Cluster statistics and gene expression analysis

Marta Łuksza

2011

Dissertation zur Erlangung des Grades
eines Doktors der Naturwissenschaften (Dr. rer. nat.)
am Fachbereich Mathematik und Informatik
der Freien Universität Berlin

1. Referent: Prof. Dr. Martin Vingron
2. Referent: Prof. Dr. Johannes Berg

Tag der Promotion: 15 Juli 2011

Preface

Acknowledgements. This work was carried out at the Department of Computational Molecular Biology at the Max Planck Institute for Molecular Genetics in Berlin. I would like to thank my supervisor Martin Vingron for his support and encouragement. I have been co-supervised by Johannes Berg and Michael Lässig from the Institute for Theoretical Physics in Cologne University. I really enjoyed our collaboration. I am grateful for the patience they showed in introducing me to statistical mechanics and for their guidance during the thesis. Their influence has shaped my scientific thinking to a great extent.

I would like to thank all the members of the Computational Molecular Biology Department at the Max Planck Institute in Berlin and the members of the Cologne group: I had a wonderful working environment. I am especially grateful to my office mates: Ewa Szczurek, Marcel Schulz, and Matthias Heinig, for the great time we had and for all the help with daily problems of my thesis. Special thanks to Marcel and Matthias again, for the help with preparation of the German version of the summary of this thesis.

I would like to thank my parents, for their ongoing support and Rafał, for discussions on the philosophical aspects of clustering, for his comments on the manuscript of my thesis, and for being there for me all this time.

I have been a member of the International Research Training Group "Genomics and Systems Biology of Molecular Networks". I have also been supported by SFB 680 "Molecular Basis of Evolutionary Innovations". I am very grateful to the Kavli Institute for Theoretical Physics in Santa Barbara for hosting me on their workshops during my PhD time.

Publications. Parts of the material covered in Chapter 4 were published in Physical Review Letters [68].

Marta Łuksza Berlin, Mai 2011

Contents

Preface **i**

1 Introduction **1**
1.1 Clustering 1
1.2 Genes and gene expression 4
1.3 Thesis organisation 8

2 Elements of statistical mechanics **11**
2.1 Physical system 11
2.2 Mathematical tools 14
2.3 Energy distribution 17
2.4 Disordered systems 19
2.5 Relation to information theory: the maximum entropy principle . . . 21

3 Statistical theory of clusters **25**
3.1 General setting 25
3.2 Clusters based on point density and positional bias 27
3.3 Clusters based on point density 30
3.4 Clusters based on positional bias 32
3.5 Clusters based on directional density 32
3.6 Summary 36

4 Statistical significance analysis of clusters **39**
4.1 Significance analysis 39
4.2 Statistics of clusters based on directional density 42
4.3 Statistics of clusters based on positional bias 48
4.4 Cluster score statistics and extreme value theory 50
4.5 Resampling-based methods 52
4.6 Application to gene expression data 55
4.7 Summary 60

5 Estimating dependencies between experimental conditions **61**
5.1 Motivation 61
5.2 Covariance and correlation matrix 62
5.3 Generalised statistical theory of clusters 67
5.4 Mixture-model for estimation of component dependencies 68
5.5 Application in tumour classification 75

5.6 Summary and discussion 81

6 Significance-based clustering algorithm 85
6.1 Introduction 85
6.2 Previous work: semi-supervised and constrained clustering 86
6.3 Significance-constrained mixture-model 87
6.4 Application to gene expression data 91
6.5 Summary 99

7 Summary and outlook 101

Bibliography 105

Appendix A: Normalisation constants in the spherical model 115

Appendix B: Details of the EM algorithm 117

Appendix C: Replica calculation for the maximal cluster score distribution 119

Zusammenfassung 133

Summary 135

Curriculum Vitae 137

Chapter 1

Introduction

The goal of this thesis is to establish a statistical grounding for the analysis of high-dimensional data, with a particular focus on gene expression. The analysis involves ordering data elements, by discovering "hidden" structures of a potential functional importance. In our approach, we consider *cluster structures* – groups of elements with high level of mutual similarity. We are concerned both with clusters of data elements and with clusters of data components; and finally, with an intricate interplay between them.
It is a topic of ongoing debate, whether clustering can be recognised as a reliable scientific procedure with an objective validation scheme [14, 19, 103]. Clustering is sometimes referred to as an "ill-defined problem" [69]. In the first part of the introduction, we stress the main concerns about clustering and we provide a context for our solution to the cluster validation problem – the statistical significance analysis of clusters.
In the second part, we provide a brief biological introduction to genetics, gene expression, and to common computational approaches in the analysis of such data. The aim of this part is to build the intuition behind the particular probabilistic models proposed in this thesis.
In the last section, we discuss briefly the main developments of this thesis: the statistical significance-analysis for clusters, the method for estimating vector-component dependencies, and the significance-based clustering algorithm.

1.1 Clustering

Clustering, which involves dividing data elements into classes based on their observed properties, is one of the main tools in exploratory data analysis. "Clustering relates data to knowledge and is a basic human activity" [103]. In general terms, the task of clustering can be formulated as follows: given a set of N elements, find its partition into K classes, such that the elements within groups are more similar to each other then the elements that belong to different groups. Such classes are meant to express the structure of the data, not given *a priori*.

Questioning the clustering as a "non-scientific" or an "ill-defined" method is related to its important features: its dependence on *free parameters* and the role of the *context* of the dataset in choosing appropriate clustering algorithm.

Parameters. The clustering task is usually formulated as an optimisation problem - with a *scoring function*. The scoring function depends on free parameters: the most important scoring parameter weighs number versus size of clusters and is contained explicitly (e.g., the number k in k-means clustering) or implicitly (e.g., the temperature in superparamagnetic [12] and information-based clustering [92]) in all clustering procedures. Choosing smaller values of k will give fewer, but larger clusters with lower average similarity between elements. Larger values of k will result in more, but smaller clusters with higher average similarity. None of these choices is by principle better than any other; both tight and loose clusters may reflect important structural similarities within a dataset.

Context. From the technical point of view, any clustering depends on two ingredients: a notion of similarity between elements of the dataset, and an algorithmic procedure that groups elements into clusters. Diverse methods address both aspects of clustering: similarities can be defined by Euclidean or by information-theoretic measures [95, 92], and there are many different clustering algorithms ranging from classical k-means [70] and hierarchical clustering [105] to recent message-passing techniques [36].

The choice of the clustering algorithm or the similarity measure cannot be made in abstraction from the context of a particular dataset and expectations about structure of clusters. Finding a phylogenetic ordering of organisms, for example, requires taking into account their evolutionary history and modelling a hierarchical process that reflects these relationships. In another class of problems, clusters are formed through interactions between elements; e.g., diseased individuals in a population form a cluster by spreading of an epidemic. Such clusters can have arbitrary shapes and are often modelled with graph-based approaches. Yet another class of problems assumes a "shallow" cluster generating process and views clusters as dense structures of data elements centred, with some deviation, around a typical value.

Therefore, the choice of clustering method and parameters must take into account preexisting knowledge and interpretation of the data, and it involves making a hypothesis about the structure of the dataset.

As a consequence, notion of a valid or "true" cluster is tightly related to the specific context of a given clustering problem and the goal the clustering is performed for. As an example, regard the starry sky. One may be tempted to say that some of the discernible groups of stars form "true" clusters – the ones which have been formed through an astrophysical process and consist of stars bound to each other by gravity. But to be exact, one would have to add that by "true" we understand "the ones that share a common origin". In other application, we may indeed be interested in finding patterns formed by stars that appear close to each other in two dimensions on the celestial-sphere. Such clusters are not *per se* less "true" then the other ones – they are "true", relatively to the focus and the context of the search.

1.1.1 Is clustering a science?

Clustering validation. While the algorithmic aspect of the problem is well-studied, the problem of *clustering validation* still poses a conceptual challenge [14, 19, 103]. In particular, the question is whether one can formulate general, problem-independent techniques for assessing the quality of clusterings [103]. In other words, is clustering a *scientific problem*, does it provide objectively testifiable hypotheses? Such a validation is possible in *supervised classification*, where data elements are assigned to predefined classes. Classification can be tested experimentally, and the quality assessment is clearly defined, based on the misclassification rate. But in case of clustering, which is an *unsupervised classification*, the "true" structure is not known, therefore there is no objective point of reference for the validation.

Clustering and scientific discovery. If we roughly define heuristics as methods of problem-solving based on experience, which find solutions that "work" without conducting a thorough, exhaustive search, then clustering is, indeed, a heuristic method. A vast part of a clustering procedure, aimed at finding the "structure" of the data, involves factors which are not "objective", and requires formulation of *ad hoc* hypotheses, based on a specific, domain-dependent context. In other words, in the process of clustering we do not exactly know what we are looking for, until we find it. But does it mean that clustering is not "scientific"? One can claim quite the opposite. The very nature of scientific discovery requires making of hypotheses which are not deductible from existing theories. Any breaking-through scientific research cannot be governed entirely by universal, generally-applicable rules, precisely because it aims at discovering the new and the unknown. From that point of view, clustering is not an "ill-defined" procedure, but a perfect example of a fruitful scientific method.

We put aside the problem of scientific nature of clustering, a question remains: is there any field where clustering can be improved in a unified, objective manner?

1.1.2 Statistical significance of clusters.

"It is conceivable that one could define the problem of structure discovery precisely enough to allow one to then analyse the statistical significance of the structures so discovered, but we do not have any concrete ideas concerning this" (Ulrike von Luxburg [103]).

In this thesis, we are concerned with the problem of estimating the *statistical significance* of clusters. The aim is to assign a "confidence score" to clusters resulting from a clustering procedure. The confidence score tells how much a cluster deviates from the background of unclustered data. Low statistical significance suggests that the cluster under consideration is just a spurious effect of random density fluctuations in the background. High statistical significance suggests that the cluster *is likely* to reflect the underlying structure of the data. Statistical significance is a necessary

(but still not sufficient) condition for a group of data elements to be considered a cluster.

The statistical significance analysis is not another "optimal", domain-specific clustering algorithm: it is a meta-criterion that can govern clustering across different models of data.

Analogy: the local alignment score significance. The problem of cluster significance can be contrasted with the sequence alignment – a method widely used to detect similarities due to common ancestry in genomic sequences. For a given pair of input sequences, an alignment is a sequence of ordered pairings of their elements. Every pair contributes a score to the alignment, depending on whether the aligned letters are a match or a mismatch. Similarly to the clustering algorithms, the alignment algorithm solves an optimisation problem – in this case finding the highest score-matching for the two input sequences. The algorithm finds a solution even if the sequences are not evolutionary related. As such, the score of an alignment is not meaningful itself, while the statistical significance of this score tells how likely it is to obtain such an alignment by chance.

No sequence comparison is complete without significance estimation: standard computational tools for alignment produce high-scoring alignments together with their significance, and alignments failing stringent significance tests are routinely discarded. The statistics of the gapless local alignment scores are well characterised by extreme value statistics [48], the theory proposed by Karlin and Altschul [57] and extended in the subsequent work [25, 4, 109, 52]. In this thesis, we propose an analogous theory for clustering.

1.2 Genes and gene expression

In this section, we provide a brief introduction to the basic biological concepts related to genetics and gene expression.

Genome. The genome encodes the information that a living organism requires to function and to reproduce. Biochemically, the genome is a sequence of nucleotides: either DNA or RNA, in case of some viruses. The sequence has a very specific architecture with elements of varied functionality. The well studied part are the *coding segments* called *genes*. Each gene provides a formula for a specific *protein*. Proteins are also polymers, which are build of 20 different types of *amino-acids*. Proteins are the functional units of an organism: they are used as structural blocks and are involved in all biological processes.

A genome may have as little as only several genes for a virus, to thousands for bacteria or eukaryotes, such as human. Genes constitute only a small fraction of the genetic

sequence, e.g 2% in humans. Most of the DNA sequence is noncoding and its role is in large part unknown, but evolutionary analysis suggests its functionality [89]. Some parts of the noncoding sequence are better recognised. The *regulatory sequences* act together with specific proteins in the cell, and decide which genes are synthesised into proteins. The protein synthesis is described by the so-called *central dogma of molecular biology*: genes are first *transcribed* into *messenger RNA* (mRNA) which in turn is *translated* into protein. The mRNAs are small molecules, copies of the corresponding DNA coding sequence.

Cell identity: the gene expression profile. *Cell* is the *building block of life* [2]: it is a minimal functional unit of a living organism, that can replicate and synthesise proteins. Some organisms, such as bacteria, are unicellular, another have evolved to form complex forms of many cooperating cells of different types, grouped into tissues. The replication process, in which the new cells are generated from the old ones, is prone to errors, and mutations can occur in the replicated genome sequence. The rate of mutations depends on the reparatory mechanism developed by the organism, and is relatively low for humans and high for viruses. However, the most important and actually "programmed" source of differences between cells is not in what is "written" in the genome, but in what is "read" by the cell. The *regulatory mechanisms* are complex networks of dependencies between genes and proteins. A protein, by binding to the non-coding sequence in the *promoter region* of another gene, may *inhibit* or *enhance* its expression and thus prevent or stimulate synthesis of proteins coded by that gene. A cascade of such programmed interactions is called a *pathway*.

Depending on the content and concentrations of proteins in the cell, different regulatory mechanisms are activated and different sets of genes are transcribed into mRNA. Some genes, responsible for the basic functioning of the cell (metabolism etc.), called the *constitutive genes*, are not regulated, and are always expressed, independently of the cell state or type. Others are expressed only under specific conditions, or in cells from specific tissues, e.g. only in a stem or in a leaf of a plant.

The biological function of a cell is reflected by the *concentration* of the mRNA or protein molecules [18, 27]. Gene regulatory programs in a cell depend not only qualitatively on its presence, but also *quantitatively* on the amount of proteins in the cell. A *gene expression profile* of a cell, i.e. expression levels of all genes, is thus a signature of the cell's identity and state. The gene expression profiles of cells from different tissues are dissimilar. Moreover, within a given cell type the profiles change over time, e.g. during the cell cycle or as a reaction to changing environmental conditions.

1.2.1 Measuring gene expression levels.

DNA microarrays. The development of high-throughput technologies has enabled simultaneous measurements of mRNA concentrations of all genes in a cell. The most

commonly used is the microarray technology. The microarrays are small glass plates divided into "pixels" [27], where typically each pixel corresponds to one gene. In each such pixel, a *probe* (i.e. a short characteristic DNA subsequence of one gene) is placed. The traditional microarrays (which will be analysed in this thesis) are specially designed for a specific species, whose genome sequence is known and whose genes have been annotated. Among exceptions are the *tiling arrays* designed such that probes cover entire genome, not only the coding regions.

In the gene-expression analysis of a given tissue, cell type or organism, the mRNA molecules are extracted from the cells under consideration. The mRNA molecules are reversely transcribed into DNA and the sample is then placed on the microarray. The DNA sequences in the sample will then attach with high affinity (*hybridise*) to their complementary DNA probes. The amount of the sequences bound to each DNA probe is then measured: the array is illuminated with a laser and the intensity of light emanating from a pixel depends on the concentration of the bound DNA sequences. Such fluorescence measurements are then reported as gene expression levels.

The microarray technology has several limitations [74]. First of all, the analysis is restricted only to the sequences present on the array. Another issue is the background: unspecific hybridisation of non-matching DNA transcripts and probes. This effect hinders correct estimation of expression of less abundant mRNA transcripts of lowly expressed genes. Moreover, probe sequences can have varying hybridisation properties [43]. Expression of a gene can be still compared between samples, but this effect has to be accounted for when comparing expression of different genes within the same sample.

RNA-seq [104, 74]. An alternative to the microarray gene expression measurements are the sequencing-based approaches. A population of the target mRNA is first converted into a library of short DNA fragments. The library is then sequenced with one of the high-throughput technologies (e.g. provided by 454 Life Sciences [73] or Illumina [8]). The resulting *reads* are then either aligned to the reference genome or reference transcripts, or they are used to *de novo* reconstruct the transcript sequences [11, 88], together with an estimation of their expression levels. The knowledge of the reference gene sequences is thus not required, which makes this method useful for the analysis of non-model organisms. For that reason, the RNA-seq can be used in population studies to detect differences in the sequence of transcribed genes. RNA-seq has proven to be more sensitive than microarrays and it shows only a relatively low level of background noise.

Proteomics and mass spectrometry [82]. Likewise, there are technological attempts to measure concentrations of translated proteins in cells. Mass spectrometry allows to determine the amounts of protein peptides (parts of digested proteins) in a high-throughput fashion. This technology poses computational challenges related to

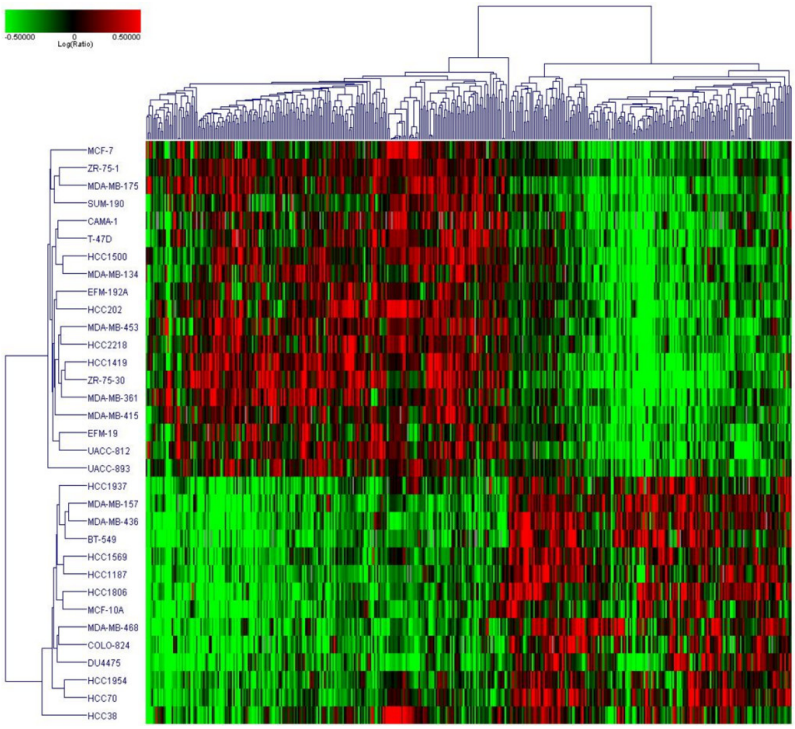

Figure 1.1: Example of clusters and differential expression in gene expression data, reproduced from Finn et al. [33]. The figure presents a heat-map of expression levels of a subset of human genes (in rows) in two types of breast cancer cell lines: sensitive and resistant to growth inhibition by PD 0332991 (in columns). Green stands for low and red stands for high expression level. The clustering partitioned the genes into two groups with differential expression between the two types of cell lines.

the annotation of the signal: the measurement of the concentration of the protein's peptides, and then the identification of related proteins.

1.2.2 Computational analysis of gene expression data

Gene expression in an "intermediate phenotype" of gene activity and so far is less understood than the well-studied processes happening on the sequence level. Gene expression is governed by complex interactions between genes and the experimental measurements suffer from technical noise. An attempt to find structures in such data is thus a first step in understanding the underlying biology. Mutual similarities of many elements are usually related to functional importance. Clustering has been used widely in the analysis of gene expression data to find genes that are active in common biological processes or to find groups of tissues that have similar expression profiles.

Gene co-expression. Genes showing similar expression patterns across a set of experimental conditions are said to be *co-expressed*. Co-expression is usually a signature of a biological interaction: if genes are taking part in the same process, they are "needed" by the cell at the same time and hence they are expressed at the same time. Clusters in gene expression data are thus a natural consequence of pathways and regulatory processes taking place in a cell [30, 81]. In practice, clustering can be used to discover such underlying but yet unknown, or only partially known mechanisms. For example, if several genes in a cluster are known to take part in some process, it is likely that the rest of the genes is also involved [16].

Clustering of samples. A complementary approach seeks clusters of experiments, based on their expression profiles. Such a classification has been especially useful in cancer diagnostics. Cancer cells originate from normal cells but carry a broadly understood *mutation*: a change of possibly genetic type, which results in a dis-regulation and distorted functioning of the cell. The complexity of biological machinery gives many possible ways of such distortions, and there are no two identical tumours. Still, some mechanisms are common to different tissues and they can be detected by finding groups of similar tumour tissues from different patients. An approach combining clustering of genes and experiments is called *biclustering*. This method aims at finding genes co-expressed at a subset of experiments. The biclustering analysis can be used for the task of tumour classification combined with detection of gene-interactions that are specific to the given set of samples [66].

Identifying differentially expressed genes. A related problem concerns identification of *differentially expressed* genes. The setting is partly *supervised*: given two groups of experiments, for example normal and tumour tissues, what are the the pathways or regulatory factors that lead to consistent differences between these profiles [91]? Such changes can involve down- or up-regulation of individual genes, which are then reflected in the change of their expression levels. Another possible effect is turning -off or -on of the entire biological mechanism. With a loss of co-regulation, the involved genes are no longer bound to be co-expressed and the cluster is lost. However, this event does not necessarily lead to an up- or down-regulation of individual genes [59, 21, 93].

1.3 Thesis organisation

The topic of this thesis is the statistics of clusters in high dimensional real datasets. Our motivation was gene expression analysis, which raises simple questions: (i) How to measure similarity of expression patterns? (ii) Given a set of genes, how to decide whether these genes are co-expressed?

Our approach was strongly influenced by solutions from statistical mechanics, which studies large systems of elementary particles. We present the basic concepts and computational methods in Chapter 2.

In Chapter 3, we propose a probabilistic framework to model clusters in data of many high-dimensional vectors. We discuss variants of a similarity measure and of a background distribution for the vectors. We define a cluster scoring function, which in the light of Chapter 2, can be thought of as an energy function of a system of data vectors. This unified framework becomes the "language" used throughout the thesis.

In Chapter 4, we address the problem of the statistical significance of clusters. Given a set of vectors, how likely is it that we observe a similar degree of similarity between random vectors? We present an analytical solution for computing the cluster score p-value. The calculation is based on a mapping to a problem from the statistical mechanics of disordered systems, presented in Chapter 2.

In the second part of the thesis, in Chapter 5, we show that presence of clusters in data has a very strong influence on the estimation of dependencies between the *components of data vectors*: e.g. in application to gene expression, gene-clusters and experiment-clusters have a considerable interplay and should not be treated independently. We propose a mixture-model based inference method, which disentangles the spurious effect of clusters from the true signal of the vector-component dependencies. Such correct estimation of the dependencies is crucial both for clustering of the samples (where it serves as a similarity estimate) and also for clustering of data vectors (where it serves as a metric properly weighing information content of the data vector-components).

In the last part of the thesis, in Chapter 6, we present the *significance-based clustering* algorithm: an algorithm which employs all concepts discussed earlier in the thesis. Using a proper estimation of data component dependencies, the algorithm finds only statistically significant clusters. The algorithm is a simple extension of the method proposed in Chapter 5 and it uses the probabilistic framework proposed in Chapter 3.

Chapter 2

Elements of statistical mechanics

This chapter is an introduction to the basic concepts of statistical mechanics. The presented theory will first be used in Chapter 3, in design of probabilistic models for clusters, and secondly in Chapter 4, in a solution to the problem of the statistical significance of clusters.
We start with presenting a probabilistic description of a physical system. A typical example of a physical system is gas with non-interacting molecules, but the definition is appropriate for any set of a large number of elementary components. A key ingredient of a physical system is an energy function, which maps configurations of the elementary components in a system to an energy value. Statistical mechanics studies the collective behaviour of such components, for different energy functions and with possible interactions between the components. Here, we discuss a class of simple models with independent components and with an additive energy function. We follow with a discussion on physical systems with a *disorder*, which are characterised by an additional set of parameters. Lastly, we discuss the maximum entropy principle and show the relation between the statistical mechanics and the information theory.
This chapter heavily draws from the exposition in the textbook by Marc Mézard and Andrea Montanari [77] and a set of lectures by Jonathan Y. Yedidia [80].

2.1 Physical system

Statistical mechanics is a branch of physics which studies large systems of elementary components. An example of such a system is a gas with many particles, but the concept is quite general and the examples of many interacting entities can be encountered, for example in the fields of economy or biology. In this thesis, we will consider a system of genome-wide measurements of gene expression.

In a statistical mechanics analysis, one aims at providing a *probabilistic description* of the behaviour of the elementary components. But a detailed, deterministic description, with respect to each component, is in most cases not feasible. However, the probabilistic description at the *microscopic level* of the system's components is sufficient to obtain a *deterministic description* at the *macroscopic level*, i.e. to characterise the behaviour of the system. That is because, due to the large size of

the system, the uncertainties about the exact state of each of the components are averaging out, simply by the law of large numbers.

Below we describe components of a *physical system*.

Space of configurations. We consider a system of N components, where each of the components takes a value from some space $\mathcal{X}$. Value $x_i \in \mathcal{X}$ represents the state of the ith system component. Space $\mathcal{X}$ is specific to the system, it provides "coordinates" of a component, e.g. position and momenta of molecules in a gas. The *space of configurations* for a system with N elements is a product of space $\mathcal{X}$ for each of the N components,

$$\mathcal{X}_N = \mathcal{X} \times \ldots \times \mathcal{X} \ . \tag{2.1}$$

A concrete configuration $X = (x_1, \ldots, x_N) \in \mathcal{X}_N$ of the system is called a *state* of the system.

Energy function. Each system state has some specific value of *energy*. Energy is determined by the Hamiltonian (also called the energy function), which is a real-valued function of the system's state, $\mathcal{H} : \mathcal{X}_N \to \mathbb{R}$. If components of the system do not interact, the energy function has a simple additive form,

$$\mathcal{H}(X) = \sum_{i=1}^{N} e_i(x_i) \ , \tag{2.2}$$

for some functions $e_i : \mathcal{X} \to \mathbb{R}$. In a more general case of k-interacting elements, the energy is given by

$$\mathcal{H}(X) = \sum_{i_1,\ldots,i_k} e_{i_1,\ldots,i_k}(x_{i_1}, \ldots, x_{i_k}) \ . \tag{2.3}$$

From now on, we will restrict to a simple case of a system with noninteracting elements. We will also assume that energies of all elements are given by the same function,

$$e(x) = e_1(x) = \ldots = e_N(x), \quad e : \mathcal{X} \to \mathbb{R} \ . \tag{2.4}$$

Entropy. Energy value E can be realised by many different states $X \in \mathcal{X}_N$. The *microcanonical entropy* is a measure of *uncertainty* of a system: once the energy value is known, what is the exact state of the system? For a finite and discrete space of configurations $\mathcal{X}_N$, with discrete energy values, the entropy is simply the *logarithm of the number* of states with energy value equal to E,

$$\Omega(E) \equiv \log\left(|X \in \mathcal{X}_N : \mathcal{H}(X) = E|\right) \ . \tag{2.5}$$

For an uncountable (e.g. continuous) space of configurations, the entropy is defined as the *logarithm of the density* of states with energy E. The entropy is thus related to the *probability* of the state with energy E, given by

$$p(E) = e^{\Omega(E)}/|\mathcal{X}_N| \ , \tag{2.6}$$

for discrete and finite $\mathcal{X}$. For uncountable $\mathcal{X}$, we similarly write the *probability density function*, $p(E) = e^{\Omega(E)}$.

Boltzmann distribution and the partition function. The probability that a system is in state X is given by the *Boltzmann distribution*:

$$p_\beta(X) = \frac{1}{Z(\beta)} e^{-\beta \mathcal{H}(X)} \ . \tag{2.7}$$

Parameter β in terms of physics is understood as the inverse of the *temperature* of the system, $\beta \equiv 1/T$. Function $Z(\beta)$ is called the *partition function*. For fixed β, the partition function plays a role of the normalisation constant and is given by

$$Z(\beta) = \int_{\mathcal{X}_N} e^{-\beta \mathcal{H}(X)} \, dX \ . \tag{2.8}$$

The rationale for formula (2.7), stemming from the maximum entropy principle, will be discussed in detail in section 2.5.

The partition function has a very prominent role in the theory of statistical mechanics. In particular, by computing the partition function of a system, one can also obtain its entropy $\Omega(E)$ (2.5). We discuss this calculation in detail in section 2.3.

The Boltzmann distribution has interesting mathematical properties: depending on the value of the inverse temperature β, it describes very different situations. In the so called *high-temperature limit* with $\beta \to 0$, the distribution (2.7) is flat, with every state being reached with the same probability. Assuming a finite and discrete space of configurations $\mathcal{X}$, the probability of state X is then

$$\lim_{\beta \to 0} p_\beta(X) = \frac{1}{|\mathcal{X}_N|} \ , \tag{2.9}$$

where $|\mathcal{X}_N|$ is the size of configuration space $\mathcal{X}_N$. In particular, in this limit the distribution $p_\beta(X)$ does not depend on the energy function $\mathcal{H}(X)$.

Conversely, in the *low-temperature limit*, the partition function is "dominated" by low energy states; for $\beta \to \infty$ only the lowest energy states contribute. These states are called the *ground states*. We denote the set of ground states by $\mathcal{X}_0 \subseteq \mathcal{X}_N$. The probability distribution in the limit $\beta \to \infty$ is

$$p_\beta(X) = \begin{cases} \frac{1}{|\mathcal{X}_0|} & X \in \mathcal{X}_0 \ , \\ 0 & X \notin \mathcal{X}_0 \ . \end{cases} \tag{2.10}$$

Free energy. The free energy is a so-called *thermodynamic potential*, a scalar function of the inverse temperature β and the system state's energy $E(X)$. It is defined by means of the partition function,

$$F(\beta) = -\frac{1}{\beta} \log Z(\beta) \ . \tag{2.11}$$

The free energy has an important property: by taking derivatives over β, one can compute the expected energy of a system. The calculation follows as

$$\frac{\partial}{\partial \beta}(\beta F(\beta)) = \int_{\mathcal{X}_N} \mathcal{H}(X) e^{\beta \mathcal{H}(X)} dX = \mathbb{E}[\mathcal{H}(X)] \ . \tag{2.12}$$

The thermodynamic limit. Statistical mechanics studies systems of elementary components with the aim of characterising their behaviour at the macroscopic level. The number N of elementary components is typically large and the fluctuations in such systems average out in the *thermodynamic limit* of $N \to \infty$. Among exceptions are systems with strong interactions between its components, where the thermodynamic limit need not exist.

We refer to the quantities that scale with the system's size N, as *extensive*. The additive energy $\mathcal{H}(X)$ (2.2) is an example of an extensive quantity. Conversely, *intensive* quantities do not scale with the system size, for example the energy of a single system component, $e(x)$ (2.4). We will use a notion of the *intensive free energy*,

$$f(\beta) \equiv \lim_{N \to \infty} F(\beta)/N = \lim_{N \to \infty} -\frac{1}{\beta} \log Z(\beta)/N \ , \tag{2.13}$$

and the *intensive entropy* (of the intensive energy $e = E/N$),

$$\omega(e) \equiv \lim_{N \to \infty} \Omega(E)/N \ . \tag{2.14}$$

2.2 Mathematical tools

In this section, we discuss mathematical techniques and concepts which are crucial for understanding of the following material in this chapter.

2.2.1 Dirac-delta function

The Dirac-delta function, $\delta : \mathbb{R} \to \mathbb{R}$, is a construction which can be used for counting configurations with a given value of some property, for example the system states with energy value E. Despite its name, the Dirac-delta function is not a function in a strict sense. It is heuristically defined by two properties:

1. it has unit area,
$$\int_{-\infty}^{+\infty} \delta(x)dx = 1 \ , \tag{2.15}$$

2. it is infinitely peaked at the origin and is zero elsewhere,
$$\delta(x) = \begin{cases} +\infty & x = 0 \\ 0 & \text{otherwise} \ . \end{cases} \tag{2.16}$$

The Dirac-delta function can be rigorously defined as a probability measure, characterised by a cumulative distribution function
$$F(x) = \begin{cases} 1 & x \geq 0 \\ 0 & 0 \ . \end{cases} \tag{2.17}$$

Integral representation. The Dirac-delta function has a useful integral representation which can be derived using a Fourier transformation. A Fourier transform of a function $f(x)$ is
$$\tilde{f}(s) \equiv \int_{-\infty}^{\infty} \frac{e^{ixs}}{\sqrt{2\pi}} f(x)dx \ . \tag{2.18}$$
The Fourier transform is reversible, function $f(x)$ can be expressed in terms of its transform $\tilde{f}(s)$ as
$$f(x) = \int_{-\infty}^{+\infty} \frac{e^{-ixs}}{\sqrt{2\pi}} \tilde{f}(s)ds \ . \tag{2.19}$$
Now, the Fourier transform of the delta function is
$$\tilde{\delta}(s) = \int_{-\infty}^{\infty} \frac{e^{ixs}}{\sqrt{2\pi}} \delta(x)dx = \frac{1}{\sqrt{2\pi}} \ , \tag{2.20}$$
as the delta function is zero everywhere apart from $x = 0$. Inserting this result into (2.19), we obtain the integral representation:
$$\delta(x) = \int_{-\infty}^{+\infty} \frac{e^{-ixs}}{2\pi} ds \ . \tag{2.21}$$

2.2.2 Saddle-point approximation

The saddle-point approximation of an integral, also known as the method of steepest descent or the Laplace method, is a method for extracting asymptotic behaviour of a class of definite integrals of exponential functions. The basic idea is as follows: for large values of N, the value of an integral of the form
$$\int_a^b dx \ e^{Nf(x)} \ , \tag{2.22}$$

(with $f(x)$ of order one) is dominated by a narrow part of the integration domain around the maximum x_0 of $f(x)$, the so-called *saddle-point*. A Taylor expansion to second order around that maximum (assumed to be unique and to lie on the interval $a < x_0 < b$) gives a Gaussian integral

$$\begin{aligned} \frac{1}{N}\log\int_a^b dx\ e^{Nf(x)} &\approx \frac{1}{N}\log\int_a^b dx\ e^{Nf(x_0)-Nf(x-x_0)^2/2} \\ &= f(x_0) + 1/(2N)\log\left(\frac{2\pi}{Ng}\right)\ . \end{aligned} \tag{2.23}$$

The coefficient of the linear term of the Taylor expansion is zero at the maximum x_0, and $g \equiv |\partial^2 f/\partial x^2|_{x_0}$ denotes the absolute value of the second derivative of $f(x)$ evaluated at x_0. For large values of N, the saddle-point integral is thus dominated by the maximum $f(x_0)$, with the second-order term of the Taylor expansion yielding a contribution which vanishes relative to the dominant zeroth-order term as $(\log N)/N$. The corresponding terms of higher order in $(x-x_0)$ of the Taylor expansion, sometimes referred to a the *finite size correction*, similarly give relative contributions to the integral which vanish for large N.

2.2.3 Legendre transform

The Legendre transform of a real valued and differentiable function is an operation which gives a new, *dual* function f^*. The idea behind the transformation is that information about a functional relation, $(x_0, f(x_0))$, can be be equivalently expressed by another set of points of the form $(f'(x_0), p_0)$, where p_0 is an intercept of the line tangent to $f(x)$ at point x_0 and $f'(x) \equiv \partial f(x)/\partial x$ is the derivative of function $f(x)$ over x, see Fig. 2.1 for an illustration. The Legendre transform is formally defined as

$$f^*(y) = \sup_x [xy - f(x)]\ . \tag{2.24}$$

To find a supremum of $(xy - f(x))$ with respect to x, we solve

$$\frac{\partial}{\partial x}(xy - f(x)) = 0\ , \tag{2.25}$$

which is met by $y = \frac{\partial}{\partial x} f(x) = f'(x)$. The intercept of the tangent to function $f(x)$ at x_0 is then $f^*(f'(x_0))$, so the point $(x_0, f(x_0))$ is now mapped to a point $(f'(x_0), f^*(f'(f(x_0))))$.

An important property of the Legendre transform is its duality: function f is also a Legendre transform of f^*,

$$f(x) = \sup_y \left[xy - f^*(y)\right]\ . \tag{2.26}$$

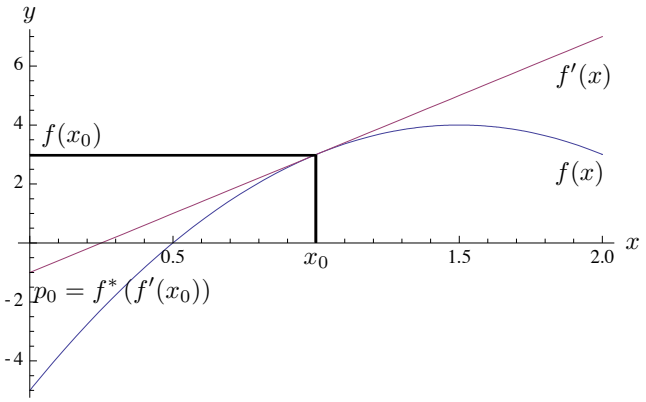

Figure 2.1: Legendre transform of a function. Function $f(x)$ (blue line) can be described by a set of points of the form $(x_0, f(x_0))$. A dual representation is achieved by using a function tangent to $f(x)$ at x_0, (red line). The new set of points is of the form $(f'(x_0), f^*(f'(x_0)))$, where $f^*(y)$ is the Legendre transform of $f(x)$ and point $f^*(f'(x_0))$ is an intercept of the tangent with y-axis.

As we will show later in this chapter, the intensive entropy and the intensive free energy of a system are in such a dual relation,

$$\omega(e) = \sup_{\beta} \left[\beta e - \beta f(\beta)\right] \ . \tag{2.27}$$

2.3 Energy distribution

Using properties of the Dirac-delta function, we can compute the probability that a system is in a state with energy E:

$$\begin{aligned}
p(E) &= \int_{\mathcal{X}_N} \delta\left(E - \mathcal{H}(X)\right) dX && (2.28)\\
&= \int_{(-\infty,0)}^{(+\infty,0)} \frac{1}{2\pi} e^{i\beta E} \left[\int_{\mathcal{X}_N} e^{-i\beta\mathcal{H}(X)}\, dX\right] d\beta && (2.29)\\
&= \int_{(0,-\infty)}^{(0,\infty)} \frac{1}{i2\pi} e^{\beta E} \left[\int_{\mathcal{X}_N} e^{-\beta\mathcal{H}(X)}\, dX\right] d\beta && (2.30)\\
&= -\frac{i}{2\pi} \int_{(0,\infty)}^{(0,\infty)} e^{\beta E + \log Z(\beta)} d\beta && (2.31)\\
&= \frac{1}{2\pi} \int_{-\infty}^{+\infty} e^{\beta E + \log Z(\beta)} d\beta \ . && (2.32)
\end{aligned}$$

In line (2.28), we integrate over all states of the system with the Dirac-delta function *collecting* only the states with energy E. In line (2.29), we expand the delta function to its integral representation with variable β. The integrand here is complex and we

denote the integration interval by specifying both the real and the imaginary part. The integral contour is along the real axis. In line (2.30), we perform a change of variable, $\beta \leftarrow i\beta$, which also involves changing of the integration contour to the imaginary axis. The term in squared brackets is equal to the partition function $Z(\beta)$ (see Eq. (2.8)) which we write in line (2.31). According to the rules of complex integration, the integral over β becomes a real integral in line (2.32).

Computation of integral (2.32) depends on the specific physical system and its energy function. Assuming that N is large and that the system components are independent and identically distributed, we get

$$Z(\beta) = \left[\int_{\mathcal{X}} e^{-\beta e(x)} dx\right]^N = e^{-N\beta f(\beta)} \ , \tag{2.33}$$

where $e(x)$ is the intensive energy and $f(\beta)$ is the intensive free energy function (2.13). Inserting (2.33) in Eq. (2.32), and taking $E = Ne$,

$$p(E) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} e^{N[\beta e - \beta f(\beta)]} d\beta \ . \tag{2.34}$$

The number of elements N is a large constant and the integral in (2.34) can be solved with a saddle-point approximation:

$$\log p(E) \simeq N\left[\beta_0 e - \beta_0 f(\beta_0)\right] + \frac{1}{2}\log\left(\frac{2\pi}{Ng}\right) \ , \tag{2.35}$$

where β_0 is the saddle-point of the function in the exponent. The second term on the right-hand side of Eq. (2.35) is the finite size correction and is explained in the definition of the saddle-point approximation in Eq. (2.23). In terms of physics, β_0 is the temperature in which the system has most typically energy E. The term in the square brackets, which gives the *exponential* and hence dominating contribution to the probability density, is the intensive entropy (2.14),

$$\omega(e) = \max_{\beta}[\beta e - \beta f(\beta)] \ . \tag{2.36}$$

Derivation with a Legendre transform. The same result can be derived based on the saddle-point approximation and the Legendre transform. Let us again consider the partition function

$$Z(\beta) = \int_{\mathcal{X}_N} e^{-\beta \mathcal{H}(X)} dX = \int p(E) e^{-\beta E} dE \ . \tag{2.37}$$

The second step collects all configurations of vectors X with energy E, so $p(E)$ denotes the density of states as a function of energy E. Replacing the extensive energy with the intensive quantity, $E = Ne$, and using $p(E) = \frac{1}{N}p(e)$, we get

$$\int p(E) e^{-\beta E} dE = \frac{1}{N} \int e^{-N\beta e + \log p(e)} de \simeq \frac{1}{N} e^{N \sup_e (\log p(e)/N - \beta e)} \ , \tag{2.38}$$

where in the third step, assuming N is large, we used the saddle-point approximation. We now have

$$\log Z(\beta)/N = \sup_e \left[\log p(e)/N - \beta e\right] \ , \tag{2.39}$$

i.e. the normalised logarithm of the partition function, $\log Z(\beta)/N \equiv -\beta f(\beta)$ is a Legendre transform (see Section 2.2.3) of the normalised logarithm of the probability, $\log p(e)/N$. Exploiting the duality of the Legendre transform (see Eq. (2.26)), we get

$$\begin{aligned} \log p(e) &\simeq -N \sup_\beta \left[\beta f(\beta) + \beta e\right] &(2.40)\\ &= N\left[\beta_0 e - \beta_0 f(\beta_0)\right] \ , &(2.41) \end{aligned}$$

with β_0 the saddle-point of the function in the squared brackets. What follows,

$$\log p(E) = \log p(e) + \log\left(\frac{1}{N}\right) \simeq N\left[\beta_0 e - \beta_0 f(\beta_0)\right] + \log\left(\frac{1}{N}\right) \ , \tag{2.42}$$

which is identical to the result in Eq. (2.35), up to the finite size correction term.

2.4 Disordered systems

So far we assumed that an energy of a physical system depends solely on a state of the system. We now consider a class of problems in which a system is described by an additional set of parameters. These parameters are set *independently* of the system components and are *fixed* for a given realisation of a system. They are called a *quenched disorder* of a system, relating to spin glasses, where the term was coined. Spin glasses consist of atoms (possessing magnetic moment, i.e. spin) distributed randomly in a solid, the solid also consists of atoms (which are non-magnetic). The magnetic couplings between different spins are predefined and fixed. The spins then evolve in such a fixed realisation of the energy landscape induced by the couplings. The fixed magnetic couplings are the disorder, which is called "quenched" (frozen) in analogy with the rapid cooling of a metal in the forging process [79]. The disordered systems have also been considered in the context of formal models of neural networks [49, 31].

Many optimisation problems can be mapped onto a disordered system from statistical mechanics. An interesting example from the field of computational biology is a local alignment of two sequences [52, 53]. Here, the quenched disorder consists of two parameters, which are the instances of two sequences to be aligned, $\mathbf{a} = [a_1, a_2, \ldots, a_l]$ and $\mathbf{b} = [b_1, b_2, \ldots, b_k]$. These sequences are fixed, while the optimal alignment is formed. The physical system is an *alignment path*, an ordered set of pairings of the sequences' letters, $\mathbf{X} = [\{a_{i_1}, b_{j_1}\}, \ldots, \{a_{i_n}, b_{j_n}\}]$, where $1 \leq i_1 \leq \ldots \leq i_n \leq l$, $1 \leq j_1 \leq \ldots \leq j_n \leq k$. The space of configurations is the set of all possible alignment paths. The Hamiltonian $\mathcal{H}(X|\mathbf{a}, \mathbf{b})$ is given by the cost function of an

alignment, which most commonly is defined as a function of the number of matches and mismatches of letters in the alignment pairings.

Let us denote the disorder by a set of parameters $\Theta \in \mathcal{P}$, where $\mathcal{P}$ is some space of parameters' values. The energy of a system state X depends on parameters Θ,

$$\mathcal{H}(X|\Theta) = \sum_{i=1}^{N} e(x_i|\Theta) \ . \tag{2.43}$$

We now want to compute the distribution of the number of states with a given energy in such a system, i.e. the microcanonical entropy. Computation of the entropy involves computation of the free energy function, see Eq. (2.35). The disorder Θ will be averaged out: coming back to the sequence alignment example, averaging over the disorder means that we are interested in the statistics of alignment scores (energies) of arbitrary sequences, not just some given instance of two sequences $\Theta = (\mathbf{a}, \mathbf{b})$.

We thus want to compute the so called *annealed average* of the free energy,

$$\langle\langle F(\beta) \rangle\rangle = -\frac{1}{\beta} \langle\langle \log Z(\beta) \rangle\rangle = \int_{\mathcal{P}} \left[-\frac{1}{\beta} \log Z(\beta|\Theta) \right] d\Theta \ , \tag{2.44}$$

where $\langle\langle (.) \rangle\rangle = \int_{\mathcal{P}} (.) d\Theta$ is a shorthand for the integral over Θ. The average of a logarithm of the partition function, as in the last step in line (2.44), is in a general case difficult to compute. There have been different heuristic or approximation methods proposed for computing the free energy in an disordered system: the variational approach [65], the cavity method [78], and the replica-trick [17, 31, 41, 79]. We discuss the latter in the following section.

