with a facility for monitoring the amount of folding and neighbourhood preservation in the projection manifold [Tiño et al., 2001a]. The details of how these geometric properties of manifold can be used during visual data mining are presented in Section 3.4.

Moreover, it has been argued that a single two-dimensional projection, even if it is non-linear, is not usually sufficient to capture all of the interesting aspects of a large high-dimensional datasets. Hierarchical extensions of visualisation methods allow the user to "drill down" into the data; each plot covers a smaller region and it is therefore easier to discern the structure of the data. Hierarchical GTM (HGTM) is a hierarchical visualisation system which allows the user to explore interesting regions in more detail [Tiño et al., 2001b]. In Chapter 4, we also demonstrate how probabilistic hierarchical visualisation models can be used to develop effective local prediction models [Maniyar and Nabney, 2005].

The results presented, on a real-life dataset from the chemoinformatics domain, clearly show that the interface developed using the framework proposed in this chapter provides a useful platform for visual data mining of large high-dimensional datasets. Projection results of GTM are analytically compared with projection results from other methods traditionally used in the drug discovery domain.

The remainder of this chapter is organised as follows. The next section discusses related work in the visual data mining community. In Section 3.3, we provide an overview of some existing techniques for mapping a high-dimensional data set to a low dimensional embedding. The main information visualisation and interaction techniques we used are described in Section 3.4. The integrated visual data exploration framework we propose is discussed in Section 3.5 with a brief discussion of the software implementation we have developed. In Section 3.6 we present the evaluation methods used to assess the quality of the projection results. A detailed case study is presented in Section 3.7 on a real-life datasets from chemoinformatics. In Section 3.8, we discuss computational costs for the projection algorithms. Finally, we draw the main conclusions from this work in Section 3.9.

3.2 Related work

Popular data exploration software in pharmaceutical research, such as SpotFire, include traditional statistical projection algorithms (PCA, FA, MDS) and powerful visual techniques for further exploration, but not much work has been done to utilise advanced projection algorithms from machine learning community to explore large high-dimensional chemoinformatics datasets. Though the projection methods implemented in SpotFire have been of some help, they fail to provide effective grouping (clustering) while projecting high-dimensional complex datasets onto lower dimensions.

Research in the visual data mining domain has attempted to bring machine learning and visual techniques together to some extend but mainly the focus here has been the visual techniques. A typical use of visualisation in mining consists of visually conveying the results of a mining task, such as clustering or classification, to enhance user interpretation. Following are a few examples of such systems.

Hoffman et al. [1997] provided a case study describing how high-dimensional visual data exploration techniques such as RadViz [Ankerst et al., 1996], parallel coordinates [Inselberg and Dimsdale, 1990] and Sammon's mapping have been used in combination with rule-based classifiers and neural networks to classify DNA sequences. It proved to be a good way of providing visual interaction but Sammon's mapping is not always effective for large high-dimensional datasets. Another example is given by the BLOB and H-BLOB clustering algorithms [Gross et al., 1995; Sprenger et al., 2002], which use implicit surfaces for visualising data clusters. The SOM has been applied in data mining and multidimensional exploratory data analysis in several domains [Vesanto, 1999, 2000]. Hinneburg et al. [1999] illustrates tight coupling of visualisation resources into a mining technique. Other recent work of importance which has focussed on forming the proximity matrix includes Isomap [Tenenbaum et al., 2000], where an approximation to geodesic distance is used to obtain spectral clustering. The proximity data is derived from a graph which is generated over all data points by connecting points i and j if they are closer than ϵ (ϵ-isomap).

As discussed above, there has been some efforts to utilise recent development in unsupervised projection algorithms in the visual data exploration frameworks, but traditionally the visual data mining domain has been dominated by information visualisation researchers. Scientists in information visualisation domain have developed powerful techniques to visualise and provide effective interactions, but advances in machine learning have not been fully exploited. In the last 10 years, the machine learning and statistics communities have developed a range of principled projection algorithms which can provide effective projection for high-dimensional complex datasets found in chemoinformatics domain. One of our main aims in this research is to develop an integrated visual data exploration tool which integrates the most effective projection techniques from machine learning with powerful visual aids from information visualisation to provide effective solutions for challenges in chemoinformatics.

In the following section we briefly introduce important projection algorithms.

3.3 Projection algorithms

The problem of finding a low-dimensional representation of high-dimensional data is not new and a considerable number of models have been suggested in the statistics and machine learning literature. This section provides a brief introduction to some of the important projection algorithms, they are broadly categorised into

- **projection models:** They aim at finding low-dimensional manifolds in the space of the data, such that the distance between data and its projection on the manifold is small. Principal component analysis (PCA) is reviewed in Section 3.3.1 as an example of this category.

- **generative models:** They try to model the distribution of the data by defining a density model with low intrinsic dimensionality in the data space. In this category, we discuss factor analysis (FA), probabilistic PCA (PPCA), generative topographic mapping (GTM) and hierarchical GTM (HGTM) in Sections 3.3.2 to 3.3.4.
- **other models:** Models not belonging to any of the previous two categories are classified as other models. In this category, we discuss self-organizing maps (SOM) in Section 3.3.5 and approaches based on multidimensional scaling in Section 3.3.6.

Finally we end this section with a discussion in section 3.3.7, highlighting important issues and giving a comparison matrix.

3.3.1 Principal component analysis (PCA)

Principal component analysis (PCA) [Jolliffe, 2002] is a multivariate procedure which rotates the data such that maximum variabilities are projected onto the axes. Essentially, a set of correlated variables are transformed into a set of uncorrelated variables which can be ordered by reducing variability. The uncorrelated variables are linear combinations of the original variables, and in many cases the last few of these variables can be removed with minimum loss of information.

The first principal component is the combination of variables that explains the greatest amount of variation. Subject to being orthogonal to the first principal component, the second principal component defines the next largest amount of variation to the first principal component. If the data covariance matrix has full rank, there can be as many principal components as there are variables.

Given a set of observations $\mathbf{t}_n \in \mathbb{R}^D, n = 1, \ldots, N$, which are centred, $\sum_{n=1}^{N} \mathbf{t}_n = 0$, in PCA we find the principal components by diagonalising the covariance matrix,

$$C = \frac{1}{N} \sum_{n=1}^{N} \mathbf{t}_n {\mathbf{t}_n}^{\top}, \tag{3.1}$$

and then finding its eigen-structure

$$\mathbf{CU} = \mathbf{U\Lambda}. \tag{3.2}$$

U is a $D \times D$ matrix which has the unit length eigenvectors, $\mathbf{u}_1, \ldots, \mathbf{u}_D$, as its columns and Λ is diagonal matrix with the corresponding eigenvalues, $\lambda_1, \ldots, \lambda_D$, along the diagonal. The eigenvectors are the principal components and the eigenvalues are the corresponding variances.

It is difficult to control and navigate more than two-dimensional representation of data on a computer screen which is a two-dimensional device. Thus our aim is to project higher-dimensional datasets onto two dimensions and use three or more dimensional representation only in rare cases. Since we want to project multi-dimensional data onto two dimensions, we use the first two principal components for projection. PCA is the most commonly used of the projection models in current practice. The fact that PCA defines a linear, orthogonal model space gives it favourable computational properties, but it is also its main limitation since any non-linear correlation between variables will not be captured.

3.3.2 Factor analysis (FA)

Traditionally, factor analysis (FA) [Bartholomew, 1984; Gorsuch, 1996] has been the 'generative cousin' of PCA; in fact, the two techniques are sometimes confused. The key difference is that where PCA is focusing on variance, FA focus on covariance. Covariance between a set of observed variables is seen as an indication that these variables are, if only to a certain extent, functions of a common latent factor. The effect of this difference becomes apparent when the observed variables are subject to significantly different noise levels. While PCA will try to capture all variance in the data, including variance due to noise affecting only individual variables, FA will focus on the covariance, regarding additional variability in the observed variables as noise.

Factor analysis represents an observed D-dimensional continuous variable, $\mathbf{t}$, as a linear function of an L-dimensional continuous latent variable, $\mathbf{x}$, and an independent Gaussian noise process, $\boldsymbol{\epsilon}$,

$$\mathbf{t} = \mathbf{W}\mathbf{x} + \boldsymbol{\mu} + \boldsymbol{\epsilon}. \tag{3.3}$$

Here $\mathbf{W}$ is a $D \times L$ matrix defining the linear function relating the two sets of variables, while the parameter vector $\boldsymbol{\mu}$ permits the model to have non-zero mean. The motivation is that, with $D < L$, the latent variables will offer a more parsimonious explanation of the dependencies between the observations. Conventionally, $\mathbf{x} \sim N(\mathbf{0}, \mathbf{I})$, and the latent variables are defined to be independent and Gaussian with unit variance. By additionally specifying the error, or noise, model to be likewise Gaussian $\boldsymbol{\epsilon} \sim N(0, \psi)$, equation (3.3) induces a corresponding Gaussian distribution for the observations $\mathbf{t} \sim N(\boldsymbol{\mu}, \mathbf{W}\mathbf{W}^{\mathsf{T}} + \psi)$. The model parameters can then be determined by maximum-likelihood, although because there is no closed-form analytic solution for $\mathbf{W}$ and ψ, their values must be obtained via an iterative procedure.

The motivation, and indeed key assumption, for the factor analysis model is that, by constraining the error covariance ψ to be a diagonal matrix whose elements ψ_i are usually estimated from the data, the observed variables t_i are conditionally independent given the values of the latent variables $\mathbf{x}$. These latent variables are thus intended to explain the correlations between observation variables while ϵ_i represents variability unique to a particular t_i. This is where factor analysis fundamentally differs from standard PCA (Section 3.3.1), which effectively treats covariance and variance identically.

3.3.3 Probabilistic PCA (PPCA)

Tipping and Bishop [1999b] proposed a new probabilistic formulation of PCA, probabilistic PCA (PPCA), which derives PCA as a latent variable model, and can be regarded as a special case of factor analysis. In PPCA, given a set centred of D-dimensional data $\{\mathbf{t}_n\}_{n=1}$ and denoting the latent variable associated with each data point, $\mathbf{x}_n$, we write the likelihood for an individual data point as

$$p(\mathbf{t}_n|\mathbf{W},\beta) = \int p(\mathbf{t}_n|\mathbf{x}_n,\mathbf{W},\beta)p(\mathbf{x}_n)\,\mathrm{d}\mathbf{x}_\mathrm{n}, \tag{3.4}$$

where $p(\mathbf{t}_n|\mathbf{x}_n,\mathbf{W},\beta) = N(\mathbf{t}_n|\mathbf{W}\mathbf{x}_n,\beta^{-1}\mathbf{I})$ and $p(\mathbf{x}_n)$ is Gaussian distributed with unit covariance, $p(\mathbf{x}_n) = N(\mathbf{x}|0,\mathbf{I})$. The solution for $\mathbf{W}$ and β can then be found by assuming that $\mathbf{t}_n$ are i.i.d. and maximising the likelihood of the dataset,

$$p(\mathbf{T}|\mathbf{W},\beta) = \prod_{n=1}^{N} p(\mathbf{t}_n|\mathbf{W},\beta), \tag{3.5}$$

where $\mathbf{T} = [\mathbf{t}_1 \ldots \mathbf{t}_N]^\top$ is the $N \times D$ design matrix.

The PPCA approach brings PCA into the family of generative models, which in turn opens up a whole range of possibilities. In particular, Tipping and Bishop [1999a] show how to construct mixtures of probabilistic principal component analysers, which are fitted to data using a simple extension of the Expectation-Maximization (EM) algorithm [Dempster et al., 1977] for basic probabilistic PCA. This is achieved by introducing a mean $\boldsymbol{\mu}^i$ for each model i and re-estimating $p(\mathbf{t}_n|i)$ and the prior probabilities for each model, $p(i)$, in each step of the EM algorithm.

Mixture of PPCA [Tipping and Bishop, 1999a] models is an extension to the PPCA, which can determine the principal sub-space of the data through maximum-likelihood estimation of the data through maximum-likelihood estimation of the parameters in a Gaussian latent variable model. It can be extended to a hierarchical representation as in [Bishop and Tipping, 1998]. Later on, Tiño and Nabney [2002] formulated a generic version of this approach which is briefly discussed in the next section.

3.3.4 Generative topographic mapping (GTM)

The GTM models a probability distribution in the (observable) high-dimensional data space, $\mathcal{D} = \mathbb{R}^D$, by means of low-dimensional latent, or hidden, variables [Bishop et al., 1998]. The data is visualised in the latent space, $\mathcal{H} \subset \mathbb{R}^L$.

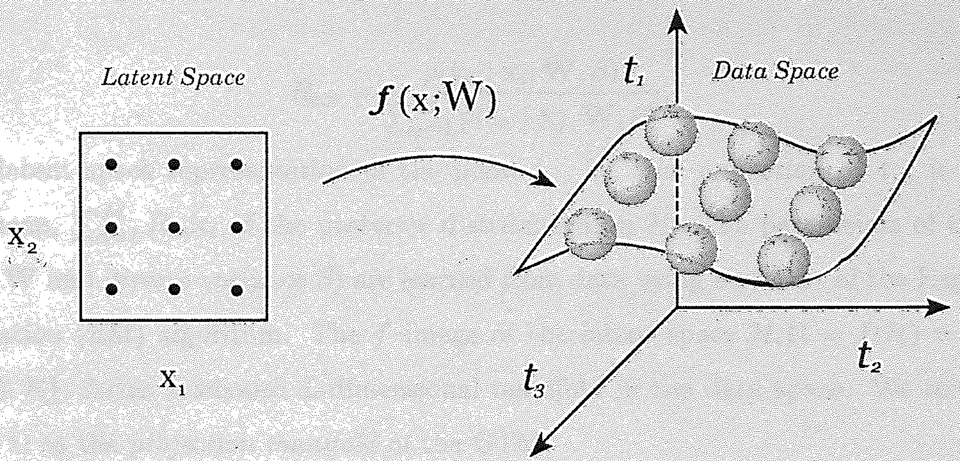


Figure 3.1: Schematic representation of the GTM model.

As demonstrated in Figure 3.1 (adapted from [Bishop et al., 1998]), we cover the latent space, $\mathcal{H}$, with an array of M latent space centres, $\mathbf{x}_i \in \mathcal{H}, i = 1, 2, ..., M$. The non-linear GTM transformation, $f : \mathcal{H} \Rightarrow \mathcal{D}$, from the latent space to the data space is defined using a radial basis function (RBF) network. To this end, we cover the latent space with a set of K fixed non-linear basis functions (we use Gaussian functions of the same width σ), $\phi : \mathcal{H} \Rightarrow \mathbb{R}, j = 1, 2, ..., K$, centred on a regular grid in the latent space. Given a point $\mathbf{x} \in \mathcal{H}$ in the latent space, its image under the map f is

$$f(\mathbf{x}_i, \mathbf{W}) = \boldsymbol{\phi}(\mathbf{x}_i)\mathbf{W}, \tag{3.6}$$

where $\boldsymbol{\phi}(\mathbf{x}_i) = (\phi_1(\mathbf{x}_i), ..., \phi_K(\mathbf{x}_i))^T$ and $\mathbf{W}$ is a $K \times D$ matrix of weight parameters.

GTM creates a generative probabilistic model in the data space by placing a radially symmetric Gaussian with zero mean and inverse variance β around images, under f, of the latent space centres $\mathbf{x}_i \in \mathcal{H}, i = 1, 2, ..., M$. We refer to the Gaussian density associated with the centre $\mathbf{x}_i$ by $p(\mathbf{t} \mid \mathbf{x}_i, \mathbf{W}, \beta)$. Defining a uniform prior over $\mathbf{x}_i$, the density model in the data space provided by the GTM is

$$p(\mathbf{t} \mid \mathbf{W}, \beta) = \frac{1}{M} \sum_{i=1}^{M} p(\mathbf{t} \mid \mathbf{x}_i, \mathbf{W}, \beta). \tag{3.7}$$

For the purpose of data visualisation, we use Bayes' theorem to invert the transformation f from the latent space $\mathcal{H}$ to the data space $\mathcal{D}$. The posterior distribution on $\mathcal{H}$, given a data point $\mathbf{t}_n \in \mathcal{D}$, is a sum of delta functions centred at centres $\mathbf{x}_i$, with coefficients equal to the posterior probability $R_{i,n}$ that the i-th Gaussian (corresponding to the latent space center $\mathbf{x}_i$) generated $\mathbf{t}_n$ [Bishop et al., 1998],

$$R_{i,n} = \frac{p(\mathbf{t}_n \mid \mathbf{x}_i, \mathbf{W}, \beta)}{\sum_{j=1}^{M} p(\mathbf{t}_n \mid \mathbf{x}_j, \mathbf{W}, \beta)}. \tag{3.8}$$

The latent space representation of the point $\mathbf{t}_n$, i.e. *the projection of* $\mathbf{t}_n$, is taken to be the mean, $\sum_{i=1}^{M} R_{in}\mathbf{x}_i$ of the posterior distribution on $\mathcal{H}$. The parameters of the GTM (weights $\mathbf{W}$ and inverse variance β) are learned from data using a variant of the Expectation Maximisation (EM) algorithm. The f–image of the latent space $\mathcal{H}, \Omega = f(\mathcal{H}) = \{f(\mathbf{x}) \in \mathbb{R}^D \mid \mathbf{x} \in \mathcal{H}\}$, forms a smooth L-dimensional manifold in the data space. We refer to the manifold Ω as the *projection manifold* of the GTM.

Recently we have extended the GTM algorithm to estimate feature relevance simultaneously with the training of the visualisation model [Maniyar and Nabney, 2006a]. This extension is presented in Chapter 5.

The hierarchical PPCA model has been extended to arbitrary generative models trained using a variant of EM algorithm by Tiňo and Nabney [2002]. The application of this general result for GTM is described next.

Hierarchical GTM (HGTM)

The hierarchical GTM (HGTM) arranges a set of GTMs and their corresponding plots in a tree structure $\mathcal{T}$ [Tiňo and Nabney, 2002]. An example HGTM structure is shown in the Figure 3.2.

The *Root* of the hierarchy is at level 1, i.e. *Level(Root)* = 1. Children of a model $\mathcal{N}$ with *Level*($\mathcal{N}$) = ℓ are at level ℓ+1, i.e. *Level*($\mathcal{M}$) = ℓ+1, for all $\mathcal{M} \in$ *Children*($\mathcal{N}$). Each model $\mathcal{M}$ in the hierarchy, except for *Root*, has an associated parent-conditional mixture coefficient, or prior $\pi(\mathcal{M} \mid \textit{Parent}(\mathcal{M}))$. The priors are non-negative and satisfy the consistency condition: $\sum_{\mathcal{M} \in \textit{Children}(\mathcal{N})} \pi(\mathcal{M} \mid \mathcal{N}) = 1$. Unconditional priors for the models are recursively calculated as follows: $\pi(\textit{Root}) = 1$, and for all other models

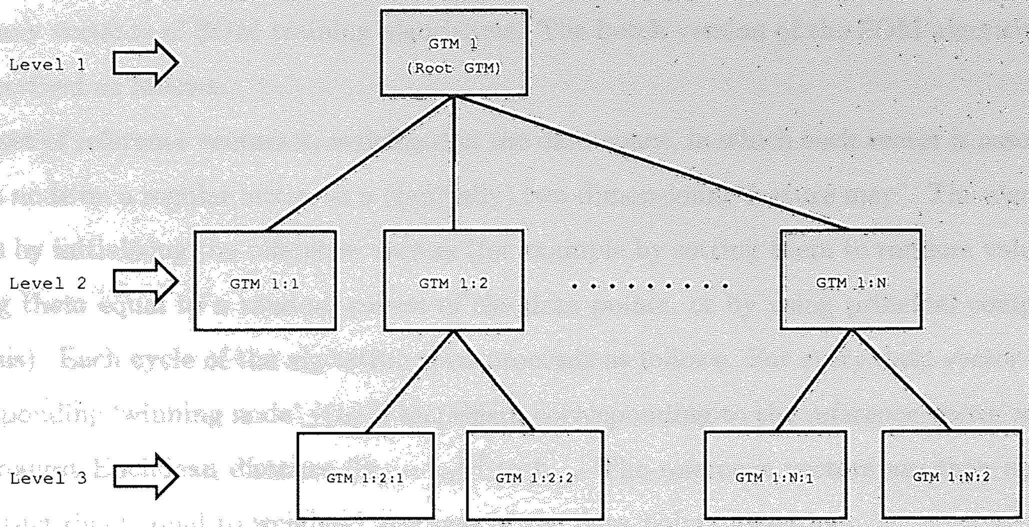


Figure 3.2: An example structure for the HGTM model.

$$\pi(\mathcal{M}) = \prod_{i=2}^{Level(\mathcal{M})} \pi(Path(\mathcal{M})_i \mid Path(\mathcal{M})_{i-1}), \qquad (3.9)$$

where $Path(\mathcal{M}) = (Root, ..., \mathcal{M})$ is the N-tuple ($N = Level(\mathcal{M})$) of nodes defining the path in $\mathcal{T}$ from $Root$ to $\mathcal{M}$.

The distribution given by the hierarchical model is a mixture of leaf models of $\mathcal{T}$,

$$p(\mathbf{t} \mid \mathcal{T}) = \sum_{\mathcal{M} \in Leaves(\mathcal{T})} \pi(\mathcal{M}) p(\mathbf{t} \mid \mathcal{M}). \qquad (3.10)$$

Thus we obtain a *soft* segmentation of the input space from the leaf models HGTM.

Non-leaf models not only play a role in the process of creating the hierarchical model, but in the context of data visualisation can be useful for determining the relationship between related subplots in the hierarchy.

The HGTM is trained using a variant of the EM algorithm to maximise its likelihood with respect to the data sample $\varsigma = \{\mathbf{t}_1, \mathbf{t}_2, ..., \mathbf{t}_N\}$. Training of a hierarchy of GTMs proceeds in a recursive fashion. Visualisation and interaction is provided to the domain experts during the intermediate steps of training an HGTM model. A detailed description of the interaction we provide in the software tool we developed is given in section 3.5.

3.3.5 Self-organizing maps (SOM)

The SOM is a popular unsupervised learning algorithm, based on a grid of artificial neurons whose weights are adapted to match input vectors in a training set [Kohonen, 1995]. There are many variants of SOM training algorithms. The batch version of the SOM algorithm can be described as follows.

A set of reference vectors $\mathbf{x}_i$ is defined in the data space, in which each vector is associated with a node on a regular lattice in a (typically) two-dimensional 'feature map'. The algorithm begins by initialising the reference vectors (for example by setting them to random values, by setting them equal to a random subset of the data points, or by using principal component analysis). Each cycle of the algorithm then proceeds as follows. For every data vector $\mathbf{t}_n$ the corresponding 'winning node' $j(n)$ is identified, corresponding to the reference vector $\mathbf{x}_j$ with the smallest Euclidean distance $||\mathbf{x}_j - \mathbf{t}_n||^2$ to $\mathbf{t}_n$. The reference vectors are then updated by setting them equal to weighted averages of the data points given by

$$\mathbf{x}_i = \frac{\sum_n h_{ij(n)} \mathbf{t}_n}{\sum_n h_{ij(n)}}. \tag{3.11}$$

where h_{ij} is a neighbourhood function associated with the ith node. This is generally chosen to be a uni-modal function of the feature map coordinates centred on the winning node, for example a Gaussian. The steps of identifying the winning nodes and updating the reference vectors are repeated iteratively. A key ingredient in the algorithm is that the width of the neighbourhood function h_{ij} starts with a relatively large value and is gradually reduced after each iteration.

3.3.6 Multidimensional scaling (MDS)

We have already mentioned several projection techniques which rely on learning a mapping from a latent space (the embedded space) to the data space. In this section we will briefly review methods that use proximity data to obtain a projection in the opposite direction. Broadly speaking, these methods are all variants or enhancements of the technique known as multidimensional scaling (MDS) [Cox and Cox, 2001]. Given an $N \times N$ matrix of 'distances', $\mathbf{D}$, between N points, MDS gives a corresponding set of N points, $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ in an L-dimensional space, such that the distances between points in $\mathbf{X}$ reflect those given in $\mathbf{D}$. The 'distances' need not be Euclidean distances, but can be more general, e.g. distance measures for categorical variables or subjective measures of similarity, in which case they are often called dissimilarities. These dissimilarities are the only information about the data that is required, so indeed the data does not even need to have an explicit form. However, in the context that we are interested in, where the data has an explicit representation as a set of points in $\mathcal{R}^D$, for which the Euclidean distance is the obvious dissimilarity measure, it can be shown that MDS corresponds to PCA. More precisely, the set of points found by MDS, $\mathbf{X}$, correspond(up to scaling and rotation) to the projection of the data on its first L principal components. In this form, MDS is known as principal coordinate analysis. Following are the popular variants of MDS model.

Sammon's mapping

Sammon's mapping [Sammon, 1969] represents a particular form of MDS [Ripley, 1996] – the basic idea is the same, but Sammon's mapping pays more attention to smaller distances, thereby achieving a varying resolution in the new representation of the data. Regions with a dense population of data points, between which distances are large, will be 'magnified' in the new representation because of relative distance differences. To formalise, given a set of 'distances' between N data points[1], the Sammon's mapping tries to find the set of points $\{\mathbf{x}_n\}$, $n = 1, 2, \ldots, N$, in $\mathbb{R}^L$ that minimises the stress measure, E. The stress measure emphasises large distances (because of squared term) as,

$$E = \sum_{i}^{N} \sum_{j>i}^{N} (d^*_{ij} - d_{ij})^2 \tag{3.12}$$

$$d^*_{ij} = \parallel \mathbf{x}_i - \mathbf{x}_j \parallel, \tag{3.13}$$

$$d_{ij} = \parallel \mathbf{t}_i - \mathbf{t}_j \parallel . \tag{3.14}$$

Neuroscale

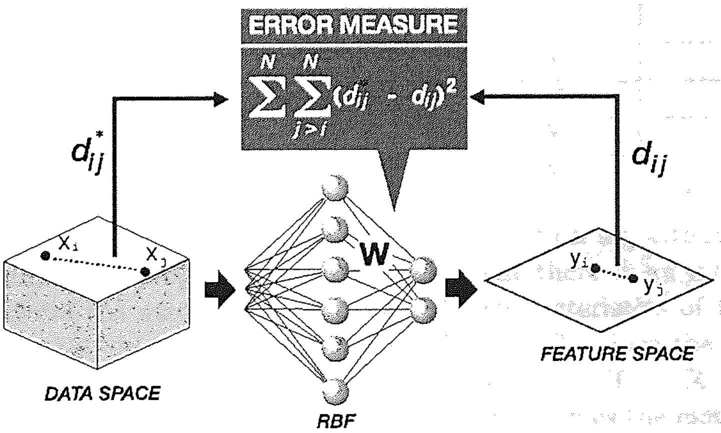


Figure 3.3: Schematic representation of the Neuroscale model.

[1] We assume that these distances are symmetric and that the distance from a point to itself is zero.

Neuroscale [Lowe and Tipping, 1997] is a novel neural network implementation of Sammon's mapping. It is a projection method which attempts to preserve relative 'similarity' (in this case characterised by *Euclidean* distance) between points in the data space ($\mathbf{t}$) and latent space ($\mathbf{x}$). As shown in Figure 3.3 (adapted from [Tipping and Lowe, 1998]).

The topographic nature of the transformation is imposed by the "stress" term, similar to (3.12), which accounts for the preservation of inter-point similarity.

Points are projected from the data space onto the latent space by means of a radial basis function (RBF) neural network which adapts to the required projection mapping. The RBF parameters are defined by minimising the "stress" term, rather than the traditional residual error minimisation. Now the targets are projected using the RBF as

$$\mathbf{t} = \mathbf{W}\phi(\mathbf{x}). \tag{3.15}$$

where $\mathbf{W}$ is a weight matrix and $\phi(\cdot)$ are the basis functions.

3.3.7 Discussion

We have reviewed a number of algorithms intended to capture the low-dimensional structure of data in high-dimensional spaces, or at least provide a low-dimensional representation of this data. FA and PCA, are both linear models, but FA allows for a richer noise model than PCA. All other algorithms mentioned in this section have a non-linear interpretation. In Table 3.1 we have summarised some of the properties of these algorithms/models.

	Proximity	$\mathbf{X} \rightarrow \mathbf{T}$	$\mathbf{T} \rightarrow \mathbf{X}$	Non-linear	Probabilistic	Convex
PCA	I	Y	Y		I	Y
FA		Y	Y		Y	Y
PPCA		Y			Y	
GTM		Y		Y	Y	
SOM	Y			Y		
Sammon's mapping	Y			Y		
Neuroscale	Y		Y	Y		

Table 3.1: Overview of the relationship between the projection algorithms. A 'Y' indicates the algorithm exhibits that property, an 'I' indicates that there is an interpretation of the algorithm that exhibits the associated property. The characteristics of the algorithm are: proximity: is the method based on proximity data? $\mathbf{X} \rightarrow \mathbf{T}$: does the method lead to a mapping from the embedded (latent-space) to the data-space? $\mathbf{T} \rightarrow \mathbf{X}$: does the method lead to a mapping from data to embedded space? Non-linear: does the method allow for non-linear embedding? Probabilistic: is the method probabilistic? Convex: algorithms that are considered convex have a unique solution, for the others local optima can occur (presenting the relationships in this way was inspired by a similar table in [Lawrence, 2005]).

Although the SOM has been the subject of a considerable amount of research and has been applied to a wide range of tasks (including in drug discovery domain), the major problem with SOM is that it does not define a density model in the data space. Because of this there are other issues like, the SOM training algorithm does not optimise an objective function, there is no general guarantee the training algorithm will converge, there is no theoretical framework based on which appropriate values for the model parameters can be chosen, and it is not obvious how SOM models should be compared to other SOM models. This has inspired the search for re-formulations of the SOM within the framework of probability theory and statistics. In fact, GTM was proposed as a principled alternative to the SOM [Bishop et al., 1997a].

A striking fact is that two of the 'non-generative' models that we considered - PCA and the SOM - have been re-interpreted or reformulated for the purpose of bringing them into the family of generative models. The attraction of this type of model stems from the fact it fits into the much wider framework of probability theory and statistics. They can therefore directly make use of well-founded theory for fitting models to data, combining models, treatment of incomplete data, and other extensions such as developing guided local models (discussed in Chapter 4) and estimating feature relevance (discussed in Chapter 5).

The GTM outlined above is typically designed to embed a data set in two dimensions; it relies on either randomly sampled latent space centres, or a grid of points in the latent space to achieve this embedding, this causes problems when the dimensionality of the latent space increases. For the GTM, the data distribution is not that important as (with enough kernels) it can model an arbitrary distribution.

