UNIVERSIDADE DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

Learning from High-Dimensional Data using Local Descriptive Models

Rui Miguel Carrasqueiro Henriques

Supervisor: Doctor Sara Alexandre Cordeiro Madeira

Thesis approved in public session to obtain the PhD Degree in
Information Systems and Computer Engineering

Jury final classification: **Pass with Distinction and Honor**

Jury

Chairperson:	Chairman of the Scientific Board
Members of the Committee:	Doctor Mário Alexandre Teles de Figueiredo
	Doctor Joaquín Dopazo Blásquez
	Doctor Miguel Francisco de Almeida Pereira da Rocha
	Doctor Francisco João Duarte Cordeiro Correia dos Santos
	Doctor Sara Alexandre Cordeiro Madeira

2016

UNIVERSIDADE DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

Learning from High-Dimensional Data using Local Descriptive Models

Rui Miguel Carrasqueiro Henriques

Supervisor: Doctor Sara Alexandre Cordeiro Madeira

Thesis approved in public session to obtain the PhD Degree in
Information Systems and Computer Engineering

Jury final classification: **Pass with Distinction and Honor**

Jury

Chairperson: Chairman of the Scientific Board

Members of the Committee: Doctor Mário Alexandre Teles de Figueiredo, *Full Professor, Instituto Superior Técnico, Universidade de Lisboa*
Doctor Joaquín Dopazo Blásquez, *Coordinator Researcher, Centro de Investigación Príncipe Felipe, Spain*
Doctor Miguel Francisco de Almeida Pereira da Rocha, *Associate Professor, Escola de Engenharia, Universidade do Minho*
Doctor Francisco João Duarte Cordeiro Correia dos Santos, *Associate Professor, Instituto Superior Técnico, Universidade de Lisboa*
Doctor Sara Alexandre Cordeiro Madeira, *Assistant Professor, Instituto Superior Técnico, Universidade de Lisboa*

Funding Institution
Fundação para a Ciência e Tecnologia

2016

Abstract

Models learned from high-dimensional data, where the high number of features usually exceeds the number of observations, have higher propensity to either overfit or underfit data. In this context, it is thus important to focus the learning on regions of interest, such as subsets of features, guaranteeing that these regions are both informative and statistically significant. Although a composition of relevant regions can be learned under specific assumptions to offer these guarantees, the state-of-the-art learning methods place restrictive constraints on the allowed structure, coherency and quality of regions. This has prevented the understanding of how the properties of the selected regions affect the performance of descriptive and classification methods in both tabular and structured data contexts.

In this work, we propose robust, flexible and statistically significant local descriptive models and study their relevance to improve (associative) classification in high-dimensional data contexts. This task is tackled in three major steps. *First*, we propose new local descriptive models from tabular and structured data with robustness and flexibility guarantees. In the presence of matrices and network data, the focus is placed on learning biclustering models able to tackle existing challenges: learn from regions with flexible coherency (additive, symmetric, plaid and order-preserving models); guarantee scalable searches; robustness to varying forms and degree of noise; model regions from sparse data; and effectively incorporate background knowledge. In the presence of structured data, possibly given by multivariate time series or multi-sets of events, the focus is placed on new deterministic and generative methods to learn local descriptive models given by cascades of modules or arrangements of informative events. *Second*, we propose principles to both assess and guarantee the statistical significance of these descriptive models. *Third*, the previous contributions are extended towards labeled data contexts, and new training and testing functions are proposed to learn associative classification models. In this context, we assess the impact of varying structures, coherencies and quality of local descriptive models on the performance of classifiers, and combine statistical significance and accuracy views to study and revise their behavior. Finally, we extend these contributions for data with structured classes to adequately answer predictive tasks.

The proposed contributions were applied to tackle a wide-set of real-world tasks in biomedical and social domains, including the learning of descriptive and predictive models from gene expression data, repositories of health records, clinical data, collaborative filtering data, and (biological and social) networks.

Keywords:
High-Dimensional Data
Structured Data
Biclustering
Local Descriptive Models
Associative Classification
Biomedical Data Analysis
Statistical Significance
Multivariate Time Series
Multi-Sets of Events
Sparse Data

Resumo

A aprendizagem de modelos a partir de dados com elevada dimensionalidade, onde o número de atributos pode exceder o número de observações, é propensa aos riscos de sobre- e sub-ajustamento. Neste contexto, é importante focar a aprendizagem em regiões de interesse, como subconjuntos de atributos, garantindo que estas regiões são informativas e estatisticamente significativas. No entanto, o estado-da-arte em aprendizagem coloca estritas restrições na estrutura, coerência e qualidade destas regiões. Isto previne a compreensão de como as propriedades das regiões seleccionadas afectam a performance de métodos para a descrição e classificação de dados tabulares e estruturados.

Para responder a este problema, este trabalho propõe modelos descritivos locais com garantias de robustez, flexibilidade e significância estatística, e estuda a sua relevância para melhorar a classificação em dados de elevada dimensionalidade. Este objectivo é enderaçado em três passos. Primeiro, novos modelos descritivos locais – flexíveis e robustos – são propostos. Na presença de dados tabulares e redes, o foco é colocado na aprendizagem de modelos de biclustering capazes de endereçar os desafios existentes: aprender regiões com coerências não-triviais (modelos aditivos, simétricos, sobrepostos e baseados em ordenações); promover a escalabilidade das procuras; garantir a robustez a differentes formas e graus de ruído; modelar regiões a partir de dados esparsos; e incorporar conhecimento disponível. Na presença de dados esruturados, possivelmente caracterizados por séries temporais multivariadas ou multi-conjuntos de eventos, o foco é colocado na definição de métodos determinísticos e estocásticos para a aprendizagem de modelos descritivos locais associados a cascatas de módulos ou arranjos de eventos. Segundo, novos princípios são propostos para avaliar e garantir a significância estatística destes modelos descritivos. Terceiro, as contribuições anteriores são alargadas a dados anotados, e novas funções de treino e teste são propostas para a aprendizagem de modelos de classificação. Neste contexto, este trabalho mede o impacto do uso de modelos descritivos locais (com variável estrutura, coerência e qualidade) na performance dos classificadores, e estuda e revê o seu comportamento de acordo com critérios de significância estatística. Por fim, estas contribuições são estendidas a dados com classes estruturadas para responder a problemas de previsão.

As contribuições propostas foram aplicadas num conjunto alargado de problemas reais em domínios biomédicos e sociais, incluindo a aprendizagem de modelos descritivos e classificadores em dados de expressão genética, repositórios de eventos clínicos, dados colaborativos, e redes sociais e biológicas.

Palavras Chave:
Dados com Elevada Dimensionalidade
Dados Estruturados
Biclustering
Modelos Descritivos Locais
Classificação Associativa
Análise de Dados Biomédicos
Significância Estatística
Séries Temporais Multivariadas
Multi-Conjuntos de Eventos
Dados Esparsos

Reconhecimentos

O presente documento é o resultado de um longo trabalho realizado em circunstâncias únicas, dentro do qual me revejo como humilde facilitador. Ele é primariamente o resultado das pessoas que o inspiraram, das mentes científicas que o precederam e nele cooperaram, e dos seres que comigo partilharam este percurso.

Antes demais, o meu profundo agradecimento aos meus pais, Rui e Elsa. Ao seu incessável apoio, dedicação e contínuo respeito pela forma como conduzo a minha vida. A vocês, o meu Amor.

A minha profunda gratidão a AJ Miller, Mary Luck, Inelia Benz, Almine, Manuela Melo, Luís Morgado e Leslie Temple Thurston pela forma como tocaram e transformaram a minha vida. À minha musa, Maria Flávia de Monsaraz, por me ter revelado a Ordem da Vida.

O meu agradecimento ao primeiro responsável por este trabalho. Sara Madeira. Obrigado. A sua integridade, confiança e compromisso para uma comunicação aberta constituíram, sem dúvida, o pilar do presente trabalho. A sua humanidade e genuína atenção foram o maior segredo para a condução deste trabalho, preenchendo as minhas madrugadas de trabalho com ânimo. Mais, a sua aguda visão científica foi essencial para o desenvolvimento dos conteúdos. Em 2012, os apontamentos gráficos e coloridos do seu moleskine azulão preencheram o meu mundo por breves dias, dando origem (quatro anos mais tarde) à actual tese.

Agradeço às pessoas que contribuíram para a aprimoração desta tese. Aos membros do júri por toda a sua pronto atenção e tempo dedicados, e inúmeras partilhas em prol da qualidade do presente trabalho e da minha formação pessoal. O meu sincero agradecimento a Cláudia Antunes por ter despoletado em mim a genuína paixão pela aprendizagem automática e pelo seu papel nas contribuições dos capítulos IV-3 e VI-7. A sua presença durante os meus primeiros anos de investigação marcou positivamente o meu percurso. Ainda, o meu agradecimento a todos os revisores científicos dos artigos decorrentes deste trabalho. Ao Franscisco Ferreira pelo seu papel na produção das interfaces gráficas do software decorrente desta tese. Aos meus colegas de equipa, Telma Pereira, Sofia Teixeira e Rita Levy, pela sua presença radiosa nos meus dois últimos anos de trabalho.

Agradeço à Fundação para a Ciência e Tecnologia e aos cidadãos portugueses o financiamento do meu doutoramento, através da bolsa SFRH/BD/75924/2011, que possibilitou a realização harmoniosa desta tese. Acredito que as contribuições decorrentes deste trabalho demonstram o meu empenho para avançar o estado da ciência computacional em Portugal e não só. Quero ainda agradecer às instituições de acolhimento, Inesc-ID, e de atribuição do grau, IST (Universidade de Lisboa), pela plataforma de suporte conduciva à realização dos trabalhos.

A minha gratidão a Marta Oliveira, Pedro Gonçalves, Pedro Cosme, Gonçalo Ferreira, Sílvia Pina, Elsa Torres, Mikel Damon Miller, Alexandre Camões, João Belchior, Samantha Nogueira, Sylvia Alves, Teresa Castanheira, Pedro Policarpo, Maria Júdice, César Moniz, Oet Grebke e António Barreto. Amigos de alma, cujos percursos se cruzaram com o meu de uma forma irreversível. E, claro, ao meu irmão Miguel.

A todos vocês, e ao leitor atento, dedico esta tese com verdade e humildade.

8 de Outubro de 2015,

Notation

Some structures apart from the usual text, figures and tables are used in this work. Their inclusion aims to better organize the conveyed ideas so its content can be easily assimilated by the reader.

Definitions. The introduction of critical concepts are framed by a light blue box. Exemplifying:

Def. 1 A **definition** is a passage describing, and possibly formalizing, the meaning of a concept.

Requirements and Contributions. Key premises for the thesis validation are framed by a green box:

R1 This is an illustrative *requirement*.

Conceptual Maps. Taxonomies are introduced in the beginning of each section to guide the reading. Their coloring aims to better separate concepts, not having other meaning than that.

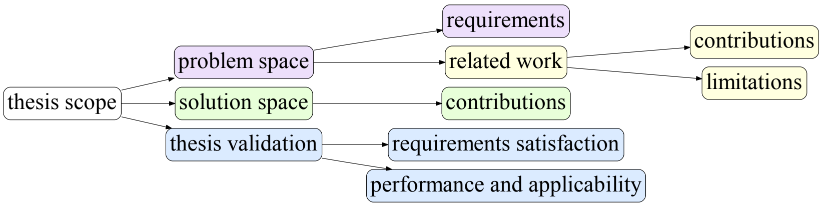


Figure 1: Illustrative conceptual map to structure the covered contents of one section, thus promoting clarity.

Framed text. Colored frames are either used to illustrate basic concepts (Basics label) or to expose contextual contents and complementary readings (Pointers label) in order to guarantee that the main text remains concise.

Basics: Suggestion

Experts may choose to skip these frames.

Pointers: Further information

Examples of pointers include alternative lines of research and applications not directly related with the task under analysis.

Source Code. Algorithms are presented using pseudo-code. Illustrating:

Algorithm 1: The Min-Error algorithm with an earliest first heuristic

```
Input: Set of tasks and processors
Output: Mapping of tasks to processors
while Unscheduled tasks remaining do
    foreach Processor $j$ do
        if $FT(a,j) \leq FT(a,c)$ then $c = j$;
    Schedule(Task $a$ on Processor $c$) ;
```

Cross-Referencing. The document is organized in seven books (I-VII), each grouping a set of chapters. References to sections, figures, tables and the previous structures inside the referee chapter are presented standardly (e.g. *Section 1.1*), while references to contents outside of the chapter are preceded by the book index (e.g. *Section I-1.1*).

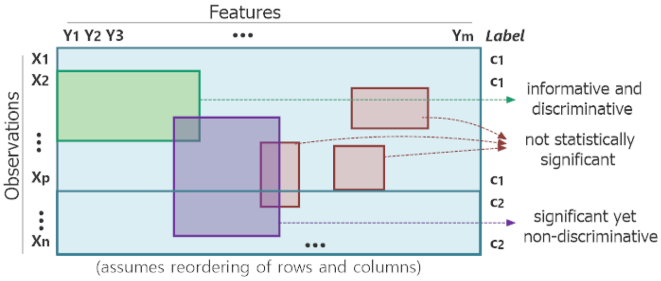
Contents

I

Foundations

1 Introduction

Learning from high-dimensional data, where the high number of features can exceed the number of observations, is challenged by an inherent complexity and generalization difficulty. In these data contexts, these challenges can be minimized by focus the learning on specific regions of interest (such as subsets of features) [316, 302]. However, the lack of flexibility on how the existing learning methods select such regions is associated with three major problems: *1)* the inclusion of non-relevant regions (promoting overfitting), *2)* the exclusion of relevant regions (promoting underfitting), and *3)* the modeling of apparently relevant regions, yet not statistically significant [105, 316, 22]. As illustrated in Figure 1.1, learning from high-dimensional data is a challenging task since not all regions are equally informative and, even when informative, regions may not be statistically significant. Furthermore, they can be significantly informative yet non-significantly discriminative.



Features
Y1 Y2 Y3
•••
Ym
Label
Observations
X1
X2
⋮
Xp
⋮
Xn
c1
c1
c1
c2
c2
informative and discriminative
not statistically significant
significant yet non-discriminative
•••
(assumes reordering of rows and columns)

Figure 1.1: Learning from high-dimensional data: relevance of selecting coherent, discriminative, significant regions.

This work aims to tackle these challenges by learning flexible, robust and statistically significant descriptive models and associative classification models from relevant regions of a high-dimensional data space. This learning task is addressed for tabular and structured data. In this context, our goal is to systematically study how does the performance of descriptive and classification models vary with the homogeneity, discriminative power and statistical significance of the selected regions. The underlying *hypothesis* is that the adequate selection and composition of regions improves the performance guarantees of local descriptive models and (associative) classifiers learned from high-dimensional data. As a result, this understanding opens a window of opportunity: new principles can be inferred to revise the behavior of existing learning methods.

The importance of this thesis is driven by two major observations. First, the need to validate the increasing number of scientific statements from the analysis of high-dimensional data without proper statistical assessments [316]. This is particularly critical across biomedical domains, due to the severity of implications that some statements might have on human health and upcoming research. Second, the need to face the increasing dimensionality of the available data, without being susceptible to the problems of feature selection and peer procedures for dimensionality reduction. The focus on a small subset of features is commonly associated with the exclusion of relevant regions and inclusion of non-relevant elements, contributing to the over/underfitting risk.

The in-depth study of how to optimize the performance of the target learning methods, while guaranteeing their statistical significance, is thus critical to answer a wide-set of real-world learning problems. In this work, we address the tasks of learning from: *1)* tabular data from biomedical domains with a high number of molecular units or clinical features per sample or patient, and social domains with a high number of rated items, traits or behavioral features per subject; *2)* weighted graphs given by large-scale biological and social networks; *3)* sequential data associated with (multivariate) time series with a high number of time points and/or (sliding) features; and *4)* structured data mapped from high-dimensional multi-sets of events, such as repositories of health records, trading decisions, (e-)commerce operations and browsing events.

This chapter is organized as follows. *Section 1.1* formalizes the universe of discourse of this thesis. *Section 1.2* explores the current limitations and opportunities of learning from high-dimensional data. *Section 1.3* structures the problem space according to its requirements. *Section 1.4* provides a high-level view on the contributions of this thesis. Finally, a roadmap for an easy exploration of the contents in this dissertation is provided in *Section 1.5*.

1.1 Universe of Discourse

This section formalizes the target learning task. Acording to Figure 1.2, follows a characterization of the possible *inputs* (high-dimensional data) in *Section 1.1.1*, the desirable *outputs* (learned models) in *Section 1.1.2*, and the learning *functions* in *Section 1.1.3*.

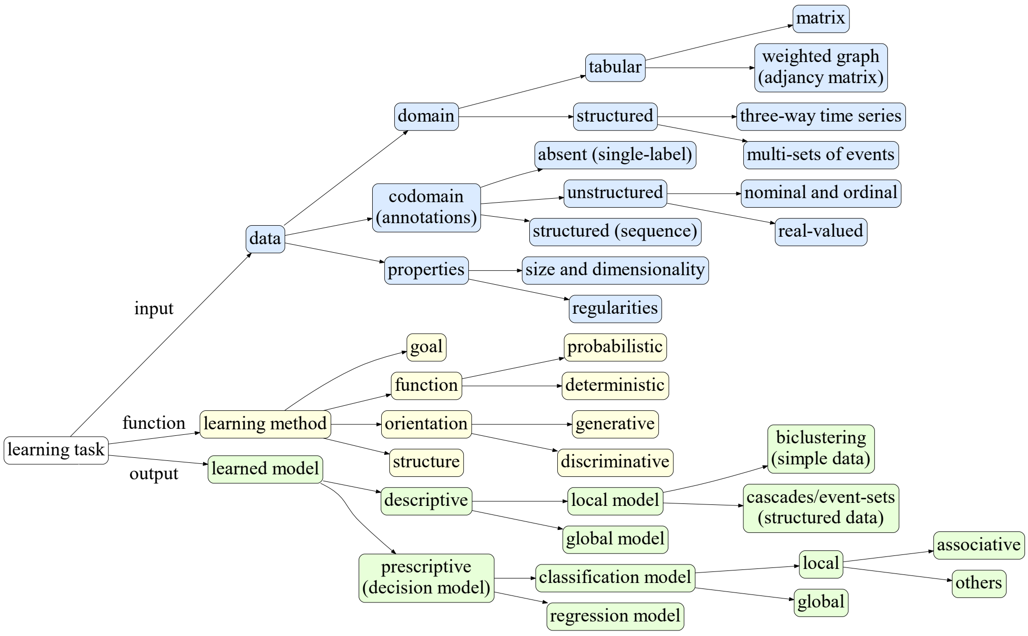


Figure 1.2: Learning task according to its input, function and output.

1.1.1 Input Data

Learning from high-dimensional data has both challenges dependent and independent of the input data format. In this work, we first tackle the task of learning from (high-dimensional) data given by real-valued matrices and weighted graphs. Second, we move towards learning from structured data given by multivariate time series, itemset sequences, multi-sets of events and multi-dimensional data.

Basics 1.1 Notation

A random variable (or random vector/matrix in bold) is a function denoted by a capital letter ($\mathbf{X}$, $\mathbf{Y}$, $\mathbf{A}$, C) from an event space into a sample space (denoted by caligraphic style: $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{A}$, $\mathcal{C}$), with outcomes denoted by the corresponding lower case ($\mathbf{x}$, $\mathbf{y}$, $\mathbf{a}$, c).

A dataset is defined by a set of observations, a sample of a (possibly) random vector $\mathbf{X}$ taking values on a sample space $\mathcal{X}$ with probability function $P_{\mathbf{X}}(\mathbf{x})$, or simply $P_{\mathbf{X}}$. In labeled datasets, the $\mathbf{X}$ observations are (possibly) conditionally dependent on the assigned labels $c \in C$, also referred as classes, and thus described by a class-conditional probability function $P_{\mathbf{X}}(\mathbf{x}|c)$, or simply $P_{\mathbf{X}|C}$. As we move from tabular to structured datasets, the observations take values on a structured sample space $\mathcal{X}$.

Data Structures

Below we introduce tabular data structures. Real-valued matrices and weighted graphs can be seen as specializations of tabular data where features are numeric and thus $\mathcal{X} = \mathbb{R}^m$ (where m is the number of features). Missing interactions from graphs are seen as interactions with a zero weight. Tables 1.3-1.6 illustrate these data structures.