2.4.1 The replica-trick

The idea behind the replica-trick is to represent a disordered system with an equivalent system for which computation of the free energy function is easier. The basis of the method is an algebraic identity

$$\log Z = \lim_{n \to 0} \frac{1}{n} \left(Z^n - 1 \right) \ . \tag{2.45}$$

Instead of computing the average $\langle\langle \log Z(\beta|\Theta) \rangle\rangle$ in Eq. (2.44), one can perform another computation of $\langle\langle Z(\beta|\Theta)^n \rangle\rangle$. The crux of the heuristic is to first assume that n is integer, compute Z^n, and then compute the limit for $n \to 0$. The first step is

$$\begin{aligned} \langle\langle Z(\beta|\Theta)^n \rangle\rangle &= \int_{\mathcal{P}} \left[\int_{\mathcal{X}^N} P(X) e^{-\beta \mathcal{H}(X|\Theta)} dX \right]^n d\Theta \\ &= \prod_{a=1}^{n} \left[\int_{\mathcal{X}^N} P(X_a) \left(\int_{\mathcal{P}} e^{-\beta \mathcal{H}(X_a|\Theta)} d\Theta \right) dX_a \right] \ . \end{aligned} \tag{2.46}$$

In line (2.46), the system is, heuristically, represented as a set of n independent systems with identical parameters. Thus, $Z^n(\beta|\Theta)$ represents a partition function of an n-times *replicated* system with identical parameters. Each replica of the system is indexed with a, $a = 1, \ldots, n$.

The integration order, over the disorder Θ and possible configurations X of the system, can now be inverted and the disorder Θ can be integrated out. We obtain a partition function of a "standard" physical system: it is more complex as a set of n replicas, but it again depends on a single set of variables, $X = \{X_1, \ldots, X_n\}$, where each $X_a = \{x_{1,a}, \ldots, x_{N,a}\}$.

Intuitively, the replica trick can be understood as follows: the energy landscape of the original energy function depends on "couplings" between system variables X and the disorder Θ. After replicating the system n times and averaging over the disorder, we replace the couplings of variables X with a disorder Θ by pairwise, similarly binding, couplings between replicated variables X_a. The new energy function does no longer depend on the disorder, but it is more complex, as it now involves interactions between system components. One of the key features is its dependence on n, the number of replicas. In the last step of the method, the constraint on n being integer is dropped, and the limit for real $n \to 0$ is computed by analytic continuation.

The replica-trick is clearly a heuristic method and the last step of performing the limit $n \to 0$ involves mathematical subtleties [79]. Nevertheless, the method has been successfully applied in solutions for various physical systems, which could later be confirmed rigorously. An example is the Sherrington-Kirkpatrick model for spin glasses, where system variables are spins denoted by S_i and the disorder are pairwise couplings between corresponding spins represented by variables J_{ij}: a positive coupling is ferromagnetic and a negative if a coupling is antiferromagnetic. The couplings can be observed between any pair of spins, not necessarily neighbouring ones. The Hamiltonian is given by $\mathcal{H}(S_1, \ldots, S_N) = -\sum_{i>j} J_{ij} S_i S_j$. The replica solution for this system was provided by Parisi [83] and it was later proven by Guerra and Talagrand [102].

We will show an example of a full computation employing a replica-trick, for the problem of the maximum cluster score, in Chapter 4 and in Appendix C.

2.5 Relation to information theory: the maximum entropy principle

Statistical mechanics can be regarded as a form of statistical inference and its computational rules are a consequence of the maximum-entropy principle from the field of information theory [54].

Information entropy. The key in the information theory is a measure of *uncertainty* associated with the value of a random variable. The *information entropy* [90] was designed to have the following properties: (i) a random variable which follows a broad probability distribution reveals more uncertainty than a one which follows a strongly peaked distribution; (ii) the uncertainty measure should be additive with respect to independent sources introducing uncertainty. Assume $\mathbf{x}$ is a discrete random variable taking values $\{x_1, \ldots, x_n\}$, according to a probability distribution $p(x_i)$. A function fulfilling the above requirements is given by

$$\Omega_I(X) = -\sum_{i=1}^{n} p(x_i) \log p(x_i) \ . \tag{2.47}$$

An extension of the information entropy measure for continuous random variables was proposed by E.T. Jaynes [54]. The continuous solution has the same properties as the discrete case and for the ease of notation, we will restrain to the discrete case in the following presentation.

The information entropy $\Omega_I(X)$ (2.47) appears to be related to the microcanonical entropy $\Omega(E)$ (2.5) (see section 2.1). Assuming for now that E is discrete and follows distribution $p(E)$ (2.6), we compute the information entropy for the energy in a physical system:

$$\Omega_I(E) \equiv -\sum_i p(E_i) \log p(E_i) = \mathbb{E} \log \left[\frac{1}{p(E_i)} \right] = \log |\mathcal{X}| - \mathbb{E}[\Omega(E)] \ . \tag{2.48}$$

In the third step we used $p(E) = e^{\Omega(E)}/|\mathcal{X}|$ from Eq. (2.6). Apart from the trivial additive shift $\log |\mathcal{X}|$ and a sign difference, the main difference of these definitions is that the microcanonical entropy is the logarithm of the number of configurations at a *given value of energy*, while the information entropy is the *expected value* of the negative logarithm of the number of configurations over *all energy values*.

The maximum entropy principle. Assume that we are given a random variable $\mathbf{x}$ taking values $\{x_1, \ldots, x_n\}$, with an unknown probability distribution $p(x_i)$. Additionally, we are also given a prior information about the random variable: the expected value a of some property, here described by function $f(x_i)$,

$$a = \mathbb{E}\left[f(x_i)\right] = \sum_{i=1}^{N} p(x_i) f(x_i) \ . \tag{2.49}$$

The question is: what is the unbiased inference about the distribution $p(x_i)$? In other words, what is the distribution which does not reduce the amount of uncertainty about the random variable?

The *maximum entropy principle* states that the probability distribution should maximise the information entropy subject to the prior knowledge about the random variable. If there is no prior information, the solution is, quite intuitively, a uniform

distribution assigning the same probability to every value of the random variable. In the presence of a constraint from Eq. (2.49) and given the normalisation constraint,

$$\sum_i p(x_i) = 1 \ , \tag{2.50}$$

we can infer distribution $p(x_i)$ using the Lagrange multipliers. The solution is

$$p(x_i) = e^{-\lambda - \beta f(x_i)} \ , \tag{2.51}$$

where constants λ and β are inferred such that (2.49) and (2.50) are met. The solution can be written in an equivalent form as

$$p(x_i) = e^{-\beta f(x_i)} / Z(\beta) \ , \tag{2.52}$$

where

$$\log Z(\beta) = \lambda \ , \tag{2.53}$$

$$Z(\beta) = \sum_i e^{-\beta f(x_i)} \tag{2.54}$$

and importantly

$$-\frac{\partial}{\partial \beta} \log Z(\beta) = \sum_i e^{-\beta f(x_i)} / Z(\beta) f(x_i) = \mathbb{E}\left[f(x_i)\right] = a \ . \tag{2.55}$$

Substituting x for X, a state of a physical system, and $f(x)$ for $\mathcal{H}(X)$, the Hamiltonian of the system, we obtain the Boltzmann distribution (2.7). This shows that the Boltzmann distribution is the maximum entropy distribution for a system with a given observed energy value.

Chapter 3

Statistical theory of clusters

In this chapter, we discuss probabilistic models for clusters in a high-dimensional real space. Cluster is a group of *similar* data vectors, which *deviate* from a background constituted of identically and independently distributed vectors. First, we propose a general framework for scoring clusters in data. We consider three properties defining a cluster: (i) *point density* – cluster as a dense agglomeration of points is some region in the data space; (ii) *positional bias* – cluster characterised by atypical location of data vectors; (iii) *directional density* – cluster as a dense agglomeration of vectors pointing in a similar direction in the data space. These properties are related to different choices of a similarity measure and of the background distribution. We arrive at a classification of clustering schemes for different combinations of considered cluster properties.

3.1 General setting

Let us consider data vectors in M-dimensional space, $\mathbf{x}_1, \ldots, \mathbf{x}_N \in \mathbb{R}^M$. The vectors are *independently* distributed according to some *background distribution* $P_0(\mathbf{x})$: in this sense, the background distribution describes a typical behaviour of data elements. The background distribution $P_0(\mathbf{x})$ is not unique, it depends on the type of data. For example, gene expression is often modelled with the Gaussian distribution, while for the RNA-seq count data, the Poisson or the negative binomial distribution are commonly used [40].

The background distribution $P_0(\mathbf{x})$ is contrasted with an alternative hypothesis: vector $\mathbf{x}$ being part of a *cluster*, a group of vectors distinguished by *enhanced mutual similarity*. We denote the alternative, so-called "cluster" probability distribution of such a vector $\mathbf{x}$ by $Q(\mathbf{x}|\theta)$, where θ is a set of the cluster model-specific parameters.

To be more precise, we focus on a class of cluster models that describe convex and spherical clusters, defined by two properties:

1. *cluster centre* vector $\mathbf{z} \in \mathbb{R}^M$,

2. *expected similarity* of cluster elements to the cluster centre $\mathbf{z}$, for some similarity measure $\mathrm{sim}(\mathbf{x}, \mathbf{z})$,

$$a = \mathbb{E}\left[\mathrm{sim}(\mathbf{x}, \mathbf{z})\right] = \int_{\mathbb{R}^M} \mathrm{sim}(\mathbf{x}, \mathbf{z}) Q(\mathbf{x}|\mathbf{z}) d\mathbf{x} \ , \tag{3.1}$$

where $Q(\mathbf{x}|\mathbf{z})$ is a distribution of vectors in a cluster with centre $\mathbf{z}$. Intuitively, a defines "width" of the cluster.

Using the maximum entropy principle [54], discussed in Chapter 3, we obtain a statistically unbiased distribution fulfilling constraint (3.1),

$$Q(\mathbf{x}|\mathbf{z}, \eta) = \frac{1}{Z_\eta} P_0(\mathbf{x}) e^{\eta \, \mathrm{sim}(\mathbf{x}, \mathbf{z})} \ . \tag{3.2}$$

The normalisation constant Z_η depends on the value of the *scoring parameter* η as described by Eq. (2.53) and (2.54). Parameter η is in a one-to-one relation with the value of a, the expected similarity $\mathrm{sim}(\mathbf{x}, \mathbf{z})$ of vectors following distribution $Q(\mathbf{x}|\mathbf{z}, \eta)$. This relation is, following Eq. (2.55),

$$\frac{\partial}{\partial \eta} \log Z_\eta = \mathbb{E}\left[\mathrm{sim}(\mathbf{x}, \mathbf{z})\right] = a \ . \tag{3.3}$$

In other words, parameter η equally determines the cluster "width" as the corresponding constant a does. Intuitively, the larger the value of η, the smaller the expected width of the cluster. We will thus relate to η as the *width parameter*. Note that for $\eta = 0$, the cluster model $Q(\mathbf{x}|\mathbf{z}, \eta)$ is the same as the background model $P_0(\mathbf{x})$.

Log-likelihood score. The deviations of the cluster distribution from the null model define the *log-likelihood score*, which takes the simple form

$$s(\mathbf{x}|\mathbf{z}, \eta) \equiv \log \frac{Q(\mathbf{x}|\mathbf{z}, \eta)}{P_0(\mathbf{x})} = \eta \, \mathrm{sim}(\mathbf{x}, \mathbf{z}) - \log Z_\eta \ . \tag{3.4}$$

By construction, the log-likelihood score assigns positive score values to vectors which are more likely to be in a cluster with centre $\mathbf{z}$ and scoring parameter η, than in the background. The exact form of the scoring function depends on the similarity measure $\mathrm{sim}(\mathbf{x}, \mathbf{z})$ and, via the normalisation constant Z_η, on the background model $P_0(\mathbf{x})$. In the following sections, we discuss several choices of the similarity measure and the background model, and we show resulting scoring functions.

Cluster score. In a set of data vectors $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, for a given vector $\mathbf{z}$ and a scoring parameter η, a *cluster* is a subset of all vectors $\mathbf{x}_i$ with positive score $s(\mathbf{x}_i|\mathbf{z}, \eta)$. The *cluster score* is the sum of the scores of the cluster elements,

$$S(\mathbf{x}_1, \ldots, \mathbf{x}_N|\mathbf{z}, \eta) = \sum_{i=1}^{N} \max[s(\mathbf{x}_i|\mathbf{z}, \eta), 0] \ . \tag{3.5}$$

The cluster score is determined both by the number of elements and by their similarities with the cluster centre, that is, tighter clusters with fewer elements can have comparable scores to looser but larger clusters.

Translational- and rotational invariance of the cluster score. We say that score is *translationally invariant* if a common shift of vectors $\mathbf{x}_i$ and $\mathbf{z}$ does not change its value: $\forall \mathbf{a} \in \mathbb{R}^M$ we have $S(\mathbf{x}_1 \ldots, \mathbf{x}_N|\mathbf{z}, \eta) = S(\mathbf{x}_1 + \mathbf{a}, \ldots, \mathbf{x}_N + \mathbf{a}|\mathbf{z} + \mathbf{a}, \eta)$.

The score is *rotationally invariant* if a common rotation of vectors $\mathbf{x}_i$ and $\mathbf{z}$ by the same angle α does not change its value: $\forall \alpha \in [0, 2\pi]$ we have $S(\mathbf{x}_1 \ldots, \mathbf{x}_N|\mathbf{z}, \eta) = S(r(\mathbf{x}_1, \alpha), \ldots, t(\mathbf{x}_N)|r(\mathbf{z}, \alpha), \eta)$, where $r(\mathbf{x}, \alpha)$ is a rotation operation.

3.2 Clusters based on point density and positional bias

We will now turn to concrete examples of the similarity measure and the background distribution. Our choices will be motivated by characteristics of gene expression data, especially its high-dimensionality.

The choice of the background distribution requires specification of typical properties of non-clustered vectors. We are concerned with vectors in a high-dimensional real space, $\mathbf{x} \in \mathbb{R}^M$. We assume for now that each data component x^μ, $\mu = 1, \ldots, M$, is independently drawn from the same probability distribution $p_0(x^\mu)$ (we will discuss a more general case of dependent variables in Chapter 5). Further, we assume that $p_0(x^\mu)$ has finite moments and without loss of generality, we set the first moment (the mean) to 0 and the second (the variance) to 1, $\sigma_0^2 = 1$. Note that here we introduced prior expectations about the background distribution: there exists a typical data component value x^μ and a finite deviation from that value. The statistically unbiased distribution fulfilling the two constraints, is, by the maximum entropy principle, a Gaussian distribution with mean 0 and variance 1. We can now write the background distribution for a data vector $\mathbf{x}$,

$$P_0(\mathbf{x}) = \prod_{\mu=1}^{M} p_0(x^\mu) = \frac{1}{Z_0} e^{-\frac{1}{2}\mathbf{x}\cdot\mathbf{x}} , \tag{3.6}$$

where Z_0 is a normalisation constant, $Z_0 = (2\pi)^{-M/2}$. The Gaussian distribution is a common choice to model distribution of the relative gene expression levels from microarray experiments [7, 84], see also Fig. 3.1.

The Euclidean distance between two vectors "compares" both their directions and lengths. We consider the Euclidean distance-based similarity measure,

$$\mathrm{sim}(\mathbf{x}, \mathbf{z}) = -\frac{1}{2}\|\mathbf{x} - \mathbf{z}\|^2 = -\frac{1}{2}(\mathbf{x} - \mathbf{z}) \cdot (\mathbf{x} - \mathbf{z}) . \tag{3.7}$$

Given the background distribution (3.6) and similarity (3.7), we write the cluster model,

$$Q(\mathbf{x}|\mathbf{x}, \eta) = \frac{1}{Z_\eta} P_0(\mathbf{x}) e^{-\frac{\eta}{2}(\mathbf{x}-\mathbf{z})\cdot(\mathbf{x}-\mathbf{z})} , \tag{3.8}$$

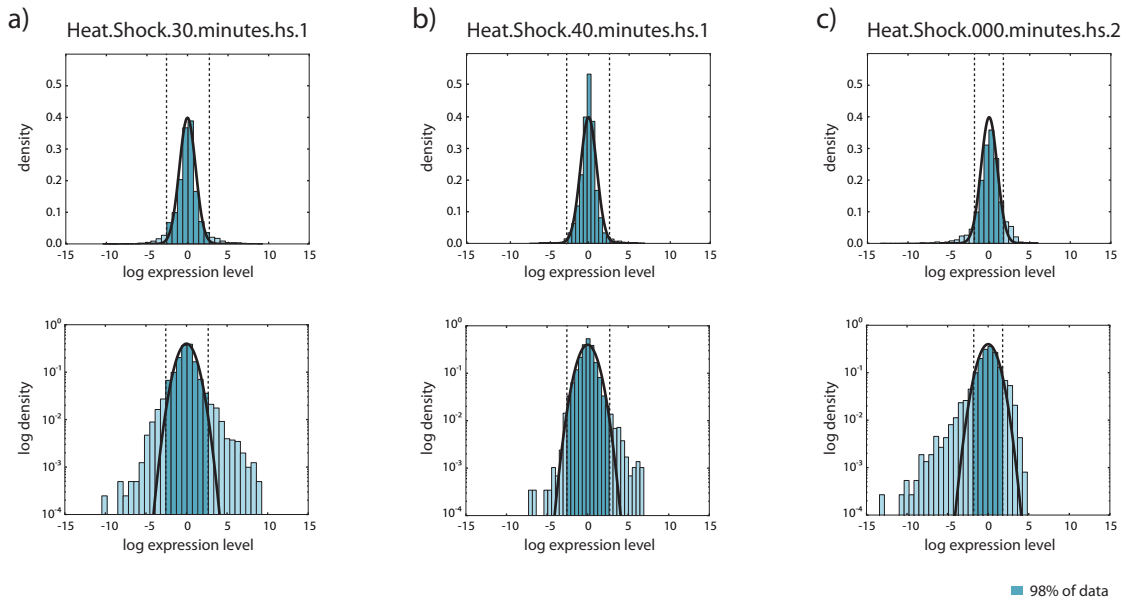

Figure 3.1: Expression levels in microarray experiments follow a Gaussian distribution. Gene expression data from yeast [42] was log-transformed and gene- and experiment-wise centred. The diagrams show the distribution of such relative log-expression levels in three experiments, both on a normal (top) and semi-logarithmic (bottom) scale. The black solid line shows the standard Gaussian curve. The main bulk of the data (98%), coloured with darker blue, shows a good agreement with the Gaussian curve. The data also contains outliers, in both tails of the distribution, which are not in agreement with the Gaussian background model.

where the normalising factor is $Z_\eta = (\eta + 1)^{M/2}$. According to this model, cluster is a spherical agglomeration of vectors, centred around the cluster centre $\mathbf{z}$ with a variance anti-proportional to the value of η.

The log-likelihood score (3.4) is for this specific case given by

$$s(\mathbf{x}|\mathbf{z}, \eta) = -\frac{\eta}{2}(\mathbf{x} - \mathbf{z}) \cdot (\mathbf{x} - \mathbf{z}) + \frac{1}{2}\mathbf{x} \cdot \mathbf{x} - \frac{M}{2} \log(\eta + 1) \ . \tag{3.9}$$

The score is a *quadratic* function of $\mathbf{x}$. The first and the second term of the scoring function (3.9) correspond to two "strategies" for increasing the score: (i) *point density* – via term $-\frac{\eta}{2}(\mathbf{x} - \mathbf{z})(\mathbf{x} - \mathbf{z})$, the score increases with a decreasing distance of vector $\mathbf{x}$ to cluster centre $\mathbf{z}$; (ii) *positional bias* – via term $\frac{1}{2}\mathbf{x} \cdot \mathbf{x}$, the score increases with the length of vector $\mathbf{x}$ and its distance to the mean $\mathbf{0}$ of the background model. Note that in extreme cases, this property of the scoring scheme can result in a high scoring, one-element cluster formed by a long outlier vector.

As a consequence of the positional bias property, the total cluster score $S(\mathbf{x}_1, \ldots, \mathbf{x}_N|\mathbf{z}, \eta)$ is not translationally invariant: clusters located far from $\mathbf{0}$ are scored higher than clusters located in the regions densely populated by elements from the background. The score is rotationally invariant. We illustrate this scoring scheme on examples in Figure 3.2.

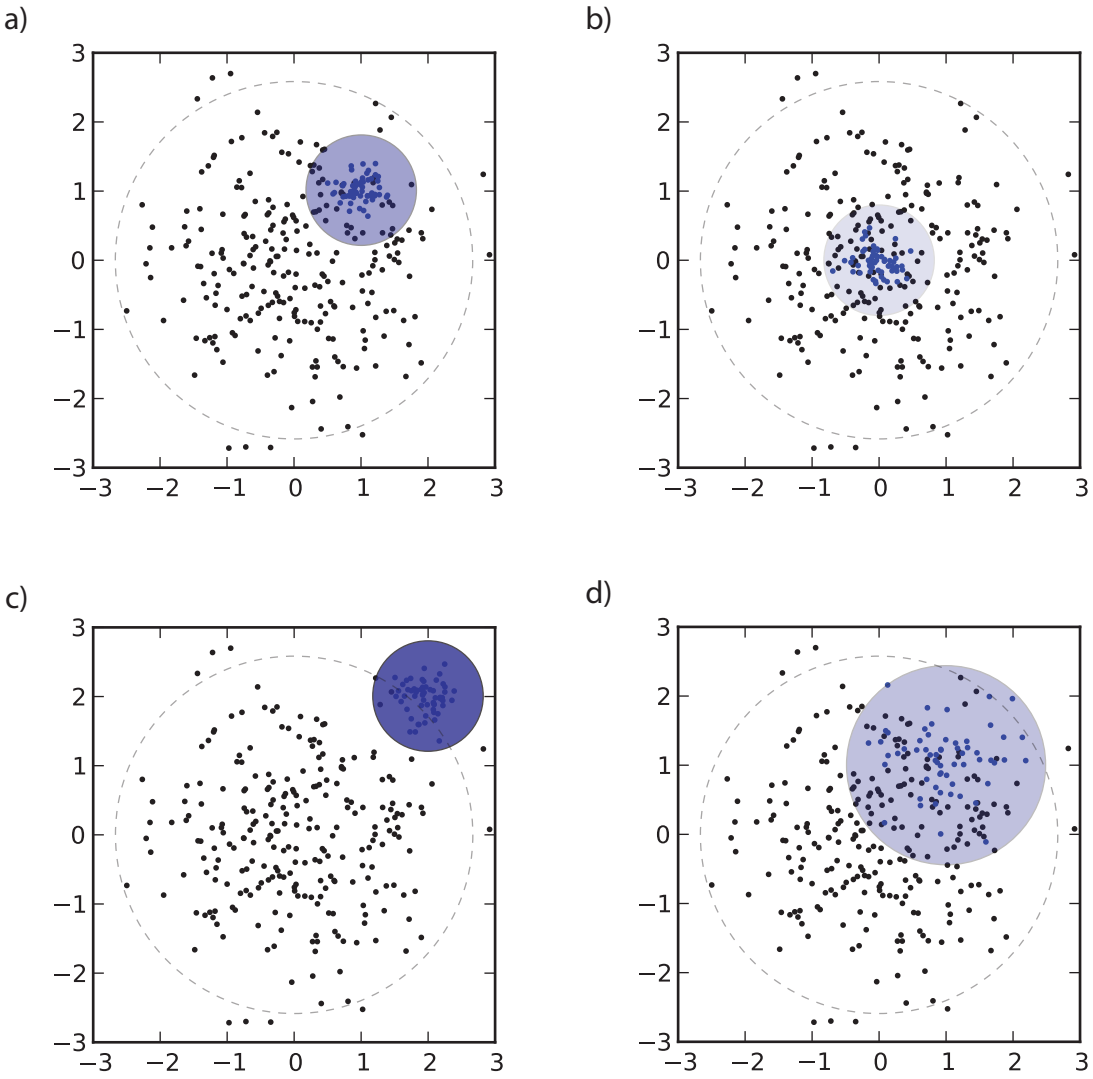

Figure 3.2: Clusters defined by point density and positional bias. Data in the background (black dots) was generated from the Gaussian distribution (3.6). The region of "typical" vector positions (with probability density $P_0(\mathbf{x}) > 10^{-2}$) is marked with a dashed circle. Cluster is formed by an agglomeration of vectors with an enhanced similarity to the cluster centre vector or by vectors located far from the origin. Cluster members are marked with blue dots. Different widths η, and positions of a cluster $\mathbf{z}$ lead to different cluster scores, here represented by different shading of circles embracing clusters. (a) Cluster of 60 vectors, with the width parameter $\eta = 5$, centred at $(1, 1)$. (b) A low-scoring cluster of 60 vectors, defined by $\eta = 5$ and centred at $(0, 0)$, the average location of the background data vectors. (c) A high-scoring cluster of 60 vectors, defined by $\eta = 5$, centred at $(2, 2)$, in the region of low density of the background data vectors. (d) Cluster of 60 vectors, centred at $(1, 1)$ with a smaller width parameter, $\eta = 2$. The cluster is thus scored lower than the "thighter" cluster in (a).

3.3 Clusters based on point density

Another popular choice for the background distribution $P_0(\mathbf{x})$ of data component values x^μ, $\mu = 1, \ldots, M$, is a simple *uniform distribution*, over some large bounded region $C \subseteq \mathbb{R}^M$. This solution has been applied to model so-called "non-conforming data", like noise and outliers, within the mixture model framework [6, 24, 38, 20] .

The uniform distribution is in fact a sub-case of our general background model distribution (3.6): we drop the fixation of the variance parameter, $\sigma_0 = 1$, and set the variance to a very large value, $\sigma_0 >> 1$. In this somewhat heuristic construction, the Gaussian distribution becomes "flat" over a large region of $\mathbb{R}^M$ centred at $\mathbf{0}$,

$$P_0(\mathbf{x}) = (2\pi\sigma_0)^{-\frac{M}{2}} \, e^{-\frac{1}{2\sigma_0^2}\mathbf{x}\cdot\mathbf{x}} \approx \frac{1}{Z_0} \; , \tag{3.10}$$

for some large Z_0, such that $\log Z_0 \simeq \frac{M}{2} \log \sigma_0$.

The Euclidean-based similarity measure leads to a Gaussian cluster model,

$$Q(\mathbf{x}|\mathbf{z}, \eta) = \frac{1}{Z_\eta} \frac{1}{Z_0} e^{-\frac{\eta}{2}(\mathbf{x}-\mathbf{z})\cdot(\mathbf{x}-\mathbf{z})} \; , \tag{3.11}$$

with $Z_\eta Z_0 = (\eta/(2\pi))^{M/2}$, equal to the Gaussian normalisation constant. The log-likelihood score is

$$s(\mathbf{x}|\mathbf{z}, \eta) = -\frac{\eta}{2}(\mathbf{x}-\mathbf{z})\cdot(\mathbf{x}-\mathbf{z}) - \left(\frac{M}{2} \log(\eta/2\pi) - \log Z_0 \right) \; . \tag{3.12}$$

This score is again a *quadratic* function of $\mathbf{x}$. An important difference from the previous variant (3.9) is that term $\frac{1}{2}\mathbf{x}\cdot\mathbf{x}$ is missing. The only strategy to increase the score is by decreasing distance $\mathbf{x}$ to $\mathbf{z}$, i.e. by the point density property.

The cluster score $S(\mathbf{x}_1, \ldots, \mathbf{x}_N|\mathbf{z}, \eta)$ based on this setting is both translationally and rotationally invariant: clusters can be positioned anywhere within the domain of the distribution and their position does not affect the score value (see Fig. 3.3).

Note that as Z_u is large and η takes rather moderate values, the *offset* in the log-likelihood score, $\log Z_\eta = (M/2 \log(\eta/(2\pi)) - \log Z_0)$, is negative. As a result, the log-likelihood score (3.12) is always positive. Using a uniform background distribution we tend to favour assigning data elements to clusters over leaving them unclustered in the background.

A uniform background is an implicit assumption of many clustering algorithms. For example, the k-means algorithm seeks a partition of data elements into K clusters, $\mathbf{C} = \{C_1, C_2, \ldots, C_K\}$, which maximises the so called *within-sum of squares* criterion,

$$f(C) = -\sum_{k=1}^{K} \sum_{\mathbf{x}_i \in C_k} (\mathbf{x}_i - \mathbf{z}_k)\cdot(\mathbf{x}_i - \mathbf{z}_k) \; . \tag{3.13}$$

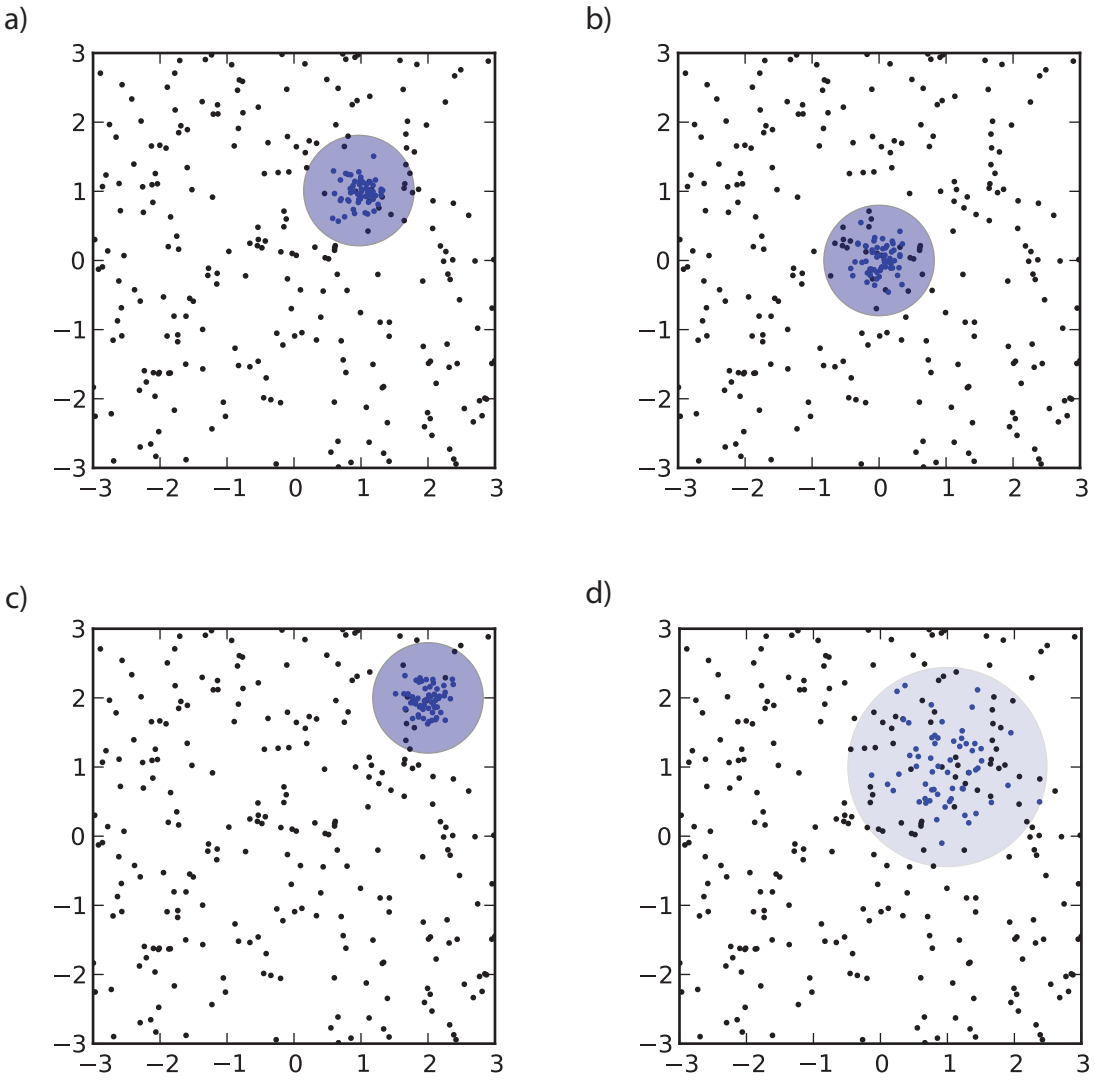

Figure 3.3: Clusters defined by point density. Data in the background (black dots) was generated from a uniform distribution in the depicted region of two-dimensional real space. Cluster is an agglomeration of vectors distinguished by an enhanced similarity to the cluster centre vector. Cluster members are marked with blue dots in all examples. Different locations do not have influence on the cluster score. Different point density in a cluster, specified by parameter η, affects the cluster score, which is represented by a shading of a circle embracing a cluster. (a), (b) and (c) Examples of clusters with 60 vectors and width parameter $\eta = 5$, centred at $(1, 1)$, $(0, 0)$ and $(2, 2)$ respectively. All clusters have similar score. (d) A "wider" cluster of 60 vectors with $\eta = 2$, centred at $(1, 1)$ is scored lower then the "tighter" clusters from (a), (b) and (c).

Cluster centres $\mathbf{z}_k$ are chosen as the centres of mass of clusters C_k, $\mathbf{z}_k = \sum_{\mathbf{x}_i \in C_k} \mathbf{x}_i / |C_k|$. A contribution of a single data element in (3.13), $-(\mathbf{x} - \mathbf{z}_k) \cdot (\mathbf{x} - \mathbf{z}_k)$, is up to a constant term, equivalent to the log-likelihood score (3.12), with parameter $\eta = 1$, $s(\mathbf{x}|\mathbf{z}, \eta = 1) = -\frac{1}{2}(\mathbf{x} - \mathbf{z}) \cdot (\mathbf{x} - \mathbf{z}) + \log Z_0$.

3.4 Clusters based on positional bias

We now consider another sub-case of the general scoring scheme from section 3.2. As the background model $P_0(\mathbf{x})$ we use again the Gaussian distribution with mean $\mathbf{0}$ and variance fixed to 1 in each dimension, as given by Eq. (3.6). Moreover, we now also fix the cluster width parameter η to 1. This way, the spread of elements in a cluster is the same as the spread of elements in the background. Clusters are thus distinguished solely by the positional bias.

The log-likelihood score (3.9) simplifies to a *linear* form,

$$s(\mathbf{x}|\mathbf{z}, \eta = 1) = \mathbf{x} \cdot \mathbf{z} - \frac{1}{2}\mathbf{z} \cdot \mathbf{z} \ . \tag{3.14}$$

The score can also be written in terms of the lengths of vectors $\mathbf{x}$ and $\mathbf{z}$ and the angle θ between them,

$$s(\mathbf{x}|\mathbf{z}, \eta = 1) = \|\mathbf{z}\| \left(\|\mathbf{x}\| \cos\theta - \|\mathbf{z}\|/2\right) \ . \tag{3.15}$$

For a fixed cluster centre $\mathbf{z}$, and $\theta \in [-\pi/2, \pi/2]$, the score increases with the length of vector $\mathbf{x}$ (i.e. its distance from the origin $\mathbf{0}$, the mean of the background distribution). The score also increases with decreasing angle θ. It can be shown that the score is positive for $\|\mathbf{x}\| > \|\mathbf{z} - \mathbf{x}\|$, and negative otherwise: vector $\mathbf{x}$ is assigned to the cluster with centre $\mathbf{z}$ if distance between $\mathbf{x}$ and $\mathbf{z}$ is smaller then distance between $\mathbf{x}$ and $\mathbf{0}$.

The total cluster score $S(\mathbf{x}_1, \ldots, \mathbf{x}_N|\mathbf{z})$ is not translationally invariant: clusters are indeed defined by their location. The score is rotationally invariant. We show an illustration of this scoring scheme in Fig. 3.4

3.5 Clusters based on directional density

As discussed in section 3.2, the so-called positional bias property may reward one-element clusters formed by long outlier vectors. The positional bias is specific to the scoring scheme with the Gaussian background combined with Euclidean distance-based similarity measure. As an alternative, we consider another property of clusters: the so-called *directional density*. To this end, we employ a correlation-based similarity measure, which compares only cluster directions, but disregards their lengths:

$$\mathrm{sim}(\mathbf{x}, \mathbf{z}) = M \frac{\mathbf{x} \cdot \mathbf{z}}{\sqrt{\mathbf{x} \cdot \mathbf{x}}\sqrt{\mathbf{z} \cdot \mathbf{z}}} = \hat{\mathbf{x}} \cdot \hat{\mathbf{z}} \ . \tag{3.16}$$

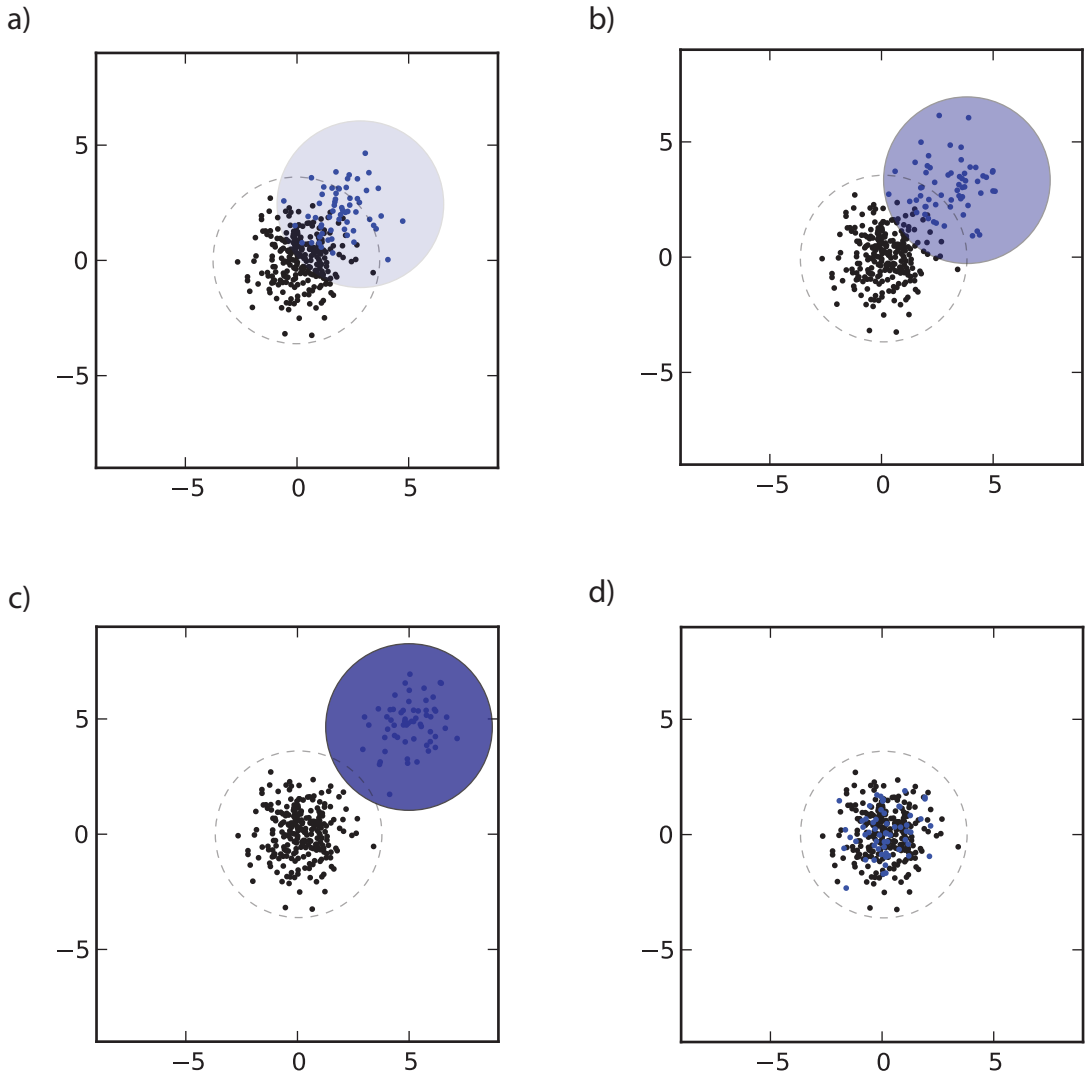

Figure 3.4: Clusters defined by positional bias. Data in the background (black dots) was generated from the Gaussian distribution (3.6). The region embracing background vectors is marked with a dashed circle. Cluster is formed by an agglomeration of vectors with an enhanced similarity to the cluster centre vector, located far from the origin $(0, 0)$. Cluster members are marked with blue dots in all examples. All clusters and the background data are characterised by the same spread of data vectors. Different positions of a cluster lead to different cluster scores, here represented by different shading of circles embracing clusters. (a) A low-scoring cluster of 60 vectors centred at $(2, 2)$, relatively close to the origin $(0, 0)$. Most of its elements are closer to the point of origin than to the cluster centre and they do not contribute to the cluster score $S(\mathbf{x}_1, \ldots, \mathbf{x}_N|\mathbf{z})$. (b) A higher-scoring cluster of 60 vectors centred at $(3, 3)$. A small fraction of vectors from the cluster is still closer to the origin than to the cluster centre. (c) A high-scoring cluster of 60 vectors centred at $(5, 5)$. All cluster members are closer to the cluster centre than to the origin. (d) A zero-scoring cluster of 60 vectors, centred at the origin $(0, 0)$.

Vectors $\hat{\mathbf{x}}$ and $\hat{\mathbf{z}}$ are length-normalised vectors pointing in the direction of the original vectors $\mathbf{x}$ and $\mathbf{z}$, normalised such that $\|\hat{\mathbf{x}}\| = \sqrt{M}$ and $\|\hat{\mathbf{z}}\| = \sqrt{M}$. We set the lengths of vectors to $\sqrt{M}$, the expected length of a vector under the standard Gaussian distribution (which was used as the background distribution in the previous sections of this chapter).

This form of the score assumes projection of the data on the surface of M-dimensional sphere centred at $\mathbf{0}$, with radius $\sqrt{M}$. If the original vectors are Gaussian distributed as in our background model (3.6), then the projected data is *uniformly distributed* on the surface of the sphere. We write the spherical background model as

$$P_0(\hat{\mathbf{x}}) = \frac{1}{Z_0}\delta(\hat{\mathbf{x}} \cdot \hat{\mathbf{x}} - M) \ , \tag{3.17}$$

with normalisation constant $Z_0 \simeq \exp\{M\,(1/2(1 + \log(2\pi))\}$ given by the surface of M-dimensional sphere with radius $\sqrt{M}$.

Correlation appears as a common choice for measuring similarity of vectors in high dimensional spaces [87, 30, 3]. In case of gene expression patterns, the correlation reflects the biological intuition: two genes are co-expressed if they react in the same fashion across different experimental conditions. The amplitude of the changes can be gene-specific and does not contribute to the similarity.

For the central vector $\hat{\mathbf{z}}$, the cluster model is

$$Q(\hat{\mathbf{x}}|\hat{\mathbf{z}}, \eta) = \frac{1}{Z_\eta} P_0(\hat{\mathbf{x}}) e^{\eta \hat{\mathbf{x}} \cdot \hat{\mathbf{z}}} \ , \tag{3.18}$$

with normalisation factor

$$Z_\eta \simeq \exp\left\{M\left((\gamma - 1)/2 + \eta^2/(2\gamma) - \log\gamma/2\right)\right\} \tag{3.19}$$

for $\gamma = (1 + \sqrt{1 + 4\eta^2})/2$. The derivation for this result, asymptotic in the number of dimensions M, uses a saddle point approximation, both for Z_0 and Z_η (shown in Appendix A).

The log-likelihood score,

$$s(\hat{\mathbf{x}}|\hat{\mathbf{z}}, \eta) = \eta\, \hat{\mathbf{x}} \cdot \hat{\mathbf{z}} - \log Z_\eta \ . \tag{3.20}$$

is a *linear* function of $\hat{\mathbf{x}}$. For length-normalised vectors, the scalar product is proportional to the cosine of the angle between vectors $\hat{\mathbf{x}}$ and $\hat{\mathbf{z}}$. Hence, score (3.20) increases with decreasing angle between the vectors.

The cluster score $S(\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_N|\hat{\mathbf{z}}, \eta)$ is rotationally invariant. Vectors $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{z}}$ are restricted to the sphere surface, so the cluster score is no longer translationally invariant. We illustrate this scoring scheme in Fig. 3.5.

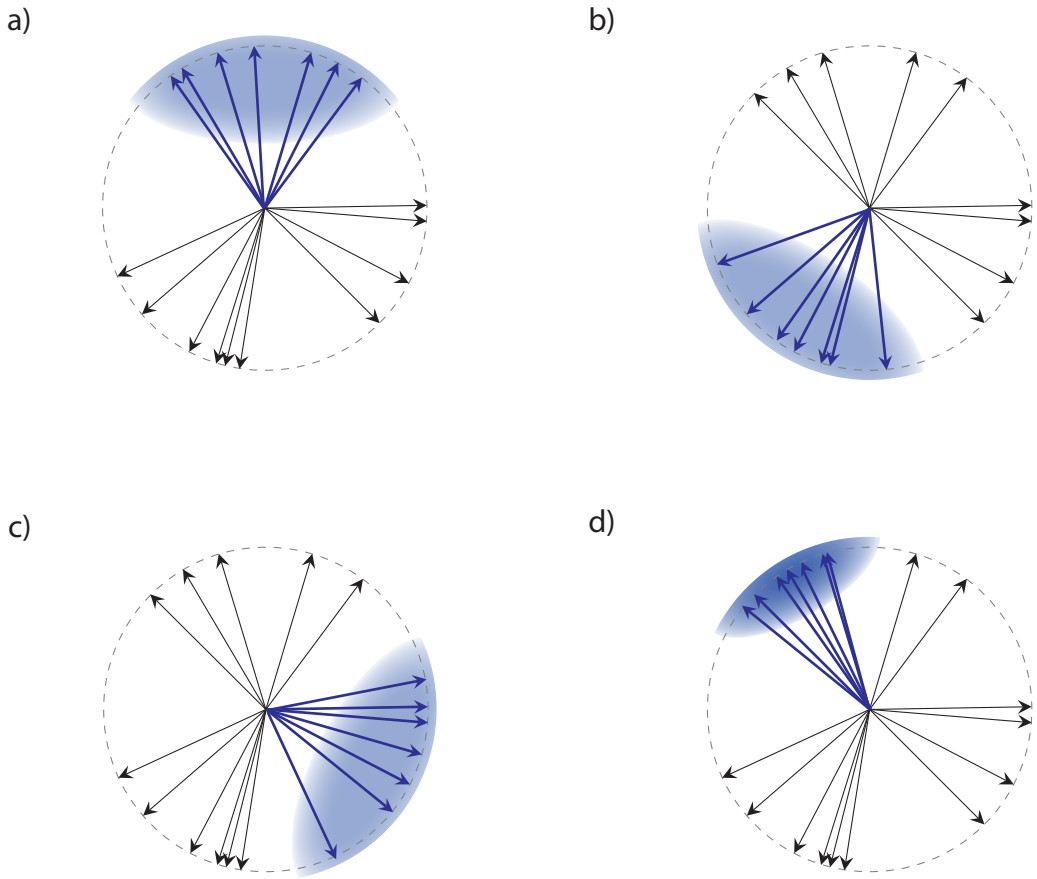

Figure 3.5: Clusters defined by directional density. Data in the background is uniformly distributed on a surface of a sphere (black arrows). Cluster is formed by an agglomeration of vectors with an enhanced similarity between cluster members $\hat{\mathbf{x}}$ and cluster centre $\hat{\mathbf{z}}$ (here correlation i.e. cosine of the angle between $\hat{\mathbf{x}}$ and $\hat{\mathbf{z}}$). The direction of a cluster does not have influence on the cluster score. Directional density, specified by parameter η, affects cluster score, which in the following examples is depicted by the shading of a sphere cap embracing vectors in a cluster. (a), (b) and (c) examples of clusters with 7 vectors and the same width parameter η. All clusters have similar score. (d) A "tighter" cluster of 7 vectors with a larger value of η is scored higher then the "wider" clusters from (a), (b) and (c).