Point representations of the latent space are useful because they allow for non-linear models: each point is easy to propagate through the non-linear mapping to the data space. These non-linear mappings are designed to address the weaknesses in visualising data sets that arise when using standard statistical tools that rely on linear mappings, such as principal component analysis (PCA) and factor analysis (FA): with a linear mapping it may not be possible to reflect the structure of the data through a low dimensional embedding.

Sammon's mapping suffer from a weakness in that the projection of data points which were not in the original data set can be computationally demanding, i.e. despite their name they do not provide an explicit mapping between the data and latent-space. The lack of a mapping was addressed by the Neuroscale algorithm, so in this thesis we use Neuroscale for benchmarking this category of projection methods.

3.4 Information visualisation techniques

There is a large number of visual techniques developed in the information visualisation domain which can be used for visualising data or results. In addition to standard 2D/3D graphing, there are a number of more sophisticated visualisation techniques. Keim [2002] provided an informative overview of different classes of information visualisation techniques. Other than the usual facilities such as zoom, rotate, etc., we provide following information visualisation aids to support the projection obtained from the principled machine learning algorithms to create a powerful visual data exploration framework.

3.4.1 Magnification factors (MF)

It is useful to understand the geometry of the projection manifold. One of the main advantages of using GTM–based models is that it is possible to analytically calculate the magnification factors (MF) [Bishop et al., 1997b] and the directional curvature (DC) [Tiňo et al., 2001a] of the GTM projection manifold. MFs of a GTM projection manifold, Ω, are calculated as the determinant of the Jacobian of the GTM map f [Bishop et al., 1997b]. Magnification factors plots are used to observe the amount of stretching in a GTM manifold at different parts of the latent space, which helps in understanding the data space, outlier detection, and cluster separation.

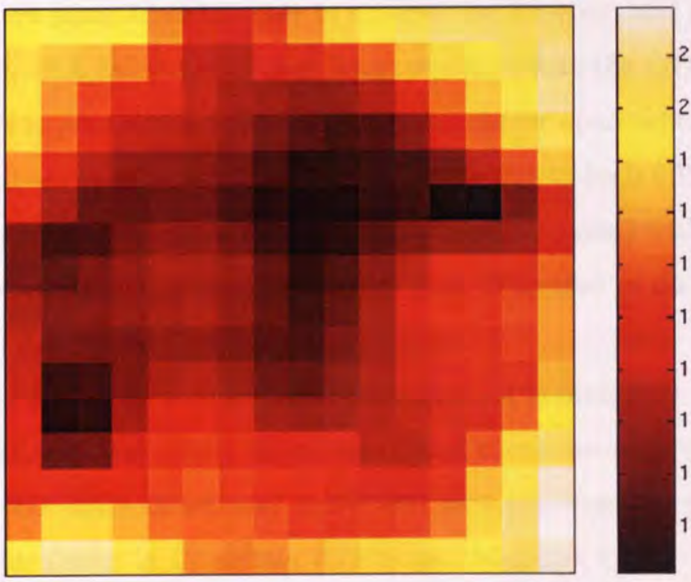


Figure 3.4: An example magnification factors plot on a $\log_{10}$ scale.

The MF are represented by colour shading in the projection manifold (e.g., see Figure 3.4). The lighter the colour, the more stretch in the projection manifold.

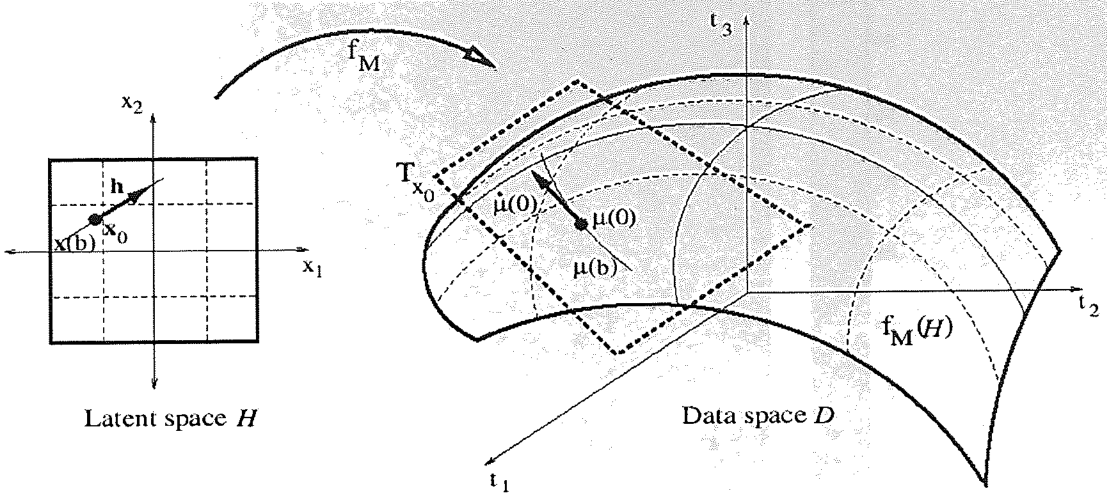


Figure 3.5: An explanation of local directional derivative of the visualisation manifold. A straight line $\mathbf{x}(b)$ passing through the point $\mathbf{x}_0$ in the latent space $\mathcal{H}$ is mapped via f to the curve $\mu(b) = f(\mathbf{x}(b))$ in the data space $\mathcal{D}$. Curvature of μ at $f(\mathbf{x}_0) = \mu(0)$ is related to the directional curvature of the projection manifold $f(\mathcal{H})$ with respect to the direction $\mathbf{h}$. The tangent vector $\dot{\mu}(0)$ to μ at $\mu(0)$ lies in $\mathbf{T}_{\mathbf{x}_0}$ (dashed rectangle), the tangent plane of the manifold $f(\mathcal{H})$ at $\mu(0)$ (adapted from [Tiňo et al., 2001a]).

3.4.2 Directional curvatures (DC)

Tiňo et al. [2001a] derived a closed-form formula for directional curvatures of the GTM projection manifold, Ω, for a latent space point $\mathbf{x} \in \mathcal{H}$ and a directional vector $\mathbf{h} \in \mathcal{H}$. Directional curvature plots allow the user to observe the direction and amount of folding in the GTM manifold. This can help the user detect regions where the GTM manifold does not fit the data well. It is possible that groups of data points far apart when projected onto the projection manifold are close together in the data space due to high folding in the manifold. This neighbourhood preservation in the data space can be spotted with a strong curvature band on the corresponding directional curvature plot. The idea of directional curvature is explained in figure 3.5.

The direction of folding in the projection manifold plot is presented using a small line for each part of the projection manifold in the directional curvature plots (e.g., see Figure 3.6). In this example, and other directional curvatures plot presented in this thesis, direction curvatures were calculated in 16 certain directions. Maximal curvature was plotted as a small line for each region. The length and the shade of the background colour represents the magnitude of folding. The longer the line and the lighter the background colour, higher the folding (curvature).

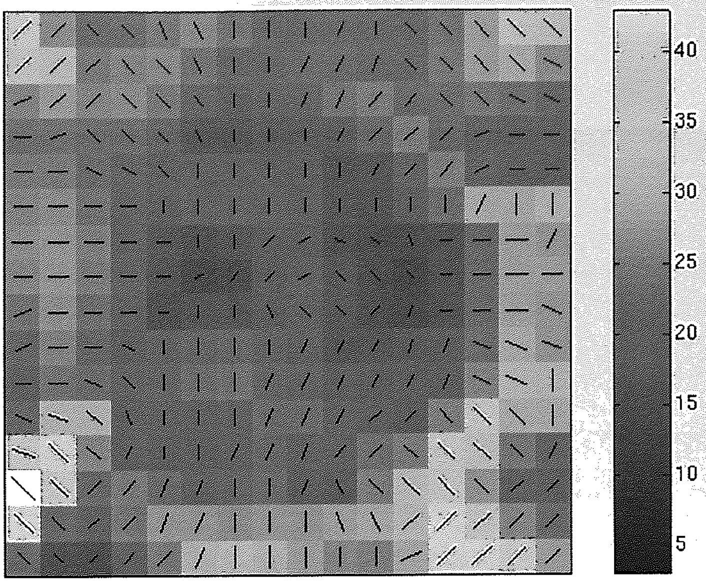
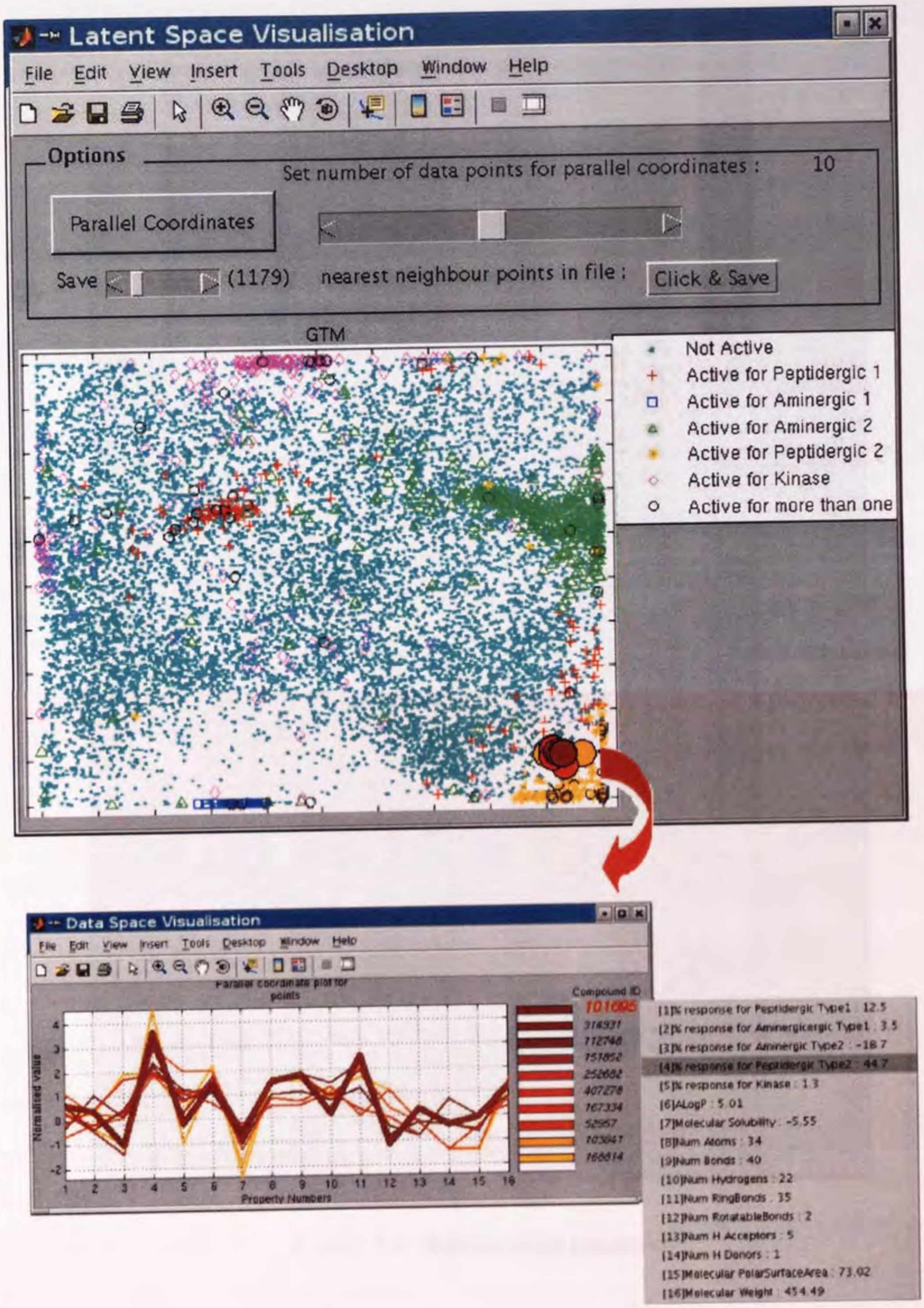


Figure 3.6: An example directional curvatures plot.

3.4.3 Local parallel coordinates

The parallel coordinates technique [Inselberg and Dimsdale, 1990] maps a D-dimensional data space onto two display dimensions by using D equidistant axes which are parallel to one of the display axes. It displays each multi-dimensional data point as a polygonal line which intersects the horizontal dimension axes at the position corresponding to the data value for the corresponding dimension.

Instead of displaying parallel coordinates for all the data points together, which is impractical for a large dataset, we provide an interactive facility to let the user select a point on the projection manifold and display parallel coordinates for a few nearest neighbours of that selected point. Figure 3.7 displays an example of parallel coordinates used on a GTM projection: when the user clicks on a point in the projection (upper plot), the data space visualisation graph shows a colour coded plot of normalised property values for a group of points close in the projection space. We call this a *local* parallel coordinates technique. This facility has proved very useful for the domain experts at Pfizer to understand large high-dimensional datasets. Using this facility, the user can study properties of a point in the high-dimensional data space while working with the lower-dimensional latent (projection) space. A detailed example discussing how local parallel coordinates are used to explore a projection manifold is presented in Section 3.7.4.

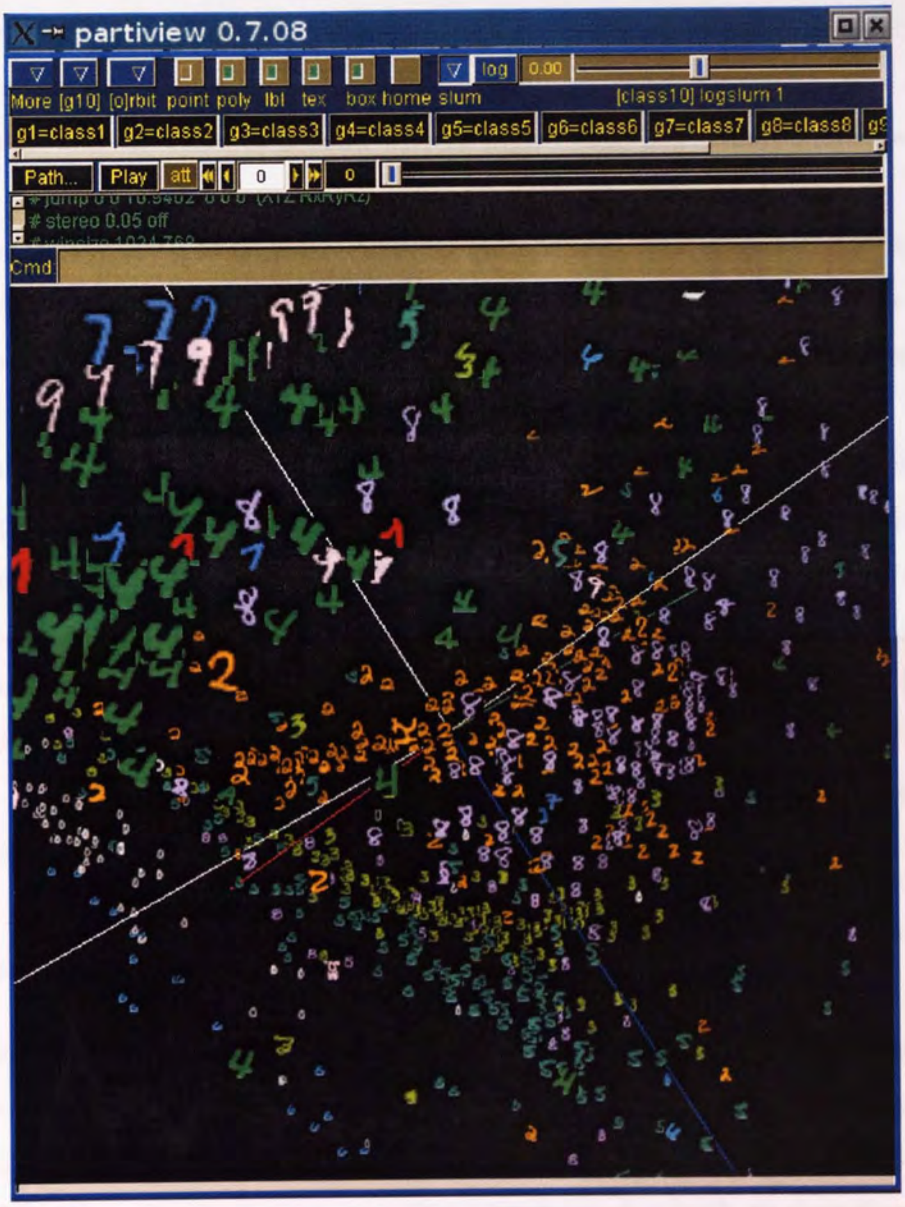


Set number of data points for parallel coordinates : 10

Save (1179) nearest neighbour points in file :

[1]% response for Peptidergic Type1 : 12.5
[2]% response for Aminergicergic Type1 : 3.5
[3]% response for Aminergic Type2 : -18.7
[4]% response for Peptidergic Type2 : 44.7
[5]% response for Kinase : 1.3
[6]ALogP : 5.01
[7]Molecular Solubility : -5.55
[8]Num Atoms : 34
[9]Num Bonds : 40
[10]Num Hydrogens : 22
[11]Num RingBonds : 35
[12]Num RotatableBonds : 2
[13]Num H Acceptors : 5
[14]Num H Donors : 1
[15]Molecular PolarSurfaceArea : 73.02
[16]Molecular Weight : 454.49

Figure 3.7: The projection and the local parallel coordinates.



```
# jump 0 0 16.9402 0 0 0 (XYZ RxRyRz)
# stereo 0.05 off
# winsize 1024 768
```

Figure 3.8: Billboarding example.

3.4.4 Billboarding

For many real-life datasets which have a natural representation, e.g. chemical compound structure, handwritten digit recognition, face recognition, galaxy classification, etc., using this natural representation of data points in the projection is more helpful for understanding the data compared with data represented by labelled and/or coloured dots.

Here the term 'billboarding' means visualising a natural representation of a data point in the form of an image, in such a way that the image always faces the viewer (even in 3D). A chemical compound structure or a hand written digit image is certainly more user-friendly than a dot.

Partiview [Levy, 2001] is an interactive 3D visualisation tool supporting a billboarding facility, primarily created for astronomy-related applications. But recently it has been successfully used for visualising the output of some machine learning algorithms [Surendran and Levy, 2004].

The number of pictures that can be displayed at a time depends on how much graphics memory is present. Figure 3.8 presents a close up of the points visualised for the MNIST database [LeCun et al., 1998] using Laplacian eigenmaps [Belkin and Niyogi, 2003]. Billboarding presentation of images of the handwritten digits provides us an intuitive visualisation and can help us to identify why certain data points are misclassified (e.g., notice that in Figure 3.8, images of 7s and 9s on the top left corner of the plot are quite similar). Partiview also provides many useful interaction facilities, such as 3D zooming and traversal, selective plotting of classes, properties displace, etc. [Levy, 2001].

3.5 The integrated visual data exploration framework

The integrated visual data exploration framework combines principled projection algorithms, discussed in Section 3.3, and visual techniques, discussed in Section 3.4, to achieve a better understanding of a high-dimensional data space. It follows Shneiderman's mantra [Shneiderman, 1996], "Overview first, zoom and filter, details on demand", to provide an effective interface.

To support the 'overview first' stage of Shneiderman's mantra, output of the projection algorithms and basic visualisation aids such as coloured labelling, rotate, etc., are provided. For the second stage, 'zoom and filter', visualisation aids such as zooming, filtering interesting regions on the projection manifold with the use of magnification factor and directional curvatures plots, etc., are provided. This allows the user to identify and concentrate on interesting subsets of the projection we obtained in the first stage. The third stage, 'details-on-demand', is supported using local parallel coordinates and billboarding. Integration with other visualisation tools is also possible at various stages.

Moreover, since a single two-dimensional projection, even if it is non-linear, is not usually sufficient to capture all of the interesting aspects of a large high-dimensional data sets, a hierarchical system which allows the user to interactively drill down in the projection can be useful.

Interactive visual methods support the construction of hierarchical models, such as HGTM, and allow the user to informatively explore interesting regions in more detail. Visual aids described in Section 3.4 are provided at each stage of the HGTM model development. First, a base (*Root*) GTM is trained and used to visualise the data. Then the user identifies interesting regions on the visualisation plot that they would like to explore in greater detail. In particular, the user chooses a collection of points, $c_i \in \mathcal{H}$, by clicking on the projection plot. The "regions of interest" given by these points (centres) are then transformed into the data space as Voronoi compartments [Aurenhammer, 1991] defined by the mapped points $f_{Root}(c_i) \in \mathcal{D}$, where f_{Root} is the map of the *Root* GTM. The child GTMs are initiated by local PCA in the corresponding Voronoi compartments [Tiňo and Nabney, 2002]. After training the child GTMs and seeing the lower level visualisation plots, the user may decide to proceed further and model in greater detail some portions of the lower level plots, etc.

When the dataset is very large, the higher-level projection plots may be cluttered and confused (with densely clustered and overlapping projections). This makes it difficult for the user to select locations for submodels at the next level. In such cases, an alternative semi-automatic submodel initialisation algorithm [Nabney et al., 2005], based on minimum message length (MML) [Wallace and Dowe, 1999] criteria, which decides both the number of submodels and their location can be used for higher-level projections of the visualisation hierarchy and then the domain expert can take control to guide the lower-level projections.

Visualisation is a valuable tool for exploring and understanding data, but in many applications the fundamental task is one of prediction. It has been argued that a single global classification/regression model can rarely capture the full variability of a huge multi-dimensional dataset. Instead, local models, each focused on a separate area of input space (a cluster), often work better since the mapping in different areas may vary. The framework also supports the guided mixture of local experts model, which uses the soft segmentation obtained using probabilistic hierarchical visualisation algorithms, such as HGTM, to formulate the guided local mixture of experts model [Maniyar and Nabney, 2005]. Thus the visual mining framework is not just a visual exploration tool but also supports guided modelling where the domain expert is closely involved. Details on this are in Chapter 4.

One of the main novelties of this framework is that we have identified principled projection algorithms from the machine learning domain which effectively project multi-dimensional datasets found in the drug-discovery domain and integrated them with suitable visual exploration techniques from the information visualisation domain.

The software tool

We have developed an interactive software tool that supports this integrated visual data exploration framework [Maniyar, 2006]. The interface was developed in MATLAB[2] using the NETLAB toolbox [Nabney, 2001]. The tool supports several projection methods, such as PCA, PPCA, GTM, HGTM, SOM and Neuroscale, and useful information visualisation techniques discussed in Section 3.4. The interface has proved useful for domain experts to understand and mine large high-dimensional datasets. A website[3] is set up for the tool which provides access to the user manual and other information.

3.6 Evaluation methods

In some datasets there are labels attached to data points. We would like the visualisation projection to show good separation between these classes. The class information is not included when training the visualisation models but is used for better presentation (eg. through colour or marker style or both). Though visually we can observe the effectiveness of a projection in such a coloured plot, it is hard to compare objectively projections obtained using different methods. We employed the following three evaluation methods to compare different aspects of the projections.

3.6.1 Kullback-Leibler (KL) divergence

It is useful to get an analytical measurement of the separation between different data classes in the projections. To obtain such a measurement, first we fit a Gaussian mixture model (GMM) [Bishop, 1995] to each class in the projection space and then we calculate the

[2] The MathWorks Inc., http://www.mathworks.com/

[3] http://www.ncrg.aston.ac.uk/~maniyard/dvms/

Kullback-Leibler (KL) divergence [Cover and Thomas, 1991] between the fitted GMMs:

$$D_{KL}(p_a \parallel p_b) = \sum_x p_a(x) \log \frac{p_a(x)}{p_b(x)}, \tag{3.16}$$

where p_a and p_b are the GMMs for classes a and b respectively. Since KL-divergence is an asymmetric measure we calculate D_{KL} for each class pair in both directions and sum the values up to obtain a KL-divergence sum, s_{KL}, as below

$$s_{KL} = \sum_{a=1}^{M} \sum_{b=1}^{M} D_{KL}(p_a \parallel p_b), \tag{3.17}$$

where M is the number of classes. The greater the value of KL-divergence sum, the greater the separation between classes.

3.6.2 Nearest-Neighbour (NN) classification error

Though data visualisation is an unsupervised learning problem, it can be useful to objectively evaluate the quality of a classifier based on the visualisation output. We calculate the Nearest-Neighbour (NN) classification error when we classify each data point according to the class of its nearest neighbour in the two dimensional latent space obtained by the visualisation algorithms.

3.6.3 Magnification Factors (MF) sum

As discussed in Section 3.4.1, one of the main advantages of using GTM–based models is that it is possible to analytically calculate manifold properties such as the magnification factors (MF). We sum the MF for each grid on the projection manifold to obtain an overall measure of the magnification. This measurement is useful to compare manifold properties of GTM-based models in chapter 5 where we present an extension to GTM for feature selection.

3.7 A case study : HTS dataset

The physicochemical properties of a drug have an important impact on its 'drug-likeness' and safety aspects so a careful study these properties coupled with their biological response against a target is crucial for a successful drug discovery programme. A typical challenge in the early stages of the drug discovery process is to understand and explore large datasets containing high-throughput screening (HTS) results (biological activity) alongside some whole-molecule properties [Englebienne, 2005]. Screening scientists are interested in studying and exploring

Label Description	Marker	Compounds
Not active in any screen	•	21540
Active for peptidergic type1	+	236
Active for peptidergic type2	*	362
Active for aminergic type1	□	100
Active for aminergic type2	△	818
Active for kinase	◇	412
Active for more than 1 screen	○	132

Table 3.2: Marker information and compound distribution across labels for the HTS dataset.

clusters of active compounds to understand the data and make informed decisions for future library design and assays development. The HTS dataset provided by the chemists at Pfizer is described below:

3.7.1 The dataset

The HTS dataset is composed of 23,600 compounds with values for biological activity data (% of response[4]) for five different biological targets and 11 whole-molecular physicochemical properties.

Out of these five biological targets, two are peptidergic G-Protein coupled receptor (GPCR) targets, two are aminergic GPCR targets, and one is a kinase target. The four GPCR targets are of related receptor types whilst the kinase is a completely unrelated enzyme target class. Table 3.2 lists the label information and distribution of compounds in different labels. Table 3.3 lists the physicochemical properties used.

3.7.2 Preprocessing

Since different input variables in the dataset have different ranges, before the development of visualisation models we apply a linear transformation (Z-score transformation) to have similar ranges for all variables. Each variable is treated independently and is rescaled as follows:

$$\mu_i = \frac{1}{N}\sum_{n=1}^{N} {x_i}^n \tag{3.18}$$

$${\sigma_i}^2 = \frac{1}{N-1}\sum_{n=1}^{N}({x_i}^n - \mu_i)^2, \tag{3.19}$$

[4] Efficacy of compound at a single concentration, which is expressed as $\frac{(Measured\ response - Minimum\ response)}{(Maximum\ response - Minimum\ response)} * 100$.

Property name	Description
ALogP	Logarithm of the ratio of a molecule's solubility in n-octanol to its solubility in water
Molecular solubility	Logarithm of solubility of compound in water (measured in mol/litre)
Number of atoms	Total number of atoms in the compound
Number of bonds	Total number of bonds in the compound
Number of hydrogens	Total number of hydrogens in the compound
Number of ring bonds	Total number of rings in the compound
Number of rotatable bonds	Number of single bonds between heavy atoms that are both not in a ring and not terminal
Number of hydrogen acceptors	Total number of hydrogen acceptors in the compound
Number of hydrogen donors	Total number of hydrogen donors in the compound
Molecular polar surface area	Total surface area for nitrogen and oxygen atoms, and any atom with a non-zero formal charge
Molecular weight	The molecular weight of the compound

Table 3.3: Molecular physicochemical properties used in the HTS dataset.

where $n = 1, ..., N$ indexes the patterns, and μ_i and ${\sigma_i}^2$ represent mean and variance of variable i respectively. Then the values are scaled by

$$\tilde{x}_i^n = \frac{{x_i}^n - \mu_i}{\sigma_i}, \tag{3.20}$$

where $\tilde{x}_i^n$ is the scaled value of variable i for the pattern n. Figure 3.9 shows histograms of all the 16 variables after the scaling. These show approximately normal distributions.

3.7.3 The analysis

We consider two applications using this dataset. In the first application, the aim is to understand and explore a large dataset containing past HTS results (biological activity data) alongside the whole-molecule properties. The objective in the second application using this dataset is to visualise the dataset with only the whole-molecule physicochemical properties as input to understand and prioritise unscreened compounds from a virtual library[5] to select compounds for future HTS campaigns.

50% of the dataset was used as the training set and the remaining 50% was used as the

[5] A compound library which has no physical existence, being constructed solely in electronic form. The building blocks required for such a library may not exist, and the chemical steps for such a library may not have been tested. These libraries are used in the design and evaluation of possible libraries (physical).