Def. 1.1 A **real-valued matrix A** with n observations (rows) $\mathbf{x}_i \in \mathbb{R}^m$, m features (columns) $\mathbf{y}_j \in \mathbb{R}^n$, and $n{\times}m$ elements $a_{ij} \in \mathbb{R}$ is a (n,m)-space. Let Σ be a set of categoric values (classes), a labeled real-value matrix given by a (n,m)-space is described by n labeled observations $(\mathbf{x}_i, c_i)$, where $\mathbf{x}_i \in \mathbb{R}^m$ and $c_i \in \Sigma$.

Def. 1.2 A **tabular dataset A** has n observations $\mathbf{x}_i \in \mathcal{X}$ (possibly labeled), m features $\mathbf{y}_j \in \mathcal{Y}_j$, and $n{\times}m$ elements $a_{ij} \in \mathcal{Y}_j$, where $\mathcal{Y}_j$ defines the domain of the $\mathbf{y}_j$ feature: either nominal, ordinal or numeric.

Basics 1.2 Tabular data structures

Figures 1.3 and 1.4 provide a real-valued matrix ($\mathbf{A}_1$) and an alternative tabular dataset ($\mathbf{A}_2$). $\mathbf{A}_1$ is a labeled (n=6,m=7)-space with 2 classes (4 c_1-conditional observations and 2 c_2-conditional observations). $\mathbf{A}_2$ has 5 observations and 6 features, each feature $\mathbf{y}_i$ taking values on a specific sample space $\mathcal{Y}_i$ with either numeric values ($\mathcal{Y}_1$ and $\mathcal{Y}_5$), nominal values with varying cardinality ($\mathcal{Y}_3$, $\mathcal{Y}_4$ and $\mathcal{Y}_6$), or ordinal values ($\mathcal{Y}_2$). Illustrating the concept of data elements: $a_{2,3}$=3.9 in $\mathbf{A}_1$ and $a_{2,3}$=b in $\mathbf{A}_2$.

Figure 1.3: Illustrative real-valued matrix ($\mathbf{A}_1$)

	$\mathbf{y}_1$	$\mathbf{y}_2$	$\mathbf{y}_3$	$\mathbf{y}_4$	$\mathbf{y}_5$	$\mathbf{y}_6$	$\mathbf{y}_7$	class
$\mathbf{x}_1$	2.7	-0.9	2.2	2.7	-0.9	1.1	0.9	c_1
$\mathbf{x}_2$	0.8	-2.1	3.9	0.1	-2.1	-0.1	1.9	c_1
$\mathbf{x}_3$	1.3	-3.8	2	-2.8	-3.8	0	0.1	c_1
$\mathbf{x}_4$	-2.7	-1.8	3.7	1.9	-2.2	-0.9	2.1	c_1
$\mathbf{x}_5$	0.7	2.9	0.2	-2	-0.8	1.9	-1.1	c_2
$\mathbf{x}_6$	-2.6	-3.1	-0.1	3.9	0.9	-2	1.2	c_2

Figure 1.4: Illustrative tabular dataset ($\mathbf{A}_2$)

	$\mathbf{y}_1$ ($\mathcal{Y}_1$=$\mathbb{R}$)	$\mathbf{y}_2$ ($\mathcal{Y}_2$=$\mathbb{N}$)	$\mathbf{y}_3$ ($\lvert\mathcal{Y}_3\rvert$=3)	$\mathbf{y}_4$ ($\lvert\mathcal{Y}_4\rvert$=7)	$\mathbf{y}_5$ ($\mathcal{Y}_2$=$\mathbb{N}$)	$\mathbf{y}_6$ ($\mathcal{Y}_6$={Y,N})	class
$\mathbf{x}_1$	0.3	3	A3	A4	63	Y	c_1
$\mathbf{x}_2$	1	5	B3	A4	22	Y	c_1
$\mathbf{x}_3$	-2.1	3	A3	F4	31	N	c_1
$\mathbf{x}_4$	-0.1	1	C3	E4	28	Y	c_2
$\mathbf{x}_5$	3.2	4	B3	A4	42	N	c_2

Def. 1.3 A *weighted bipartite graph* is defined by two disjoint sets of nodes $\mathbf{X}$=$\{\mathbf{x}_1,..,\mathbf{x}_n\}$ (observations) and $\mathbf{Y}$=$\{\mathbf{y}_1,..,\mathbf{y}_m\}$ (features) where $\mathbf{x}_i \in \mathbb{R}^m$ and $\mathbf{y}_j \in \mathbb{R}^n$, and weighted interactions $a_{ij} \in \mathbb{R}$ between nodes $\mathbf{x}_i$ and $\mathbf{y}_j$. A **weighted graph** is defined by a set nodes X=$\{\mathbf{x}_1,..,\mathbf{x}_n\}$ (observations and features) where $\mathbf{x}_i \in \mathbb{R}^n$ and $a_{ij} \in \mathbb{R}$ interactions between nodes $\mathbf{x}_i$ and $\mathbf{x}_j$. Given a set of labels Σ, nodes can be labeled $(\mathbf{x}_i, c_i \in \Sigma)$.

Basics 1.3 Network data: (weighted) graphs

Figure 1.5 depicts a weighted graph with labeled nodes. Figure 1.6 provides the result of mapping this graph into a real-valued matrix ($\mathbf{A}_3$), with 5 observations and 5 features. The weight of interaction between nodes with identifiers i and j corresponds to the a_{ij} element in the mapped matrix.

Figure 1.5: Illustrative weighted graph ($\mathbf{A}_3$).

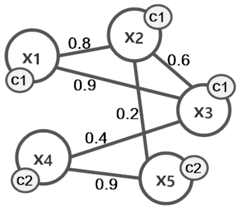


Figure 1.6: Real-valued matrix ($\mathbf{A}_3$) mapped from Figure I-1.5.

	$\mathbf{y}_1$ ($\mathbf{x}_1$)	$\mathbf{y}_2$ ($\mathbf{x}_2$)	$\mathbf{y}_3$ ($\mathbf{x}_3$)	$\mathbf{y}_4$ ($\mathbf{x}_4$)	$\mathbf{y}_5$ ($\mathbf{x}_5$)	class
$\mathbf{x}_1$	0	0.8	0.9	0	0	c_1
$\mathbf{x}_2$	0.8	0	0.6	0	0	c_1
$\mathbf{x}_3$	0.9	0.6	0	0.4	0	c_1
$\mathbf{x}_4$	0	0	0.4	0	0.9	c_2
$\mathbf{x}_5$	0	0.2	0	0.9	0	c_2

The introduced data structures are consider to be the default input along *Books II* and *III* of this document. Since new data structures are becoming increasingly prominent, bringing new challenges towards traditional learning tasks, we also consider multivariate time series, sequential databases and multi-sets of events. Figures 1.7-1.9 illustrate these data structures. We also provide mappings of multi-dimensional and relational databases into multi-sets of events in order to guarantee a wide-coverage of real-world (high-dimensional) data structures. An analysis of their application domains is provided in the next chapter.

Def. 1.4 A **three-way time series** (or cube) is a set of observations $\{\mathbf{x}_1, .., \mathbf{x}_n\}$, where each observation $\mathbf{x}_i$ defines a matrix with m features $\mathbf{y}_j \in \mathbb{R}^p$, p time points $\mathbf{t}_k \in \mathbb{R}^m$, and elements $a_{ijk} \in \mathbb{R}$ relating observation $\mathbf{x}_i$, feature $\mathbf{y}_j$ and time point $\mathbf{t}_k$. A cube is also referred as a *multivariate time series* database, where each time series has a m order and p time points. Given a set of labels Σ, observations (time series) can be labeled ($\mathbf{x}_i, c_i \in \Sigma$).

Basics 1.4 Multivariate time series

Figure 1.7 instantiates a labeled time series database $\mathbf{A}_4$, with 4 multivariate time series labeled with a class (*time series ⇒ class*), each time series (matrix) has a 5 multivariate order with measurements along 4 time points. Alternatively, Figure 1.7 can be seen as an integer cube with 4 matrices (or observations), each with 5 rows and 4 columns.

Figure 1.7: Illustrative set of multivariate time series ($\mathbf{A}_4$)

	$\mathbf{x}_1 \Rightarrow c_1$			
	$\mathbf{t}_1$	$\mathbf{t}_2$	$\mathbf{t}_3$	$\mathbf{t}_4$
$\mathbf{y}_1$	1	0	-2	-2
$\mathbf{y}_2$	1	1	-2	-2
$\mathbf{y}_3$	1	2	2	2
$\mathbf{y}_4$	0	2	2	2
$\mathbf{y}_5$	-2	1	0	1

	$\mathbf{x}_2 \Rightarrow c_1$			
	$\mathbf{t}_1$	$\mathbf{t}_2$	$\mathbf{t}_3$	$\mathbf{t}_4$
$\mathbf{y}_1$	1	-1	-2	-2
$\mathbf{y}_2$	0	0	-2	-2
$\mathbf{y}_3$	-1	2	2	2
$\mathbf{y}_4$	0	2	2	2
$\mathbf{y}_5$	1	0	1	0

	$\mathbf{x}_3 \Rightarrow c_2$			
	$\mathbf{t}_1$	$\mathbf{t}_2$	$\mathbf{t}_3$	$\mathbf{t}_4$
$\mathbf{y}_1$	1	-1	0	1
$\mathbf{y}_2$	2	2	0	-1
$\mathbf{y}_3$	2	2	-2	-2
$\mathbf{y}_4$	0	1	2	2
$\mathbf{y}_5$	-1	-1	2	2

	$\mathbf{x}_4 \Rightarrow c_2$			
	$\mathbf{t}_1$	$\mathbf{t}_2$	$\mathbf{t}_3$	$\mathbf{t}_4$
$\mathbf{y}_1$	0	-2	-1	0
$\mathbf{y}_2$	2	2	0	0
$\mathbf{y}_3$	2	2	-2	-2
$\mathbf{y}_4$	-1	0	2	2
$\mathbf{y}_5$	-2	-1	2	2

Def. 1.5 Given a set of items $\mathcal{L}$, let an (itemset) sequence be an ordered set of itemsets $<I_1, .., I_m>$, where $I_i \subseteq \mathcal{L}$. A *sequential database* is a set of n sequences (observations). Sequences can be labeled.

Def. 1.6 Let an event μ from a source (observation) $\mathbf{x}_i$ be a tuple (y_j, a_{ijk}, t_{ijk}), where $y_j \in \mathcal{Y}_j$ is the type of event (feature), a_{ijk} is its value and t_{ijk} the timestamp. A repository of *multi-sets of events* is a set of n observations $\mathbf{x}_i \in \mathcal{X}$, where each observation is a set of timestamped events. Observations can be labeled.

Basics 1.5 Sequential databases and multi-sets of events

Figures 1.8 and 1.9 show respectively an instance of a sequential database ($\mathbf{A}_5$) and of a multi-set of events ($\mathbf{A}_6$).

In Figure 1.8, itemset sequences were represented with a compact format: co-occurring items are delimited by curve parentheses and itemsets are concatenated. Illustrating, $\mathbf{x}_1$=$<\{a,h\},\{d\},\{a,b\},\{b,e,g\},\{a,c,f\}>$=$(ah)d(ab)(beg)(acf)$. $\mathbf{A}_5$ is a set of 5 labeled itemset sequences (3 c_1-conditional and 2 c_2-conditional observations) with items in $\mathcal{L}$ ($|\mathcal{L}|$=8). Illustrating further properties, $\mathbf{x}_4$ is an ordered set of 6 itemsets with a total of 12 items and an average number of 2 items per itemset.

The labeled multi-sets of events provided in Figure 1.9 ($\mathbf{A}_5$) has 4 observations (4 distinct sources of events) and 3 features (3 distinct types of events) with different feature domains ($\mathcal{Y}_1$=$\mathbb{N}$, $\mathcal{Y}_2$={y} and $\mathcal{Y}_3$={a,b,c}). Each observation has an arbitrary number of timestamped events per feature. For instance, $\mathbf{x}_1$ has a total of 4 event occurrences, 2 associated with $\mathbf{y}_1$ feature ((3,t_2) and (5,t_4)), no event associated with $\mathbf{y}_2$ feature and two events associated with $\mathbf{y}_3$ that co-occur in time point t_1.

Figure 1.8: Illustrative sequential database ($\mathbf{A}_5$)

	sequence ($\mathcal{L}$={a,b,c,d,e,f,g,h})	class
$\mathbf{x}_1$	$(ah)d(ab)(beg)(acf)$	c_1
$\mathbf{x}_2$	$(bd)(ab)(be)(bf)a$	c_1
$\mathbf{x}_3$	$(de)h(ab)g(bef)(fg)$	c_1
$\mathbf{x}_4$	$b(ab)(abce)(de)b(df)$	c_2
$\mathbf{x}_5$	$(ad)(ac)f(ad)(cdf)(ab)g$	c_2

Figure 1.9: Illustrative multi-sets of events ($\mathbf{A}_6$)

	$\mathbf{y}_1$ ($\mathcal{Y}_1$=$\mathbb{N}$)	$\mathbf{y}_2$ ($\lvert\mathcal{Y}_2\rvert$=1)	$\mathbf{y}_3$ ($\lvert\mathcal{Y}_3\rvert$=3)	class
$\mathbf{x}_1$	{(3,t_2),(5,t_4)}	$\emptyset$	{(a,t_2),(c,t_2)}	c_1
$\mathbf{x}_2$	{(2,t_2),(3,t_3),(4,t_4)}	{(y,t_3)}	{(c,t_1)}	c_1
$\mathbf{x}_3$	$\emptyset$	{(y,t_2),(y,t_4)}	{(b,t_3),(c,t_4)}	c_2
$\mathbf{x}_4$	{(2,t_1),(1,t_4)}	{(y,t_1)}	{(b,t_1),(a,t_3),(c,t_3)}	c_2

Labels: Data Codomain

As introduced, a labeled dataset is a sample from a class-conditional probability function $P_{\mathbf{X}|C}$, where $\mathbf{X}$ is a random vector taking values on a (possibly structured) sample space $\mathcal{X}$ (the *domain*), and C is a random variable with classes from a sample space $\mathcal{C}$ (often referred as *codomain*). Let Σ be a set of labels, the classes can be *nominal*, $\mathcal{C}$=Σ (default case), *ordinal* when Σ is an ordered set, or *numeric* when $\mathcal{C}$=$\mathbb{R}$. Like domains, codomains can be structured. In particular, we tackle the case where labels are associated with *categoric vectors*, $\mathcal{C}$=Σ^h.

Data Properties

A dataset is primarily characterized by the number of observations (*size*), dimensionality, and data regularities.

The *dimensionality* of a dataset is given by the number of columns/features in tabular data (m); number of nodes in weighted graphs; product of the number of features (multivariate order) and time points in multivariate time series ($m \times p$); average number of items per itemset sequence in sequential databases; and average number of events per observation in multi-sets of events. In this context, the criteria to decide whether a dataset is high-dimensional deserves some attention. **High dimensionality** has been seen not only as a product of dimensionality, but also dependent on the complexity of the learning task [553, 316]. There is a considerably agreement that a dimensionality superior to 100 can be considered already high for common learning tasks (based on the suggested cut-off thresholds to apply feature selection) [589, 215, 343]. Complementarily, the higher the learning complexity, the lower the dimensionality threshold to consider a dataset to be high-dimensional. As such, high-dimensionality can be seen as a result of three major aspects:

- number of observations and classes. The lower the ratio $n/|C|$ (where $|C|=1$ for non-labeled data), the higher is the learning complexity due to an increased difficulty to generalize;
- type of input data. In structured data contexts, the learning complexity increases more rapidly with an increasing number of features (or types of events) than with an increasing number of time points/partitions. Illustrating, for a multivariate time series database, the ratio $m\sqrt{p} > 100$ can be considered to be a more fair verification of high-dimensionality;
- the regularities of the input data. Illustrating, the higher the number of correlated features, the higher the learning complexity. A high number of studies aim to theoretically or empirically predict the minimum number of observations for an adequate learning based on the regularities of a given dataset with fixed dimensionality [218, 110, 178, 54]. High-dimensionality is here assumed when the number of available observations is lower than the expected minimum number of observations.

Basics 1.6 Data dimensionality

Consider the data from Figures 1.3-1.7. Their dimensionality is: *dim*($\mathbf{A_1}$)=m=7, *dim*($\mathbf{A_2}$)=m=6, *dim*($\mathbf{A_3}$)=n=5 and *dim*($\mathbf{A_4}$)=$m\times p$=20.

Let the number of itemsets of an itemset sequence $\mathbf{x}$ be $|\mathbf{x}|$, the i^{th} itemset of $\mathbf{x}$ be $\mathbf{x}^i$, and the number of items of an itemset I be $|I|$, then *dim*= $\frac{1}{n}\Sigma_{i=1}^{n}\Sigma_{j=1}^{|\mathbf{x}_i|}|\mathbf{x}_i^j|$.

Given a multi-set of events, let the set of events of type $j \in J$ from source $\mathbf{x}$ be $\mathbf{x}^j$, then *dim*= $\frac{1}{n}\Sigma_{i=1}^{n}\Sigma_{j\in J}|\mathbf{x}_i^j|$.

Accordingly, the dimensionality of datasets in Figures 1.8 and 1.9 is respectively *dim*($\mathbf{A_5}$) $= 11.2$ and *dim*($\mathbf{A_6}$) $= 4.75$

1.1.2 Output Models

Given a dataset $\mathbf{A}$ characterized by a set of underlying stochastic regularities, $P_{\mathbf{A}}$ (given by $P_{\mathbf{X}}$ or $P_{\mathbf{X}|C}$), a *learning task* aims to infer a model M from this (n,m)-space such that the error over $P_{\mathbf{A}}$ is minimized.

In this thesis, we tackle different learning tasks according to three major qualities: **flexibility** (the ability to affect the properties of the output model), **robustness** (the ability to deal with noisy and missing data) and statistical **significance** (the ability to exclude spurious regularities).

Two major types of models are considered: *descriptive* and *classification* models. These models can be further categorized according to the extent of the space coverage (*global* or *local*) and properties.

Descriptive Models

Def. 1.7 A **descriptive model** abstracts either locally or globally the regularities of a (possibly labeled) dataset, $M(\mathbf{A})$. A *regularity* is a trend in data. While *unsupervised* descriptors ($|C|=1$) model observations, $P_{\mathbf{X}}$, *supervised* descriptors ($|C|>1$) model class-conditional observations, $P_{\mathbf{X}|C}$.

A descriptive model (Def.1.9), also referred as an observation model, is either unsupervised or supervised depending on whether there is knowledge regarding the assignment of probability functions (labels per observation). Supervised descriptors are also referred as class-conditional observation models. Understandably, supervised descriptive models differ from decision models, such as classification models, since their goal is description and not the labeling of new observations.

Two distinct criteria of locality can be considered to identify whether a model is global or local. First criterion: the model is able to separate groups of observations with distinct regularities. Under this criterion, a descriptive model that approximates a distribution for each feature of a tabular dataset is not local. As such, descriptive models that define a multivariate distribution per class are global and thus not able to accurately describe datasets where groups of observations with a shared class show distinct regularities. Contrasting, clustering models are able to accommodate this locality criteria (Def.1.8). In particular, there are dedicated research streams on clustering to describe not only tabular data, but also multivariate time series, sequential databases and multi-sets of events [74, 36]. In tabular datasets, clustering can be alternatively applied to group subsets of features with correlated values across observations. This possibility can be seen as an alternative form of locality.

Def. 1.8 Given a dataset with $\mathbf{X}$ observations, a **clustering model** is a set of subsets of observations (clusters), $\{\mathbf{X}_1, .., \mathbf{X}_l\}$ where $\mathbf{X}_i \subseteq \mathbf{X}$, with intra-cluster and inter-cluster guarantees of (dis)similarity between observations.

Basics 1.7 Global mixtures *versus* clustering models

Considering the illustrative dataset $\mathbf{A_1}$ (Figure 1.3) with 2 classes and 7 real-valued features. Assuming independence between features, let us consider the simplistic task of learning a global mixture given by class-conditional multivariate Gaussian distributions. Given $\mathbf{A_1}$, then $\mathbf{y}_1|c_1 \sim N(\mu{=}\frac{1}{4}\Sigma_{i=1}^4 a_{i1}{=}0.5, \sigma^2{=}\frac{1}{4-1}\Sigma_{i=1}^4(a_{i1}\text{-}0.5)^2{=}5.3)$, $\mathbf{y}_1|c_2 \sim N(\text{-}1.0, 5.4)$, $\mathbf{y}_2|c_1 \sim N(\text{-}2.2, 1.5)$, $\mathbf{y}_2|c_2 \sim N(\text{-}0.1, 18.0)$, and so forth. Understandably, this model suffers from a major drawback. The inability to distinguish between subsets of observations with a shared class yet with different regularities leads to: biases in the Gaussian's mean and an increased variance that blurs the discriminative power of a feature. On top of this observation, there is a high overfitting propensity for data with a limited number of observations, such as $\mathbf{A_1}$. The high variance associated with the observed values for the illustrated features $\mathbf{y_1}$ and $\mathbf{y_2}$ shows that they are insufficient to effectively characterize and distinguish c_1 and c_2-conditional regularities.