3.6 Summary

We proposed a probabilistic framework for modelling clusters in a high dimensional space. First, we introduced a background model characterising distribution of vectors which are not clustered. We then contrasted the background model with the cluster model: the probability distribution of a vector belonging to a cluster. Based on these two models, we proposed a cluster scoring function. We focused on a class of "spherical" clusters: agglomerations of data points distributed around a central vector. We did not model any cluster generating process, which would be appropriate, for example, for a problem involving a hierarchical structure of clusters. Such a solution would require another specification of the cluster model $Q(\mathbf{x})$.

We considered three variants of the background distribution for unclustered vectors: the Gaussian distribution, the uniform distribution in the real space, and the uniform distribution on the surface of a sphere. In making these choices, we were particularly interested in the context of gene expression data, but the applicability of our models is much broader.

Background model has an influence on the cluster scoring function. Many clustering algorithms disregard this issue and implicitly assume the uniform background distribution. As we had shown on an example of the yeast expression data, this assumption is not correct for gene expression, which rather follows the Gaussian distribution. As a consequence of using a wrong background model, the standard algorithms, such as k-means, are likely to find spurious clusters which arise from high density regions of the background distribution.

In Table 3.1, we summarise all combinations of discussed models. The presented framework will be used in the subsequent chapters of the thesis. In particular, in Chapter 4 we present an analytical solution for computing the statistical significance of a cluster score. This solution is valid for models which show a linear dependence on data vectors $\mathbf{x}$, namely the positional-bias and the directional-density based models from our classification.

	Background model	η	$\|\mathbf{z}\|$	Trans. inv.	Rot. inv	Score as a function of $\mathbf{x}$	Clusters scored by
A	Multivariate Gaussian	any	any	-	+	quadratic	width
B	Uniform	any	any	+	+	quadratic	width, location
C	Multivariate Gaussian	1	any	-	+	linear	location
D	Uniform on M-sphere	any	$\sqrt{M}$	-	+	linear	angular width

Table 3.1: Classification of cluster-scoring schemes. The table summarises the scoring schemes discussed in this chapter by specifying: the background distribution, the range of the multiplicative parameter η, the length constraint on the cluster centre $\mathbf{z}$, translational/rotational invariance and the property based on which the clusters are scored. Models A and C, which are characterised by a non-uniform background distribution, lead to a scoring scheme which is not translationally invariant. All discussed background models lead to rotationally invariant scoring.

Chapter 4

Statistical significance analysis of clusters

Agglomerations of densely distributed vectors can arise even in the set of independently distributed vectors, i.e. the background data, simply as a result of random density fluctuations. In this chapter, we formulate the cluster significance problem. To distinguish the true and spurious clusters, we compute the cluster score p-value: the probability that a cluster of score S or higher arises in a set of random vectors from the background distribution. We solve this problem analytically, establishing a connection between the physics of quenched disorder and multiple-testing statistics in clustering and related problems. We illustrate our results by application to clustering of gene expression data, where high-dimensional data vectors are generated by multiple measurements of a gene under different experimental conditions.

4.1 Significance analysis

In Chapter 3, we proposed a probabilistic theory for modelling clusters and the background data. This formulation led to a well-defined scoring scheme for clusters: for a given group of vectors we compute the cluster score S which quantifies its clustering properties, such as the point density, the positional bias or the directional density.

Clusters in data are usually a signature of an underlying functional mechanism, e.g. a biological pathway causes co-expression of involved genes. However, even unrelated vectors drawn from the background distribution can form agglomerations which by chance resemble clusters. To illustrate this problem (see Fig. 4.1) we generated 100 vectors in a 50-dimensional space from a standard normal distribution. As such, the data did not contain any predefined clusters. Still, due to random density fluctuations, we observed dense structures arising in the data. Any clustering algorithm would also detect these structures, if ran with improperly chosen parameters. We ran the k-means to find a partition into $k = 3$ clusters. The algorithm returned a partition of data into clusters, but it did not assess the significance of this result, which, in this case, is a partition into spurious clusters.

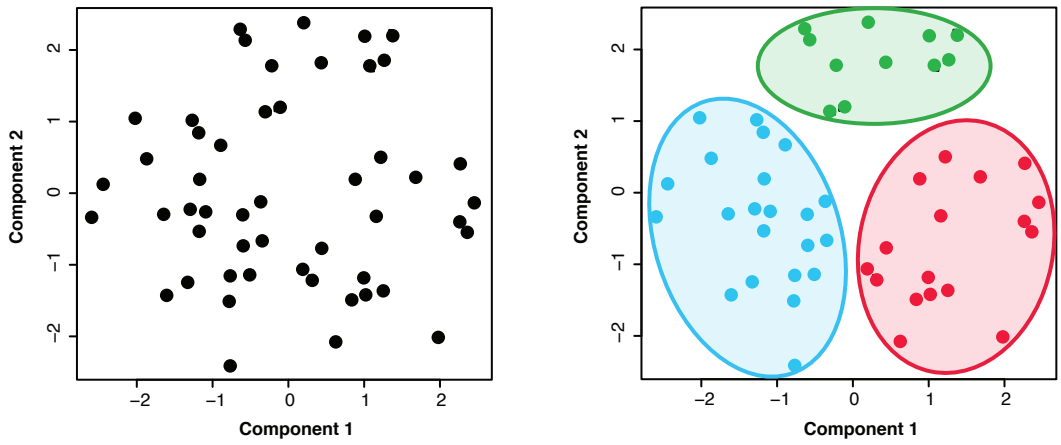

Figure 4.1: Random clusters in uniformly distributed data. 100 vectors in 50-dimensional space were generated from a standard normal distribution. Here we show a scatter plot displaying the data in the coordinates given by the first two principal components (left). The k-means algorithm was run on this data with $k = 3$. The resulting clusters are marked with colours on the right scatter plot.

Quality of a "spurious" cluster can also be quantified with a cluster scoring function, yielding some score S_0. To distinguish the true and random clusters, we need to characterise the distribution of the cluster score $p(S)$ for vectors from the background distribution. The p-value of score S_0 is then defined as the probability that a random data set contains a cluster with score greater than or equal to S_0. In the statistical significance analysis we proceed as follows: given a group of vectors with some score S_0, we formulate a null hypothesis: "These vectors are drawn from the background distribution". To test this hypothesis, we compute the p-value of score S_0: low p-value suggests that the null hypothesis is unlikely and allows for rejecting it. Importantly, a low p-value does not yet say that the group of vectors is indeed a cluster. Low p-value provides a necessary but not a sufficient condition in this direction.

Problem setting. We consider an ensemble of N vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$, which are drawn independently from background distribution $P_0(\mathbf{x})$. We are specifically interested in data vectors with a large number of components, M. Clusters of vectors in such high dimensional spaces are generically supported by multiple vector components, which is the source of the intricate cluster statistics discussed below. In Chapter 3, we characterised concrete choices of the background distribution. Here, we will consider a more general class of models: we will make a minimal assumption that distribution $P_0(\mathbf{x})$ factorizes in the vectors components,

$$P_0(\mathbf{x}) = p_0(x^1) \ldots p_0(x^M) \ , \tag{4.1}$$

and that the marginal distribution $p_0(x^\mu)$ has finite mean and variance, set to 0 and 1 without loss of generality.

We will consider two classes of linear scoring functions for vectors in a cluster:

1. Linear score with a predefined score offset μ and length-constrained cluster centre $\hat{\mathbf{z}}$, $\|\hat{\mathbf{z}}\| = \sqrt{M}$,
$$s_1(\mathbf{x}|\hat{\mathbf{z}}, \mu) = \mathbf{x} \cdot \hat{\mathbf{z}} - \mu \ . \tag{4.2}$$
The log-likelihood score (3.20) for clusters based on directional density belongs to this class.

2. Linear score for clusters based on positional bias, as formulated in Eq. (3.14), (with not length-constraint on the cluster centre),
$$s_2(\mathbf{x}|\mathbf{z}) = \mathbf{x} \cdot \mathbf{z} - \frac{1}{2}\mathbf{z} \cdot \mathbf{z} \ . \tag{4.3}$$

Again, a cluster is a subset of positively scoring vectors. The cluster score is a sum of score contributions from vectors in the cluster, see definition in Eq. (3.5).

Framework of the solution. In this chapter, we propose an analytical approach to deriving the distribution of cluster score under the null model. Our approach is based on an intimate connection between cluster score statistics and the physics of disordered systems: the calculation employs the statistical mechanics of a system whose Hamiltonian is given by (minus) the cluster score function (3.5). In this system, $\log p(S)$ is the entropy of all data vector configurations with energy below $-S$. We evaluate this entropy in the limit where both the number of random vectors, and the dimension of the vector space are large. High-scoring clusters have to be found in each *fixed configuration* of the random data vectors, which act as quenched disorder for the statistics of clusterings. The disorder turns out to generate correlations between the scores of clusters centred on different directions of the data vector space. These correlations, which become particularly significant in high-dimensional datasets, show that clustering is an intricate multiple-testing problem: spurious clusters may appear in many different directions of the data vectors.

In the first step towards our solution, we compute $p_c(S)$, distribution of score S with a predefined, *fixed cluster centre* $\mathbf{z}$. Of course, a cluster centre is always optimised for a given group of vectors. We follow with a full solution, which is distribution $p(S)$ of the *maximal cluster score* in data, i.e. the distribution of the score of the highest scoring cluster in an ensemble of random vectors. We then show how these two distributions are related.

In section 4.2, we present the solution for the cluster score based on scoring function (4.2), with a length constrained centre $\hat{\mathbf{z}}$. We show that this solution is a valid approximation for the distribution of the cluster score defined by directional density with length-normalised data vectors (see section 3.5). In section 4.3, we present the solution for the cluster score based on scoring scheme (4.3), which quantifies positional bias of a cluster (see section 3.4).

4.2 Statistics of clusters based on directional density

Clusters in a fixed direction. We first compute the distribution $p_c(S)$ of cluster scores for clusters with a *fixed* centre $\mathbf{z}$. Since the background distribution $P_0(\mathbf{x})$ is rotationally invariant, $p_c(S)$ does not depend on the direction of the cluster centre $\hat{\mathbf{z}}$. Without loss of generality, we set $\hat{\mathbf{z}} = [1, 1, \ldots, 1]$. We define a random variable, the *overlap* of vector $\mathbf{x}$ and $\hat{\mathbf{z}}$, as a scalar product

$$x_i \equiv \mathbf{x}_i \cdot \hat{\mathbf{z}} = \sum_{\mu=1}^{M} x^\mu \ . \tag{4.4}$$

The overlap is a sum of components of vector $\mathbf{x}$, i.e. it is a sum of M identically distributed random variables with mean 0 and variance 1. By the central limit theorem, x_i is approximately Gaussian-distributed with mean 0 and variance M, $P(x_i) = \sqrt{1/(2M\pi)}\exp\{-x^2/(2M)\}$.

We can now rewrite cluster score (3.5) as a function of the overlaps of data vectors with the cluster centre,

$$S(x_1, \ldots, x_N|\mu) \equiv S(\mathbf{x}_1, \ldots, \mathbf{x}_N|\hat{\mathbf{z}}, \mu) = \sum_{i=1}^{N} \max\left[x_i - \mu, 0\right] \ . \tag{4.5}$$

Computation of the distribution of the score S is straightforward from the derivation shown in Chapter 2 and requires calculation of the partition function:

$$\begin{aligned}
Z_c(\beta, \mu) &= \int_{\mathbb{R}^{\mathbb{N}}} e^{\beta S(x_1,\ldots,x_N|\hat{\mathbf{z}},\mu)} P(x_1)\ldots P(x_N)\, dx_1 \ldots dx_N \\
&= \left[\int_{-\infty}^{+\infty} e^{\beta \max[x-\mu,0]} P(x)dx\right]^N \\
&= \left[\int_{-\infty}^{\mu} P(x)dx + \int_{\mu}^{+\infty} e^{\beta(x-\mu)} P(x)dx\right]^N \qquad (4.6) \\
&= \left[(1 - H(\mu)) + e^{\frac{(\beta)^2}{2} - \beta\mu} H(\mu - \beta)\right]^N \qquad (4.7)
\end{aligned}$$

with $H(x) = \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}y^2} dy$, the complementary cumulative Gaussian distribution. In line (4.6) the integration is divided in two intervals: below the score threshold μ, the score is zero, which contributes the cumulative distribution $\int_{-\infty}^{\mu} dx/(2\pi)^{1/2}\ \exp[-x^2/2]$ to the generating function. Above the score threshold, the score is positive, which generates a contribution of $\int_{\mu}^{\infty} dx/(2\pi)^{1/2}\ \exp[-x^2/2 + \beta(x-\mu)]$. The free energy function reads

$$-\beta f_c(\beta, \mu) = \log\left[(1 - H(\mu)) + e^{\frac{\beta^2}{2} - \beta\mu} H(\mu - \beta)\right] \ , \tag{4.8}$$

and the entropy is

$$\omega_c(s, \mu) = -\max_{\beta}[\beta s + \beta f_c(\beta, \mu)] \ . \tag{4.9}$$

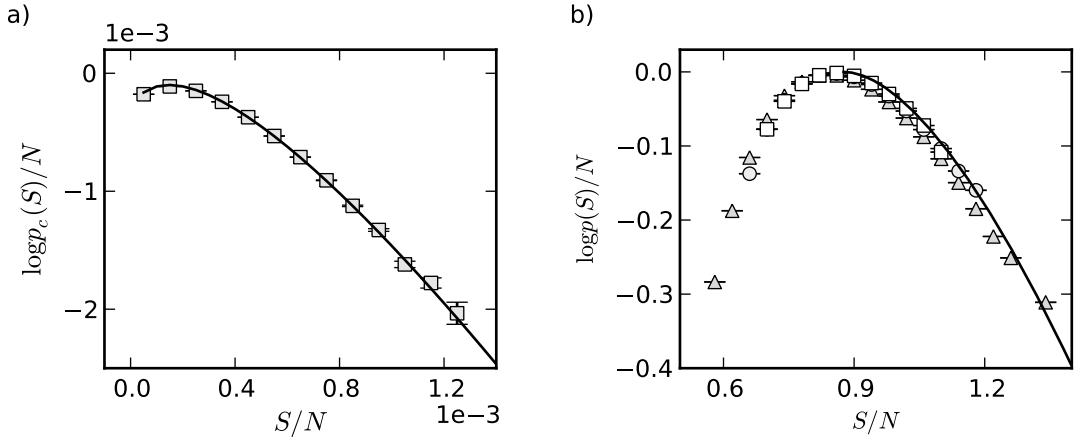

Figure 4.2: Cluster score distributions in random data for fixed and optimal cluster direction. Analytical distributions $p(S)$ (solid lines) are plotted against the score per element, $s = S/N$, and are compared to normalized histograms obtained from numerical experiments with 10^6 samples (squares). (a) Distribution $p(S)$ of the cluster log likelihood score (3.5) with parameter $\mu = 0.37$ for fixed cluster centre and datasets of $N = 6000$ vectors with $M = 70$. Error bars show the standard error due to the finite size of the sample. (b) Distribution of the maximum cluster score (4.11) with parameter $\mu = 0.1$ for $N = 40$ (triangles), $N = 80$ (circles) and $N = 120$ (squares), keeping $M/N = 0.5$ fixed. The analytical solution is valid asymptotically for large N, but good agreement with the numerics is visible already for moderate values of N.

As described in Chapter 2,

$$\log p_c(S, \mu) \simeq N\omega_c(S/N, \mu) - \frac{1}{2}\log N \ . \tag{4.10}$$

The resulting cluster score distribution is plotted in Fig. 4.2 (a) together with a score distribution obtained from simulations of randomly generated data vectors, showing excellent agreement. The leading asymptotics of $p_c(S, \mu)$ can also be derived using large deviation statistics [23].

Maximal scoring clusters. To gauge the statistical significance of high-scoring clusters in actual datasets, we need to know the distribution of the *maximum cluster score* in random data. The maximum cluster score is implicitly related to the *optimal cluster direction* in a dataset: for a given subset of vectors $\mathbf{x}_1, \ldots, \mathbf{x}_k$, the maximal cluster score is reached if the direction of the center $\hat{\mathbf{z}}$ coincides with the direction of the "centre of mass", $\mathbf{x}_{\rm av} = (\mathbf{x}_1 + \cdots + \mathbf{x}_k)/k$. However, adding or removing vectors shifts the centre of mass $\mathbf{x}_{\rm av}$ of the cluster and changes the score of each vector. Thus, finding the maximum score for a given dataset

$$S_{\max}(\mathbf{x}_1, \ldots, \mathbf{x}_N|\mu) = \max_{\hat{\mathbf{z}}, \|\hat{\mathbf{z}}\|=\sqrt{M}} S(\mathbf{x}_1, \ldots, \mathbf{x}_N|\hat{\mathbf{z}}, \mu) \tag{4.11}$$

is a hard algorithmic problem, in particular for large dimensions M. The algorithmic difficulty is reflected by non-trivial statistics of the maximal cluster score.

We use the same statistical mechanics framework to compute the distribution of the maximal cluster score. We compute the partition function

$$Z(\beta,\mu) = \int_{\mathbb{R}^{M,N}} e^{\beta S_{\max}(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mu)} P(\mathbf{x}_1)\ldots P(\mathbf{x}_N)\, d\mathbf{x}_1 \ldots d\mathbf{x}_N \ , \tag{4.12}$$

where the "energy" function is now given by the maximum cluster score (4.11). This computation is more difficult because of the max function defining $S_{\max}$. To make it analytically treatable, we use the integral representation of $S_{\max}$,

$$e^{\beta S_{\max}(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mu)} = \lim_{1/\beta'\to 0} \left[\int \delta(\hat{\mathbf{z}}\cdot\hat{\mathbf{z}} - M) e^{\beta' (S(\mathbf{x}_1,\ldots,\mathbf{x}_N|\hat{\mathbf{z}},\mu) d\hat{\mathbf{z}}}\right]^{\beta/\beta'} \tag{4.13}$$

for the statistical weight of a configuration $\mathbf{x}_1,\ldots,\mathbf{x}_N$. The Dirac-delta function assures that the integration is performed over vectors $\hat{\mathbf{z}}$ constrained to the sphere surface. We introduce an auxiliary variable β' which will be taken to infinity. For large values of β', only directions $\hat{\mathbf{z}}$ with a high cluster score $S(\mathbf{x}_1,\ldots,\mathbf{x}_N|\hat{\mathbf{z}},\mu)$ contribute to this integral, and the maximum over the cluster score (4.11) is reproduced in the limit $\beta/\beta' \to 0$.

The calculation uses the so-called replica trick [17, 31, 41, 79](see section 2.4), representing the power $n = \beta/\beta'$ of the integral in (4.13) by a product of n copies (replicas):

$$\left[\int e^{\beta' S(\mathbf{x}_1,\ldots,\mathbf{x}_N|\hat{\mathbf{z}},\mu) d\hat{\mathbf{z}}}\right]^{\beta/\beta'} = \prod_{a=1}^{n} \int \delta(\hat{\mathbf{z}}_a\cdot\hat{\mathbf{z}}_a - M) e^{\beta' S(\mathbf{x}_1,\ldots,\mathbf{x}_N|\hat{\mathbf{z}}_a,\mu) d\hat{\mathbf{z}}_a} \ . \tag{4.14}$$

The calculation proceeds for integer values of n, and the limit $n \to 0$ ($\beta' \to \infty$) is taken by analytic continuation. A key ingredient is the average *overlap* $q = \langle \hat{\mathbf{z}}\cdot\hat{\mathbf{z}}'\rangle/M$ between directions of different cluster centres for the same configuration of data vectors at finite temperature $1/\beta'$. We find a unique ground state (i.e., $q \to 1$ for $\beta' \to \infty$) and a low-temperature expansion

$$q = 1 - \frac{a}{\beta'} + O\left(\frac{1}{\beta'^2}\right),$$

of the average overlap, similar to the case of directed polymers in a random potential [51], which arises in the statistics of sequence alignment [52]. Thus, the effect of centre optimisation on cluster p-values is related to the fluctuations between subleading cluster centres for the same random dataset.

The full calculation is lengthy and is shown in Appendix C. We obtain the free energy function

$$-\beta f_1(\beta,\mu) = \min_a \left[-\beta f_c\left(\beta, \mu - \frac{a}{2}\right) + \frac{M}{2N}\log\left(\frac{a+\beta}{a}\right)\right] \ . \tag{4.15}$$

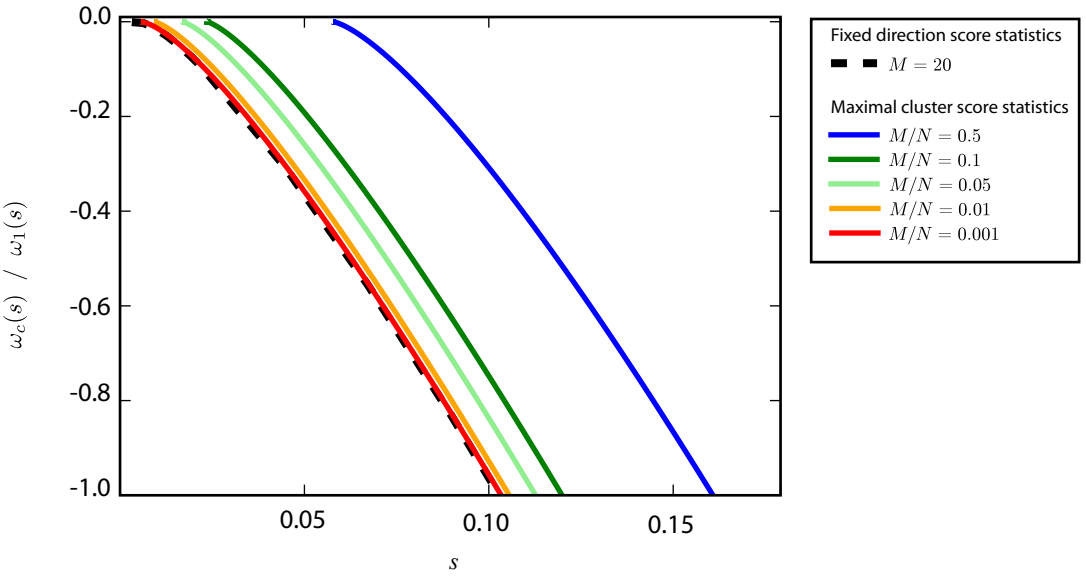

Figure 4.3: Score entropies $\omega_c(s)$ and $\omega_1(s)$ of the fixed direction cluster score and the maximal cluster score. The entropy determines the leading asymptotics of the distributions of the intensive score $s = S/N$; Analytical solutions are shown for $\mu = 0.37$, $M = 20$, and $M/N = 0.5$ (blue line), 0.1 (dark green), 0.05 (light green), 0.01 (orange) and 0.001 (red). The distribution $\omega_c(s)$ does not depend on M/N (black dashed line). In the limit of large N, the distribution of the optimal cluster score converges to the distribution of the fixed direction cluster score.

This expression is to be understood in the asymptotic limit $N \to \infty$ with M/N kept fixed. It involves the variation over an additional parameter a, which is related to the similarity between competing cluster centres for the same configuration of vectors $\mathbf{x}_1, \ldots, \mathbf{x}_N$. Compared to the corresponding expression (4.7) for fixed cluster centre, there is an effective shift $a/2$ in the score cutoff μ and an additional entropy-like term. This solution determines the asymptotic form of the distribution of maximum cluster score $S_{\max} = S$. The intensive entropy reads

$$\omega_1(s, \mu) = -\max_{\beta}[\beta s + \beta f_1(\beta, \mu)] \ . \tag{4.16}$$

and the resulting probability distribution is

$$\log p_1(S, \mu) = N\omega_1(S/N, \mu) + \mathcal{O}(\log N) \ . \tag{4.17}$$

This is plotted in Fig. 4.2 (b) together with numerical simulations for several values of M and N, showing good agreement already for moderate values of N.

According to (4.15), the effect of centre optimisation on the score statistics increases with the number of data components, M, and decreases with the size of the dataset, N. This effect is shown in Fig. 4.3: we keep fixed $M = 20$ and increase N. As the ratio M/N is decreasing, the maximal score entropy $\omega_1(s, \mu)$ is converging to the fixed direction score entropy $\omega_c(s, \mu)$.

Score distribution for small M/N. For small values of M/N, we can expand the solution to leading order and obtain $-\beta f_1(\beta,\mu) = -\beta f_c(\beta,\mu) + (M/2N)\log N +$ const., which leads to a distribution of maximum cluster scores given by

$$\log p_1(S,\mu) = \log p_c(S,\mu) + \frac{M}{2}\log N = N\omega_c(s,\mu) + \frac{M-2}{2}\log N \tag{4.18}$$

up to terms of order N^0. This expansion is appropriate for the sizes M, N encountered with the typical genome-wide gene expression datasets.

p-value. The p-value of a cluster score S is the probability that the score is greater than or equal to S,

$$p\text{-value}(S,\mu) = \int_S^{+\infty} p_1(S',\mu)dS' \ . \tag{4.19}$$

Inserting (4.17) shows that this p-value equals $p(S)$ up to a proportionality factor of order 1. Thus, we will simply use $p_1(S)$ to denote the cluster score p-value.

Clusters on a sphere. We showed a derivation for the distribution of the cluster score based on a scoring scheme (4.2) which quantifies the directional density property. Our derivation is valid for a wide class of background distributions modelling high dimensional data with identically and independently distributed data components. That is because, for a large number of dimensions M, the distribution of the score per vector $s_1(\mathbf{x}|\hat{\mathbf{z}},\mu)$ is Gaussian by the law of large numbers. Here, we show that the derived distribution $p(S,\mu)$ is also an asymptotic solution for a distribution of the maximal cluster score of data vectors *uniformly distributed on a sphere*.

Consider a vector $\mathbf{x} \in \mathbb{R}^M$ following a standard Gaussian distribution. The squared length of $\mathbf{x}$ is a random variable which follows a chi-square distribution with M degrees of freedom,

$$\|\mathbf{x}\|^2 = \sum_{\mu=1}^{M}(x^\mu)^2 \sim \chi^2_M \ , \tag{4.20}$$

which has mean M and variance $2M$. Consequently, the length of $\mathbf{x}$ has the expected value equal to the radius of the sphere,

$$\mathbb{E}\left[\|\mathbf{x}\|\right] = \sqrt{M} \ , \tag{4.21}$$

and from $\mathrm{Var}[f(X)] \approx (f'(E[X]))^2\mathrm{Var}[X]$, the variance

$$\mathrm{Var}\left[\|\mathbf{x}\|\right] \approx \frac{1}{2} \ . \tag{4.22}$$

Hence, the ratio of the standard deviation to the expected vector length,

$$\mathrm{sd}\left[\|\mathbf{x}\|\right] / \mathbb{E}\left[\|\mathbf{x}\|\right] = \frac{1}{\sqrt{2M}} \ , \tag{4.23}$$

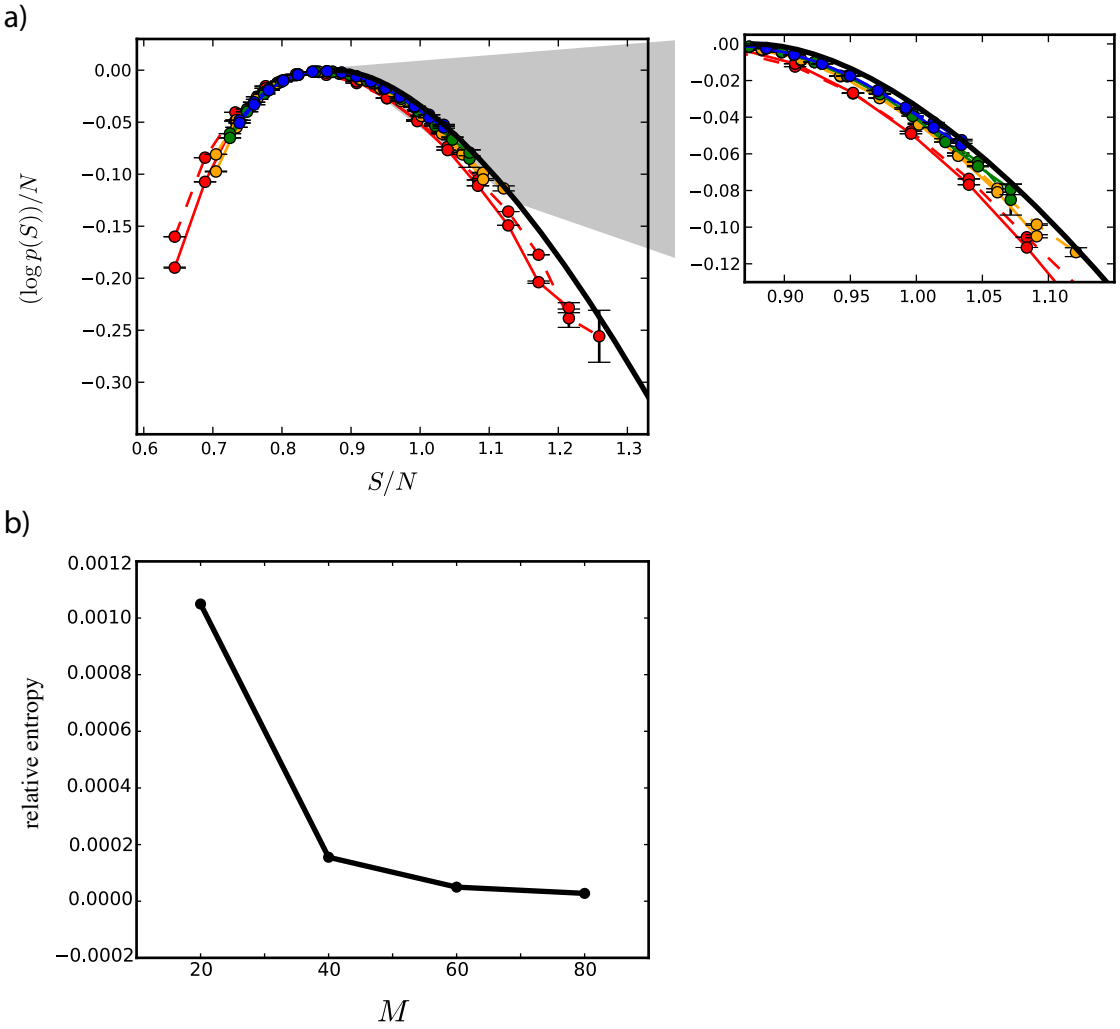

Figure 4.4: Maximal cluster score distribution for Gaussian and spherical data vectors. (a) We compared distribution of the maximal cluster score in Gaussian distributed vectors (dashed lines) and in vectors uniformly distributed on a sphere (solid lines). The numerical distributions were obtained with simulations with 10^6 samples for $M = 20, 40, 60, 80$, (red, orange, green, blue) keeping $M/N = 0.5$ fixed. The cluster elements were scored with a linear score (4.2), $s_1(\mathbf{x}|\hat{\mathbf{z}}, \mu) = \mathbf{x}\cdot\hat{\mathbf{z}} - \mu$, with $\mu = 0.1\sqrt{M}$. To compare between different system sizes, we plot the intensive score S/N against the intensive log probability, $\log p_1(S)/N$. The difference between the dashed and the solid line decreases with increasing M. The black solid line shows the analytical result for $\log p_1(S)/N$, see Eq. (4.17). (b) The relative entropy of each pair of the numerical distributions from (a) decreases as a function of M.

converges to 0 as $M \to \infty$. In other words, the expected position of vector $\mathbf{x}$ is on the surface of M-dimensional sphere of radius $\sqrt{M}$, and the more dimensions, the smaller the deviation from that position.

We thus expect that the analytical solution (4.17) should also fit the distribution of the maximal cluster score for an ensemble of vectors uniformly distributed on a sphere. We tested this with numerical simulations, during which we were recording the maximal cluster score in Gaussian distributed vectors and in vectors uniformly distributed on a sphere. The result of simulations with parameter $\mu = 0.1\sqrt{M}$ and $M/N = 0.5$, for $M = 20, 40, 60, 80$, are shown in Fig. 4.4. The two distributions are very similar, the differences between them become insignificant (within the error margins) with increasing M. To quantify the convergence, we computed the relative entropy between each pair of the numerical distributions: the distance appeared to decrease with M.

Let us now take vector a length-constrained vector $\hat{\mathbf{x}}$. The log-likelihood score function (3.20) can be written in terms of scoring function $s_1(\hat{\mathbf{x}}|\hat{\mathbf{z}}, \mu)$ as

$$s(\hat{\mathbf{x}}|\hat{\mathbf{z}}, \eta) = \eta\hat{\mathbf{x}} \cdot \hat{\mathbf{z}} - \log Z_\eta = \eta(\hat{\mathbf{x}} \cdot \hat{\mathbf{z}} - (\log Z_\eta)/\eta) = \eta s_1(\hat{\mathbf{x}}|\hat{\mathbf{z}}, (\log Z_\eta)/\eta)) \ , \quad (4.24)$$

where Z_η is the normalisation constant defined in Eq. (3.19). The cluster score probability for the spherical case is thus given by a simple scaling of the result (4.17) by a factor η,

$$\log p(S, \eta) = \log p_1(S, \log Z_\eta/\eta) + \log \eta \ . \quad (4.25)$$

4.3 Statistics of clusters based on positional bias

We perform a similar replica-based calculation for the scoring function (4.3), $s_2(\mathbf{x}|\mathbf{z}) = \mathbf{x}\cdot\mathbf{z} - \frac{1}{2}\mathbf{z}\cdot\mathbf{z}$. As described in Chapter 3, this function defines a cluster by its positional bias. The scoring function differs from the one considered in the previous section by the use of vector $\mathbf{z}$, which is no longer constrained by the length. In fact, the length of $\mathbf{z}$ acts as a score threshold, by means of the negative term $-\frac{1}{2}\mathbf{z} \cdot \mathbf{z}$.

For the cluster scoring function $S(\mathbf{x}_1, \ldots, \mathbf{x}_N) = \sum_{i=1}^{N} \max[s_2(\mathbf{x}|\mathbf{z}), 0]$, the maximal cluster score is

$$S_{\max}(\mathbf{x}_1, \ldots, \mathbf{x}_N) = \max_{\mathbf{z}} S(\mathbf{x}_1, \ldots, \mathbf{x}_N|\mathbf{z}) \ . \quad (4.26)$$

Using the same method as before (full calculation shown in Appendix C), we obtain the free energy function,

$$-\beta f_2(\beta) = \min_a \max_{q \geq 0} \left[-\beta f_c \left(\beta q, \frac{q^2 - a}{2q} \right) + \frac{M}{2N} \log \left(\frac{a + \beta q^2}{a} \right) \right] \ . \quad (4.27)$$

The new parameter $q = \sqrt{\langle \mathbf{z} \cdot \mathbf{z} \rangle / M}$ is the average length of replicated cluster centres. In the previous result, q is strictly fixed to 1, compare Eq. (4.15). With entropy

$$\omega_2(s) = -\max_\beta [\beta s + \beta f_2(\beta)] \ , \quad (4.28)$$

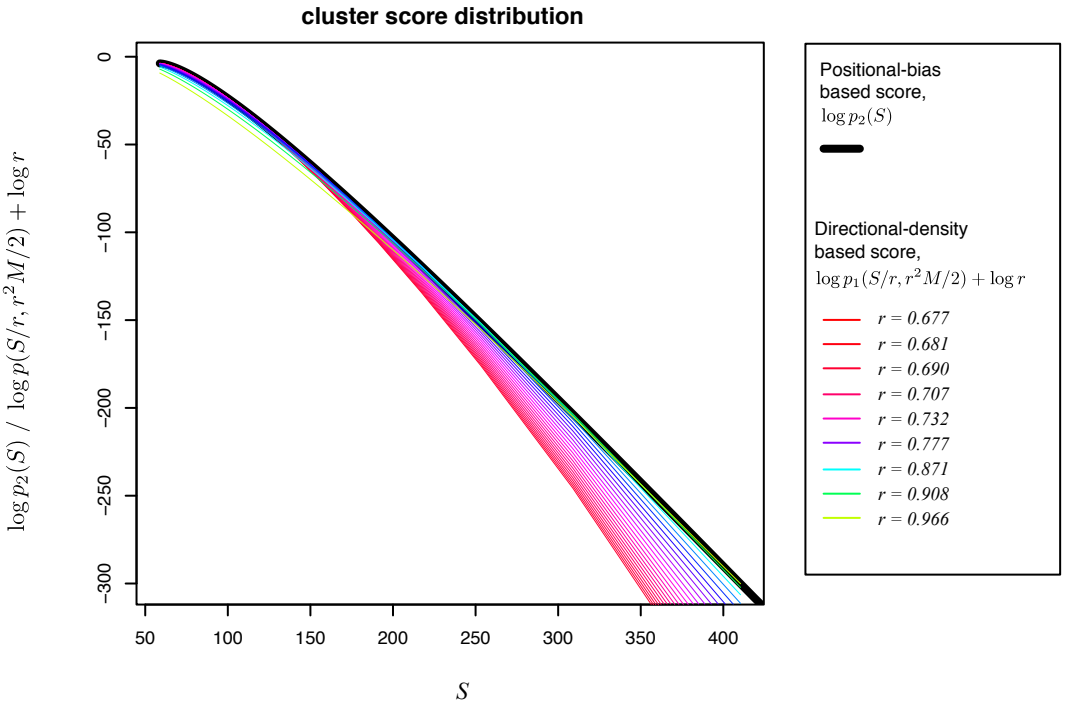

Figure 4.5: Relation between distributions of the maximal cluster score under the positional-bias and the directional-density based models (see text).

the cluster score p-value of the maximal score is

$$\log p_2(S) = N\omega_2(S/N) + \mathcal{O}(N) \ . \tag{4.29}$$

Relation to statistics of clusters based on directional density. The statistics of the position-bias-based score are very similar to the statistics of the directional-density-based score. The free energy function of the former, Eq. (4.27) incorporates the additional parameter q, which is strictly set to 1 in the latter, Eq. (4.15). Parameter q can be interpreted as the length of the optimal cluster centre $\mathbf{z}$. Keeping this in mind, we can decompose the cluster centre $\mathbf{z}$ into the direction part and the length part, $\mathbf{z} = q\hat{\mathbf{z}}$ with $q = \sqrt{\mathbf{z}\cdot\mathbf{z}/M}$ and $\|\hat{\mathbf{z}}\| = \sqrt{M}$. Score $s_2(\mathbf{x}|\mathbf{z})$ (4.3) can then be expressed in terms of score $s_1(\mathbf{x}|\mathbf{z},\mu)$ (4.2) as

$$s_2(\mathbf{x}|\mathbf{z}) = q\,\mathbf{x}\cdot\hat{\mathbf{z}} - q^2M/2 = qs_1(\mathbf{x}|\hat{\mathbf{z}}, qM/2) \ . \tag{4.30}$$

The probability $p_2(S)$ can be expressed in terms of the probability $p(S,\mu)$ as

$$\log p_2(S) = \log p_1(S/q, qM/2) + \log q \ . \tag{4.31}$$

What is thus the relation between statistics of these two scoring schemes?

Intuitively, under the positional-bias-based scoring scheme, in the search for the maximal scoring cluster, the optimisation is performed over the unconstrained cluster

centre $\mathbf{z}$, i.e. both over its direction $\hat{\mathbf{z}}$ and length q. In case of the directional-density based scoring scheme, the cluster centre is constrained by length, and the optimisation is performed solely over the direction $\hat{\mathbf{z}}$.

Hence, given the relation (4.31), the probability of score S under the positional-bias based model is always greater than or equal to the corresponding probability under the directional-density based model,

$$\forall r \geq 0, \quad \log p_2(S) \geq \log p_1(S/r, r^2 M/2) + \log r \ . \tag{4.32}$$

The equality is met for $r = q$, the value that appeared to be the optimal cluster length for score S, in Eq. (4.27).

In Fig. 4.5 we plot the maximal cluster score distribution $\log p_2(S)$, with $M = 20$ and $N = 40$, and compare it to the family of distributions $(\log p_1(S/r, rM/2) + \log r)$, parameterised by r ranging from 0.67 to 1. The curve given by $\log p_2(S)$ forms an "envelope" for the other curves, which are tangent to $\log p_2(S)$ at specific score values, but never cross it. The decay of $\log p_2(S)$ is slower than the decay of any $\log p_1(S/r, r^2 M/2) + \log r$.

4.4 Cluster score statistics and extreme value theory

It is instructive to compare cluster score statistics with distributions of maxima known from extreme value theory. Given a set of random numbers drawn *independently* from some probability distribution $p_0(S)$, extreme value theory describes the statistics of the maximum (or minimum) of these numbers. Depending on the tail of $p(S)$, the distribution of the extremum falls into one of three possible universality classes [48, 39, 15]:

1. **Gumbel type**: unbounded distribution $p_0(S)$ with tail decaying faster than any power k in S^k. The probability density function of the Gumbel distribution is

$$f(S, \mu, \beta) = \frac{1}{\lambda} e^{-\frac{1}{\lambda}(S-k)} e^{-e^{-\frac{1}{\lambda}(S-k)}} \ , \tag{4.33}$$

   $\lambda > 0$, which for large S is roughly exponential is S,

$$f(S, \mu, \beta) \approx \frac{1}{\lambda} e^{-\frac{1}{\lambda}(S-k)} \ . \tag{4.34}$$

2. **Fréchet type**: tail of $p_0(S)$ follows the power-law decay.

   The Fréchet distribution is

$$f(S) = \frac{k}{\lambda} \left(\frac{\lambda}{S} \right)^{k+1} \exp^{-\left(\frac{\lambda}{S}\right)^k} \ , \tag{4.35}$$

   with $k, \lambda > 0$.

3. **Weibull type**: distribution $p_0(S)$ is strictly bounded.

   The Weibull distribution is

$$f(S,\lambda,k) = \begin{cases} \frac{k}{\lambda}\left(\frac{S}{\lambda}\right)^{k-1} e^{-\left(\frac{S}{\lambda}\right)^k} & S \geq 0, \\ 0 & S < 0\ , \end{cases} \tag{4.36}$$

   which for large S is dominated by the exponential part, $e^{-\left(\frac{S}{\lambda}\right)^k}$.

For example, the maxima of sequence alignment scores are described by the Gumbel class [57, 4].

In section 4.2, we characterised the distribution $p_c(S,\mu)$ (4.10) of the fixed direction cluster score. The tail of this distribution decays like $e^{-\lambda S^2}$ (i.e. faster than power law), which we checked numerically. A rationale behind the Gaussian-like behaviour of the tail (i.e. distribution of *large scores*) follows from the central limit theorem: the score is a sum of identically distributed contributions of many elements. (This reasoning may not valid for small score values, characteristic to clusters with a smaller number of elements.)

We will now examine whether our solutions for the distribution of the maximal cluster score fall into any of the above described classes of extreme value statistics.

Deviations from the assumptions of extreme value statistics. The statistics of maximum cluster score appears to be quite different from the assumptions of extreme value theory. The latter assumes independent draws from the distribution $p_c(S)$. In sequence alignment, for example, this is justified, since high-scoring islands are well separated from each other. There is no such separation in clustering: any data vector can potentially contribute to many clusters in the dataset. This leads to a higher chance of overlap between clusters and, hence, to *correlated scores*.

Such overlaps are most pronounced in high-dimensional data spaces. As an extreme case, consider two random, independently drawn cluster directions $\hat{\mathbf{z}}_1$ and $\hat{\mathbf{z}}_2$. Such random vectors are, in a typical case, orthogonal (scalar product has expectation value zero), with a variance anti-proportional to the number of dimension M, by the law of large numbers. Taking $\mathbf{x} = \sqrt{M/2}\,(\hat{\mathbf{z}}_1 + \hat{\mathbf{z}}_2)$ one finds $\mathbf{x}\cdot\hat{\mathbf{z}}_1 = \mathbf{x}\cdot\hat{\mathbf{z}}_2 = \sqrt{M/2}$ to leading order in M. Thus, provided $\mu < \sqrt{M/2}$, there exists a vector $\mathbf{x}$ which is an element of both clusters: clusters with random directions can overlap. By the same token, for a pair of randomly picked vectors one can find a cluster direction such that both vectors are members of that cluster.

An additional deviation arises in the positional-bias based model. Here, the statistics of the single cluster score are given by $p_c(S, \mathbf{z}\cdot\mathbf{z}/2)$ for a given centre $\mathbf{z}$. As such, the distribution of the cluster score depends on the length of the cluster centre. In transition from one to many clusters, we encounter cluster centres of different lengths, and hence the clusters scores are not identically distributed.