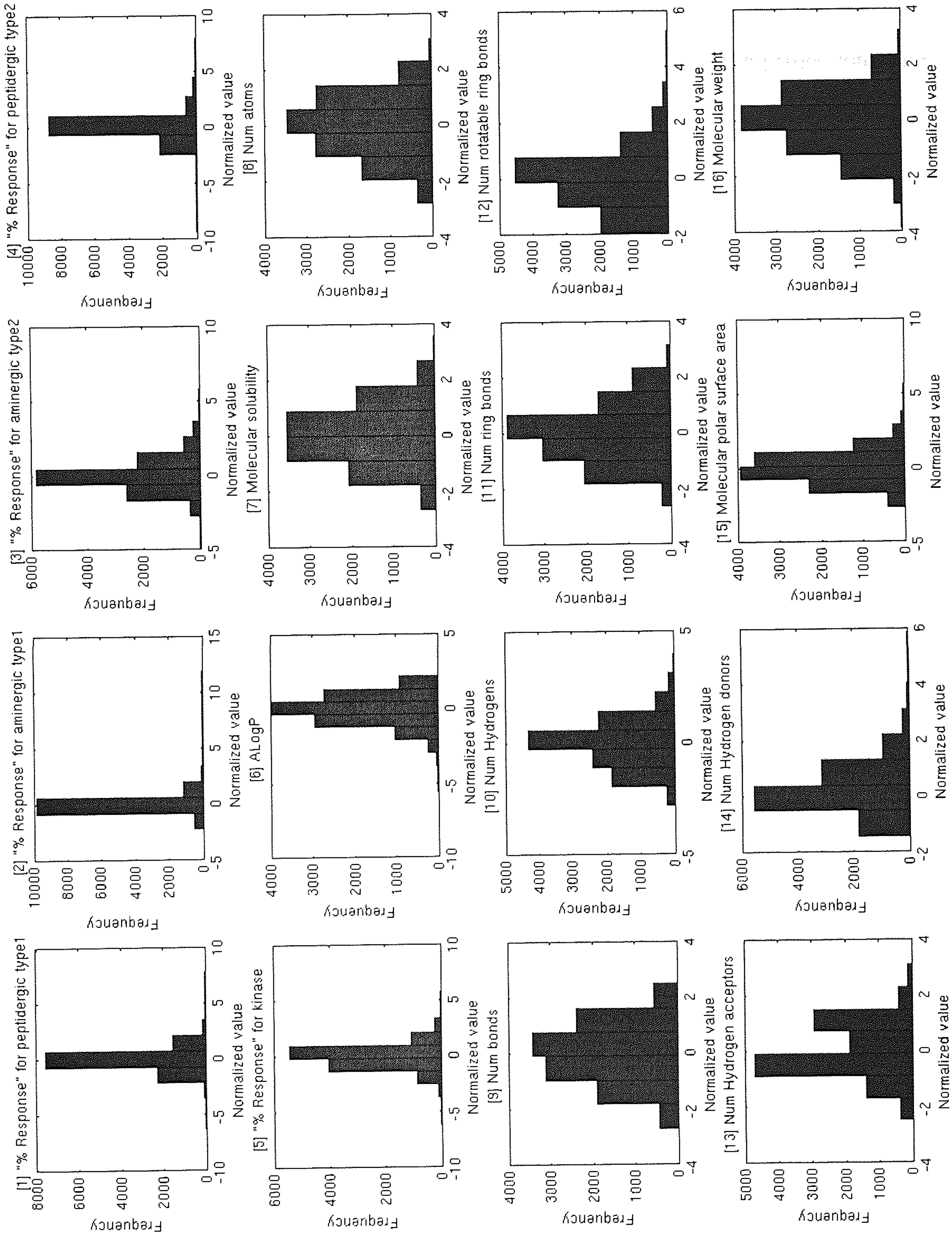
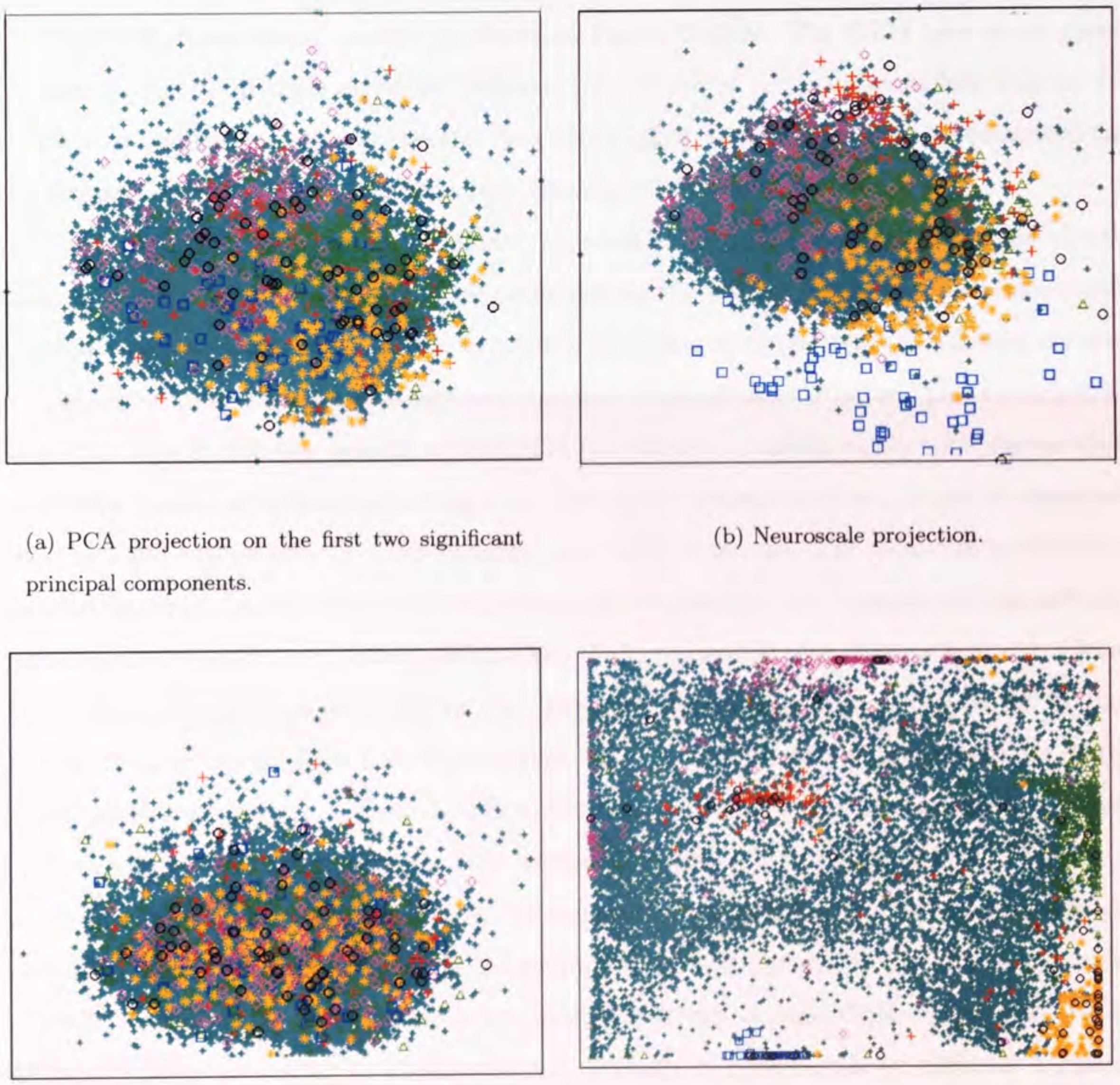


Figure 3.9: Histogram of scaled input variables for test set of the HTS dataset.

test set. Visualisation results presented are on the test set.

3.7.4 Application 1

In this section, we elaborate on the results obtained using some of the projection algorithms discussed in Section 3.3. For this analysis, we use all 16 descriptors (5 biological activity data + 11 physicochemical properties) in the dataset to obtain an effective projection onto a 2-dimensional manifold. The aim is to obtain clusters of compounds active against different targets in the projection.



(a) PCA projection on the first two significant principal components.

(b) Neuroscale projection.

(c) SOM projection.

(d) GTM projection.

Figure 3.10: PCA, Neuroscale, SOM, and GTM projections. Refer to Table 3.2 for legend.

Not surprisingly, for an extremely large high-dimensional dataset such as this, PCA, Neuroscale, and the SOM did not prove effective. The projections obtained using PCA (on the first two significant principal components), Neuroscale, and SOM visualisation are presented in Figure 3.10(a), Figure 3.10(b), and Figure 3.10(c) respectively. The visualisation results of all these three algorithms do not show any useful separation of the active compounds from the inactive compounds. The projected data is like a 'blob' and there is no apparent clustering; this does not give much insight into the detailed structure of the data. Active compounds are distributed all over the plots instead of forming any cluster and thus the results are not very useful to understand the dataset.

The GTM visualisation results are shown in Figure 3.10(d). The GTM plot shows clear clusters for the compounds active for different targets and is certainly more informative. It is easier to understand and explore the data space using the GTM projection compared to the projection we obtained using the other visualisation techniques.

The information visualisation facilities (discussed in Section 3.4) integrated in the visual data exploration framework have proved useful during the various stages of data exploration. Local parallel coordinate plots help us to observe variations in the patterns in different regions of a projection plot. Figure 3.11 shows how patterns of biological activity and physicochemical properties vary in different regions of the GTM projection. A careful study with the parallel coordinate technique reveals interesting structures in the projection space. It can be observed that the active compounds for different targets are nicely clustered. The compounds active for peptidergic type 1 (marked as $+$) and peptidergic type 2 (marked as $*$) targets are respectively clustered at the middle and bottom-right of the GTM projection plot (Figure 3.10(d)). Close study using the software tool reveals that the compounds marked as '$\circ$', present in the clusters for peptidergic type 1 and peptidergic type 2, are the active compounds for both of the peptidergic targets. That is in line with the fact that some compounds are active for both of the peptidergic targets. The compounds active for aminergic type 1 (marked as $\Box$) and aminergic type 2 (marked as $\triangle$) targets are respectively clustered separately at bottom-left and middle-right of the GTM projection plot (Figure 3.10(d)). The compounds active for kinase target (marked as $\Diamond$) are mostly clustered at top-middle of the plot. Such different clusters are useful to understand the diversity of compounds for different targets. The compounds active for more than one target (marked as $\circ$) are useful to observe overlaps and to understand the similarity of compounds active for different targets. It was observed that many inactive compounds (marked as ●) near the active compounds have activities

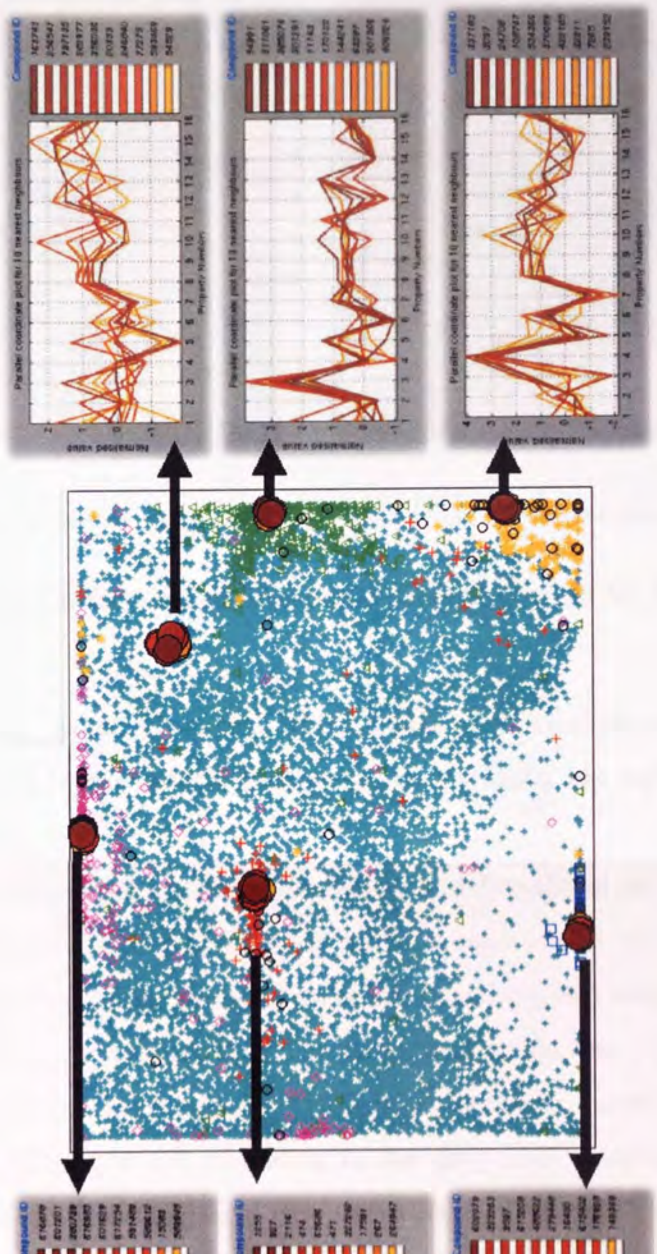
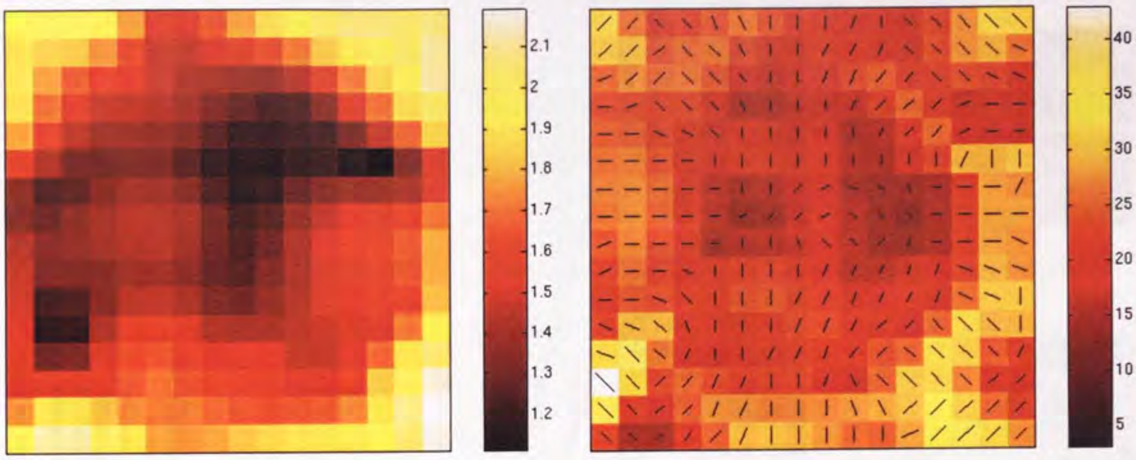


Figure 3.11: Local parallel coordinates demonstrating variations in the patterns in different regions of the GTM projection (plot 3.10(d)). Refer to Table 3.2 for legend.



(a) Magnification factors plot on a $\log_{10}$ scale.

(b) Directional curvatures plot.

Figure 3.12: Magnification factors and directional curvatures plots for the GTM projection (plot 3.10(d)).

values near to 40% (the threshold we set), which means they are close to being active. They are mostly the border line cases which, because of the threshold, are separated in a different bin.

Moreover, the corresponding magnification factors and directional curvatures plots, for the GTM projection, presented in Figure 3.12, are useful to understand the structure of data in the data space. Using these plots, we can observe the stretching and folding of the projection manifold in the data space respectively. The magnification factors plot is represented by colour shading in the projection manifold. The lighter the colour, more the stretching in the projection manifold. The direction of folding in the projection manifold plot is presented using the direction line in the directional curvatures plots. The length and the shade of the background colour represents the amount of folding. The longer the line and the lighter the background colour, higher the folding (curvature).

Magnification factor and directional curvature plots are also useful for making decisions about number and the positions of the centres for GTM subplots during the training of an HGTM model. For example, the lighter bands at the bottom right corner in the directional curvature plot (see 3.12(b)) reveals a large fold in the projection manifold to cover the data space. This helps us to understand that there could be a cluster there even though the data points are not marked (labelled) differently. Magnification factors and directional curvatures plots are mainly used to understand projection space and data space in detail. But if the data are not coloured (labelled) (for example if we do the analysis on a virtual compound

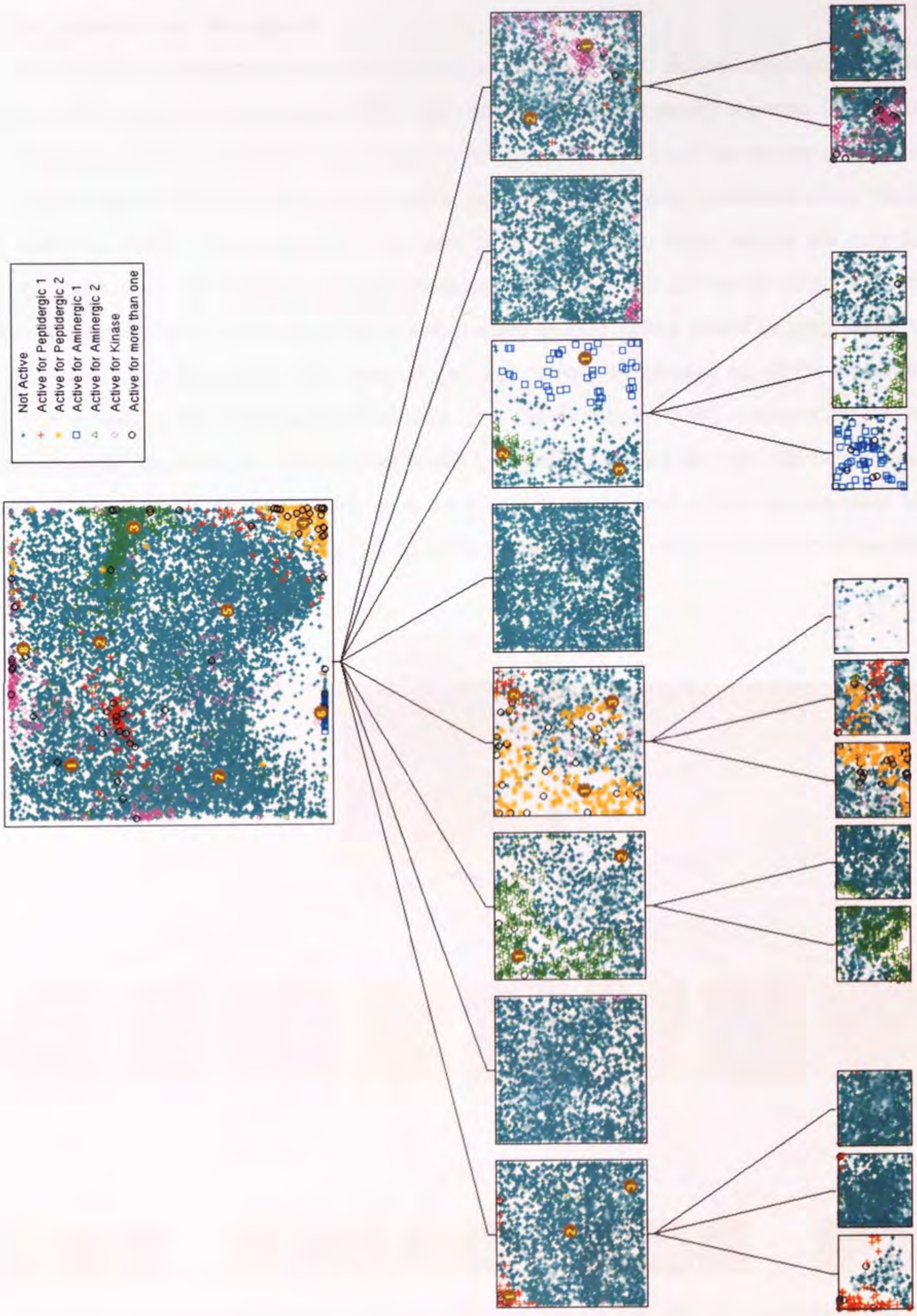


Figure 3.13: HGTM projection. Refer to Table 3.2 for legend.

library), magnification factor and directional curvature plots can be used to observe clusters in the projection and data space.

The HGTM visualisation results are presented in Figure 3.13. Active compounds can be seen in different clusters in the root GTM. The deeper level plots clearly separate interesting local regions. At each level, the magnification factors and directional curvatures plots, presented in Figure 3.14 and Figure 3.15 respectively, are used to make decisions about where to place the 'centre' of a subplot for the next level. Note that these results are only for 11,800 compounds. The number of compounds one has to consider during the drug discovery process is enormous; in such situations a well-trained HGTM model would be very useful to explore data at deeper levels. Also note that all data points are plotted on all the submodel projection plots in the visualisation hierarchy, with the density of "ink" in proportion to the corresponding responsibility which a submodel projection plot has for that particular data point. Thus, if one particular submodel in the hierarchy takes most of the responsibility for a particular data point, then that point will effectively be visible only on that corresponding submodel plot.

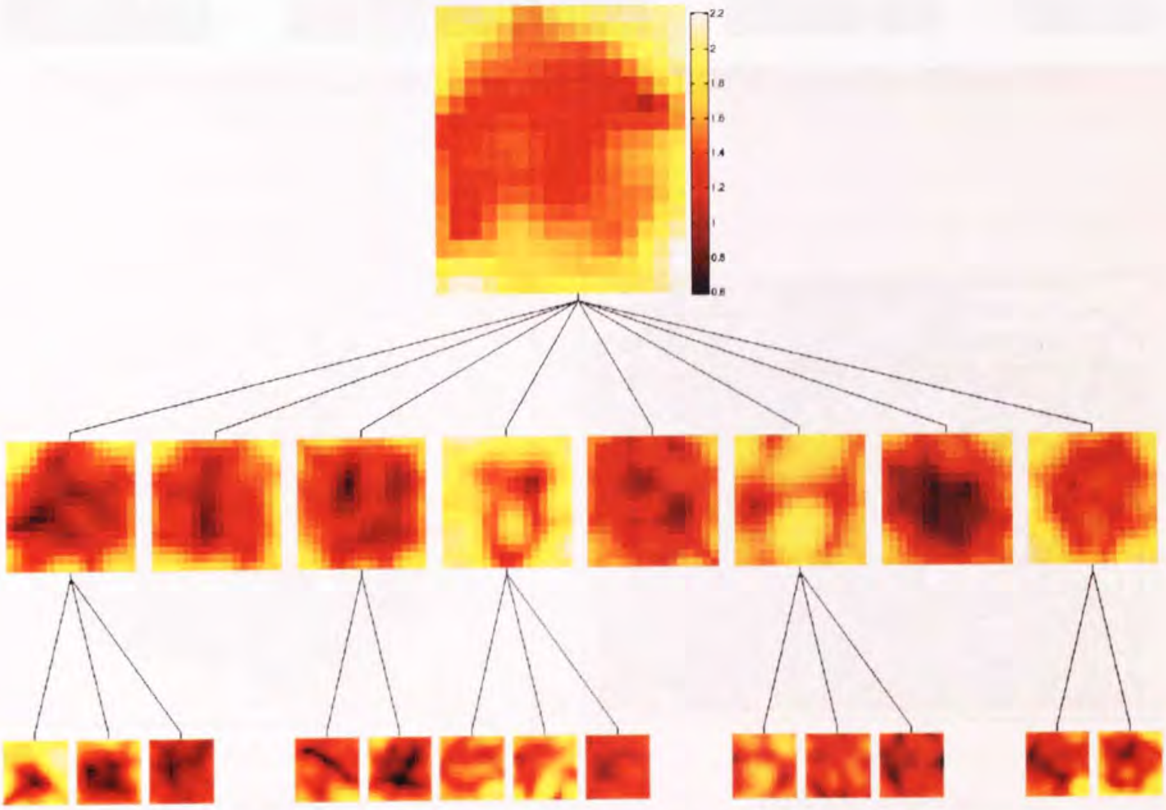


Figure 3.14: Magnification factors plot on a $\log_{10}$ scale for the HGTM projection (Figure 3.13).

Though visually we can easily observe the effectiveness of GTM projection on this dataset,

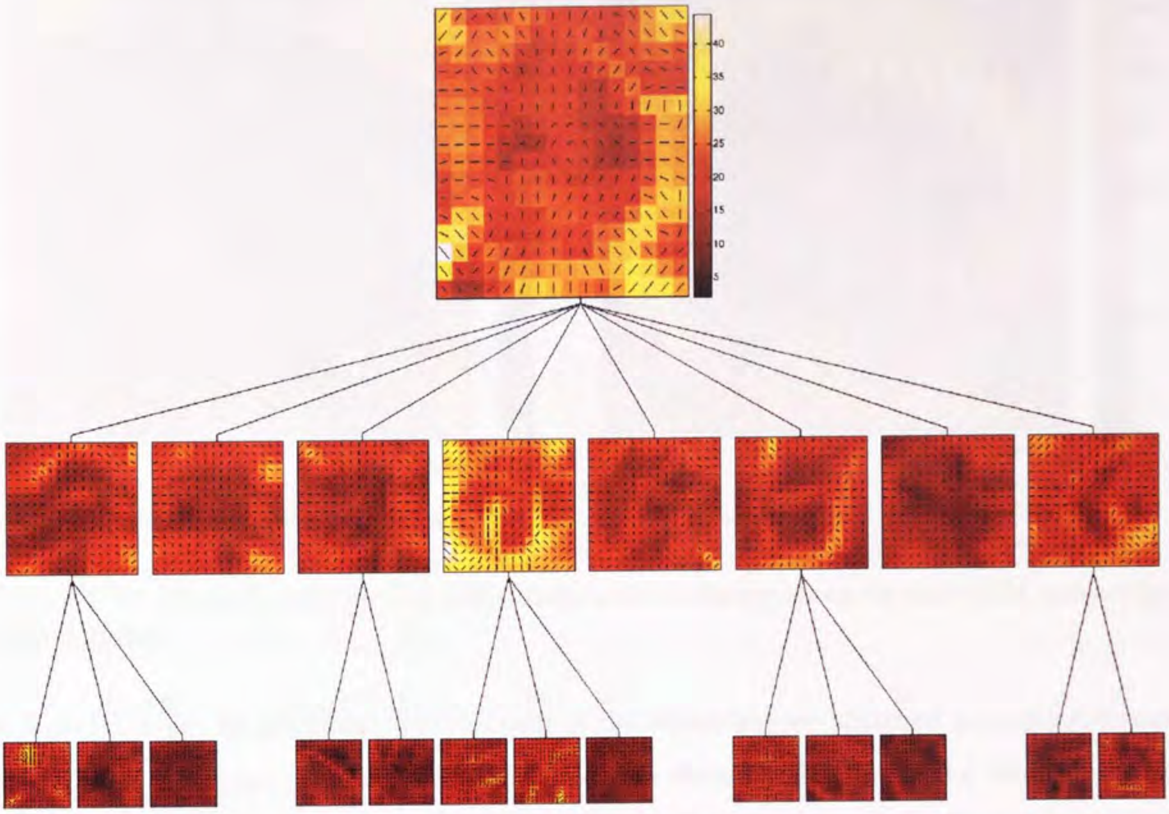
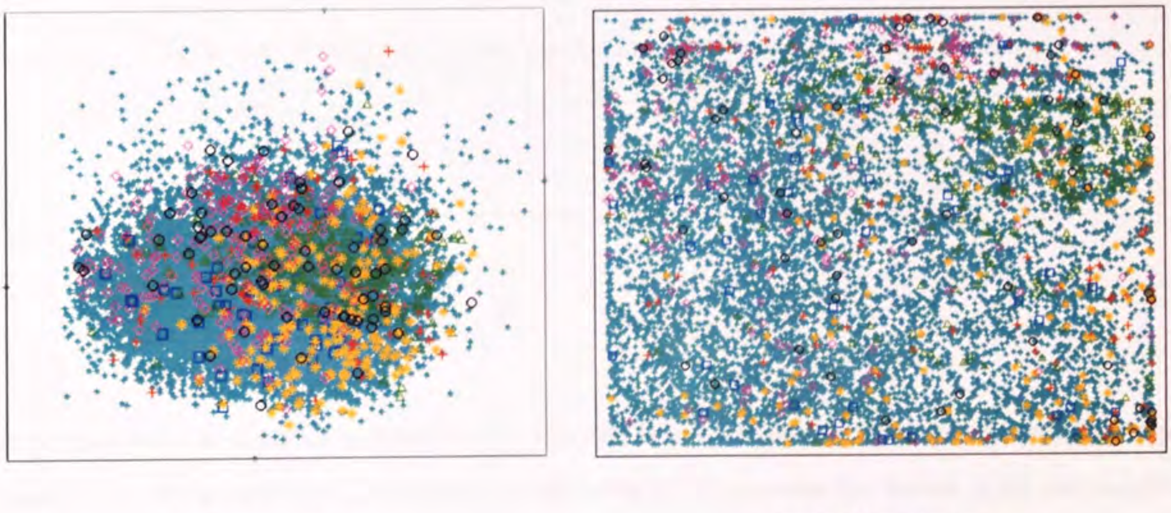


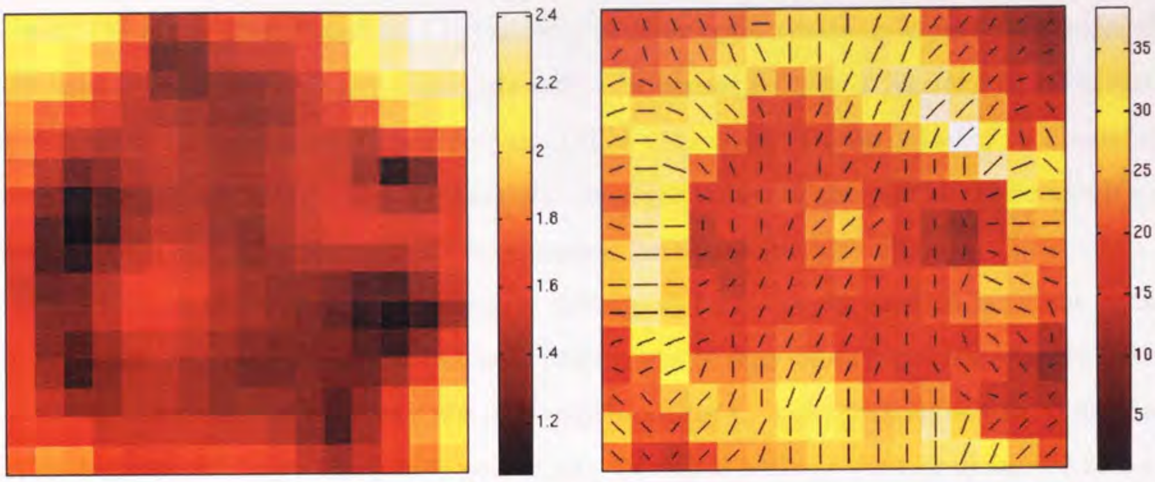
Figure 3.15: Directional curvatures plot for the HGTM projection (Figure 3.13).



(a) PCA projection.

(b) GTM projection.

Figure 3.16: PCA and GTM projections using only the whole-molecule physicochemical properties. Refer to Table 3.2 for legend.



(a) Magnification factors plot on a $\log_{10}$ scale.

(b) Directional curvatures plot.

Figure 3.17: Magnification factors and directional curvatures plots for the GTM projection (plot 3.16(b)).

it is useful to get an analytical measurement of the separation we obtained amongst different data classes in the projections. Evaluation methods discussed in Section 3.6 were employed for analytical comparison of projections obtained using different models. Table 3.4 presents performance of different models on these evaluation criteria. Since a screening scientist is interested in increased accuracy of prediction for active compounds, the NN-classification error for active compounds is reported in Table 3.4 instead of overall NN-classification error.

Table 3.4: Evaluation of the projection results on the HTS dataset.

Method	PCA	Neuroscale	SOM	GTM
MF sum	-	-	-	**125.92**
KL divergence	50.25	52.18	56.37	**128.17**
NN error (%)	93.10	92.85	92.40	**38.32**

3.7.5 Application 2

It is also useful to visualise a dataset with only the whole-molecule physicochemical properties as input to understand and prioritise the unscreened compounds for future HTS campaigns.

Figure 3.16 presents the PCA and GTM projections we obtained using the dataset with only the 11 physicochemical properties described in Section 3.5. Here, we can observe a soft grouping in the GTM projection (Figure 3.16(b)) for this dataset compared to a blob we obtain in the PCA projection (Figure 3.16(a)) Neuroscale and SOM projection results are similar to PCA. The dominance of compounds active for different targets in different regions of the GTM projection (Figure 3.16(b)) could give us an opportunity to focus on the selected compounds. For example, the compounds active for aminergic type 2 (marked as △) target are dominant in the top-right corner of the GTM projection (Figure 3.16(b)) so a careful study of compounds grouped in and near the top-right corner of the plot could be useful to prioritise compounds for the next HTS campaigns for aminergic type 2 target.