Contrasting with this global mixture, let us consider a mixture given by a clustering model with 2 clusters of observations per class based on their Euclidean distance, where a cluster is characterized by the average value per feature (mean centroid). Given $\mathbf{A_1}$, the similarity between 2 observations is given by $d(\mathbf{x}_a, \mathbf{x}_b) = \sqrt{\Sigma_{j=1}^7(a_{aj} - a_{bj})^2}$, leading to the following groups of cluster for class c_1: cluster$_1$={$\mathbf{x}_2, \mathbf{x}_4$} and cluster$_2$={$\mathbf{x}_1, \mathbf{x}_3$}. The mean centroids of these clusters are [-0.95,-1.95,3.8,1,-2.15,-0.5,2] and [2,-2.35,2.1,-0.05,-2.35,0.55,0.5], respectively. For the given distance, this clustering model is able to achieve a reasonable intra-cluster similarity, low inter-cluster similarity and some notable differences between clusters from c_1 and c_2 classes.

Second criterion of locality: the model is able to identify regions given by subsets of observations and subsets of features in tabular data (or subsets of time points, items and events in structured data). Illustrating, in the presence of real-valued matrices, a local descriptive model is a composition of learned regions from the original space, where each region $\mathbf{B}_i$ is a (r_i, s_i)-space where $r_i \leq n \wedge s_i \leq m$. We consider this criterion to be the default locality criterion in this work due to its higher flexibility to discard non-informative and non-discriminative regions, a critical condition when learning from high-dimensional data (Figure 1.1). Under this criterion, clustering models, are considered to be global. Contrasting, more flexible descriptive models, such as biclustering models, are local.

Local Descriptive Models

Def. 1.9 A **local descriptive model** is a composition of regions from a (possibly structured) dataset. A *region* is a subset of overall data elements that satisfies certain homogeneity and (possibly) discriminative criteria.

Local descriptors aim to learn relevant regions, subspaces from a (possibly structured) sample data space. Given a tabular dataset, an element a_{ij} relates the $\mathbf{x}_i$ observation and $\mathbf{y}_j$ feature or nodes $\mathbf{x}_i$ and $\mathbf{x}_j$. Given a three-way time series, an element a_{ijk} relates the $\mathbf{x}_i$ observation (time series), $\mathbf{y}_j$ feature and $\mathbf{t}_k$ time point. Given a sequential database and multi-set of events, an element corresponds respectively to an occurring item and event.

As introduced, the properties of a local descriptive model depend on the structure of the input data. Below, we define flexible descriptive models for real-valued matrices, multivariate time series and itemset sequences. For these data structures, regions of interest are respectively given by biclusters, triclusters and sequential patterns. The formalized concepts associated with these descriptive models are instantiated in Basics 1.8, 1.9 and 1.10. To preserve conciseness, an in-depth analysis of these models, as well as of additional variants, is provided throughout *Books III* and *IV* of this document.

Def. 1.10 Given a real-valued matrix $\mathbf{A}$ with n observations (rows) $\mathbf{X}$ in $\mathbb{R}^m$ and m features (columns) $\mathbf{Y}$ in $\mathbb{R}^n$, a *bicluster* $\mathbf{B} = (\mathbf{I}, \mathbf{J})$ is a region/subspace of the original (n, m)-space, where $\mathbf{I} \subseteq \mathbf{X}$ is a subset of rows and $\mathbf{J} \subseteq \mathbf{Y}$ a subset of columns. The **biclustering** task aims to find a set of biclusters $\{\mathbf{B}_1, .., \mathbf{B}_l\}$ such that each bicluster $\mathbf{B}_i$ satisfies specific criteria of *homogeneity*, *discriminative power* in the presence of labels, and statistical *significance*.

Basics 1.8 Biclustering real-valued matrices

Considering the introduced $\mathbf{A_1}$ matrix in Figure 1.3, the (2,4)-space given by $\mathbf{B}_1$=($\mathbf{I}$={$\mathbf{x}_2, \mathbf{x}_4$}, $\mathbf{J}$={$\mathbf{y}_2, \mathbf{y}_3, \mathbf{y}_5, \mathbf{y}_7$}) is coherent, discriminative and significant, and can thus be seen as one bicluster from a biclustering model learned from $\mathbf{A_1}$. To facilitate the analysis of the properties of $\mathbf{B}_1$, Figure 1.10 provides a variant matrix of $\mathbf{A_1}$ where the values were rounded and both rows and columns were reordered. First, $\mathbf{B}_1$ has approximately constant values across observations, a common form of homogeneity. Second, the combination of values per bicluster's row, {-2,4,-2,2}, is supported by 2 observations with class c_1 and 0 observations with class c_2, indicating a potentially high discriminative power. Finally, half of the total elements from c_1-conditional data are covered by the bicluster, possibly indicating its statistical significance.

Figure 1.10: Illustrative bicluster in $\mathbf{A}_1$ matrix (rounded values and reordered rows/columns)

	$\mathbf{y}_1$	$\mathbf{y}_2$	$\mathbf{y}_3$	$\mathbf{y}_5$	$\mathbf{y}_7$	$\mathbf{y}_4$	$\mathbf{y}_6$	C
$\mathbf{x}_1$	3	-1	2	-1	1	3	1	c_1
$\mathbf{x}_2$	1	**-2**	**4**	**-2**	**2**	0	0	c_1
$\mathbf{x}_4$	-3	**-2**	**4**	**-2**	**2**	2	-1	c_1
$\mathbf{x}_3$	1	-4	2	-4	0	-3	0	c_1
$\mathbf{x}_5$	1	3	0	-1	-1	-2	2	c_2
$\mathbf{x}_6$	-3	-3	0	1	1	4	-2	c_2

Def. 1.11 Given a real-valued cube (multivariate time series database) $\mathbf{A}$ with n observations (matrices) $\mathbf{X}$, m features (rows) $\mathbf{Y}$ and p time points (columns) $\mathbf{T}$: a *tricluster* $\mathbf{B} = (\mathbf{I}, \mathbf{J}, \mathbf{K})$ is a subspace of the original space, where $\mathbf{I} \subseteq \mathbf{X}$ and $\mathbf{J} \subseteq \mathbf{Y}$ are subsets of observations and features, and $\mathbf{K} \subseteq \mathbf{T}$ is a subset of contiguous time points. Given $\mathbf{A}$, the **triclustering** task aims to find a set of triclusters $\{\mathbf{B}_1, .., \mathbf{B}_l\}$ such that each tricluster $\mathbf{B}_i$ satisfies specific criteria of *homogeneity*, *discriminative power* in the presence of labels, and statistical *significance*.

Def. 1.12 Given a real-valued cube $\mathbf{A}$ with n observations and a set of triclusters (modules) supported by the same subset of observations, there is a high chance that these modules are correlated. A *cascade* (or frequent response) R is a set of l modules $\{\mathbf{B}_1, .., \mathbf{B}_l\}$ related through r temporal dependencies $\mathbf{D} = \{d_1, .., d_r\}$, where d_i is a sequential constraint defining either a parallel occurrence $\{\mathbf{B}_i, \mathbf{B}_j\}$ or precedence $\mathbf{B}_i \Rightarrow \mathbf{B}_j$ between two modules. Given $\mathbf{A}$, the task of **modeling cascades** aims to learn a set of cascades $\{R_1, .., R_s\}$ satisfying specific criteria of *homogeneity*, *discriminative power* in the presence of labels, and statistical *significance*.

Basics 1.9 Modeling triclusters and cascades from real-valued cubes

Considering the integer cube $\mathbf{A_4}$ provided in Figure 1.7, some of its regularities $P_{\mathbf{A_4}}$ can be given by diverse coherent modules. Figure 1.11 illustrates some of these modules given by triclusters (subsets of observations, features and time points). We highlight 4 triclusters: 2 triclusters for c_1-conditional observations ($\mathbf{B}_1$=($\mathbf{I}_1$={$\mathbf{x}_1, \mathbf{x}_2$}, $\mathbf{J}_1$={$\mathbf{y}_3, \mathbf{y}_4$}, $\mathbf{K}_1$={$\mathbf{t}_2, \mathbf{t}_3, \mathbf{t}_4$}) and $\mathbf{B}_2$=($\mathbf{I}_2$={$\mathbf{x}_1, \mathbf{x}_2$}, $\mathbf{J}_2$={$\mathbf{y}_1, \mathbf{y}_2$}, $\mathbf{K}_2$={$\mathbf{t}_3, \mathbf{t}_4$})) and 2 triclusters for c_2-conditional observations ($\mathbf{B}_3$=($\mathbf{I}_3$={$\mathbf{x}_3, \mathbf{x}_4$}, $\mathbf{J}_3$={$\mathbf{y}_2, \mathbf{y}_3$}, $\mathbf{K}_3$={$\mathbf{t}_1, \mathbf{t}_2$}) and $\mathbf{B}_4$=($\mathbf{I}_4$={$\mathbf{x}_3, \mathbf{x}_4$}, $\mathbf{J}_4$={$\mathbf{y}_3, \mathbf{y}_4, \mathbf{y}_5$}, $\mathbf{K}_4$={$\mathbf{t}_3, \mathbf{t}_4$})). All of these triclusters: 1) show constant values per feature, a commonly accepted form of homogeneity; 2) are supported by observations of a single class, and thus discriminative; and 3) appear to be statistically significant as they are supported by all of the observations of a particular class and include at least 20% of the total elements from these observations.

Figure 1.11: Triclusters (and implicit cascades of triclusters) from the $\mathbf{A}_4$ cube

$\mathbf{x}_1 \Rightarrow c_1$

	$\mathbf{t}_1$	$\mathbf{t}_2$	$\mathbf{t}_3$	$\mathbf{t}_4$
$\mathbf{y}_1$	1	0	**-2**	**-2**
$\mathbf{y}_2$	1	1	**-2**	**-2**
$\mathbf{y}_3$	1	**2**	**2**	**2**
$\mathbf{y}_4$	0	**2**	**2**	**2**
$\mathbf{y}_5$	-2	1	0	1

$\mathbf{x}_2 \Rightarrow c_1$

	$\mathbf{t}_1$	$\mathbf{t}_2$	$\mathbf{t}_3$	$\mathbf{t}_4$
$\mathbf{y}_1$	1	-1	**-2**	**-2**
$\mathbf{y}_2$	0	0	**-2**	**-2**
$\mathbf{y}_3$	-1	**2**	**2**	**2**
$\mathbf{y}_4$	0	**2**	**2**	**2**
$\mathbf{y}_5$	1	0	1	0

$\mathbf{x}_3 \Rightarrow c_2$

	$\mathbf{t}_1$	$\mathbf{t}_2$	$\mathbf{t}_3$	$\mathbf{t}_4$
$\mathbf{y}_1$	1	-1	0	1
$\mathbf{y}_2$	**2**	**2**	0	-1
$\mathbf{y}_3$	**2**	**2**	**-2**	**-2**
$\mathbf{y}_4$	0	1	**2**	**2**
$\mathbf{y}_5$	-1	-1	**2**	**2**

$\mathbf{x}_4 \Rightarrow c_2$

	$\mathbf{t}_1$	$\mathbf{t}_2$	$\mathbf{t}_3$	$\mathbf{t}_4$
$\mathbf{y}_1$	0	-2	-1	0
$\mathbf{y}_2$	**2**	**2**	0	0
$\mathbf{y}_3$	**2**	**2**	**-2**	**-2**
$\mathbf{y}_4$	-1	0	**2**	**2**
$\mathbf{y}_5$	-2	-1	**2**	**2**

Understandably, due to the temporal nature inherent to $\mathbf{A_4}$, relations between the discovered triclusters can be hypothesized, leading to meaningful cascades of coherent behavior. Considering triclusters $\mathbf{B}_1$ and $\mathbf{B}_2$, they seem to occur in parallel, being the coherency of $\mathbf{B}_2$ possibly triggered by the starting of $\mathbf{B}_1$. Alternatively, $\mathbf{B}_3$ and $\mathbf{B}_4$ triclusters appear to related through a causal relation. From this observation, we can infer two cascades: R_1 with co-occurring modules {$\mathbf{B}_1, \mathbf{B}_2$}, and R_2 with precedent modules $\mathbf{B}_3 \Rightarrow \mathbf{B}_4$.

Def. 1.13 Let a sequence of itemsets $<I_{11}...I_{1n}>$ be contained in another sequence of itemsets $<I_{21}...I_{2m}>$ if $\exists_{1\leq i_1<..<i_n\leq m}$: $I_{11} \subseteq I_{2i_1},..,I_{1n} \subseteq I_{2i_n}$. Given a sequential database **A** with n itemset sequences (observations), a *sequential pattern* s is an itemset sequence contained in a significant subset of the observations. Given **A**, the **sequential pattern mining** task aims to discover a set of sequential patterns satisfying specific criteria of statistical *significance*, *discriminative power* in the presence of labels, and *dissimilarity* between patterns.

Basics 1.10 Modeling sequential databases

Most of existing local descriptive models for sequential databases are a composition of temporal patterns. Some of these temporal patterns assume temporal contiguity (such as motifs and strings), while others exclude the possibility to model co-occurrences. Contrasting, local descriptive models given by sequential patterns flexibly capture frequent co-occurrences and precedences (Def.1.13). Given $\mathbf{A_4}$ database provided in Figure 1.7 and a strict criteria of significance by requiring patterns to be supported by all observations of a given class and to have a minimum number of 6 items, 2 sequential patterns can be retrieved: $\mathbf{s}_1$=$d(ab)(be)f$ and $\mathbf{s}_2$=$a(ac)d(df)$. $\mathbf{s}_1$ is supported by all c_1-conditional observations and $\mathbf{s}_2$ by all c_2-conditional observations. Both have 4 frequent precedences and an average number of 1.5 co-occurring items per itemset. Besides the significance criteria, these sequential patterns are dissimilar and appear to be discriminative as they are supported by observations of a single class only.

Classification Models

We now move from descriptive to prescriptive settings to be able to answer a wider range of learning tasks.

Def. 1.14 Given a set of labeled observations, a *decision model* is a mapping function between observations and classes, $M : \mathcal{X} \rightarrow C$. A decision model is a **classification model** in the presence of categoric labels, C=Σ, and a *regression model* in the presence of numeric labels, C=$\mathbb{R}$.

Decision models are inferred from the conditional regularities of the input space, $P_{\mathbf{X}|C}$.

Given a set of labeled observations $(\mathbf{x}_i, c)$, classifiers learn a mapping from $\mathcal{X}$ to C given by *decision rules* for labeling (unlabeled) observations. When observations are labeled with real values, we are in the presence of parameter estimation problems (also called point estimation problems), in which an unknown scalar quantity can be estimated using regression models.

Similarly to descriptive models, decision models can either be conceptually divided as global or local, depending on whether relations are inferred from the overall data space or from informative and discriminative regions.

Local Classification Models

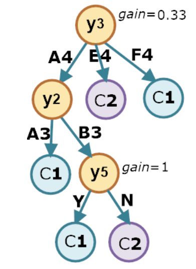
Def. 1.15 Given a set of labeled observations, a **local classification model** is a meaningful composition of decision rules inferred from regions of interest (*supervised local descriptive models*) to label new observations.

Similarly to the introduced descriptive models, the properties of the regions used within a local classifier highly depend on the input data. Illustrating, given an labeled real-valued matrix in a (n,m)-space, a decision rule $M_i(\mathbf{x})$ is inferred from a discriminative subspace $\mathbf{B}_i$, where $\mathbf{B}_i$ is a (r_i,q_i)-space where r_i<$n \wedge q_i$<m. A widely known local classifier is a decision tree, where each path from the root to the leaf defines a decision rule associated with a region with specific interestingness criteria such as high information gain.

Basics 1.11 Learning decision trees

Given a set of labeled observations, a decision tree is a local classifier that gathers class decisions from conjunctions of tests on the values of discriminative features (see Figure 1.12). Decisions trees are commonly learned by iteratively selecting a feature with the highest information gain (or lowest entropy) and splitting data accordingly for subsequent branched decisions. Given a $\mathbf{X}$ set of n observations with labels in C that are divided $\{\mathbf{X}^1,..,\mathbf{X}^{|\mathcal{Y}^j|}\}$ according to their values on feature $\mathbf{y}_j$, information gain is given by $H(\mathbf{X})$-$\Sigma_{i=1}^{|\mathcal{Y}_j|}\frac{|\mathbf{X}^i|}{n}H(\mathbf{X}^i)$ where $H(\mathbf{X})$=-$\Sigma_{c_i\in C}\frac{n_i}{n}log_2(\frac{n_i}{n})$ is its entropy. Accordingly, Figure 1.12 shows the learned decision tree for the tabular dataset $\mathbf{A}_2$ in Figure 1.4. Understandably, paths from root to leaf form a region in the original dataset given by a subset of observations respecting the accepted values and the set of tested features.

Figure 1.12: Decision tree learned from $\mathbf{A_2}$ data.



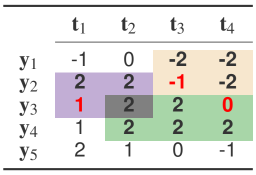
Associative classification models are a specialization of local classification models. They define a set of weighted decision rules from informative and discriminative regions, thus combining simplicity and flexibility.

Def. 1.16 Given a labeled dataset $\mathbf{A}$, an *associative model* is a composition of p weighted association rules, where each rule $\mathbf{B} \Rightarrow^{s} C$ has s-weight and maps a region of interest $\mathbf{B}$ (rule's antecedent) with a subset of classes $C \subset C$ (rule's consequent). Given $\mathbf{A}$, an **associative classification model** defines a matching criteria M to label a new observation $\mathbf{x}_{new}$ against a (possibly pre-computed) associative model learned from $\mathbf{A}$.

Basics 1.12 Learning a simplistic associative classifier

Considering the $\mathbf{A_4}$ dataset provided in Figure 1.7, 4 regions of interest that can be retrieved and mapped as the following set of rules: $\mathbf{B}_1 \Rightarrow^{s_1} \{c_1\}$, $\mathbf{B}_2 \Rightarrow^{s_2} \{c_1\}$, $\mathbf{B}_3 \Rightarrow^{s_3} \{c_2\}$, $\mathbf{B}_4 \Rightarrow^{s_4} \{c_2\}$ (see *Basics I-1.9*). From these rules, scores can be inferred s_i based for instance on the discriminative power and significance of each region ($s_1 > s_2 \wedge s_4 > s_3$). Given a new observation, the simplest way to compute the strength of each class is to sum the score of rules with matched regions. Assuming s_1=2.4, s_2=1.6, s_3=1.4, s_4=3.2, and that the decision model M considers a valid match between the newly observed region and a scored region if over 70% of the elements of these regions are approximately equal. Given the observation in Figure 1.13, we can see that 3 regions satisfy this criterion: $\mathbf{B}_1$, $\mathbf{B}_2$, and $\mathbf{B}_3$, and thus the strength of each class is $c_1 = \frac{s_1+s_2}{s_1+s_2+s_3}$=74% (the decision) and $c_2 = \frac{s_3}{s_1+s_2+s_3}$=26%.

Figure 1.13: New observation $\mathbf{x}_{new}$ for $\mathbf{A}_4$ cube.



	$\mathbf{t}_1$	$\mathbf{t}_2$	$\mathbf{t}_3$	$\mathbf{t}_4$
$\mathbf{y}_1$	-1	0	**-2**	**-2**
$\mathbf{y}_2$	**2**	**2**	**-1**	**-2**
$\mathbf{y}_3$	**1**	**2**	**2**	**0**
$\mathbf{y}_4$	1	**2**	**2**	**2**
$\mathbf{y}_5$	2	1	0	-1

Since the underlying goal of this work is to explore the impact of selecting relevant regions when learning from high-dimensional data, the focus is placed on local (descriptive and classification) models. In this context, we use the term *local model* – a composition of regions of interest – not only as the foundation for learning descriptors but also for learning decision rules.

1.1.3 Learning Function

The chosen learning method determines the properties of the learned models and thus their adequacy to answer a given problem. The quantification of the performance of descriptive and decision models is formally expressed via a loss function, L. For descriptive models, the loss function is typically given by: 1) match scores between the learned regularities against new observations or expectations (when assuming background knowledge regarding P_A) or, alternatively, by 2) merit functions that measure the coherency of the learned abstractions (in the absence of testing observations and background knowledge of P_A). For classification models, a loss function measures the incurred cost of erroneous decisions, $L(y, \hat{y})$. Although this quantification can be also seen optimistically as an utility function to measure the gain of correct decisions, its symmetric function defines a loss function and therefore we use the term loss function interchangeably. Illustrative loss functions for classification include functions based on accuracy or sensitivity metrics in the presence of nominal classes, and the normalized or root mean squared errors in the presence of ordinal classes.