A well known example of divergence from the extreme value statistics, due to deviation from the i.i.d assumption, is known to arise in principal component analysis, where the maximal eigenvalue of the covariance matrix follows a Tracy-Widom distribution [55]. Below we investigate how strong is the effect of these deviations on the maximal cluster score statistics. We consider distributions of the two classes of the cluster score discussed in this chapter: (i) in a model based on directional density (4.17) and (ii) in a model based on positional bias (4.29).

Agreement with extreme value statistics in the limit of large scores. Our solutions, i.e. distribution $p(S, \eta)$ (4.25) of the cluster score under the directional-density-based model and distribution $p_2(S)$ (4.29) of the cluster score under positional-bias-based model, are not given in a closed form. We checked numerically the asymptotic behaviour of these curves. Our observations, which we illustrate with examples obtained with $M = 20$, $N = 40$ and $\eta = 1$, were consistent, independently of the choice of the parameter values.

The two cases show different statistics. The maximal cluster score under the positional-bias-based model becomes asymptotically exponential in $-S$, i.e. it agrees with the asymptotics of the Gumbel distribution. In Fig 4.6, we plot this distribution for $M = 20$ and $N = 40$ and a fitting Gumbel distribution ($\lambda = 1$ and $k = 92.965$). However, the fit becomes valid for very large scores: as can be read from the figure, it is still not close at $p_2(S) = e^{-200}$. The expected score values, i.e. the maxima of the two plots, do not coincide. Practically, for low and medium score values the Gumbel distribution gives an over-conservative p-value approximation.

The distribution of the cluster score under the directional-density-based model is decaying faster, like $e^{-(S/\lambda)^2}$, which places it in the Weibull class. We fitted the Weibull distribution to the tail of this distribution, fixing parameter $k = 2$; we obtained $\lambda = 19.51$. On the large scale, the two distributions converge (see bottom of Fig 4.6), however the fit again becomes valid only for very large scores, with the "pre-asymptotic" limit even wider than in the previous case.

Strong pre-asymptotic corrections. In both cases, the correlations between scores of overlapping clusters have a strong effect on the statistics of the scores encountered in real-life applications. Cluster scores can be claimed significant already at p-values around 10^{-3}, which is far above the point where the extreme-value statistics approximations become relevant. The pre-asymptotic limit, in which the extreme-value statistics was not giving good fit, has a very wide score range.

4.5 Resampling-based methods

The cluster significance problem has been addressed by so-called *resampling-based methods*. The general idea is to repeatedly apply the clustering algorithm on a per-

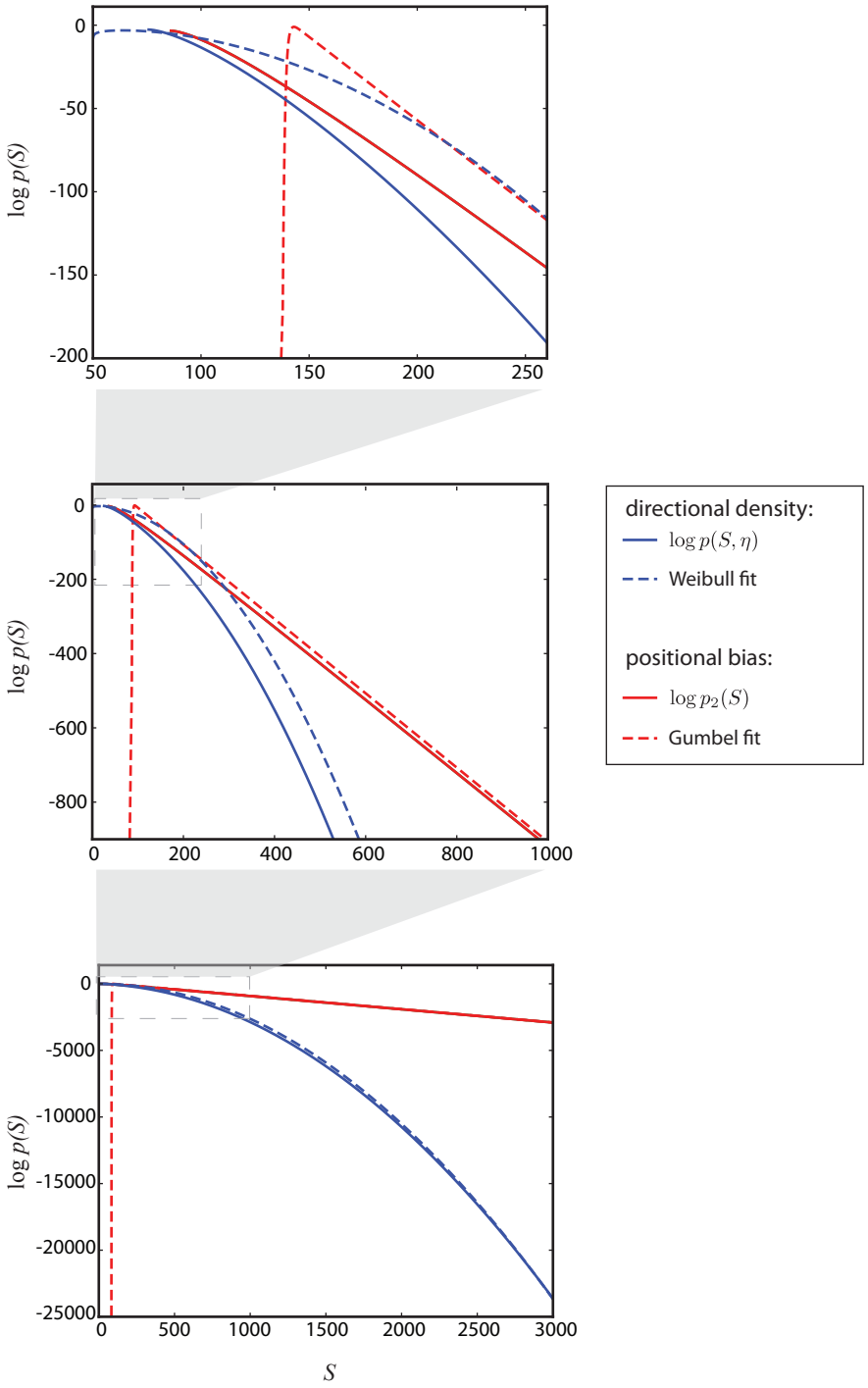

Figure 4.6: The maximal cluster score distributions and extreme value statistics. We plotted the score distribution for $M = 20$ and $N = 40$, under the positional-bias-based model (red) and under the directional-density-based model, with $\eta = 1$ (blue). The Gumbel and the Weibull fits are plotted with dashed lines using the respective colours. The top to bottom plots show a gradual zoom-out of the same curves.

turbed data and then to compute a *stability score* for the series of the resulting partitions. In some of these approaches, the stability score is computed globally for the whole set of clusters and it is used to guide the clustering parameter choice. In other approaches, a stability score is computed individually for each cluster and, similarly as the p-value, can be used to disregard unstable clusters from the set of clusters. Below we describe some exemplary methods of both types.

Roth et. al [86] perform a series of iterations, $i = 1, \ldots, T$, in which they split the data $\mathbf{X}$ into two parts of the same size, $\mathbf{X}_1^i$ and $\mathbf{X}_2^i$, and cluster them with the same algorithm independently. This procedure returns partition of the two sets, P_1^i and P_2^i respectively. Subsequently, the elements of $\mathbf{X}_1^i$ are being assigned to clusters from P_2^i. The classification is done based on similarity to the elements in a cluster. This procedure results in another partition of data $\mathbf{X}_1^i$, which we denote by P_{cl}^i. Partitions P_1^i and P_{cl}^i can now be compared, with a method of choice (for example the Rand index [50]). The more similar the two partitions, the more stable the clustering is. The total stability measure is computed as an average over many iterations of this procedure.

In the approach by Levine and Domany [64], the clustering algorithm is first applied to the full dataset $\mathbf{X}$ resulting in a data partition P. In subsequent iterations, a subset of the data is chosen $\mathbf{X}^i \subseteq \mathbf{X}$ to which the algorithm is applied with the same parameters as in the case of the full dataset. The similarity of partitions P and P^i is then computed with a specially designed measure; the total stability is again computed as the average over many iterations.

Suzuki and Shimodaira [99, 100] propose a bootstrapping method for assessing significance of dendrograms of hierarchical clusterings. The method is based on previously developed methods for resampling of phylogenetic trees [28, 29, 32]. As in the previously discussed methods, bootstrap samples are generated in many iterations. In each of the iterations a bootstrap replicate of the dendrogram is created by applying the hierarchical clustering algorithm on the sample. The p-value of each cluster is then computed based on the frequency with which the cluster appears in the bootstrap replicates.

The bootstrap-based approaches do not assume any explicit background model and as such are non-biased and can capture non-trivial dependencies in data. On the other hand, the *stability* is a heuristic measure and there is no clear meaning associated with the value of the threshold. The drawback of these methods is also their computational complexity, which can be seriously prohibitive in application to large datasets.

In section 4.6.2, we will show a comparison between our cluster score p-value and a resampling based approach, in application to clusters in yeast gene expression data.

4.6 Application to gene expression data

4.6.1 Gene-expression data preprocessing.

In the remaining part of this Chapter and also in Chapters 5 and 6, we will show examples of application of our methods to real gene expression datasets. In all cases we use the same preprocessing scheme:

1. The gene expression dataset is log-transformed.
2. The gene vectors are mean-centred, $\mathbf{x}_i \leftarrow \mathbf{x}_i - \sum_{\mu=1}^{M} x_i^\mu / M$. In this way, the average over all experiments plays a role of a control sample and for a given gene we consider only its deviations from its typical expression level.
3. Subsequently, also the experiment vectors are mean-centred, $\mathbf{x}^\mu \leftarrow \mathbf{x}^\mu - \sum_{i=1}^{N} x_i^\mu / N$, which is a standard transformation: in principle the overall average expression should be the same in all experiments and possible deviations are due to experimental or technical noise.

4.6.2 Yeast expression data under environmental shock conditions

Clusters with high statistical significance may contain elements with a common mechanism responsible for their similarity. Here, we test the link between our p-value and biological function of clusters in a dataset of gene expression in yeast [42].

The dataset contains expression levels from 173 samples for $N = 6152$ genes. We used the standard preprocessing described earlier in section 4.6.1. Gene expression vectors were then length-normalised, using a weighted metric which accounts for dependencies between experiments. The method for metric estimation is described in detail in Chapter 5. In the following analysis, we used the directional-density-based scoring scheme for clusters and the corresponding cluster score p-value $p(S, \eta)$ from Eq. (4.25).

Comparison with a resampling-based approach. We compare the analytical score p-value to the stability measure obtained with a bootstrap-based approach. The question is whether the analytical approach, with a defined background model, can capture the same dependencies in data as an unbiased bootstrapping method.

To evaluate stability of individual clusters, in line with our analytical method, we adapt a resampling-based approach of Levine and Domany [64]. The input to the problem is a clustering result $\mathbf{C}$, which partitions data into K clusters. The method proceeds in many iterations, the steps within an iteration are as follows:

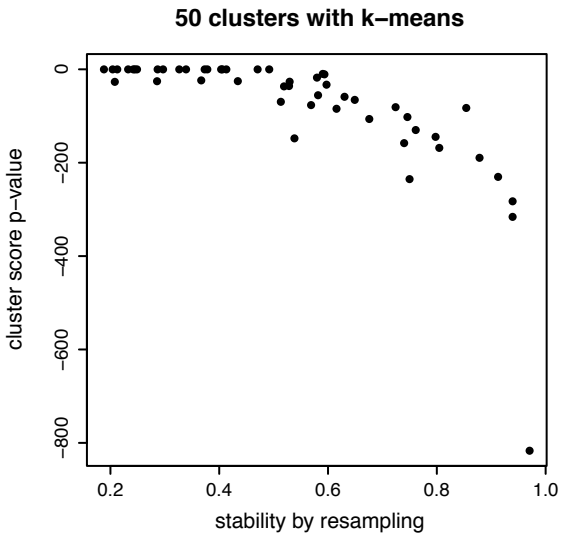

Figure 4.7: Cluster significance and stability after resampling: application to yeast expression data [42]. The statistical significance of a cluster is plotted against its stability under resampling, (see text). The plot shows the average stability over many realisations of the resampling.

1. Take a random sample of $N/2$ data vectors, run the clustering algorithm to find clustering $\mathbf{C}^i$ with K clusters.

2. Considering only elements in the random sample, map "sub-clusters" from $\mathbf{C}$ (i.e. clusters formed by elements from clusters from $\mathbf{C}$ after removing elements not present in the random sample) to clusters in P^i. The quality measure for each cluster is the fraction of elements that are shared by a sub-cluster from P and its mapped cluster from $\mathbf{C}^i$.

We expect that highly significant clusters are less affected by the removal of vectors than the insignificant ones. We clustered the yeast data using the k-means algorithm and compared cluster score p-values to their stability. In Fig. 4.7, we plot these two quantities for an example with $k = 50$ clusters (the same trend was observed for other choices of k). As expected, the plot shows that just as about half of the remaining vectors are still clustered under bootstrapping; the cluster starts becoming significant. The link between significance and stability is then maintained up to very high stabilities. Scatter stems from instances of cluster pairs that are close to one another, and so vectors are assigned to different clusters under resampling. To include this effect into our p-value results, multiple cluster centres would have to be considered in our analytical approach.

Statistical significance is correlated with functional relevance. The probabilistic description for the clustering problem was motivated by underlying biology of gene expression data. In particular, the hypothesis was that clusters of similar gene expression profiles reflect an underlying regulatory mechanism which relates the

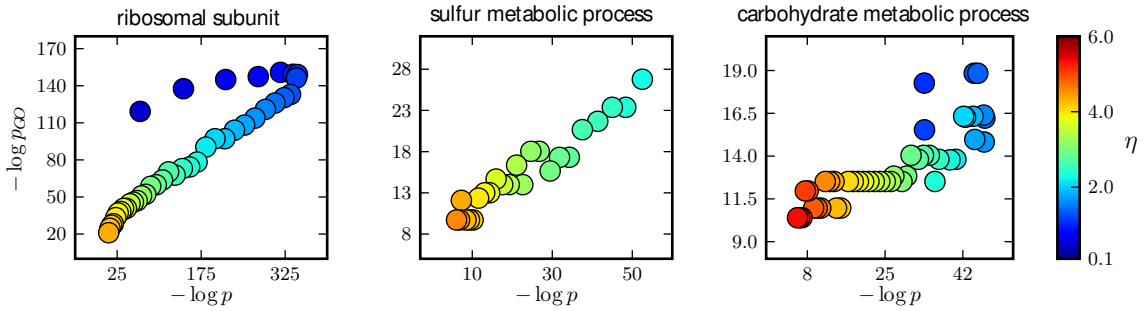

Figure 4.8: Statistical significance of clusters is correlated with functional annotation for yeast expression data. The diagrams show the significance $-\log p_{\mathrm{GO}}$ of gene annotation terms vs the cluster score significance, $-\log p(S,\eta)$, traced over a range of the scoring parameter η (shown by the colour scale) of three representative clusters involved in translation(ribosomal genes), the sulphur metabolic process, and the carbohydrate metabolic process.

corresponding genes. Is the cluster-score significance related to its biological relevance?

To answer this question, we quantify "biological relevance" of a cluster by its enrichment in gene ontology GO-terms. We compute the p-value $p_{\mathrm{GO}}(C)$ of the most significantly enriched GO-term in a cluster C, using parent-child enrichment analysis [47] with a Bonferroni correction. A cluster with small $p_{\mathrm{GO}}(C)$ is thus significantly enriched in at least one GO-annotation, which points to a functional relationship between its genes.

We trace several high-scoring clusters over the range of η where they give a positive score. As η decreases, the cluster opening-angle decreases, leading to a tighter, smaller cluster. The cluster p-value also changes continuously, and the genes contained in the cluster also change. We ask if specific functional annotations (gene ontology GO-terms) appear repeatedly in the genes of a cluster, and how likely it is for such a functional enrichment to arise by chance. As shown in Fig. 4.8, the parameter dependence of the cluster score significance $p(S(C))$ and the significance $p_{\mathrm{GO}}(C)$ of gene annotation terms, is strikingly similar. In particular, the parameter minimising the score p-value also produces a low GO p-value. The statistical measure based on cluster score p-values thus is a good predictor of functional coherence of its elements.

4.6.3 Gene co-expression and ageing in mice

An interesting application of the cluster score p-value is the assessing of *differential gene co-expression*: changes in co-expression clusters, as observed for the same set of genes on two different sets of experimental conditions, for example normal and tumour tissues [59, 21]. The usual problem of cross-dataset comparisons concerns the dependence of scoring functions on dataset-specific parameters. For example, a

given value of the correlation or Euclidean distance has different significance in a high and a low dimensional dataset. For the same reason, we cannot compare scores of clusters from two different datasets. The p-value, to the contrary, is a parameter-free quantity, and as such allows for an unbiased comparison between datasets.

Southworth et al. [93], investigate how co-expression of genes is affected by ageing. The claim of the study is that *pairwise* correlations of gene expression patterns decline with age. Motivated by their result, we ask about changes in co-expression at the level of *clusters*.

The AGEMAP data [110] is a collection of genome-wide microarrays from 17 different tissues in mice at different ages. Every tissue and age is represented by multiple biological replicates. We analysed expression of 1, 16 and 24 months old mice. The data was downloaded from NCBI Gene Expression Omnibus (accession code GSE9909). We formed three separate sets, for each of the three ages. We normalised the data with the standard approach described in section 4.6.1. Vectors in each dataset were then length-normalised, using a weighted metric which accounts for dependencies between data components (here tissues). The method for metric estimation is described in detail in Chapter 5. In the following analysis, we were using the directional-density-based cluster scoring scheme with an appropriate version of the cluster score p-value (4.25).

In our analysis, we obtained a partition of genes into clusters by running a clustering algorithm on the first dataset of the 1-month-old mice. We used k-means algorithm to find $k = 40$ clusters. We then analysed and compared the behaviour of such obtained groups of genes on the two other datasets of the 16- and 24-month-old mice. The clustering, we stress, was optimised with respect to the first dataset only. In particular, a group of genes clustered in the first dataset need not be clustered in the second or third dataset.

For each of such obtained groups of genes, we found the maximum-likelihood parameters η and $\mathbf{z}$ in each of the three datasets. We then computed their scores and the cluster score p-values. Thus, for one group of genes (clustered on the first dataset), we obtained three cluster score p-values, for each of the datasets. As turned out, the groups of genes identified as clusters on the first dataset, were also significantly clustered on the 16- and 24-months-old mice datasets. Moreover, the cluster score p-values showed a very good correlation between all three datasets, see Fig. 4.9 (a).

We then compared significance of clusters at age 16 and 24 months. Again, as the clusters were obtained on the separate dataset of 1-month-old mice, no bias was introduced towards tighter correlations of cluster elements in any of the other two datasets. We found that clusters at old age show a consistent decrease of significance, in line with the previously reported result [93], Fig. 4.9 (b). The decrease of significance is caused by an overall decrease in pairwise similarities (here correlations) between members of a cluster. We illustrate this effect on an exemplary cluster in Fig 4.9 (c).

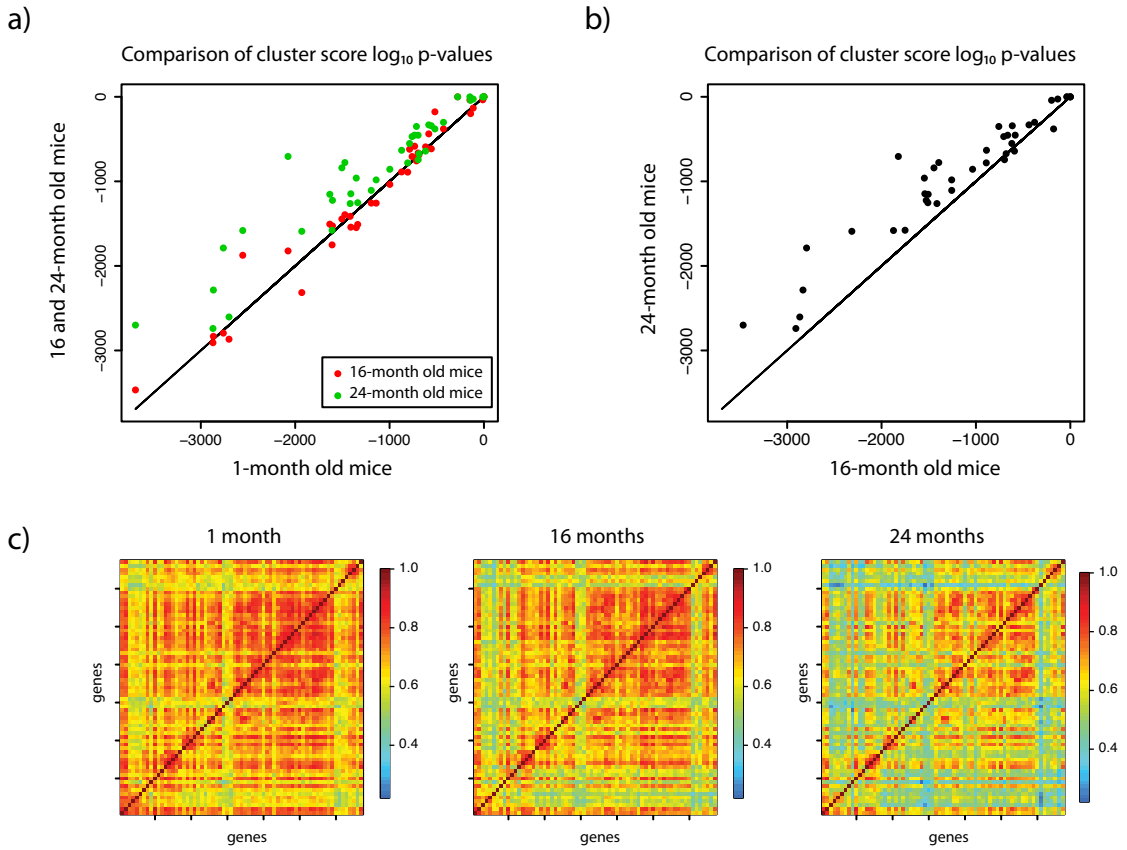

Figure 4.9: Gene clusters in mouse expression data are becoming less significant with age. We have clustered gene expression data of 1-month-old mice into 40 clusters. The resulting clusters were then "projected" on the other two datasets of 16- and 24-month-old mice. The scatter plots show cluster score p-values of the clusters. a) Clusters are well conserved between the 3 datasets, showing a very good correlation of cluster score p-values, here we show the relation between significance of clusters on the original dataset to their significance on the 16- and 24-months datasets. b) Clusters in the 24-months dataset are consistently less significant than clusters in the 16-months dataset, pointing at decreasing co-expression between genes with age. c) Example of a cluster and the pairwise similarity map (Pearson correlation) of its members, in the 1-month (left), 16-months (middle) and 24-months (right) datasets. The correlations are weakening with age, as compared between 16- and 24-months.

4.7 Summary

Using a mapping to a problem of disordered systems from the statistical mechanics, we derived the distributions of the cluster score for two different scoring schemes: based on the directional-density and on the positional bias. These scoring schemes are characterised by a linear dependence on data elements $\mathbf{x}_i$. The solution for the quadratic score would involve performing another replica-method calculation, this time with a quadratic scoring function $s(\mathbf{x}|\mathbf{z}, \eta)$. In principle, the calculation would be similar to the ones presented here, but most likely would involve more order parameters.

Our result provides a conceptual and practical improvement over current methods of estimating p-values by simulation of an ensemble of random data sets, which are computationally intensive [100] and, hence, often omitted in practice. The solutions we presented appeared not to follow any of the known universality classes of the extreme value statistics. For larger score values S, we did observe a convergence to the asymptotics of the Gumbel distribution (the directional-density-based model) and to the Weibull distribution (the positional-bias-based model). However, the agreement was reached only for very high scores, far above the scores encountered in practical applications. The "pre-asymptotic" statistics of the cluster score are governed by non-trivial correlations between clusters in data and are correctly described by our solutions.

In application to real gene expression data, we compared the analytically computed cluster score p-values to the "stability" of clusters, as estimated with a numerical, resampling-based approach. The comparison showed a good agreement between the two measures. The power of the resampling method is that it does not assume any underlying data statistics. The drawback is that it requires many iterations of clustering on resampled datasets, which is very costly and practically not applicable to large, genome-wide datasets. The agreement between the analytically computed cluster score p-values and the cluster stabilities suggests validity of the proposed null models for modelling of gene expression.

The p-value of a cluster score S tells how likely it is to observe by chance a cluster with score S. Similarly to the score, the p-value quantifies the cluster "quality". Dissimilarly from the score, the p-value does not depend on any parameters and it can be compared between clusters of different width or formed on datasets of different dimensionality. We showed an application of the cluster score p-values for cross-dataset comparisons of clusters, here the co-expression across tissues in mice at different age.

In an application to yeast data, the p-value appeared to reflect the biological significance of co-expressed genes. We exploit this correlation later, in the *significance-based clustering algorithm* discussed in Chapter 6.

Chapter 5

Estimating dependencies between experimental conditions

In this chapter, we discuss another important aspect of the statistics of high-dimensional data, which is an "orthogonal" problem to clustering of data vectors: dealing with dependencies between the components of data vectors. Such dependencies are prevalent for experimental conditions in gene expression data, for example between subsequent time points in time-course experiments. The correct estimation of such dependencies is crucial for clustering of experimental conditions, for example in the task of a tumour sample classification. Moreover, the estimation also affects computation of similarities between data vectors and hence their clustering. Here, we show that the estimation of the *vector component dependencies* requires accounting for an important confounding factor: the presence of *clusters of data vectors*. We apply our method in the problem of tumour sample classification.

5.1 Motivation

Real data often contains intrinsic dependencies between components of data vectors (later referred to as *data components*). Gene expression experiments are a perfect example: in time course data, the subsequent experiments record gene activity in close time intervals. If a gene is active and highly expressed in the first time point, its expression is likely to be observed also in the second time point, see Fig. 5.1. Expression profiles of evolutionarily related tissues, such as the liver and the kidney, are more similar then those of tissues of a more distant common origin, such as the liver and the brain. Proper estimation of such correlations is important for clustering of data elements: the similarity measure should be able to down-weigh the signal coming from related components and properly count the information content.

In some applications, one is also interested in grouping of the data components themselves. A very common example is cancer classification. Tumours originate from normal tissues by the process of several consecutive mutations in oncogenes or tumour suppressors. Mutations are broadly understood changes in a cell, they can be of various kind: genetic or epigenetic, simple single point mutations and copy number

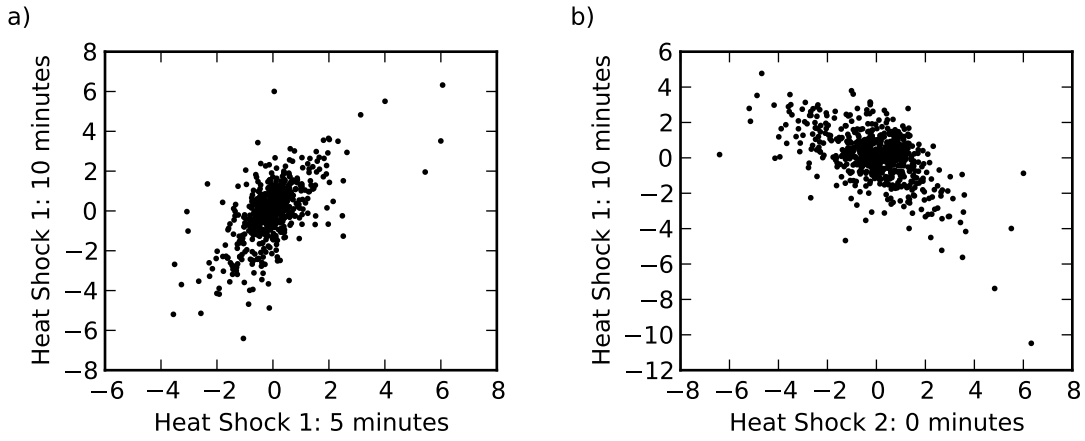

Figure 5.1: Correlation of experiments in yeast expression data. These scatter plots show two examples of pairwise experiment dependencies in gene expression data. Every dot corresponds to a gene and its position depends on the relative expression level in the first (x-axis) and the second (y-axis) experiment.

variations, translocations, microsatellite expansions or chromosomal abnormalities. As a result of such mutations, the functioning of a cell changes, which in turn is reflected in a change of its gene expression levels. Microarray experiments have been used to measure gene expression in tumour samples; many studies based on simple clustering approaches have proven successful in determining types, subtypes and the origin of the tumour samples based on this kind of data [46, 58, 61, 85, 97, 106].

Of course, such clusterings depend on the similarity measure for the tumour samples. In this chapter, we discuss the pitfalls of the task of estimating correlations of data components. We show that the correct estimation of such dependencies needs to account for presence of clusters of data elements. We propose a solution based on a mixture-model, which uses the probabilistic models of clusters and the background data presented in Chapter 3.

5.2 Covariance and correlation matrix

In our approach, we focus on linear dependencies between data components. Assume there are M vector components represented by M random variables, $\mathbf{x}^1, \ldots, \mathbf{x}^M$. We denote the joint probability density function of all variables by $f(\mathbf{x})$, the pairwise joint probability density function of variables $\mathbf{x}^\mu$ and $\mathbf{x}^\nu$ by $f_{\mu\nu}(x^\mu, x^\nu)$, and the probability density of a single variable $\mathbf{x}^\mu$ by $f_\mu(x^\mu)$.

The *covariance matrix* $\mathbf{G}$ quantifies pairwise-dependencies of all pairs of data com-

ponents and is defined by

$$G^{\mu\nu} = \mathrm{cov}(\mathbf{x}^\mu, \mathbf{x}^\nu) = \mathbb{E}\left[(\mathbf{x}^\mu - \bar{\mathbf{x}}^\mu)(\mathbf{x}^\nu - \bar{\mathbf{x}}^\nu)\right] = \int_{\mathbb{R}}\int_{\mathbb{R}} f_{\mu\nu}(x^\mu, x^\nu)(x^\mu - \bar{x}^\mu)(x^\nu - \bar{x}^\nu)dx^\mu dx^\nu \ , \tag{5.1}$$

where $\bar{x}^\mu = \mathbb{E}[x^\mu] = \int_{\mathbb{R}} f_\mu(x^\mu)x^\mu dx^\mu$ is the expected value of variable $\mathbf{x}^\mu$.

If data components $\mathbf{x}^\mu$ and $\mathbf{x}^\nu$ are statistically independent, the covariance $G^{\mu\nu}$ is zero.

Correlation is the normalised covariance,

$$\hat{G}^{\mu\nu} = \mathrm{cor}(\mathbf{x}^\mu, \mathbf{x}^\nu) = \frac{G^{\mu\nu}}{\sqrt{G^{\mu\mu}G^{\nu\nu}}} \ . \tag{5.2}$$

The correlation takes values in the range $[-1, 1]$.

Spectral decomposition. If all variables $\mathbf{x}^1, \ldots, \mathbf{x}^M$ are linearly independent, the covariance matrix $\mathbf{G}$ can be factorized,

$$\mathbf{G} = \mathbf{D}\Lambda\mathbf{D}^{-1} \ , \tag{5.3}$$

where $\mathbf{D}$ is a matrix of orthogonal eigenvectors and Λ is a diagonal matrix of eigenvalues, $\lambda_1, \ldots, \lambda_M$. The eigenvectors are directions of variance in the data. If the covariance matrix is diagonal, i.e. the original data components are independent, the eigenvectors are given by the original coordinate system. The eigenvalues "tell" how much variance there is in the corresponding eigen-directions. As such, they describe the "shape" of the data in the space in which the data points are located. In Fig. 5.2, we show three examples of data in two-dimensional space, characterised by covariance matrices of different types of the spectral decomposition.

5.2.1 Sample data covariance and correlation

We now focus on the problem of estimating the covariance matrix from data with N data vectors, $\mathbf{X} = \{x_i^\mu | i = 1, \ldots, N, \mu = 1, \ldots, M\}$.

The standard and unbiased way of estimating the data covariance is to compute the *sample covariance matrix*,

$$G^{\mu\nu} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i^\mu - \bar{x}^\mu)(x_i^\nu - \bar{x}^\nu), \tag{5.4}$$

where the mean component values are also estimated by $\bar{x}^\mu = \sum_{i=1}^N x_i^\mu/N$. This estimator is, for large N, equivalent to the maximum-likelihood estimation with an

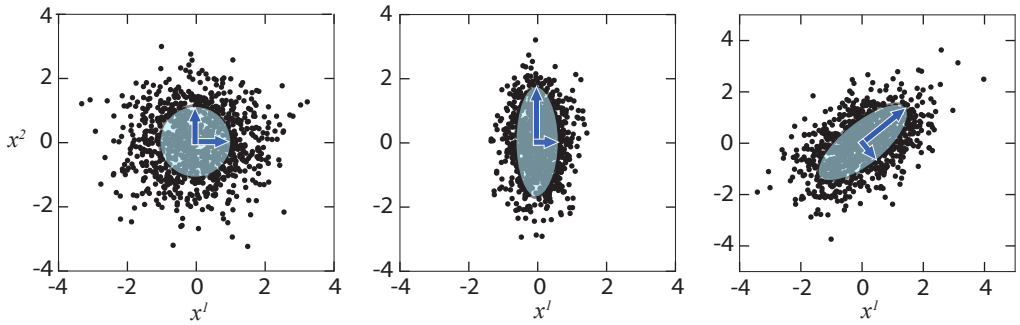

Figure 5.2: Covariance examples and the spectral decomposition. In the scatter plots we show examples of Gaussian distributed data with different covariance of its components. (left) data components $\mathbf{x}^1$ and $\mathbf{x}^2$ are i.i.d., the covariance matrix is diagonal, with eigenvectors given by the original coordinates and equal eigenvalues. (middle) data components $\mathbf{x}^1$ and $\mathbf{x}^2$ are independent but not identically distributed, the covariance matrix is still diagonal, the eigenvectors are given by the original coordinate system but the eigenvalues are unequal. (right) data components $\mathbf{x}^1$ and $\mathbf{x}^2$ are not independent and the eigenvectors differ from the original coordinate system.

underlying Gaussian model, $f(\mathbf{x}) = (\det(\mathbf{G})2\pi)^{-\frac{M}{2}}\, e^{-\frac{1}{2}\mathbf{x}\cdot(\mathbf{G}^{-1})\cdot\mathbf{x}}$. The Gaussian model assumption leads to the estimate

$$G^{\mu\nu} = \frac{1}{N}\sum_{i=1}^{N}(x_i^\mu - \bar{x}^\mu)(x_i^\nu - \bar{x}^\nu)\ . \tag{5.5}$$

The difference to Eq. (5.4) is in N replacing $(N-1)$ in the denominator. Factor $(N-1)$ is due to the bias caused by the fact that the sample mean is also estimated.

The corresponding correlation matrix $\hat{\mathbf{G}}$, computed following Eq. (5.2) by normalising the sample covariance matrix $\mathbf{G}$, is called the *sample correlation matrix*. The sample data covariance and the sample correlation are commonly used to compute similarities between experiments in gene expression data [44, 94].

5.2.2 Sample covariance and spurious dependencies in data

As discussed in the previous section, the sample covariance estimator is, for large N, equivalent to the maximum likelihood estimator for an underlying standard Gaussian distribution of data components $\mathbf{x}^1, \ldots, \mathbf{x}^M$. In this respect, the presence of clusters constitutes a deviation from this model. What follows is that the sample covariance includes contributions from the *true dependencies* but also from the *spurious cluster-effect*, which in fact may dominate the direction of variance in data. In Fig. 5.3, we illustrate this effect with a background data and a cluster in two-dimensional space.

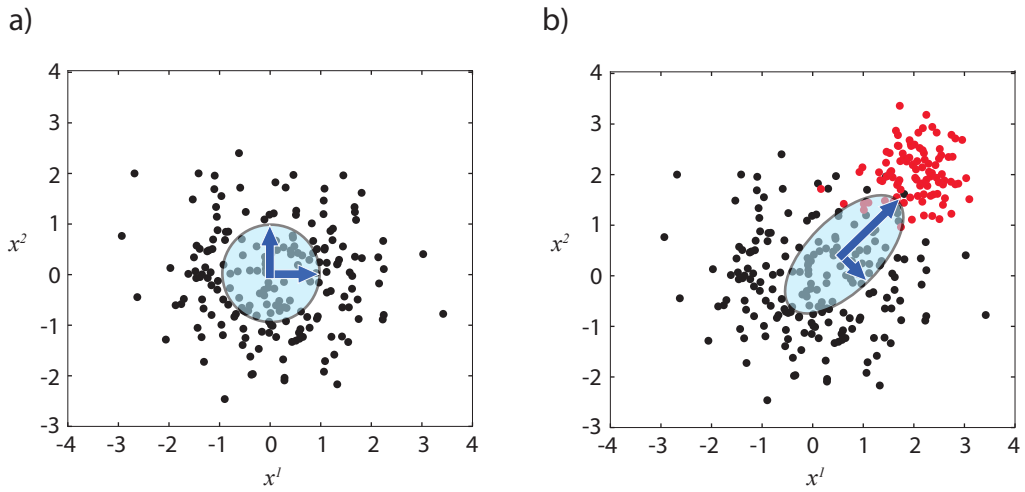

Figure 5.3: Sample covariance estimation in a presence of a cluster. Two dimensional data was generated from the standard Gaussian distribution. (a) The eigenvectors of the data covariance matrix are depicted with blue. (b) A cluster of points (red dots) changes the total data covariance: the largest variance in data is now in the direction of the cluster and the leading eigenvalue reflects the distance of the cluster centre to the centre of the background data.

Example: simulated data in a high-dimensional space. The presence of clusters dominates covariance estimates also in high dimensional data. To illustrate this effect, we generated $N = 1000$ vectors from a standard Gaussian distribution in $M = 60$ dimensions. We then estimated the sample covariance matrix, which was diagonal: the off-diagonal dependencies arose only due to random fluctuations and were insignificant. Similarly, all eigenvalues of the matrix were, up to small fluctuations, equal to 1, see Fig. 5.4 (a). Subsequently, a cluster of 200 vectors was inserted in the data: another Gaussian component with a randomly drawn centre $\hat{\mathbf{z}}_1$, normalised such that $\|\hat{\mathbf{z}}_1\| = \sqrt{M}$, and variance 1 in all directions. The sample covariance matrix was no longer diagonal: The presence of a cluster introduces spurious correlations between components in the direction of $\hat{\mathbf{z}}_1$. Similarly, the leading eigenvalue was no longer ~ 1; in turn, its value reflected the increased data variance in the direction of $\hat{\mathbf{z}}_1$. This effect is illustrated in Fig. 5.4 (b). In the last step we added a second cluster of 200 vectors with centre $\hat{\mathbf{z}}_2$ again chosen randomly, with a constraint $\|\hat{\mathbf{z}}_2\| = \sqrt{M}$, and with variance 1 in all directions. The new cluster generates another eigen-direction which dominates the total variance in data. The covariance matrix has even more significant off-diagonal dependencies. Now, the two leading eigenvalues significantly deviate from 1, reflecting presence of two clusters in data. The last example is illustrated in Fig 5.4 (c).

In the following part of this chapter, we will show how to account for the presence of clusters in data and estimate the true data-component dependencies.

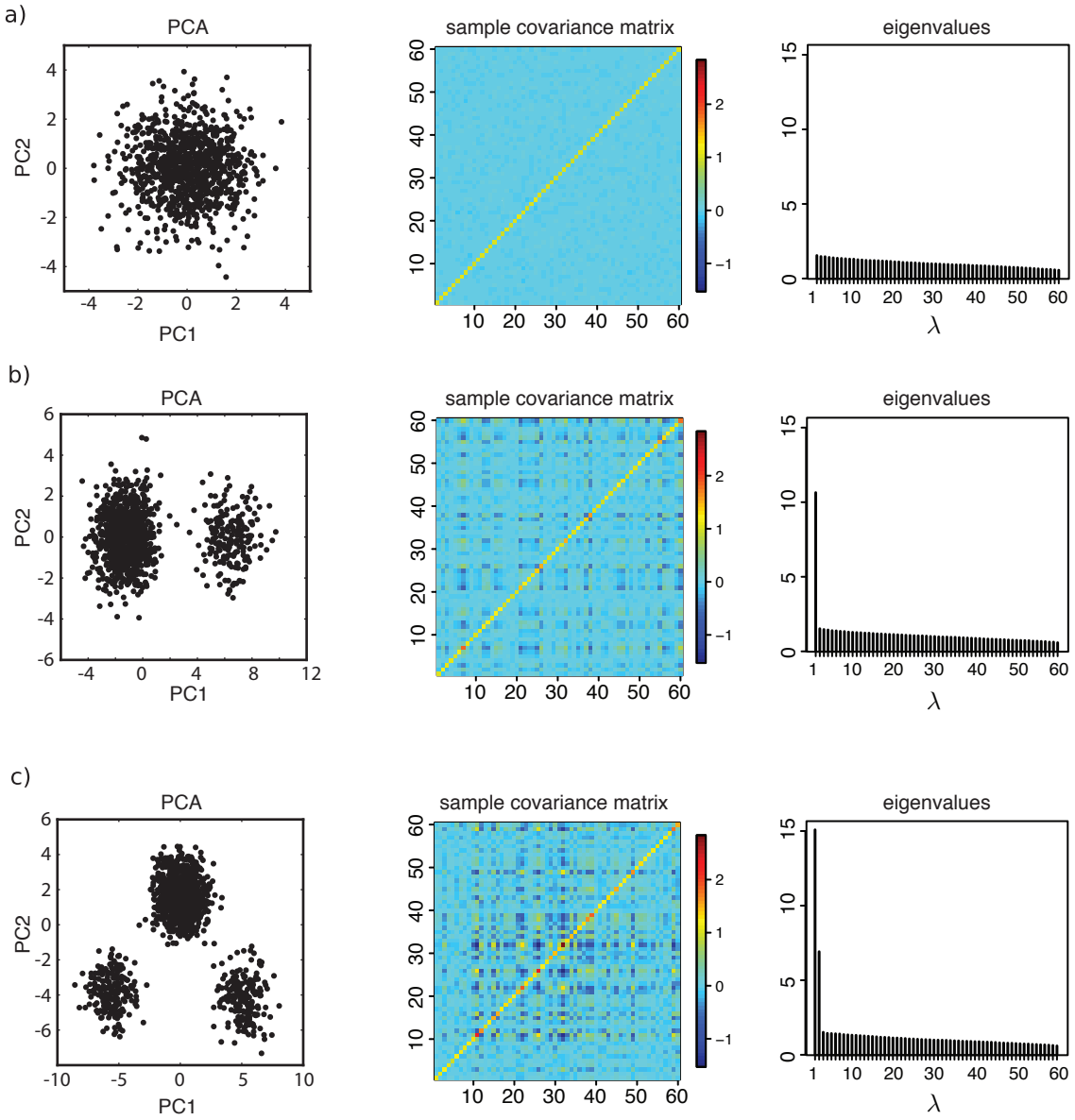

Figure 5.4: Covariance estimation in simulated data. The simulated data consisted of 1000 vectors with 60 independent data components: (a) homogenous, Gaussian distributed data with mean $\mathbf{0}$ and variance 1 in every data component, (b) data with one cluster added, 200 of the vectors are generated from a Gaussian distribution with cluster centre $\hat{\mathbf{z}}_1 \neq 0$, such that $\|\hat{\mathbf{z}}_1\| = \sqrt{M}$. (c) data with two clusters of size 200 each with random cluster centres $\hat{\mathbf{z}}_1$ and $\hat{\mathbf{z}}_2$, both meeting the length constraint $\|\hat{\mathbf{z}}_1\| = \|\hat{\mathbf{z}}_2\| = \sqrt{M}$. The diagrams show: (left) the principal component analysis of the total data, displaying a data scatter plot for the first two principal components; (middle) heat map showing the sample covariance matrix; (right) Eigenvalues of the sample covariance matrix, plotted in the decreasing order.

5.3 Generalised statistical theory of clusters

The key components of the theory presented in Chapter 3 were the *background model* and the *similarity measure* for vectors. We will now show how to incorporate data component dependencies into this theory.

Background model. The background model discussed in Chapter 3 assumes independent data components. Here, we extend this model to allow for an arbitrary covariance $\mathbf{G}$. Thus, the constraints on the distribution are: the mean fixed to $\mathbf{0}$ (as before), and the covariance matrix fixed to $\mathbf{G}$. Again, the maximum entropy distribution meeting these constraints is a Gaussian distribution, this time parameterised by $\mathbf{G}$,

$$P_0(\mathbf{x}|\mathbf{G}) = \frac{1}{Z_0} e^{-\frac{1}{2}\mathbf{x}\cdot\mathbf{H}\cdot\mathbf{x}} , \tag{5.6}$$

with $Z_0 = (\det(\mathbf{G})2\pi)^{M/2}$ and $\mathbf{H} = \mathbf{G}^{-1}$, the inverse of the covariance matrix.