Note that in these plots, the biological activity data is merely used to label the data points on the plot for better presentation. Projections of only unscreened compounds from the virtual compound library may not have any labelling as if the compounds are from the virtual compound library there may not be any biological activity data available for them. Magnification factors and directional curvatures plots for the GTM projection manifold, Figure 3.17, are useful in such a situation to understand the GTM projection manifold. It is also useful to add some already screened compounds in the dataset so that we can easily locate regions of interest on the projection manifold by observing the positions of the screened active compounds on the projection manifold.

3.7.6 Discussion

The number of input variables, and the structure and quantity of the data, make it difficult to obtain a good projection using traditional visualisation algorithms. An attempt to preserve relative similarities between the higher-dimensional input data and lower-dimensional projection space using PCA fails because of its linear nature. As we can see from the component matrix (Table 3.5), factor loadings of all the biological activity variables are negligible for the first two significant principal components (PC). The activity data is explained mostly by PC 4 and 5. Figure 3.18 shows projection using these two PCs. This projection provides a much better clustering of active compounds than the PCA projection obtained using the first two significant PCs (Figure 3.10(a)). The problem in doing so is that the physicochemical properties do not contribute greatly, so any analysis of inactive compounds adjacent to the actives is compromised.

Projection using Neuroscale gives us a blob because the outliers dominate the stress matrix used to fit the visualisation model. The GTM first models the distribution of data in the data space, and then it gives us a uniformly distributed projection which provides better clustering. Projection results using GTM not only enabled us to characterise hit populations from different target classes (i.e. peptidergic GPCRs vs. aminergic GPCRs vs. kinases) but

	Significant Principal Components (with Eigenvalue > 1)				
	1	2	3	4	5
Number of atoms	.984				
Number of bonds	.974				
Molecular weight	.968				
Number of hydrogens	.757			.327	
Number of ring bonds	.740	−.310	.310	−.312	
Molecular solubility	−.726	.478	.301		
Number of rotatable ring bonds	.584	.403	−.268	.411	−.321
Molecular polar surface area	.393	.781			
AlogP	.511	−.715			
Number of hydrogen acceptors	.517	.684	.268		
Number of hydrogen donors		.660	−.395		
Active for aminergic type2			.694		
Active for aminergic type1				.561	
Active for peptidergic type2				.392	.389
Active for kinase				−.355	
Active for peptidergic type1					.768

Table 3.5: Component matrix with factor loading sorted in ascending order. The threshold for a loading, l, to be included is $\mid l \mid > 0.3$.

also to understand areas of overlap. The observation that a few physicochemical parameters with a combination of screening results can be successfully used to characterise active from inactive compounds is in agreement with the recent work of Diller and Hobbs [2004]. It is also in line with the observations at Pfizer that the properties (molecular weight, AlogP, polar surface area etc.) of molecules active against different target classes show differences in the distribution and median of the property values [Gribbon and Sewing, 2005]. Visualisation allows the screening scientists to understand these relationships in relation to specific compounds and in greater depth and detail.

A single 2D projection is not enough for a large dataset since many points are projected on top of each other. Thus the HGTM is useful to explore regions of a top level GTM projection in detail.

One of our main aims in this analysis, to understand and explore biological activity data combined with other whole-molecule physicochemical properties (Application 1, Section 3.7.4), was triggered from the needs of the screening scientists at Pfizer. The GTM and HGTM projections have proved very useful for the purpose and have out performed the projections obtained from other visualisation techniques. Even with only whole-molecule physicochemical properties as input during the model creation (Application 2, Section 3.7.5), the GTM gave a relatively soft but still useful grouping. As one can expect, the grouping in the projection obtained with only whole-molecule physicochemical properties as input is less clear than the grouping in the projection obtained with the biological activity data also included, since in the former case the model has less information to learn from. The performance could be improved by including more useful whole-molecule properties in input dataset (structure information, past HTS results on similar targets, etc.).

Clearly, the leaf nodes of the HGTM hierarchy in Figure 3.13 represent individual groups of compounds. Using these groupings, it is possible to develop local classifiers to classify compounds for activity using physicochemical properties and past screening data. A single global classification/regression model can rarely capture the full behavioural variability of a huge multi-dimensional dataset such as one used here. Instead, local classification/regression (expert) models, each focused on a separate area of input space, often work better since the behaviour of different areas may vary.

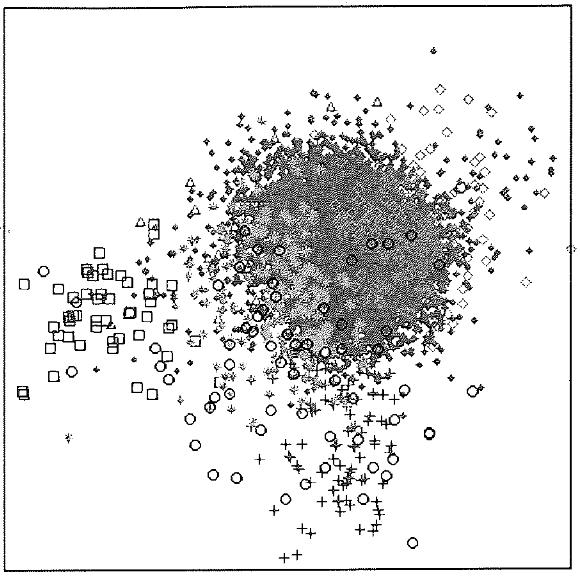


Figure 3.18: PCA projection on PC 4 and 5. Refer to Table 3.2 for legend.

3.8 Computational cost

Although the rapid development of high-performance computing has to some extent altered our perception of computational complexity, this issue cannot be ignored in a visual data exploration framework where user interaction is important.

The computational cost for PCA scales linearly, $\mathcal{O}(N)$, in the number of data points (N). Neuroscale suffers from the fact that the computational demands grow with the square of the number of data points, $\mathcal{O}(N^2)$. This is because each evaluation of the stress error requires the computation of $N(N-1)/2$ inter-point distances. In practice, for large data sets, it is common to apply an initial clustering phase to the data set (using for example the K-*means* algorithm), to generate a set of K prototype vectors (where $K \ll N$). Neuroscale can then be applied to these prototype vectors at a much reduced computational cost. Here we used Neuroscale with the fast shadow targets training algorithm [Tipping and Lowe, 1998].

The distance calculation between data points and mixture components of reference vectors, respectively, is identical in SOM and GTM training algorithms. Updating the parameters in SOM training depends on the neighbourhood function. In the experiments presented here it was continuous on the latent space so the parameter updating scales as $\mathcal{O}(M^2ND + M^2)$, where M is the number of grid points in the SOM and D is the dimension of the data space. When updating parameters, the GTM requires a matrix inversion of an $K \times K$ matrix, where K is the number of basis functions, followed by a set of matrix multiplications. The matrix inversion scales as $\mathcal{O}(K^3)$, while the matrix multiplications scales as

$\mathcal{O}(MND)$, where M is the number of grid points in the GTM latent space[6].

Table 3.6 shows the time taken to train different projection models on the training set using an Intel Pentium 4 - 2.4GHz machine with 2GB of RAM. The implementation of the algorithms in C/C++ instead of MATLAB could further improve the speed.

The model	Time (seconds)	Architecture
PCA	1	-
Neuroscale	546	-
SOM	36	$M = 256$
GTM	42	$M = 256, K = 64$

Table 3.6: Training time for different projection models (N = 11800, 20 iterations) for the HTS dataset.

Once the models are trained, the computational cost to project data for the subsequent test set scales linearly, $\mathcal{O}(N)$, in the number of data points (N) in the test set.

3.9 Conclusions

To understand a large high-dimensional dataset, close integration of principled projection methods and information visualisation techniques is useful to develop an effective visual data exploration framework.

Traditional projection algorithms used in drug discovery, such as PCA, Neuroscale, and SOM, are not powerful enough for many real-life scientific problems. For example, for the dataset analysed in this chapter, these techniques proved to be ineffective and failed to generate additional knowledge, as we could not distinguish populations of molecules active for different biological targets. GTM certainly gave much better results. Several evaluation methods clearly showed the effectiveness of the clustering we obtain using GTM compared to PCA, Neuroscale, and SOM. The GTM algorithm is known as a 'principled' alternative to SOM because it is derived from probability theory and statistics, whereas the SOM is motivated by heuristic and empirical arguments. Because of its sound theoretical base, other than similar or better projection results than SOM, useful manifold properties such as magnification factors and directional curvatures can be calculated for a GTM projection.

Magnification factor and directional curvature plots of GTM helped to provide a better understanding of the projection manifold and its fitting on data in the data space. The local parallel coordinates technique proved to be a useful tool to understand data points in

[6] To be exact, the matrix multiplications scales as $\mathcal{O}(MKD + MND)$, but normally the number of data points, N, exceeds the number of basis functions, K.

interesting regions of the projection manifold more in detail. Since the structure of compounds is very important in drug discovery, billboarding could be a useful feature for the domain experts at Pfizer to visualise chemical structures in the projection manifold.

Hierarchical GTM models are useful to explore clusters and interesting local regions in details in a large dataset. The number of compounds one has to consider during the drug discovery process is enormous; in such situations, a single GTM projection can look cluttered but a well trained HGTM model could be very useful to provide a better grouping. Effective groupings obtained using HGTM can be used to develop powerful local predictive models [Maniyar and Nabney, 2005] as described in the next chapter.

The computational cost of training a GTM model is acceptable for inclusion in the visual data exploration framework. The GTM and HGTM algorithms are scalable so having a large number of data points during training, causes no difficulty beyond increased computational cost.

The interface developed using this framework and following Shneiderman's design guidelines provided us with a useful tool for better understanding and exploration of large high-dimensional datasets. A loose integration (by having capability of exporting the projection results) of this tool with other industry standard software used in drug discovery domain such as SpotFire[7] has given the domain experts more flexibility and a greater range of tools to work with. Thus the scientists could now apply new algorithms like GTM and HGTM while making best use of their existing software. Combining effective use of all these software tools is likely to increase the chances of identifying active molecules and linking compound properties with biological activity.

[7] Spotfire: http://www.spotfire.com/

Chapter 4

Guided Mixture of Local Experts

In the previous chapter we introduced a visual data exploration framework which helps in exploring large high-dimensional datasets. Though, understanding data is very useful, many tasks involved in chemoinformatics are of prediction, either classification or regression [Waterbeemd and Gifford, 2003; Tiňo et al., 2004; Plewczynski et al., 2006]. A single global prediction model cannot capture the full variability of a large & complex data space, such as chemical space, since the mapping in different regions of the data space may vary. Probabilistic hierarchical visualisation techniques can provide an effective soft segmentation, which means that points belong to more than one region, of an input space by a visualisation hierarchy whose leaf nodes represent different regions of the input space. We use this soft segmentation to develop a guided mixture of local experts (GME) algorithm. Moreover, in this approach the domain experts are more involved in the model development process which is appropriate for a task, such as drug discovery, that requires intuition and domain knowledge for its successful completion. The performance of the algorithm on real-world datasets from chemoinformatics is better than the conventional mixture of experts model and popular global predictive models.

4.1 Introduction

As discussed in Chapter 2, because the high overall attrition rate in drug discovery is caused mostly by limited 'drug-likeness' of the compounds, the early prediction and analysis of drug-likeness has became common practice in pharmaceutical research. The aim of molecule screening according to 'drug-likeness' properties is early identification and elimination of candidate molecules that are unlikely to survive later stages of drug discovery ('fail-early, fail-cheap'). Various statistical and machine learning methods are applied to predict 'drug-likeness' parameters (see Section 2.3.2 for further details). Most of these methods involve developing a single global model for a particular series of compounds or region of the input space. A barrier to effective prediction using such models is that reliably accurate prediction is limited to a particular region of chemistry space that is covered by the compounds in the training set, they fail if the datasets have great diversity of compounds. Moreover, screening scientists (chemists, biologists, etc.) see many prediction models as 'black boxes' as often prediction models developed using compounds from certain region of chemical space do not work effectively for compounds from a different region of chemical space. In a domain knowledge-driven research process like drug discovery, it is important to involve the domain experts in the model development process. Therefore, we have developed a modelling approach which not only can effectively work with heterogeneous spaces such as chemical space but also involves domain experts.

It has been argued that a single global prediction model can rarely capture the full variability of a huge multi-dimensional dataset. Instead, local models, each focused on a separate area of input space, often work better since the mapping in different areas may vary. One of the most important aspect of developing such local model is to segment the input space effectively so that each sub-model is trained on a restricted region with limited overlap.

For classification problems, a widely applied method for implementing the Bayes classifier is based on obtaining the posterior probabilities of class membership through the estimation of class prior probabilities and class-conditional densities [Duda et al., 2000]. One of the popular ways to obtain these estimates is independently to apply density estimation methods to each class-labelled dataset (a hard segmentation). However, such an approach does not benefit from the existence of any common characteristics among data of different classes (i.e. segmentation is based on target rather than input). For example, different area of input space (clusters) may have common characteristics, e.g. it is common to have clusters in chemical space containing molecules from different series with similar behaviour.

Alternatively, with advances in probabilistic approaches it is possible to divide the problem into sub-problems which can have common elements – a 'soft split' of the input space into a series of overlapping clusters. Local models developed using soft segmentations, such as mixture of experts (MoE), are popular in the machine learning community for prediction tasks [Jacobs et al., 1991]. The MoE can be viewed as a conditional mixture model in which the distribution of the target variables is given by a mixture of component distributions in which the components, as well as the mixing coefficients, are conditioned on the input variables. The component distributions are referred to as experts, while the mixing coefficients are controlled by a gating network. Values for the model parameters can be estimated using maximum likelihood, for which there exists an efficient EM algorithm (further details of MoE are presented in the next section). Thus the segmentation of the input space is determined simultaneously with the training of the local experts. Though this process is automatic and faster then the two step approach presented in this chapter, for large multi-dimensional input space, the soft segmentation so obtained is not always appropriate, which can affect the overall performance of the model. Moreover, the lack of guidance from a domain expert implies that the segmentation of the input space may not have a useful interpretation. This feature is very important in drug discovery where the screening scientists and medicinal chemists would like to understand and interpret the model and the input space.

The algorithm introduced in this chapter takes advantage of probabilistic visualisation techniques which provide a hierarchy whose leaf nodes represent different regions of the input space. Because of their probabilistic nature, the sub-model conditional density estimates (responsibilities) provide a soft segmentation which can be directly used as mixing coefficients for the development of the mixture of local experts models. Once the responsibilities have been obtained from the visualisation hierarchy, a separate local expert is trained for each leaf node in the hierarchy. Each local expert can be relatively simple, while the lack of flexibility of individual models is compensated for by the overall flexibility of the complete hierarchy. The overall model structure is similar in spirit to the MoE approach, but since the domain experts guide the visualisation hierarchy, the advantage of this approach is that the domain experts are directly involved during the model development process and they may be able to provide a better (and certainly more meaningful) segmentation.

Here we introduce a general framework for using probabilistic hierarchical visualisation to develop a guided mixture of local experts model. We also discuss several ways of using the responsibility matrix obtained from the probabilistic visualisation hierarchy. Building on recent developments in probabilistic hierarchical visualisation algorithms, we apply a semi-automatic method for the development of the visualisation hierarchy which is particularly useful when working with very large datasets.

In the next section we briefly introduce some of the popular prediction models in drug discovery, as mentioned in chapter 2, as they are used to benchmark our approach. In Section 4.3 we review the probabilistic hierarchical visualisation approach and discuss how we can use it to develop an user-informed visualisation hierarchy. In Section 4.4, the guided mixture of local experts algorithm is presented with a discussion on various ways of using soft segmentation obtained using a trained hierarchy to weight models. A straightforward extension to the Bayesian committee machine algorithm is introduced in Section 4.5. Results on two different real-life datasets from chemoinformatics are reported and discussed in Section 4.6. Finally, in Section 4.7 we draw the main conclusions from this chapter.

4.2 Prediction models

In this section, we briefly introduce some popular global prediction models widely used in drug discovery domain and a few local models which have similarities with the guided mixture of local experts model.

4.2.1 Global models

Here, a single model is trained for the problem which is responsible to model the entire input space.

Linear regression (LR)

Linear regression (LR) consists of a linear combination of the input variables. So the output y is a linear combination of input values $\mathbf{x}$

$$y = y(\mathbf{x}; \mathbf{w}) = \sum_{i=1}^{d} w_i x_i + b, \tag{4.1}$$

where d is the dimensionality of the input space and $\mathbf{w} = (w_1, ..., w_d, b)$ is the parameter vector. The following sum-of-squares error function is minimised to find the optimal weights.

$$E = \frac{1}{2} \sum_{n=1}^{N} \{y(\mathbf{x}_n; \mathbf{w}) - t_n\}^2 . \tag{4.2}$$

where t_n is the target and N is the number of data points.

Here since E is a quadratic function of the weights (eq. 4.2), the optimum weights can be found using the pseudo-inverse of the data matrix, a standard technique from linear algebra [Golub and Loan, 1996].

A similar approach for classification known as linear discriminate analysis (LDA) has also been used in drug discovery domain [Mahmoudi et al., 2005].

Multi-layer perceptron (MLP)

The multi-layer perceptron, a traditional artificial neural networks (ANN) architecture, is a non-linear prediction model. Conventional two-layered MLP consists of two layers of adaptive weights with full connectivity between inputs and hidden units, and between hidden units and outputs.

The first layer of the network forms N_{hid} linear combinations of these inputs to give a set of intermediate activation variables which are then transformed by the non-linear activation functions of the hidden layer, here we choose to the tanh function, to give the hidden unit outputs z_j

$$z_j = \tanh\left(\sum_{i=1}^{d} w_{ji}^{(1)} x_i + b_j^{(1)}\right) \qquad j = 1, \ldots, N_{\text{hid}}. \tag{4.3}$$

Here $w_{ji}^{(1)}$ represents the elements of the first-layer weight matrix and $b_j^{(1)}$ are the bias parameters associated with the hidden units.

The z_j are then transformed by the second layer of weights and biases to give the network output y according to an activation function.

$$y = \sum_{j=1}^{N_{\text{hid}}} w_j^{(2)} z_j + b^{(2)}. \tag{4.4}$$

Training of MLP is typically performed using variations of gradient descent based algorithms trying to minimise an error function (according to the regression or classification task). To avoid overfitting cross-validation can be used for finding optimal complexity of the network (number of units in the hidden-layer).

Gaussian process regression (GP)

Though Gaussian processes regression (GP) have just relatively recently become popular in the machine learning community, they have a longer history in spatial statistics, where the technique is also known as "kriging" [Rasmussen and Williams, 2006]. Gaussian processes are particularly suited to regression problems since in these circumstances we can perform the first level of Bayesian inference (computing the posterior distribution of the parameters) analytically.

A Gaussian process is a stochastic process $Y(\mathbf{x})$ where every joint density function is Gaussian and is therefore defined completely by its mean and covariance. For simplicity, we will consider only Gaussian processes with zero mean. The covariance of $Y(\mathbf{x})$ and $Y(\mathbf{x}')$ is usually defined by a function $C(\mathbf{x}, \mathbf{x}')$.

Consider a training data set consists of ordered pairs $(\mathbf{x}_1, t_1), \ldots, (\mathbf{x}_{N_{trn}}, t_{N_{trn}})$. Now suppose that t_i is a sample from a random variable $T(\mathbf{x}_i)$. To make a prediction T^* at a new input $\mathbf{x}^*$ we need to compute the conditional distribution $p(T^*|T_1, \ldots, T_{N_{trn}})$. Since our model is a Gaussian process, this distribution is also Gaussian and is completely specified by its mean and variance. Let $\mathbf{K}$ denote the covariance matrix of the training data, $\mathbf{k}$ denote the $N_{trn} \times 1$ covariance between the training data and T^*, and $\mathbf{k}^*$ denote the variance of T^*. Then $\mathbf{K}_+$, the $(N_{trn}+1)(N_{trn}+1)$ covariance matrix of $(T_1, T_2, ..., T_{N_{trn}}, T^*)$, can be partitioned

$$\mathbf{K}_+ = \begin{bmatrix} \mathbf{K} & \mathbf{k} \\ \mathbf{k}^T & k^* \end{bmatrix} \tag{4.5}$$

The conditional mean and variance at $\mathbf{x}^*$ are given by

$$E[T^*] = \mathbf{k}^T\mathbf{K}^{-1}\mathbf{t}^{trn} \tag{4.6}$$

$$\mathtt{var}[T^*] = k^* - \mathbf{k}^T\mathbf{K}^{-1}\mathbf{k} \tag{4.7}$$

We use the mean as our prediction, while the covariance can be used to compute error bars.

The covariance function is defined by the spherical Gaussian kernel of width σ^2

$$C(\mathbf{x}, \mathbf{x}') = \exp\left(\frac{(\mathbf{x}-\mathbf{x}')^2}{2\sigma^2}\right) \tag{4.8}$$

It is well-known that GP prediction scales badly with the number of training examples. This is because K is an $N_{trn} \times N_{trn}$ matrix, and the matrix inversion needed in (4.6) is $\mathcal{O}(N_{trn}^3)$. This is problematic in chemoinformatics applications, as we need to deal with large training sets to obtain good generalisation performance in chemoinformatics. We therefore applied a sparse on-line version of GP prediction developed by Csato and Opper [2002].

4.2.2 Local models

Local regression models use a combination of models, each of which works on a smaller part of the input space. In this section, we briefly introduce two popular local models we used.

k-nearest neighbour (k-NN) algorithms

Initially, the k-nearest neighbour algorithm (k-NN) was introduced as a method for classifying objects based on closest training examples in the feature space. It can also be used for regression with slight modification of output determination.

First the training points are mapped into multidimensional feature space and then they are used for prediction (classification or regression).

For classification of a new point using k-NN, the point is assigned to the class c if it is the most frequent class label among the k nearest training samples. Usually Euclidean distance is used.

For regression problems: given a test input $\mathbf{x}$ the prediction y^j of the j-th expert is equal to the target value $t_{j(\mathbf{x})}^{trn}$ of the training pair $(\mathbf{x}_{j(\mathbf{x})}^{trn}, t_{j(\mathbf{x})}^{trn})$ whose input $\mathbf{x}_{j(\mathbf{x})}^{trn}$ is the j-th closest point to $\mathbf{x}$ among all the training inputs $\{\mathbf{x}_1^{trn}, ..., \mathbf{x}_{N_{trn}}^{trn}\}$. The output of the model is then the average target value for the K training data points closest to $\mathbf{x}$.

The accuracy of the k-NN algorithm can be severely degraded by the presence of noisy or irrelevant features, or if the features scales are not consistent with their relevance. The algorithm is easy to implement, but it is computationally intensive, especially when the size of the training set grows.

Regression trees (RT)

Regression trees (RTs) are computationally efficient nonparametric models that constitute a good compromise between comprehensibility and predictive accuracy. A regression tree consists of a hierarchy of nodes. With the exception of the bottom nodes (leaves) of the tree, each node contains a logical test on one of the input variables $x_i, i \in \{1, ..., d\}$. Each test has the form [Variable Operator Value] (e.g. $x_2 < 5.7$) and has two possible outcomes, true or false. Any path from the top node (root) to a leaf can be seen as a conjunction (i.e. logical and) of logical tests on the input coordinates. These conjunctions are logical representations of a partition of the input space. Each leaf contains a local predictive model, which in the case of standard RTs is simply a constant value. The local model associated with a leaf operates over a corresponding region of the input space that is defined by the conjunction.

However, constant values in the leaves lead to a regression surface that is not continuous (in fact, a step function). A smoother model is achieved by allowing nonconstant leaf models, e.g. linear functions of inputs $\mathbf{x}$ [Torgo, 1997]. An example of a regression tree with linear models in the leaves is shown in Figure 4.1

The construction of RTs involved a pruning mechanism, where for each hyperparameter value a sequence of trees is generated (using a method called lowest statistical support [Torgo, 1999]), and for every regression tree in that sequence its generalisation error is estimated via cross validation on the training set. The representative RT for each particular hyperparameter

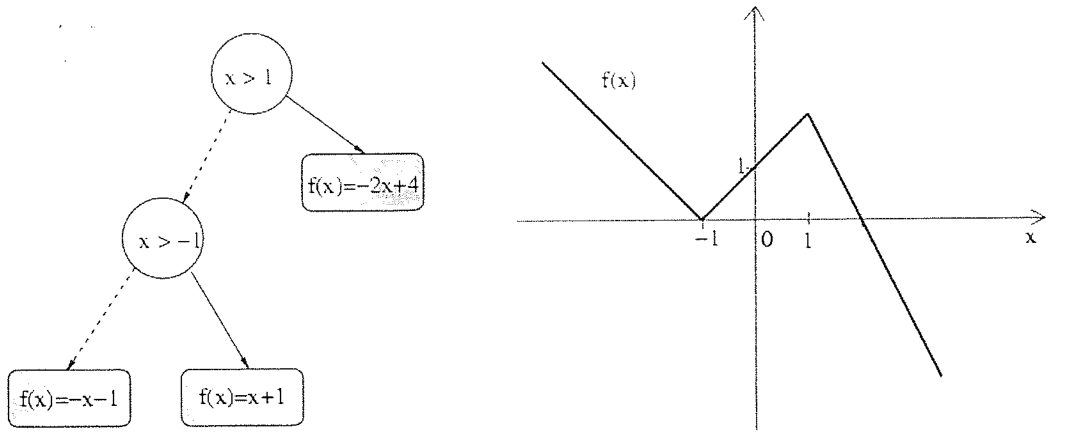


Figure 4.1: Example of regression tree with linear models in the leaves. The tree defines a piece-wise linear function on real line. Arcs corresponding to *true* and *false* decisions are shown as solid and dashed lines, respectively (adapted from [Tiño et al., 2004])

setting is the one with the lowest estimated generalisation error.

Mixture of experts (MoE)

Jacobs et al. [1991] introduced the idea of mixture-of-experts model, which determines decomposition of the data as part of the learning process. The architecture of the mixture-of-experts model is shown in Figure 4.2.

Here all of the expert networks, and the gating network are trained together. The goal of the training procedure is to have the gating network learn an appropriate decomposition of the input space into different regions, with each expert network responsible for generating the outputs for input vectors falling within a specific region. The error function is given by the negative logarithm of the likelihood with respect to a probability distribution given by a mixture of M Gaussians of the form

$$E = -\sum_n \ln \left\{ \sum_{i=1}^{M} g_i(\mathbf{x}^n) O_i(\mathbf{t}^n|\mathbf{x}^n) \right\}, \tag{4.9}$$

where the $O_i(\mathbf{t}|\mathbf{x})$ are regression model with Gaussian noise given by

$$O_i(\mathbf{t}|\mathbf{x}) = \frac{1}{(2\pi)^{c/2}} \exp\left\{ -\frac{(\mathbf{t} - \boldsymbol{\mu}_i(\mathbf{x}))^2}{2} \right\} \tag{4.10}$$

There is one expert network for each Gaussian, and the output of the ith expert network is a vector representing the corresponding conditional mean $\boldsymbol{\mu}_i(\mathbf{x})$ where $\mathbf{x}$ is the input vector. The mixing coefficients $g_i(\mathbf{x})$ are determined by the outputs γ_i of the gating network through

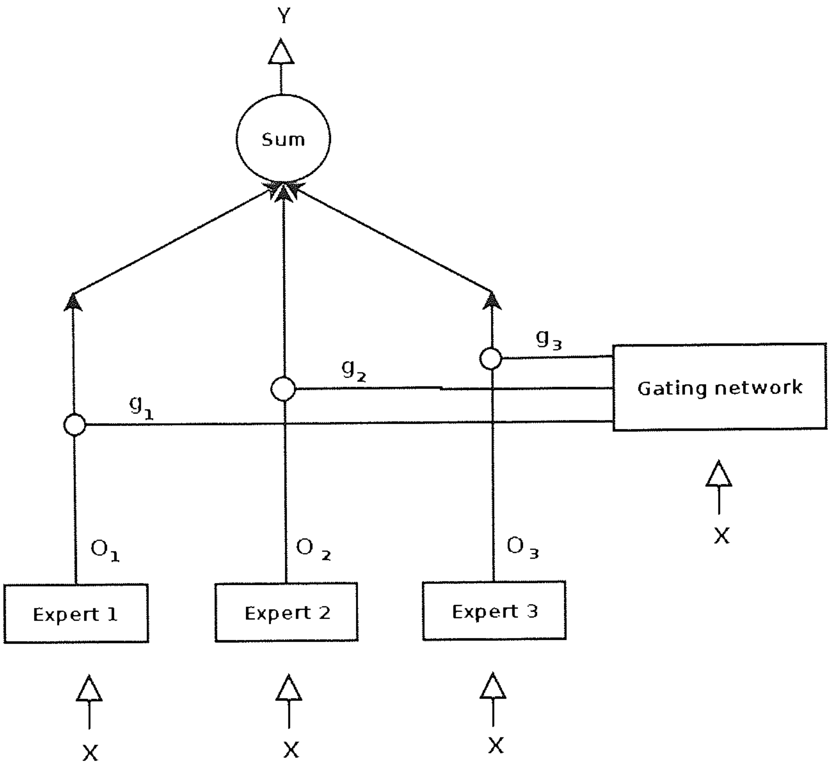


Figure 4.2: Architecture of the mixture-of-experts network during prediction (adapted from [Bishop, 1995]).

a softmax activation function

$$g_i = \frac{\exp(\gamma_i)}{\sum_{j=1}^{M} \exp(\gamma_i)}. \tag{4.11}$$

The gating network has one output for each of the expert networks, as indicated in Figure 4.2.

The mixture-of-experts network is trained by minimising the error function (4.9) simultaneously with respect to the weights in all of the expert networks and in the gating network. The standard choices for gating and expert networks are linear regression (LR) models or multi-layer perceptrons.

When the trained network is used to make predictions, the input vector is presented to the gating network and all of the expert networks. The output vector of a MoE is the weighted (by the gating network outputs) mean of the expert outputs:

$$\mathrm{y}(\mathrm{x}) = \sum_{i=1}^{M} g_i(\mathrm{x}) O_i(\mathrm{x}) \tag{4.12}$$

The gating network outputs $g_i(\mathrm{x})$ can be regarded as the probability that input $\mathbf{x}$ is attributed to expert i. This probabilistic interpretation is ensured because of the choice of gating network as the soft-max function (eq. 4.11).

Jordan and Jacobs [1994] also formulated hierarchical mixture of experts model (HMoE).

The hierarchy can be specified by the depth D of the hierarchical tree and the branching factor BF-the number of children of each internal node. The number of experts in the hierarchy, i.e., the number of leaves of the hierarchical tree, is given by $(BF)^{D-1}$.