Def. 1.17 Given a (labeled) dataset $\mathbf{A}$, a descriptive or classification *learning task* aims to learn a model M that achieves a specific optimality criterion with regards to a specific loss function L or set of loss functions.

The properties of the target learning models are mainly driven by the learning function. The learning function can either be *probabilistic* or *deterministic*, as well as *generative* or *discriminative*.

Probabilistic functions place assumptions on the stochasticity of observations to model their regularities $P_{\mathbf{A}}$. This is done by either fitting observations against mixtures or more structured models (Basics 1.13). The learning task thus consists of estimating the parameters of these models, and it is thus often seen as an optimization problem. Contrasting, deterministic functions typically rely on greedy or exhaustive searches to extract relevant data aspects, which can be seen as an implicit model of the true data regularities $P_{\mathbf{A}}$. Probabilistic and deterministic functions should not be confused with probabilistic and deterministic outputs of classification models. Given a set of labels C, the output or decision of a classifier is probabilistic when discloses the probability of labeling a given observation for each class in C, and deterministic when simply returns the class with higher probability.

In labeled data contexts, these probabilistic and deterministic functions can either be used to learn generative or discriminative models. In generative contexts, each class-conditional probability function $P_{X|c}$ is learned separately. Generative learning needs to be sufficiently powerful to model regularities specific to a single class for data contexts with subtle differences between class-conditional functions. Discriminative learning aims to discover meaningful boundaries that separate observations from different classes. Discriminative functions can either be derived directly from data, such as decision trees, or from generative functions by focusing on their class-conditional differences (Basics 1.14). Illustrating, the target associative classifiers typically rely on the generative learning of class-conditional regions of interest. Yet, the desirable focus on dissimilar regions between classes also requires discriminative learning. Alternatively, some learning approaches combine generative and discriminative functions via generative embeddings [344, 386]. In the context of unlabeled data, generative and discriminative learning is respectively associated with the description and differentiation of specific regions of interest. However, to preserve simplicity, this work only applies these terms in labeled data contexts.

Basics 1.13 Probabilistic models: unstructured vs. structured, generative vs. discriminative

Given $\mathbf{A_1}$ dataset, the learned multivariate Gaussian mixture: $\mu|c_1$=[0.5,-2.2,3,0.5,-2.3,0,1.3] and $\mu|c_2$=[-1,-0.1,0.2,1,0.1,-0.1,0] (see Basics 1.7) is an unstructured and generative probabilistic model. Given an observation $\mathbf{x}_{new}$, the probability of $\mathbf{x}_{new}$ being generated by each class-conditional mixture (also referred as fit) determines the strength of each class. Nevertheless, as illustrated in Figure 1.14, these class-conditional mixtures can be use as input for a discriminative function to place decisions. Considering the same $\mathbf{A_1}$ dataset, the learning function can consider more complex stochastic assumptions, possibly given by structured probabilistic models such as the model illustrated in Figure 1.15.

Figure 1.14: Discriminative decisions from multivariate Gaussian distributions learned from $\mathbf{A_1}$ data.

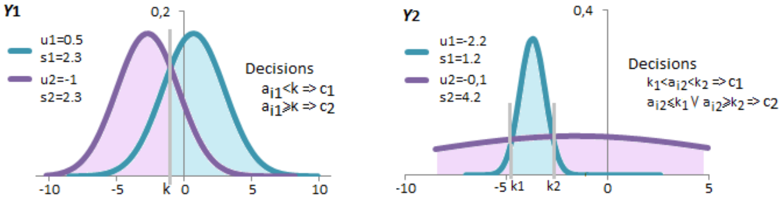


Figure 1.15: Structured generative model learned from $\mathbf{A}_1$.

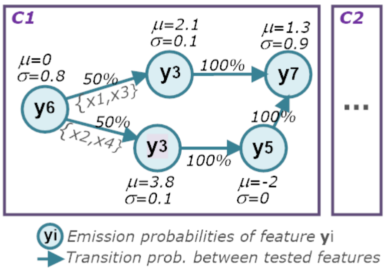
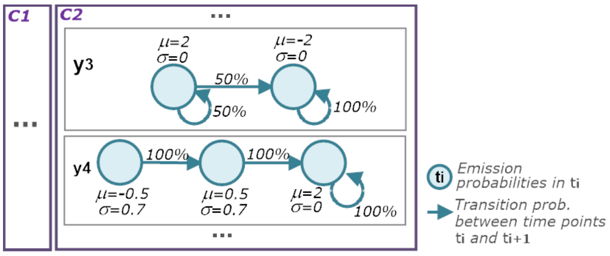


Figure 1.16: Structured generative model learned from $\mathbf{A}_4$.



Considering the $\mathbf{A_4}$ three-way time series. Similarly, both unstructured and structured probabilistic functions can be learned from $\mathbf{A_4}$. In this data context, a multivariate Gaussian mixture can be learned with parameters dependent on time (e.g. $\mu(t)|c_1$=[$\mu_{\mathbf{y}_1}(t)|c_1$, .., $\mu_{\mathbf{y}_5}(t)|c_1$]=[0.38$t^2-2.93t$+3.63, -$t$+1.75, .., -0.25$t^2$+1.55$t$-1.75] when assuming a polynomial regression) or by matrices where the Normal assumption is considered for each time point (e.g. $\mu(t_1)|c_1$=[1.0,0.5,0,0,-0.5]). For testing new observations, a generative fitting schema can be considered or discriminative decisions inferred from the underlying mixtures. Contrasting, an illustrative structured model learned from $\mathbf{A_4}$ where temporal dependencies are explicitly modeled is depicted in Figure 1.16.

Basics 1.14 Deterministic models: unstructured vs. structured, generative vs. discriminative

Given $\mathbf{A_2}$ dataset, the illustrated decision tree in Fig.1.12 is a structured and discriminative deterministic model. Given $\mathbf{A_4}$ dataset, the illustrated associative classifier in Basics 1.12 given by a set of weighted rules is an unstructured and discriminative deterministic model. Given $\mathbf{A_5}$ dataset, the c_1-conditional $d(ab)(be)f$ and c_2-conditional $a(ac)d(df)$ sequential patterns (see Basics 1.13) can be either seen as a generative model to test the fit of new observations or as a discriminative model when dissimilarity guarantees are provided. In fact, the associative models learned from $\mathbf{A}_4$ and $\mathbf{A}_5$ data require both generative and discriminative learning functions to, respectively, model class-conditional regularities and guarantee their discriminative power.

1.2 Problem Motivation

Figure 1.17 lists the major challenges and commonly applied principles to learn from high-dimensional data.

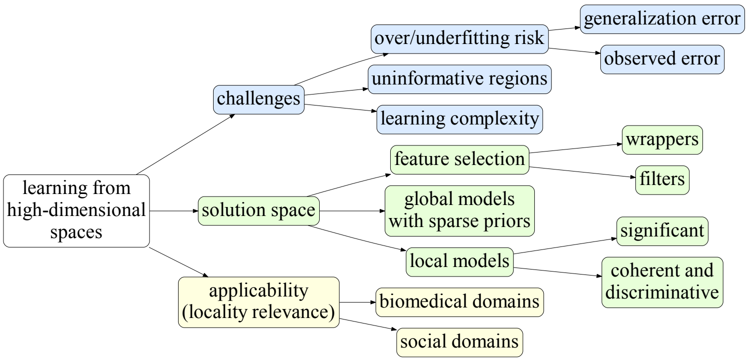


Figure 1.17: Motivating the learning in high-dimensional spaces: open challenges, contributions and applications.

A well-known challenge of learning from high-dimensional data is associated with the propensity of the resulting models to either overfit or underfit the observed data [646, 316, 173]. Minimizing this risk requires essentially an optimal trade-off between the *observed error* – error estimated from assessing the learned model on the observed data –, and the *generalization error*, often given by the mean and variability of the error estimates collected from assessing the model on new sets of observations [182]. In this context, adjusting the complexity (also referred capacity) of the learning function is necessary to achieve good generalization [26]. Complex functions guarantee a low observed error but often perform poorly on unseen data (overfitting propensity), while overly simple functions may not be able to model relevant data regularities (underfitting propensity).

However, in data contexts where the number of features exceeds the number of observations, the complexity term cannot be explored since models may not be able to generalize. This property is often referred as perfect overfitting towards the observed data [646]. To illustrate this problem, let us consider the following simplistic global model: a linear hyperplane $M(\mathbf{x})$ in $\mathbb{R}^m$ defined by a vector $\mathbf{w} \in \mathbb{R}^m$ and point b to either separate two classes, $sign(\mathbf{w} \cdot \mathbf{x} + b)$, predict a real-valued outcome, $\mathbf{w} \cdot \mathbf{x} + b$, or describe the input observations, $\mathbf{X} \sim \mathbf{w} \cdot \mathbf{x} + b$. As illustrated in Figure 1.18, a linear hyperplane in $\mathbb{R}^m$ can perfectly model up to $m + 1$ observations, either as a global classifier $\mathcal{X} \rightarrow \{\pm 1\}$, as a regression model $\mathcal{X} \rightarrow \mathbb{R}$ or as a global descriptive model of $\mathbf{X}$. Although the assessment of these models using the same observations is associated with a zero observed error, in the presence of new observations, the generalization error can be significantly high due to the risk of perfect overfitting towards the training data.

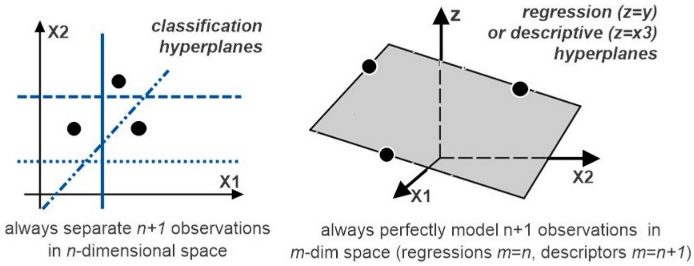

X2
classification hyperplanes
X1
always separate n+1 observations in n-dimensional space
z
regression (z=y) or descriptive (z=x3) hyperplanes
X1
X2
always perfectly model n+1 observations in m-dim space (regressions m=n, descriptors m=n+1)

Figure 1.18: Linear hyperplanes cannot generalize when the number of features (data dimensionality) is larger than the number of observations (data size), $m \geq n + 1$.

As a result of these observations, learning from specific regions of interest from high-dimensional data has been presented as an option to avoid perfect overfitting. However, assessing the statistical impact of selecting regions is critical since small regions are highly prone to be relevant by chance [343].

In tabular data contexts, regions given by a small number of features and/or observations can be highly coherent (descriptive tasks) or highly discriminative (classification tasks), yet their probability of occurrence might not be statistical significant (see Basics 1.15). This observation is also valid in structured data contexts, where a region is additionally associated with a subset of time points, item occurrences or events.

Basics 1.15 Statistical significance of regions

In tabular data contexts, a *region* is a subset of observations and features with a specific form of homogeneity. Given a tabular dataset $\mathbf{A}$, the *statistical significance* of a region defines the probability of its occurrence against a null data model to deviate from expectations. In this context, we use the term region to either describe an *observed region* in the data space, as well as an *unobserved region* (whose statistical significance can be also assessed). Given these considerations, the probability of occurrence of an observed region is non-necessarily 100% since this probability is computed against data expectations. According to Figure 1.1, we identify three red regions as not statistically significant. This is a likely condition since their low number of observations and features increases their probability to occur. *Book V* is dedicated to adequately assessing the statistical significance of regions with varying properties.

Illustrating, consider a real-valued matrix defining a (n=50,m=10000)-space with an Uniform distribution of values per feature $\mathbf{y}_j$~U(-1,1) and two balanced classes. Consider a region given by a subset of the original features, $\mathbf{B}$=($\mathbf{I}$=$\mathbf{X}$, $\mathbf{J}$⊆$\mathbf{Y}$), defining a (50,5)-space. The combination of values for the selected subset of 5 features (assuming an entropy ratio above 90% according to Basics 1.11) is highly likely to occur by chance and therefore this region is not statistically significant. Alternatively, let us neglect the labels and consider a (n=20,m=10)-space. The probability that this region $\mathbf{B}$=($\mathbf{I}$, $\mathbf{J}$) has constant values $\forall_{i\in\mathbf{I}}\forall_{j\in\mathbf{J}}a_{ij} \in [0.2, 1]$ across observations is 44% assuming a simplistic binomial calculus, $\binom{m}{|\mathbf{J}|}\sum_{x=|\mathbf{I}|}^{n}\binom{n}{x}p_{\varphi_{\mathbf{B}}}^{x}(1-p_{\varphi_{\mathbf{B}}})^{n-x}$. Again, this region is not statistically significant.

Although the selection of small regions is highly prone to be either informative or discriminative by chance, many classifiers: 1) rely on feature selection to deal with high-dimensionality, or 2) infer decisions from regions given by (possibly small) subsets of features. Illustrative classifiers with propensity towards this behavior are decision trees. Decision trees (see Fig.1.12) typically select a minimum subset of features, whose combination of values is able to discriminate a specific class. As a result, decision trees and peer classifiers show a high variable performance (when assessed from a collection of error estimates) and generalization error.

Understandably, the selection of non-significant regions is associated with the risk of underfitting the observed data. This is the tackled problem in this work since this risk is not structural, meaning that it can be minimized. For this aim, the impact of mapping an original data space into a set of regions needs to be addressed.

In addition to this problem, the selection of uninformative regions increases the learning complexity and can introduce unnecessary biases on the learned models.

1.2.1 Problems of Dimensionality Reduction

Let us further explore the facets of this problem. Three major learning options have been considered for the learning of descriptive and classification models from high-dimensional data.

First, *feature selection* methods have been applied as a filter or a wrapper. Filters select subsets of features as an independent preprocessing stage according to some measure of feature relevance, which often neglects the statistical significance of the selected spaces [682]. Wrappers can be alternatively applied to minimize this problem since they can estimate the generalization error of the model by evaluating the learned model against multiple subsets of features [216, 558]. However, minimizing the generalization error does not guarantee that the selected subsets of features are statistically significant. Additionally, wrappers degrade the learning efficiency and are dependent on the chosen model, that is, there are no guarantees that a subset of features chosen for one model is adequate for other models.

Pointers 1.16 The problem of feature extraction

A rather less-studied problem is associated with the learning from tabular data spaces with features extracted from structured data spaces (e.g. physiological signals, repositories of health-records). In order to guarantee that a reasonable set of informative features is extracted, existing studies tend to generate a wide-range of statistical, temporal and geometric features[a] [285, 397, 354]. Understandably, in the presence of a limited number of observations, an informative feature can easily be discriminative by chance. Furthermore, this problem is aggravated for feature extraction due to biases towards the extraction of purely discriminative features.

[a] Illustrative methods for feature extraction from temporal structured data include rectangular tonic-phasic windows; moving and sliding features (as moving and sliding mean and median); transformations (Fourier, wavelet, empirical, Hilbert, singular-spectrum); principal, independent and linear component analysis; projection pursuit; nonlinear auto-associative networks; multidimensional scaling; and self-organizing maps.

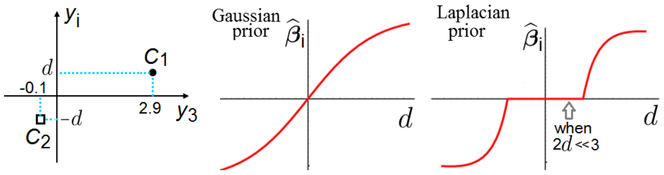
In real-valued data contexts, an alternative simplistic way of reducing the dimensionality of a given dataset is to use a mapping function, also referred as a projection or hyper-dimensional transformation, from the observed data space into a new data space with lower dimensionality $\phi : \mathbb{R}^m \rightarrow \mathbb{R}^d$ where $d<m$. An illustrative projection of $\mathbf{A_1}$ data space (Table 1.3) is $\phi(\mathbf{x_i})=\phi(a_{i1},..,a_{im})=(a_{i3},a_{i5},2a_{i7},a_{i1}\times a_{i6})$. Contrasting with feature selection, projections can affect the value distributions of features, thus often facilitating the subsequent learning task. However, even in the presence of complex mapping functions, these procedures are not able to flexibly select an arbitrary number of regions from the input data space.

Second, and complementarily to feature selection, global models can be learned using *sparse kernels*. A sparse kernel is a parametric learning function that it is able to guarantee a focus on relevant regions by placing assumptions to collapse or disregard parameters associated with uninformative regions, thus minimizing the learning complexity and fostering the model's generalization [153, 647]. Sparse kernels are often associated with (but not limited to) the learning of probabilistic models [215]. In these contexts, irrelevant and redundant parameters rapidly converge to zero [378, 214]. For this end, specific a priori knowledge regarding the probability function $P_{\mathbf{A}}$, referred as prior, have been used to promote sparsity of both unstructured and structured models for both descriptive and classification tasks (see Basics 1.17). Although the covered sparse kernels offer the possibility to discard non-informative data and to balance the over/underfitting by controlling the number of iterations associated with the learning of a parametric model, they show some inherent challenges. Since sparsity is determined by the model's parameters, it is not expressive enough to guarantee a flexible selection of regions of interest due to two major challenges. First, although recent contributions can be used to avoid the need to specify or estimate the degree of sparseness of the models [217], there is still a high complexity associated with the definition of sparse priors. Second, sparsity is primarily used to either discard non-relevant features and/or specific ranges of values per feature, thus preventing the flexible selection of subsets of both observations and features/events/time points.

Basics 1.17 Illustrative discriminative and generative sparse kernels

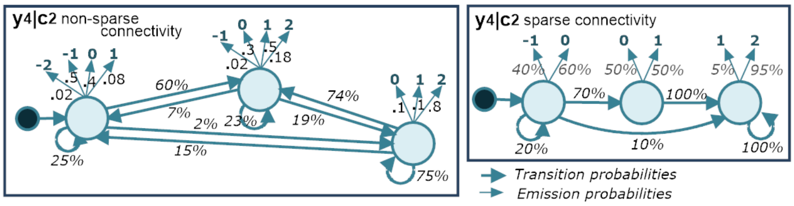
Let a support vector machine be a parametric learning function aiming to learn a hyperplane from a given feature space: $M(\mathbf{x})=\sum_{i=1}^{n} w_i x_i+b=\mathbf{w}^T\mathbf{x}+b$ where m is the dimensionality and $\mathbf{w}$ is the vector of parameters. The hyperplane can be learned to approximate observations (description and regression) or, alternatively, to separate observations with distinct labels (classification). The learning function can be applied over the original or projected feature spaces. Since not all features are equally informative, sparsity is used to guarantee that less informative features are discarded by placing a loss assumption that forces some of $\mathbf{w}$ elements to converge to zero [624]. This assumption guarantees the learning of observation models with lower generalization error and the learning of classification models with less propensity towards overfitting (hyperplane with larger margins separating observations). Given the c_1-conditional observations from $\mathbf{A}_1$ (Table 1.3), $\mathbf{w}$=[0.2,-1.3,1.7,0.2,-1.4,-0.01,0.7] defines a descriptive hyperplane where $\{w_1,w_4,w_6\}$ converge to zero with sparsity enforcement. Given the same set of c_1-conditional observations, now consider the learning of a mixture given by a vector of parameters $\mathbf{w}$ applied over the original space ($\mathbf{w}=[w_1,..,w_m]$) or projected feature space ($\mathbf{w}=[w_1,..,w_p]$ where $p\in\mathbb{N}^+$). The mixture illustrated in Basics 1.7 placed a Gaussian assumption, where $\mathbf{w}=[\mu|c_1,\sigma|c_1]$. Assuming a Laplacian (rather than a Gaussian) prior, $\mathbf{w}$ can be optimized to yield a maximum posterior estimate with certain sparseness degree by removing irrelevant and redundant parameters. The Laplacian prior sets estimates as 0 when they are non-discriminative (Fig.1.19). For the given c_1-conditional observations, $\mathbf{y}_4$ is non-discriminative and $\mathbf{y}_2$ is redundant with $\mathbf{y}_5$, thus $w_4=w_2=0$.

Figure 1.19: Laplacian priors to discard non-informative features from $\mathbf{A}_1$ dataset (charts adapted from Figueiredo [213]).



Contrasting with the previous models, now consider a structured model given by an automaton with fully-interconnected transitions and certain emissions per states. In this context, a parametric learning function can be given to learn the probabilities associated with the transitions and emissions of the underlying automaton. By placing assumptions that enforce the delineate convergence of the probability of transitions and emissions, sparse models (low number of high probable paths) can be learned. Given the y_3 feature of c_2-conditional times series from $\mathbf{A}_4$ (including $\mathbf{s}_1$=<0,1,2,2> and $\mathbf{s}_2$=<-1,0,2,2>), Fig.1.20 illustrates structured models in the absence and presence of sparse priors (assuming a mixture of Dirichlets [95]).