Euclidean distance. Let us consider again the Euclidean distance-based similarity measure, $\mathrm{sim}(\mathbf{x}, \mathbf{z}) = -\frac{1}{2}\|\mathbf{x} - \mathbf{z}\|^2 = -\frac{1}{2}(\mathbf{x} - \mathbf{z}) \cdot (\mathbf{x} - \mathbf{z})$, where the dot operator is multiplication with an identity matrix, or simply

$$\mathbf{x}_1 \cdot \mathbf{x}_2 = \mathbf{x}_1^T \mathbf{x}_2 = \sum_\mu x_1^\mu x_2^\mu . \tag{5.7}$$

Here, every data component is given the same weight and all data components give equal contribution to the overall value of the similarity measure. Let $\mathbf{G}$ be the covariance matrix of data components. In the formulation in Chapter 3, we implicitly assumed that the covariance matrix $\mathbf{G}$ is an identity matrix. If data contains dependencies between components, we can appropriately down- and up-weigh them by using $\mathbf{H}$ as a metric:

$$\mathbf{x}_1 \cdot \mathbf{H} \cdot \mathbf{x}_2 = \mathbf{x}_1^T \mathbf{H} \mathbf{x}_2 = \sum_\mu \sum_\nu x_1^\mu H^{\mu\nu} x_2^\nu . \tag{5.8}$$

The distance computed by means of metric $\mathbf{H}$ is known as the *Mahalonobis distance* [72]. We redefine the similarity measure (3.7) as

$$\mathrm{sim}(\mathbf{x}, \mathbf{z}|\mathbf{G}) = -\frac{1}{2}(\mathbf{x} - \mathbf{z}) \cdot \mathbf{H} \cdot (\mathbf{x} - \mathbf{z}) . \tag{5.9}$$

Correlation. In the new metric space, we may consider the correlation-based similarity measure for data vectors,

$$\mathrm{sim}(\mathbf{x}, \mathbf{z}|\mathbf{G}) = M \frac{\mathbf{x} \cdot \mathbf{H} \cdot \mathbf{z}}{\sqrt{(\mathbf{x} \cdot \mathbf{H} \cdot \mathbf{x})}\sqrt{(\mathbf{z} \cdot \mathbf{H} \cdot \mathbf{z})}} = \hat{\mathbf{x}} \cdot \hat{\mathbf{z}} , \tag{5.10}$$

where vectors $\hat{\mathbf{x}} \equiv \mathbf{x}\sqrt{M/(\mathbf{x}\cdot\mathbf{H}\cdot\mathbf{x})}$ and $\hat{\mathbf{z}} \equiv \mathbf{z}\sqrt{M/(\mathbf{z}\cdot\mathbf{H}\cdot\mathbf{z})}$ are vectors pointing in the same direction as vectors $\mathbf{x}$ and $\mathbf{z}$, but they are length-normalised with respect to metric $\mathbf{H}$.

The null model for clusters defined by directional density, is a *uniform distribution* on a surface of a *hyper-ellipsoid* defined by metric $\mathbf{H}$,

$$P_0(\hat{\mathbf{x}}|\mathbf{G}) = \frac{1}{Z_0}\delta(\hat{\mathbf{x}}\cdot\mathbf{H}\cdot\hat{\mathbf{x}} - M) \ , \tag{5.11}$$

where $Z_0 \simeq \exp\{M/2(1+\log(2\pi))\}$ as before.

Cluster model. The general definition of the cluster model is unchanged,

$$Q(\mathbf{x}|\mathbf{z},\eta,\mathbf{G}) = \frac{1}{Z_\eta}P_0(\mathbf{x}|\mathbf{G})e^{\eta\,\mathrm{sim}(\mathbf{x},\mathbf{z}|\mathbf{G})} \ . \tag{5.12}$$

The difference is that in the new definitions, the background model and the similarity measure depend on $\mathbf{G}$.

In cases where there are no doubts about the assumed metric $\mathbf{H}$, we will use notation "$\cdot$" instead of "$\cdot\mathbf{H}\cdot$".

5.4 Mixture-model for estimation of component dependencies

We propose a maximum-likelihood based estimation of data dependencies, which uses a *mixture-model* [75]. The aim of this approach is to explicitly disentangle the spurious effect of clusters from the "true" data component dependencies.

As discussed in Chapter 3 and in the previous section, we model unclustered vectors with a so-called background model, $P_0(\mathbf{x}|\mathbf{G})$. Clustered vectors are generated from a cluster model $Q(\mathbf{x}|\mathbf{z},\eta,\mathbf{G})$. Assuming there are K different clusters in data, the probability density function of the mixture-model is

$$f(\mathbf{x}|\mathbf{G},\mathbf{z}_k,\eta_k : k=1,\ldots,K) = \tau_0 P_0(\mathbf{x}|\mathbf{G}) + \sum_{k=1}^{K}\tau_k Q(\mathbf{x}|\mathbf{z}_k,\eta_k,\mathbf{G}) \ , \tag{5.13}$$

where parameters τ_k are the mixture proportions, satisfying constraints

$$0 \le \tau_k \le 1, \quad \sum_{k=0}^{K}\tau_k = 1 \ . \tag{5.14}$$

It is important to note that in this construction we assume that *the same data component dependencies* $\mathbf{G}$ hold in the background and in all cluster components. In

such a setting, clustering of vectors is independent of component dependencies. Of course, a model introducing different dependencies in the background and in the cluster components is also plausible. The difference would be to introduce independently estimated covariances matrices, e.g. $\mathbf{G}_0$ for the background, and $\mathbf{G}_k$ for each cluster k. Here, we discuss a mixture-model framework for estimation of component dependencies in the minimal model with a single covariance matrix $\mathbf{G}$. An extension to the more general model sketched above is straightforward but it is not included in this thesis.

The data log-likelihood is

$$\mathcal{L}_{\mathrm{MIX}}(\mathbf{G}, \mathbf{z}_k, \eta_k : k = 1, \ldots, K) = \sum_{i=1}^{N} \log \left[\tau_0 P_0(\mathbf{x}_i|\mathbf{G}) + \sum_{k=1}^{K} \tau_k Q(\mathbf{x}_i|\mathbf{z}_k, \eta_k, \mathbf{G}) \right] . \tag{5.15}$$

We want to find the covariance matrix $\mathbf{G}$ which maximises the log-likelihood of the data under the mixture-model (5.15), i.e. we are looking for the solution of

$$\frac{\partial}{\partial G^{\mu\nu}} \mathcal{L}_{\mathrm{MIX}}(\mathbf{G}, \mathbf{z}_k, \eta_k : k = 1, \ldots, K) = 0 , \tag{5.16}$$

for $\mu, \nu = 1, \ldots, M$. In the general case, other clustering parameters, $\mathbf{z}_k$, η_k and τ_k, are also unknown, so we need to solve the full clustering problem, with additional conditions,

$$\frac{\partial}{\partial z_k^{\mu}} \mathcal{L}_{\mathrm{MIX}}(\mathbf{G}, \mathbf{z}_k, \eta_k : k = 1, \ldots, K) = 0 \tag{5.17}$$

$$\frac{\partial}{\partial \eta_k} \mathcal{L}_{\mathrm{MIX}}(\mathbf{G}, \mathbf{z}_k, \eta_k : k = 1, \ldots, K) = 0 . \tag{5.18}$$

and the constraints on the mixing proportions τ_k from (5.14). The above maximum-likelihood equations are difficult to solve analytically because of the logarithm of the sum in Eq. (5.15). An analytical trick of introducing the so-called *hidden data*, in this case assignments of data elements to mixture components, makes the problem analytically tractable. The solution uses the standard expectation-maximisation method [26, 76], which we briefly describe in the following section.

5.4.1 The EM algorithm

The expectation-maximisation algorithm is a general approach to solving maximum-likelihood problems in case of incomplete data. The algorithm is an iterative method which generates a sequence of improving parameter approximations. The algorithm alternates two types of steps: (i) the expectation-step in which the expectation of the log-likelihood is computed, given a current estimate of the hidden variables; (ii) and the maximisation-step, in which parameters of the model are estimated to maximise the expected log-likelihood. It has been rigorously proven that the algorithm converges to a local maximum [107].

We assume that we have the observed data $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ and the hidden data $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$ from some space $\mathcal{Y}$. In the clustering problem, the hidden data are the cluster-assignment vectors, $\mathbf{y}_i = [y_{i0}, \ldots, y_{iK}]$, defined for $k = 0, 1, \ldots, K$ as

$$y_{ik} = \begin{cases} 1 & \text{iff } \mathbf{x}_i \text{ was generated from component } k \text{ of the mixture,} \\ 0 & \text{otherwise.} \end{cases} \tag{5.19}$$

Index $k = 0$ corresponds to the background component $P_0(\mathbf{x}|\mathbf{G})$.

We denote the set of model parameters by Θ. For the clustering problem, Θ is

$$\Theta = \{\mathbf{G}, \mathbf{z}_k, \eta_k : k = 1, \ldots, K\} \ , \tag{5.20}$$

the covariance matrix $\mathbf{G}$ and the parameters of K clusters: widths η_k, and cluster centres $\mathbf{z}_k$.

Given the hidden data $\mathbf{Y}$, we can write the joint probability of the *complete data* as

$$P(\mathbf{X}, \mathbf{Y}|\Theta) = P(\mathbf{Y}|\mathbf{X}, \Theta)P(\mathbf{X}|\Theta) \ . \tag{5.21}$$

The aim is to optimise the standard *observed-data log-likelihood*, $\mathcal{L}(\Theta) \equiv \log P(\mathbf{X}|\Theta)$ (which in our problem is the mixture log-likelihood (5.15)). The log-likelihood $\mathcal{L}(\Theta)$ is computed by integrating out the unknown data $\mathbf{Y}$,

$$\mathcal{L}(\Theta) = \log P(\mathbf{X}|\Theta) = \log \left(\int_{\mathcal{Y}} P(\mathbf{X}, \mathbf{Y}|\Theta) d\mathbf{Y} \right) \ . \tag{5.22}$$

The EM algorithm is to maximise the likelihood $\mathcal{L}(\Theta)$ (5.22) over parameters Θ. It proceeds with iterations, where the unknown values of variables $\mathbf{Y}$ are replaced with conditional expectations $\hat{\mathbf{Y}}$; the conditional expectations are computed using the current estimation of the parameters, in the nth step denoted by Θ^n. The estimated values $\hat{\mathbf{Y}}$ are then used to replace the hidden values in the complete-data likelihood function (5.21). An update of parameters, Θ^{n+1}, is computed to maximise the log-likelihood (5.15).

Let us denote the optimal set of parameter by Θ^*. If $\Theta^n \neq \Theta^*$, then we have $\mathcal{L}(\Theta^*) > \mathcal{L}(\Theta^n)$. By applying the Jensen's inequality to the difference $\mathcal{L}(\Theta^*) - \mathcal{L}(\Theta^n)$, we obtain

$$\mathcal{L}(\Theta^*) \geq \mathcal{L}(\Theta^n) + \int_{\mathcal{Y}} P(\mathbf{Y}|\mathbf{X}, \Theta^n) \log \left(\frac{P(\mathbf{X}, \mathbf{Y}|\Theta^*)}{P(\mathbf{X}, \mathbf{Y}|\Theta^n)} \right) d\mathbf{Y} \ . \tag{5.23}$$

Instead of performing a direct maximisation over Θ of the log-likelihood $\mathcal{L}(\Theta)$ (5.22), one can iteratively optimise the right hand side of the resulting inequality (5.23). The latter, in many applications, is easier to compute.

Further, we define an auxiliary function $R(\Theta, \Theta^n)$, which gives the same maximum-likelihood argument as the right-hand side of (5.23), but is deprived of the terms independent of Θ,

$$R(\Theta, \Theta^n) = \int_{\mathcal{Y}} P(\mathbf{Y}|\mathbf{X}, \Theta^n) \log P(\mathbf{X}, \mathbf{Y}|\Theta) d\mathbf{Y} = \mathbb{E}_{\mathbf{Y}|\mathbf{X}, \Theta^n} \left[\log P(\mathbf{X}, \mathbf{Y}|\Theta) \right] \ . \tag{5.24}$$

It can be shown (see Appendix B), that maximisation of $R(\Theta, \Theta^n)$ over Θ, indeed increases the log-likelihood, i.e. by setting

$$\Theta^{n+1} = \arg\max_{\Theta} R(\Theta, \Theta^n) \tag{5.25}$$

we obtain improved log-likelihood,

$$\mathcal{L}(\Theta^{n+1}) \geq \mathcal{L}(\Theta^n) \ .$$

Each step of the iteration increases the log-likelihood and the whole iterative procedure eventually reaches a (local) maximum. To summarise, this result defines two steps of the EM iteration:

1. *E-step.* For a given value of parameters Θ^n, compute the expected value $R(\Theta, \Theta^n)$ (5.24), as a function of Θ. This computation involves computation of the conditional expectations of the hidden data, $\mathbf{Y}$.
2. *M-step.* Find the maximum-likelihood estimate Θ^{n+1}, following Eq. (5.25).

These steps are repeated alternately, until convergence. The expectation-maximisation algorithm always converges to a *local* maximum [107] but its is not guarantied to reach the *global* maximum. In this respect, its performance strongly depends on the starting conditions Θ^0.

5.4.2 EM for the mixture-model

Using the EM algorithm framework, we aim to locate the maximum-likelihood parameters of the log-likelihood function for a mixture-model defined in Eq. (5.15). The hidden data $\mathbf{Y}$ and the parameters Θ are defined in Eq. (5.19) and Eq. (5.20).

The joint probability of the complete data (5.21) is for our problem given by

$$P(\mathbf{X}, \mathbf{Y}|\mathbf{G}, \mathbf{z}_k, \eta_k : k = 1, \ldots, K) = \prod_{i=1}^{N} \left[(\tau_0 P_0(\mathbf{x}_i|\mathbf{G}))^{y_{i0}} \prod_{k=1}^{K} (\tau_k Q(\mathbf{x}_i|\mathbf{z}_k, \eta_k, \mathbf{G}))^{y_{ik}} \right] . \tag{5.26}$$

Cluster labels y_{ik} simply "pick" appropriate mixture components for data elements.

E-step. The current estimate of parameters, after n steps of the EM algorithm, is denoted by $\Theta^n = \{\mathbf{G}^n, \mathbf{z}_k^n, \eta_k^n : k = 1, \ldots, K\}$. In the *E-step* of the algorithm, we compute

$$R(\Theta, \Theta^n) = \int_{\mathcal{Y}} P(\mathbf{Y}|\mathbf{X}, \Theta^n) \log P(\mathbf{X}, \mathbf{Y}|\Theta) d\mathbf{Y} \tag{5.27}$$

$$= \int_{\mathcal{Y}} \left[\prod_{i=1}^{N} \prod_{k=0}^{K} (\rho_{ik})^{y_{ik}} \right] \log P(\mathbf{X}, \mathbf{Y}|\Theta) d\mathbf{Y} \ , \tag{5.28}$$

where $\rho_{ik} = P(y_{ik} = 1|\Theta^n)$ is the probability that $y_{ik} = 1$,

$$\rho_{ik} = \frac{P(y_{ik} = 1)P(\mathbf{x}_i|y_{ik} = 1|\Theta^n)}{P(\mathbf{x}_i|\Theta^n)} , \tag{5.29}$$

as derived from the Bayes' rule. Expanding (5.29) for components of the mixture-model from Eq. (5.13), we get

$$\rho_{i0} = \frac{\tau_0 P_0(\mathbf{x}_i|\mathbf{G})}{\tau_0 P_0(\mathbf{x}_i|\mathbf{G}) + \sum_{k=1}^{K} \tau_k Q(\mathbf{x}_i|\mathbf{z}_k, \eta_k, \mathbf{G})} , \tag{5.30}$$

$$\rho_{ik} = \frac{\tau_k Q(\mathbf{x}_i|\mathbf{z}_k, \eta_k, \mathbf{G})}{\tau_0 P_0(\mathbf{x}_i|\mathbf{G}) + \sum_{k=1}^{K} \tau_k Q(\mathbf{x}_i|\mathbf{z}_k, \eta_k, \mathbf{G})} . \tag{5.31}$$

By performing integration over $\mathbf{Y}$ in the objective function $R(\Theta, \Theta^n)$ (5.28), we obtain

$$R(\Theta, \Theta^n) = \sum_{i=1}^{N} \left[\rho_{i0} \log \tau_0 P_0(\mathbf{x}_i|\mathbf{G}) + \sum_{k=1}^{K} \rho_{ik} \log \tau_k Q(\mathbf{x}_i|\mathbf{z}_k, \eta_k, \mathbf{G}) \right] . \tag{5.32}$$

M-step. Equation (5.32) is a general form of the objective function for the M-step of the EM algorithm in a mixture-model. The equations solved in the M-step are of the form

$$\frac{\partial}{\partial \Theta} R(\Theta, \Theta^n) = 0. \tag{5.33}$$

The exact form depends on the specific choice of the background model $P_0(\mathbf{x}|\mathbf{G})$ and the cluster model $Q(\mathbf{x}|\mathbf{z}, \eta, \mathbf{G})$.

M-step: mixture-model with Gaussian background. For the sake of estimating data component dependencies, we will focus on the model with a Gaussian background and clusters defined by point density and positional bias, see Chapter 3.

The application of the mixture-model to length-constrained data, for clusters defined by directional density, is more problematic for an unknown covariance matrix $\mathbf{G}$: the data is to be normalised with respect to a yet unknown metric, which in turn is to be estimated from the data. Here, we will not show the solution to this problem. We will still use the directional-density-based model for clustering gene expression data, but only in case of an *a priori* determined covariance matrix, see Chapter 6.

We present the maximum-likelihood equations for a mixture of the Gaussian background and clusters defined by point density and positional bias. In the M-step of the algorithm, we maximise the objective function $R(\Theta, \Theta^n)$ (5.32) with respect to parameters $\Theta = \{\mathbf{G}, \mathbf{z}_k, \eta_k : k = 1, \ldots, K\}$ for some current estimate $\Theta^n = \{\mathbf{G}^n, \mathbf{z}_k^n, \eta_k^n :$

$k = 1, \ldots, K\}$. By solving the maximum-likelihood equations, we obtain the following update rules for the parameters:

$$\mathbf{z}_k = \frac{\sum_{i=1}^{N} \rho_{ik} \mathbf{x}_i}{\sum_{i=1}^{N} y_i^k} , \tag{5.34}$$

$$\eta_k = \frac{M \sum_{i=1}^{N} \rho_{ik}}{\sum_{i=1}^{N} \rho_{ik} \mathbf{x}_i \cdot \mathbf{x}_i - N \tau_k \mathbf{z}_k \cdot \mathbf{z}_k} , \tag{5.35}$$

$$G^{\mu\nu} = \frac{1}{N} \left[\sum_{i=1}^{N} \rho_{i0} x_i^\mu x_i^\nu + \sum_{k=1}^{K} \eta_k \left(\sum_{i=1}^{N} \rho_{ik} x_i^\mu x_i^\nu - \tau_k z_k^\mu z_k^\nu \right) \right] , \tag{5.36}$$

$$\tau_k = \sum_{i=1}^{N} \rho_{ik} / N . \tag{5.37}$$

Initialisation of parameters. As discussed earlier, the EM algorithm may converge to a local maximum and the final solution depends strongly on the initialising set of parameters Θ^0. Following Fraley and Raftery [35, 34], we use the model-based hierarchical clustering algorithm to find the initiating cluster assignments $\mathbf{Y}$ and the set of parameters $\Theta^0 = \{\mathbf{G}^0, \mathbf{z}_k^0, \eta_k^0 : k = 1, \ldots, K\}$. The implementation of the algorithm [34] does not include the background component. The algorithm optimises the *classification log-likelihood*,

$$\mathcal{L}_{\mathrm{C}}(\mathbf{G}, \mathbf{z}_k, \eta_k : k = 1, \ldots, K) = \sum_{i=1}^{N} Q(\mathbf{x}_i | \mathbf{z}_{\gamma_i}, \eta_{\gamma_i}, \mathbf{G}) . \tag{5.38}$$

where γ_i is a label pointing at the cluster component Q_k to which element $\mathbf{x}_i$ is classified. This criterion differs from the maximum log-likelihood (5.15): here every element is strictly assigned to exactly one component. The algorithm starts with N Q-components, each describing a singleton cluster, with cluster centres $\mathbf{z}_i \equiv \mathbf{x}_i$ and some arbitrary guess for parameters η_i and matrix $\mathbf{G}$. The algorithm proceeds by merging pairs of clusters; in each step a pair which leads to the greatest increase of the classification log-likelihood (5.38) is chosen. After a merging of two clusters, the parameters of the resulting cluster are computed using rules (5.34) - (5.37), and setting $\rho_{ik} \equiv \delta_{\gamma_i = k}$. The algorithm proceeds until there are K components left.

Choosing the number of clusters. The number of clusters in data is usually unknown *a priori*. In the probabilistic setting of the mixture-model, the problem of choosing the number of clusters is a well known problem of *model selection*: selecting the model which "best" describes the data.

Comparison of models with different number of components cannot be done by comparing the data log-likelihood (5.15): more complex models with a larger number of components have an advantage in fitting to the data and in general yield higher log-likelihood values. Here, we use an approach based on the Bayesian model selection.

Let us denote a class of mixture-models with K components by M_K and its number of parameters by L. The idea is to find the class of models with the highest posterior probability $p(M_K|\mathbf{X})$. By Bayes' rule, the posterior probability is

$$P(M_K|\mathbf{X}) = P(\mathbf{X}|M_K)P(M_K)/P(\mathbf{X}) \ . \tag{5.39}$$

In the absence of any prior knowledge, the usual choice for the prior distribution $P(M_K)$ is a uniform one, which gives the same probability to all models. To maximise (5.39), we simply maximise the probability of the data given a class of models M_K, $P(\mathbf{X}|M_K)$. To compute $p(\mathbf{X}|M_K)$, we integrate over unknown parameters of model M_K,

$$P(\mathbf{X}|M_K) = \int P(\mathbf{X}|\Theta, M_K)P(\Theta|M_K)d\Theta_k \ . \tag{5.40}$$

This integral is in general difficult to compute. However, if we assume that there exists a dominating maximum Θ^*, we can expand the integrand around the maximum point and approximate it with a Gaussian integral (see section 2.2.2 on the saddle-point integration):

$$\begin{aligned} P(\mathbf{X}|M_K) &= \int P(\mathbf{X}|\Theta, M_K)P(\Theta|M_K)d\Theta \\ &= \int e^{\log P(\mathbf{X}|\Theta, M_K)} P(\Theta|M_K)d\Theta \\ &= P(\Theta^*|M_K)e^{\log P(\mathbf{X}|\Theta^*, M_K)} \sqrt{\left|\frac{(2\pi)^L}{\det D^2(\Theta^*))}\right|} \ , \end{aligned} \tag{5.41}$$

where $D^2(\Theta^*) = \frac{d^2}{d\Theta_K^2} \log[P(\mathbf{X}|\Theta^*, M_K)]$ is a Hessian matrix of second derivatives of the log-posterior. The square-root term, coming from the Gaussian integration, is the so called *volume factor*, which scales exponentially with L, the size of the set of parameters Θ. As such, it penalises complex models with many parameters. The data log-likelihood $\log[P(\mathbf{X}|\Theta^*, M_K)]$ scales with N, the size of data $\mathbf{X}$. Given that, the very rough approximation tells that the square-root term scales like $N^{-L/2}$. Dropping the constant terms, we obtain the *Bayesian information criterion* proposed by Schwarz [45],

$$\mathrm{BIC}(M_K) \equiv \log P(\mathbf{X}|M_K) \simeq \log P(\mathbf{X}|\Theta^*, M_K) - \frac{L}{2} \log N \ . \tag{5.42}$$

The log-likelihood of a model can be approximated by the maximum-likelihood fit minus a term penalising model complexity (a function of the number of parameters L).

The calculation above is an approximation, alternative information criterions have been proposed: a well known is the *Aikake information criterion* [1], which has a very similar form, $\mathrm{AIC}(M_K) = \log P(\mathbf{X}|\Theta^*, M_K) - \log L$. The penalty of $\log L$ is less constraining than the penalty of the BIC.

5.4.3 Application

Simulated data example. We repeated the estimation of the covariance matrix on illustrative examples discussed in section 5.2.2 and in Fig. 5.4: (a) simulated data with a background component only, (b) with one cluster component and (c) with two cluster components. In each case we ran the full clustering procedure as described in Section 5.4.2. The number of cluster components was in each case estimated correctly by the BIC (i.e. it was zero, one, and two for (a), (b), and (c) respectively). The covariance matrices were diagonal, with only insignificant off-diagonal entries. The eigenvalues of the estimated covariance matrices were close to 1 (deviations are due to insignificant fluctuations). This result is shown in Fig. 5.5.

Human variation data. We performed another analysis on the human variation gene expression data [96, 10]. Gene expression was measured in the lymphoblastoid cell lines in 210 unrelated individuals from the Hap Map populations: 60 Utah residents with ancestry from northern and western Europe (CEU), 45 Han Chinese in Beijing (CHB), 45 Japanese in Tokyo (JPT) and 60 Yoruba in Ibadan, Nigeria (YRI). We show dependencies in expression between individuals from the European population (CEU) only. Since the individuals are not related, we expect them to have independent gene expression profiles. Therefore, there should be no off-diagonal structure in the covariance matrix: the eigenvectors should roughly agree with the directions specified by the original coordinates. Moreover, we do not expect large differences in the genome-wide variance of expression between the individuals, i.e. the distribution of the eigenvalues should be peaked around the mean value.

The sample covariance matrix, due to the presence of clusters, is not diagonal. Moreover, the difference between the first two leading eigenvalues is almost 10-fold.

On the other hand, the result of our approach is in agreement with our expectations. The mixture-model was fit to the data with $K = 122$ cluster components (yielding the optimal BIC). The estimated covariance matrix is diagonal with all eigenvalues roughly equal. The comparison is shown in Fig. 5.6.

5.5 Application in tumour classification

A straightforward application of the covariance and the correlation matrix estimation is in the problem of tumour sample clustering and classification. The goal is to group tumour samples, on the basis of their expression profiles' similarities, into classes that correspond to tumour types. This clustering problem is different in nature to clustering of genes based on their expression across samples: here the number of dimensions is much larger then the number of elements to be clustered. The high-dimensionality is a challenge for the probabilistic mixture-model-based approaches. Very standard

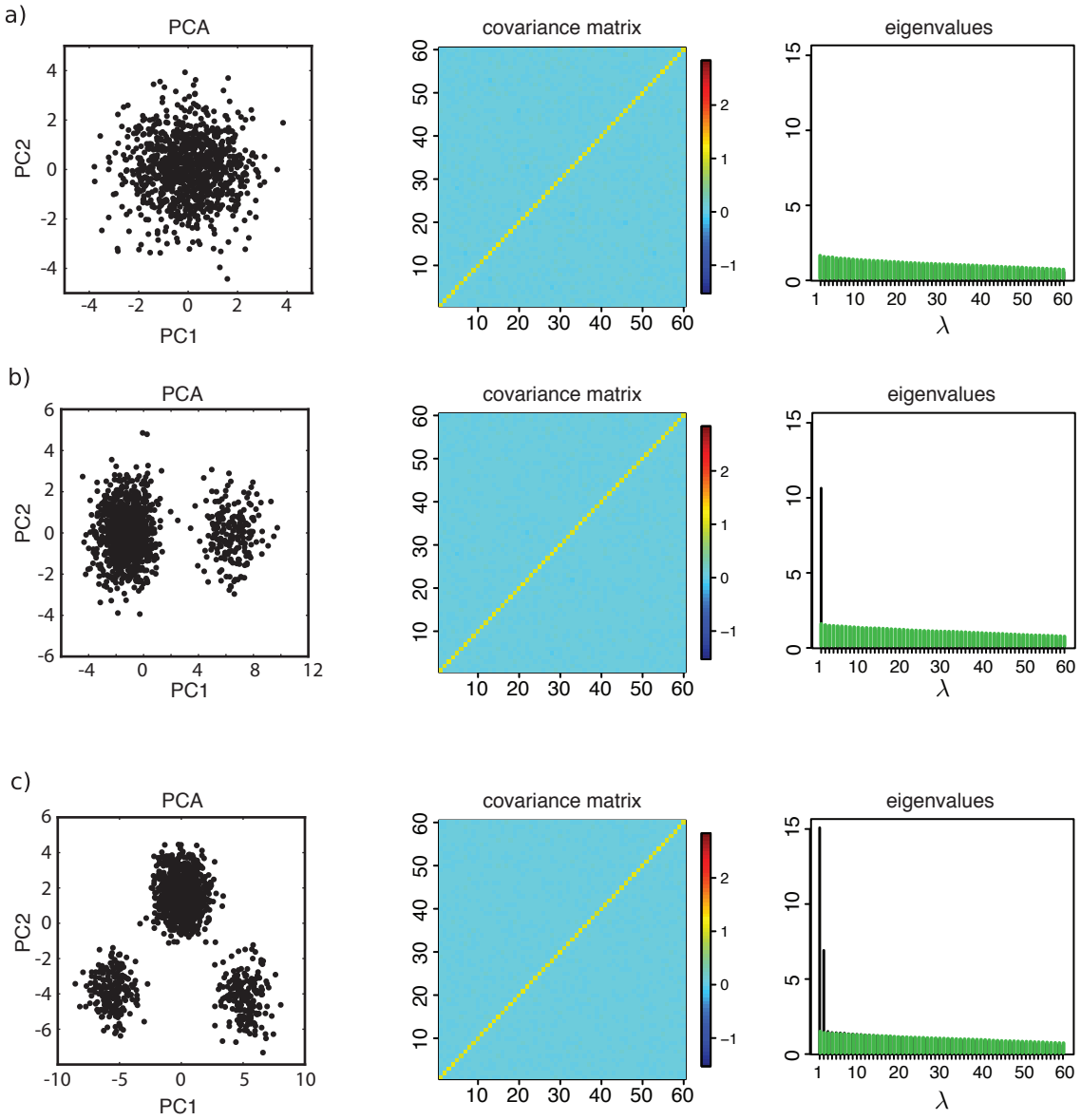

Figure 5.5: Corrected covariance estimation in simulated data. Description of the data in the caption of Fig. 5.4. The diagrams show: (left) the principal component analysis of the total data, displaying a data scatter plot for the first two principal components; (middle) heat map showing the estimated covariance matrix; (right) distribution of the eigenvalues, in the decreasing order. The new estimation (green) is superimposed on the old result, of the sample covariance matrix (black). Both the eigenvectors and the eigenvalues are now correctly estimated.

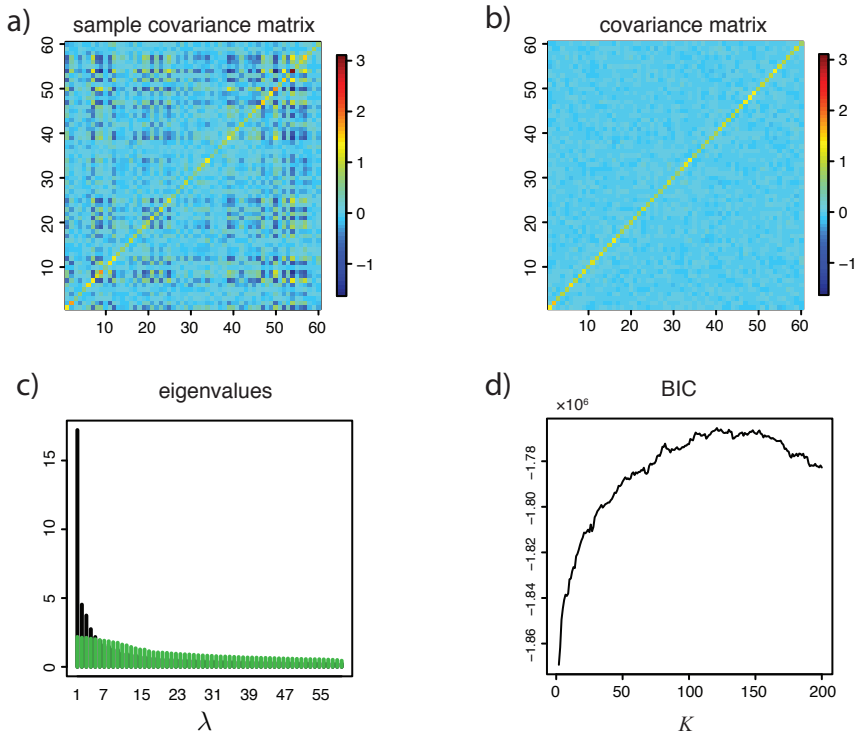

Figure 5.6: Covariance estimation in human variation data. (a) The sample covariance matrix shows an off-diagonal structure emerging due to gene clusters in data. (b) This structure is no longer seen in the corrected covariance matrix. (c) The sorted eigenvalues of the sample covariance matrix (black) and the corrected covariance matrix (green). This result suggests statistical independence of gene expression profiles between individuals in the analysed population. (d) The corrected covariance matrix was estimated with the maximum BIC mixture-model with K=122 clusters.

clustering algorithms are usually employed, most commonly the hierarchical agglomerative methods. Such methods operate on a pairwise similarity measure for data elements, such as the sample-correlation coefficient or Euclidean distance. In here, we investigate the effect of the improved covariance/correlation estimation on the tumour classification performance. As a similarity measure we use the "corrected" correlation coefficient (5.2), i.e.

$$\mathrm{sim}(\mathbf{x}^\mu, \mathbf{x}^\nu) \equiv \hat{G}^{\mu\nu} = G^{\mu\nu}/(G^{\mu\mu}G^{\nu\nu}). \tag{5.43}$$

5.5.1 Estimation of tumour correlations

We consider three genome-wide microarray datasets measuring expression in tumour samples: leukaemia dataset by Golub et al. [46], and two datasets collecting a wide range of tumour types, by Su et al. [97] and Ramaswamy et al. [85].

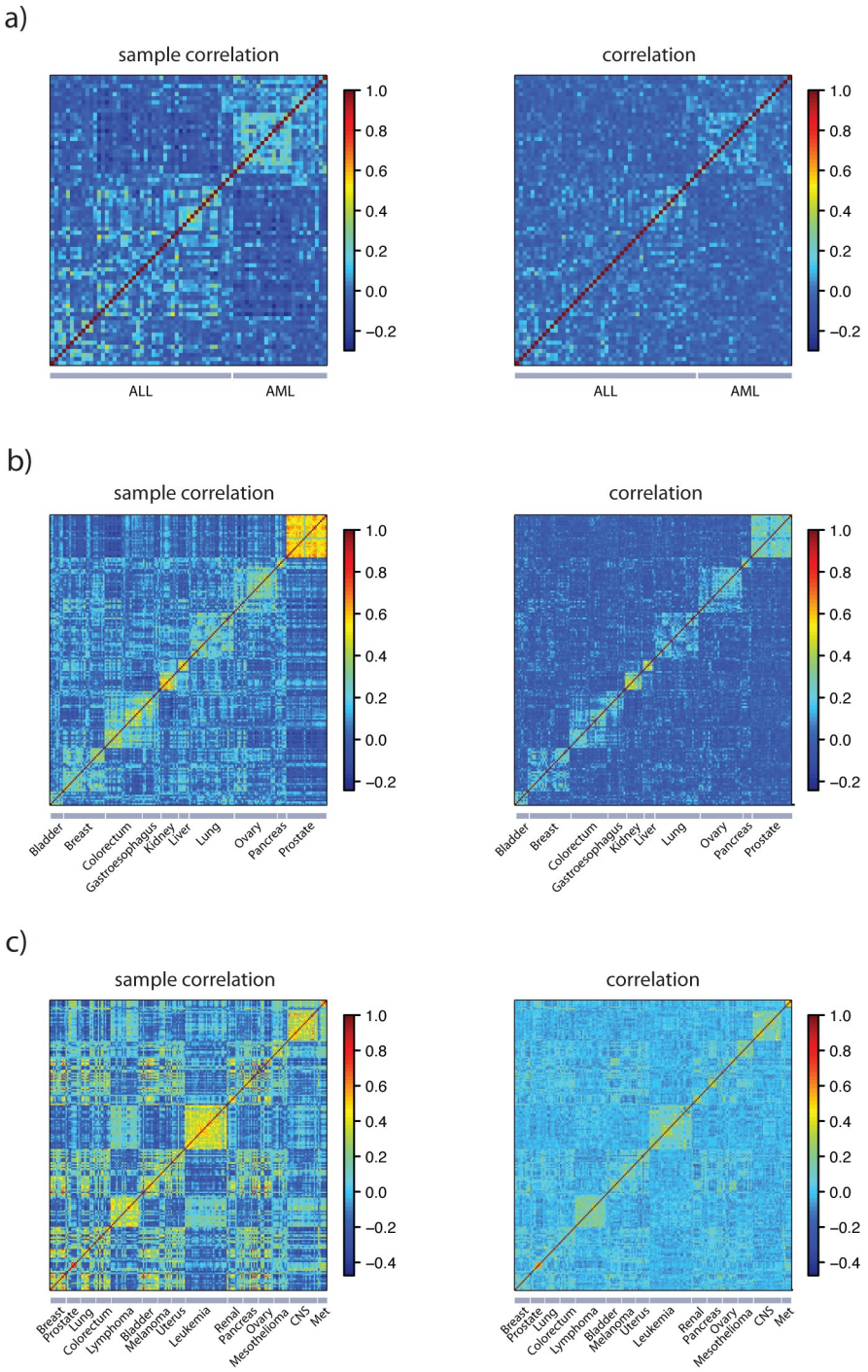

Figure 5.7: Similarities between tumour samples in three datasets. The pictures compare the sample correlation matrices (left) and the correlation matrices computed using the mixture-model based approach in three tumour datasets (a) Golub data, (b) Su data and (c) Ramaswamy data. In all cases, the off-diagonal correlations are much stronger in the sample correlation matrix.

Leukemia dataset [46]. The dataset by Golub et al. [46], is one of the microarray cancer datasets which are widely used as a benchmark to test performance of new methods. The gene expression was measured in 72 bone marrow samples collected from acute leukaemia patients. Two types of leukaemia were diagnosed: 47 of the samples as acute lymphoblastic leukaemia (ALL) and 25 as acute myeloid leukaemia (AML). The experiment was performed with Affymetrix high-density oligonucleotide microarrays which contained 7129 human genes. We fitted the mixture-model with $K = 1, \ldots, 100$. The maximum Bayesian information criterion was obtained at $K = 20$ gene clusters. The background component $P_0(\mathbf{x}|\mathbf{G})$ had the largest weight, $\tau_0 = 0.349$, the rest of the data was assigned to clusters. In Fig. 5.7 (a) we compare the sample correlation matrix to the corrected correlation matrix obtained with this mixture-model. The former shows much stronger off-diagonal (cross-experiment) correlations than the latter. The presence of clusters has a strong effect on the estimation of the sample correlation matrix.

Su dataset [97]. The second dataset consists of 174 samples of carcinomas of: prostate, breast, lung, ovary, colorectum, kidney, liver, pancreas, bladder/ureter, and gastroesophagus (10 types). Expression of 12533 genes was measured. We ran the EM algorithm to fit the mixture-model with $K = 1, \ldots, 70$ components. The maximum BIC value was obtained by a mixture-model with 32 cluster components. The background component $P_0(\mathbf{x}|\mathbf{G})$ had the largest weight, $\tau_0 = 0.68$, suggesting that large part of the data is not in clusters. However, the effect of the rest of the data on the covariance estimation is strong: the sample correlation matrix shows much stronger cross-dependencies between samples. The resulting correlation matrices are shown in Fig 5.7 (b).

Ramaswamy dataset [85]. This dataset is a subset of samples from the Global Cancer Map [85], constructed to find differential gene expression between a wide variety of tumour types. We used 190 primary and 8 metastatic tumour samples. There were 14 different primary types of tumours: breast, prostate, lung, colorectum, lymphoma, bladder, melanoma, uterus, leukaemia, renal, pancreas, ovary, and central nervous system. The optimal BIC was obtained for a mixture-model with 33 cluster components. The background component $P_0(\mathbf{x}|\mathbf{G})$ again had the largest weight among all mixture components, $\tau_0 = 0.193$. The resulting correlation matrices are shown in Fig. 5.7 (c).

In the following section, we will use the estimated correlation matrices in clustering of tumour samples.

5.5.2 Tumour clustering

Validating tumour classification. For each of the analysed datasets, the annotation of tumour samples was provided. We expect a clustering to reproduce these data

partitions. We used the adjusted Rand Index [50] to measure quality of such tumour classifications. The basic Rand Index reports the number of pairs of data elements that are in the same relationship in both partitions : they are either in the same cluster, s, or are in different clusters, d, in both partitions,

$$R = (s + d)/\binom{N}{2} , \tag{5.44}$$

where N is the number of data elements. The adjusted Rand Index corrects for chance,

$$ARI = \frac{R - \mathbb{E}[R]}{\max[R] - \mathbb{E}[R]} . \tag{5.45}$$

The index can be intuitively understood as a "correlation" between two partitions of elements: it takes values from the interval $[-1, 1]$, with 1 standing for a perfect agreement (the same partition) and 0 or negative values standing for a random-like relation.

Classification based on differentially expressed genes. In many approaches, only a subset of genes identified as tumour-specific is used for the task of tumour classification. The idea is that only a this subset is informative about tumour identity, while the rest is neutral and brings no information that can help in tumour recognition. In some cases, these genes can also introduce noise in the classification. As candidates for the tumour-specific genes, the so-called *differentially expressed* genes are used.

Detection of differentially expressed genes is a known problem in the analysis of gene expression. In [94], 24 classical cancer gene expression datasets were analysed, the three datasets used here are included. For each of the datasets, the authors prepared a subset of differentially expressed genes. The selection procedure was as follows: first, the 10% of genes with the largest and smallest values in any of the tumour samples were removed. In the subsequent step, about 10% of the original genes with the highest degree of variation among samples were selected.

Results. We used the hierarchical clustering algorithm with the average-linkage cluster merging criterion [105]. For each of the three datasets, we compared the classifications obtained with our approach (i.e. the corrected correlation measure), to the ones obtained with standard similarity measures, which do not disentangle the true data dependence and the spurious effect of clusters: the sample correlation and the Euclidean distance. The classification was done both on entire datasets and on the subsets of differentially expressed genes.

From a given hierarchical clustering, by cutting the dendrogram at the appropriate level, we can obtain a data partition into any number of clusters $K = 1, \ldots, N$, where N is the number of elements. In Fig 5.8, we plot the number of clusters versus the adjusted Rand Index, for each of the datasets and the compared similarity measure.

For all three datasets, we mark the true number of tumour classes with a grey stripe, set at 2, 10 and 14 for the Golub, Su and Ramaswamy datasets respectively. We first consider the clustering results obtained with the similarity measures estimated on a basis of the full sets of genes. Here, the corrected correlation similarity measure clearly outperforms other methods: the peak of the adjusted Rand index measure is in all three cases reached at the true number of clusters. No other method has this ability.

Basing the classification on the set of differentially expressed genes only, the performance of the sample correlation-based classification improves on the Su and the Ramaswamy datasets. The Euclidean distance-based classification deteriorates on the Ramaswamy data and improves on the Su data, but it does not find the true number of classes. Interestingly, on these two datasets, restricting the set of genes does not have any significant effect on the performance of our method.

The results obtained on the subset of the differentially expressed genes from the Golub dataset, are for all methods significantly worse to the ones obtained the full dataset. A possible reason is a false identification of the differentially expressed genes, which is a challenging step itself.

In summary, our method outperformed other approaches and best classified the tumour samples. The performance of our method was best when applied on the full dataset of genes. Hence, our method does not rely on the problematic step of identification of differentially-expressed genes. Moreover, this property may suggest that the important signal about sample similarities is conveyed also in lowly expressed genes from the background.

5.6 Summary and discussion

Clusters of data elements have a strong effect on estimating dependencies between data components. This fact has to be taken into account in the analysis of expression data, where quite often experiments show correlated expression profiles.

Our method is a probabilistic inference, based on a mixture-model which explicitly models clusters of data elements and the dependencies between data components. The parameters are estimated with a maximum-likelihood approach, using the EM algorithm. In an application to the tumour classification problem, we showed that the dependencies computed with our method give more accurate predictions than the standard methods.

Dimensionality reduction problem. The observation that clusters have a strong effect on observed data component dependencies has a straightforward consequence in the problem of *dimensionality reduction* in statistical pattern recognition. Dimensionality reduction is used in cases of high dimensional problems, where data

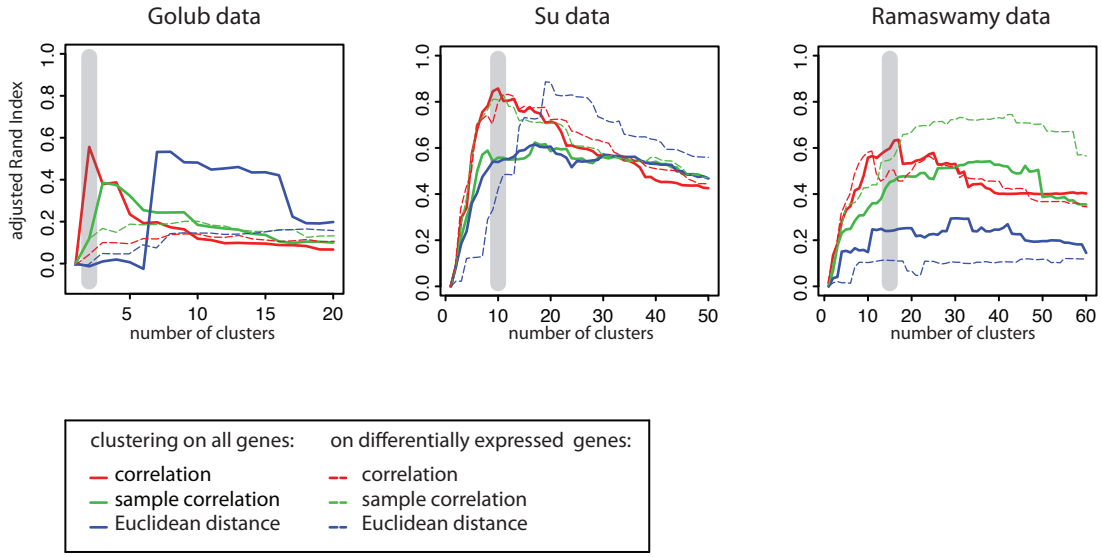

Figure 5.8: Tumour classification: agreement between sample clusterings and their annotations. The hierarchical clustering was run with three different similarity measures: the correlation estimated with our approach (red), the sample correlation (green) and the Euclidean distance (blue) on the full set of genes (solid lines) and on a subset of differentially expressed genes (dashed lines). The plots show the relation between the number of clusters and the adjusted Rand Index. The grey stripe marks the true number of tumour classes in a given dataset. Notably, for all three datasets, it coincides with the optimal clustering obtained with our corrected correlation, but it does not for the sample correlation and the Euclidean distance.

components show many dependencies. Performing similarity searches is costly in large, high-dimensional datasets and can be significantly fastened, by projecting the data onto a space of a lower number of meaningful features. Another reason is the presence of noise; the noise can be reduced, by leaving only the relevant data directions. In both cases, the task is to estimate the number of dimensions which preserve the information contained in the original data.