Bayesian committee machine (BCM)

There are two important difficulties in applying standard GPs (reviewed in 4.2.1) to large and heterogeneous datasets. Firstly, inference on the GP scales poorly with the dataset size; typically requiring $\mathcal{O}(N^3)$ time, where N is the number of data points. Secondly, GP models are usually stationary as the same covariance structure is used throughout the entire input space. In chemoinformatics applications, where different regions of chemical space may have different behaviour, this limitation is unacceptable. These shortcomings can be addressed by partitioning the input space into regions, and fitting separate GPs within each region. Partitioning allows for non-stationary behaviour, and can reduce some of the computational demands (by fitting models to less data).

Recently, Tresp [2000] introduced the Bayesian committee machine (BCM) approach, which is a principled way of combining estimators which were trained on different datasets. In BCM, the data are partitioned into M data sets $\mathcal{D} = D^1, \ldots, D^M$ (of typically approximately same size) by a clustering algorithm such as K-means [Bishop, 1995] and then the data of each cluster is used to train a separate estimator (e.g. a GP). When applied to a set of query points, each of the M GP systems outputs a prediction $E(f^q|D^i)$ together with covariance $cov(f^q|D^i)$, calculated employing equations (4.6) and (4.7).

The BCM combines the M estimates and calculates an approximation to the expected values $\hat{E}(f^q|\mathcal{D})$ of the functional values at the query points as

$$\hat{E}(f^q|\mathcal{D}) = C_{BCM}^{-1} \sum_{i=1}^{M} cov(f^q|D^i)^{-1} E(f^q|D^i) \tag{4.13}$$

$$C_{BCM} = \widehat{cov}(f^q|\mathcal{D})^{-1} = -(M-1)(\Sigma^{qq})^{-1} + \sum_{i=1}^{M} cov(f^q|D^i)^{-1}. \tag{4.14}$$

Thus the prediction of each GP system i is weighted by the inverse covariance of its prediction.

In Section 4.5 we introduce a guided BCM (GBCM) approach which uses segmentation of the input space obtained using probabilistic hierarchical visualisation technique described in the following section.

4.3 Probabilistic hierarchical visualisation models

Probabilistic hierarchical visualisation based on a mixture of latent variable models can provide a hierarchy whose top-level projection displays the entire dataset, perhaps revealing the presence of clusters, while lower-level projections display internal structure within individual clusters, such as the presence of subclusters, which might not be apparent in the higher-level projection [Bishop and Tipping, 1998]. This has been proved to be an effective way of visualising large multi-dimensional datasets.

Since, in such a visualisation hierarchy, the data is modelled with a probabilistic mixture of latent variable models, we obtain a soft partitioning of the dataset into "clusters", and at the same time obtain multiple visualisation plots corresponding to the clusters. The corresponding density model takes the form

$$p(\mathbf{x}) = \sum_{i=1}^{M} \pi_i p(\mathbf{x} \mid i), \tag{4.15}$$

where M is the number of components in the mixture, and the parameters π_i are the mixing coefficients, or prior probabilities, corresponding to the mixture components $p(\mathbf{x} \mid i)$. Each component is an independent latent variable model with required parameters. These model parameters can be set using a variant of the EM algorithm [Bishop and Tipping, 1998]. During the derivation of the EM algorithm, the missing data now also includes labels which specify which component is responsible for each data point. The prior expectations for the component labels are given by the π_i and corresponding posterior probabilities, or responsibilities, are calculated as

$$R_{in} = P(i|\mathbf{x}_n) = \frac{\pi_i p(\mathbf{x}_n \mid i)}{\sum_{i'} \pi_{i'} p(\mathbf{x}_n \mid i')}. \tag{4.16}$$

This mixture distribution forms the second level in the hierarchical mixture of latent variable model. The hierarchical structure, $\mathcal{T}$, can be extended to any desired number of levels, for any component distributions from the exponential family. [Tiňo and Nabney, 2002] have presented a full hierarchical derivation for a hierarchical mixture model using generative topographic mapping (GTM) [Bishop et al., 1998] as the latent model. Model responsibilities, $\mathbf{R}_i$, for models $\mathcal{M}_i$, $i = 1, \ldots, M$, in the hierarchical structure, $\mathcal{T}$, are calculated as follows:

$$R_{in} = P(\mathcal{M}_i \mid \mathit{Parent}(\mathcal{M}_i), \mathbf{x}_n), \tag{4.17}$$

$$= \frac{\pi(\mathcal{M}_i \mid \mathit{Parent}(\mathcal{M}_i)) P(\mathbf{x}_n \mid \mathcal{M}_i)}{\sum_{\mathcal{N} \in [\mathcal{M}_i]} \pi(\mathcal{N} \mid \mathit{Parent}(\mathcal{M}_i)) P(\mathbf{x}_n \mid \mathcal{N})}, \tag{4.18}$$

where $[\mathcal{M}_i] = \mathit{Children}(\mathit{Parent}(\mathcal{M}_i))$.

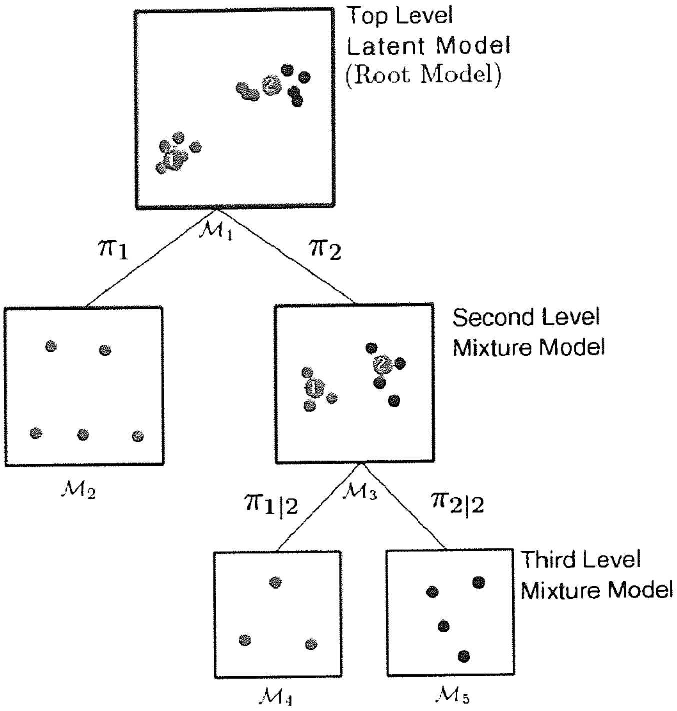


Figure 4.3: Example structure of a hierarchical model. The numbered circles indicate the submodel centers.

Imposing $P(Root \mid \mathbf{x}_n) = 1$, the unconditional (on parent) model responsibilities are recursively determined by the formula:

$$P(\mathcal{M}_i \mid \mathbf{x}_n) = P(\mathcal{M}_i \mid \mathcal{M}_p, \mathbf{x}_n) P(\mathcal{M}_p \mid \mathbf{x}_n). \tag{4.19}$$

where $\mathcal{M}_p = Parent(\mathcal{M}_i)$. The eq. (4.19) automatically satisfies the relation

$$\sum_{\mathcal{N} \in [\mathcal{M}_i]} R_{\mathcal{N}n} = R_{\mathcal{M}_p n} \tag{4.20}$$

where $[\mathcal{M}_i] = Children(Parent(\mathcal{M}_i))$ and $\mathcal{M}_p = Parent(\mathcal{M}_i)$. The eq. (4.20) implies that responsibility of each model at level L for a given data point n is shared by a partition of unity between the corresponding group of offspring models at level $L+1$.
Thus the model responsibility matrix, $\mathbf{R}$, has an important property

$$\sum_{\mathcal{N} \in [\mathcal{M}_l]} R_{\mathcal{N}n} = 1 \qquad \forall\, n. \tag{4.21}$$

where $[\mathcal{M}_l] = Leaves(\mathcal{T})$ and $\mathcal{T}$ is the hierarchy structure.

Eq. (4.21) confirms the *soft* segmentation of the input space we obtain from the hierarchical mixture of latent models. It corresponds to the segmentation derived from the softmax function in the trained gating network in the MoE.

The latent variable models in the hierarchy could be linear or non-linear. Experimental results presented in this chapter use a hierarchy of linear latent model based on probabilistic principal component analysis (PPCA) [Tipping and Bishop, 1999a] and a hierarchy of the non-linear latent model based on GTM [Tiňo and Nabney, 2002].

Figure 4.3 depicts an example of the structure of a hierarchical model. The hierarchy can be built interactively in a top-down fashion using a software tool, DVMSv1.8, supporting visual data mining framework [Maniyar and Nabney, 2006c] introduced in the previous chapter. This software tool supporting advanced information visualisation facilities to assist interactive exploration of the projections obtained at each level of the visualisation hierarchy is provided to the domain experts. The targets (e.g. class labels) can be used to guide the development of the visualisation hierarchy using colour, symbols or both. After studying a projection carefully, a domain expert selects "regions of interest" by selecting centers which becomes the centers of the submodels in the next level of the hierarchy. We refer this process as 'drilling-down'. For example, the numbered circles in the visualisation hierarchy plot (see Figure 4.3) indicates the selected submodel centers.

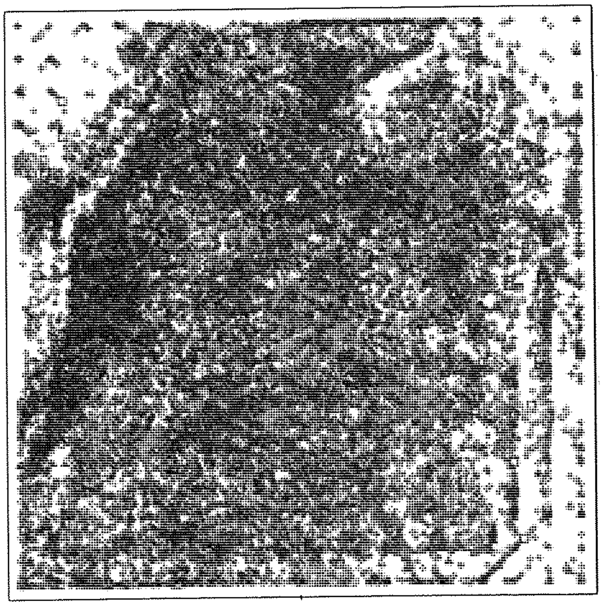


Figure 4.4: An example of strongly overlapping clusters: projection of a chemical compound dataset with 23000 compounds.

The construction of the visualisation hierarchy guided by the domain expert using an interactive software tool is a powerful method when the clusters are separated clearly in the two-dimensional latent space. On the other hand, when the dataset is very large, it is difficult for a domain expert to select locations for submodels at the next level as the higher-level projection plots may be cluttered and confusing due to densely clustered and overlapping projections, eg. Figure 4.4.

Recently a semi-automatic submodel initialisation algorithm, based on a minimum message length (MML) criterion, which decides both the number of submodels and their location, was introduced for Hierarchical GTM (HGTM) [Nabney et al., 2005]. At any stage in the hierarchy construction process, either the semi-automatic or manual method can be applied. We use this algorithm to obtain the higher-level projections of the visualisation hierarchy for a large dataset and then the domain expert can take control to guide the lower-level projections. The visualisation hierarchy semi-automatically developed in such a way benefits from the automatic submodel initialisation at the higher levels of the hierarchy and involvement of the domain expert at lower levels to better guide the hierarchy.

4.4 Guided mixture of local experts (GME)

The GME models are developed in a two stage process. First a probabilistic visualisation hierarchical model is developed as discussed in the previous section. Then the responsibility matrix, **R**, calculated using the trained visualisation hierarchy, is used to train a guided mixture of local experts as described in Procedure 1 below.

4.4.1 Training

Procedure 1 (Training). *1. Using a previously trained visualisation hierarchy, calculate the model responsibility matrix,* **R**, *for all the training points (Eq. 4.17).*

2. For each leaf node in the visualisation hierarchy, train a corresponding expert model. Train each local expert, $\phi_i(\mathbf{t} \mid \mathbf{x})$, individually on all the training points using the corresponding responsibility vector to weight the error function.

3. During the training of each expert, $\phi_i(\mathbf{t} \mid \mathbf{x})$, select the best architecture through cross-validation (or some other appropriate regularisation scheme).

Note that in the step 2 of Procedure 1, all the data points of the training set are used to train each local expert. The corresponding responsibilities are utilised for weighting the error function during the training of the local experts. For example, for regression problems, the sum-of-squares error function for expert i is weighted by the responsibility as below.

$$E_i = \frac{1}{2}\sum_{n=1}^{N} R_{in}(\phi_i(\mathbf{t}_n \mid \mathbf{x}_n) - \mathbf{t}_n)^2 \tag{4.22}$$

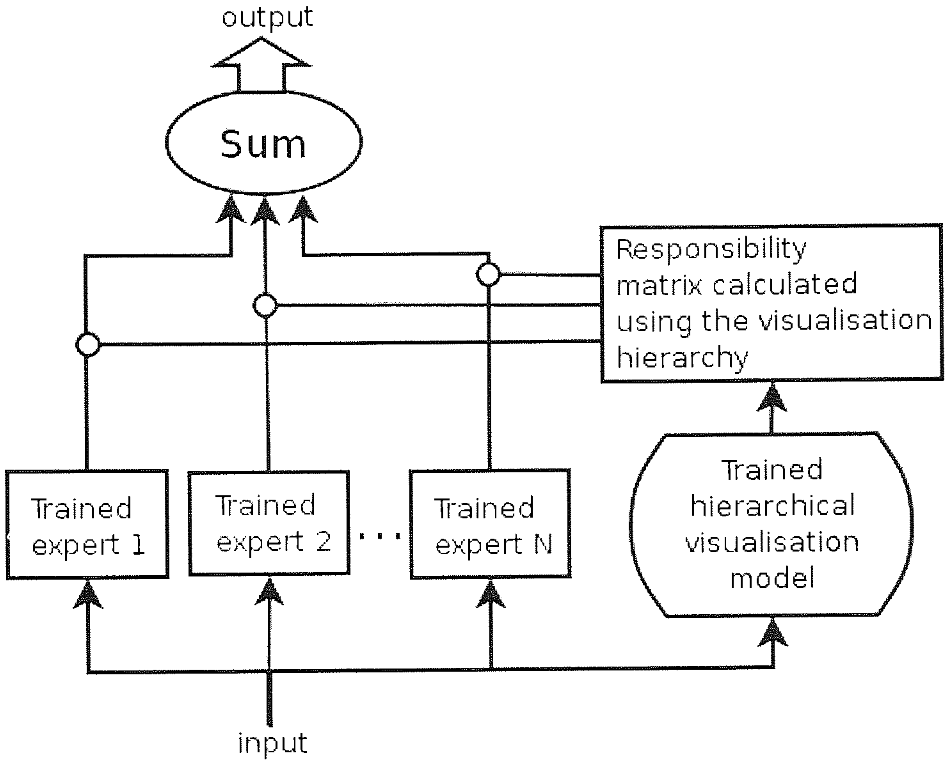

output
Sum
Responsibility matrix calculated using the visualisation hierarchy
Trained expert 1
Trained expert 2
...
Trained expert N
Trained hierarchical visualisation model
input

Figure 4.5: Prediction using guided mixture of local experts (GME).

where $\phi_i(\mathbf{t}_n \mid \mathbf{x}_n)$ is the output from the expert i and $\mathbf{t}_n$ is the target for the nth pattern.

Thus by weighting error function appropriately, we give more weight to those data points which belong to an input region related to a particular local expert. The individual experts can arbitrarily be a linear or a non-linear regression or classification models (eg. LR, LDA, MLP, GP, etc.).

4.4.2 Prediction

The prediction process is presented in procedure 2 and is depicted in Figure 4.5. For prediction, the inputs are first presented to the trained visualisation hierarchy and responsibilities for each expert are calculated using eq. (4.17). The responsibilities are used to weight the outputs of the local experts.

Procedure 2 (Prediction). *1. Calculate the model responsibility matrix,* $\mathbf{R}$, *for all the testing points using a trained visualisation hierarchy.*

2. *Each trained expert is presented with all the inputs (see Figure 4.5). All experts produce an output for all the input patterns.*

3. *These outputs are then weighted by the corresponding model responsibilities and summed*

to obtain the final output:

$$\mathbf{y}_n = \sum_{i=1}^{M} R_{in}\phi_i(\mathbf{t}_n \mid \mathbf{x}_n), \tag{4.23}$$

where $\phi_i(\mathbf{t}_n \mid \mathbf{x}_n)$ *is the output from the trained expert* i.

According to the task different activation functions are used. For example, for regression tasks, the weighted output of all experts, obtained from the step 2 of the testing procedure, is summed to obtain the final output of GME. For classification problems, the weighted posterior class probability obtained from all local expert is summed individually and the class with highest posterior probability wins.

4.4.3 Discussion

Another way of using the responsibility matrix to train a guided mixture of experts is to select only those data points which 'belong' to a particular local region to train the expert responsible for modelling that region. This is achieved by training each expert, $\phi_i(\mathbf{t} \mid \mathbf{x})$, in step 2 of the training procedure (Procedure 1) individually on only those training points, $\mathbf{x}_n$, for which $R_{i,n}$ is greater than a threshold. Different thresholds can be tried and validated. The remaining training procedure remains the same as Procedure 1. Similarly, the threshold is also applied during the prediction process.

Weighting the error functions with the model responsibilities is a probabilistic and more principled way of utilising the model responsibility than the responsibility threshold approach. The main drawback of the responsibility threshold approach is that using cross-validation to select the threshold is time consuming since for each validation iteration all the experts have to be trained and evaluated. However, to use the segmentation in a setting where a hard split is required, like in the next section, the threshold method can be used to decide which points belong to which segment.

4.5 Guided Bayesian committee machine (GBCM)

The soft segmentation obtained through the responsibility matrix can also be used with other established ways of combining local experts such as the BCM described in Section 4.2.2. As proposed by Tresp [2000], BCM is trained by first clustering the data using k-means and by then assigning the data of each cluster to a separate estimator. Then the estimators are combined to produce a consistent estimate of the output distribution. A better clustering could not only lead to a better BCM based model but is also useful in understanding the model. For many real-life large high-dimensional datasets, the clustering obtained using the k-means algorithm is not effective since the user must specify the number of clusters (i.e. k) which is very difficult *a priori* for many datasets and if the data does not naturally fall into separate clusters, the clustering results are poor.

Here instead of using k-means to separate the data, we use the threshold method discussed in Section 4.4.3 which utilises the model responsibility matrix obtained using a trained visualisation hierarchy. Then a separate predictor is trained for each leaf node in the hierarchy as in BCM. The predictions from the estimators are then combined as in BCM.

Summary

First, a trained visualisation hierarchy is used to calculate the model responsibility matrix, $\mathbf{R}$, using (eq. 4.17). Once the hierarchy is obtained, GBCM can be trained or used for prediction as described below:

- **Training:** Train an estimator corresponding to each leaf node in the hierarchy. Each estimator, $\phi_i(\mathbf{t} \mid \mathbf{x})$, (e.g. a GP) is trained individually on the training points, $\mathbf{x}_n$, for which $R_{i,n}$ is maximum amongst all the leaf nodes.

- **Prediction:** While combining estimators, the output of each estimator, $\phi_i(\mathbf{t} \mid \mathbf{x})$, is weighted by the inverse covariance of its prediction like in BCM (see Section 4.2.2). Thus estimators uncertain of their predictions are automatically weighted less than estimators which are certain about their prediction.

Though developing a GBCM model requires user interaction and thus more time than developing a BCM model which automatically splits the data using k-means, the prediction process is straightforward once we have a trained hierarchy.

4.6 Experiments

The development of the guided mixture of local experts model was motivated from problems in the chemoinformatics domain where there is a need for computational models that work with large heterogeneous datasets.

The formation of the mixture of experts model depends on the soft segmentation of the input space we obtain. Thus, it is important to evaluate the quality of the soft segmentation we obtained using different local models. To do so, we measure their entropy [Ellis, 1985].

The entropy is calculated as follows:

$$H = -\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{n=1}^{N} P_m(\mathbf{x}_n)\log P_m(\mathbf{x}_n), \tag{4.24}$$

where $P_m(\mathbf{x}_n)\log P_m(\mathbf{x}_n)$ is defined as 0 if $P_m(\mathbf{x}_n) = 0$. For MoE, $P_m(\mathbf{x}_n)$ is the output of the gating network for the mth expert, and input point x_n, while for all GME-based models, $P_m(\mathbf{x}_n)$ is the model responsibility, $R_{m,n}$. A smaller entropy corresponds to a sharper (and more interpretable) segmentation.

The NETLAB toolbox [Nabney, 2001] was used to develop LR/LDA, MLP, GP, and k-NN models. Experiments for MoE were carried out using the Mixlab toolbox [Moerland, 2000]. RTs with constant and linear regression models in the leaves were trained using the system RT4.1 [Torgo, 1999]. The BCMv1.0 toolbox [Schwaighofer, 2005] was used to create the BCM models.

The DVMSv1.8 software, developed using the visual data exploration framework discussed in the previous chapter, was used to develop expert-guided and semi-automatic visualisation hierarchies. Facilities such as magnification factors, directional curvatures and local parallel coordinates (described in Section 3.4) have proved helpful to the domain experts at Pfizer to guide informed visualisation hierarchies.

The MATLAB code for the GME algorithm is also available in the DVMSv1.8 software (`http://www.ncrg.aston.ac.uk/~maniyard/dvms/`). The GBCM code is based on the BCMv1.0 toolbox.

In this section we provide two case studies from the chemoinformatics domain; a classification problem and a regression problem.

4.6.1 Case study 1: The HTS dataset (classification)

As mentioned in chapter 2 it is useful to relate properties of compounds to their biological activity both to explore quickly a large chemical library for potency and to develop biological assays for HTS future campaign [Lipinski and Hopkins, 2004]. Our aim in this case study is to show the usefulness and suitability of local models over global models.

We used the HTS dataset (see Section 3.7.1) to compare the performance of different classifiers. We used the 11 whole-molecule physicochemical properties as input (see Section 3.7.1 for details). We aim to classify compounds as inactive or active for any of the 5 different biological targets.

As can be seen from the distribution presented in Table 3.2, the inactive compounds are dominant (~94% compounds were inactive for all five targets). A screening scientist

Model	Visualisation training mode	Latent model	True positive rate		Architecture	Entropy
			Training set	Test set		
LDA	-	-	17.53%	15.18%	-	-
MLP	-	-	18.02%	15.39%	$N\_hid = 12$	-
k-NN	-	-	10.02%	8.18%	$k = 6$	-
MoE	-	-	26.45%	21.37%	$N_{experts} = 11$	0.2195
GME	Expert-guided	GTM	**43.30%**	**39.10%**	$N_{experts} = 9$	0.0274
GME	Semi-automatic	GTM	36.76%	32.26%	$N_{experts} = 8$	0.0357
GME	Expert-guided	PPCA	30.54%	26.19%	$N_{experts} = 7$	0.0472

Table 4.1: Performance of different global and local models for the HTS dataset.

is interested in increased accuracy of prediction for active compounds, and thus the true positive rate for the classification of active compounds. True positive rate is very important, screening scientists do not mind false negatives.

50% of the dataset was used as the training set and the remainder was used as the test set.

Results

The architecture of the MLP (number of hidden nodes) and k-NN (optimum k) was decided using 10-fold cross validation.

We developed three different visualisation hierarchies using the training set; a completely expert-guided visualisation hierarchy with GTMs as the latent variable models, a semi-automatically (using MML, as described in Section 4.3) hierarchy with GTMs as the latent variable models, and an expert-guided hierarchy with PPCAs as the latent variable models. The hierarchical visualisation plots obtained on the test set using these three trained hierarchies are displayed in Figure 4.6, Figure 4.7 and Figure 4.8 respectively. Note that all data points are plotted on all the submodel projection plots in the visualisation hierarchy, with the density of "ink" in proportion to the corresponding responsibility which a submodel projection plot has for that particular data point. Thus, if one particular submodel plot in the hierarchy takes most of the responsibility for a particular data point, then that point will effectively be visible only on that corresponding submodel plot.

The true positive rate for the active compounds, model architecture and average entropy for different models are presented in Table 4.1. In all mixture of experts models, LDAs are used as the experts.

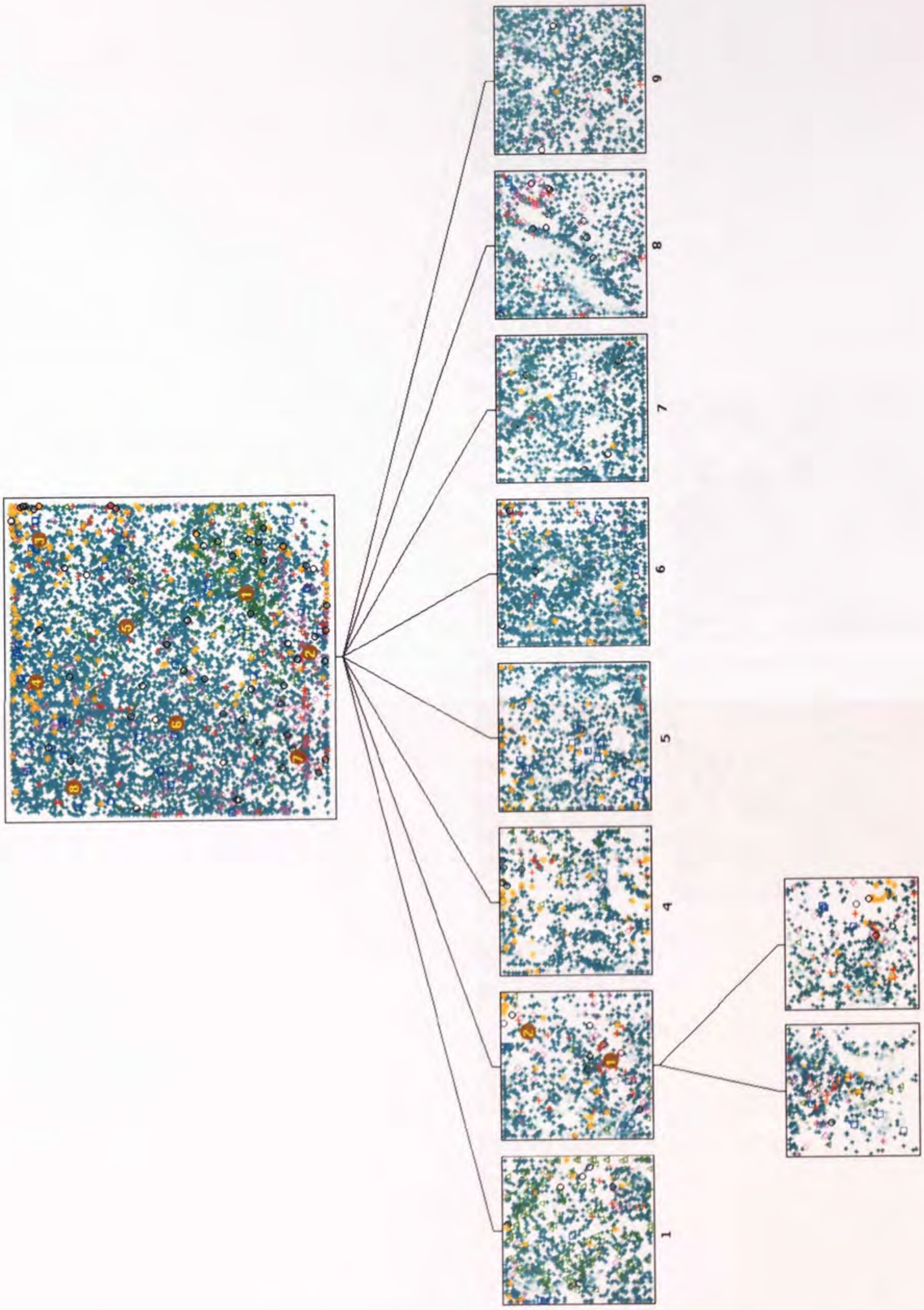


Figure 4.6: Expert-guided visualisation hierarchy with GTMs as the latent models for the HTS dataset. Refer to Table 3.2 for legend. The leaf nodes are numbered left to right for discussion.

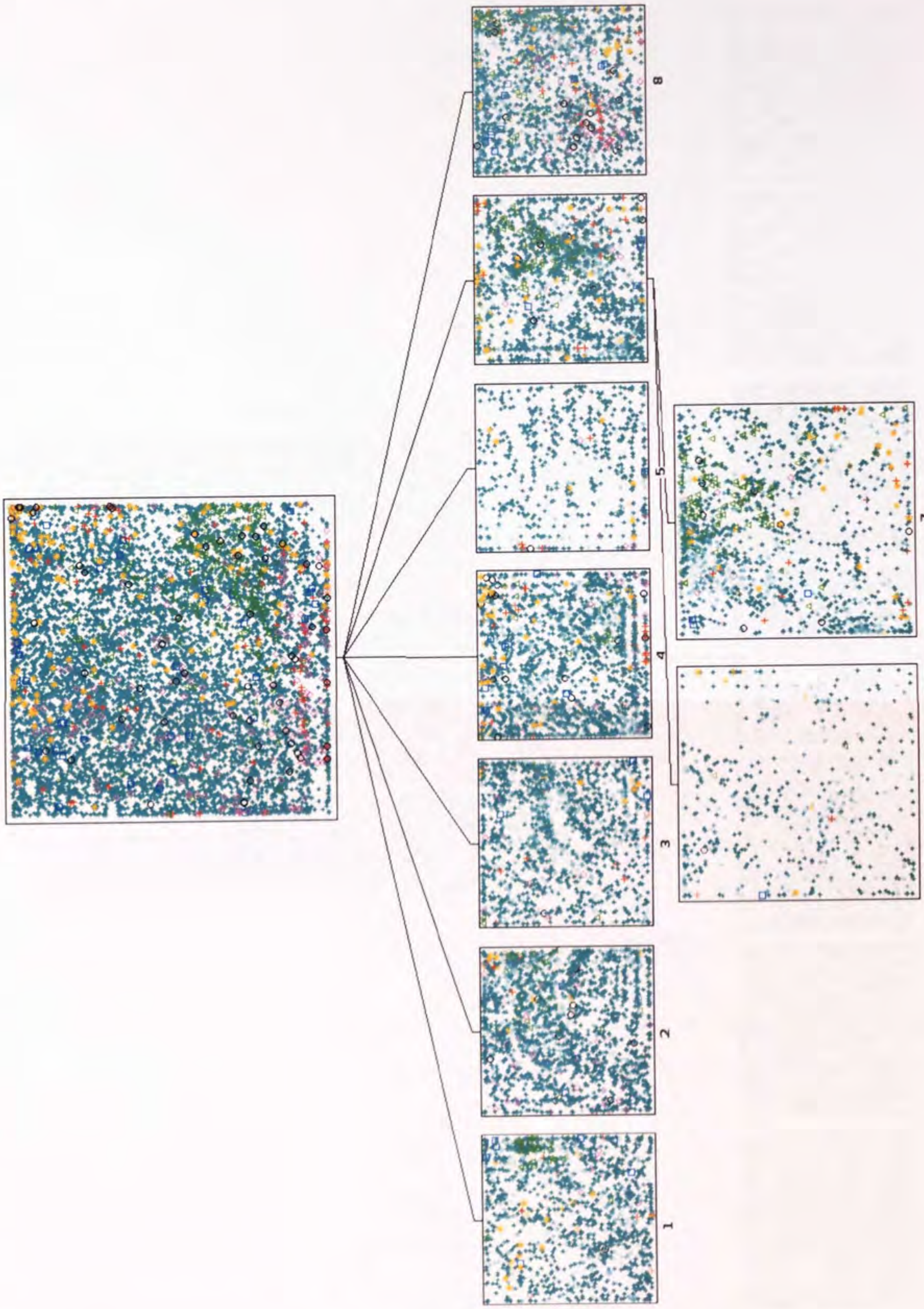


Figure 4.7: Automatically trained (using MML) visualisation hierarchy with GTMs as the latent models for the HTS dataset. Refer to Table 3.2 for legend. The leaf nodes are numbered left to right for discussion.