Figure 1.20: Impact of sparse kernels to enforce path convergence of structured models learned from $\mathbf{A}_4$ time series.



Pointers 1.18 Complementary readings on sparse kernels

Sparse kernels can be considered when learning both descriptive and decision models [216, 78]. Sparsity can be accommodated for unstructured generative models given by mixtures with varying properties [254, 217] or more structured generative models such as hidden Markov model, dynamic Bayesian networks or neural networks by placing assumption on the underlying lattice connectivity [143, 95]. Well-known ways of obtaining sparse global models include parametric functions with a Laplacian prior [232, 504, 384] or support vector machines [254, 153]. Other illustrative sparse kernels include multinomial sparse logistic regressions [78], variants of the expectation-maximization algorithm with sparse priors [505, 215], mixtures of Dirichlets [95], among others [378, 214]. In order to avoid the need to specify or estimate the degree of sparseness of the resulting models, a hierarchical interpretation of the Laplacian prior has been applied with the Jeffreys' hyper-prior [217].

Third, some learning functions infer descriptions and decisions from sets of regions of the original data space. These functions are associated with local descriptive models, such as biclustering models, and local decision models, such as associative classification models. The problem of how to guide the learning of these models to adequately select and compose regions of interest is the central task of this work. Naturally, the implications of this study can be further used to assess and extend methods for dimensionality reduction (including but not limited to feature selection) as well as to affect the learning of global models.

1.2.2 Relevance of Local Models: Applications

To further motivate the relevance of learning local descriptive models, Table 1.1 provides a set of biomedical and social data domains characterized by the presence of meaningful local regions. These data domains are characterized by a high-dimensionality associated with a high number of genes per sample, health-records per patient, molecules per biological network, time points per physiological signal, browsing actions per user, trading decisions per business, or interactions per user in social contexts.

	Data	*Illustrative subspaces with relevance for learning tasks*
Biomedical	physiological [108, 201, 211]	Sets of (sliding) features and signal partitions with coherent values across case or stimuli-elicited responses.
	clinical [302, 122]	Groups of patients with correlated clinical features or health records (shared treatments, diagnoses, prescriptions).
	structural variations [207, 324]	Correlated groups of mutations and copy number variations.
	biological networks [53]	Modules of genes, proteins or metabolites with meaningful interaction (from matrices with pairwise connections).
	gene expression [429, 312]	Groups of genes involved in functional processes and pathways only active under certain conditions.
	genome-wide [662, 640]	Conserved functional subsequences (sequence alignments), factor binding sites and insertion mutagenesis.
	other	Local regularities in translational [175], chemical [415] and nutritional data [393].
Social	social networks [257]	Groups of individuals with correlated activity and intercommunication; groups of contents based on accessors.
	text mining [38, 171]	Content-related documents and web pages (from matrices weighting categories/words across text segments).
	(e-)commerce [35]	Hidden browsing patterns containing relationships between sets of (web) users, (web) pages and operations.
	financial trading [334]	Indicators producing similar profitability for specific trading points (buy, hold and sell signals) in the stock market.
	collaborative filtering [159]	Groups of users who share preferences and behaviorial patterns for a subset of available actions.

Table 1.1: Disclosing the meaning of regions across (high-dimensional) biomedical and social data contexts.

1.3 Thesis Requirements

The underlying **hypothesis** is that *learning from relevant regions of high-dimensional data improves the performance guarantees of local descriptive models and (associative) classification models*. Naturally, testing this hypothesis leads us into the *how*. First, how does performance vary with the properties of the selected regions? Second, how can this understanding be used to improve the learning of descriptive and classification models? Figure 1.21 lists the key requirements and premises to validate the target hypothesis.

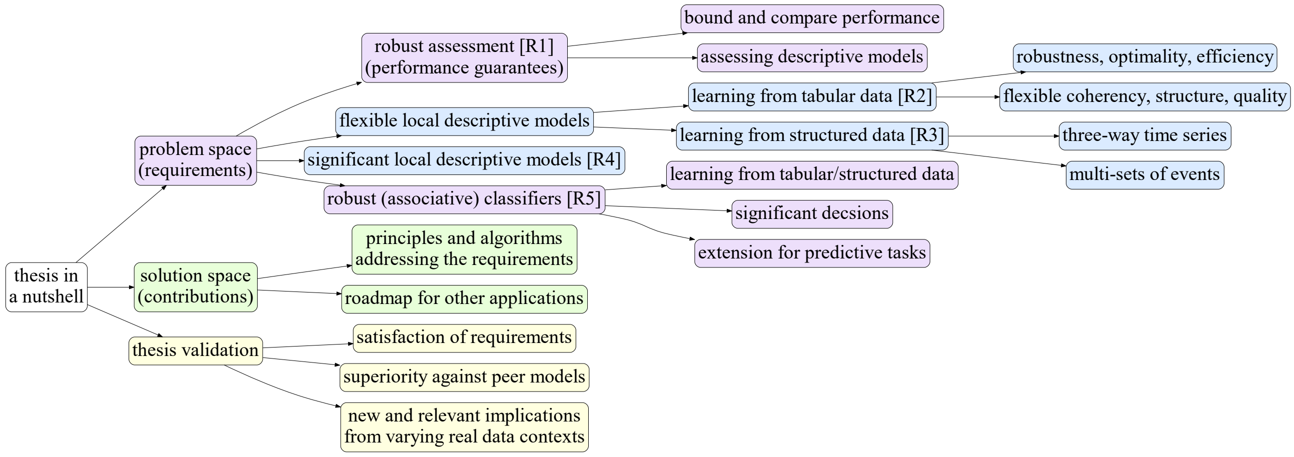

thesis in
a nutshell
problem space
(requirements)
solution space
(contributions)
thesis validation
robust assessment [R1]
(performance guarantees)
flexible local descriptive models
significant local descriptive models [R4]
robust (associative) classifiers [R5]
principles and algorithms
addressing the requirements
roadmap for other applications
satisfaction of requirements
superiority against peer models
new and relevant implications
from varying real data contexts
bound and compare performance
assessing descriptive models
learning from tabular data [R2]
learning from structured data [R3]
learning from tabular/structured data
significant decsions
extension for predictive tasks
robustness, optimality, efficiency
flexible coherency, structure, quality
three-way time series
multi-sets of events

Figure 1.21: Structured view of the thesis scope: requirements and premises to validate the underlying hypothesis.

First, in order to validate the proposed hypothesis, we decompose its assertion according to an incremental set of five major requirements. These requirements define the problem space.

R1 Robust assessment of descriptive and classification models learned from high-dimensional data.

By satisfying the first requirement, we have a systematic way to validate our hypothesis, that is, to measure and compare the impact of modeling regions with varying properties of interest on the target learning tasks.

R2 Learning of flexible and robust local descriptive models from tabular data.

The satisfaction of this requirement allows the systematic exploration of the impact that distinct biclustering models have in the ability to learn from high-dimensional data. This requires the scalable discovery of flexible structures of biclusters with parameterizable homogeneity criteria, yet offering optimality guarantees to properly assess their impact on descriptive and prescriptive tasks.

R3 Learning of flexible and robust local descriptive models from structured data.

Although the satisfaction of **R2** already covers different high-dimensional data contexts, such as matrices and network data, it excludes other data structures that are becoming increasingly relevant, such as multivariate time series, sequential databases and multi-sets of events. These learning challenges are thus specifically addressed under this requirement.

R4 Guarantee the statistical significance of local descriptive models.

To answer the introduced need to assess the impact of reducing dimensionality or selecting regions of interest (*Section 1.2*), we require the target local descriptive models to be statistically significant. Addressing this requirement implies the presence of a robust statistical assessment to guarantee that regions with varying coherence and quality (either from tabular or structured data) are not prone to occur by chance. This allows the inference of constraints based on the properties of these regions and the original data, that can be used to guide the learning.

R5 Learning effective classifiers from flexible, robust and statistically significant local descriptive models.

This requirement combines the previous focus on local descriptive models with the need to guarantee their discriminative power in labeled data contexts. Its satisfaction allows the assessment of the impact that the coherency, quality, significance and discriminative power of the selected regions have in the performance of classification models. The significance assessment of descriptive models is also extended for (associative) classification models and used to affect the learning. All the previous contributions are thus used at this point to guarantee both the accuracy and statistical significance of classification decisions. As a result, an integrative view of the pros and cons of learning local descriptive models to perform classification from distinct high-dimensional data domains is required.

Finally, this requirement is further extended in this work to guarantee the ability to learn from structured codomains given by sequences of classes for the adequate answering of predictive tasks.

Table 1.2 provides a non-exhaustive decomposition of these five structural requirements.

Table 1.2: Decomposition of the five requirements: list of the tackled requirements.

Requirement
R1: Robust assessment of models learned from high-dimensional data; **R1.1**: Performance guarantees of classification models; **R1.2**: Performance guarantees of local descriptive models; **R1.3**: Adequate generation of synthetic data for non-biased and complete assessments;
R2: Learning biclustering models from tabular data; **R2.1**: Flexible structures of biclusters with optimality guarantees; **R2.2**: Biclustering models with varying coherency: additive, multiplicative, plaid and order-preserving models; **R2.3**: Robustness of biclustering models to: 1) different forms of noise, 2) discretization and 3) missings; **R2.4**: Scalability of biclustering searches (with optimality guarantees); **R2.5**: Extension of contributions towards network data; **R2.6**: Effective and efficient learning in the presence of background knowledge; **R2.7**: Sound integration of previous contributions;
R3: Learning local descriptive models from structured data; **R3.1**: Learning cascade models from three-way time series; **R3.2**: Learning arrangements of events from multi-sets of events; **R3.3**: Stochastic modeling of structured data;
R4: Guarantee the significance of flexible local descriptive models; **R4.1/4**: Robust assessment of the statistical significance of discrete and real-valued biclusters; **R4.2**: Robust assessment of additive, multiplicative, symmetric, order-preserving and plaid models; **R4.3/7**: Robust assessment of regions with arbitrary-high levels of noisy and missings; **R4.5**: Robust assessment of cascades and arrangements of events from structured data; **R4.6**: Learning of local descriptive models from previous statistical views;
R5: Learning accurate and significant classification models from high-dimensional data; **R5.1**: Learning effective associative classifiers from tabular data; **R5.2**: Learning effective associative classifiers from structured data contexts; **R5.3**: Learning classifiers with guarantees of statistical significance; **R5.4**: Multi-period classification: extending previous contributions for the learning of sequences of classes;

Given the formulation of these requirements, our work becomes a matter of testing whether they can be simultaneously satisfied (solution space), and whether their satisfaction is associated with an improved learning in high-dimensional spaces. The thesis statement is thus asserted upon the verification of the three following sub-hypotheses: *1)* the proposed learning models satisfy the introduced requirements, *2)* these learning models offer distinctive behavior of interest against state-of-the-art learners, and *3)* their application across real-world data domains can be used to unravel new, meaningful and significant relations from data.

1.4 Solution Space

Multiple contributions resulted from addressing the introduced requirement. Contributions take two major forms: *1)* principles, and *2)* algorithms and (assessment) methodologies that rely on one or more principles. Many of these contributions are not only relevant to tackle the target problem, but can also be applied to answer other problems. In order to not compromise the line of focus of this work, the applicability of our contributions to other problems are properly identified along the text using dedicated frames (see *Notation*). As we address and answer each requirement, we expect that the proposed research produces the following scientific contributions:

- **C1.** Methods to bound and compare the performance of local descriptors and classifiers in high-dimensional data contexts, including adequate loss functions (able to measure the impact of selecting regions with varying properties), robust error estimators and generators of data for non-biased and complete assessments;
- **C2.** New descriptors of tabular data able to efficiently discover flexible structures of biclusters with optimality guarantees and robustness to varying forms of noise. Algorithms to retrieve non-constant coherencies, such as plaid and order-preserving models; to guarantee an adequate analysis of varying forms of tabular data (including network data); and to effectively incorporate background knowledge;

C3. Structured view on the increasingly relevant problems of learning cascades models from three-way time series and arrangements of events from multi-sets of events. Principles to handle the inherent complexity and variability of local responses in these data contexts, combining temporal and cross-attribute views with (possible) misalignments across observations. Deterministic and stochastic algorithms integrating these principles;

C4. Statistical views to robustly assess the significance of regions from tabular and structured data with regards to their coherency, quality and size (with upper limits on the risk of false discoveries). Revised algorithms to combine homogeneity (C2-C3) and significance (C4) views for guiding the learning;

C5. Principles for an adequate discovery (C2-C4), composition, scoring and testing of (informative and discriminative) regions from tabular and structured data. New associative classifiers able to incorporate previous principles. Principles to assess and promote the statistical significance of classification decisions. Systematic analysis of the performance impact of varying the properties of the underlying regions and learning functions across data domains. Extension of the proposed classifiers, preserving the accuracy and significance of the proposed learning functions (C5), to learn sequences of classes for predictive tasks.

Transversally to these set of major contributions, we additionally: 1) survey the contributions and limitations of state-of-the-art methods, and experimentally compare them against the proposed methods; and 2) show the relevance of the learned models to unravel significant and non-trivial relations across data domains, with a particular incidence on biomedical domains.

An integrative view of the proposed contributions of this thesis is provided in *Chapter VII-1*.

1.4.1 Scientific Dissemination

According to the introduced groups of requirements, we list below the current status of the dissemination of the contributions from our thesis near the scientific community. This list contains only peer-reviewed publications, excluding other forms of dissemination, such as invited speeches, tutoring and teaching activities, collaborations in international projects, and scientific meetings and symposiums. From the listed articles, we highlight three publications in the *Data Mining and Knowledge Discovery* journal (one describing **C4** contributions, other dedicated to a subset of **C3** and **C5** contributions, and the remaining to a subset of **C5** contributions) and additional publications in *Pattern Recognition*, *BMC Bioinformatics*, *IEEE Transactions in Computational Biology and Bioinformatics*, and *Algorithms for Molecular Biology* journals dedicated to disseminate **C2** contributions. Table 1.22 lists some of the publications proposed in the context of this thesis (see Appendix for additional published work).

1.5 Contents

The dissertation document is organized as a set of books. *Books II* to *VI* expose the core contributions of our thesis, each book tackling one of the introduced requirements. The contents within each book are carefully discussed at their start. A book is organize in chapters. Each chapter addresses a finer requirement and delivers a compact set of contributions that become available for the following chapters and books. In this way, contents are incrementally built upon previous contents, until we are able to test the cogency of the target hypothesis.

Figure 1.23 provides an illustrative view on the dependencies between books. This view supports a sound navigation through the contents provided in this dissertation.

Book II defines an assessment methodology to validate subsequent contributions. The new methods for learning flexible local descriptive models from tabular data contexts proposed in *Book III* are extended in *Book IV* towards structured data contexts, and combined in *Book V* with guarantees of statistical significance.

Book VI proposes classifiers based on the previous models and tackles the problem of guaranteeing the statistical significance of their decisions.

Finally, *Book VII* discusses the conditions on which the thesis statement is satisfied, provides an integrative view of the proposed contributions, and summarizes their major implications.

Accepted and under revision publications per book	State (July 2015)	Tackled requirements
Book II Performance Guarantees of Models Learned from High-Dimensional Data		
P1.1: R Henriques and SC Madeira, Towards Robust Performance Guarantees for Models Learned from High Dimensional Data, 2015, Chap.3, Big Data in Complex Systems, Vol.9, Studies in Big Data Series, 71-104, Springer;	Accepted	R1.1,R1.2
P1.2: BiGen: Synthetic Data Generation for Biclustering;	Under revision	R1.3
Book III Learning Local Descriptive Models from Tabular Data		
P2.1: R Henriques, C Antunes and SC Madeira, A Structured View on Pattern Mining-based Biclustering, 2015, Pattern Recognition, Elsevier;	Accepted	R2.1,R1.2
P2.2: R Henriques and SC Madeira, BicPAM: Pattern-based biclustering for biomedical data analysis, 2014, 9(1):27-, Algorithms for Molecular Biology, BioMed Central Ltd;	Accepted	R2.2,R2.3
P2.3: R Henriques and SC Madeira, Biclustering with Flexible Plaid Models to Unravel Interactions between Biological Processes, 2015, IEEE/ACM Transactions on Computational Biology and Bioinformatics;	Accepted	R2.2
P2.4: R Henriques and SC Madeira, BicSPAM: Flexible Biclustering using Sequential Patterns, 2014, BMC Bioinformatics, 15:130, BioMed Central Ltd;	Accepted	R2.2,R2.3
P2.5: R Henriques, SC Madeira and Cláudia Antunes, 2013, F2G: Efficient Discovery of Full-Patterns, In ECML/PKDD IW on New Frontiers to Mine Complex Patterns, Springer-Verlag, Prague, Czech Republic;	Accepted	R2.4
P2.6: R Henriques, C Antunes and SC Madeira, Methods for the Efficient Discovery of Large Item-Indexable Sequential Patterns, 2014, Lecture Notes in Computer Science, 100-116, Springer I.P.;	Accepted	R2.4
P2.7: R Henriques and SC Madeira, BicNET: Flexible module discovery in large-scale biological networks using biclustering, 2016, Algorithms for Molecular Biology, 11(1):1–30;	Accepted	R2.5
P2.8a: R Henriques and SC Madeira, BiC2PAM: constraint-guided biclustering for biological data analysis with domain knowledge, 2016, Algorithms for Molecular Biology, 11:23;	Accepted	R2.6
P2.8b: R Henriques and SC Madeira, Pattern-based Biclustering with Constraints for Gene Expression Data Analysis, 2015, In Comp. Methods in Bioinformatics and Systems Biology (EPIA-CMBSB), LNAI Series, Springer;	Accepted	R2.6
P2.9: R Henriques, FL Ferreira and SC Madeira, BicPAMS: Software for Biological Data Analysis with Pattern-based Biclustering, 2017, BMC Bioinformatics, 18:82;	Accepted	R2.7
Book IV Learning Local Descriptive Models from Structured Data		
P3.1: Modeling Regulatory Cascades from Gene Expression Multivariate Time Series;	Under revision	R3.1
P3.2: R Henriques, C Antunes and SC Madeira, Generative Modeling of Repositories of Health Records for Predictive Tasks, 2015, 29(4):999-1032, Data Mining and Knowledge Discovery, Springer US;	Accepted	R3.2,R3.3
Book V Significance Guarantees of Local Descriptive Models		
P4.1: R Henriques and SC Madeira, 2017, BSig: A Method to Robustly Evaluate the Statistical Significance of Biclustering Solutions, Data Mining and Knowledge Discovery, Springer US;	Accepted	R4
Book VI Learning Effective Classifiers from Local Descriptive Models		
P5.1: Learning classifiers from high-dimensional data using discriminative biclusters with non-constant coherencies;	Under revision	R5.1
P5.2: Impact of modeling statistically significant regions in the performance of classifiers;	Under revision	R5.3
P5.3: R Henriques, C Antunes and SC Madeira, Generative Modeling of Repositories of Health Records for Predictive Tasks, 2015, 29(4):999-1032, Data Mining and Knowledge Discovery, Springer US;	Accepted	R5.2
P5.4: R Henriques, SC Madeira and C Antunes, Multi-period Classification: Learning Sequent Classes from Temporal Domains, 2015, 29(3):792-819, Data Mining and Knowledge Discovery, Springer US;	Accepted	R5.4

Figure 1.22: State of scientific publications made in the context of this dissertation on January 2016.