The commonly used Principal Component Analysis [56] performs a spectral decomposition of the sample data covariance matrix and reduces the original space to a lower dimensional space spanned by a subset of K leading eigenvectors. The respective eigenvalues tell how much variance in the data is covered by the chosen subset of eigenvectors, i.e. what is the cumulative fraction of K leading eigenvalues,

$$\left(\sum_{\mu=1}^{K}\lambda_\mu\right) \Big/ \left(\sum_{\mu=1}^{M}\lambda_\mu\right) \quad . \tag{5.46}$$

As we have seen, the spurious, cluster-related data dependencies dominate the eigenvalue distribution. In Fig. 5.9 we show an example of dimensionality reduction in yeast expression data [42], applying the PCA on the sample covariance matrix and

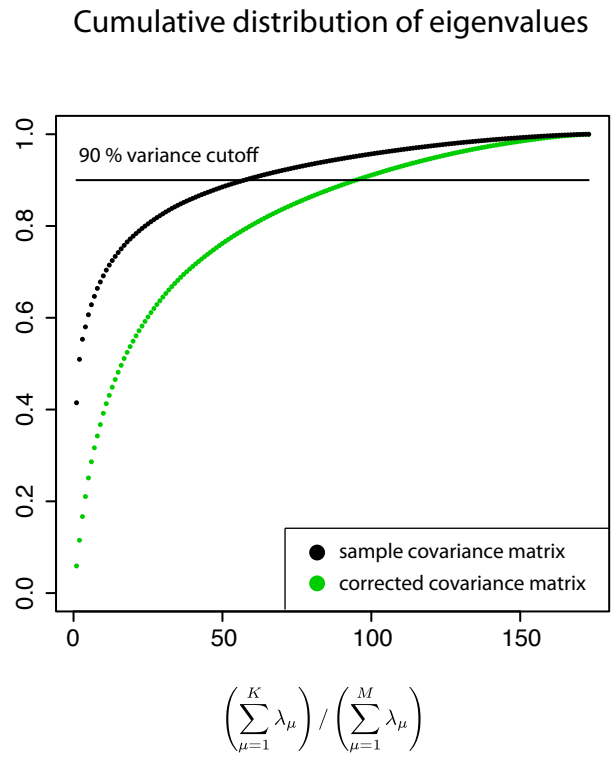

Figure 5.9: Dimensionality reduction in yeast expression data. The yeast expression data is a series of time course experiments under different environmental shock conditions, such as the heat shock, starvation, nitrogen depletion. Each such condition includes several time points; there are 173 experiments in total. The data components show many dependencies; we perform a dimensionality reduction to explain 90% of variance. Using the original PCA with a sample covariance matrix, we reduce the dimensionality to only 58 leading eigenvectors. Accounting for the structures in data, which introduce spurious data component dependencies, we retain 95 leading eigenvectors. In the plot, we show the cumulative distributions of eigenvalues (5.46) for both cases.

on the corrected covariance matrix. We set a threshold of 0.9 for the cumulative distribution of eigenvalues (5.46). As expected, the original PCA reduces the number of dimensions in a more radical way, leaving 58 as opposed to 95 dimensions for the corrected covariance matrix. Moreover, 58 eigenvectors of the corrected covariance matrix explain only 79% of variance, suggesting that the original PCA leads to a more severe loss of information than controlled by the threshold parameter.

Chapter 6

Significance-based clustering algorithm

In Chapters 4 and 5, we discussed two concepts which concern the statistics of high dimensional data: an analytical approach for assessing the statistical significance of clusters and a model allowing for estimation of dependencies between data components. Solutions for both problems were based on probabilistic models for clustered and unclustered data, which we presented in Chapter 3. Here, we show the *significance-based clustering*, a method that joins these concepts. The algorithm is an extension of the expectation-maximisation algorithm for mixture-models, and it is designed to find only statistically significant clusters.

6.1 Introduction

In Chapter 4, we presented a method for estimating p-values of cluster scores in high-dimensional data. Such a p-value can be computed for a subset of data vectors to decide if they form a cluster: high p-values do not allow rejecting the null hypothesis, according to which the vectors are independently distributed. As such, the significance analysis can be used on top of any clustering algorithm that assumes a compatible cluster model. For a given dataset partition (a list of clusters), one would compute cluster score p-values for all clusters, and then reject the insignificant ones. The members of the rejected clusters are then assumed to follow the null model.

In here, we propose a *significance-based clustering* method which integrates significance analysis with the process of forming clusters: it returns clusters with p-value not greater than a given threshold, assigning the remaining elements to the background model. This approach is different from rejecting insignificant clusters *after* performing a clustering: the data vectors included in the background have an influence on the data log-likelihood function, and hence on the clusters. After rejecting a cluster, its elements can either be assigned to the background or to other clusters. Additional elements in a cluster have a direct effect on the cluster centre, its optimal score and the p-value.

As discussed earlier, we conducted the analytical p-value calculation for the linear cluster score: (i) for the directional-density-based scoring scheme, with length-normalised data and scalar product similarity measure and (ii) for the positional-bias-based scoring scheme, in a model with a Gaussian background, Euclidean distance based similarity measure, and with all clusters of the same width (parameter $\eta = 1$). Our significance-based clustering is implemented for these two models.

6.2 Previous work: semi-supervised and constrained clustering

An idea of performing a clustering with imposed constraints on clusters (here, the significance of cluster scores) has already been exploited in the context of *semi-supervised learning* and the so called *constrained clustering*. In these problems, a prior knowledge about the required data grouping is given, e.g. partial class-labelling for some of the elements or pairwise constraints on the elements naming the *must-links* and the *cannot-links*. These problems have been addressed with probabilistic approaches employing mixture-models.

Semi-supervised learning. Assume a dataset of N vectors, $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, is sampled independently from some (unknown) distribution $P(\mathbf{x}_i)$. In the *unsupervised learning* problems (clustering), the underlying distribution $P(\mathbf{x}_i)$ is assumed to be a mixture-model. The inference problem, as discussed in Chapter 5, is to estimate parameters of the mixture components.

In the *supervised learning* setting, the clusters (or rather classes) are "named". Formally, we are additionally provided with *data labels* $\mathbf{Y}$ (such labels are artificially introduced in the EM algorithm for the inference in the unsupervised problem). The complete data is $(\mathbf{X}, \mathbf{Y}) = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$. The problem is to infer the joint probability distribution of the data points and the labels, $P(\mathbf{x}, y)$. The difference to the unsupervised setting is that the vectors with the same label are not necessarily related to each other by enhanced mutual similarity in a standard sense. The relating property and the type of the underlying distribution may be complex and are yet to be determined.

The *semi-supervised classification* [111] is an in-between case: labels are available for part of the data vectors only. These labels can be regarded as *constraints* in the inference problem: they impose assignment of corresponding elements to specific clusters. The aim is still to infer the joint probability distribution $P(\mathbf{x}, y)$, but this time using the information from the unlabelled data as well. The information contained in the unlabelled data can be useful if an additional assumption is made: the property relating data vectors from the same class is indeed an enhanced mutual similarity for a known similarity measure.

Constrained clustering. A weaker way of introducing prior knowledge on the assignments of data vectors to clusters is by setting constraints on the resulting clustering. The prior knowledge can be of any kind: for example about the upper bound on the cluster sizes or one may enforce that the cluster sizes are "balanced" [5]. The most common constraint is to impose or forbid *links* (assignment to the same cluster) between pairs of elements in the data. In the probabilistic setting, this kind of constraints is introduced by means of a prior distribution, which assigns zero or small probability to solutions which violate the constraints [60, 22, 9, 62, 67]. This prior distribution is now part of the likelihood function optimised by the clustering. The EM algorithm can still be used to solve the constrained optimisation problem.

6.3 Significance-constrained mixture-model

The significance-based clustering uses a similar approach as the above discussed constrained clustering. However, in here the constraint is not imposed by the prior knowledge but by the expectation that clusters should significantly deviate from the background. Below we show how to introduce such constraints in the probabilistic framework of mixture-models and the EM algorithm.

6.3.1 Significance constraint in the prior probability

We introduce a constraint on the significance of clusters: we penalise an insignificant cluster by means of a prior distribution on assignment of vectors to this cluster (denoted by τ_k in the mixture model (5.13). Clusters with a log p-value greater than threshold t will have zero or very small prior probability τ_k.

Formally, to penalise insignificant clusters, we use a sigmoid function, defined for a cluster score S as

$$\mathrm{sig}(S) = \frac{1}{1 + e^{-\omega(t - \log p(S))}} \ , \tag{6.1}$$

where ω is a parameter specifying how strictly should the borderline values be treated and $p(S)$ is the p-value of cluster score S. This function has limit 1 for log p-values converging to $-\infty$, $\lim_{\log P(S) \to -\infty} \mathrm{sig}(S) = 1$. On the other hand, for insignificant clusters with log p-value converging to 0, the function is converging to $1/(1 + e^{-\omega t})$, which in turn converges to 0 for ω converging to ∞, $\lim_{\omega \to \infty, \log p(S) \to 0} \mathrm{sig}(S) = 0$. The function rapidly changes its value around t, the steepness of the step depends on the value of ω, as shown in Fig. 6.1.

The distribution of the prior probability of cluster k is then

$$p(y_k = 1) = \tau_k \mathrm{sig}(S_k) \ , \tag{6.2}$$
$$p(y_k = 0) = 1 - \tau_k \mathrm{sig}(S_k) \ , \tag{6.3}$$

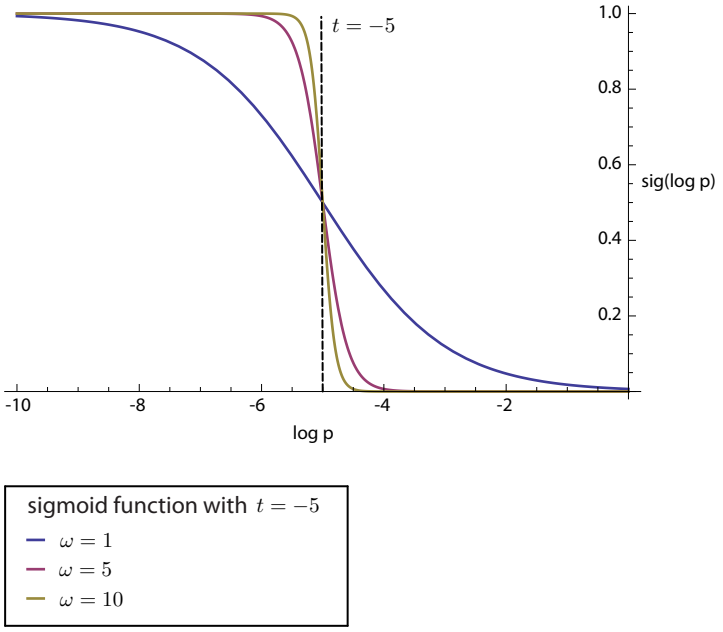

Figure 6.1: Sigmoid function penalising insignificant clusters. The plotted function is $\mathrm{sig}(\log p) = \frac{1}{1+e^{-\omega(t-\log p)}}$ with $t = -5$ and varying $\omega = 1, 5, 10$. The step becomes steeper with increasing ω.

where S_k is the score and τ_k is the mixing proportion of the kth cluster. In the unconstrained case, the prior probability $P(y_k = 1)$ is simply the mixing proportion τ_k.

6.3.2 Implementation of the EM algorithm

The difference to the EM algorithm described in section 5.4.2 appears in the E-step, in which which the posterior probabilities for cluster assignments are estimated. The maximisation step depends on the specific background and the cluster model. In here, we discuss two cases for which we have calculated the analytical cluster p-value: the directional-density-based model and the positional-bias-based model.

E-step. The E-step involves estimation of function $R(\Theta, \Theta^n) = \mathbb{E}_{\mathbf{Y}|\mathbf{X}} P(\mathbf{X}, \mathbf{Y}|\Theta)$, see Eq. (5.32). In particular, the conditional probability of cluster k given the data, $\rho_{ik} = P(y_{ik} - 1|\mathbf{x}_i)$ is affected by the change of the cluster prior: from an expansion with a Bayes' rule

$$\rho_{ik} = \frac{P(\mathbf{x}_i|y_{ik} = 1, \Theta)P(y_{ik} = 1)}{P(\mathbf{x}_i|\Theta)}$$

we now obtain

$$\rho_{i0} = \frac{\tau_0 P_0(\mathbf{x}_i|\mathbf{G})}{\tau_0 P_0(\mathbf{x}_i|\mathbf{G}) + \sum_{k=1}^{K} \tau_k \mathrm{sig}(S_k) Q(\mathbf{x}_i|\mathbf{z}_k, \eta_k, \mathbf{G})}, \tag{6.4}$$

$$\rho_{ik} = \frac{\tau_k \mathrm{sig}(S_k) Q(\mathbf{x}_i|\mathbf{z}_k, \eta_k, \mathbf{G})}{\tau_0 P_0(\mathbf{x}_i|\mathbf{G}) + \sum_{k=1}^{K} \tau_k \mathrm{sig}(S_k) Q(\mathbf{x}_i|\mathbf{z}_k, \eta_k, \mathbf{G})} \ . \tag{6.5}$$

After the algorithm has converged, the data elements are assigned to clusters based on the posterior probabilities:

$$c(i) = \arg\max_k \rho_{ik} \ . \tag{6.6}$$

M-step: the positional-bias based model. The objective function $R(\Theta, \Theta^n)$ (5.32) does not change for our significance-constrained model. Hence, the maximum-likelihood equations for the model parameters: cluster centres $\mathbf{z}_k$ (5.34) and the covariance matrix $\mathbf{G}$ (5.36), and the cluster mixing proportions τ_k (5.37), remain unchanged. Parameters η_k are not estimated in the course of the EM algorithm and are strictly fixed to value 1.

M-step: the directional-density based model. Under the length-constrained model, the covariance matrix $\mathbf{G}$ is provided on the input and is not estimated within the clustering procedure (as discussed in section 5.4.1). The vectors are normalised with respect to the metric defined by the inverse of $\mathbf{G}$, $\mathbf{H} = \mathbf{G}^{-1}$. The similarity measure is a scalar product of vectors, again with metric $\mathbf{H}$. Model parameters are cluster centres and cluster widths, $\Theta = \{\mathbf{z}_k, \eta_k : k = 1, \ldots, K\}$.

The probability distribution density function of a vector in a cluster, $Q(\hat{\mathbf{x}}|\hat{\mathbf{z}}, \eta, \mathbf{G}) \sim \delta(\hat{\mathbf{x}} \cdot \hat{\mathbf{x}} - M) e^{\eta \hat{\mathbf{x}} \cdot \hat{\mathbf{z}}}$ can be approximated, using the saddle point integration, by

$$Q(\mathbf{x}|\hat{\mathbf{z}}, \eta, \mathbf{G}) = \frac{1}{Z_0 Z_\eta} e^{-\frac{1}{2}\mathbf{x}\cdot\mathbf{x} + \eta \mathbf{x}\cdot\hat{\mathbf{z}}} \ , \tag{6.7}$$

where $Z_0 = \sqrt{\det(\mathbf{G})(2\pi/\gamma)^M}$, $Z_\eta = \exp\{M[(\gamma - 1)/2 + \eta^2/(2\gamma) - (\log\gamma)/2]\}$ is the normalization constant, with $\gamma = (1 + \sqrt{1 + 4\eta^2})/2$ (see Appendix A).

Approximation (6.7) is simply a Gaussian distribution with mean $\eta\hat{\mathbf{z}}/\gamma$ and covariance $\mathbf{G}/\gamma$. Under this distribution, the constraint on the length of vectors $\mathbf{x}$ is imposed only *softly*. It can be shown that vectors following distribution $Q(\mathbf{x}|\hat{\mathbf{z}}, \eta, \mathbf{G})$ are *expected* to have length $\sqrt{M}$ (with respect to metric $\mathbf{H}$),

$$\int_{\mathbb{R}^M} (\mathbf{x} \cdot \mathbf{H} \cdot \mathbf{x})^{\frac{1}{2}} Q(\mathbf{x}|\mathbf{z}, \eta, \mathbf{G}) d\mathbf{x} = \sqrt{M} \ . \tag{6.8}$$

The vectors are thus expected to be positioned near the surface of the sphere, with some fluctuations. The standard deviation of the length of a vector $\|\mathbf{x}\| = (\mathbf{x}\cdot\mathbf{H}\cdot\mathbf{x})^{\frac{1}{2}}$ is of order 1: hence the larger M, the smaller the length fluctuations with respect to the

expected length of vectors, and the"better" the approximation (6.7). In section 4.2, we presented in detail a similar argument for the background model $P_0(\mathbf{x})$. Notably, approximation (6.7) is also valid for the background model itself, via the identity $P_0(\mathbf{x}|\mathbf{G}) = Q(\mathbf{x}|\hat{\mathbf{z}}, \eta = 0, \mathbf{G})$.

Using the Gaussian approximation, we can easily compute the maximum-likelihood estimations of the model parameters. First, we introduce a set of Lagrange multipliers l_k to impose length constraints on inferred cluster centres $\hat{\mathbf{z}}_k$. The objective function, extending (5.32) by the set of Lagrange multipliers, is

$$\begin{aligned} R(\Theta, \Theta^n) &= \sum_{i=1}^{N} \left[\rho_{i0} \log \tau_0 P_0(\mathbf{x}_i|\mathbf{G}) + \sum_{k=1}^{K} \rho_{ik} \log \tau_k Q(\mathbf{x}_i|\hat{\mathbf{z}}_k, \eta_k, \mathbf{G}) \right] \\ &- \sum_{k=1}^{K} l_k(\hat{\mathbf{z}} \cdot \hat{\mathbf{z}} - M) \ . \end{aligned} \tag{6.9}$$

Let $\bar{\mathbf{x}}_k$ denote the average vector under cluster k, $\bar{\mathbf{x}}_k = \sum_{i=1}^{N} \rho_{ik} \mathbf{x}_i$. The maximum of (6.9) is at

$$\hat{\mathbf{z}}_k = 2\eta_k \bar{\mathbf{x}}_k / \tau_k, \tag{6.10}$$

$$\eta_k = (\tau_k M \bar{\mathbf{x}}_k \cdot \hat{\mathbf{z}}_k) \,/\, \left(\tau_k^2 M^2 - (\bar{\mathbf{x}}_k \cdot \hat{\mathbf{z}}_k)^2\right), \tag{6.11}$$

$$l_k = \frac{\eta_k}{2} \sqrt{\bar{\mathbf{x}}_k \cdot \bar{\mathbf{x}}_k / M}, \tag{6.12}$$

$$\tau_k = \sum_{i=1}^{N} \rho_{ik} / N \ . \tag{6.13}$$

Equations (6.10)-(6.13) are parameter update rules in the M-step.

Choosing the number of clusters. In section 5.4.2, we discussed model-selection by means of the Bayesian information criterion. The information criterion function has two components: the maximum log-likelihood value and the penalty term associated with model complexity (here, the number of clusters). The penalty term is crucial as it prevents selection of complex models over-fitted to the data. In particular, a model with N singleton clusters with centres $\mathbf{z}$ equal or pointing in the same directions as the data elements would give the highest log-likelihood.

Here, we incorporate a different penalty for model complexity: the significance constraint itself provides a penalty and an upper bound for the number of clusters. In particular, the above mentioned model with N singleton clusters would not pass the significance-criterion: (i) In the positional-bias-based model, a singleton cluster can have a significant score only if the vector is significantly longer than expected, i.e. is an outlier with a small probability density. The likelihood of all N data elements being significantly long is geometrically smaller. (ii) In the directional-density-based model, a singleton cluster has score $\eta \hat{\mathbf{x}} \cdot \hat{\mathbf{x}} - \log Z_\eta = \eta M - \log Z_\eta$, which is always insignificant (any data vector forms a cluster with this score value).

Taking these considerations into account, we use the maximum-likelihood criterion for choosing the number of clusters in the significance-based clustering: for a given threshold $t \leq 0$, we select the maximum-likelihood clustering satisfying a criterion that all clusters have p-values lower than e^t. The value of threshold t is a free-parameter, but it has a clear meaning: the minimum significance of a cluster.

6.4 Application to gene expression data

In Chapter 4, we showed on an example of gene-expression in yeast that there exists a remarkable agreement between the cluster score significance and the biological significance of clusters, as measured with enrichment in GO terms. To further investigate this agreement, we apply the significance-based clustering to two genome-wide expression datasets: yeast expression under environmental stress [42] and the human dataset with expression measured across multiple tissues [98]. We compare the *performance* of the algorithm, i.e. its ability to find biologically relevant clusters, to other frequently used clustering methods.

6.4.1 Significance-based clustering of gene expression data.

Both datasets were preprocessed in a standard way, as described in section 4.6.1. We tested both versions of the algorithm: the directional-density- and the positional-bias-based models. The directional-density-based version was applied on the length-normalised data $\hat{\mathbf{X}} = \{\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_N\}$. The normalisation was done with a metric $\mathbf{H}$ obtained with the positional-based version of the algorithm, applied on the original data $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ with significance threshold $t = -10$.

The significance-based clustering was run to find $K = 1, \ldots, 100$ clusters. For each k, we started with threshold $t_0 = 1$ (i.e. imposing no significance-constraint), with which we obtained an initial clustering $\mathbf{C}_k^{t_0}$, indexed both by the imposed number of clusters and the significance threshold. The subsequent clustering runs were as follows: for a given clustering $C_k^{t_i}$, we were gradually decreasing threshold t, setting t_{i-1} to log p-value of the *least significant* cluster in the clustering. Lowering the threshold may lead to an elimination of the least significant cluster and assignment of its elements to the background component. Alternatively, it can also lead to a reorganisation of existing clusters: the cluster may take on elements from other components (including the background), such that all clusters remain significant. The lowering of the threshold t was performed until there were no clusters left.

In Fig. 6.2 we show a series of such clusterings with starting $K = 25$, performed on both datasets and in each case with both implemented models. The scatter plots show how the clusters move in terms of their statistical significance ($-\log p(S)$) and their biological significance ($-\log p_{GO}$, see section 4.6.2)). The colours and sizes of dots correspond to the threshold t imposed in that clustering: the bigger the dot, the

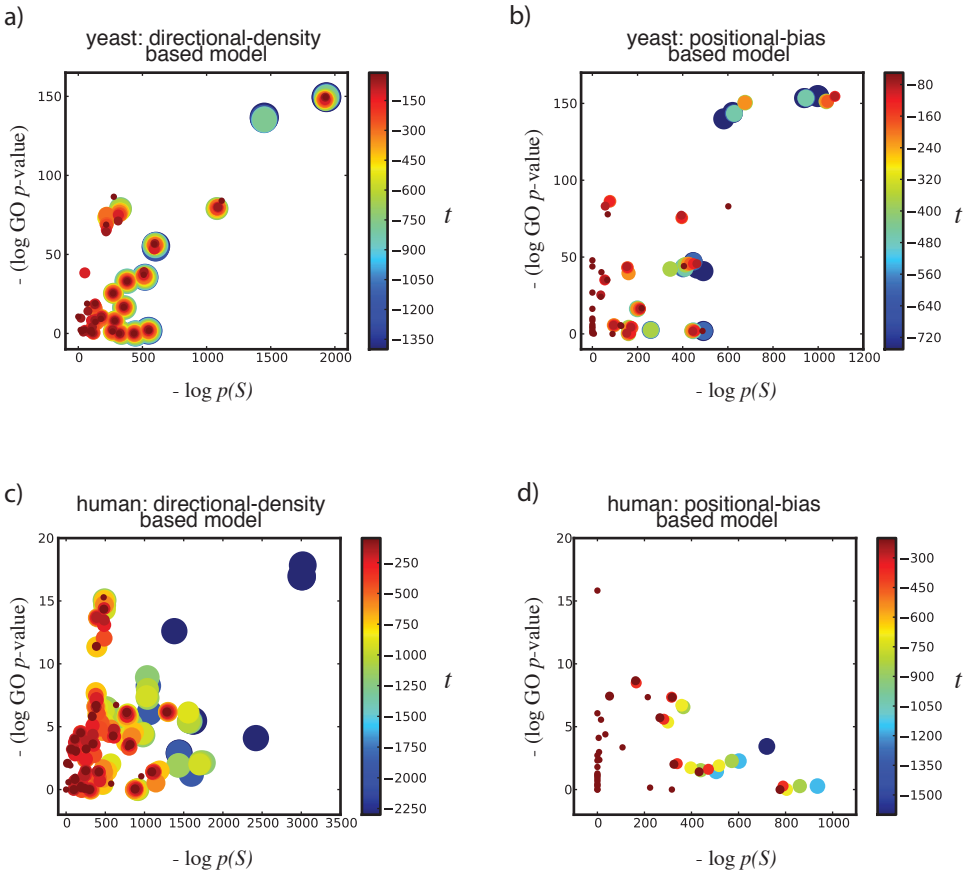

Figure 6.2: Example of significance-based clustering: the statistical and the biological significance of resulting clusters. The significance-based clustering was ran on the yeast ((a) and (b)) and the human ((c) and (d)) gene expression datasets. Initially the algorithm was ran with $K = 25$ and $t > 0$ (no significance threshold), both with the direction-density and the positional-bias based cluster models. In the subsequent runs, the threshold t was gradually decreased, leading to a decrease of the number of clusters. The scatter plots show the (negative) cluster score $\log p$-values against their (negative) GO $\log p$-values, for all the clusters found in any of the runs. The colours and sizes of dots reflect the threshold t used for a respective clustering. We expect large dots to be located towards the top-right side.

smaller the threshold (i.e the more severe the significance-constraint). The clusterings with a low threshold tend to have well GO-enriched clusters. The significance criterion has a strong effect on the clusterings obtained with the positional-bias-based algorithm: on both datasets, imposing even a weak significance-constraint, leads to eliminating of many clusters. The clusterings found with the directional-density-based model are in general more enriched in GO terms, suggesting that correlation is a better measure for capturing similarities in gene expression data than Euclidean distance.

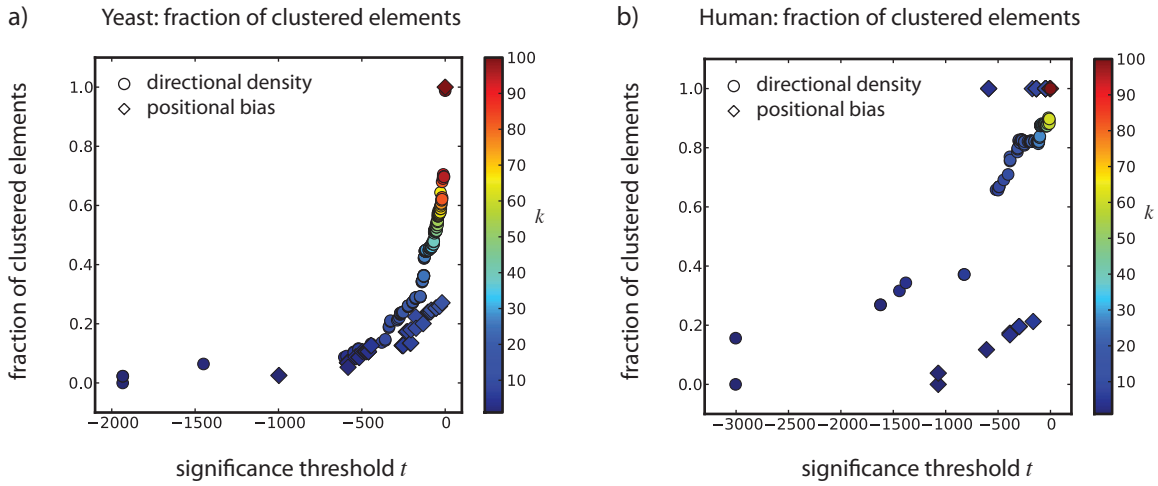

Figure 6.3: Statistics of the results of the significance-based clustering. We obtained a series of optimal significance-based clustering results for threshold value $t \in [-2000, 0]$ in yeast and $t \in [-3000, 0]$ in human datasets. A series of such clusterings has form $[C_{k_0}^{t_0}, C_{k_1}^{t_1}, \ldots, C_{k_{\min}}^{t_{\min}}]$. We plot the threshold value t against the fraction of data vectors that are assigned to clusters. Of course, the more severe the significance constraint (the lower the value of t) the more data vectors are assigned to the background component. The colours of markers correspond to the number of clusters. In both the yeast and the human datasets, the directional-density-based model yields more statistically significant clusters.

The series of clusterings for $K = 1, \ldots, 100$, resulted in a set of clustering results $\{\mathbf{C}_k^t\}$, where each clustering was defined by the number of clusters k and the imposed significance threshold t. For each threshold t, we then selected from $\{\mathbf{C}_k^t\}$ a clustering maximising the data log-likelihood, such that all clusters had a p-value lower than e^t. This step yielded a series of *optimal clusterings*, $[C_{k_0}^{t_0}, C_{k_1}^{t_1}, \ldots, C_{k_{\min}}^{t_{\min}}]$, for each threshold value sampled from the interval $[t_{\min}, t_0]$. In Fig. 6.3, we show the statistics of these clusterings. As the significance threshold t is decreasing, more elements are being assigned to the background component and the fraction of elements in clusters is decreasing. We also show the dependance of the number of clusters k on the imposed significance threshold t. As pointed out earlier, many of the clusters found under the positional-bias-based model, turned out statistically insignificant: in both datasets, if the significance constraint was imposed (i.e. the threshold t was negative), there were less than 20 clusters remaining in an optimal clustering.

6.4.2 Comparison with other clustering algorithms.

Validation by Gene Ontology terms enrichment. We will validate a clustering by considering Gene Ontology enrichment of clusters. For each cluster C in a clustering we compute the p-value $p_{\mathrm{GO}}(C)$ (see section 4.6.2). For the whole clustering with

K clusters, $\mathbf{C} = \{C_1, \ldots, C_K\}$, we now consider two measures: the *total significance* computed as a combined biological significance over all clusters

$$\Sigma_{\mathrm{GO}}(\mathbf{C}) = -\sum_{C \in \mathbf{C}} \log_{10} p_{\mathrm{GO}}(C) \ , \tag{6.14}$$

and the *median biological significance* of clusters,

$$\mathrm{median}_{\mathrm{GO}}(\mathbf{C}) = -\mathrm{median}_{C \in \mathbf{C}} \log_{10} p_{\mathrm{GO}}(C) \ . \tag{6.15}$$

Note that both significance measures are defined based on *negative* log *p-values* and are hence positive. The first measure is related to the number of clusters (we expect a clustering to find as many biologically relevant clusters as possible), while the second quantifies if clusters are on average biologically relevant. A good clustering maximises both measures. However, there is an obvious trade-off between the two quantities: the median is the best (the highest) for a clustering with a single, most enriched cluster and with the remaining data vectors assigned to the background component. Addition of any new cluster may decrease the median. The total significance shows a completely opposite trend: it increases when an enriched cluster is added or remains unchanged if a random-like cluster is added.

Other methods. To benchmark our significance-based approach, we clustered the considered gene expression datasets using standard algorithms: k-means clustering, and the hierarchical clustering with different cluster merging methods (Ward, single-linkage, complete-linkage, and average-linkage). Both algorithms use Euclidean distance as the similarity measure. Furthermore, we tested our approach against three less well-known algorithms: the model-based clustering [35], the information-based clustering [92] and the superparamagnetic clustering [12, 13].

The model-based clustering has been successfully applied to gene expression data [108] This method is very similar to our approach: it models the data as a mixture of K Gaussian components, where each component represents a cluster. An important feature is that it allows for imposing a parametrisation of the covariance matrices, to control their shape, both individually, and also across components. We used three such parameterisations: (1) VVV: each cluster has an individually estimated covariance matrix, (2) EEE: all clusters are imposed to have the same covariance matrix (similarly as in our positional-bias-based model) and (3) EII: all clusters are imposed to have the same *diagonal* covariance matrix (this model is a probabilistic analog of the the k-means algorithm). The algorithm uses the Bayesian information criterion to select the number of clusters. Its implementation is provided in the `mclust` R package [34].

The information-based clustering algorithm [92] compares vectors using the mutual information. For two vectors $\mathbf{x}_i$ and $\mathbf{x}_j$, the mutual information is computed as

$$s(\mathbf{x}_j, \mathbf{x}_j) = \sum_{\mu} P_{ij}(x_i^\mu, x_j^\mu) \log \frac{P_{ij}(x_j^\mu, x_j^\mu)}{P_i(x_i^\mu) P_j(x_j^\mu)} \ , \tag{6.16}$$

where $P_i(x)$ and $P_j(x)$ are distributions of the expression values over experiments of gene i and gene j, and $P_{ij}(x_1, x_2)$ is the joint distribution of expression of these two genes under the same experiment. Distributions $P_i(x)$, $P_j(x)$ and $P_{ij}(x_1, x_2)$ are empirically estimated from data. In the estimation, it is assumed that multiple experiments provide independent drawings from the same probability distribution. This assumption is in disagreement with the fact that the gene expression experiments may show different degrees of dependency (see Chapter 5). The advantage of the mutual information is that, contrary to Euclidean distance and the correlation, it can capture non-linear dependencies between vectors.

The superparamagnetic clustering [12, 13] has also been applied in the context of gene expression data analysis [44, 66]. This approach does not make specific assumptions about the underlying distribution of data nor about cluster characteristics, such as its shape or size. It is inspired by the Potts spin model from statistical mechanics: the system elements are N *spins* s_i which can take integer values from $\{1, \ldots, K\}$. The spins symbolise assignment of data elements to one of K clusters. The Hamiltonian of this system (i.e. the energy/scoring function of a clustering) is

$$\mathcal{H}(s_1, \ldots, s_N) = -\sum_{i,j=1}^{N} J(\|\mathbf{x}_i - \mathbf{x}_j\|)\delta_{s_i,s_j} \ , \tag{6.17}$$

where $J(\|\mathbf{x}_i - \mathbf{x}_j\|)$ is a positive and monotonically decreasing function of the distance between vectors $\mathbf{x}_i$ and $\mathbf{x}_j$. The summation is performed only over pairs of vectors assigned to the same cluster, which is here imposed by the Kronecker-delta function δ_{s_i,s_j}, which takes value 1 of $s_i = s_j$ and 0 otherwise. As a result of optimisation of (6.17), dense regions of "interacting" elements emerge. This algorithm implicitly assumes an interaction-based cluster generating process, which we discussed in the Introduction.

The superparamagnetic algorithm additionally implements a stability criterion for clusters [64] (see section 4.5): it selects the number of clusters K which raises the most *stable* clusters. Hence, the algorithm returns a *single*, optimally stable data partition, instead of a *family* of data partitions, for each K.

Significance-constraint improves biological relevance of clusters. In Fig. 6.4 and Fig. 6.5, we compare performance of the significance-based clustering to the other methods, run on the yeast [42] and the human [98] datasets respectively. In the figures, we show scatter plots of the median $\text{median}_{\text{GO}}$ (6.15) versus the total Σ_{GO} (6.14) biological significance for each clustering. According to our evaluation measures, the most "biologically relevant" clusterings are located towards the top-right corner in the plots. The colours of markers reflect the number of clusters in the corresponding clustering.

The figures include a scatter plot for each of the compared method. On one scatter plot we plot the results of the compared method and the results of both versions of the

significance-based algorithm. For the significance-based clustering (abbreviated with SBC), we plot a family of dots, one for each applied significance threshold t_i (which yield clusterings $C_{k_0}^{t_0}$,$C_{k_1}^{t_1}$,..., $C_{k_{\min}}^{t_{\min}}$). For the information based clustering, the hierarchical clustering and the k-means algorithm, we plot a family of dots for k densely sampled from interval $[1, 100]$. We present the results of the hierarchical clustering with the Ward cluster-merging criterion, which showed the best performance among all considered criterions. The model-based clustering (`mclust`) returns the optimal BIC clustering for a given class of models. We plot results for three classes of models: VVV, EEE and EII (see description above). The superparamagnetic clustering returns one, optimally stable clustering.

In application to the yeast dataset, the best performance was recorded by the directional-density-based version of the SBC algorithm. For the significance threshold $t = -100$, the algorithm resulted in 30 clusters with the total significance, $\Sigma_{\mathrm{GO}} = 730.975$, and the median significance, $\mathrm{median}_{\mathrm{GO}} = 17.53$. The model-based clustering with the EEE model found only 17 clusters, both with lower total significance $\Sigma_{\mathrm{GO}} = 559.54$ and lower $\mathrm{median}_{\mathrm{GO}} = 16.16$. Other clustering algorithms had on average 2-fold lower median significance than the SBC clusterings with the same number of clusters. Notably, the superparamagnetic clustering reported a poor performance: even though it found as much as 49 clusters, they were of a low total significance $\Sigma_{\mathrm{GO}} = 469.53$ and $\mathrm{median}_{\mathrm{GO}} = 2.89$. For a comparison, the directional-density based SBC clustering with 49 clusters (obtained for significance threshold $t = -51.33$) had total significance $\Sigma_{\mathrm{GO}} = 764.86$ and $\mathrm{median}_{\mathrm{GO}} = 7.97$. The positional-bias-based SBC algorithm resulted in few clusters (at most 16 for $t < 0$), but they showed high average biological significance, in the worst case $\mathrm{median}_{\mathrm{GO}} = 19.86$.

The general observations on the performance of the algorithms are similar in application to the human dataset. The difference is that the GO-annotation of human data is poorer and hence the GO-enrichment p-values are much lower than in the case of yeast clusters. The SBC-algorithm based on the directional-density model results in the most biologically significant family of clusterings, with respect to both measures (Σ_{GO} and median_{GO}). In one case, the information based clustering obtained similarly good results: for 10 clusters, it returned a clustering with total significance $\Sigma_{\mathrm{GO}} = 55.16$, and $\mathrm{median}_{\mathrm{GO}} = 5.36$. A corresponding SBC clustering with 10 clusters yielded $\Sigma_{\mathrm{GO}} = 54.72$, and $\mathrm{median}_{\mathrm{GO}} = 5.74$. The superparamagnetic algorithm again showed poor performance, resulting in 43 clusters with $\Sigma_{\mathrm{GO}} = 20.93$, and $\mathrm{median}_{\mathrm{GO}} = 0.04$. This result can again be compared to SBC results of 43 clusters (obtained with significance threshold $t = -40.38$) resulting in $\Sigma_{\mathrm{GO}} = 88.69$ and $\mathrm{median}_{\mathrm{GO}} = 1.32$) The positional-bias-based SBC algorithm found only at most 10 clusters with a statistically significant score. This clustering was characterised by $\Sigma_{\mathrm{GO}} = 40.67$ and $\mathrm{median}_{\mathrm{GO}} = 3.31$.

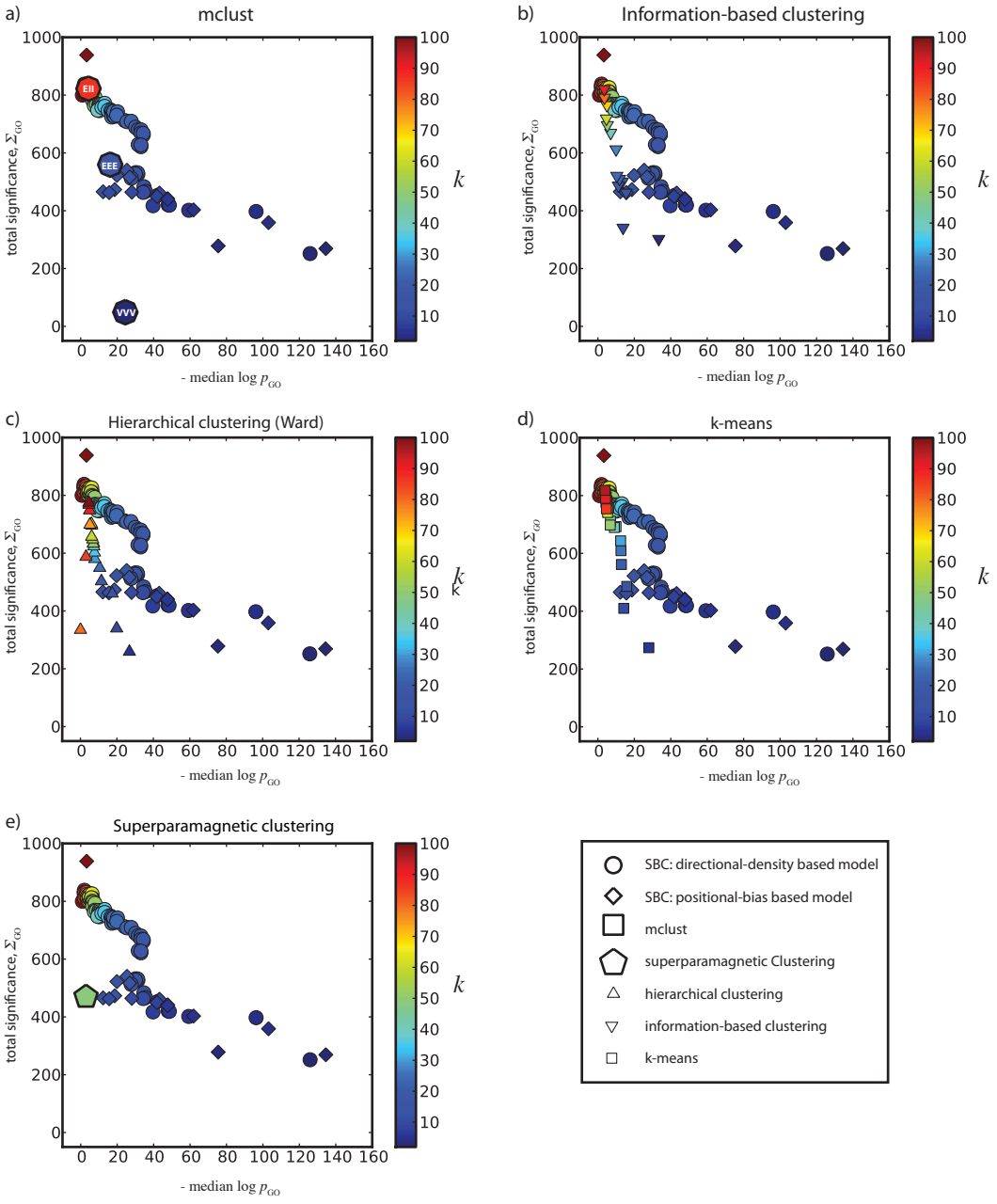

Figure 6.4: Biological validation of clusters in yeast: significance-based clustering compared to other methods. The median biological significance (6.15) is plotted against the total biological significance (6.14), for different clustering algorithms and across different parameter choices (see legend and text). We expect a good clustering to be placed towards the top-right corner in the plots.

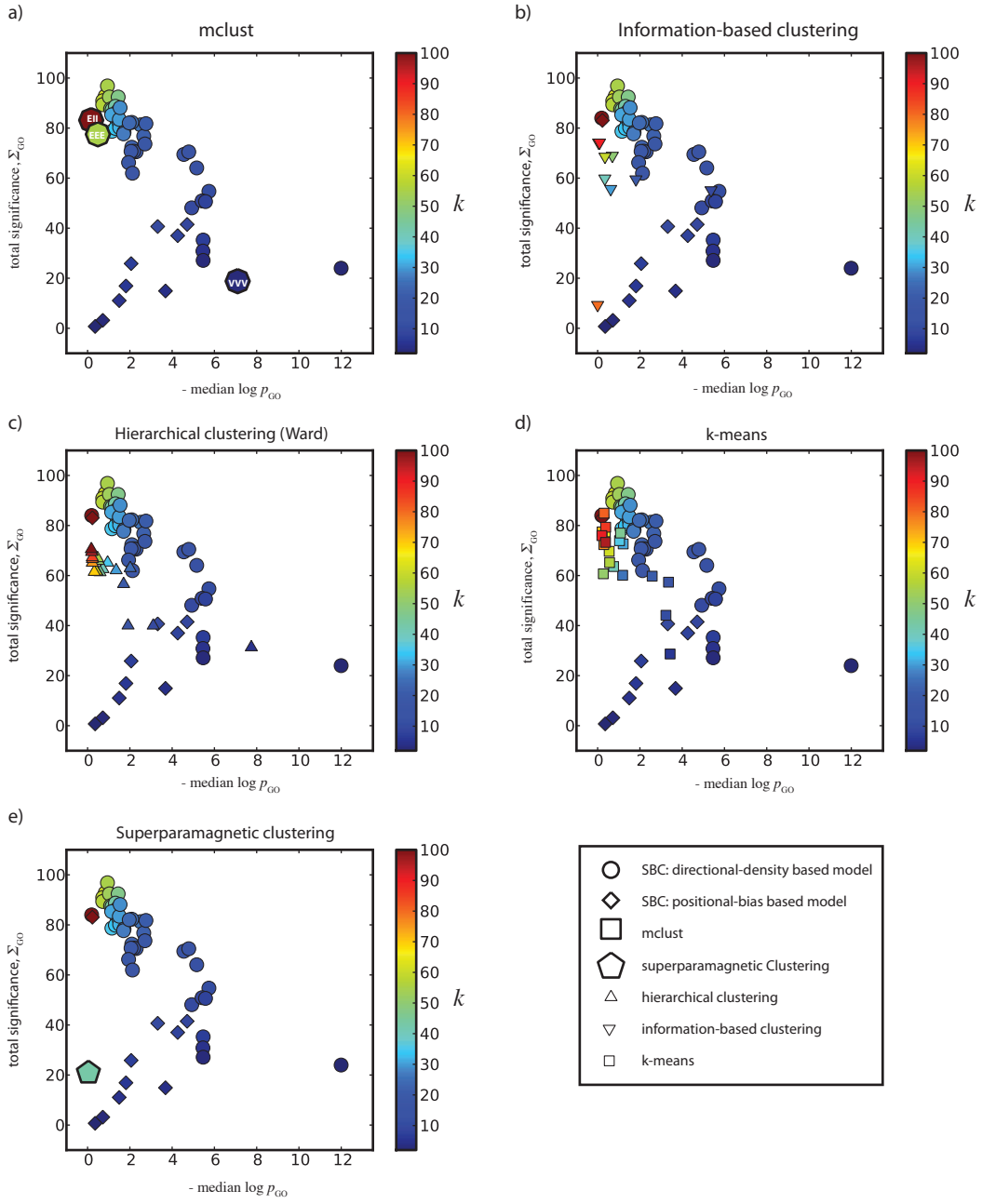

Figure 6.5: Biological validation of clusters in human: significance-based clustering compared to other methods. The median biological significance (6.15) is plotted against the total biological significance (6.14), for different clustering algorithms and across different parameter choices (see legend and text). We expect a good clustering to be placed towards the top-right corner in the plots.