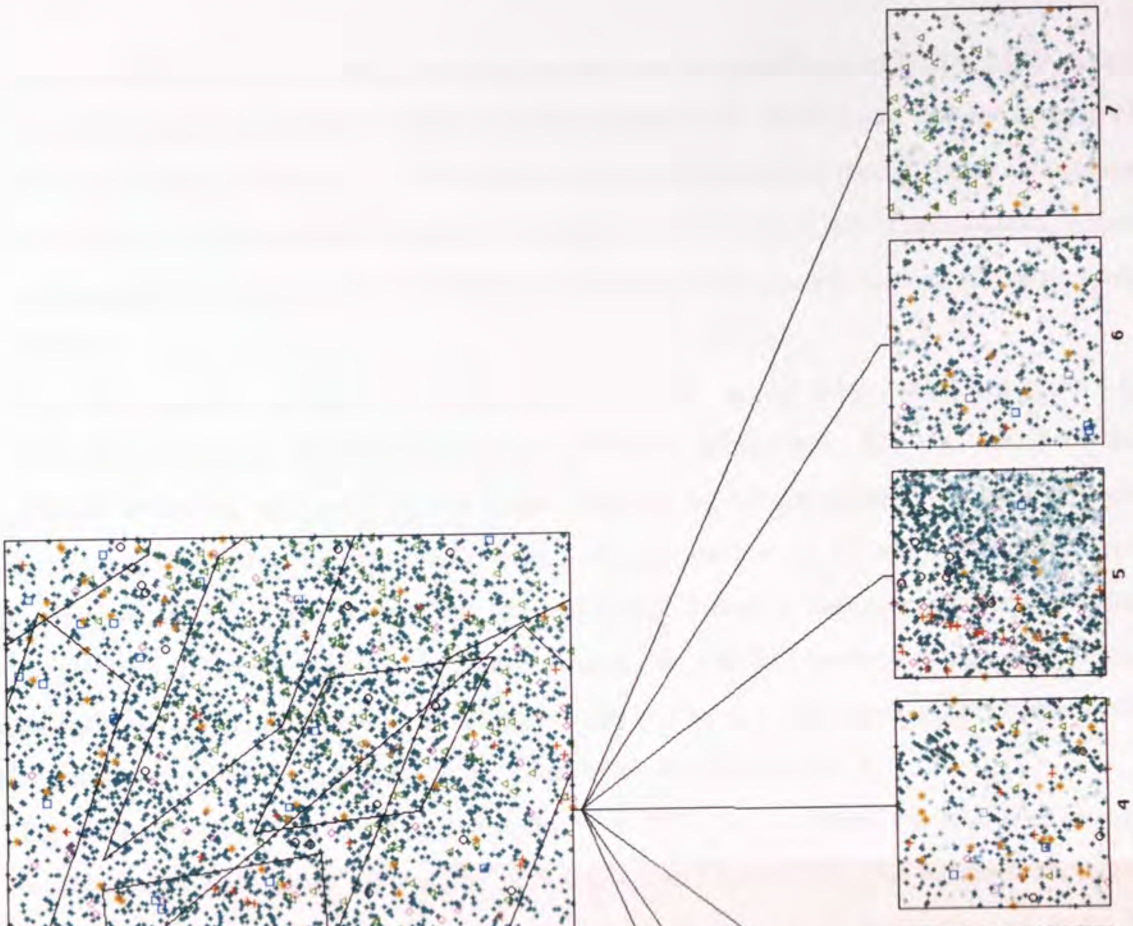


Figure 4.8: Expert-guided visualisation hierarchy with PPCAs as the latent models for the HTS dataset. Refer to Table 3.2 for legend. The leaf nodes are numbered left to right for discussion.

Discussion

The local models (MoE and GME) perform better than the global model (LDA, MLP). This supports our belief that a group of local models, each of which applied to a different region of the chemical space, is likely to be more effective than a single global model trying to capture the full behavioural variability of the chemical space. Since the dataset is imbalanced, which is true for most datasets in chemoinformatics, LDA and MLP classify almost all compounds as inactive.

The expert-guided visualisation hierarchy with GTM as the latent model, Figure 4.6, exhibits the dominance of different classes in different submodels. For e.g. most of the compounds active for aminergic type 2 target (marked as △) are grouped in the leaf node 1, the compounds active for peptidergic type 1 target (marked as +) and peptidergic type 2 target (marked as ∗) are grouped in the leaf nodes 3 and 4, the compounds active for aminergic type 1 target (marked as □ are grouped in the leaf node 5 with some of the compounds active for aminergic type 2 target (marked as △), and most of the compounds active for kinase target (marked as ◊) are populated in the leaf node 8.

The semi-automatic visualisation hierarchy with GTM as the latent model has relatively uniformly distributed classes among the submodels (see Figure 4.7). For example the compounds active for aminergic type 2 target (marked as △) are spread across the leaf nodes 1, 7 and 8.

The expert-guided visualisation hierarchy with PPCA as the latent model does not provide any visible clustering even in the root projection (see Figure 4.8). This is due to PPCA being a linear projection method that fails to project large high-dimensional complex datasets such as the HTS dataset.

It is useful to evaluate the entropy measure for different mixture-of-experts models. As we can observe from the values presented in Table 4.1, the soft segmentation obtained using the GME-based models have less disorder than the soft segmentation obtained using the gating network of the MoE model. Among the GME-based models, the expert-guided visualisation hierarchy with GTM as the latent model has the least disorder. This shows that guiding a visualisation hierarchy completely interactively gives a soft segmentation with less overlap between submodels.

Using a limited set of physicochemical properties to predict the activity of compounds does not give useable results as the activity of compounds depends on many other features such as topographical properties, etc. The data available to us is limited due to confidentiality issues. Models developed after careful selection of input descriptors are likely to improve the results further.

4.6.2 Case study 2: LogP prediction (regression)

Lipophilicity is the key physicochemical parameter linking membrane permeability - and hence drug absorption and distribution - with the route of clearance (metabolic or renal). The lipophilicity of a compound is readily amenable to automated measurement. The gold standard for expressing lipophilicity is the partition coefficient P (or LogP to have a more convenient scale) in an octanol/water system. There is continued interest in developing and improving LogP calculation programs, and there are many such programs available. Most calculation approaches rely on fragment values, although simple methods based on molecular size and hydrogen-bonding indicators for functional groups to calculate LogP values have also been shown to be extremely versatile.

The datasets

We studied a dataset used in [Tiño et al., 2004]. The aim is to develop a predictive model for the LogP using a novel molecular representation (an interaction fingerprint) developed at Pfizer. In this encoding, the molecule is represented by fourteen numerical variables based on a two-dimensional representation of the molecular structure. Variables 1-10 are the InterAction Fingerprints, IAFs [Lösel, 1998] of the compounds. IAFs are the average counts for noncovalent interactions (strong, medium, weak hydrogen bonds, van der Waals and pi-interactions) around individual atom types as found in experimental structures deposited in the Cambridge Structure Database summed up over the whole molecule. Value 11 is the sum of volumes of Voronoi polyhedra which were used to determine the IAFs and is used as a measure of size. Values 12-14 are halogen counts for fluorine, chlorine, and bromine.

Dataset I: This dataset has a set of 6912 compounds together with their LogP values that are freely available on the Internet. 20% of the set was used as a test set, and, when needed, another 10% was set apart as a validation set for model selection. The remaining data was used as a training set.

Dataset I+II: It consists of Data set I augmented by Data set II, a new set of 226 compounds whose LogP values were measured at Pfizer. Data set II served as a completely blind test set for models trained on the whole of Data set I. When needed, 10% of Data set I was set apart as a validation set for model selection.

Performance measures

As in [Tiño et al., 2004], we evaluated the test-set model performance via two related measures, namely mean square error and percent improvement over the naïve model.

Suppose we are given N_{trn} and N_{tst} training and test input-target pairs $(\mathbf{x}_n^{trn}, t_n^{trn})$ and $(\mathbf{x}_n^{tst}, t_n^{tst})$, respectively. Model prediction, given a test input $\mathbf{x}_n^{tst}$, is denoted by $\hat{t}_n$.

Mean squared error (MSE) measures the average squared difference between model predictions t_n and the corresponding targets t_n^{tst}

$$\mathrm{MSE} = \frac{1}{N_{tst}} \sum_{n=1}^{N_{tst}} (\hat{t}_n - t_n^{tst})^2. \tag{4.25}$$

The naïve predictor always predicts the unconditional mean of the training targets

$$\hat{t}_{\text{naïve}} = \frac{1}{N_{trn}} \sum_{n=1}^{N_{trn}} t_n^{trn} \tag{4.26}$$

$\mathrm{MSE}_{\text{naïve}}$ is the MSE of this predictor

$$\mathrm{MSE}_{\text{naïve}} = \frac{1}{N_{tst}} \sum_{n=1}^{N_{tst}} (\hat{t}_{\text{naïve}} - t_n^{tst})^2. \tag{4.27}$$

The degree of improvement (expressed in percentages) of the model over the Naive predictor is quantified by the improvement over Naive (ION) measure

$$\mathrm{ION} = \frac{\mathrm{MSE}_{\text{naïve}} - \mathrm{MSE}}{\mathrm{MSE}_{\text{naïve}}} 100\%. \tag{4.28}$$

ION is closely related to squared multiple correlation that uses variance in test targets instead of $\mathrm{MSE}_{\text{naïve}}$.Variance in test targets can be viewed as MSE of a simple predictor always predicting the mean of the test targets. Usually the variance over the training targets is close to that of the test targets, in which case squared multiple correlation can be interpreted as measuring ION.

Results

The experimental results for sets Dataset I and Dataset I+II are presented in Tables 4.2. The architecture of models were determined by the performance on the validation set: the architectures with the smallest validation set MSE were selected. A user guided visual hierarchy (Figure 4.9) was used to train GBCM and GME models. The hierarchy has GTMs as the latent models. Entropy values of the soft segmentation obtained using the trained hierarchy are 0.0063 and 0.0069 for Dataset I and Dataset I+II, respectively. Entropy of the

Model	Dataset I			Dataset I+II		
	MSE	ION	Architecture	MSE	ION	Architecture
Naïve	2.6902	0	-	4.0751	0	-
LR	0.7993	70.3	-	1.1904	70.8	-
MLP	0.6413	76.2	$N\_hid = 15$	1.0751	73.6	$N\_hid = 15$
GP	0.6012	77.7	$\sigma^2 = 40$	1.0739	73.6	$\sigma^2 = 50$
k-NN	0.9182	65.9	$k = 4$	1.0751	57.6	$k = 4$
RT	0.7372	72.6	LR, N=21, L=11	0.9791	76.0	LR, N=21, L=11
MoE	0.6611	75.4	$N\_experts = 9$	0.9610	76.4	$N\_experts = 9$
BCM	0.6101	77.3	$k = 8$	0.9621	76.3	$k = 8$
GBCM	0.6038	77.6	$N\_experts = 8$	0.9522	76.6	$N\_experts = 8$
GME	0.6481	75.9	$N\_experts = 8$	0.9557	76.5	$N\_experts = 8$

Table 4.2: Performance of different global and local models for the LogP dataset.

soft segmentation obtained using MoE algorithm are 0.1362 and 0.1398 for Dataset I and Dataset I+II, respectively. This demonstrates that soft segmentation obtained using user guided trained hierarchy is relatively ordered and has less overlap. For a fair comparison of GBCM with BCM, we kept k (*a priori* for k-means) as 8.

Using the local parallel coordinate (LPC) facility, we can observe diversity of patterns of the compounds in different regions in Figure 4.9. Magnification factors and directional curvatures plots were used to guide the development of this hierarchy.

Discussion

Our aim in this analysis was to apply the novel guided local expert models to compare their performance with other methods. Clearly guided models, GBCM and GME, provided comparable results. Though the two-stage process of guiding hierarchy first and then training the model only improves the accuracy slightly, it also enhances the interpretability of the models. Guiding a visualisation hierarchy without class labels during a regression problem can be dealt with using information visualisation tools such as local parallel coordinates and calculated manifold properties such as magnification factors and directional curvature plots.

4.7 Conclusions

The essence of our idea is to exploit hierarchical non-linear visualisation approaches and to allow user interaction to obtain a meaningful segmentation of the input space into regions with similar behaviour which is then used to train a mixture of local experts. The use of the user guided hierarchy means that expert beliefs is used to guide the segmentation of the

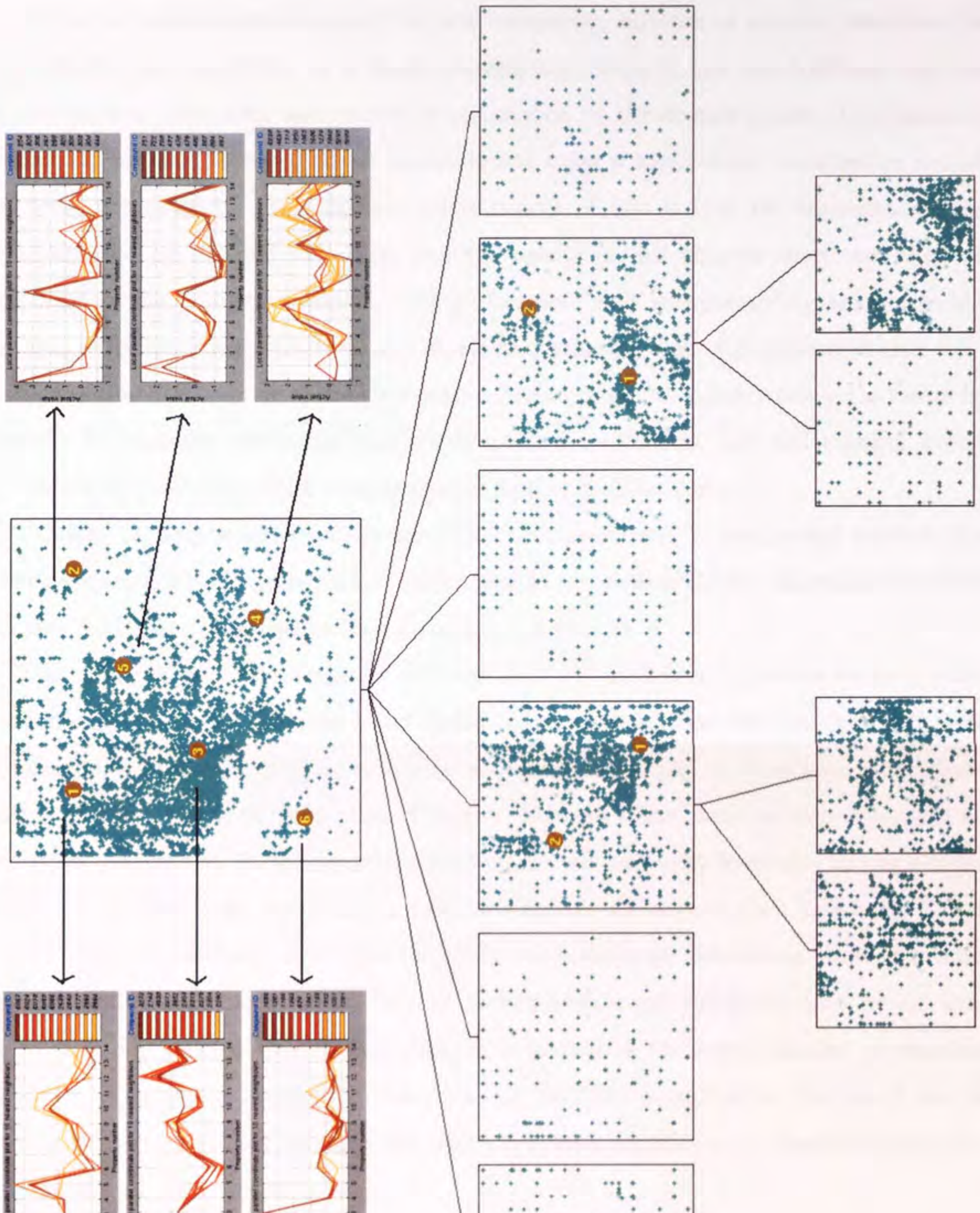


Figure 4.9: Expert-guided visualisation hierarchy with GTMs as the latent models for the LogP dataset. Example local parallel coordinate (LPC) plots shows variability of patterns in different regions of the root GTM projection.

input space, which could be very useful when dealing with large heterogeneous datasets.

While the conventional approach to local modelling, mixture of experts, combines the segmentation and prediction in a single training algorithm, it has the drawback that the segmentation is often poor and cannot be understood by the domain expert. Our approach divides the problem into two steps: segmentation using a hierarchical visualisation model, and local prediction on each segment. The benefit of this is that the segmentation can be interpreted by the domain expert and the results in this chapter show that this can also improve the prediction results. The performance and interpretability are worthwhile advantages for spending more time and efforts in developing a good prediction model.

Semi-automatic training of the visualisation hierarchy at the higher levels of the hierarchy is useful for datasets where the higher-level plots are cluttered, but still requires expert interaction to determine which models require further 'drilling down'.

Utilising the segmentation of the input space obtained using a user-guided hierarchy for other local experts approaches such as shown here in the form of Guided Bayesian committee machine (GBCM) has proven to be a successful experiment.

The main goal of the development of the guided local models is to provide better *in silico* prediction of molecular (biological and chemical) properties. One key aspect of this is to understand, model, and interpret structural aspects of molecules, as these have a significant impact on their biological properties. However, because Pfizer need to keep structural information confidential, in our research collaboration, we have only been able to use features computed by Pfizer from structure, and this has limited the accuracy we have been able to achieve with our methods. Although our predictive techniques outperform those in routine use and leading edge methods, the level of performance is not yet at the level we are aiming at, and our belief is that further research is needed on the representation of drug-like molecules. A proposed avenues of research in this direction is outlined in Chapter 6. In the next chapter we introduce an algorithm which estimates significance of features during the training of a data visualisation model.

Chapter 5

Data Visualisation with Simultaneous Feature Selection

Real-life datasets in chemoinformatics frequently have large number of descriptors. Data visualisation algorithms and feature selection techniques are both widely used in chemoinformatics but as distinct analytical approaches. Until recently, not much research has been done on estimating feature saliency while training a data visualisation model since feature selection for unsupervised learning is a challenging task. In this chapter, we derive a generative topographic mapping (GTM)-based data visualisation approach which estimates feature saliency simultaneously with the training of the visualisation model. The approach not only provides a better projection by modelling irrelevant features with a separate noise model but also gives feature saliency values which help the user to assess the significance of each feature. We compare the quality of projection obtained using the new approach with the projections from traditional GTM and self-organizing maps (SOM) algorithms. The results obtained on a synthetic and real-life chemoinformatics datasets demonstrate that the proposed approach successfully identifies feature significance and provides coherent (compact) projections.

5.1 Introduction

As discussed in Chapter 3, data visualisation is an important part of the visual data exploration framework. In many real-life problems in chemoinformatics and bioinformatics we are required to work with datasets with large number of descriptors [Baldi and Hatfield, 2002; Liu, 2004]. Many of these descriptors (features) may not be relevant to obtaining an effective projection. In principle, the more information we have about each pattern, the better a visualisation algorithm is expected to perform. This seems to suggest that we should use as many features as possible to represent the patterns. However, this is not the case in practice. Some features can be just "noise". For a large multivariate dataset, feature selection is important for several reasons, the fundamental one being that noisy features can degrade the performance of most learning algorithms.

Feature selection has been widely studied in the context of supervised learning and applied to many supervised learning problems in pharmaceutical research [Blum and Langley, 1997; Xing et al., 2001; Li et al., 2004]. Feature selection algorithms for supervised learning problems can be broadly divided into two categories *filters* and *wrappers*. Filter approaches evaluate the relevance of each feature (subset) using the data set alone, regardless of the subsequent learning algorithm [Blum and Langley, 1997]. On the other hand, wrapper approaches [Kohavi and John, 1997] invoke the learning algorithm to evaluate the quality of each feature.

Feature selection for unsupervised problems is more difficult and has received comparatively little attention because, unlike in supervised learning, there are no class labels for the data and, thus, no obvious criteria to guide the search [Mitra et al., 2002; Dy and Brodley, 2004]. Recently Law et al. [2004] proposed a solution to the feature selection problem in unsupervised learning using Gaussian mixture models (GMM) by casting it as an estimation problem, thus avoiding any combinatorial search. Instead of selecting a subset of features, they estimate a set of real-valued (in $[0, 1]$) variables (one for each feature) which are called the *feature saliencies*. They adopted a minimum message length (MML) [Wallace and Dowe, 1999] penalty for model selection. This approach can be classified as of wrapper type.

As described in Section 3.3.4, GTM is a principled probabilistic mixture-based data visualisation algorithm where each data point is modelled as having been generated by one of a set of probabilistic models. Since GTM is a mixture-based projection method, it is possible to adopt the feature selection approach proposed for GMM in [Law et al., 2004] to the GTM. For clustering using data visualisation, the GTM provides many advantages over GMM because of their ability to intuitively visualise data on a low dimensional representation space (projection) and probabilistic interpretation of the projection manifold (magnification factors and directional curvature plots).

In this chapter, we introduce a GTM-based data visualisation with simultaneous feature selection (GTM-FS) approach which not only provides a better visualisation by modelling irrelevant features ("noise") using a separate shared distribution but also gives a saliency value for each feature which helps the user to assess their significance. Such notion of feature saliency is more appropriate than a "hard" feature selection (a feature is either selected or not) for many real-life datasets [Modha and Spangler, 2003].

The remainder of this chapter is organised as follows: the proposed approach, GTM with feature saliency (GTM-FS) determination, is introduced and mathematically derived in Section 5.2. The experimental results on both synthetic and real-life chemoinformatics datasets are reported in Section 5.3. In Section 5.4, we discuss computational costs for the projection algorithms. A similar research, which was recently brought to our attention, is discussed in Section 5.5. Finally, we draw the main conclusions in Section 5.6.

5.2 GTM with feature saliency (GTM-FS) determination

As discussed in Section 3.3.4, the generative topographic mapping (GTM) is a probability density model which describes the distribution of data in a space of several dimensions in terms of a smaller number of latent (or hidden) variables. The map $f : \mathcal{H} \Rightarrow \mathcal{D}$ between the latent space, $\mathcal{H}$, and the data space, $\mathcal{D}$, is non-linear, which implies that the image of the (flat) latent space is a curved and stretched manifold in the data space. It uses a mixture of Gaussians as a latent grid to model the data in the data space. Given a point $\mathbf{z}_m \in \mathcal{H}$ in the latent space, its image under the map f is

$$f(\mathbf{z}_m, \mathbf{W}) = \mathbf{\Phi}(\mathbf{z}_m)\mathbf{W}, \tag{5.1}$$

where $\mathbf{\Phi}(\mathbf{z}_m) = (\phi_1(\mathbf{z}_m), ..., \phi_K(\mathbf{z}_m))^T$ is a set of fixed non-linear basis functions, $\mathbf{W}$ is a $K \times D$ matrix of weight parameters and $f(\mathbf{z}_m, \mathbf{W})$ forms the centre of the Gaussian component, m, in the data space.

In GTM, the Gaussians are chosen to have spherical covariance as this corresponds to uniform noise in data. This may not be appropriate for many real-life datasets where we may have irrelevant features. To model this noise effectively and to calculate feature saliency, we assume that the features are conditionally independent given the mixture component label. In the particular case of Gaussian mixtures, the conditional independence assumption is equivalent to adopting diagonal covariance matrices. So instead of having a mixture of spherical Gaussians, as in GTM, we use a mixture of diagonal Gaussians with a common variance for each feature. Then the probability density function presented in (3.7) changes to,

$$p(\mathbf{t}_n|\boldsymbol{\theta}) = \sum_{m=1}^{M} \alpha_m \prod_{d=1}^{D} p(t_{nd}|\theta_{md}), \tag{5.2}$$

where M is the total number of components in the mixture (equal to the number of grid points in latent space), and as in GTM, we take the mixing coefficient, α_m, to be constant and equal to $\frac{1}{M}$. D is the total number of features in the input space, $\mathbf{t}_n$ is a D-dimensional vector representing the input point n, and $p(\cdot\,|\theta_{md})$ is the pdf of the dth feature for the mth component, with parameters $\boldsymbol{\theta}_{md} = \{\Phi_m \mathbf{w}_d, {\sigma_d}^2\}$. The variance ${\sigma_d}^2$ is common across all the components for each feature d. Thus $p(t_{nd}|\theta_{md})$ is a one dimensional Gaussian with the form

$$p(t_{nd}|\theta_{md}) = \frac{1}{\sqrt{2\pi\sigma_d^2}} \exp\left\{ -\frac{(t_{nd} - \Phi_m \mathbf{w}_d)^2}{2\sigma_d^2} \right\}. \tag{5.3}$$

Before we introduce the notion of feature saliency in the Gaussian mixture model described in (5.2), it is important to note that this mixture model structure is a constrained GMM where all components of the mixture share the same variance in each direction d and the mixing coefficients are all fixed to $\frac{1}{M}$. In contrast, Law et al. [2004] used unconstrained GMM with model selection criteria to decide clustering according to the final structure of the mixture. Our aim is effective data visualisation with simultaneous feature saliency estimation where we treat the mixture as the projection manifold. Like in GTM, the constraints imposed here will favour a two-dimensional structured manifold to represent the data and desirable uniform distribution of data on resulting projection.

The dth feature is irrelevant if its distribution is independent of the component labels, i.e., if it follows a common 'background' density, denoted by $q(t_{nd}|\lambda_d)$ which is taken to be a diagonal Gaussian, with λ_d as the set of parameters. Let $\Psi = (\psi_1, ..., \psi_D)$ be an ordered set of binary parameters, such that $\psi_d = 1$ if feature d is relevant and $\psi_d = 0$, otherwise. Now the probability density is

$$p(\mathbf{t}_n|\boldsymbol{\Delta}) = \frac{1}{M} \sum_{m=1}^{M} \prod_{d=1}^{D} [p(t_{nd}|\theta_{md})]^{\psi_d} [q(t_{nd}|\lambda_d)]^{(1-\psi_d)}, \tag{5.4}$$

where $\boldsymbol{\Delta} = \{\{\theta_{md}\}, \{\lambda_d\}, \psi_d\}$.

The notion of feature saliency is modelled as follows: 1) The ψ_ds are treated as "missing variables" in the expectation maximization (EM) algorithm [Dempster et al., 1977] sense and 2) the feature saliency is defined as $\rho_d = P(\psi_d = 1)$, the probability that the dth feature is relevant.

Now the resulting model can be written as

$$p(\mathbf{t}_n|\Theta) = \frac{1}{M}\sum_{m=1}^{M}\prod_{d=1}^{D}(\rho_d p(t_{nd}|\theta_{md}) + (1-\rho_d)q(t_{nd}|\lambda_d)), \tag{5.5}$$

where $\Theta = \{\{\theta_{md}\},\{\lambda_d\},\{\rho_d\}\}$ is the set of all the parameters of the model. An intuitive way to see how (5.5) is obtained is to notice that $[p(t_{nd}|\theta_{md})]^{\psi_d}[q(t_{nd}|\lambda_d)]^{(1-\psi_d)}$ can be written as $\psi_d p(t_{nd}|\theta_{md}) + (1-\psi_d)q(t_{nd}|\lambda_d)$, because ψ_d is binary.

Figure 5.1 illustrates the notion of data visualisation with simultaneous feature selection in a GTM-FS model for three-dimensional data with feature 1 (d_1) and feature 2 (d_2) as salient features and feature 3 (d_3) as an irrelevant feature ("noise"). Then the fitting of a GTM with four components (given by (5.2), represented as a two dimensional manifold, shown as 'Latent Space' in Figure 5.1) can be illustrated schematically as four oblate spheroids (flat disks) on the manifold having larger width (variance) in the directions of features d_1 and d_2 and near-zero width in the direction of the d_3 in the data space. The separate shared pdf, $q(\cdot|\lambda)$, which models the irrelevant features, d_3, is displayed as a prolate spheroid in the middle of the manifold in the data space.

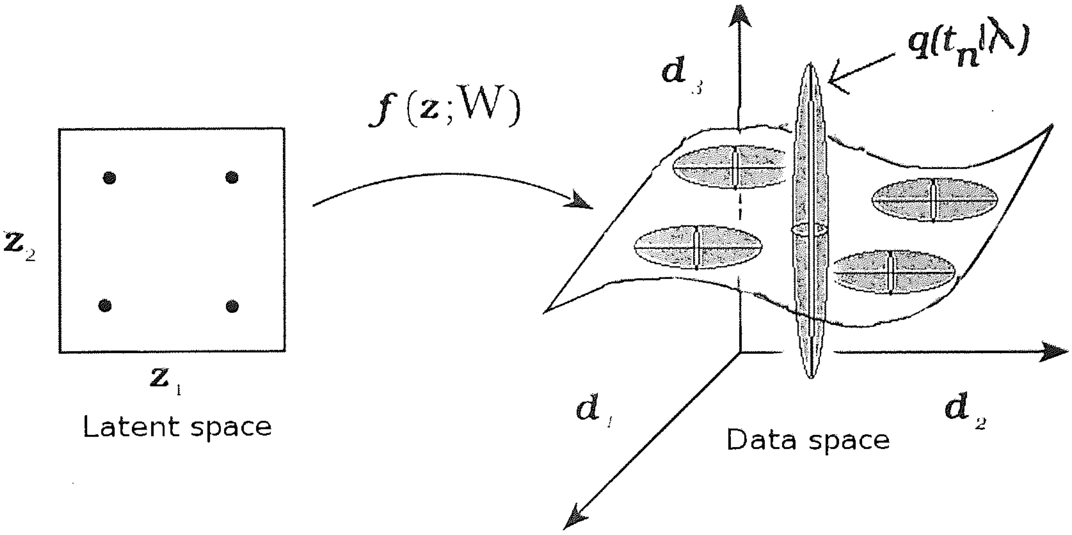


Figure 5.1: Schematic representation of the GTM-FS model. Features d_1 and d_2 have high saliency and d_3 has low saliency.