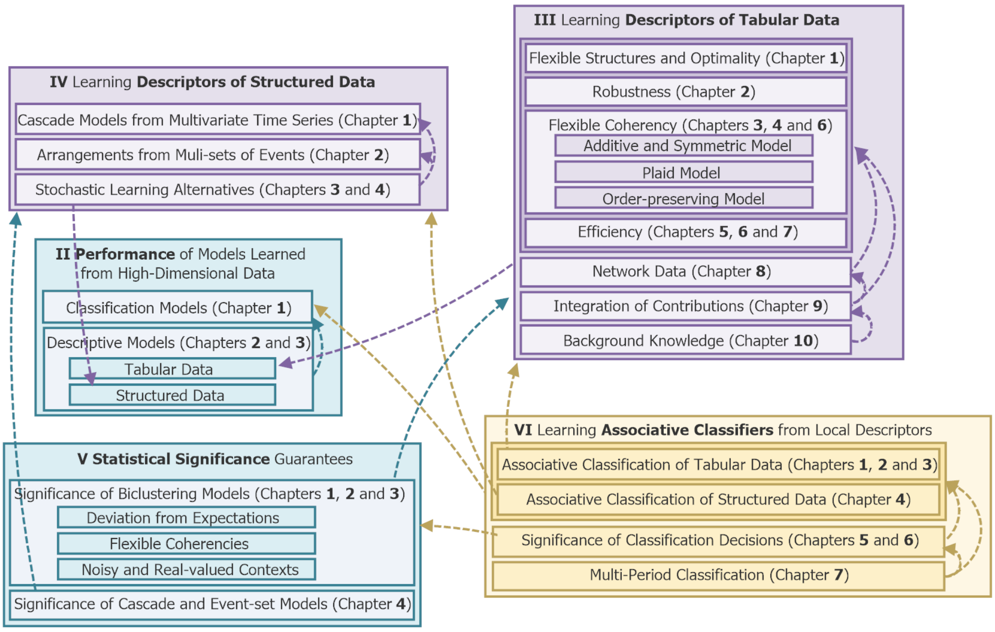

III Learning Descriptors of Tabular Data
Flexible Structures and Optimality (Chapter 1)
Robustness (Chapter 2)
Flexible Coherency (Chapters 3, 4 and 6)
Additive and Symmetric Model
Plaid Model
Order-preserving Model
Efficiency (Chapters 5, 6 and 7)
Network Data (Chapter 8)
Integration of Contributions (Chapter 9)
Background Knowledge (Chapter 10)
IV Learning Descriptors of Structured Data
Cascade Models from Multivariate Time Series (Chapter 1)
Arrangements from Muli-sets of Events (Chapter 2)
Stochastic Learning Alternatives (Chapters 3 and 4)
II Performance of Models Learned from High-Dimensional Data
Classification Models (Chapter 1)
Descriptive Models (Chapters 2 and 3)
Tabular Data
Structured Data
V Statistical Significance Guarantees
Significance of Biclustering Models (Chapters 1, 2 and 3)
Deviation from Expectations
Flexible Coherencies
Noisy and Real-valued Contexts
Significance of Cascade and Event-set Models (Chapter 4)
VI Learning Associative Classifiers from Local Descriptors
Associative Classification of Tabular Data (Chapters 1, 2 and 3)
Associative Classification of Structured Data (Chapter 4)
Significance of Classification Decisions (Chapters 5 and 6)
Multi-Period Classification (Chapter 7)

Figure 1.23: Thesis storyline: cohesive books of contents and their dependencies.

II

Performance Guarantees of Models Learned from High-Dimensional Data

Overview

Assessing the performance of descriptive and classification models is essentially a function of the selected error estimators and the properties of the input data and output model. In high-dimensional data contexts, performance is more susceptible to different sources of error. The challenges associated with learning from these data contexts were explored in *Section I-1.2*. As such, robustly assessing the performance guarantees of the learners is critical to validate and weight the increasingly available scientific statements derived from their application over real data. For this aim, this book proposes an assessment methodology, parameterized with robust error estimators, to validate the contributions of the following books.

Chapter 1 tackles the problem of bounding and comparing the performance of classification models learned from high-dimensional data. First, a set of prominent challenges is synthesized, including the need to adequately measure the over/underfitting propensity of the assessed models and test the statistical significance of the error estimator. Second, a set of principles is proposed to answer the identified challenges. These principles provide a roadmap of decisions to define robust statistical tests, to select adequate performance views and sampling schema, and to infer performance guarantees in the presence of multiple datasets and parameterizations of classifiers.

Chapter 2 extends these contributions towards the assessment of descriptive models learned from both tabular and structured data contexts. This an essential problem due to the lack of consensus on the applicable loss functions and the absence of principles to bound and compare the performance of descriptive models. In particular, the focus is placed on the assessment of biclustering, triclustering and cascade models due to their locality criteria, inherent flexibility and role in our thesis. As a result, this chapter surveys, compares and proposes loss functions to robustly assess these models in the presence and absence of knowledge regarding the underlying data regularities. New error estimators parameterized with these loss functions are proposed for an adequate assessment of their generalization error.

Despite the relevance of previous principles, they are insufficient to guarantee robust assessments when the performance of a learning function is evaluated over synthetic data with optimistic biases towards its behavior or from real data without a clear ground truth. In this context, *Chapter 3* proposes generators of synthetic data with parameterizable properties for an in-depth understanding of the behavior of the assessed methods. In particular, we provide generators of: matrix and network data with planted regions given by biclusters with varying structure, homogeneity and significance; multivariate time series with complex and diverse cascades; multi-sets of events with arrangements of informative events; and labeled (tabular and structured) data with global and local class-conditional regularities.

Index of Requirements and Contributions

Tables 1-3 provide an exhaustive listing of the tackled requirements and proposed contributions throughout this book. These tables aim to promote the traceability of the discussed contents per chapter and serve as an indexation and consultation tool, not dispensing the reading of their background provided by each chapter.

Table 1: Contributions to robustly assess *classification models* in high-dimensional data contexts (*Chapter* II-1).

R1: Robust assessment of models learned from high-dimensional data;
R1.1: Performance guarantees of classification models;
R1.1.1: Robust statistical tests to bound and compare models;
  C1.1.1a: Pairwise and multiwise comparisons with varying levels of conservatism and computational complexity;
  C1.1.1b: Confidence intervals sensitive to (high) error variability, weighted by biasedness factors;
  C1.1.1c: Adequate loss functions with possible smoothing factors for probabilistic models and/or outputs;
R1.1.2: Measure propensity towards under/overfitting;
  C1.1.2: Decomposition of performance in bias and variance components (understand generalization);
R1.1.3: Guarantee the significance of error estimators;
  C1.1.3a: Statistical tests of the feasibility of error estimates against loose settings given by null data or null models;
  C1.1.3b: Adequate sampling schema (binomial tests to fix optimum training-testing split);
R1.1.4: Understand impact of data size and dimensionality;
  C1.1.4a: Fitted curves for extrapolation of performance guarantees as a function of data size and dimensionality;
  C1.1.4b: Parameter-dependent guarantees from the learned model (discriminant analysis and VC) and data properties;
R1.1.5: Performance guarantees from multi-data and multi-parameter assessments;
  C1.1.5a: Summarization, (non-linear) regressions, visualization;
  C1.1.5b: Generalized guarantees from joint analysis of estimates or learned models;
R1.1.6: Extensibility towards imbalanced data contexts;
  C1.1.6a: Principles for generating data with varying properties (inc. global and local regularities and distinct sources of noise);
  C1.1.6b: Adequate loss functions and estimators for imbalanced class-conditional representativity and complexity;

Table 2: Contributions to robustly assess *local descriptive models* in high-dimensional data contexts (*Chapter* II-2).

R1: Robust assessment of models learned from high-dimensional data;
R1.2: Performance guarantees of descriptive models;
R1.2.1: Bounding and comparing models' performance;
R1.2.1.1: Robust estimators (collection of error estimates);
  C1.2.1.1a: Sampling, randomization, peer data and multi-performance views (real data);
  C1.2.1.1b: Instantiation with preserved properties (synthetic data);
R1.2.1.2: Extension of requirements R1.1-R1.7 for descriptors;
  C1.2.1.2: Extensibility of C1.1 contributions (including smoothing factors for generative descriptors and feasibility tests);
R1.2.2: Adequate loss functions for (local) descriptors;
R1.2.2.1: Robust metrics for biclustering models (tabular data);
R1.2.2.1.1: Effective similarity scores (synthetic data);
  C1.2.2.1.1a: Comparison of clustering views (purity, entropy, recall, precision), match scores (subspace clustering, Jaccard-based, consensus);
  C1.2.2.1.1b: New integrative scores;
R1.2.2.1.2: Effective merit functions and domain-driven scores to assess relevance (real data);
  C1.2.2.1.2a: Principles on how to use merit functions;
  C1.2.2.1.2b: Functional enrichment tests and applicable corrections from knowledge bases, semantic sources and literature;
R1.2.2.2: Robust metrics for local descriptors learned from cube data;
R1.2.2.2.1: Effective metrics for triclustering models;
  C1.2.2.2.1a: Biclustering views and extensions towards integrative cube data;
  C1.2.2.2.1b: Time-sensitive similarities and merit functions;
R1.2.2.2.2: Effective metrics for cascade models (module/causality-centric and integrative);
  C1.2.2.2.2a: Module-centric and causality-centric scores;
  C1.2.2.2.2c: New integrative scores inspired on sequence alignments;

Table 3: Contributions on the generation of synthetic data for non-biased and complete assessments (*Chapter* II-3).

R1: Robust assessment of models learned from high-dimensional data;
R1.3: Adequate synthetic data for non-biased and complete assessments;
R1.3.1: Effective generator of tabular data;
  C1.3.1.1a: Structured view on the properties of biclustering models;
  C1.3.1.1b: Matrix data with parameterizable size, background distributions and planted regions with varying number, shape (including extent of differences in size within a single data), positioning, coherency strength, and quality;
  C1.3.1.1c: Effective generation of different forms of coherency assumption (constant, additive, multiplicative, symmetric, order-preserving) and penalization factors on their size to guarantee their statistical significance;
  C1.3.1.1d: Generation of plaid structures able to model complex overlapping effects between groups of biclusters;
  C1.3.1.2: Generation of network data (homogeneous or heterogeneous with weighted or labeled interactions) with parameterizable density, distributions of edges per node, distributions of interactions' score, and planted modules;
  C1.3.1.3: Benchmark datasets preserving the statistical properties of experimental biological data;
R1.3.2: Effective generator of structured data;
R1.3.2.1: Generation of three-way time series;
  C1.3.2.1a: Procedures to generate three-way time series with varying properties, including diverse cascades with varying distributions on the number and duration of modules, their dependencies, support and coherency;
  C1.3.2.1b: Simulation of stochasticity: adequate generation of: 1) temporal and structural misalignments, and 2) bifurcations;
R1.3.2.2: Generation of multi-sets of events;
  C1.3.2.2: Extension of sequential databases to plant timestamps, explore varying sparsity, and create a multiplicity of attributes;
R1.3.3: Effective generator of labeled data for classification;
  C1.3.3a: Procedures to generate tabular and structured data with parameterizable number and imbalance of classes;
  C1.3.3b: Combined global and local class-conditional regularities (regions effectively planted according to varying discriminative criteria);

1
Performance Guarantees of Classification Models

A major challenge in machine learning is to guarantee that the learned relations from high-dimensional data are statistically significant, that is, they are not learned by chance. This is particularly important when the number of observations is not substantially larger than the number of features, as well as when the learned relations are inferred from (possibly small) regions of the original data space that are informative by chance. In particular, learning from high-dimensional data with a limited number of observations is typically subjected to different forms of biases, leading to a deterioration of the performance of classification models for new observations. In this context, despite the relevance of assessing the statistical significance of the learned relations, this is insufficient since a good significance does not necessarily imply good performance. Based on the observation that the variability of performance and the statistical significance of classification models are (inversely) correlated, some studies provide statistical tests to bound or compare the performance of classifiers in an attempt to offer more adequate assessments, while others explicitly study the effects of the data size and dimensionality on the performance of classification models [357, 348, 553, 3, 646, 471, 333, 180, 669, 276]. However, an integrative view of their potentialities and limitations is still lacking. Some of the most prominent challenges of existing contributions are five-fold. First, commonly applied error estimators are often inadequate due to the placed choices regarding the test sample size and the selected loss functions [55]. Second, the attempts to generalize the performance of classification models based on fitted learning curves and simulated surfaces fail to provide robust statistical guarantees since the performance is not affected by the variability of the observed errors [471, 669]. Third, there is a generalized absence of principles to perform robust assessments from synthetic data. The available studies often ignore the fact that real-world data exhibit complex mixtures of global and local regularities. Fourth, most of the existing contributions are not prepared to adequately deal with high-dimensional data with numerous classes with imbalance. Finally, there are not yet consensus on how the error is explained by the overfitting and underfitting components of the learned model.

This chapter addresses these challenges. For this aim, it identifies the major requirements to assess the performance guarantees of classification models and surveys critical principles for their adequate satisfaction. In particular, we rely on existing contributions and on collected empirical evidence to derive these structural principles. Additionally, their integration through a new methodology is discussed. We extend the target methodology to accommodate both data-independent and data-dependent assessments. In this context, the proposed assessment methodology offers three new critical contributions to the big data community:

- integration of statistical principles to provide a solid foundation for the definition of robust estimators of the true performance of classification models learned in high-dimensional spaces, including adequate loss functions, sampling schema (or parametric estimators), statistical tests and strategies to adjust performance guarantees in the presence of high variance and bias of performance;
- robust performance views to measure the propensity of classifiers towards over- and under-fitting;
- inference of general performance guarantees for classifiers tested over multiple datasets with varying regularities and under multiple parameterizations.

Figure 1.1 synthesizes the major challenges and the proposed contributions to infer robust performance guarantees.

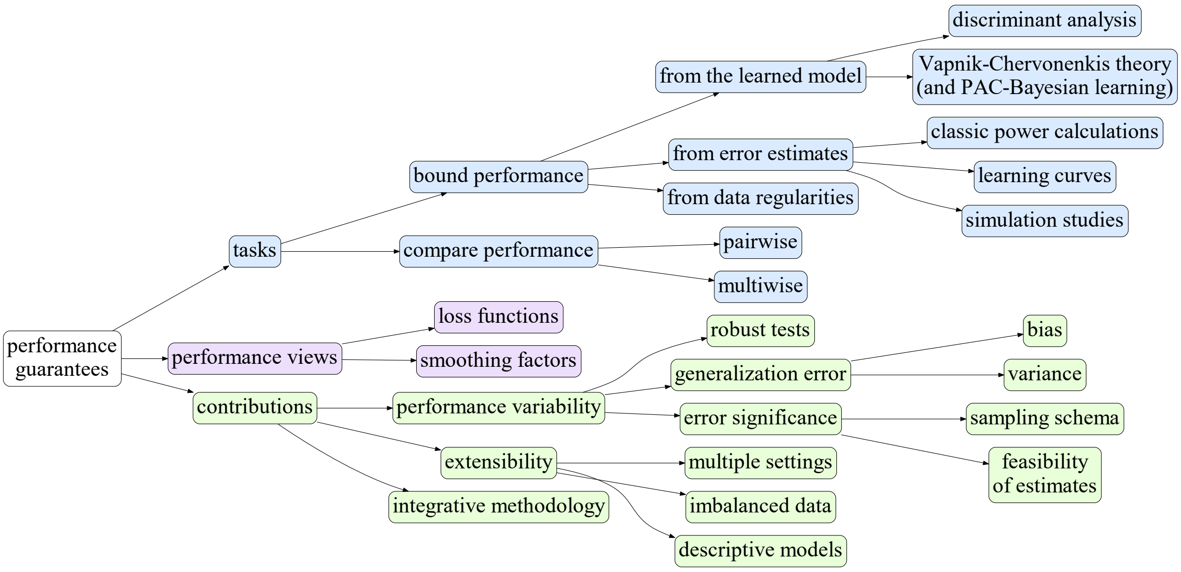


Figure 1.1: Challenges and contributions for assessing classification models in high-dimensional data contexts.

Although the focus is on bounding and comparing the performance of models in high-dimensional datasets, the proposed principles can be complementary used for two other purposes. First, as a means to investigate and adapt the behavior of the classifiers. Second, to fix the minimum number of observations that enables the learning of models with low performance variability. This task, often referred as minimum sample size estimation, is useful to avoid the expensive labor of obtaining or labeled observations in biological, clinical and some social settings.

Basics 1.1 Revisiting the applications of the target assessment

In Table 1.1, we listed high-dimensional data domains where the number of (class-conditional) observations often does not exceed the number of features, including omic data (e.g. expression data, structural genomic variations, biological networks), clinical data, collaborative filtering data and random fields, as well as data with a high number of features either extracted from unstructured data (including text and web-based sources) or mapped from multi-dimensional databases. The disparity between data size and dimensionality is particularly critical in biomedical domains, due to the scarcity of monitored patients with certain conditions and the associated costs of collecting biological samples and clinical tests. Due to the inherent difficulty of learning from these data contexts, robust assessments are essential to understand the ability of classifiers to answer different biomedical problems (including discrimination of tumor samples, disease prognosis, prediction of healthcare needs, proteomic mass spectral classification, chemosensitivity prediction, survival analysis, among others) [316].

This chapter is organized as follows. *Section 1.1* provides the background required for the definition and comprehension of the target task. *Section 1.2* surveys research streams with important contributions for this task, covering their major challenges. *Section 1.3* introduces a set of key principles derived from existing contributions to address the identified challenges. These are then coherently integrated within a simplistic assessment methodology. *Section 1.4* discusses the relevance of these principles based on experimental results and existing literature. Finally, concluding remarks and future research directions are synthesized. Performance guarantees are complemented in *Books V* and *VI* with statistical significance guarantees.

1.1 Background

The first requirement of this thesis is to robustly evaluate models learned from high-dimensional spaces. In this context, consider that the asymptotic probability of misclassification of a particular classification model M is given by ε_{true}, and a non-biased estimator of the observed error for a sample set of observations is given by $\theta(\varepsilon_{true})$. The problem of computing the *performance guarantees* of a specific model M on a dataset can either be given by its performance bounds or by the verification of its ability to perform better than other models. The task of computing the $(\varepsilon_{min}, \varepsilon_{max})$ *performance bounds* for a M model learned from a data set with n observations and m-dimensionality, can either be derived from the learned regularities $P_{X|Y}$ or from a collection of error estimates:

$$[\varepsilon_{min}, \varepsilon_{max}] : P(\varepsilon_{min} < \theta(\varepsilon_{true}) < \varepsilon_{max} \mid n, m, M, P_{X|Y}) = 1 - \delta, \tag{1.1}$$

where the performance bounds are intervals of confidence tested with 1-δ statistical power.

The task of *comparing a set of models* $\{M_1, .., M_l\}$ for a given dataset can be defined as the discovery of significant differences in performance between the models while controlling the family-wise error, the probability of making one or more false comparisons among all the $l \times l$ comparisons.

Defining an adequate estimator of the true error $\theta(\varepsilon_{true})$ for a target $(n, m, M, P_{X|Y})$ setting is, thus, the central task to guarantee robust assessments.

Let us assume that ε_{true} can be theoretically derived from assumptions regarding the data regularities $P_{X|Y}$ or experimentally approximated by testing its asymptotic behavior when generating new observations according to $P_{X|Y}$. Under this assumption, we can further test the minimum number of observations required to adequately learn from a m-dimensional data space. This can be performed by comparing the estimated error for n observations with the true error, $min_n : P(\theta_n(\varepsilon_{true}){<}\varepsilon_{true} \mid m, M, P_{X|Y}) > 1\text{-}\delta$, rejected at α, and by possibly allowing relaxations $\theta_n(\varepsilon_{true}){<}(1+\gamma)\epsilon_{true}$ when the observed error does not rapidly converge to ε_{true}, $\lim_{n\to\infty} \theta_n(\varepsilon_{true}) \neq \varepsilon_{true}$. For example, γ=0.1 ensures that the expected probability of correct classification will be within 10% of the best possible model (residual error). This is often important when there is a high similarity between the approximated class-conditional regularities from the available observations, so that class membership cannot be determined with confidence [382].

When the number of features exceeds the number of observations, the learned model can be prone to large biases as a result of the (possibly) applied dimensionality reduction procedures, complexity term and significance guarantees of the selected learning function. A coverage of these biases was done in *Book I*. In the presence of error estimates, estimators such as the mean and q-percentiles are not able to measure these underlying biases, while t-Student tests and peer statistical test become hardly meaningful in the presence of large biases due to the high variability across error estimates. This is the reason why new estimators need to be defined to deal with these structural challenges and further measure different aspects (including over/underfitting propensity) associated with the performance of classifiers in high-dimensional data contexts.

1.2 Related Work: Limitations and Contributions

In this section we, first, describe the major streams of research to assess the performance of classifiers as a function of the available data size (a key factor when learning from high-dimensional data). Second, we identify the major requirements of the target task and identify the major drawbacks of each stream to answer them. Finally, we provide an initial view on how these drawbacks can be satisfied when combining available contributions.

Bounding Performance. We first survey *estimators* to bound performance as a function of the data size and dimensionality. A synthesized view of their limitations and contributions is provided in Tables 1.1, 1.2 and 1.3. We grouped existing efforts according to six major streams of research: classic statistics, model-driven analysis, data-driven analysis, learning curves, simulation studies, and risk minimization theory. Existing estimators have their roots on, at least, one of these research streams.