6.5 Summary

We proposed a clustering algorithm based on a probabilistic framework of clusters presented in Chapter 3 and its statistical significance analysis described in Chapter 4. Using a Bayesian approach, we introduced a prior distribution on probability of assignment of data elements to clusters, which depends on the cluster score p-value. The cluster prior penalises clusters with a p-value larger than a given threshold value, by yielding small data log-likelihood values for such solutions. As a result, clustering solutions include only significant clusters, with non-clustered data elements assigned to the background component.

Our algorithm finds tightly co-expressed groups of genes, which are unlikely to be observed by chance and hence are hypothesised to be an outcome of an underlying biological process. We applied the algorithm to genome-wide expression datasets of yeast and human. The significance constraint improved the biological relevance of clusters (measured with GO-term enrichment). It also worked favourably as compared to other clustering methods.

The comparison between two versions of our algorithm: based on the directional-density model and on the positional-bias based model, shows an advantage of the former. The advantage may have two sources:

1. correlation being a more relevant similarity measure than the Euclidean distance,
2. regulation of cluster "widths" with the variable parameter η.

The positional-bias model defines clusters solely based on their position. Moreover, it assumes that all clusters, including the background component, have the same width. In the application to the yeast and human datasets, there were less than 20 significant (p-value $\leq 10^{-3}$) clusters found under this model, while using the directional-density-based model, there were respectively 82 and 61 clusters for the same p-value threshold, see Fig. 6.3. The possible reason is that the "true" clusters indeed have variable widths; the positional-bias-based model is not able to capture small and tight clusters. Such clusters would be captured by another cluster model discussed in Chapter 3: the point-density-and-positional-bias-based one. The significance-based algorithm could also be implemented for this model, once the distribution of the quadratic cluster score has been solved.

An interesting observation is that the superparamagnetic algorithm showed a poor performance in our benchmarking experiment. This algorithm assumes an interaction-based cluster generating process, and it allows clusters to have an arbitrary shape. The other algorithms, which we considered, seek clusters that have ellipsoidal-shape (in terms of the applied similarity measure): the model-based clustering, the information-based clustering, and k-means; or clusters that are convex: the hierarchical clustering. In application to the gene expression datasets, all these methods showed better performance than the superparamagnetic algorithm; this suggests that the "centric" cluster

model, like the one adopted in this thesis, may be more appropriate for modelling of gene co-expression.

Chapter 7

Summary and outlook

This thesis establishes a statistical grounding for cluster analysis in high-dimensional data. Below we summarise the important developments.

Probabilistic models for clustered data. We proposed a unified framework of a probabilistic theory for clusters. Motivated by the characteristics of gene expression, we considered different properties defining a cluster: the point density, the positional bias and the directional density. These properties are related to different choices of a similarity measure and of a background distribution for unclustered vectors. We considered several combinations of such background distributions and similarity measures, and we arrived at well-defined scoring schemes for clusters.

Of course, the model choices we made are not unique for all applications. In particular, we focussed on one class of "centric" clusters. Still, the proposed framework is general enough so that other solutions can be implemented, to account for properties of other types of data.

Statistical significance of clusters. Given a group of vectors, do they show more mutual similarity than could have happen by chance, without an underlying mechanism? Standard statistical methods such as the t-test can distinguish significant correlations between *pairs* of expression patterns from spurious correlations, which can easily arise in the case of a large number of genes and *few* expression data for each gene. Clusters of *many* co-expressed genes contain more information on function than pairs, but their statistical significance is more difficult to estimate. In particular, for high-dimensional datasets, spurious clusters can occur in many centre directions: this creates an intricate multiple-testing problem. Here, we have established a link between quenched disorder physics and the multiple-testing statistics in clustering. This connection applies to a much broader class of problems, which involve the parallel testing of an exponentially large number of hypotheses on a single dataset. If the scores of different hypotheses are correlated with each other, the distribution of the maximal score may diverge from the distributions described by universality classes of extreme value statistics. It may still be computable by the methods used here: the state space of the problem is the set of all hypotheses tested (here, the centres

of all clusters), and configurations of data vectors generated by a null model act as quenched random disorder.

Estimation of data component dependencies. We proposed a method for estimation of data component dependencies. Gene-expression experiments are very often correlated: for example biological or technical replicates which are expected to be a very close repetition of the same expression profiles, time course data with successive time points measurements, or expression measured in evolutionary related tissues. These dependencies need to be accounted for when computing similarities between data elements, to properly weigh the information coming from each of the experiments. We showed that presence of cluster structures in data is a strong confounding factor in estimation of such dependencies. Our solution is a mixture-model based method, which estimates the dependencies and finds clusters in data at the same time. We successfully applied our method to classify tumour samples from several gene expression datasets.

Our method is of importance for any application that relies on the estimation of the data covariance matrix. An example is the principal component analysis, which is used to determine the independent coordinates best-describing the data.

Gene expression is not the only type of data with prevalent data component dependencies. Remaining in the field of computational biology, examples include different types of binding assay experiments, such as the protein-DNA, or antigen-antibody affinity measurements.

Significance-constrained clustering algorithm. Merging the above concepts, we proposed a probabilistic clustering approach, which seeks the best representation of data as a mixture of the background and of clusters characterised by a statistically significant score. In the application to two gene expression datasets, we showed that the statistical significance of clusters correlates with their "biological relevance" (as measured by enrichment in Gene Ontology terms). We compared clusters obtained with our method to other commonly used clustering methods; we found that incorporating the significance-constraint improved the overall biological relevance of clusters.

Extension to biclustering. In Chapter 5, we showed that similarities of data elements (rows) and data components (columns) are intrinsically dependent: clustering of elements may lead to overestimation of component similarities and vice versa. A straightforward observation follows: what if clusters are observed only on a subset of data components, for example only in healthy or only in tumour tissues? This problem has been addressed by *biclustering* algorithms [71]. Biclustering is similar in spirit to detecting differential co-expression, but is done in an *unsupervised* manner: the set of experiments with differing patterns is to be determined and is not given *a priori*. An advantage of biclustering, over the standard clustering, is that a data

element can belong to many bicluster groups. This situation is much closer to the biological context: a gene can take part in many biological processes.

With our method for estimation of data component dependencies, we arrive at a well-defined concept of a bicluster: in a principled way, we can distinguish the true clustering of *data elements* from the spurious ones, resulting from the orthogonal clustering of *data components*. The extension of our clustering algorithm to biclustering would affect all concepts developed in this thesis; starting from the adjustment of the scoring schemes in a bi-cluster theory, through the modification of the mixture-model such that it allows single elements to belong to many biclusters, and finishing with the statistical significance analysis for biclusters.

Here we sketch a possible mixture-model extension. Let us assume that a bicluster $B(E)$ is defined on a subset E of data components, $E = \{\mu_1, \ldots, \mu_L\} \subseteq \{1, \ldots, M\}$. Let $\mathbf{x}[E]$, $\mathbf{z}[E]$ and $\mathbf{G}[E]$ be vectors and a covariance matrix restricted to components from E. We will also use $\mathbf{H}[E]$ to denote the inverse of covariance matrix $\mathbf{G}[E]$, i.e $\mathbf{H}[E] = \mathbf{G}[E]^{-1}$.

Just as before, we can assume that clusters follow a Gaussian distribution on a subset E with centre $\mathbf{z}[E]$,

$$Q(\mathbf{x}[E]|\mathbf{z}[E], \eta, \mathbf{G}[E]) = \frac{1}{Z_\eta[E]} e^{-\frac{\eta}{2}(\mathbf{x}[E]-\mathbf{z}[E])\cdot\mathbf{H}[E]\cdot(\mathbf{x}[E]-\mathbf{z}[E])} , \tag{7.1}$$

with the normalisation constant $Z_\eta[E] = \sqrt{(2\pi\eta)^L/(\det \mathbf{H}[E])}$. Similarly, the background model can also be defined for a subset of data components E,

$$P_E(\mathbf{x}[E]) = \frac{1}{Z_0[E]} e^{-\frac{1}{2}\mathbf{x}[E]\cdot\mathbf{H}[E]\cdot\mathbf{x}[E]} , \tag{7.2}$$

with $Z_0[E] = \sqrt{(2\pi)^L/(\det \mathbf{H}[E])}$.

Assume there are K biclusters in data, $B_1(E_1), \ldots, B_K(E_K)$. A data element can belong to many biclusters of different data components, so there are 2^K possible combinations of a cluster assignment. Let $\mathbf{B}_i$ be a binary vector of length K acting as a characteristic function of the i-th combination. The probability density function of such a combination is

$$f(\mathbf{x}|\mathbf{B_i}) = P_{E_i^0}(\mathbf{x}_i[E_i^0]) \prod_{k=1}^{K} B_i^k Q_{E_k}(\mathbf{x}[E_k]|\mathbf{z}[E_k], \eta_k, \mathbf{G}[E_k]) , \tag{7.3}$$

where E_i^0 is a set of data components not covered by any of the biclusters from the ith configuration, $E_i^0 = \{1, \ldots, M\} \setminus \bigcup_{i=1}^{K} E_i$.

The full mixture model is a sum over all such combinations,

$$f(\mathbf{x}) = \sum_{i=1}^{2^K} \tau_i f(\mathbf{x}|\mathbf{B}_i) , \tag{7.4}$$

where τ_i are the mixing proportions as before.

The formulation of the bicluster mixture-model appears straightforward. However, the numerical implementation of the model-based biclustering leads to an algorithmic challenge: the search space becomes exponential both in the number of biclusters and in the number of data components, from which the bicluster subsets have to be selected. This problem could be approached with message-passing techniques, recently introduced to clustering [37, 63].

Another challenge is the significance-analysis for biclustering. For a given cluster centre $\mathbf{z}$, we can now choose the data components which contribute to the cluster score. Because of that, the negative score contributions can be more easily eliminated and there are many more ways to achieve a high score. The maximisation of the bicluster score is done both over possible *cluster centres* $\mathbf{z}$ and over *subsets of data components*. The statistics of bicluster scores may significantly differ from the statistics of scores of regular clusters.

Bibliography

[1] H. Akaike. A new look at the statistical model identification. *IEEE Trans Automat Contr*, 19:716–23, 1974.

[2] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. *Molecular Biology of the Cell*. Garland Science, New York, 2002.

[3] U. Alon, N. Barkai, D. Notterman, K. Gish, S. Ybarra, D. Mack, and A. Levine. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *P Natl Acad Sci USA*, 96(12):6745, 1999.

[4] S. F. Altschul, R. Bundschuh, R. Olsen, and T. Hwa. The estimation of statistical parameters for local alignment score distributions. *Nucleic Acids Res*, 29(2):351–361, 2001.

[5] A. Banerjee and J. Ghosh. Scalable clustering algorithms with balancing constraints. *Data Min Knowl Disc*, 13(3):365–395, Jan 2006.

[6] J. D. Banfield and A. E. Raftery. Ice floe identifcation in satellite images using mathematical morphology and clustering about principle curves. *J Am Stat Assoc*, 87:7–16, 1992.

[7] M. Bengtsson, A. Ståhlberg, P. Rorsman, and M. Kubista. Gene expression profiling in single cells from the pancreatic islets of langerhans reveals lognormal distribution of mrna levels. *Genome Res*, 15(10):1388–92, Oct 2005.

[8] S. T. Bennett, C. Barnes, A. Cox, L. Davies, and C. Brown. Toward the 1,000 dollars human genome. *Pharmacogenomics*, 6(4):373–82, Jun 2005.

[9] M. Bilenko, S. Basu, and R. J. Mooney. Integrating constraints and metric learning in semi-supervised clustering. *Proceedings of the twenty-first international conference on Machine learning*, Jan 2004.

[10] C. P. Bird, B. E. Stranger, M. Liu, D. J. Thomas, C. E. Ingle, C. Beazley, W. Miller, M. E. Hurles, and E. T. Dermitzakis. Fast-evolving noncoding sequences in the human genome. *Genome Biol*, 8(6):R118, Jan 2007.

[11] I. Birol, S. D. Jackman, C. B. Nielsen, J. Q. Qian, R. Varhol, G. Stazyk, R. D. Morin, Y. Zhao, M. Hirst, J. E. Schein, D. E. Horsman, J. M. Connors, R. D. Gascoyne, M. A. Marra, and S. J. M. Jones. De novo transcriptome assembly with ABySS. *Bioinformatics*, 25(21):2872–2877, Nov 2009.

[12] M. Blatt, S. Wiseman, and E. Domany. Superparamagnetic clustering of data. *Phys Rev Lett*, 76:3251–3254, 1996.

[13] M. Blatt, S. Wiseman, and E. Domany. Data clustering using a model granular magnet. *Neural Comput*, 9(8):1805–1842, Nov 1997.

[14] A. Blum. Thoughts on clustering. In *NIPS 2009 Workshop on Clustering Theory, Vancouver, Canada*, December 2009.

[15] J. Bouchaud and M. Mézard. Universality classes for extreme-value statistics. *J Phys A Math Gen*, 30(23):7997–8015, Dec 1997.

[16] P. C. Boutros and A. B. Okey. Unsupervised pattern recognition: an introduction to the whys and wherefores of clustering microarray data. *Brief Bioinform*, 6(4):331–43, Dec 2005.

[17] A. Bovier. *Statistical Mechanics of Disordered Systems: A Mathematical Perspective*. Cambridge University Press, Cambridge, UK, June 2006.

[18] S. M. Brady, D. A. Orlando, J.-Y. Lee, J. Y. Wang, J. Koch, J. R. Dinneny, D. Mace, U. Ohler, and P. N. Benfey. A high-resolution root spatiotemporal map reveals dominant expression patterns. *Science*, 318(5851):801–806, Nov 2007.

[19] J. M. Buhmann. Information theoretic model validation for clustering. In *Information Theory Proceedings (ISIT), 2010 IEEE International Symposium on Information Theory*, pages 1398 – 1402, June 2010.

[20] J. G. Campbell, C. Fraley, D. Stanford, F. Murtagh, and A. E. Raftery. Model-based methods for real-time textile fault detection. *Int J of Imag Syst Tech*, 10:339–346, 1999.

[21] J. K. Choi, U. Yu, O. J. Yoo, and S. Kim. Differential coexpression analysis using microarray data and its application to human cancer. *Bioinformatics*, 21(24):4348–55, Dec 2005.

[22] I. G. Costa, A. Schonhuth, C. Hafemeister, and A. Schliep. Constrained mixture estimation for analysis and robust classification of clinical time series. *Bioinformatics*, 25(12):i6–i14, Jun 2009.

[23] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley, New York, 1991.

[24] A. Dasgupta and A. E. Raftery. Detecting features in spatial point processes with clutter via model-based clustering. *J Am Stat Assoc*, 93:294–302, 1998.

[25] A. Dembo and S. Karlin. Strong limit theorems of empirical functionals for large exceedances of partial sums of i.i.d variables. *Ann Probab*, 19(4):1737–1755, 1991.

[26] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood for incomplete data via the EM algorithm. *J R Stat Soc*, 39(1), 1977.

[27] E. Domany. Cluster analysis of gene expression data. *J Stat Phys*, 110(3):1117–1139, 2003.

[28] B. Efron. Bootstrap methods: another look at the jackknife. *Ann Stat*, Jan 1979.

[29] B. Efron, E. Halloran, and S. Holmes. Bootstrap confidence levels for phylogenetic trees. *Proc Natl Acad Sci USA*, 93(14):7085–90, Jul 1996.

[30] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *P Natl Acad Sci USA*, 95(25):14863–68, December 1998.

[31] A. Engel. Complexity of learning in artificial neural networks. *Theor Comput Sci*, 265(1-2):285–306, 2001.

[32] J. Felsenstein. Confidence limits on phylogenies: an approach using the bootstrap. *Evolution*, 39(4):783–791, Jul 1985.

[33] R. S. Finn, J. Dering, D. Conklin, O. Kalous, D. J. Cohen, A. J. Desai, C. Ginther, M. Atefi, I. Chen, C. Fowst, G. Los, and D. J. Slamon. Pd 0332991, a selective cyclin d kinase 4/6 inhibitor, preferentially inhibits proliferation of luminal estrogen receptor-positive human breast cancer cell lines in vitro. *Breast Cancer Res*, 11(5):R77, Jan 2009.

[34] C. Fraley and A. Raftery. *mclust: Model-Based Clustering / Normal Mixture Modeling*, 2009. R package version 3.1-10.3.

[35] C. Fraley and A. E. Raftery. Model-based clustering, discriminant analysis, and density estimation. *J Am Stat Assoc*, 97:611–631, 2002.

[36] B. J. Frey and D. Dueck. Clustering by passing messages between data points. *Science*, 315(5814):972–976, February 2007.

[37] B. J. Frey and D. Dueck. Clustering by passing messages between data points. *Science*, 315(5814):972 – 976, February 2007.

[38] C. J. G., F. C., M. F., and R. A. E. Linear flaw detection in woven textiles using model-based clustering. *Pattern Recogn Lett*, 18:1539–1548, 1997.

[39] J. Galambos. *The Asymptotic Theory of Extreme Order Statistics*. Malabar, FL: Krieger, 1987.

[40] D. Gao, J. Kim, H. Kim, T. L. Phang, H. Selby, A. C. Tan, and T. Tong. A survey of statistical software for analysing RNA-seq data. *Hum Genomics*, 5(1):56–60, Oct 2010.

[41] E. Gardner and B. Derrida. Optimal storage properties of neural network models. *J Phys A Math Gen*, 21(1):271–284, 1988.

[42] A. P. Gasch, P. T. Spellman, C. M. Kao, O. Carmel-Harel, M. B. Eisen, G. Storz, D. Botstein, and P. O. Brown. Genomic expression programs in the response of yeast cells to environmental changes. *Mol Biol Cell*, 11(12):4241–4257, December 2000.

[43] L. Gautier, L. Cope, B. M. Bolstad, and R. A. Irizarry. affy—analysis of affymetrix genechip data at the probe level. *Bioinformatics*, 20(3):307–315, Jul 2004.

[44] G. Getz, E. Levine, and E. Domany. Coupled two-way clustering analysis of gene microarray data. *P Natl Acad Sci USA*, 97(22):12079–84, Oct 2000.

[45] S. Gideon. Estimating the dimension of a model. *Ann Stat*, 6(2):461–464, 1978.

[46] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander. Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. *Science*, 286(5439):531–537, October 1999.

[47] S. Grossmann, S. Bauer, P. N. Robinson, and M. Vingron. Improved detection of overrepresentation of Gene-Ontology annotations with parent child analysis. *Bioinformatics*, 23(22):3024–3031, November 2007.

[48] E. J. Gumbel. *Statistics of Extremes*. Columbia University Press, New York, 1958.

[49] J. A. Hertz, A. S. Krogh, and R. G. Palmer. *Introduction To The Theory Of Neural Computation*. Addison-Wesley, Redwood City CA, 1991.

[50] L. Hubert and P. Arabie. Comparing partitions. *J classif*, 2(1):193–218, 1985.

[51] D. A. Huse and C. L. Henley. Pinning and roughening of domain walls in ising systems due to random impurities. *Phys Rev Lett*, 54(25):2708–2711, Jun 1985.

[52] T. Hwa and M. Lässig. Similarity-detection and localization. *Phys Rev Lett*, 76:2591–2594, 1996.

[53] T. Hwa and M. Lässig. Optimal detection of sequence similarity by local alignment. In *RECOMB '98 Proceedings of the second annual international conference on Computational molecular biology*, January 1998.

[54] E. T. Jaynes. Information theory and statistical mechanics. *Phys Rev*, 106(4):620–630, May 1957.

[55] I. Johnstone. On the distribution of the largest eigenvalue in principal components analysis. *Ann Stat*, 29:295–327, 2001.

[56] I. T. Jolliffe. *Principal Component Analysis*. Springer, New York, second edition, October 2002.

[57] S. Karlin and S. F. Altschul. Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *P Natl Acad Sci USA*, 87(6):2264–2268, March 1990.

[58] J. Khan, J. S. Wei, M. Ringnér, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, and P. S. Meltzer. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nat Med*, 7(6):673–679, June 2001.

[59] D. Kostka and R. Spang. Finding disease specific alterations in the co-expression of genes. *Bioinformatics*, 20 Suppl 1:i194–9, Aug 2004.

[60] T. Lange, M. Law, A. Jain, and J. Buhmann. Learning with constrained and unlabelled data. In *Computer Vision and Pattern Recognition, 2005*, volume 1, pages 731 – 738, june 2005.

[61] J. Lapointe, C. Li, J. P. Higgins, M. van de Rijn, E. Bair, K. Montgomery, M. Ferrari, L. Egevad, W. Rayford, U. Bergerheim, P. Ekman, A. M. DeMarzo, R. Tibshirani, D. Botstein, P. O. Brown, J. D. Brooks, and J. R. Pollack. Gene expression profiling identifies clinically relevant subtypes of prostate cancer. *P Natl Acad Sci USA*, 101(3):811–816, January 2004.

[62] M. H. C. Law, A. Topchy, and A. K. Jain. Clustering with soft and group constraints. *Lect Notes Comput Sc*, 3138:662–670, 2004.

[63] M. Leone, Sumedha, and M. Weigt. Clustering by soft-constraint affinity propagation: applications to gene-expression data. *Bioinformatics*, 23(20):2708–15, Oct 2007.

[64] E. Levine and E. Domany. Resampling method for unsupervised estimation of cluster validity. *Neural Comput*, 13(11):2573–93, Nov 2001.

[65] P. Lloyd and P. R. Best. A variational approach to disordered systems. *J Phys C Solid State*, 8, Jan 1975.

[66] J. Lotem, D. Netanely, E. Domany, and L. Sachs. Human cancers overexpress genes that are specific to a variety of normal human tissues. *P Natl Acad Sci USA*, 102(51):18556–61, Dec 2005.

[67] Z. Lu and T. Leen. Semi-supervised learning with penalized probabilistic clustering. *Adv Neur In*, 17:849–856, 2005.

[68] M. Łuksza, M. Lässig, and J. Berg. Significance analysis and statistical mechanics: an application to clustering. *Physical review letters*, 105(22):220601, Nov 2010.

[69] S. B.-D. M. Ackerman and D. Loker. Towards property-based classification of clustering paradigms. In *Neural Information Processing Systems (NIPS). Vancouver, Canada*, 2010.

[70] J. MacQueen. Some methods for classification and analysis of multivariate observations. *Proc 5th Berkeley Symp Math Stat Probab*, 1:281–197, 1967.

[71] S. C. Madeira and A. L. Oliveira. Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Trans Comput Biol Bioinform*, 1(1):24–45, Jan 2004.

[72] P. Mahalanobis. On the generalized distance in statistics. *P Natl Acad Sci Calcutta*, 12:49, 1936.

[73] M. Margulies, M. Egholm, W. E. Altman, S. Attiya, J. S. Bader, L. A. Bemben, J. Berka, M. S. Braverman, Y.-J. Chen, Z. Chen, S. B. Dewell, L. Du, J. M. Fierro, X. V. Gomes, B. C. Godwin, W. He, S. Helgesen, C. H. Ho, G. P. Irzyk, S. C. Jando, M. L. I. Alenquer, T. P. Jarvie, K. B. Jirage, J.-B. Kim, J. R. Knight, J. R. Lanza, J. H. Leamon, S. M. Lefkowitz, M. Lei, J. Li, K. L. Lohman, H. Lu, V. B. Makhijani, K. E. McDade, M. P. McKenna, E. W. Myers, E. Nickerson, J. R. Nobile, R. Plant, B. P. Puc, M. T. Ronan, G. T. Roth, G. J. Sarkis, J. F. Simons, J. W. Simpson, M. Srinivasan, K. R. Tartaro, A. Tomasz, K. A. Vogt, G. A. Volkmer, S. H. Wang, Y. Wang, M. P. Weiner, P. Yu, and R. F. B. J. M. Rothberg. Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 437:376–380, Jul 2005.

[74] J. Marioni, C. Mason, S. M. Mane, M. Stephens, and Y. Gilad. RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res*, 18:1509–1517, June 2008.

[75] G. J. McLachlan and K. E. Basford. *Mixture Models: Inference and Applications to Clustering*. Marcel Dekker, New York, 1988.

[76] G. J. McLachlan and T. Krishnan. *The EM Algorithm and Extensions*. Wiley, New York, 1997.

[77] M. Mézard and A. Montanari. *Information, Physics and Computation*. Oxford University Press, USA, February 2009.

[78] M. Mézard and G. Parisi. The cavity method at zero temperature. *J Stat Phys*, 111(1-2):1–34, 2003.

[79] M. Mézard, G. Parisi, and M. A. Virasoro. *Spin Glass Theory and Beyond*. World Scientific, Singapore, 1987.

[80] L. Nadel and D. L. Stein. *1992 Lectures In Complex Systems (Santa Fe Institute Studies in the Sciences of Complexity Lecture Notes)*. Westview Press, Boulder, Colorado, 1994.

[81] R. R. Nayak, M. Kearns, R. S. Spielman, and V. G. Cheung. Coexpression network based on natural variation in human gene expression reveals gene interactions and functions. *Genome Res*, 19(11):1953–62, Nov 2009.

[82] T. Nilsson, M. Mann, R. Aebersold, J. R. Yates, A. Bairoch, and J. J. M. Bergeron. Mass spectrometry in high-throughput proteomics: ready for the big time. *Nat Methods*, 7(9):681–685, Sep 2010.

[83] G. Parisi. Infinite number of order parameters for spin-glasses. *Phys Rev Lett*, 43(1754), December 1979.

[84] J. Quackenbush. Microarray data normalization and transformation. *Nat Genet*, 32 Suppl:496–501, Dec 2002.

[85] S. Ramaswamy, P. Tamayo, R. Rifkin, S. Mukherjee, C.-H. Yeang, M. Angelo, C. Ladd, M. Reich, E. Latulippe, J. P. Mesirov, T. Poggio, W. Gerald, M. Loda, E. S. Lander, and T. R. Golub. Multiclass cancer diagnosis using tumor gene expression signatures. *P Natl Acad Sci USA*, 98(26):15149–15154, December 2001.

[86] V. Roth, T. Lange, M. Braun, and J. Buhmann. A resampling approach to cluster validation. In *COMPSTAT: Proceedings in Computational Statistics: 15th Symposium Held in Berlin, Germany, 2002*, page 123, 2002.

[87] M. Schena, D. Shalon, R. Heller, A. Chai, P. O. Brown, and R. W. Davis. Parallel human genome analysis: microarray-based expression monitoring of 1000 genes. *P Natl Acad Sci USA*, 93(20):10614–9, Oct 1996.

[88] M. H. Schulz, D. R. Zerbino, M. Vingron, and E. Birney. Oases: modelling alternative isoforms to improve de novo transcriptome assembly. In submission, 2011.

[89] G. Sella, D. A. Petrov, M. Przeworski, and P. Andolfatto. Pervasive natural selection in the drosophila genome? *PLoS Genet*, 5(6):e1000495, Jun 2009.

[90] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, Jul 1948.

[91] D. K. Slonim and I. Yanai. Getting started in gene expression microarray analysis. *PLoS Comp Biol*, 5(10):e1000543, Oct 2009.

[92] N. Slonim, G. S. S. Atwal, G. Tkačik, and W. Bialek. Information-based clustering. *P Natl Acad Sci USA*, 102(51):18297–18302, December 2005.

[93] L. K. Southworth, A. B. Owen, and S. K. Kim. Aging mice show a decreasing correlation of gene expression within genetic modules. *PLoS Genet*, 5(12):e1000776, Dec 2009.

[94] M. C. P. D. Souto, I. G. Costa, D. S. A. D. Araujo, T. B. Ludermir, and A. Schliep. Clustering cancer gene expression data: a comparative study. *BMC Bioinformatics*, 9(1):497, Jan 2008.

[95] S. Still and W. Bialek. How many clusters? An information-theoretic perspective. *Neural Comput*, 16(12):2483–506, Dec 2004.

[96] B. E. Stranger, M. S. Forrest, A. G. Clark, M. J. Minichiello, S. Deutsch, R. Lyle, S. Hunt, B. Kahl, S. E. Antonarakis, S. Tavaré, P. Deloukas, and E. T. Dermitzakis. Genome-wide associations of gene expression variation in humans. *PLoS Genet*, 1(6):e78, Jan 2005.

[97] A. I. Su, J. B. Welsh, L. M. Sapinoso, S. G. Kern, P. Dimitrov, H. Lapp, P. G. Schultz, S. M. Powell, C. A. Moskaluk, H. F. Frierson, and G. M. Hampton. Molecular Classification of Human Carcinomas by Use of Gene Expression Signatures. *Cancer Res*, 61(20):7388–7393, October 2001.

[98] A. I. Su, T. Wiltshire, H. Lapp, K. A. Ching, D. Block, J. Zhang, R. Soden, M. Hayakawa, G. Kreiman, M. P. Cooke, J. R. Walker, and J. B. Hogenesch. A gene atlas of the human and mouse protein-encoding transcriptomes. *P Natl Acad Sci USA*, 101(41):6062–6067, 2004.

[99] R. Suzuki and H. Shimodaira. pvclust: an R package for assessing the uncertainty in hierarchical clustering. *Bioinformatics*, 22(12):1540–2, Jun 2006.

[100] R. Suzuki and H. Shimodaira. *pvclust: Hierarchical Clustering with P-Values via Multiscale Bootstrap Resampling*, 2009. R package version 1.2-1.

[101] M. Talagrand. On the gaussian perceptron at high temperature. *Math Phys Anal Geom*, 5(1):77–99, 2002.

[102] M. Talagrand. *Spin glasses: A Challenge for Mathematicians : cavity and mean field models*. Springer, 1 edition, August 2003.

[103] U. von Luxburg, R. C. Williamson, and I. Guyon. Clustering: Science or Art? In *NIPS 2009 Workshop on Clustering Theory, Vancouver, Canada*, December 2009.

[104] Z. Wang, M. Gerstein, and M. Snyder. RNA-seq: a revolutionary tool for transcriptomics. *Nat Rev Genet*, 10(1):57–63, Jan 2009.

[105] J. H. Ward. Hierarchical grouping to optimize an objective function. *J Am Stat Assoc*, 58(301):236–244, March 1963.

[106] M. West, C. Blanchette, H. Dressman, E. Huang, S. Ishida, R. Spang, H. Zuzan, J. A. Olson, J. R. Marks, and J. R. Nevins. Predicting the clinical status of human breast cancer by using gene expression profiles. *P Natl Acad Sci USA*, 98(20):11462–11467, September 2001.

[107] C. J. Wu. On the convergence properties of the em algorithm. *Ann Stat*, 11(1):95–103, Mar 1983.

[108] K. Yeung, C. Fraley, A. Murua, A. Raftery, and W. Ruzzo. Model-based clustering and data transformations for gene expression data. *Bioinformatics*, 17(10):977, 2001.

[109] Y.-K. Yu and T. Hwa. Statistical significance of probabilistic sequence alignment and related local hidden markov models. *J Comput Biol*, 8(3):249–282, 2001.

[110] J. M. Zahn, S. Poosala, A. B. Owen, D. K. Ingram, A. Lustig, A. Carter, A. T. Weeraratna, D. D. Taub, M. Gorospe, K. Mazan-Mamczarz, E. G. Lakatta, K. R. Boheler, X. Xu, M. P. Mattson, G. Falco, M. S. H. Ko, D. Schlessinger, J. Firman, S. K. Kummerfeld, W. H. Wood, A. B. Zonderman, S. K. Kim, and K. G. Becker. AGEMAP: a gene expression database for aging in mice. *PLoS Genet*, 3(11):2326–2337, Nov 2007.

[111] X. Zhu and A. B. Goldberg. Introduction to semi-supervised learning. *Synthesis Lect Artif Intell Mach Learn*, 3(1):1–130, Jan 2009.

Appendix A: Normalisation constants in the spherical model

Normalisation constant of the background model. The null distribution for a vector $\hat{\mathbf{x}}$ uniformly distributed on a sphere of radius $\sqrt{M}$ is

$$P_0(\hat{\mathbf{x}}) = \frac{1}{Z_0}\delta(\hat{\mathbf{x}}\cdot\hat{\mathbf{x}} - M) \ . \tag{.5}$$

The normalisation constant Z_0, giving the surface of M-dimensional sphere of radius $\sqrt{M}$, can be computed as

$$\begin{aligned}
Z_0 &= \int_{\mathbf{R}^M} \delta(\hat{\mathbf{x}}\cdot\hat{\mathbf{x}} - M)d\hat{\mathbf{x}} && (.6)\\
&= \int \frac{1}{4\pi i}\left[\int_{\mathbf{R}^M} \exp\left\{\frac{M\gamma}{2} - \frac{\gamma\sum_\mu(\hat{x}^\mu)^2}{2}\right\} d\hat{x}^1\ldots d\hat{x}^M\right] d\gamma && (.7)\\
&= \int \frac{1}{4\pi i}\exp\left\{M\left(\frac{\gamma}{2} - \frac{1}{2}\log\gamma + \frac{1}{2}\log(2\pi)\right)\right\} d\gamma && (.8)\\
&= \int \frac{1}{4\pi i} e^{Mf(\gamma)} d\gamma \ , && (.9)
\end{aligned}$$

with

$$f(\gamma) = \frac{\gamma}{2} - \frac{1}{2}\log\gamma + \frac{1}{2}\log(2\pi) \ . \tag{.10}$$

Eq. (.9) can be computed with the saddle point approximation. Taking the derivative of (.10), we get the saddle point at $\gamma^* = 1$. The normalisation constant of the background distribution is asymptotically, for large M,

$$Z_0 = \exp\left\{M\left(\frac{1}{2} + \frac{1}{2}\log(2\pi)\right)\right\} \ . \tag{.11}$$

Normalisation constant of the cluster model. The cluster-distribution for a length-constrained vector $\hat{\mathbf{x}}$ is

$$Q(\hat{\mathbf{x}}|\hat{\mathbf{z}},\eta) = \frac{1}{Z_\eta}P_0(\hat{\mathbf{x}})e^{\eta\hat{\mathbf{x}}\cdot\hat{\mathbf{z}}} \ , \tag{.12}$$

where $\|\hat{\mathbf{z}}\| = \sqrt{M}$. Here we compute the normalisation constant Z_η. To simplify the computation we set the cluster direction to $\hat{\mathbf{z}} = [1, 1, \ldots, 1]$, without loss of generality.

The calculation proceeds as:

$$
\begin{aligned}
Z_\eta &= \int_{\mathbb{R}^M} \frac{1}{Z_0}\delta(\hat{\mathbf{x}}\cdot\hat{\mathbf{x}} - M)e^{\eta\hat{\mathbf{x}}\cdot\hat{\mathbf{z}}}d\hat{\mathbf{x}} && (.13)\\
&= \frac{1}{Z_0}\int \frac{d\gamma}{4\pi i}\int_{\mathbb{R}^M} \exp\left\{\frac{M\gamma}{2} - \frac{\gamma\sum_\mu(\hat{x}^\mu)^2}{2} + \eta\sum_\mu \hat{x}^\mu\right\} d\hat{x}^1\ldots d\hat{x}^M && (.14)\\
&= \frac{1}{Z_0}\int \frac{d\gamma}{4\pi i}\exp\left\{M\left(\frac{\gamma}{2} + \frac{\eta^2}{2\gamma} - \frac{1}{2}\log\gamma + \frac{1}{2}\log(2\pi)\right)\right\}\ . && (.15)
\end{aligned}
$$

As before, assuming the number of dimensions M is large, we can compute (.15) with a saddle point approximation:

$$Z_\eta = \frac{1}{Z_0}\int \frac{d\gamma}{4\pi i}e^{Mf(\gamma,\eta)} \tag{.16}$$

with

$$f(\gamma,\eta) = \frac{\gamma}{2} + \frac{\eta^2}{2\gamma} - \frac{1}{2}\log\gamma + \frac{1}{2}\log(2\pi)\ . \tag{.17}$$

Setting the derivative of (.17) to zero, we get a saddle point solution,

$$\gamma^* = \frac{1+\sqrt{1+4\eta^2}}{2}\ . \tag{.18}$$

The normalisation constant of the cluster model is, asymptotically for large M, given by

$$Z_\eta = \exp\left\{M\left(\frac{\gamma^*-1}{2} + \frac{\eta^2}{2\gamma^*} - \frac{1}{2}\log\gamma^*\right)\right\}\ . \tag{.19}$$

Appendix B: Details of the EM algorithm

From the Bayes rule, after log-transformation, we have

$$\mathcal{L}(\Theta) = \log P(\mathbf{X}|\Theta) = \log P(\mathbf{X}, \mathbf{Y}|\Theta) - \log P(\mathbf{Y}|\mathbf{X}, \Theta) \ . \tag{.20}$$

Taking the conditional expectation of $\mathbf{Y}$ with respect to $\mathbf{X}$ and Θ^n, we obtain

$$\begin{aligned}
\mathcal{L}(\Theta) - \mathcal{L}(\Theta^n) &= \int_{\mathcal{Y}} \log P(\mathbf{X}, \mathbf{Y}|\Theta) P(\mathbf{Y}|\mathbf{X}, \Theta^n) d\mathbf{Y} \\
&- \int_{\mathcal{Y}} \log P(\mathbf{Y}|\mathbf{X}, \Theta) P(\mathbf{Y}|\mathbf{X}, \Theta^n) d\mathbf{Y} \\
&= R(\Theta, \Theta^n) - R(\Theta^n, \Theta^n) \qquad (.21) \\
&+ \int_{\mathcal{Y}} P(\mathbf{Y}|\mathbf{X}, \Theta^n) \log \frac{P(\mathbf{Y}|\mathbf{X}, \Theta^n)}{P(\mathbf{Y}|\mathbf{X}, \Theta)} d\mathbf{Y} \ . \qquad (.22)
\end{aligned}$$

The last term (.22) in the sum is the Kullback-Leibler divergence between distributions $P(\mathbf{Y}|\mathbf{X}, \Theta)$ and $P(\mathbf{Y}|\mathbf{X}, \Theta^n)$, and as such it is always nonnegative. Substituting Θ by Θ^{n+1}, (which by definition (5.25) maximizes $R(\Theta, \Theta^n)$), we get

$$\mathcal{L}(\Theta^{n+1}) - \mathcal{L}(\Theta^n) \geq R(\Theta^{n+1}, \Theta^n) - R(\Theta^n, \Theta^n) \geq 0 \ ,$$

and what follows

$$\mathcal{L}(\Theta^{n+1}) \geq \mathcal{L}(\Theta^n) \ . \tag{.23}$$

Appendix C: Replica calculation for the maximal cluster score distribution

Cluster score based on directional density

We compute the distribution of the maximal cluster score, where the individual vector contributions are

$$s(\mathbf{x}|\mathbf{z},\mu) = \mathbf{x}\cdot\mathbf{z} - \mu \tag{.24}$$

and the cluster centre $\mathbf{z}$ is constrained by $\mathbf{z}\cdot\mathbf{z} = 1$. We assume the Gaussian background model,

$$P_0(\mathbf{x}) = (2\pi)^{-\frac{M}{2}} e^{-\mathbf{x}\cdot\mathbf{x}/2} \ . \tag{.25}$$

Computing the score of the maximal scoring cluster involves optimising over all possible directions $\mathbf{z}$ of a cluster

$$S_{\max}(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mu) = \max_{\mathbf{z}\in\mathbb{R}^M,||\mathbf{z}||=1} S(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mathbf{z},\mu) \ . \tag{.26}$$

We use an integral representation of the maximum over $\mathbf{z}$,

$$e^{\beta S_{\max}(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mu)} = \lim_{\beta'\to\infty}\left[\int \delta(1-||\mathbf{z}||)e^{\beta' S(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mathbf{z},\mu)}d\mathbf{z}\right]^{\frac{\beta}{\beta'}} . \tag{.27}$$

The delta function constraints vector $\mathbf{z}$ to the sphere of radius 1. For large β', the integral is dominated by contributions of high scoring clusters, for β' converging to infinity the maximum cluster score is retrieved.

The generating function for the maximum score now reads

$$\begin{aligned} Z(\beta) \equiv e^{-N\beta f(\beta,\mu)} &= \int\left(\prod_{i=1}^N P_0(\mathbf{x}_i)\right) e^{\beta S_{\max}(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mu)} d\mathbf{x}_1\ldots d\mathbf{x}_N \\ &= \lim_{\beta'\to\infty}\int\left(\prod_{i=1}^N P_0(\mathbf{x}_i)\right)\left[\int d\mathbf{z}\ \delta(1-||\mathbf{z}||)e^{\beta' S(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mathbf{z},\mu)}\right]^{\frac{\beta}{\beta'}} d\mathbf{x}_1\ldots d\mathbf{x}_N \\ &= \lim_{\beta'\to\infty}\int\left(\prod_{i=1}^N P_0(\mathbf{x}_i)\right)\bar{Z}(\beta')^{\frac{\beta}{\beta'}} d\mathbf{x}_1\ldots d\mathbf{x}_N \ , \end{aligned} \tag{.28}$$

with $\bar{Z}(\beta') = \int \delta(1 - ||\mathbf{z}||) e^{\beta' S(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mathbf{z},\mu)} d\mathbf{z}$. Denoting $\beta/\beta' = n$, we can rewrite (.28)

$$Z(\beta) = \lim_{n\to 0} \langle\langle \bar{Z}(\beta' = \beta/n)^n \rangle\rangle \ , \tag{.29}$$

where $\langle\langle(.)\rangle\rangle = \int \left(\prod_{i=1}^N P_0(\mathbf{x}_i)\right) d\mathbf{x}_1 \ldots d\mathbf{x}_N$ is a shorthand for the integral over $\mathbf{x}_1, \ldots, \mathbf{x}_N$.

In the terminology of the statistical mechanics of disordered systems the average over $\mathbf{x}_i$ is called the *quenched disorder average*. As discussed in Chapter 2, the label "quenched disorder" refers to a clear distinction between the variables $\mathbf{x}_i$ describing data vectors, and the variables $\mathbf{z}$ describing the cluster centre. Both sets of variables are averaged over, but $\mathbf{z}$ is integrated over to locate the optimal cluster direction *while keeping the data vectors* $\mathbf{x}_i$ *fixed*. Then, the data vectors $\mathbf{x}_i$ are integrated over to determine the properties of the optimal cluster in different configurations of the data vectors $\mathbf{x}_i$.

Much of the remaining calculation described below formally follows the calculation of the storage capacity of neural network models with a penalty on storage errors, see [41]. Rigorous results on the capacity of neural networks have been obtained by Talagrand [101] and can be expected to carry over to the present problem.

Here, we use the so-called *replica method* [79] to evaluate (.29): $\langle\langle \bar{Z}(\beta')^n \rangle\rangle$ is first computed for integer n and then the limit $n \to 0$ is taken by analytic continuation. The expression $\bar{Z}(\beta')^n$ can be viewed as a system with n *replicated* $\mathbf{z}$ variables,

$$\begin{aligned} \bar{Z}(\beta')^n &= \prod_{a=1}^n \bar{Z}(\beta') = \prod_{a=1}^n \int d\mathbf{z}_a \delta\left(1 - ||\mathbf{z}_a||\right) e^{\beta' \sum_a S(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mathbf{z}_a,\mu)} && (.30) \\ &= \prod_{a=1}^n \int \frac{dE_a}{2\pi} \int d\mathbf{z}_a e^{i\sum_a E_a - i\sum_a E_a ||\mathbf{z}_a|| + \beta' \sum_a S(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mathbf{z}_a,\mu)} \ . && (.31) \end{aligned}$$

The new set of integrals over E_a in (.31) comes from the integral representation of the delta function in (.30).