We exploit the latent-variable structure of the model in the same way as for standard GTM and use an EM algorithm to estimate the parameters in the model as described in the next section. The learning algorithm was modified from the one in [Law et al., 2004] to account for the constraints on component means imposed by the nonlinear mapping, $f(\mathbf{z}_m, \mathbf{W})$, and the common σ_d. Moreover we do not prune components during the training as in [Law et al., 2004] as the component grid in the latent space is the manifold and, like in GTM, our aim is to fit the manifold uniformly in the data space for data visualisation purposes.

5.2.1 An EM algorithm for GTM-FS

For each feature $d = \{1, \ldots, D\}$, we flip a biased coin whose probability of a head is ρ_d; if we get a head, we use the mixture component $p(\cdot\,|\theta_{md})$ to generate the dth feature; otherwise, the common density $q(\cdot\,|\lambda_d)$ is used.

We treat $\mathbf{y}$ (the hidden class labels) and the ψ_ds as the missing variables. In the E-step we use the current parameter set, Θ^{now}, to evaluate the posterior probabilities (responsibilities), $R_{nm} = P(y_n = m|\mathbf{t}_n)$, of each Gaussian component m for every data point $\mathbf{t}_n$ using Bayes' theorem in the form

$$R_{nm} = P(y_n = m|\mathbf{t}_n) \qquad = \frac{\prod_{d=1}^{D}(\rho_d p(t_{nd}|\theta_{md}) + (1-\rho_d)q(t_{nd}|\lambda_d))}{\sum_{m=1}^{M}\prod_{d=1}^{D}(\rho_d p(t_{nd}|\theta_{md}) + (1-\rho_d)q(t_{nd}|\lambda_d))}. \tag{5.6}$$

Using the responsibilities matrix $\mathbf{R}$, we can calculate $u_{nmd} = P(\psi_d = 1, y_n = m|\mathbf{t}_n)$, which measures how important the nth pattern is to the mth component, when the dth feature is used, and $v_{nmd} = P(\psi_d = 0, y_n = m|\mathbf{t}_n)$ as follows

$$u_{nmd} = P(\psi_d = 1, y_n = m|\mathbf{t}_n) \qquad = \frac{\rho_d p(t_{nd}|\theta_{md})}{\rho_d p(t_{nd}|\theta_{md}) + (1-\rho_d)q(t_{nd}|\lambda_d)} R_{nm}, \tag{5.7}$$

$$v_{nmd} = P(\psi_d = 0, y_n = m|\mathbf{t}_n) \qquad = R_{nm} - u_{nmd}. \tag{5.8}$$

In the M-step we use the posterior probabilities to re-estimate the weight matrix $\mathbf{W}$ by solving the following system of linear equations for each feature (see Appendix A for a detailed derivation)

$$\mathbf{\Phi}^T \mathbf{G}_d \mathbf{\Phi} \hat{\mathbf{w}}_d = \mathbf{\Phi}^T \mathbf{U}_d \mathbf{t}_d, \tag{5.9}$$

where $\mathbf{\Phi}$ is a $M \times K$ matrix, $\hat{\mathbf{w}}_d$ is a $K \times 1$ weight vector (the dth column of $\mathbf{W}$), $\mathbf{U}_d$ is a $M \times N$ matrix calculated using (5.7), $\mathbf{t}_d$ is a $N \times 1$ data vector, and $\mathbf{G}_d$ is an $M \times M$ diagonal matrix with elements

$$g_{mmd} = \sum_{n}^{N} u_{nmd}. \tag{5.10}$$

Then using this re-estimated $\hat{\mathbf{W}}$, it is straight forward to obtain the centres of the mixture components in data space, using (5.1), as follows:

$$\widehat{\text{Mean }\theta_m} = \mu_m = \mathbf{\Phi}(\mathbf{z}_m)\hat{\mathbf{W}}, \tag{5.11}$$

where μ_m is a $1 \times D$ vector.

Using the updated centre locations of the components of the mixture in the data space, the width of the diagonal Gaussians in each direction, corresponding to one feature each, is re-estimated by

$$\sigma_d = \frac{\sum_m \sum_n u_{nmd}(t_{nd} - \mu_{md})^2}{\sum_m \sum_n u_{nmd}}. \tag{5.12}$$

Recall that the width is common to all the components in the mixture.

The parameters of the common density, λ_d, are updated as follows:

$$\widehat{\text{Mean } \lambda_d} = \frac{\sum_n (\sum_m v_{nmd}) t_{nd}}{\sum_{nm} v_{nmd}}, \tag{5.13}$$

$$\widehat{\text{Var } \lambda_d} = \frac{\sum_n (\sum_m v_{nmd})(t_{nd} - \widehat{\text{Mean } \lambda_d})}{\sum_{nm} v_{nmd}}. \tag{5.14}$$

It is natural that the estimates of the mean and the variance in, λ_d, are weighted sums with weight v_{nmd}.

The feature saliency variable, ρ_d, is updated as follows:

$$\hat{\rho}_d = \frac{\max(\sum_{nm} u_{nmd} - \frac{ML}{2}, \epsilon)}{\max(\sum_{nm} u_{nmd} - \frac{ML}{2}, \epsilon) + \max(\sum_{nm} v_{nmd} - \frac{S}{2}, \epsilon)}, \tag{5.15}$$

where L and S are the number of parameters in θ_{md} and λ_d, respectively. ϵ is the smallest positive number that the machine can represent. We use it to make sure ρ_d does not become zero. So, unlike in [Law et al., 2004], we do not prune $p(\cdot|\theta_{md})$ since GTM-FS has a constrained mixture model.

The term $\sum_{nm} u_{nmd}$ in (5.15) can be interpreted as how likely it is that ψ_d equals one, explaining why the estimate of ρ_d is proportional to $\sum_{nm} u_{nmd}$.

A summary of the GTM-FS algorithm is presented in Algorithm 1.

5.3 Experiments

We tested GTM-FS on a synthetic dataset and the HTS dataset. Projection results using GTM-FS are compared with the results from traditional GTM and SOM algorithms and also evaluated using the evaluation methods described in Section 3.6. The experiments were carried out 5 times with different random seeds in the training algorithm to calculate standard deviations for the estimated feature saliency values. Label information was used for better presentation of the distribution of data points from different classes in the projections. Label information was also used to calculate KL-divergence and NN classification error.

Algorithm 1: Summary of the GTM-FS algorithm

```
Input: Training dataset.
Output: Trained GTM-FS visualisation model with estimated feature saliency values
        for all the features.
begin
    Generate the grid of latent points $\{\mathbf{z}_m\} \in \mathcal{H}, m = 1, 2, \ldots, M$;
    Generate the grid of basis functions, $\Phi(\mathrm{z_m})$, centres $\{\nu_k\}, k = 1, \ldots, K$;
    Select the basis functions, $\Phi(\mathrm{z_m})$, width;
    Compute the matrix of basis function activations, $\mathbf{\Phi}$ (like in GTM [Bishop, 1995]);
    Initialise $\mathbf{W}$, randomly or using PCA;
    Initialise width of the diagonal Gaussians in the grid (mixture);
    Initialise feature weight, $\rho_d$, for each feature $d$, to 0.5;
    Initialise the mixing coefficient, $\alpha_m$, for each component, $m$, in the grid to $1/M$;
    Set the mean and the variance of the shared distribution, $q(\cdot|\lambda)$, as the mean and
    covariance of the training set;
    repeat
        Compute $\mathbf{R}$, $\mathbf{U}$ and $\mathbf{V}$ using (5.6), (5.7) and (5.8) respectively, using current
        parameters, $\Theta^{now}$;
        for $d \leftarrow 1$ to $D$ do
            Reestimate the weight vector, $\mathrm{w}_d$, using $\hat{\mathrm{w}}_d = (\mathbf{\Phi}^T\mathbf{G}_d\mathbf{\Phi})^{-1}\mathbf{\Phi}^T\mathbf{U}_d\mathbf{t}_d$, derived
            from (5.9);
        end
        Obtain the centre, $\mu_m$, of each component, $m$, of the mixture in the data space,
        using (5.11);
        Reestimate the width of the diagonal Gaussians, $\sigma_d$, using (5.12), for all the
        features;
        Reestimate the mean and the variance of the shared distribution using (5.13)
        and (5.14) respectively;
        Reestimate the feature weight, $\rho_d$, using (5.15), for all the features;
    until convergence;
end
```

5.3.1 Case study 1: A synthetic dataset

The synthetic dataset consists of 800 data points from a mixture of four equiprobable Gaussians $\mathcal{N}(\mathbf{m}_i, \mathbf{I}), i = 1, 2, 3, 4$, where $\mathbf{m}_1 = \left(\begin{smallmatrix}0\\3\end{smallmatrix}\right), \mathbf{m}_2 = \left(\begin{smallmatrix}1\\9\end{smallmatrix}\right), \mathbf{m}_3 = \left(\begin{smallmatrix}6\\4\end{smallmatrix}\right), \mathbf{m}_4 = \left(\begin{smallmatrix}7\\10\end{smallmatrix}\right)$. Eight independent "noisy" features (sampled from a $\mathcal{N}(0, 1)$ density) are then appended to this data, yielding a set of 800 10-dimensional patterns.

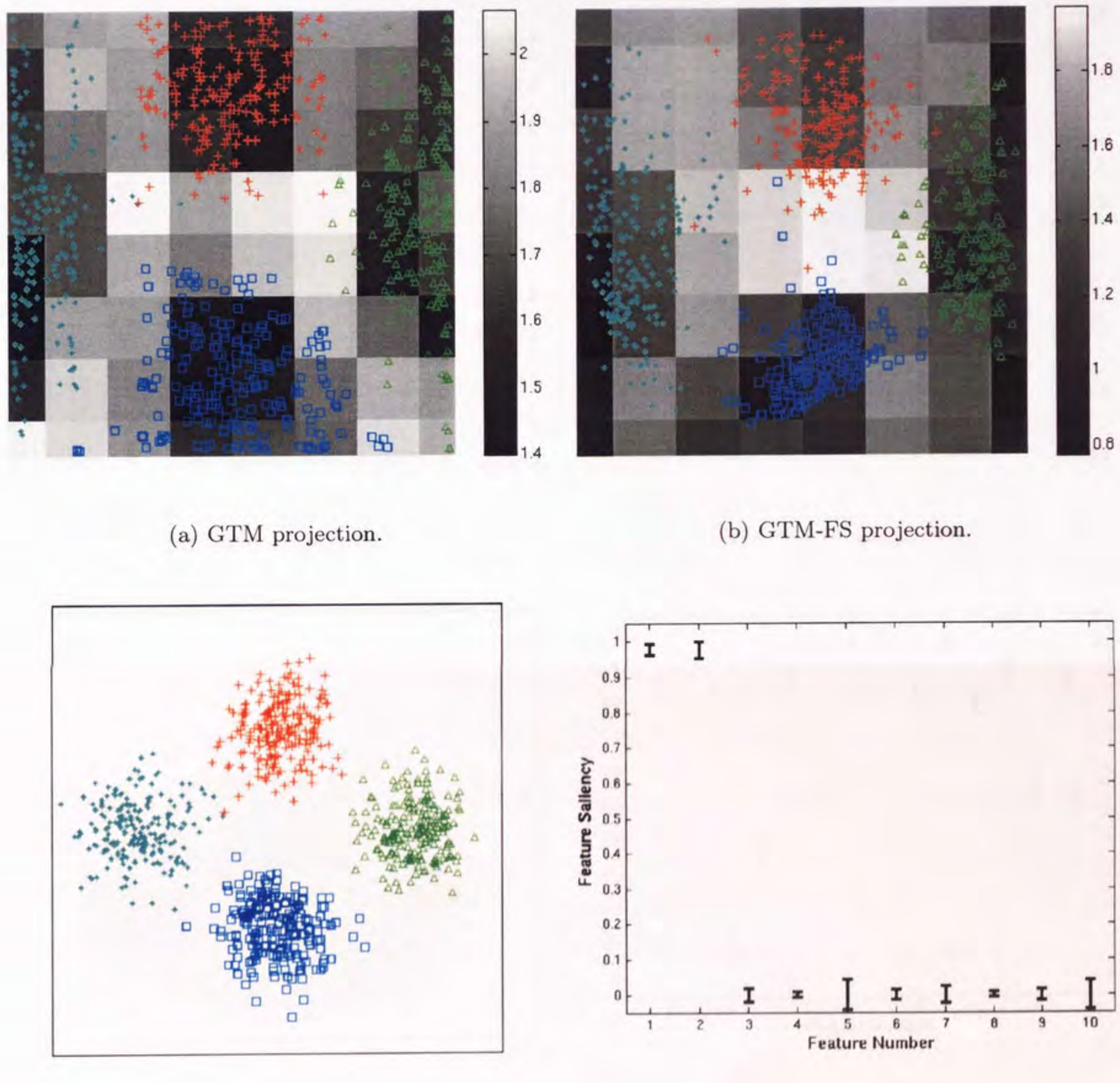
The projections obtained using GTM, GTM-FS and SOM algorithms are presented in Figure 5.2. Background colour shading in Figure 5.2(a) and Figure 5.2(b) displays the corresponding magnification factors for those projection manifolds. A comparative evaluation of these projections is presented in Table 5.1. The estimated saliencies of all the 10 features, together with standard deviations (error bars), are shown in Figure 5.2(d).

Method	Dataset	GTM	GTM-FS	SOM
MF sum	Synthetic	111.63	**82.32**	-
	The HTS dataset	125.92	**71.18**	-
KL divergence	Synthetic	15.31	**19.43**	12.34
	The HTS dataset	128.17	**167.56**	56.37
NN error (%)	Synthetic	0.75	0.75	**0.62**
	The HTS dataset	**38.32**	41.24	92.40

Table 5.1: Evaluation of visualisation models.

Discussion

As expected, all three projection algorithms gave four well separated cluster for the synthetic dataset. GTM-based algorithms create a uniform distribution in latent space so they spread the data more than SOM projection. This is also revealed from their higher KL-divergence sum value and NN error rate compared to SOM. The MF-sum of the GTM-FS manifold is smaller than the MF-sum of the GTM manifold which indicates that the GTM-FS manifold is comparatively less stretched. Close observation of Figure 5.2(a) and Figure 5.2(b) also reveals that the GTM-FS manifold is more coherent (compact). This is because in GTM-FS the irrelevant features ("noise") are modeled using the separate shared distribution, $q(\cdot|\lambda)$, and thus the actual manifold is less stretched. From the estimated feature saliency values using the GTM-FS model (Figure 5.2(d)) we can conclude that, in this case, the GTM-FS algorithm not only provided a good projection but also correctly estimated the feature saliencies.



(a) GTM projection.

(b) GTM-FS projection.

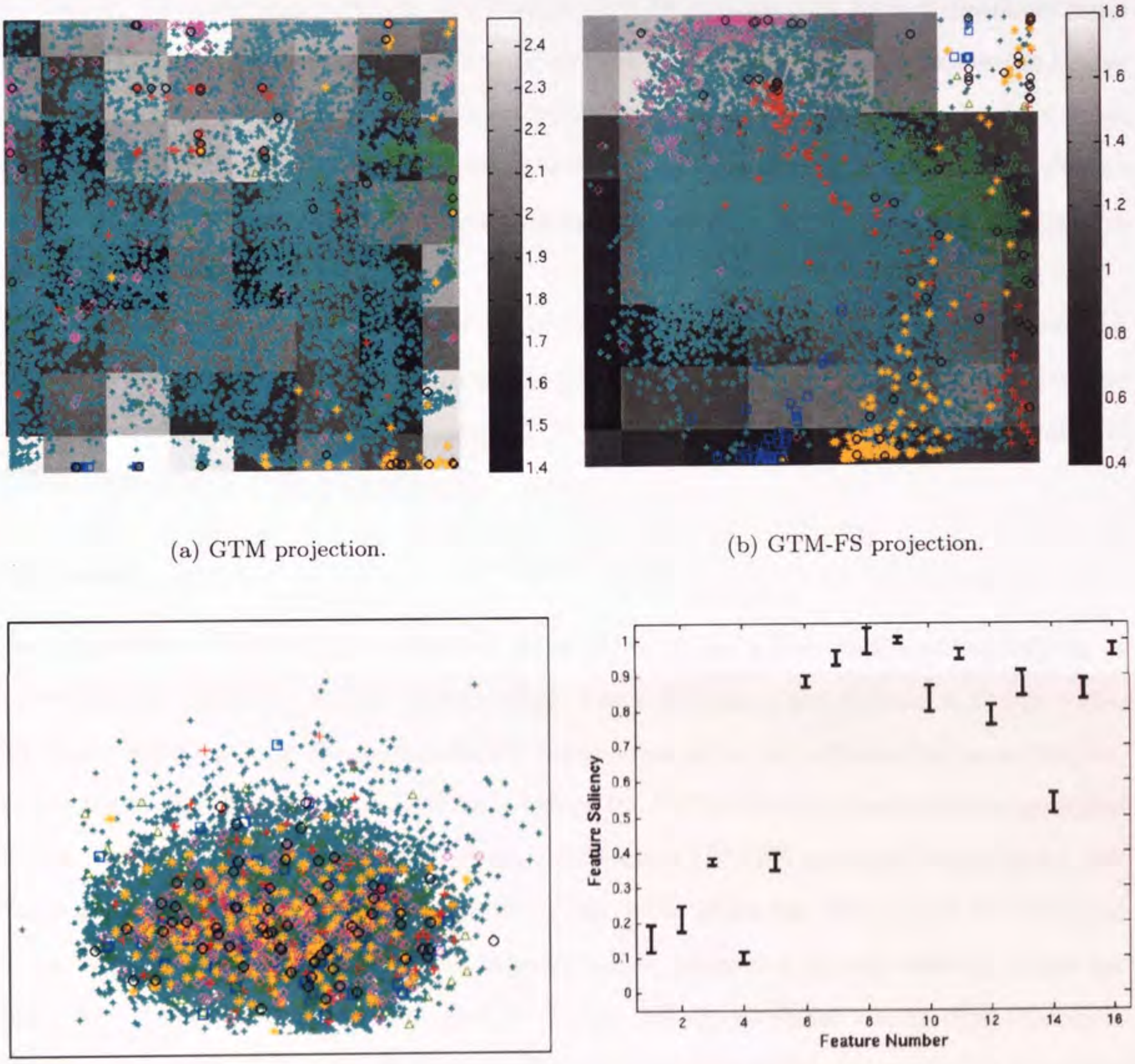
(c) SOM projection.

(d) Feature saliencies.

Figure 5.2: GTM, GTM-FS and SOM projections for the synthetic dataset consisting 800 data points from a mixture of four equiprobable Gaussians. The background in the GTM and GTM-FS plot is their corresponding magnification factors on a $\log_{10}$ scale. Figure (d) shows the estimated feature saliency mean values plus and minus one standard deviation over five runs with $\mathbf{W}$ initialised randomly.

5.3.2 Case study 2: The HTS dataset

Here we use the HTS dataset as described in Section 3.7.1. Our aim is to estimate descriptors' relevance (for all 16 descriptors) and obtain an effective presentation of the data.



(a) GTM projection.

(b) GTM-FS projection.

(c) SOM projection.

(d) Feature saliencies.

Figure 5.3: GTM, GTM-FS and SOM projections for the chemoinformatics dataset. Background in the GTM and GTM-FS plot is their corresponding magnification factors on a $\log_{10}$ scale. Please refer to Table 3.2 for legend. Figure (d) shows the estimated feature saliency mean values plus and minus one standard deviation over five runs with $\mathbf{W}$ initialised randomly.

The projections obtained using GTM, GTM-FS and SOM are presented in Figure 5.3. The background colour shading in Figure 5.3(a) and Figure 5.3(b) displays the corresponding magnification factors for those projection manifolds. The estimated saliencies of all the 16 features using GTM-FS, together with their standard deviations (error bars), are shown in Figure 5.3(d).

The main aim in analysis of this dataset, to understand and explore biological activity data combined with other whole-molecular physicochemical properties, was triggered from the need of the screening scientists at Pfizer to visually explore such high-dimensional large dataset. The GTM-based projections have proved very useful for the purpose and have outperformed the projections obtained from the traditional visualisation techniques, such as SOM. GTM-FS also gave feature relevance values. The screening scientists were able to identify common features of compounds active against different targets using the visualisation plot obtained using GTM-FS.

A comparative evaluation of these projections is presented in Table 5.1. A screening scientist is interested in increased accuracy of prediction for active compounds, and thus the NN classification error for active compounds is reported in Table 5.1 instead of overall NN classification error for the HTS dataset.

Discussion

The projection in Figure 5.3(c), obtained using SOM, is like a blob and does not help us to understand the 'structure' of data in data space. The GTM-based projections, in Figure 5.3(a) and Figure 5.3(b), show clear clusters for the compounds active for different biological targets. We get better KL-divergence and MF-sum values for GTM-FS which indicates the manifold obtained using GTM-FS is more coherent. GTM and GTM-FS provided much better NN classification error rate for active compounds than SOM where the data points are cluttered on top of each other in the projection manifold. The estimated feature saliency values for the chemoinformatics dataset, presented in Figure 5.3(d), confirms the general consensus in the pharmaceutical domain that physicochemical properties such as, molecular solubility, number of atoms, molecular weight, etc., are responsible for compounds grouping in the chemical space [Lipinski et al., 1997]. Chemists at Pfizer also confirmed that they would have expected higher feature saliency values for these physicochemical properties.

5.4 Computational cost

The distance calculation between data points and mixture components of reference vectors (used in calculation of $p(\mathrm{x}_n|\Theta)$), respectively, is identical in GTM, GTM-FS and SOM training algorithms. Updating the parameters in SOM training depends on the neighbourhood function. In the experiments presented here it was continuous on the latent space so the parameter updating scales as $\mathcal{O}(M^2ND + M^2)$, where M is the number of grid points in the SOM map and D is the dimension of the data space. When updating parameters, the GTM and GTM-FS require a matrix inversion of an $K \times K$ matrix, where K is the number of basis functions, followed by a set of matrix multiplications. The matrix inversion scales as $\mathcal{O}(K^3)$, while the matrix multiplications scales as $\mathcal{O}(MND)$[1], where M is the number of grid points in the latent space. GTM-FS requires an extra loop over the number of features, D, to reestimate the weight vector, $\hat{\mathbf{w}}_d$, in the EM algorithm.

Table 5.2 shows the time taken to train different projection models on the chemoinformatics dataset using an Intel Pentium 4 - 2.4GHz machine with 2GB of RAM. An implementation of the algorithms in C/C++ instead of MATLAB would further improve the speed.

The model	Time (seconds)	Architecture
GTM	42	$M = 256, K = 64$
GTM-FS	45	$M = 256, K = 64$
SOM	36	$M = 256$

Table 5.2: Training time for different projection models for the training set of the HTS dataset ($N = 11800$, $D = 16$, 20 iterations).

Once the models are trained, the computational cost to project data for the subsequent test set scales linearly in the number of data points (N) in the test set but is negligible by comparison.

To use GTM-FS under the visual data exploration framework (Chapter 3), the issue of the speed of the algorithm cannot be ignored as user interaction is important. The computational complexity of GTM-FS algorithm is similar to GTM, thus it can be directly used in such an interactive data mining framework.

5.5 Related work

It was brought to our attention in September 2006 that at the same time as our work Vellido et al. [2006] introduced a variant of GTM as a mixture of Student's t-distributions (t-GTM) to make it more robust to outliers and handle missing values. They then extended the t-GTM to implement simultaneous feature selection using the same mixture-based principle proposed in [Law et al., 2004]. Our method differs from [Vellido et al., 2006] in the sense that

[1] To be exact, the matrix multiplications scales as $\mathcal{O}(MKD + MND)$, but normally the number of data points, N, exceeds the number of basis functions, K.

we use diagonal Gaussian distribution as the components of the mixture. In [Vellido et al., 2006], it is mentioned that Vellido [2005] did look into feature selection with Gaussian GTM. Derivation of our algorithm was first reported in [Maniyar, 2005]. In future, a comparison of these two approaches for different datasets could be interesting. Wang and Kabán [2005] have implemented a similar feature selection approach for binary data.

5.6 Conclusions

Deriving useful information from a real-life large multivariate dataset in the chemoinformatics is difficult due to the inherent noise and the sheer amount of data. Data visualisation and feature selection are both individually important topics in data mining. Addressing both these problems jointly is not only logical but also synergistic as each endeavour could benefit from advances in the other when they are addressed jointly.

We successfully modified a feature selection method for unsupervised learning and applied it to the training of a probabilistic mixture-based data visualisation algorithm. The new algorithm, GTM-FS, not only provided a better projection by modelling irrelevant features ("noise") using a separate shared distribution but also estimated the feature saliency values correctly which helps the user assess the significance of each feature. The usefulness of the algorithm was demonstrated on both synthetic and real-life chemoinformatics datasets.

Since the estimation of feature saliencies is conveniently integrated with the training of a probabilistic mixture-based data visualisation model using a variant of EM algorithm, the computational complexity of the new algorithm remains tractable.

One of the interesting avenues for future work is to extend the approach for a probabilistic mixture-based hierarchical visualisation algorithm, such as hierarchical GTM [Tiňo and Nabney, 2002]. This is discussed further in the next chapter.

Chapter 6

Conclusions and Future Directions

In this final chapter first we summarise the work described in this thesis and draw some conclusions. Then we discuss some open questions and future avenues for the research.

6.1 Summary

This thesis has primarily been concerned with developing practical data exploration and modelling methods for large heterogeneous datasets that occur frequently during the early stages of pharmaceutical research. The work was motivated by a demand from the domain experts (i.e. screening scientists, chemists, biologists, etc.) to have better control and understanding over model development and data exploration so that informed decisions can be taken. We now summarise the main achievements in this thesis.

Visual data exploration framework (chapter 3)

In this chapter we introduced a flexible visual data exploration framework which combines advanced projection algorithms developed in the machine learning domain and powerful visual representation techniques developed in the information visualisation domain to facilitate direct involvement of the domain experts in the data exploration process. We identified appropriate probabilistic projection algorithms which not only give better clustering but also provide other projection manifold properties like magnification factors and directional curvatures plots which help the domain experts to understand the structures and the shape of projection manifolds for large datasets. Visual techniques such as local parallel coordinates and billboarding support effective data exploration by providing a means to study local patterns in different regions of a projection manifold and facilitating better navigation of a natural representation of data points (i.e. display of chemical structure image instead of a marker on the plot for chemical compounds dataset) on the projection plot, respectively. Hierarchical probabilistic visualisation algorithms provided additional insight and good segmentation for very large datasets where plotting all the data points on a single plot gives a cluttered and confusing plot. The tractable computation efficiency of the suggested algorithms demonstrates their suitability for a real-time software tool developed using the proposed visual data exploration framework.

Guided mixture of local experts (chapter 4)

In this chapter we exploited hierarchical non-linear projection algorithms and allowed user interaction using the visual data exploration framework to obtain a meaningful segmentation of the input space into regions with similar behaviour which was then used to develop user-guided variants of the mixture-of-experts model and the Bayesian committee model. The guided mixture of local experts model proved appropriate for heterogeneous pharmaceutical datasets. Though developing a model in such a way is a two-step process, i.e. first developing the hierarchy and then training the local experts according to the hierarchy, the performance and interpretability are worthwhile advantages for spending more time and effort in development. On the other hand, some kernel-based global prediction models, such as Gaussian processes, scale poorly with dataset size, typically requiring $\mathcal{O}(N^3)$ time and $\mathcal{O}(N^2)$ space, where N is the number of data points. In such cases, developing a guided mixture of local experts is not only faster but also provides better prediction by fitting individual local models to a restricted region with limited overlap.

Data visualisation with simultaneous feature selection (chapter 5)

Here we derived a generative topographic mapping (GTM) based data visualisation approach which estimates feature saliency simultaneously with the training of the visualisation model. We adopted a diagonal structure for the Gaussian components in GTM and introduced a separate background probability density function which models irrelevant features ('noise') in a dataset. The new algorithm not only provided better projection results by modelling irrelevant features using a separate background distribution but also estimated the feature saliency values correctly which helps the user to assess the significance of each feature. The results obtained using the algorithm are encouraging. There is a lot of potential in applying this algorithm to many other problems in chemoinformatics where the domain experts are always interested in finding the significance of different descriptors in a dataset. The computational complexity of the new algorithm remains tractable for use within the visual data mining framework.

6.2 Future directions

We envisage the following immediate future directions for the work presented in this thesis:

Visual data exploration framework

Though it might not be a core research area as it is mostly a software engineering issue, integrating the software developed using the visual data exploration framework with other popular data exploration tools in pharmaceutical research could be valuable for wider applications. The current version of the tool allows the results to be exported in such a way that they can be used in other tools, such as SpotFire, but tighter integration is desirable for the domain experts to have an easier access to the framework. We are currently working in this direction with Pfizer Global Research in which Pfizer plans to implement our framework as a part of their program to provide important visualisation techniques as a web-service to be used globally within Pfizer.

The research to extend important probabilistic projection algorithms, such as GTM and HGTM, for discrete data has already been done [Nabney et al., 2005]. Currently our tool does not provide a stable facility to carry out analysis on discrete datasets. In pharmaceutical research, domain experts often work with binary fingerprint[1] data of molecules so support for discrete data can be useful. Further software development is required in this direction to make that facility stable.

Guiding effective visualisation hierarchies is not a straight forward task. Experience and understanding of the manifold properties plots, such as magnification factors plots and directional curvatures plots, is required. This demands training for the domain experts in pharmaceutical research who are not used to work with such manifold properties plots. In this direction, we already have organised a training session for the researcher at Pfizer to use the software tool we developed and a tutorial for how to develop effective hierarchies. More research in automating parameter determination for GTM and providing more feedback during hierarchy development to non-statistical users is desirable.

[1] A binary representation of a molecule which describes in a computationally simple fashion absence or presence of a set of attributes (descriptors).

It is cumbersome to study patterns obtained using local parallel coordinates facility if the number of descriptors is high (more than 30). In such cases, rule induction as description for local groups of points could be useful.