Estimators from classic statistics are either centered *power calculations* [3], *deviation bounds* [283], *asymptotic estimators* of the true error ε_{true} from approximation theory, information theory and statistical mechanics [553, 492, 502], among others [348]. Illustrating, power calculations provide a critical view on the model errors (performance) by controlling both sample size n and statistical power $1{-}\gamma$, $P(\theta_n(\varepsilon_{true}) < \varepsilon_{true}){=}1{-}\gamma$, where $\theta_n(\varepsilon_{true})$ can either rely on a frequentist view, from counts to estimate the informative or discriminative power of features from a given dataset, or on a Bayesian view, more prone to deal with smaller and noisy data [3]. Here, the impact of the data size in the observed errors is essentially dependent on the entropy associated with the target (n, m)-space, which can be drawn from biased assumptions when the space does not satisfy: $n \gg m$.

Alternatively, estimators of true performance can be derived from a direct *analysis of the learning models* [644, 200, 553, 552]. This is commonly accomplished for classifiers based on discriminant functions (such as Euclidean, Fisher, quadratic or multinomial) that are able preserve the dimensionality of data (whether described by the original m features or by a subset of the original features). An analysis of how the estimated error deviates from the true error as a function of the data size n, dimensionality m and discriminant functions M was initially provided in [553] and more recently extended [105, 112]. Although robust estimations have been provided for these classifiers, they only represent a small subset of classifiers.

Contrasting, performance estimators independent from the learned model can be derived from a direct *analysis of the input data* [179]. Dobbin and Simon [180, 179] survey the impact of dimensionality, class prevalence, standardized fold change and non-trivial sources of errors in ability to learn from high-dimensional data. The problem with the principles provided by this stream of research is that the model-independence assumption forbids its extension for comparisons, as well as the computation of non-loose performance bounds.

To extrapolate the assessment of performance guarantees for an unknown sample size n, learning curves [471, 219], simulation studies [333, 669], and theoretical analyzes [646, 26] have been proposed. *Learning curves* typically approximate inverse power-law curves from samples of the observed data in order to extrapolate performance as a function of the data size and (possibly) dimensionality [471, 87]. Although extensions have been proposed to weight estimations according to their confidence [219], existing methods neglect the variability across error estimates. *Simulation studies* infer performance guarantees by generalizing the impact of multiple parameters on the learning performance [333, 669, 276]. This is often accomplished through the analysis of surfaces of error estimates collected from multiple datasets and/or parameterizations. Simulations simply aims to assess major variations in performance across different settings. In this context, statistical assessment of the inferred implications is often absent. Furthermore, both the tasks of fitting curves and of analyzing complex simulation surfaces require the data to have a high number of observations ($n > m$), which may not always be available.

Theoretical analysis of empirical risk minimization [645, 26] has been applied to find an optimal trade-off between the observed error and generalization error (see Figure 1.2). Core contributions from this research stream have their roots in Vapnik-Chervonenkis (VC) theory [646], where the sample size and dimensionality is related through the VC-dimension, a measure of the model capacity that defines the minimum number of observations required to generalize the learning in a m-dimensional space, in line with the problem illustrated in Figure 1.18. Although the VC-dimension can be theoretically or experimentally estimated for an input learning model to bound performance, the bounds are loose (conservative).

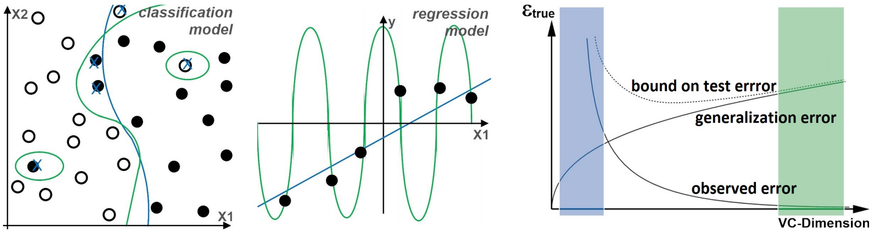


Figure 1.2: Capacity and training error impact on true error estimation for classification and regression models.

Summing, the surveyed efforts to bound the performance of models have their roots on, at least, one of these major streams of research. These streams of research assess the performance significance of a single learning model as a function of the available data size, which is a key factor when learning from high-dimensional spaces since the fewer the observations, the loose the bounds.

Since these six research streams are closely related, they can be mapped through concepts of information theory. Haussler et al. [300] proposed the first attempt to bridge contributions from statistical physics, approximation theory, multivariate analysis and VC theory within a Bayesian framework.

Comparing Performance. The task of *comparing a set of models* $\{M_1, .., M_l\}$ in a (n, m)-space can be defined as the discovery of significant differences in performance between l>1 models while controlling the family-wise error (the probability of making one or more false comparisons among all the $l \times l$ comparisons). For this end, the previously introduced estimators can be used to define robust comparisons. Alternatively, when comparing pairs of classifiers, the traditional t-Student, McNemar and Wilcoxon tests can be used directly from the collected error estimates. Friedman tests with the corresponding post-hoc tests [166] can be used for either comparing l>2 models either learned from multiple datasets or with different parameterizations. However, Friedman tests rely on pairwise Nemenyi tests that are conservative and thus only reveal highly significant differences among models. Less conservative tests have been also proposed [236]. These pairwise and multi-wise tests can either use the collected error estimates or stable performance bounds using the previously surveyed methods (preference due to the high-variability of estimates in high-dimensional spaces).

Selecting Performance Views. Finally, in order to bound and compare classifiers, a performance view needs to be specified, as well as the sampling schema if the estimator is not directly derived from the learned model or data. The performance is highly determined by the chosen *loss function* and applied smoothing factors. *Smoothing factors* can be considered to accommodate uncertainty factors for models with probabilistic outputs including (weighted error counting [256]) and in the presence of the class-conditional distributions (posterior probability estimates [408]). These smoothing factors provide more realistic performance views. The downside of weighted counting is its dependence on the error distance function, while posterior probabilities tend to be biased for small datasets.

The selected *sampling schema* can largely impact the assessments in high-dimensional data spaces [464, 186, 630]. Assuming that each error estimate is obtained from training the model on a set of observations and testing the model on independent observations, it is of critical importance to guarantee an adequate split of training-testing observations or, alternatively, estimate the minimum number of testing observations that minimizes the systematic (bias) and random (variance) uncertainty [553, 55].

Challenges. Tables 1.1 and 1.2 provide a *structured view on the existing challenges* to answer **R.1**. Understandably, the surveyed streams of research suffer from critical drawbacks since they were originally developed with a different goal – either minimum data size estimation or performance assessment in data spaces where $n \gg m$. Table 1.1 details these drawbacks according to three major requirements that define their ability to: *A)* rely on robust statistical assessments, *B)* extend the target assessment towards descriptive models and for models learned from imbalanced data, and *C)* deliver performance guarantees from multiple datasets and parameterizations. Although the surveyed challenges are lengthy in number, many of them can be answered recurring to contributions provided in literature.

Requirement	*Challenges*
A. Robust Assessments for High-Dimensional Data	1. Common estimators are not robust as they are not able to: *1)* handle the heightened variability of error estimates, and *2)* assess the impact of selecting regions of interest; 2. Inadequate assessment of the estimator's bias [333] and significance (typically tested against very loose settings) [471]; 3. Inadequate performance views. Loss functions to decompose errors and smoothing factors are commonly disregarded; 4. Inappropriate sampling scheme [55]. Assessing the variance of estimations within and across folds, and the impact of the number of folds and test sample size is critical to tune the level of conservatism of performance guarantees; 5. Others: estimators from multivariate model analysis [553] are only applicable for a small subset of learning models; data-driven estimators (originally proposed for minimum data size estimation [179]) are hardly extensible to deliver performance guarantees; theoretical analysis of the learned models are associated with conservative guarantees [645].
B. Extensibility	1. Assessment principles not easily extensible for descriptive models, including (single-class) global and local models; 2. Inadequate statistical assessment of models learned from datasets with heightened unbalance among classes and non-trivial conditional distributions $P_{X\|y_i}$. Weaker guidance for computing bounds for multi-class models ($\|\Sigma\|> 2$).
C. Data Flexibility	1. Performance guarantees can hardly be generalized as they are assessed in the context of a specific dataset; 2. Inappropriate assessments on synthetic data. Generated data should mimic real-world data, including mixtures of distributions with local regularities, dependencies among features, skewed features and varying levels of noise; 3. Data regularities are not used to better frame learning challenges and guarantees. The impact of modeling sources of variability (such as pooling, dye-swap samples and replicates in biomedical domains) is commonly disregarded [179]; 4. Lack of criteria to derive guarantees from settings where the impact of numerous parameters is studied [333, 669].

Table 1.1: Challenges for studying the performance guarantees of models learned from high-dimensional data.

Approach	*Major limitations* (non-exhaustive observations)
Bayesian and Frequentist	Originally proposed for the estimation of minimum data size and, thus, not tuned to deliver performance guarantees; Impact of feature selection is not assessed; No support as-is for descriptive tasks and hard data settings; Applied over a single dataset.
Theoretical Methods	Delivery of loose/conservative guarantees; Learning aspects need to be carefully modeled; Guarantees are often independent from the data regularities (although size and dimensionality are always considered); No support as-is for descriptive tasks.
Learning Curves	Unfeasible for small datasets or high-dimensional spaces where $m>n$; Dimensionality and the variability of errors does not explicitly affect the curves; Guarantees suitable for a single dataset; No support as-is for descriptive tasks.
Simulation Studies	Driven by error minimization and not by the statistical significance of performance; Data often rely on simplistic conditional regularities (optimistic data settings); Poor guidance to derive decisions from results.
Multivariate Analysis	Limited to (global) mixture models; Different models require different parametric analyzes; Data often rely on simplistic conditional regularities; No support as-is for descriptive tasks and hard data settings; Approximations can lead to loose bounds.
Data-driven Analysis	Not able to deliver performance guarantees (model-independent); Estimations only robust for specific data settings; Independence among features is assumed; Suitable for a single inputted dataset; Unfeasible for small samples.

Table 1.2: Limitations of existing approaches according to the introduced challenges

Contributions. Table 1.3 lists contributions that can be used to satisfy the identified limitations. This analysis triggers the need to identify sets of principles per requirement and to analyze whether it is possible or not to integrate these principles to robustly bound and compare performance. Motivated by this analysis, the solution space (*Section 1.3*) offers a structured view on these principles. To maintain conceptual clarity, the principles for the robust assessment of descriptive models are studied in a separate chapter (*Chapter 2*).

Requirements		Contributions
A.	High-Variable Performance	VC theory and discriminant analysis [645, 553]; Statistical tests for uncertain distributions [441, 539, 166].
	Over/underfitting Risk	Bias-Variance decomposition of the error [182].
	Impact of Subspace Selection	Unbiasedness principles from feature selection [589, 343].
	Expressive Loss Functions	Error views in machine learning [256, 408, 516].
	Adequate Sampling Schema	Sampling principles [186, 630]; Test-train splitting impact [55, 553].
	Feasibility	Significance of estimates computed against non-loose baseline settings [3, 471].
B.	Descriptive Models	Adequate selection of loss functions and collection of error estimates [429, 296].
	Unbalanced/Difficult Data	Adequate loss functions (e.g. sensitivity) and class-sensitive sampling [276, 55].
	Multi-class Tasks	Integration of class-centric performance bounds [55].
C.	Flexible Data Settings	Simulations on datasets with hard mixtures, local dependencies and noise [669, 333, 276, 429].
	Dimensionality Effect	Extrapolate guarantees by sub-sampling features [471, 276].
	Retrieval of Data Regularities	Data regularities to contextualize assessment [180, 553].
	Advanced Data Properties	Modeling non-trivial sources of variability [179].
	Multi-parameter Analysis	Weighted optimization methods for integrating multiple guarantees from simulations [167].

Table 1.3: Contributions with potential to satisfy the target set of requirements.

1.3 Solution: Principles to Bound and Compare Performance

Motivated by the surveyed contributions to tackle the existing limitations, this section focuses on principles to define robust estimators to assess the performance of classifiers learned from high-dimensional data. *Section 1.3.1* proposes principles to answer to the introduced challenges. Finally, *Section 1.3.2* shows that these principles can be consistently and coherently combined within a simplistic assessment methodology.

1.3.1 Principles for Robust Assessments

In this section we incrementally introduce principles to: **1)** define estimators to bound and compare classifiers able to deal with high error variability; **2)** understand the over/underfitting components of the error; **3)** place adequate sampling schema and loss functions; **4)** guarantee the feasibility of the collected error estimates; **5)** enable the inference of performance guarantees from multiple datasets and parameterizations; **6)** guarantee adequate assessments from real and synthetic data; and **7)** deal with imbalanced data.

Variability of Error Estimates (Performance Bounds). Increasing the dimensionality m for a fixed number of observations n introduces variability in the performance of the learned model that must be incorporated in the estimation of performance bounds. In this context, an adequate estimator of the true error of a classification model M is the central point of focus. An illustrative simplistic estimator is the average of a collection of observed error estimates obtained under a k-fold cross-validation: $E[\theta(\varepsilon_{true})] \approx \frac{1}{k}\Sigma_{i=1}^{k}(\epsilon_i \mid M, n, m, P_{X|Y})$, where ϵ_i is the error for the i^{th} fold. When the number of observations is not significantly large, the errors can be collected under a leave-one-out scheme, where $k{=}n$ and the ϵ_i is, thus, simply given by the application of a loss function L over a testing observation: $\epsilon_i{=}L(M(\mathbf{x}_i), c_i)$. In this context, finding performance bounds may rely on non-biased estimators from the collected error estimates, such as the q-percentiles, to provide a bar-envelope around the mean estimator (e.g. $q \in \{20\%, 80\%\}$). However, such option does not effectively consider the variability of the observed errors. A more robust alternative is to compute confidence intervals for the expected true performance from error estimates $\{\epsilon_1, .., \epsilon_k\}$ obtained from k train-test partitions by fitting an underlying distribution that is able to adequately model their variance. Despite the inherent simplicity of this strategy, it suffers two major problems[1]. First, it assumes that the variability is well-measured for each error estimate. This is commonly not true as each error estimate results from averaging a loss function across testing observations within a fold, which smooths and hides the true variability. Second, when the variance across estimates is substantially high, the resulting bounds or comparisons rapidly become not meaningful.

To address these problems, four complementary strategies derived from existing research are proposed: one for assessing classifiers able to preserve the original dimensionality, another for correcting performance guarantees for models learned from regions of the original dataset, a third strategy to reduce variability in $m \gg n$ settings, and a final strategy to obtain performance guarantees less dependent on the observed data regularities.

First, in the presence of multivariate models, their discriminant properties can be used to approximate the observed error $\theta_n(\varepsilon_{true} \mid m, M, P_{X|Y})$ and the asymptotic estimate of the true error $\lim_{n\to\infty} \theta_n(\varepsilon_{true} \mid m, M, P_{X|Y})$ [644]. An analysis of the deviations of the observed error from the true error as a function of data size n, dimensionality m and discriminant functions M was initially provided by Raudys et al. [553] and recently extended [105, 112].

Second, the unbiasedness principle from feature selection methods can be considered to affect performance guarantees. Learning models M that rely on decisions over subsets of features either implicitly or explicitly use a form of feature selection driven by core metrics, such as Mahalanobis, Bhattacharyya, Patrick-Fisher, Matusita, divergence, mutual Shannon information, and entropy. In this context, statistical tests can be made to guarantee that the value of a given metric per feature is sufficiently better than a random distribution of values when considering the original dimensionality [589, 343]. These tests return a p-value that can be used to weight the probability of the selected set of features being selected by chance and, consequently, to affect the performance bounds and the comparisons' confidence of the target models. Singhi and Liu [589] formalize selection bias, analyze its statistical properties and how they impact performance bounds.

Third, when error estimates are collected, different methods have been proposed to adequately explore the (possibly high) variability across estimates [544, 350], ranging from general principles related with sampling schema and error estimators, as well as specific principles to dedicatedly assess the true error variability for specific data (such as biological data with replicates). These options are revised with more detail in the next subsections.

Fourth, conservative bounds for a given dimensionality can be retrieved from the VC-dimension (capacity) of a classifier [645, 76]. The VC-dimension can be obtained either theoretically or experimentally [648]. A common experimental estimation option for the VC-dimension is to study the maximum deviation of errors among independently labeled datasets. An illustrative lower-bound for the estimator of the true performance of a M model composed by h mapping functions (number of decision rules based on the possible values of the m features) is: $\theta_n(\varepsilon_{true}) \geq \frac{1}{n}(log\frac{1}{\delta} + logh)$ [26], where δ is the statistical power[2]. h tends to be larger for models with higher capacity, which can degrade performance bounds in the presence of a low number of observations. For more complex models, such as Bayesian learners or decision trees, the VC-dimension can be used with assumptions that lead to

[1] A estimator addressing these problems may still not reflect the true performance bounds of a classifier due to poor sampling and loss function choices.

[2] Inferred from the probability $P(\varepsilon_{true} \mid M, m, n)$ to be consistent across the n observations.

less conservative bounds[3] [26]. Still bounds tend to be loose as they are either obtained using a data-independent analysis or, when VC-dimension is estimated from data, rely on a substantial number of approximations.

Comparing Classifiers. The definition of robust estimators is also critical to compare classifiers. For this goal, McNemar and Wilcoxon tests are robust choices to compare pairs of classifiers. In multi-wise settings, Friedman tests with the corresponding post-hoc tests [166] are the choice to obtain conservative guarantees of superiority. For less conservative alternatives, multi-wise tests based on the Bergmann-Hommel procedure [236] should be considered when comparing a compact set of models, and multi-wise tests based on Shaffer's procedure [236] for more extensive comparisons or in the presence large number of collected estimates per model.

Disclosing the Error Components. In data contexts where size is lower than dimensionality, $n < m$, the observed error of a given particular model can be further decomposed in *bias* and *variance* components in order to understand the major cause of the variability across error estimates. While the model's variance is determined by the ability to generalize a model from the available observations (see Figure 1.2), the model's bias is given by the erroneous assumptions associated with the learning function.

This decomposition (further described in *Basics 1.2*) provides clues on whether the model has higher susceptibility to under/overfit the observed data. High bias can cause an algorithm to miss the relevant relations, denoting underfitting propensity. High variance from modeling the random noise in the training data (rather the intended regularities) is associated with overfitting. These assessments are particularly important when the collection of observations is selected from a specific stratum, common for high-dimensional datasets derived from social networks, or affected by specific experimental or pre-processing techniques in biomedical data. In these contexts, the bias-variance decomposition of error provides a useful frame to study the error performance of a classification model, as it is well demonstrated by its effectiveness across multiple applications [182]. For this end, multiple metrics and sampling schema have been developed for estimating bias and variance from data, including the widely employed holdout approach of Kohavi and Wolpert [375].

Basics 1.2 Estimators of variance and bias

The variance is error from sensitivity to small fluctuations in the training data, while bias is the observed loss incurred relative to the optimal prediction. Note that their assessment for classification models differs from regression models, and thus the commonly applied quadratic loss function is inappropriate since labels are not numeric. Instead, robust zero-one loss functions are required [182, 375]. Figure 1.3 illustrates these components of the testing error. For their estimation from a given dataset $\mathbf{A}$, we apply a 10 cross-fold validation, and within each fold we generate 100 samples from the training data partition using bootstrap replacement (according [182]). Under the assumption of no noise [375], a model is then learned on each training sample. In this way, variance and bias can be measured within each fold by computing the difference of error estimates to, respectively, the average and optimal error estimate within the same fold. Consider the collected variance and bias estimates, a estimator (such as the average or q-percentil) can be considered to deliver the expected variance-bias levels associated with a given model.

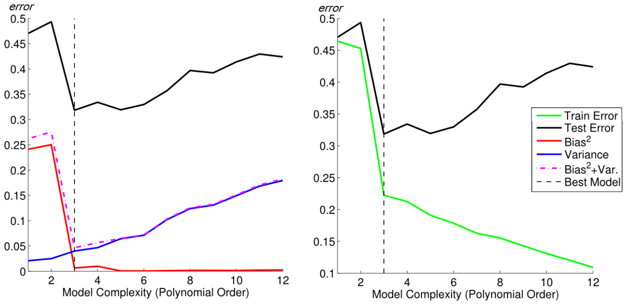


Figure 1.3: Illustrative comparison of the variance, bias and structural noise against the testing and training errors.