Scoring function s involves a maximum function and hence it depends non-linearly on scalar products $\mathbf{x}_i \cdot \mathbf{z}$. The first step is to linearize this term using delta-functions and their integral representation

$$\begin{aligned} e^{\beta' s(\mathbf{x}_i|\mathbf{z}_a,\mu)} &= e^{\beta' \max[\mathbf{x}_i \cdot \mathbf{z}_a - \mu, 0]} && (.32) \\ &= \int \delta\left(\lambda_{ia} - \mathbf{x}_i \cdot \mathbf{z}_a\right) e^{\beta' \max[\lambda_{ia} - \mu, 0]} \, d\lambda_{ia} && (.33) \\ &= \int \left[\int \frac{1}{2\pi} e^{i\lambda_{ia} y_{ia} - i y_{ia} \sum_\nu x_i^\nu z_a^\nu} e^{\beta' \max[\lambda_{ia} - \mu, 0]} dy_{ia} \right] d\lambda_{ia} \ . && (.34) \end{aligned}$$

Now $\mathbf{x}_i$ appears linearly in the exponent, at the cost of two new sets of integrals, $\{\lambda_i\}$ and $\{y_i\}$. Advantage of this expression is that the disorder average over x_i^ν can now

be performed easily. Considering only the term in which x_i^ν appears, the disorder average yields

$$\begin{aligned}\langle\langle e^{-i\sum_{i,a} y_{ia}\mathbf{x}_i\cdot\mathbf{z}_a}\rangle\rangle &= \int e^{-i\sum_{i,a} y_{ia} z_a^\nu x_i^\nu} \prod_{i=1}^{N} P_0(\mathbf{x}_i) d\mathbf{x}_1 \dots d\mathbf{x}_N \\ &= \prod_{\nu=1}^{M} \exp\left\{-\frac{1}{2}\left(\sum_{i,a} y_{ia} z_a^\nu\right)^2\right\} \\ &= \exp\left\{-\frac{1}{2}\sum_{a,b}\mathbf{z}_a\cdot\mathbf{z}_b\sum_i y_{ia}y_{ib}\right\}. \qquad (.35)\end{aligned}$$

We can disentangle the exponent by introducing the so-called *order parameter*

$$q_{ab} = \mathbf{z}_a\cdot\mathbf{z}_b\ ,$$

a symmetric $n\times n$ matrix with ones along the diagonal. For this reason it is sufficient to introduce q_{ij} via integrals over delta factors for $a>b$; for $a<b$ $q_{ab}=q_{ba}$, for $a=b$ $q_{aa}=1$. Equation (.35) now reads

$$\begin{aligned}&\int\prod_{a>b}\delta\left(q_{ab}-\mathbf{z}_a\cdot\mathbf{z}_b\right)\exp\left\{-\frac{1}{2}\sum_{a,b}q_{ab}\sum_i y_{ia}y_{ib}\right\}\prod_{a>b}\mathrm{d}q_{ab} \qquad (.36)\\ =\ &\int\left[\int\frac{1}{2\pi}\exp\left\{-i\sum_{a>b}q_{ab}\hat{q}_{ab}+i\sum_{a>b}\hat{q}_{ab}\mathbf{z}_a\dot{\mathbf{z}}_b-\frac{1}{2}\sum_{a,b}q_{ab}\sum_i y_{ia}y_{ib}\right\}\prod_{a>b}\mathrm{d}\hat{q}_{ab}\right]\prod_{a>b}\mathrm{d}q_{ab}\ .\end{aligned}$$

The advantage with respect to (.35) is that now the sums over ν and i appear separately in the exponent so the integrals over $\{y_i\}$ and $\{\mathbf{z}_a\}$ can be evaluated independently. To this end we take a "replica-symmetric" ansatz,

$$\begin{aligned}q_{ab} &= q, \quad a>b \\ i\hat{q}_{ab} &= \hat{q}, \quad a>b, \qquad (.37)\end{aligned}$$

i.e. we assume that angles between any two vectors $\mathbf{z}_a$ and $\mathbf{z}_b$, $a,b\in\{1,\dots,n\}$ are the same.

We first collect the terms involving z_a^ν:

$$\begin{aligned}&\int\prod_{a=1}^{n}\delta\left(\mathbf{z}_a\cdot\mathbf{z}_a-M\right)e^{\mathrm{i}\sum_{a>b}\hat{q}_{ab}\mathbf{z}_a\cdot\mathbf{z}_b}\mathrm{d}\mathbf{z}_1\dots\mathrm{d}\mathbf{z}_n\\ =\ &\int\frac{1}{2\pi/M}\left[\int e^{\mathrm{i}M\sum_a E_a-\mathrm{i}\sum_a E_a\mathbf{z}_a\cdot\mathbf{z}_a+\mathrm{i}\sum_{a>b}\hat{q}_{ab}\mathbf{z}_a\cdot\mathbf{z}_b}\mathrm{d}\mathbf{z}_1\dots\mathbf{z}_n\right]\mathrm{d}E_1\dots\mathrm{d}E_n\\ &\text{which factorizes in components of }\mathbf{z}\\ =\ &\int\frac{e^{\mathrm{i}M\sum_a E_a}}{2\pi/M}\left[\int\exp\left\{-\mathrm{i}\sum_a E_a z_a^2+\mathrm{i}\sum_{a>b}\hat{q}_{ab}z_a z_b\right\}\mathrm{d}z_1\dots\mathrm{d}z_n\right]^M\mathrm{d}E_1\dots\mathrm{d}E_n\ . \qquad (.38)\end{aligned}$$

We now set all the parameters enforcing the spherical constraint on vectors $\mathbf{z}_a$ from (.30) to the same value,

$$iE_a = \frac{1}{2}E \ . \tag{.39}$$

Together with the RS-Ansatz (.37), the square bracket in (.38) becomes

$$\begin{aligned}
&\int \exp\left\{-\frac{1}{2}E\sum_a z_a^2 + \frac{1}{2}\hat{q}\sum_{a,b} z_a z_b - \frac{1}{2}\hat{q}\sum_a z_a^2\right\} \mathrm{d}z_1 \dots \mathrm{d}z_n \\
= &\int \left[\int \exp\left\{-\frac{1}{2}(E+\hat{q})\sum_a z_a^2 + \sqrt{\hat{q}}t\sum_a z_a\right\} \mathrm{d}z_1 \dots \mathrm{d}z_n\right] \sqrt{1/(2\pi)} e^{-\frac{t^2}{2}} dt \qquad (.40) \\
&\text{(where } Dt \text{ is a shorthand for } \sqrt{1/(2\pi)}e^{-\frac{t^2}{2}}dt\text{, and we used } \hat{q}\textstyle\sum_{a,b} z_a z_b = \left(\sqrt{\hat{q}}\sum_a z_a\right)^2 \\
&\text{and } \textstyle\int e^{tz} Dt = e^{\frac{1}{2}z^2}\text{)} \\
= &\int \left[\frac{1}{\sqrt{E+\hat{q}}} \exp\left\{\frac{\hat{q}t^2}{2(E+\hat{q})}\right\}\right]^n Dt \qquad (.41) \\
= &(E+\hat{q})^{-\frac{n}{2}} \int \exp\left\{\frac{n\hat{q}t^2}{2(E+\hat{q})}\right\} Dt \qquad (.42) \\
&\text{(using Gaussian integrals and the fact that the integrals over } z_a \text{ in (.41) factorize)} \\
= &(E+\hat{q})^{-\frac{n}{2}} \int \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\left(1 - \frac{n\hat{q}}{(E+\hat{q})}\right)t^2\right\} = \left(1 - \frac{n\hat{q}}{E+\hat{q}}\right)^{-\frac{1}{2}} (E+\hat{q})^{-\frac{n}{2}} Dt \qquad (.43)
\end{aligned}$$

Now we do the same for the terms containing λ and y:

$$\begin{aligned}
&\prod_{i,a} \int \mathrm{d}\lambda_{ia} \int \frac{\mathrm{d}y_{ia}}{2\pi} \exp\left\{\mathrm{i}\sum_{i,a} \lambda_{ia} y_{ia} - \frac{1}{2}\sum_{a,b} q_{ab} \sum_i y_{ia} y_{ib} + \beta' \sum_{i,a} \Xi(\lambda_{ia} - \tilde{\mu})\right\} \\
= &\left[\int \mathrm{d}\lambda_a \int \frac{\mathrm{d}y_a}{2\pi} \exp\left\{\mathrm{i}\sum_a \lambda_a y_a - \frac{1}{2}\sum_{a,b} q_{ab} y_a y_b + \beta' \sum_a \Xi(\lambda_a - \tilde{\mu})\right\}\right]^N \qquad (.44)
\end{aligned}$$

The integrals over $\{y_a\}$ in Eq. (.44) read

$$\begin{aligned}
&\int \frac{\mathrm{d}y_a}{2\pi} \exp\left\{\mathrm{i}\sum_a \lambda_a y_a - \frac{1}{2}\sum_a (1-q) y_a^2 - \frac{1}{2} q (\sum_a y_a)^2\right\} \\
= &\int Dt \exp\left\{-\frac{1}{2}(1-q)\sum_a y_a^2 + \mathrm{i}\sqrt{q}t\sum_a y_a + \mathrm{i}\sum_a \lambda_a y_a\right\} \qquad (.45)
\end{aligned}$$

so the square bracket in (.44) again factorizes in a, giving

$$\begin{aligned}
&\int Dt \left[\int \mathrm{d}\lambda \int \frac{\mathrm{d}y}{2\pi} \exp\left\{-\frac{1}{2}(1-q)y^2 + \mathrm{i}\sqrt{q}ty + \mathrm{i}\lambda y + \beta' \Xi(\lambda - \tilde{\mu})\right\}\right]^n \\
= &\int Dt \left[\int \mathrm{d}\lambda \sqrt{\frac{1}{2\pi(1-q)}} \exp\left\{-\frac{(\sqrt{q}t + \lambda)^2}{2(1-q)} + \beta' \Xi(\lambda - \tilde{\mu})\right\}\right]^n . \qquad (.46)
\end{aligned}$$

Collecting all terms we have

$$\langle\langle \bar{Z}(\beta)^n \rangle\rangle = \int e^{Ng(q,\hat{q},E)} dq\, d\hat{q}\, dE \tag{.47}$$

where

$$g(q,\hat{q},E) = \frac{M}{N}\left[n\frac{E}{2} - \frac{n(n-1)}{2}q\hat{q} + g_s(\hat{q},E)\right] + g_E(q) \tag{.48}$$

$$g_s(\hat{q},E) = -\frac{1}{2}\log\left(1 - \frac{n\hat{q}}{E+\hat{q}}\right) - \frac{n}{2}\log(E+\hat{q}) \tag{.49}$$

$$g_E(q) = \log\int Dt\left[\int \mathrm{d}\lambda \sqrt{\frac{1}{2\pi(1-q)}}\exp\left\{-\frac{(\sqrt{q}t+\lambda)^2}{2(1-q)} + \beta'\max\left[\lambda-\mu,0\right]\right\}\right]^n \tag{.50}$$

The integrals over $\hat{q}$ and E are performed with the saddle point approximation. Function $g(q,\hat{q},E)$ is related to the free energy function by

$$\lim_{\beta'\to\infty} g(q^*,\hat{q}^*,E^*) = -\beta f(\beta,\mu)\ , \tag{.51}$$

where q^*, $\hat{q}^*$ and E^* give the saddle point of g.

We find that the following dependencies are met at the saddle point:

$$\hat{q}^* = \frac{q^*}{(1-q^*)(1+q^*(n-1))}\ , \tag{.52}$$

$$E^* = \frac{1+q^*(n-2)}{(1-q^*)(1+q^*(n-1)q^*)}\ . \tag{.53}$$

Thus, the free energy function can be expressed as a function of q only,

$$g(q) = \frac{M}{2N}\left[n - n\log\left(\frac{1}{1-q}\right) - \log\left(\frac{1-q}{1+q(n-1)}\right)\right] + g_E(q)\ . \tag{.54}$$

The saddle point equation reads

$$\frac{\partial g}{\partial q}(q) = \frac{qn(n-1)}{2(q-1)(1+q(n-1))} + \alpha\frac{\partial g_E}{\partial q}(q)\ . \tag{.55}$$

Derivative $\frac{\partial g_E}{\partial q}(q)$ has to be computed numerically but its form can be simplified. Let us denote the integral over λ in Eq. (.49) by a shorthand $L(q)$,

$$\begin{aligned} L(q) &= \int \mathrm{d}\lambda \frac{1}{\sqrt{2\pi(1-q)}} e^{\left(-\frac{(\sqrt{q}t+\lambda)^2}{2(1-q)} + \beta'\Xi(\lambda-\tilde{\mu})\right)} \\ &= 1 - H(t_1) + e^{\frac{1}{2}t_2^2 - t_1 t_2} H\left(t_1 - t_2\right) \end{aligned} \tag{.56}$$

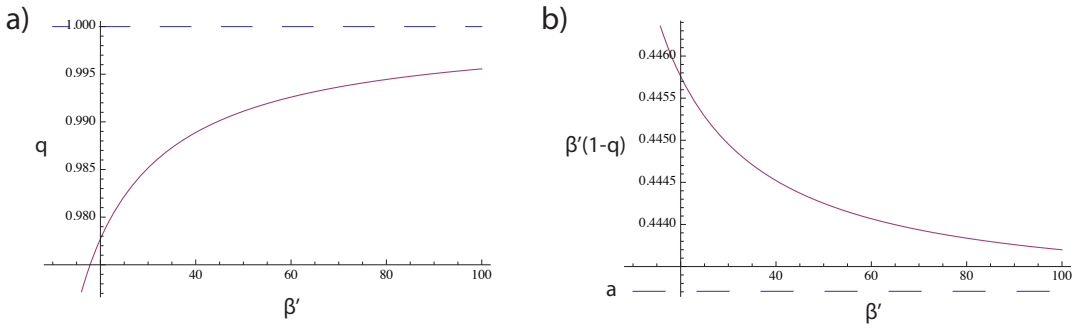

Figure .1: Asymptotic behavior of the order parameter. Example with $\mu = 1$, $M = 20$ and $N = 40$. The saddle point value of q from Eq. .55 shows the following asymptotic behavior as $\beta' \to \infty$: (a) q converges to 1 such that (b) $\beta'(1-q)$ converges to some value a, which defines a new "asymptotic" order parameter.

where $H(x) = \int_x^{+\infty} Dx$, and t_1 and t_2 are shorthand notations, $t_1 = \frac{t\sqrt{q}+\tilde{\mu}}{\sqrt{1-q}}$ and $t_2 = \beta'\sqrt{1-q}$. We will also use derivatives of t_1 and t_2 over q, with shorthands $t_1' = \frac{t+\tilde{\mu}\sqrt{q}}{2\sqrt{q(1-q)^3}}$ and $t_2' = \frac{\beta'}{2\sqrt{1-q}}$.

We have (see Eq. (.48))

$$g_E(q) = \log \int Dt \, [L(q)]^n \ . \tag{.57}$$

This gives us all the terms needed for the (numerical) computation of the saddle point equation in Eq. (.55).

Asymptotic solution and the p-value

In the limit $\beta' \to \infty$, we observe an asymptotic behaviour of the saddle point value of q^*,

$$1 - q^* \sim a/\beta', \quad a > 0 \ , \tag{.58}$$

which defines a new order parameter a, see Fig. .1.

The integral over λ in (.49) becomes a saddle-point integral in β'

$$\int \mathrm{d}\lambda \exp\left\{-\beta' \frac{(t+\lambda)^2}{2a} + \beta' \max[\lambda - \mu, 0]\right\} = \int \mathrm{d}\lambda e^{\beta' h(\lambda)} \tag{.59}$$

where

$$h(\lambda) = -\frac{(t+\lambda)^2}{2a} + \max[\lambda - \mu, 0] \ . \tag{.60}$$

The saddle point is the solution of

$$\frac{\partial h(\lambda)}{\partial \lambda} = -\frac{t+\lambda}{a} + \Theta(\lambda - \tilde{\mu}) = 0 \ , \tag{.61}$$

which, as a function of t, is given by

$$\lambda^*(t) = \begin{cases} -t & \text{if } t > \frac{a}{2} - \mu \\ a - t & \text{if } t \leq \frac{a}{2} - \mu. \end{cases} \tag{.62}$$

The double integral in Eq. (.49) now gives

$$\begin{aligned} \lim_{\beta' \to \infty} \int Dt[L(q)]^n &= \lim_{\beta' \to \infty} \int Dt[L(q)]^{\frac{\beta}{\beta'}} \\ &= \lim_{\beta' \to \infty} \int Dt \left[\sqrt{\frac{\beta'}{2\pi a}} e^{\beta' h(\lambda_0)} \right]^{\frac{\beta}{\beta'}} = \lim_{\beta' \to \infty} \left[\frac{\beta'}{2\pi a} \right]^{\frac{\beta}{2\beta'}} \int Dt e^{\beta h(\lambda_0)} \\ &= \int Dt e^{\beta h(\lambda_0)} \\ &= \int_{-\infty}^{\frac{a}{2}-\mu} Dt e^{\beta\left(\frac{a}{2}-t-\mu\right)} + \int_{\frac{a}{2}-\mu}^{+\infty} Dt \\ &= e^{\frac{\beta^2}{2}+\beta\left(\frac{a}{2}-\mu\right)} \left(1 - H\left(\beta + \frac{a}{2} - \mu\right)\right) + H\left(\frac{a}{2} - \mu\right) \end{aligned} \tag{.63}$$

and we have

$$g_E(a) = \log\left[\left(1 - H\left(\mu - \frac{a}{2}\right)\right) + e^{\frac{\beta^2}{2}-\beta\left(\mu-\frac{a}{2}\right)} H\left(\mu - \frac{a}{2} - \beta\right)\right] \tag{.64}$$

$$= -\beta f_c\left(\beta, \mu - \frac{a}{2}\right) , \tag{.65}$$

where f_c is defined in Eq. (4.8).

Function g with respect to a is

$$g(a) = \lim_{\beta' \to \infty} g(q) = \frac{M}{2N} \log\left(\frac{a+\beta}{a}\right) + g_E(a) , \tag{.66}$$

giving the free energy function

$$-\beta f(\beta, \mu, a) = \min_a \left[\frac{M}{2N}\left(\frac{a+\beta}{a}\right) - \beta f_c\left(\beta, \mu - \frac{a}{2}\right)\right] . \tag{.67}$$

The intensive entropy $\omega(s)$ is defined as the extremum with respect to both a and β, $\omega(s) = -\max_{a,\beta}[\beta s + \beta f(\beta, \mu, a)]$.

Cluster score based on positional bias

Under the null-model, vectors follow the Gaussian distribution

$$P(\mathbf{x}) = \sqrt{\frac{\det \mathbf{H}}{(2\pi)^M}} e^{-\frac{1}{2}\mathbf{x}\cdot\mathbf{x}} . \tag{.68}$$

Vectors in a cluster are biased in the direction of the central cluster direction $\mathbf{z}$, leading to the cluster model

$$Q_\eta(\mathbf{x}|\mathbf{z}) = P(\mathbf{x})e^{\mathbf{x}\cdot\mathbf{z}-\mathbf{z}\cdot\mathbf{z}/2} \ . \tag{.69}$$

The cluster log-likelihood score $s(\mathbf{x}|\mathbf{z},\eta) \equiv \log[Q(\mathbf{x}|\mathbf{z},\eta)/P_0(\mathbf{x})]$ takes the simple form

$$s(\mathbf{x}|\mathbf{z},\eta) = \mathbf{x}\cdot\mathbf{z} - \mathbf{z}\cdot\mathbf{z}/2 \ . \tag{.70}$$

This log-likelihood score assigns positive score values to vectors which are more likely to be in a cluster with centre $\mathbf{z}$ than in the background. Note the offset depends on the cluster centre $\mathbf{z}$.

Considering a system of N vectors, the negative of the additive score

$$-S(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mathbf{z}) = -\sum_{i=1}^{N} \max[s(\mathbf{x}_i|\mathbf{z})/\sqrt{M}, 0] \tag{.71}$$

can be considered the energy function of the system. The scalar product $\mathbf{x}\cdot\mathbf{z}$ scales with $\sqrt{M}$ and for convenience we normalise the log-likelihood score by this factor. Central aim is to compute the distribution of the score of the cluster centred on the direction $\mathbf{z}$ with maximal score.

The maximal scoring cluster involves optimising over all possible centres $\mathbf{z}$ of a cluster

$$S_{\max}(\mathbf{x}_1,\ldots,\mathbf{x}_N) = \max_{\mathbf{z}\in\mathbb{R}^M} S(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mathbf{z}) \ . \tag{.72}$$

Note the difference with respect to definition of the maximal score in the previous case in Eq. (.26): cluster centre $\mathbf{z}$ is no longer length constrained. This difference, together with a dependence of the offset function on $\mathbf{z}$, leads to some changes in the replica solution. As we show in the following calculation, the resulting solution has a very similar form to the previous one, but it involves and additional order parameter reflecting the typical cluster centre length.

Similarly as before (Eq. (.27)), we use an integral representation of the maximum over $\mathbf{z}$,

$$e^{\beta S_{\max}(\mathbf{x}_1,\ldots,\mathbf{x}_N)} = \lim_{\beta'\to\infty} \left[\int e^{\beta' S(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mathbf{z})} d\mathbf{z}\right]^{\frac{\beta}{\beta'}} \ . \tag{.73}$$

Again, the difference is in the lack of length constraint of $\mathbf{z}$, previously introduced by means of a delta-function.

The generating function for the maximum is

$$
\begin{aligned}
Z(\beta) \equiv e^{-N\beta f(\beta,\mu)} &= \int \prod_{i=1}^{N} P_0(\mathbf{x}_i) d\mathbf{x}_i e^{\beta S_{\max}(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mathbf{z})} \qquad (.74)\\
&= \lim_{\beta'\to\infty} \int \prod_{i=1}^{N} P_0(\mathbf{x}_i) d\mathbf{x}_i \left[\int d\mathbf{z}\ e^{\beta' S(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mathbf{z})}\right]^{\frac{\beta}{\beta'}}\\
&= \lim_{\beta'\to\infty} \int \prod_{i=1}^{N} P_0(\mathbf{x}_i) d\mathbf{x}_i \bar{Z}(\beta')^{\frac{\beta}{\beta'}}\ ,
\end{aligned}
$$

with $\bar{Z}(\beta') = \int d\mathbf{z}\ e^{\beta' S(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mathbf{z})}$. Denoting $\beta/\beta' = n$ we can rewrite (.74)

$$
Z(\beta) = \lim_{n\to 0} \langle\langle \bar{Z}(\beta' = \beta/n)^n \rangle\rangle\ . \qquad (.75)
$$

Using the replica method, as before, we obtain

$$
\bar{Z}(\beta')^n = \prod_{a=1}^{n} \bar{Z}(\beta') = \prod_{a=1}^{n} \int d\mathbf{z}_a e^{\beta' \sum_a S(\mathbf{x}_1,\ldots,\mathbf{x}_N|\mathbf{z}_a)}.
$$

We linearise the scoring term $s(\mathbf{x}|\mathbf{z})$ by using delta-functions and their integral representation,

$$
\begin{aligned}
e^{\beta' \max[s(\mathbf{x}_i|\mathbf{z}_a),0]} &= e^{\beta' \max\left[\frac{1}{\sqrt{M}}\mathbf{x}_i\cdot\mathbf{z}_a - \mu(\mathbf{z}),0\right]}\\
&= \int d\lambda_{ia} \delta\left(\lambda_{ia} - \frac{1}{\sqrt{M}}\mathbf{x}_i\cdot\mathbf{z}_a\right) e^{\beta' \max[\lambda_{ia}-\mu(\mathbf{z}),0]}\\
&= \int d\lambda_{ia} \int \frac{dy_{ia}}{2\pi} e^{i\lambda_{ia}y_{ia} - \frac{i}{\sqrt{M}} y_{ia} \sum_\nu x_i^\nu z_a^\nu} e^{\beta' \max[\lambda_{ia}-\mu(\mathbf{z}),0]}\ . \qquad (.76)
\end{aligned}
$$

Considering only the term in which x_i^ν appears, the disorder average yields

$$
\begin{aligned}
\langle\langle e^{-\frac{i}{\sqrt{M}}\sum_{i,a,\nu} y_{ia} x_i^\nu z_a^\nu} \rangle\rangle &= \int \left(\prod_i d\mathbf{x}_i P_0(\mathbf{x}_i)\right) e^{-\frac{i}{\sqrt{M}}\sum_{i,a} y_{ia} \sum_\nu z_a^\nu x_i^\nu} \qquad (.77)\\
&= \int \left(\prod_{i,\nu} dx_i^\nu\right) \frac{1}{\sqrt{(2\pi)^M}} e^{-\frac{1}{2}\sum_i \sum_\nu (x_i^\nu)^2 - \frac{i}{\sqrt{M}}\sum_{i,a} y_{ia} z_a^\nu x_i^\nu}\\
&= \prod_i e^{-\frac{1}{2M}\sum_{a,b} y_{ia} y_{ib} \sum_\nu z_a^\nu z_b^\nu}\ .
\end{aligned}
$$

We introduce an order parameter to disentangle sums in the exponent,

$$
q_{ab} = \frac{1}{M}\sum_\nu z_a^\nu z_b^\nu\ ,
$$

which is a symmetric $n \times n$ matrix. This time the diagonal terms $\mathbf{z}_a \cdot \mathbf{z}_a/M$ are no longer fixed to 1, as we have not introduced any length constraint on vectors $\mathbf{z}_a$. We

thus have $n(n+1)/2$ distinct values of q_{ab} for $a \geq b$, which we introduce via integrals over delta distributions. The disorder average now reads

$$\begin{aligned} & \prod_{a\geq b} dq_{ab} \prod_{a\geq b} \delta(q_{ab} - \frac{1}{M}\sum_{\nu} z_a^\nu z_b^\nu) e^{-\frac{1}{2}\sum_{a,b} q_{ab} \sum_i y_{ia} y_{ib}} \qquad (.78) \\ = & \prod_{a\geq b} \int dq_{ab} \int \frac{d\hat{q}_{ab}}{2\pi/M} \exp\left\{ -iM \sum_{a\geq b} q_{ab}\hat{q}_{ab} + i \sum_{a\geq b} \hat{q}_{ab} \sum_\nu z_a^\nu z_b^\nu - \frac{1}{2}\sum_{ab} q_{ab} \sum_i y_{ia} y_{ib} \right\} . \end{aligned}$$

As in the previous calculation, the sums over ν and i appear separately in the exponent, so the integrals over y and z can be evaluated independently. We take a "replica-symmetric" ansatz,

$$q_{ab} = \begin{cases} q_1 & a = b \\ q_0 & a \neq b \end{cases} , \qquad (.79)$$

$$i\hat{q}_{ab} = \begin{cases} \frac{1}{2}\hat{q}_1 & a = b \\ \hat{q}_0 & a \neq b \end{cases} , \qquad (.80)$$

i.e. this time, due to the lack of length constraint on $\mathbf{z}$, we have four order parameters instead of the two in the previous case. We first collect terms involving z_a.

$$\begin{aligned} & \prod_a \int d\mathbf{z}_a \exp\left\{ i \sum_{a\geq b} \hat{q}_{ab} \mathbf{z}_a \cdot \mathbf{z}_b \right\} \\ = & \prod_a \int d\mathbf{z}_a \exp\left\{ \frac{1}{2}\hat{q}_0 \sum_{a,b} \mathbf{z}_a \cdot \mathbf{z}_b - \frac{1}{2}\hat{q}_0 \sum_a (\mathbf{z}_a)^2 + \frac{1}{2}\hat{q}_1 \sum_a (\mathbf{z}_a)^2 \right\} \\ = & \int Dt \prod_a \int d\mathbf{z}_a \exp\left\{ -\frac{1}{2}(\hat{q}_0 - \hat{q}_1) \sum_a (\mathbf{z}_a)^2 + t\sqrt{\hat{q}_0} \sum_a (\mathbf{z}_a)^2 \right\} \\ = & \int Dt \left[\sqrt{\frac{2\pi}{\hat{q}_0 - \hat{q}_1}} e^{\frac{\hat{q}_0 t^2}{2(\hat{q}_0 - \hat{q}_1)}} \right]^n = \left(\frac{2\pi}{\hat{q}_0 - \hat{q}_1} \right)^{\frac{n}{2}} \int dt \frac{1}{2\pi} \exp\left\{ -\frac{1}{2}\left(1 - \frac{\hat{q}_0 n}{\hat{q}_0 - \hat{q}_1} \right) t^2 \right\} \\ = & \left(\frac{2\pi}{\hat{q}_0 - \hat{q}_1} \right)^{\frac{n}{2}} \left(1 - \frac{\hat{q}_0 n}{\hat{q}_0 - \hat{q}_1} \right)^{-\frac{1}{2}} . \qquad (.81) \end{aligned}$$

Now we do the same for the terms containing λ and y:

$$\begin{aligned} & \prod_{i,a} \int d\lambda_{ia} \int \frac{dy_{ia}}{2\pi} \exp\left\{ i \sum_{i,a} \lambda_{ia} y_{ia} - \frac{1}{2} \sum_{a,b} q_{ab} \sum_i y_{ia} y_{ib} + \beta' \sum_{i,a} \max\left[\lambda_{ia} - \mu(q_{aa}), 0 \right] \right\} \\ = & \left[\int d\lambda_a \int \frac{dy_a}{2\pi} \exp\left\{ i \sum_a \lambda_a y_a - \frac{1}{2} \sum_{a,b} q_{ab} y_a y_b + \beta' \sum_a \max\left[\lambda_a - \mu(q_{aa}), 0 \right] \right\} \right]^N . \qquad (.82) \end{aligned}$$

The integrals over $\{y_a\}$ in Eq. (.82) read

$$\begin{aligned} & \int \frac{dy_a}{2\pi} \exp\left\{ i\sum_a \lambda_a y_a - \frac{1}{2}\sum_a (q_1-q_0) y_a^2 - \frac{1}{2} q_0 \left(\sum_a y_a\right)^2 \right\} \\ = & \int Dt \exp\left\{ -\frac{1}{2}(q_1-q_0)\sum_a y_a^2 + i\sqrt{q_0}t\sum_a y_a + i\sum_a \lambda_a y_a \right\} , \end{aligned} \tag{.83}$$

so the square bracket in Eq. (.82) factorizes in a, giving

$$\begin{aligned} & \int Dt \left[\int d\lambda \int \frac{dy}{2\pi} \exp\left\{ -\frac{1}{2}(q_1-q_0)y^2 + i\sqrt{q}ty + i\lambda y + \beta' \max\left[\lambda - \mu(q_1), 0\right] \right\} \right]^n \\ = & \int Dt \left[\int d\lambda \sqrt{\frac{1}{2\pi(q_1-q_0)}} \exp\left\{ -\frac{(\sqrt{q_0}t+\lambda)^2}{2(q_1-q_0)} + \beta' \max\left[\lambda - \mu(q_1), 0\right] \right\} \right]^n . \end{aligned} \tag{.84}$$

Collecting all terms we have

$$\langle\langle Z^n \rangle\rangle = \int_{q_1,q_0,\hat{q}_1,\hat{q}_0} e^{N g(q_1,q_0,\hat{q}_1,\hat{q}_0)} , \tag{.85}$$

where

$$\begin{aligned} g(q_1,q_0,\hat{q}_1,\hat{q}_0) &= \frac{M}{N}\left[\frac{n}{2}\log 2\pi - \frac{1}{2} n q_1 \hat{q}_1 - \frac{n(n-1)}{2} q_0 \hat{q}_0 + g_s(\hat{q}_1,\hat{q}_0) \right] + g_E(q_1,q_0) \qquad (.86) \\ g_s(\hat{q}_1,\hat{q}_0) &= -\frac{1}{2}\log\left(1 - \frac{n\hat{q}_0}{\hat{q}_0-\hat{q}_1}\right) - \frac{n}{2}\log(\hat{q}_0-\hat{q}_1) \\ g_E(q_1,q_0) &= \log \int Dt \left[\int d\lambda \sqrt{\frac{1}{2\pi(q_1-q_0)}} \exp\left\{ -\frac{(\sqrt{q_0}t+\lambda)^2}{2(q_1-q_0)} + \beta' \max\left[\lambda - \mu(q_1), 0\right] \right\} \right]^n . \end{aligned}$$

Function $g(q_1,q_0,\hat{q}_1,\hat{q}_0)$ is related to the free energy function by

$$\lim_{\beta'\to\infty} g(q_1,q_0,\hat{q}_1,\hat{q}_0) = -\beta f(\beta) . \tag{.87}$$

The integrals over four order parameters, q_1, q_0, $\hat{q}_1$, $\hat{q}_0$, in the limit of large M, N are performed using the saddle point approximation.

We first notice that the following dependencies are met between the order parameters at the saddle point of g:

$$\hat{q}_1^* = \frac{q_0^*(n-2)+q_1^*}{(q_0^*-q_1^*)(q_0^*(n-1)+q_1^*)} , \tag{.88}$$

$$\hat{q}_0^* = -\frac{q_0^*}{(q_0^*-q_1^*)(q_0^*(n-1)+q_1^*)} . \tag{.89}$$

The free energy function can be expressed as a function of two order parameters, q_1 and q_0. Inserting those in Eq. (.86), we obtain

$$g(q_1,q_0) = \frac{M}{2N}\left[n + n\log 2\pi + n\log(q_1-q_0) - \log\left(\frac{q_1-q_0}{q_1+q_0(n-1)}\right)\right] + g_E(q_1,q_0)\ . \tag{.90}$$

We can now solve the saddle point equations,

$$\frac{\partial}{\partial q_1}g(q_1,q_0) = \frac{M}{2N}\frac{n(q_0(n-2)+q_1)}{(q_0-q_1)(q_0(n-1)+q_1)} + \frac{\partial}{\partial q_1}g_E(q_1,q_0) = 0\ , \tag{.91}$$

$$\frac{\partial}{\partial q_0}g(q_1,q_0) = \frac{M}{2N}\frac{q_0 n(n-1)}{(q_0-q_1)(q_0(n-1)+q_1)} + \frac{\partial}{\partial q_0}g_E(q_1,q_0) = 0\ . \tag{.92}$$

We start by simplifying the form of function $g_E(q_1,q_0)$. Let us denote the integral over λ in Eq. (.86) by a shorthand $L(q_1,q_0)$,

$$\begin{aligned} L(q_1,q_0) &= \int \frac{1}{\sqrt{2\pi(q_1-q_0)}}\exp\left\{-\frac{(\sqrt{q_0}t+\lambda)^2}{2(q_1-q_0)} + \beta'\max\left[\lambda-\mu(q_1),0\right]\right\}d\lambda \\ &= 1 - H(t_1) + e^{\frac{1}{2}t_2^2 - t_1 t_2}H(t_1-t_2)\ , \end{aligned} \tag{.93}$$

where $H(x) = \int_x^{+\infty}\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}y^2}dy$ is a complementary Gaussian cumulative distribution, $t_1 = \frac{t\sqrt{q_0}+\mu(q_1)}{\sqrt{q_1-q_0}}$ and $t_2 = \beta'\sqrt{q_1-q_0}$. We can now write

$$g_E(q_1,q_0) = \log\int Dt[L(q_1,q_0)]^n\ . \tag{.94}$$

Asymptotic solution

In the limit $\beta'\to\infty$, we observe that $q_1 - q_0 \sim a/\beta'$, which defines a new parameter a. We now derive the asymptotic form of the free energy function.

The integral over λ in function $L(q_1,q_0)$ becomes a saddle point integral with respect to β',

$$\begin{aligned} L(q_1,a) &= \int d\lambda\exp\left\{-\beta'\frac{(t\sqrt{q_1-a/\beta'}+\lambda)^2}{2a} + \beta'\max\left[\lambda-\mu(q_1),0\right] + \frac{1}{2}\log\beta' - \frac{1}{2}\log(2\pi a)\right\} \\ &= \int d\lambda e^{\beta' f_{\rm sp}(\lambda)}\ . \end{aligned} \tag{.95}$$

with

$$f_{\rm sp}(\lambda) = -\frac{(t\sqrt{q_1}+\lambda)^2}{2a} + \max\left[\lambda-\mu(q_1),0\right]\ . \tag{.96}$$

The saddle point is the solution of

$$\frac{\partial}{\partial\lambda}f_{\rm sp}(\lambda) = -\frac{t\sqrt{q_1}+\lambda}{a} + \Theta\left(\lambda-\mu(q_1)\right) = 0\ , \tag{.97}$$

which as a function of t, is solved by

$$\lambda^*(t) = \begin{cases} -t\sqrt{q_1} & \text{if } t > \frac{\frac{a}{2}-\mu(q_1)}{\sqrt{q_1}} \\ a - t\sqrt{q_1} & \text{if } t \leq \frac{\frac{a}{2}-\mu(q_1)}{\sqrt{q_1}} \end{cases} , \tag{.98}$$

giving

$$f_{\rm sp}(\lambda^*(t)) = \begin{cases} 0 & \text{if } t > \frac{\frac{a}{2}-\mu(q_1)}{\sqrt{q_1}} \\ \frac{a}{2} - t\sqrt{q_1} - \mu(q_1) & \text{if } t \leq \frac{\frac{a}{2}-\mu(q_1)}{\sqrt{q_1}} \end{cases} . \tag{.99}$$

Integrating $L(q_1, a)$ over t for $\beta' \to \infty$ gives

$$\begin{aligned} \lim_{\beta'\to\infty} \int [L(q_1,a)]^n \, Dt &= \lim_{\beta'\to\infty} \int [L(q_1,a)]^{\frac{\beta}{\beta'}} \, Dt \qquad (.100) \\ &= \lim_{\beta'\to\infty} \int e^{\beta f_{\rm sp}(\lambda^*(t))} Dt \\ &= \int_{-\infty}^{\frac{a-2\mu(q_1)}{2\sqrt{q_1}}} Dt \, e^{\beta\left(\frac{a}{2} - t\sqrt{q_1} - \mu(q_1)\right)} + \int_{\frac{a-2\mu(q_1)}{2\sqrt{q_1}}}^{+\infty} Dt \\ &= e^{\frac{\beta^2 q_1}{2} - \beta\mu(q_1)} H\left(\frac{\mu(q_1) - \frac{a}{2}}{\sqrt{q_1}} - \beta\sqrt{q_1}\right) + \left(1 - H\left(\frac{\mu(q_1) - \frac{a}{2}}{\sqrt{q_1}}\right)\right) , \end{aligned}$$

and

$$g_E(q_1, a) = \log\left[\left(1 - H\left(\frac{\mu(q_1) - \frac{a}{2}}{\sqrt{q_1}}\right)\right) + e^{\frac{\beta^2 q_1}{2} - \beta\mu(q_1)} H\left(\frac{\mu(q_1) - \frac{a}{2}}{\sqrt{q_1}} - \beta\sqrt{q_1}\right)\right] . \tag{.101}$$

Finally, we can write the free energy function in the limit $\beta' \to \infty$

$$\begin{aligned} -\beta f(\beta) &= \frac{M}{2N} \log\left(\frac{a^* + \beta q_1^*}{a^*}\right) + g_E(q_1^*, a^*) \qquad (.102) \\ &= \frac{M}{2N} \log\left(\frac{a^* + \beta q_1^*}{a^*}\right) - \beta f_c\left(\beta\sqrt{q_1^*}, \frac{\mu(q_1^*) - \frac{a^*}{2}}{\sqrt{q_1^*}}\right) , \qquad (.103) \end{aligned}$$

where q_1^* and a^* are the saddle point values and $f_c(\beta, \mu)$ is the free energy function of the fixed centre problem,

$$-\beta f_c(\beta, \mu) = \log\left[(1 - H(\mu)) + e^{\frac{\beta^2}{2} - \beta\mu} H(\mu - \beta)\right] . \tag{.104}$$

The probability of score S is expressed by

$$p(S) = e^{N\omega(S/N)} \tag{.105}$$

where the intensive entropy is

$$\omega(s) = -\max_{\beta} \left[\beta s + \beta f(\beta)\right] . \tag{.106}$$

Zusammenfassung

Clustering, das Gruppieren von Datenpunkten aufgrund ihrer beobachteten Eigenschaften, ist eines der wichtigsten Werkzeuge in der Datenanalyse. Es wird häufig in der Analyse von Genexpressionsdaten verwendet, um Gene zu identifizieren, die ähnliche biologischen Funktionen haben. Cluster von Genexpressionsmustern lassen oft auf einen gemeinsamen regulatorischen Prozess der beteiligten Gene schließen. Cluster von experimentellen Bedingungen, z.B. von unterschiedlichen Geweben in einem Organismus, sind ein Hinweis auf einen ähnlichen Zustand der Zelldifferenzierung. Die zuletzt genannte Eigenschaft wird häufig zur Klassifikation von Tumordaten verwendet.

Diese Dissertation etabliert statistische Grundlagen für Clustering in hochdimensionalen Daten. Die neu eingeführten Methoden basieren zu großen Teilen auf Erkenntnissen der statistischen Mechanik. Zuerst werden deshalb in Kapitel 2 grundlegende Konzepte und Algorithmen der statistischen Mechanik eingeführt.

In Kapitel 3 wird ein neues probabilistisches Model für Cluster im hochdimensionalen realen Raum vorgeschlagen. Motiviert durch die Merkmale von Genexpressionsdaten werden verschiedene Observablen eines Clusters definiert: *Punktdichte*, *Positions-Bias* und *Richtungsdichte*. Diese Observablen messen in verschiedener Weise Ähnlichkeiten zwischen Datenpunkten und beschreiben die Hintergrundverteilung zufälliger Datenpunkte. Daraus wird eine sogenannte Score-Funktionen für Cluster abgeleitet.

Obwohl Gene mit ähnlicher Funktion mit hoher Wahrscheinlichkeit Cluster in Genexpressionsdaten bilden, können auch zufällig verteilte Datenvektoren Cluster bilden und hohe Cluster-Scores erhalten. In Kapitel 4 wird deshalb die statistische Signifikanz für Cluster behandelt. Für die Score-Funktionen aus Kapitel 3 werden Verfahren zur Berechnung eines sogenannten p-Wertes vorgestellt. Der Funktion $p(S)$ gibt die Wahrscheinlickeit an, dass Zufallsvektoren einen Cluster-Score von mindestens S erhalten. Dieses Problem wir mit Methoden der statistischen Mechanik ungeordenter Systeme behandelt, die zu einer analytischen Lösung führen. In einer Anwendung auf Genexpressionsdaten aus Hefe wird gezeigt, dass Cluster-Scores p-Werte biologische Signifikanz von co-exprimierten Genen widerspiegeln; die biologische Signifikanz wird hierbei durch Gen-Ontologie-Parameter in den betrachteten Clustern gemessen. Dies zeigt, dass Gene mit ähnlichen biologischen Funktionen in der Tat als signifikante Cluster identifiert werden.

In Kapitel 5 wird ein weiterer wichtiger Aspekt statistischer Methoden für hochdimensionale Daten behandelt: Abhängigkeiten zwischen Vektorkomponenten. Solche

Abhängigkeiten sind häufig in Genexpressiondaten zu finden, beispielweise verursacht durch zeitlich aufeinanderfolgende Experimente im Rahmen von Zeitreihenexperimenten. Eine korrekte Abschätzung solcher Abhängigkeiten ist sowohl für das Clustering von experimentellen Bedingungen als auch zur Berechnung der Ähnlichkeiten von Genen von entscheidender Bedeutung. Für die Abschätzung von Abhängigkeiten von Vektorkomponenten ist die Berücksichtigung eines wichtigen Störfaktors notwendig: das Vorhandensein von Clustern von Datenvektoren. Wir schlagen eine Inferenzmethode basierend auf einer Mischverteilung vor, welche das zufällige Auftreten von Clustern vom wahren Signal trennt. In unserem Ansatz verwenden wir die probabilistischen Modelle für Cluster aus Kapitel 3. Wir wenden diese Methode auf das Problem der Tumorprobenklassifizierung an.

In Kapitel 6 wird der Algorithmus zur Berechnung von signifikanzbasiertem Clustering vorgestellt. Der Algorithmus sucht die beste Zerlegung der Daten als Mischung von zufälligen Datenvektoren (aus der Hintergrundverteilung) und statistisch signifikanten Clustern im Sinne unserer Theorie. Beim Auffinden von Clustern von Datenvektoren schätzt der Algorithmus ab, welches Ähnlichkeitsmaß die Abhängigkeiten zwischen Vektorkomponenten am besten bescheibt. Des weiteren erlaubt die probabilistische Mischverteilung die Verwendung von Ausgangwahrscheinlichkeiten, die Cluster mit grossen p-Werten bestraft. In einer Anwendung auf Genexpressionsdaten von Hefe und Mensch wird gezeigt, dass dieser Mischverteilungs-Ansatz die biologische Signifikanz der erhaltenen Cluster erhöht.

Summary

Clustering, which involves dividing data elements into classes based on their observed properties, is one of the main tools in exploratory data analysis. It is used widely in the analysis of gene expression, where one searches for structures related to the underlying biological mechanisms. Clusters of gene expression patterns are a signature of a common regulatory process of the involved genes. Clusters of experimental conditions, e.g. tissues in an organism, imply similar states of cell differentiation. The latter property is used in the tumour sample classification.

This thesis establishes a statistical grounding for cluster analysis in high-dimensional data. The methods used in the thesis are strongly influenced by solutions from the field of statistical mechanics. The basic concepts and computational methods of statistical mechanics are summarised in Chapter 2.

In Chapter 3, we propose probabilistic models for vectors in high-dimensional real space. Motivated by the characteristics of gene expression data, we discuss different properties defining a cluster: *point density*, *positional bias*, and *directional density* (defined in Chapter 3). These properties are related to different choices of a similarity measure and of a background distribution for unclustered vectors. We consider several combinations of such background distributions and similarity measures, and we arrive at well-defined scoring schemes for clusters.

Clusters in data usually arise due to an underlying functional mechanism. However, even unrelated vectors drawn from the background distribution can form agglomerations which by chance resemble clusters and yield high cluster scores. In Chapter 4, we address the problem of the statistical significance of clusters. For the scoring schemes proposed in Chapter 3, we compute the cluster score p-value, which tells how likely it is to observe a group of random vectors with the same or higher score. Our analytical solution is based on a mapping to a problem from the statistical mechanics of disordered systems. In an application to yeast gene expression data, we show that the cluster score p-value is in agreement with the biological significance of clustered genes, as measured by enrichment of considered clusters in gene ontology terms (i.e. known functional annotations of genes).

In Chapter 5, we focus on another important aspect of the statistics of high-dimensional data: dependencies between vector components. Such dependencies are prevalent in gene expression data, for example between subsequent time points in time-course experiments. Correct estimation of such dependencies is crucial both for clustering of experimental conditions, and for computation of similarities of gene expression vectors. Here, we show that the estimation of vector-component dependencies requires

accounting for an important confounding factor: the presence of clusters of data vectors. We propose a mixture-model-based inference method, which disentangles the spurious effect of clusters from the true signal. We successfully apply our method to the problem of tumour sample classification.

In Chapter 6, we propose the significance-based clustering algorithm. The algorithm seeks the best representation of data as a mixture of the background and of clusters characterised by a statistically significant score. In the implementation of this approach, we draw from all concepts discussed in the preceding chapters of this thesis: In the process of finding clusters of vectors, the algorithm estimates the metric which accounts for dependencies between components of the vectors. Further, using the probabilistic framework of the mixture-model, it assigns low prior probability, and effectively penalises, clusters with high cluster score p-value. In application to gene-expression data of yeast and human, we show that the significance-constraint improves the biological significance of resulting clusters.

Curriculum Vitae

For reasons of data protection,
the curriculum vitae is not included in the online version

For reasons of data protection,
the curriculum vitae is not included in the online version

Ehrenwörtliche Erklärung

Hiermit erkläre ich, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Hilfsmittel und Quellen verwendet habe.

Berlin, Mai 2011 Marta Łuksza