Guided mixture of local experts

Development of GBCM, presented in chapter 4, is recent and was motivated from the limitation of the popular emulator methodology to model complex models due to high dimensionality and size of the datasets. As we suggested in a recent poster [Maniyar and Cornford, 2006], in the future we can exploit the projection of high dimensional inputs to a lower dimensional probabilistic manifold in a manner similar to the "warping" methods of [Sampson and Guttorp, 1992] although this will require careful modification of GTM. We plan to implement such a method within the managing uncertainty in complex models (MUCM) project (`http://mucm.group.shef.ac.uk`), and will also employ a related method which is approximately distance preserving, a feature that might be useful for mapping prior beliefs about the model inputs into the reduced dimension representation. It will be interesting to relate these methods to the kernel PCA (KPCA) [Schölkopf et al., 1998] and the Gaussian process latent variable model (GP-LVM) [Lawrence, 2005] methods recently developed in the machine learning community.

Data visualisation with simultaneous feature selection (GTM-FS)

Here one of the immediate interesting avenues for future work is to extend this approach for a probabilistic mixture-based hierarchical visualisation algorithm, such as hierarchical GTM (HGTM). The major challenge in extending the approach for hierarchical models is in deciding the strategies to carry forward the feature significance estimation obtained at the higher levels.

It will also be interesting to compare results of GTM-FS with visualisation and feature significance estimation obtained using a Gaussian process latent variable model (GP-LVM) with integrated automatic relevance determination (ARD).

The CASE award: Improved *in silico* prediction

Although guided local predictive techniques introduced in Chapter 4 outperform those in routine use and leading edge methods, the performance is not yet at the level that Pfizer require, and our belief is that further research is needed on the *representation* of drug-like molecules.

One way forward on this issue is to study small drug-like molecules that are not directly used by Pfizer with the aim of exploring different ways of representing structural information in order to assess which are the most effective in terms of predicting properties of interest. Aston University has a 3-year CASE studentship with Pfizer for this proposed research starting from October 2006. There are a number of publicly accessible databases containing structural information on ligands. including MSD ligand chemistry and CheBI (Chemical entities of Biological Interest) from EBI, PubChem[2] etc. These, and others recommended by Pfizer, will be used as the basis for the project.

During this research, it is also planed further investigate the relationship between the molecular descriptors and data visualisation. The new molecular descriptors will be used to characterise compounds in local regions selected using the software (DVMS) developed on the basis of the work carried out in this thesis.

[2] http://pubchem.ncgi.nlm.nih.gov/

Appendix A

The M-step of the EM Algorithm for GTM-FS

We can write the complete-data log-likelihood for the model in (5.5) as

$$P(\mathbf{t}_n, y_n = m, \Theta) = \frac{1}{M} \prod_{d=1}^{D} (\rho_d p(t_{nd}|\theta_{md}))^{\psi_d} ((1-\rho_d) q(t_{nd}|\lambda_d))^{(1-\psi_d)} \tag{A.1}$$

During the E-step, we calculate following quantities as in (5.6), (5.7) and (5.8) using the current parameter estimate Θ^{now}.

$$R_{nm} = P(y_n = m|\mathbf{t}_n) \tag{A.2}$$

$$u_{nmd} = P(\psi_d = 1, y_n = m|\mathbf{t}_n) \tag{A.3}$$

$$v_{nmd} = P(\psi_d = 0, y_n = m|\mathbf{t}_n) \tag{A.4}$$

Then the expected complete data log-likelihood based on Θ^{now} is

$$\begin{aligned}\mathcal{L} = &\underbrace{\sum_{md}\sum_{n} u_{nmd} \ln p(t_{nd}|\theta_{md})}_{\text{part 1}} + \underbrace{\sum_{d}\sum_{nm} v_{nmd} \ln q(t_{nd}|\lambda_d)}_{\text{part 2}} + \\ &\underbrace{\sum_{d}\left(\ln \rho_d \sum_{nm} u_{nmd} + \ln(1-\rho_d)\sum_{nm} v_{nmd}\right)}_{\text{part 3}},\end{aligned} \tag{A.5}$$

where ρ_d is the feature saliency for feature d, $p(\cdot|\theta_{md})$ is the probability density function (pdf) of the dth feature for the mth component, with parameters $\boldsymbol{\theta}_{md} = \{\Phi_m \mathbf{w}_d, {\sigma_d}^2\}$, and $q(t_{nd}|\lambda_d)$ is a 'background' Gaussian density. Note that the densities $p(\cdot)$ and $q(\cdot)$ are univariate Gaussian and are characterised by their means and covariances. Also note that the three parts in the equation above can be maximised separately with respect to different parameters.

By differentiating (A.5) w.r.t w_{id} where $i \in 1, ..., K$ and K is the number of basis functions in the nonlinear mapping (5.1), and using (5.3), we get

$$\frac{\partial \mathcal{L}}{\partial w_{id}} = \sum_m \sum_n u_{nmd} \left[\frac{(t_{nd} - \Phi_m \mathbf{w}_d)}{\sigma_d^2} \phi_{mi} \right],$$

setting above equation to 0 we get

$$\sum_m \sum_n u_{nmd}[(t_{nd} - \Phi_m \mathbf{w}_d)\phi_{mi}] = 0. \tag{A.6}$$

We get such K equations for $i = 1, \ldots, K$ which can be written in matrix notation as

$$\mathbf{\Phi}^T \mathbf{G}_d \mathbf{\Phi} \hat{\mathbf{w}}_d = \mathbf{\Phi}^T \mathbf{U}_d \mathbf{t}_d, \tag{A.7}$$

where $\mathbf{\Phi}$ is a $M \times K$ matrix, $\hat{\mathbf{w}}_d$ is a $K \times 1$ weight vector (the dth column of $\mathbf{W}$), $\mathbf{U}_d$ is a $M \times N$ matrix calculated using (5.7), t_d is a $N \times 1$ data vector, and $\mathbf{G}_d$ is an $M \times M$ diagonal matrix with elements

$$g_{mmd} = \sum_n^N u_{nmd}. \tag{A.8}$$

Similarly, differentiating (A.5) w.r.t σ_d, we get

$$\frac{\partial \mathcal{L}}{\partial \sigma_d} = \sum_m \sum_n u_{nmd} \left[-\frac{1}{2\hat{\sigma}_d^2} + \frac{(t_{nd} - \Phi_m \hat{\mathbf{w}})^2}{2(\hat{\sigma}_d^2)^2} \right] \tag{A.9}$$

setting above equation to 0 and solving it, we get

$$\hat{\sigma}_d = \frac{\sum_m \sum_n u_{nmd}(t_{nd} - \Phi_m \hat{\mathbf{w}}_d)^2}{\sum_m \sum_n u_{nmd}} \tag{A.10}$$

Equations for the re-estimation of the parameters of $q(\cdot)$ and feature saliency ρ_d are discussed in Section 5.2.1.

Bibliography

M. H. Abraham, H. S. Chadha, and R. C. Mitchell. Hydrogen bonding. 33. factors that influence the distribution of solutes between blood and brain. *Journal of Pharmaceutical Science*, 83(9):1257–1268, 1994.

R. J. Anderson. 2020 vision: A brave new world of drug development. *Current Drug Discovery*, 2002.

M. Ankerst. Visual data mining with pixel-oriented visualization techniques. *Proceedings of the Workshop on Visual Data Mining*, 2001.

M. Ankerst, D. Keim, and K. H. P. Circle segments: A technique for visually exploring lagre dimensional data sets. In *Proceedings of the IEEE Visualization Conference*, 1996.

F. Aurenhammer. Voronoi diagrams - survey of a fundamental geometric data structure. *ACM Computing Surveys*, 3:345–405, 1991.

K. A. Bachmann and R. Ghosh. The use of *in vitro* methods to predict *in vivo* pharmacokinetics and drug interactions. *Current Drug Metabolism*, 2(3):299–314, 2001.

J. . Bai, A. Utis, G. Crippen, H. D. He. V. Fischer, R. Tullman, H. Q. Yin, C. P. Hsu, L. Jiang, and K. K. Hwang. Use of classification regression tree in predicting oral absorption in humans. *Journal of Chemical Information and Computer Science*, 44(6):2061–2069, 2004.

J. Bajorath. Rational drug discovery revisited: interfacing experimental programs with bio- and chemo-informatics. *Drug Discovery Today*, 6:989–995, 2001.

J. Bajorath. Integration of virtual screening and high-throughput screening. *Nature Reviews*, Drug Discovery 1:882–894, 2002a.

J. Bajorath. Virtual screening in drug discovery: Methods, expectations and reality. *Current Drug Discovery*, March:24–28, 2002b.

P. Baldi and G. Hatfield. *DNA microarrays and gene expression.* Cambridge University Press, Cambridge, 2002.

G. P. van Balen, C. M. Martinet, G. Caron, G. Bouchard, M. Reist, P. Carrupt, R. Fruttero, A. Gasco, and B. Testa. Liposome/water lipophilicity: Methods, information content, and pharmaceutical applications. *Medicinal Research Reviews*, 24(3):299–324, 2004.

D. J. Bartholomew. The foundations of factor analysis. *Biometrika*, 71(2):221–232, 1984.

M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, 2003.

C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, Oxford, 1st edition, 1995.

C. M. Bishop, M. Svensén, and C. K. I. Williams. GTM: a principled alternative to the self-organizing map. In M. C. Mozer, M. I. Jordan, and T. Petsche Eds., editors, *Advances in Neural Information Processing Systems 9*, pages 354–360. MIT Press, Cambridge, 1997a.

C. M. Bishop, M. Svensén, and C. K. I. Williams. Magnification factors for the GTM algorithm. *Proceedings IEE Fifth International Conference on Artificial Neural Networks*, pages 64–69, 1997b.

C. M. Bishop, M. Svensén, and C. K. I. Williams. Magnification factors for the som and gtm algorithms. *Workshop proceedings on Self-Organizing Maps*, pages 333–338, 1997c.

C. M. Bishop, M. Svensén, and C. K. I. Williams. GTM: The generative topographic mapping. *Neural Computation*, 10:215–234, 1998.

C. M. Bishop and M. E. Tipping. A hierarchical latent variable model for data visualization. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 20(3):281–293, 1998.

A. Blum and P. Langley. Selection of relevant features and examples in machine learning. *Artificial Intelligence*, 97(1-2):245–271, 1997.

R. S. Bohacek, C. McMartin, and W. C. Guida. The art and practice of structure-based drug design: a molecular modeling perspective. *Med Res Rev.*, 16(1):3–50, 1996.

M. B. Brennan. Drug discovery: Filtering out failures early in the game. *Chemical Engineering News*, 78(23):63–73, 2000.

W. L. Chen. Chemoinformatics: Past, present, and future. *Journal of Chemical Information and Modeling*, 2006. ASAP Web Release: http://pubs3.acs.org/acs/journals/doilookup?in_doi=10.1021/ci060016u.

E. Clark and S. D. Pickett. Computational methods for the prediction of 'drug-likeness'. *Drug Discovery Today*, 5:49–58, 2000.

T. Cover and J. Thomas. *Elements of Information Theory*. Wiley, New York, 1st edition, 1991.

T. F. Cox and M. A. A. Cox. *Multidimensional Scaling*. Chapman and Hall, London, 2 edition, 2001.

L. Csato and M. Opper. Sparse online Gaussian Processes. *Neural Computation*, 14(3): 641–669, 2002.

J. C. Dearden. *In silico* prediction of drug toxicity. *Journal of Computer–Aided Molecular Design*, 17(2–4):119–127, 2003.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, B 39:1–38, 1977.

D. J. Diller and D. W. Hobbs. Deriving knowledge through data mining high-throughput screening data. *Journal of Medicinal Chemistry*, 47:6373–6383, 2004.

J. A. Dimasi. Risks in new drug development: approval success rates for investigational drugs. *Clinical Pharmacology & Therapeutics*, 69:297–307, 2001.

C. M. Dobson. Chemical space and biology. *Nature reviews on drug discovery*, 432(7019): 824–828, 2004.

R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification.* John Wiley and Sons, Inc., New York, 2000.

J. G. Dy and C. E. Brodley. Feature selection for unsupervised learning. *The Journal of Machine Learning Research*, 5:845–889, 2004.

R. A. Edwards, K. Zhang, and L. Firth. Benchmarking chemistry functions within pharmaceutical drug discovery and preclinical development. *Drug Discovery World*, 67:71–76, 2002.

S. Ekins, C. L. Waller, P. W. Swaan, G. Cruciani, S. A. Wrighton, and J. H. Wikel. Progress in predicting human ADME parameters *in silico*. *Journal of Pharmacology and Toxicology Methods*, 44(1):251–272, 2000.

R. S. Ellis. *Entropy, Large Deviations, and Statistical Mechanics.* Springer-Verlag, New York, 1985.

P. Englebienne. High throughput screening: Will the past meet the future? *Frontiers in Drug Design & Discovery*, 1:69–86, 2005.

T. J. Ewing, S. Makino, A. G. Skillman, and I. D. Kuntz. Dock 4.0: search strategies for automated molecular docking of flexible molecule databases. *Journal of Computer–Aided Molecular Design*, 15:411–428, 2001.

M. C. Ferreira de Oliveira and H. Levkowitz. From visual data exploration to visual data mining: A survey. *IEEE Transaction on Visualization and Computer Graphics*, 9(3):378–394, 2003.

A. Flanagan, P. Guy, M. Steiner, J. Altshuler, and P. Tollman. A revolution in r&d: How genomics and genetics are transforming the biopharmaceutical industry, 2001.

S. Fox, S. Farr-Jones, L. Sopchak, and H. Wang. Fine-tuning the technology strategies for lead finding. *Drug Discovery World*, Summer:24–30, 2002.

S. Fox, S. Farr-Jones, and M. A. Yund. High-throughput screening for drug discovery: continually transitioning into new technologies. *Journal of Biomolecular Screening*, 4:183–186, 1999.

J. Gasteiger. *Handbook of Chemoinformatics: From Data to Knowledge, 4 Volume Set.* Wiley, Germany, 2003.

J. W. Godden and J. Bajorath. A distance function for retrieval of active molecules from complex chemical space representations. *Journal of Chemical Information and Modeling*, 43(3):1094–1097, 2006.

G. H. Golub and C. F. van Loan. *Matrix Computations.* Johns Hopkins University Press, Baltimore, 1996.

A. C. Good, S. R. Krystek, and J. S. Mason. High-throughput and virtual screening: core lead discovery techologics move towards integration. *Drug Discovery Today*, 5:S61–S69, 2000.

D. Gorse and R. Lahana. Functional diversity of compound libraries. *Current Opinion in Chemical Biology*, 4:287–294, 2000.

R. L. Gorsuch. *Factor Analysis.* Erlbaum, Hillsdale, NJ, 1996.

P. Gribbon and A. Sewing. High-throughput drug discovery: what can we expect from HTS? *Drug Discovery Today*, 10(1):17–22, January 2005.

M. Gross, T. Sprengel, and J. Finger. Visualizing information on a sphere. In *Proceedings of IEEE Information Visualization '97*, 1995.

T. A. Halgren, R. B. Murphy, R. A. Friesner, H. S. Beard, L. L. Frye, W. Thomas Pollard, and J. L. Banks. Glide: a new approach for rapid, accurate docking and scoring. 2. enrichment factors in database screening. *Journal of Medicinal Chemistry*, 47(7):1750–1759, 2004.

J. S. Handon. High-throughput screening – challenges for the future. *Drug Discovery World*, Summer:47–50, 2002.

H. H. Harman. *Modern Factor Analysis.* Univ. of Chicago Press, 1967.

A. Hinneburg, D. A. Keim, and M. Wawryniuk. HD-Eye: Visual mining of high-dimensional data. *IEEE Transactions on Computer Graphics and Applications*, 19(5):22–31, 1999.

P. Hoffman, G. Grinstein, K. Marx, I. Grosse, and E. Stanley. DNA visual and analytic data mining. In *Proceedings of the 8th IEEE Visualization Conf.*, 1997.

T. Hou and X. Xu. Adme evaluation in drug discovery. 1. applications of genetic algorithms to the prediction of blood-brain partitioning of a large set of drugs. *Journal of Molecular Modeling*, 8(12):337–349, 2002.

S. Huang, M. O. Ward, and E. A. Rundensteiner. Exploration of dimensionality reduction for text visualization. In *Proceedings the Coordinated and Multiple Views in Exploratory Visualization (CMV'05)*, pages 63–74, Washington, DC, USA, 2005. IEEE.

A. Inselberg and B. Dimsdale. Parallel coordinates : A tool for visualizing multi-dimensional geometry. In *Proceedings IEEE VISUALIZATION '90*, pages 361–375, 1990.

R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton. Adaptive mixture of local experts. *Neural Computation*, 3:79–87, 1991.

I. T. Jolliffe. *Principal Component Analysis*. Springer-Verlag, New York, 2nd edition, 2002.

M. I. Jordan and R. A. Jacobs. Hierarchical mixture of experts and the EM algorithm. *Neural Computation*, 6:181–214, 1994.

M. Karelson. *Molecular Descriptors in QSAR/QSPR*. Wiley, Germany, 2000.

D. A. Keim. Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, 8(3):255–269, 2002.

T. Kennedy. Managing the drug discovery/development interface. *Drug Discovery Today*, 2: 436–444, 1997.

E. H. Kerns and L. Di. Pharmaceutical profiling in drug discovery. *Drug Discovery Today*, 8:316–323, 2003.

D. B. Kitchen, H. Decornez, J. R. Furr, and J. Bajorath. Docking and scoring in virtual screening for drug discovery: Methods and applications. *Nature Reviews Drug Discovery*, 3(11):935–949, 2004.

R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97 (1-2):273–324, 1997.

T. Kohonen. *Self-Organizing Maps*. Springer-Verlag, Berlin, 1995.

E. L. Koua and M.-J. Kraak. Geovisualization to support the exploration of large health and demographic survey data. *International Journal of Health Geographics*, 3(12):1–13, 2004.

B. Kramer, , M. Rarey, and T. Lengauer. Evaluation of the flexx incremental construction algorithm for protein-ligand docking. *Proteins*, 37:228–241, 1999.

M. Kreuseler and H. Schumann. A flexible approach for visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):39–51, 2002.

R. Lahana. How many leads from HTS? *Drug Discovery Today*, 4:447–448, 1999.

R. Lahana. Cheminformatics - decision making in drug discovery. *Drug Discovery Today*, 7 (17):898–900, 2004.

M. H. C. Law, M. A. T. Figueiredo, and A. K. Jain. Simultaneous feature selection and clustering using mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1154–1166, 2004.

N. D. Lawrence. Probabilistic non-linear principal component analysis with Gaussian process latent variable models. *The Journal of Machine Learning Research*, 6:1783–1816, 2005.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

S. Levy. Interactive 3-d visualization of particle systems with Partiview. *Proceedings of the Int Astronomical Union Symposium*, 208:85–91, 2001.

T. Li, C. Zhang, and M. Ogihara. A comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression. *Bioinformatics*, 20 (15):2429–2437, 2004.

L. Lin, D. C. Sahakian, S. M. de Morais, J. J. Xu, R. J. Polzer, and S. M. Winter. The role of absorption, distribution, metabolism, excretion and toxicity in drug discovery. *Current Topics in Medicinal Chemistry*, 3(10):1125–1154, 2003.

C. Lipinski and A. Hopkins. Navigating chemical space for biology and medicine. *Nature*, 432:855–861, 2004.

C. A. Lipinski, F. Lombardo, B. W. Dominy, and P. J. Feeney. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Advanced Drug Delivery Reviews*, 23:3–25, 1997.

B. Liu, S. Li, and J. Hu. Technological advances in high-throughput screening. *American Journal of PharmacoGenomics*, 4(4):263–276, 2004.

Y. Liu. A comparative study on feature selection methods for drug discovery. *Journal of Chemical Information and Computer Science*, 44(5):1823–1828, 2004.

D. J. Livingstone and D. T. Manallack. Neural networks in 3d qsar. *QSAR & Combinatorial Science*, 22(5):510–518, 2003.

J. Lösel. Interaction fingerprints - a new descriptor for noncovalent interactions around functional groups. In MGMS, 1998.

D. Lowe and M. E. Tipping. Neuroscale: Novel topographic feature extraction with radial basis function networks. *Advances in Neural Information Processing Systems*, 9:543–549, 1997.

P. D. Lyne. Structure-based virtual screening: an overview. *Drug Discovery Today*, 7:1047–1055, 2002.

N. Mahmoudi, J. de Julián-Ortiz, L. Ciceron, J. Gálvez, D. Mazier, M. Danis, F. Derouin, and R. Garcia-Domenech. Identification of new antimalarial drugs by linear discriminant analysis and topological virtual screening. *Journal of Antimicrobial Chemotherapy*, 57(3): 489–497, 2005.

D. M. Maniyar. Em algorithm for GTM-FS. Technical report, Neural Computing Research Group (NCRG), Aston University, 2005. Technical Report NCRG/2005/012.

D. M. Maniyar. miniDVMSv1.8 : A user manual. Technical Report NCRG/2006/013, Neural Computing Research Group, Aston University,UK, 2006.

D. M. Maniyar and D. Cornford. Dealing with large complex models in emulators: Guided bayesian committee machine. In *Assessment and Utilization of Complex Computer Models Opening Workshop*. The Statistical and Applied Mathematical Sciences Institute (SAMSI), 2006.

D. M. Maniyar and I. T. Nabney. Guiding local regression using visualisation. *In Deterministic and Statistical Methods in Machine Learning, LNAI, Springer-Verlag*, 3635:98–109, 2005.

D. M. Maniyar and I. T. Nabney. Data visualisation with simultaneous feature selection. In *Proceedings of the IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, pages 156–163, 2006a.

D. M. Maniyar and I. T. Nabney. Visual data mining: Integrating machine learning with information visualization. In *Proceedings of the 7th International Workshop on Multimedia Data Mining*. ACM digital library, 2006b. 63–72.

D. M. Maniyar and I. T. Nabney. Visual data mining using principled projection algorithms and information visualization techniques. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 643–648, 2006c.

D. M. Maniyar, I. T. Nabney, B. S. Williams, and A. Sewing. Data visualization during the early stages of drug discovery. *Journal of Chemical Information and Modelling*, 46(4): 1806–1818, 2006.

S. L. McGovern, E. Caselli, N. Grigorieff, and B. K. Shoichet. A common mechanism underlying promiscuous inhibitors from virtual and high-throughput screening. *Journal of Medicinal Chemistry*, 45(8):1712–1722, 2002.

P. Mitra, C. A. Murthy, and S. K. Pal. Unsupervised feature selection using feature similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):301–312, 2002.

D. Modha and S. Spangler. Feature weighting in k-means clustering. *Machine Learning*, 52 (3):217–237, 2003.

P. Moerland. *Mixture models for unsupervised and supervised learning*. PhD thesis, École polytechnique fédérale de lausanne, Lausanne, 2000.

I. T. Nabney. *Netlab: Algorithms for Pattern Recognition.* Springer, London, 2001.

I. T. Nabney, Y. Sun, P. Tiňo, and A. Kabán. Semisupervised learning of hierarchical latent trait models for data visualization. *IEEE Transaction on Knowledge and Data Engineering*, 17(3):384–400, 2005.

T. I. Oprea, J. Li, S. Muresan, and K. C. Mattes. High throughput and virtual screening: choosing the appropriate leads. In Ford M., Livingstone D., Dearden J., and van de Waterbeemd H., editors, *EuroQSAR 2002 - Designing Drugs and Crop Protectants: Processes, Problems and Solutions*, pages 40–47. New York: Blackwell Publishing, 2003.

T. I. Oprea and H. Matter. Integrating virtual screening in lead discovery. *Current Opinion in Chemical Biology*, 8:349–358, 2004.

F. Osterberg, G. M. Morris, M. F. Sanner, A. J. Olson, and D. S. Goodsell. Automated docking to multiple target structures: incorporation of protein mobility and structural water heterogeneity in autodock. *Proteins*, 46:34–40, 2002.

D. Plewczynski, S. A. H. Spieser, and U. Koch. Assessing different classification methods for virtual screening. *Journal of Chemical Information and Modeling*, 46(3):1098–1106, 2006.

B. L. Podlogar, I. Muegge, and L. J. Brice. Computational methods to estimate drug development parameters. *Current Opinion in Drug Discovery & Development*, 4(1):102–109, 2001.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning.* MIT Press, Cambridge, 2006.

B. D. Ripley. *Pattern Recognition and Neural Networks.* Cambridge University Press, 1996.

B. R. Roberts. Screening informatics: adding value with meta-data structures and visualization tools. *Drug Discovery Today*, 5(1):10–14, 2000.

J. Sadowski. Optimization of chemical libraries by neural networks. *Current Opinion in Chemical Biology*, 4:280–282, 2000.

J. W. Sammon. A nonlinear mapping for data structure analysis. *IEEE Transaction on Computation*, C-18:401–409, 1969.

P. D. Sampson and P. Guttorp. Nonparametric estimation of nonstationary spatial covariance structure. *Journal of the American Statistical Association*, 87(417):108–119, 1992.

P. Sanseau. Impact of human genome sequencing for in silico target discovery. *Drug Discovery Today*, 6:316–323, 2001.

G. Schneider. Neural networks are useful tools for drug design. *Neural Networks*, 13:15–16, 2000.

B. Schölkopf, A. J. Smola, and K. R. Muller. Nonlinear component analysis as a kernel eigenvalue problem, 1998.

A. Schwaighofer. BCM toolbox. *http://ida.first.fraunhofer.de/~anton/software.html*, 2005.

P. Sebastiani, E. Gussoni, I. Kohane, and M. Ramoni. Statistical challenges in functional genomics. *Statistical Science*, 18(1):33–70, 2003.

H. E. Selick, A. P. Beresford, and M. H. Tarbit. The emerging importance of predictive ADME simulation in drug discovery. *Drug Discovery Today*, 7(2):109–116, 2002.

B. Shneiderman. The eyes have it: A task by data type taxonomy for information. visualizations. *Proceedings of the 1996 IEEE Symposium on Visual Languages*, 3(6):336–343, Sep 1996.

B. K. Shoichet. Virtual screening of chemical libraries. *Nature*, 432:862–865, 2004.

D. A. Smith and H. van de Waterbeemd. Pharmacokinetics and metabolism in early drug discovery. *Current Opinion in Chemical Biology*, 3:373–378, 1999.

T. C. Sprenger, R. Brunella, and M. H. Gross. H-blob: A hierarchical visual clustering method using implicit surfaces. In *Proceedings of IEEE Visualization 2000*, pages 61–68, 2002.

B. R. Stockwell. Exploring biology with small organic molecules. *Nature*, 432(7019):846–854, 2004.

D. Surendran and S. Levy. Visualizing high dimensional datasets using partiview. *IEEE Symposium on Information Visualization. INFOVIS '04*, 2004.

J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

M. E. Tipping and C. M. Bishop. Mixtures of probabilistic principal component analysers. *Neural Computation*, 11(2):443–482, 1999a.

M. E. Tipping and C. M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society*, 6(3):611–622, 1999b.

M. E. Tipping and D. Lowe. Shadow targets: A novel algorithm for topographic projections by radial basis functions. *Neurocomputing*, 19:211–222, 1998.

P. Tiňo and I. T. Nabney. Constructing localized non-linear projection manifolds in a principled way: hierarchical generative topographic mapping. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 24:639–656, 2002.

P. Tiño, I. T. Nabney, and Y. Sun. Using directional curvatures to visualize folding patterns of the GTM projection manifolds. In H. Bischof G. Dorffner and K. Hornik, editors, *Proceedings of the International Conference on Artificial Neural Networks*, pages 421–428, 2001a.

P. Tiño, I. T. Nabney, Y. Sun, and B. S. Williams. A principled approach to interactive hierarchical non-linear visualization of high-dimensional data. *Computing Science and Statistics*, 33, 2001b.

P. Tiño, I. T. Nabney, B. S. Williams, J. Lösel, and Y. Sun. Nonlinear prediction of quantitative structure-activity relationships. *Journal of Chemical Information and Computer Sciences*, 44(5):1647–1653, 2004.

L. Torgo. Functional models for regression tree leaves. In *Proceedings of the fourteenth International Conference on Machine Learning*, pages 385–393. Morgan Kaufmann Publishers, 1997.

L. Torgo. *Inductive Learning of Tree-based Regression Models.* PhD thesis, University of Porto, Portugal, 1999.

V. Tresp. The Bayesian committee machine. *Neural Computation*, 12(11):2719–2741, 2000.

J. H. van Drie and M. S. Lajiness. Approaches to virtual library design. *Drug Discovery Today*, 3:274–283, 1998.

A. Vellido. Preliminary theoretical results on a feature relevance determination method for generative topographic mapping. Technical report, Universitat Politècnica de Catalunya (UPC), 2005. Technical Report LSI-05-13-R.

A. Vellido, P. J. G. Lisboa, and D. Vicente. Robust analysis of mrs brain tumour data using t-gtm¿. *Neurocomuting*, 69:754–768, 2006.

S. Venkatesh and R. A. Lipper. Role of the development scientist in compound lead selection and optimization. *Journal of Pharmaceutical Sciences*, 189(2):145–154, 2000.

J. Vesanto. SOM-based data visualization method. *Intelligent Data Analysis*, 3(2):111–126, 1999.

J. Vesanto. Using SOMs in data mining. Technical report, Helsink Univ. of Technology, 2000.

C. S. Wallace and D. L. Dowe. Minimum message length and Kolmogorov complexity. *Computer Journal*, 42(4):270–283, 1999.

S. Wang and H. Wang. Knowledge discovery through self-organizing maps: data visualization and query processing. *Knowledge and Information Systems*, 4(1):31–45, 2002.

X. Wang and A. Kabán. Finding uninformative features in binary data. In M. Gallagher, J. M. Hogan, and F. Maire, editors, *Intelligent Data Engineering and Automated Learning*, pages 40–47. Springer, 2005.

H. van de Waterbeemd and E. Gifford. ADMET in silico modelling: towards prediction paradise? *Nature Reviews Drug Discovery*, 2(3):192–204, 2003.

H. van de Waterbeemd, D. A. Smith, K. Beaumont, and D. K. Walker. Property-based design: Optimisation of drug absorption and pharmacokinetics. *Journal of Medicinal Chemistry*, 44:1313–1333, 2001.

P. C. Won. Visual data mining. *IEEE Transactions on Computer Graphics and Applications*, 19(5):20–21, 1999.

Y. Xiao, A. Clauset, R. Harris, E. Bayram, P. Santago, , and J. D. Schmitt. Supervised self-organizing maps in drug discovery. 1. robust behavior with overdetermined data sets. *Journal of Chemical Information and Modeling*, 45(6):1749–1758, 2005.

E. P. Xing, M. I. Jordan, and R. M. Karp. Feature selection for high-dimensional genomic microarray data. In *Proceedings 18th International Conference on Machine Learning*, pages 601–608. Morgan Kaufmann, San Francisco, CA, 2001.

F. Young. *Multidimensional Scaling: History, Theory, And Applications*. Lawrence Erlbaum Assoc., Hillsdale, N.J., 1987.

H. Yu and A. Adedoyin. ADME-tox in drug discovery: integration of experimental and computational technologies. *Drug Discovery Today*, 8:852–861, 2003.