Sampling Schema. When the estimator of the true performance is not derived from a direct analysis of the parameters of the learned model, it typically relies on a set of performance estimates collected from assessing the

[3] The number and length of subsets of features can be used to affect the performance guarantees. For instance, a lower-bound on the performance of decision lists relying on tests with at most p features chosen from a m-dimensional space and d-depth is $\theta(\varepsilon_{true}) \geq \frac{1}{n}(log\frac{1}{\delta} + \Theta(p^d log_2 p^d))$.

learned model on samples from the input dataset. Sampling schema is defined by two major variables: sampling criteria and train-test size decisions. Error estimators in high-dimensional data strongly depend on the considered sampling method [669], with many principles being proposed for this end [464, 186, 630]. Cross-validation methods and alternative bootstrap methods (e.g. randomized, 0.632-estimator, *mc*-estimator, complex bootstrap) have been compared for a large number of classifiers and data contexts. Unlike cross-validation, bootstrap was shown to be pessimistically biased with respect to the number of training samples. Still, studies show that bootstrap becomes more accurate than its peers for data contexts with very large observed errors (as often observed for high-dimensional data where $m > n$) [186]. Resubstitution methods are optimistically biased and should be avoided. We consider both the use of k-folds cross-validation and bootstrap to be acceptable, with the number of folds, k, adjusted based on the minimum number of estimates required for a significant inference of confidence intervals (performance bounds). This implies a preference for a large number of folds when either performance is highly variable or for high-dimensional data with $n \ll m$.

An additional problem in the presence of a limited number of observations is to guarantee that the number of test instances per fold offers a reliable error estimate since the observed errors within a specific fold are also subjected to systematic (bias) and random (variance) uncertainty. Two options can be adopted to minimize this problem. First option is to find the best split of training-testing observations. Raudys et al. [553] propose an effective function to find a reasonable size of the test sample based on the train sample size and on the estimate of the asymptotic error. A second option is to model the testing sample size independently from the number of training observations (see *Basics 1.3*). This guarantees a robust assessment of the classifier, but the required number of testing instances can jeopardize the training sample size and, thus, compromise the learning. When this is the case, there is the need to post-calibrate the test-train sizes recurring to the strategies for the first option.

Basics 1.3 Minimum number of testing observations for robust error estimates

Error assessments can be described as a Bernoulli process: n_{test} instances are tested, t successes (or failures) are observed and the true performance for a specific fold can be estimated, $\hat{p}=t/n_{test}$, as well as its variance $p(1\text{-}p)/n_{test}$. The estimation of n_{test} can rely on confidence intervals for the true probability p under a pre-specified precision [55] or from the expected levels of type I and II errors using the statistical tests described by Fleiss [221]. For the biomedical data analyzed by Beleites et al. [55], 75-100 test observations are commonly the necessary minimum to achieve reasonable validation and 140 test observations (confidence interval widths 0.1) are necessary for an expected sensitivity of 90%. Understandably, these values assume the presence of over a total of one hundred of observations per dataset, which may not always be available in some biomedical data domains.

Loss Functions. As different loss functions capture different performance views, their use can lead to radically different performance guarantees. Commonly applied (loss) functions include accuracy (percentage of observations correctly classified); and area under receiver-operating curve (AUC). For settings where the use of confusion matrices is of importance due to the difficulty of the task for some classes or class imbalance, the observed errors can be further decomposed according to type-I errors (false positives), type-II errors (false negatives), sensitivity, specificity and F-Measure. Although the loss function is typically applied in the absence of corrections, *smoothing factors* should be further considered whenever possible to calibrate performance guarantees, as their use results in a more fair assessments. The application of loss functions in the absence of smoothing factors (default assumption) is often referred as error counting, where the error given by the relative number of incorrectly classified testing observations. For more robust assessments, two major smoothing factors can be identified: smooth modification of error counting (also referred as weighted error counting) [256] to accommodate uncertainty factors for models with probabilistic outputs (disclosure of the confidence for each class); and posterior probability estimates [408] for a more realist view in the presence of the multivariate models. Not only these factors provide provide a more robust measure of the true performance where correctly classified observations can contribute to the error, but also their variance is more realistic. The problem with smooth modification is its dependence on the applied distance function, while posterior probabilities tend to be biased for small datasets. Nevertheless, their use is preferred whenever in the presence of learning functions with probabilistic outputs.

Feasibility of Error Estimates. As previously prompted, different estimators of the true error can be defined to find confidence intervals or significant differences associated with the performance of classification models. For this goal, estimators can be derived from the parametric analysis of the learned models or from error estimates gathered under a specific sampling scheme and loss function. Nevertheless, the performance guarantees defined by these estimators are only valid if they are able to perform distinctively better than a null (random) model. An analysis of the significance of these estimators indicates whether we can estimate the performance guarantees or, otherwise, we would need a larger number of observations for the given dimensionality.

A simplistic validation option is to show the significant superiority of M against permutations made on the original dataset [471]. A possible permutation procedure is to construct for each of the k folds, t samples where the classes (discriminative models) or domain values (descriptive models) are randomly permuted. From the errors computed for each permutation, different density functions can be developed, such as:

$$P_{n,m}(x) = \frac{1}{kt}\Sigma_{i=1}^{k}\Sigma_{j=1}^{t}\theta(x - \varepsilon_{i,j,n,m}), \tag{1.2}$$

where $\theta(z)$=1 if $z \geq 0$ and 0 otherwise. The significance of the model is $P_{n,m}(x)$, the percentage of random permutations with observed error smaller than x, where x can be fixed using an expectation of the true error for the target model M. The average estimator, $\varepsilon_{n,m} = \frac{1}{k}\Sigma_{i=1}^{k}(\epsilon_i \mid n, m)$, or the θ^{th} percentile of the sequence $\{e_1, ..., e_k\}$ can be used as the target estimator. Both the average and θ^{th} percentile of error estimates are unbiased estimators. Different percentiles can be used to define error bar envelopes for the true error.

There are two major problems with this approach. First, the variability of the observed errors does not affect the significance levels. To account for the variability of error estimates across the k×t permutations, more robust statistical tests can be used, such as one-tailed t-test with (k×t)-1 degrees of freedom to test the unilateral superiority of the target model. Second, the significance of the learned relations of a model M is assessed against permuted data, which is a very loose setting. Instead, the same model should be assessed against data generated with similar regularities in order to guarantee that the observed superiority does not simply result from an overfitting towards the available observations. Similarly, stastical t-tests are suitable options for this scenario.

As a result of the selected feasibility analysis, the estimates that do not satisfy minimum significance criteria are not used by the target estimator. However, if this analysis reveals that no error estimate can be collected with statistical significance due to data size constraints, we propose the application of two additional strategies. A first strategy it to adopt complementary data by either: *1)* relying on identical real data with more observations (note, however, that distinct datasets can lead to quite different performance guarantees [471]), or *2)* resampling data by approximating the regularities of the original dataset and generating larger synthetic data using the retrieved distributions. A second strategy is to relax the significance levels for the inference of less conservative performance guarantees. In this case, results should be provided as indicative and exploratory.

Inferring Performance Guarantees from Multiple Settings. For the general purpose of studying the performance of a classification model, its assessment should be inferred from real and synthetic data with varying regularities. The high-dimensional datasets listed in Table I-1.1 show unique regularities that can strongly impact its performance, thus revealing its strengths and weaknesses. Complementary, other performance views can be collected in the presence of multiple estimators of the true performance (such as estimators relying on distinct loss functions). Finally, the performance of the classifiers can additionally vary depending on their parameterizations. Understandably, the multiplicity of views related with different estimators, parameters and datasets results in a large number of performance bounds and comparisons that can lead to possibly contradictory views and thus difficult the assessment of a classifier. In this context, the inference of a compact view of both general and specific performance aspects of a model learned from multiple settings is of key importance. Summarization and generalization criteria can be considered to frame the performance guarantees of a particular model M based on the combinatorial explosion of hyper-surfaces from its assessment from a (possibly) large number of settings.

When *comparing models,* simple statistics and hierarchical presentation of the inferred relations should be available. An illustrative example is the delivery of the most significant pairs of values that capture the percentage of settings where a particular model had a superior and inferior performance against another model. In order to reduce the computational complexity of these comparisons, the Nemenyi tests from the introduced Friedman framework [166] can be applied on a subset of the overall pairwise comparisons. Illustrating, when assessing several models from multiple data: superiority can be shown within a single model for different datasets or between distinct models for a single dataset.

When *bounding performance,* (non-linear) functions can be approximated to describe how the performance bounds of a given model vary with regards to a certain (numeric) parameter or data aspect (e.g. increasing dimensionality, noise or complexity of class-conditional regularities). When settings are not directly related, the error estimates collected from different setting can be gathered for the inference of more general confidence intervals. Understandably, when performance guarantees are instead derived from the direct analysis of the learned models, these confidence intervals are not inferred from the gathered set of error estimates but directly from the multiple estimators of true performance. Additional criteria from literature has been proposed to frame variables from estimates gathered from multiple estimations [167] (whether each estimation is derived from model-driven guarantees, data-driven guarantees or error estimates collected from testing the model). In order to avoid very distinct levels of difficulty across settings that penalize the inferred performance bounds, only one variable should be varied at a time and similar settings clustered to obtain a compact set of performance bounds.

Robust Assessments from Real and Synthetic Data. When the target assessment is aimed in the context of a specific real dataset, its data regularities can be retrieved to offer a more informative context for the analysis of the outputted performance guarantees[4]. Local and global descriptive models can be learned for this end.

Contrasting, whenever the assessment in not limited to a specific dataset, synthetic data with varying regularities should be generated. *Pointer 1.5* surveys major contributions from related work towards this end. Three major principles can be identified. First, when considering multivariate distributions to generate data, one should explore: varying distances between their means, covariance-matrices with a varying number of features and correlation factors, and mixtures (including Gaussian and non-Gaussian assumptions) to test non-linear learning aspects. Second, varying forms and degree of noise should be planted to assess the model's robustness. In particular, the use of different sources variability can be simulated for the inference of domain-dependent guarantees of performance (see *Pointer 1.5*). Third, additional properties should be explored, including the generation of: features with distinct discriminative power; local regularities with varying coherency and size; and data with varying number of classes and imbalance.

Pointers 1.4 Principles to generate synthetic data for classification

In simulation studies, generation procedures aim to mimic realistic regularities. Common distribution assumptions include class-conditional multivariate Gaussian distributions [669, 276, 333, 200] assuming unequal means and equal covariance matrices ($\mathbf{X}_i \mid c_1 \sim Gaussian(\mu_1, \sigma^2)$, $X_j \mid y_2 \sim Gaussian(\mu_2, \sigma^2)$, where $\mu_1 \neq \mu_2$). The covariance-matrix can be experimentally varied or estimated from real datasets. In [669], unequal covariance matrices that differ by a scaling factor are considered. While a few datasets after proper normalization have a reasonable fit, the majority of (biomedical) datasets cannot be described by such simplistic assumption. In these cases, the use of mixtures, such as the mixture of a target distribution with Boolean feature spaces [374], are considered to assess non-linear capabilities of the target classification models. Hua et. al [333] proposes a hard bimodal model, where the conditional distribution for class c_1 is a Gaussian centered at μ_0=(0,...,0) and the conditional distribution for class c_2 is a mixture of equiprobable Gaussians centered at $\mu_{1,0}$=(1,...,1) and $\mu_{1,1}$=(-1,...,-1). In [276] study, the complexity of Gaussian conditional distributions was tested by fixing μ_0=0 and by varying μ_1 from 0.5 to 0 in steps of 0.05 for $\sigma_0^2 = \sigma_1^2 = 0.2$. Additionally, one experimental setting generated data according to a mixture of Uniform $U(\mu + 3\sigma, \mu + 6.7\sigma)$ and Gaussian $N(\mu, \sigma^2)$ distributions.

Although these data assumptions can be already considered flexible, some datasets show further complex regularities. Real data commonly have features exhibiting highly skewed distributions. This is a common case with biological data. Guo et al. study [276]

[4] Complementarily, then the target real data is sufficiently large, its size can be varied using sampling techniques to approximate learning curves.

introduces varying levels of signal-to-noise in the dataset, which resulted in a critical decrease of the observed statistical power for the computed bounds. Additionally, only a subset of overall features was generated according class-conditional distributions in order to simulate the commonly observed compact set of discriminative biomarker features.

The majority of real data settings is also characterized by functionally correlated features and, therefore, planting different forms of dependencies among the m target features is of critical importance to infer performance guarantees. Hua et al. [333] proposes the use of different covariance-matrices by dividing the overall features into correlated subsets with varying number of features ($p \in \{1, 5, 10, 30\}$), and by considering different correlation coefficients ($\rho \in \{0.125, 0.25, 0.5\}$). The increase in correlation among features, either by decreasing g or increasing ρ, increases the Bayes error for a fixed dimensionality. Guo et al. [276] incorporates a correlation factor just for a small portion of the original features. Finally, biclusters can be planted in tabular data contexts to capture flexible functional relations among subsets of features and observations [310]. Similarly, triclusters, cascades, sequential patterns and further local patterns can be planted in more structured data contexts. As surveyed in Table I-1.1, this form of local dependencies are observed in a wide-range of biomedical and social data domains [429].

Pointers 1.5 Planting noise according to different sources of variability

The noise can be planted using distinct sources of variability. In real data, variability is associated with different technical and sampling sources of biases, and can be studied when there is knowledge regarding replicates and, in biological data, pooling and dye-swaps. This knowledge can be used to either shape the estimators of the true error or to further generate new synthetic data. Dobbin and Simon work [179, 180] explore how such additional sources of variability impact the observed errors. The variability added by these factors can be estimated from the available data. These factors is critical for datasets with a low number of observations and can be considered for both discriminative (multi-class) and descriptive (single-class) settings. Formulas are defined for each setting by minimizing the difference between the asymptotic and observed error, $(\lim_{n\to\infty} \varepsilon_{true|n}) - \varepsilon_{true|n}$, where $\varepsilon_{true|n}$ depends on these sources of variability. Although this work provides hints on how to address advanced data aspects with impact on the estimation of the true error, the proposed formulas provide loose performance bounds and have been only deduced in the the scope of biological data under the independence assumption among features. The variation of statistical power using ANOVA methods has been also proposed to assess these effects on the performance of models [608].

Extensibility: Performance Guarantees from Imbalance Data. Imbalance in the representativity of classes affect the performance of models and, consequently, the resulting performance guarantees. In many labeled high-dimensional data domains, such as in biomedical domain, case and control observations tend to show significant imbalance (scarce and costly access to rare conditions and phenotypes). In these contexts, performance guarantees inferred from real data should be complemented with analyzes from synthetic data with varying degrees of imbalance. Under such analysis, we can frame the performance guarantees of a specific model M with more rigor.

Additionally, an adequate selection of loss functions to compute the observed errors is required for these settings. Assuming the presence of $k{=}|\Sigma|$ classes, one strategy is to estimate k performance bounds, each associated with the sensitivity (or peer loss function) of a single class. Based on the provided principles to infer bounds from multiple settings, the set of k performance views can be collapsed for a more simplistic analysis of the performance guarantees of the learned model.

1.3.2 Assessment Methodology: Integrating the Proposed Principles

The retrieved principles can be consistently combined according to a simple methodology to enhance the assessment of the performance guarantees of models learned from high-dimensional data. First, the principles associated with the definition of performance estimators, including the selection of adequate loss functions and sampling scheme and the tests of the feasibility of error estimates, can be consistently combined. This provides a structural basis to bound the performance of models, as well as to compare the performance of models using the suggested statistical tests based on the extent of comparisons.

Second, the estimator of the true performance should be further decomposed to measure the bias and variance underlying error estimates. This decomposition offers an informative context to interpret the source of error variability, as well as to understand the propensity of the model towards the risks of underfitting and overfitting associated with learning from high-dimensional data.

Third, to avoid biased performance guarantees towards a single dataset, we propose the complementary estimation of performance guarantees on synthetic dataset with varying properties. In this context, we can easily evaluate the impact of assuming varying regularities $X|Y$, numerosity of classes and imbalance, feature dependencies, and different sources of variability. Since the result of varying a large number of parameters can result in large number of estimations, the identified strategies to deal with the inference of performance guarantees from multiple settings should be embraced in order to collapse these estimations into a compact frame of performance guarantees.

Finally, classifiers either learned after feature selection, or that inherently are able to infer decisions from data regions, should have their performance guarantees adjusted since the variability of the error estimates collected after dimensionality reduction may blur their true performance. For this aim, both the unbiasedness principle of feature selection and conservative estimations from VC-theory can be used to affect the guarantees of performance.

1.4 Results and Discussion

This section experimentally stresses the relevance of the proposed methodology. First, we compare alternative estimators and provide initial evidence for the need to consider the proposed principles to assess classifiers in high-dimensional data contexts where $n<m$. Second, we bound and compare the performance of classifiers from data with varying properties. Finally, we show the importance of selecting adequate performance views for labeled data with imbalance on the complexity and numerosity of class-conditional observations. The software implementing the assessment methodology was codified in Java (JVM version 1.6.0-24). The selected supervised learners were selected from WEKA. The following experiments were computed using an Intel Core i3 1.80GHz with 6GB of RAM.

Datasets. We rely on both real and synthetic data. Two distinct groups of real datasets were used: high-dimensional data with a small number of observations ($n<m$) and high-dimensional data with a large number of observations. For the first group we considered biological data for tumor classification collected from BIGS repository[5]: *colon* cancer data (m=2000, n=62, 2 labels), *lymphoma* data (m=4026, n=96, 9 labels), and *leukemia* data (m=7129, n=72, 2 labels). For the second group we selected a random population from the healthcare heritage prize database[6] (m=478, n=20000) which integrates claims across hospitals, pharmacies and labs. The original relational scheme was denormalized by mapping each patient as an observation with features extracted from the collected claims (400 attributes), the monthly laboratory tests and taken drugs (72 attributes), and the patient profile (6 attributes). We selected the tasks of classifying the need for upcoming interventions (2 labels) and the level of drug prescription ({*low,moderate,high*} labels), considered critical for care prevention and drug management.

Synthetic labeled datasets were generated by varying the following parameters: ratio and number of observations and features, number of classes and their imbalance, conditional distributions (mixture of Gaussians and Poissons per class), amount of planted noise, the percentage of skewed features, and the number/shape/disciminative-power of planted local dependencies. These local dependencies were carefully chosen to follow the properties of biological data [578, 500]. Table 1.4 provides the considered parameterizations.

Features	$m \in \{500, \mathbf{1000}, 2000, 5000\}$
Observations	$n \in \{100, 200, \mathbf{500}, 1000, 10000\}$
Number of Classes	$c \in \{2, \mathbf{3}, 5\}$
Degree of Imbalance (%)	{0%,**30%**,60%,80%}
Distributions (illustrative)	(c=3) {N(1,σ), N(0,σ), N(-1,σ)} with $\sigma \in \{3, 5\}$ (easy setting)
	(c=3) {N(u_1,σ),N(0,σ),N(u_3,σ)} with $u_1 \in$ {-1,2}, $u_2 \in$ {-2,1}}
	(c=3) mixtures of N(u_i,σ) and P(λ_i) where λ_1=4, λ_2=5, λ_3=6
Noise (% of values' range)	{0%,**5%**,10%,20%,40%}
Skewed Features	{**0%**,30%,60%,90%}
♯Local dependencies per class	{4,7,**10**,20}
Size of local dependencies	|**I**|={5,**10**,20,40} (observations), |**J**|={5,10,**20**,50,100} (features)
Discriminative power	{90%,**80%**,70%,60%}

Table 1.4: Parameters for the generation of labeled real-valued matrices (default setting in bold).

Challenges. An initial assessment of the performance of two simplistic classification models learned from real

[5] http://www.upo.es/eps/bigs/datasets.html

[6] http://www.heritagehealthprize.com/c/hhp/data (under a granted permission)

high-dimensional datasets under a cross-validation (with either 10 folds or n folds (leave-one-out)) and bootstrap sampling is given in Figure 1.4. Performance bounds were computed from confidence intervals of a mean estimator (assuming error estimates to be normally distributed and α=0.05 significance). The observed bounds for real data with m>n confirm a heightned variability of performance (difference between the upper and lower bounds is over 20pp). Generally, leave-one-out sampling scheme has higher variability. Although leave-one-out is able to learn from more observations (decreasing the variability of performance), the true variability of 10-fold cross-validation is masked by averaging errors per fold. The smooth effect of cross-validation sampling supports the need to increase the levels of significance to derive more realistic performance bounds. Additionally, the use of bootstrap schema with resampling methods seems to optimistically bias the true performance of the models. Contrasting, models learned from the heritage data setting, where $n \gg m$, have a more stable performance across folds. This leads to a higher number and significance of superiority relations extracted from comparing these classification models using Friedman tests.

Bounding performance using VC inference or specific percentiles of error estimates introduces undesirable bias. In fact, under similar experimental settings, the VC bounds were very pessimistic (>10pp of difference). Complementary, the use of the 0.15 and 0.85 percentiles (to respectively define lower and upper bounds) led to more optimistic bounds than the ones provided in Figure 1.4. Although percentiles can be used to control the confidence associated with the estimated bounds, they are insufficient to model the true variability of error estimates.

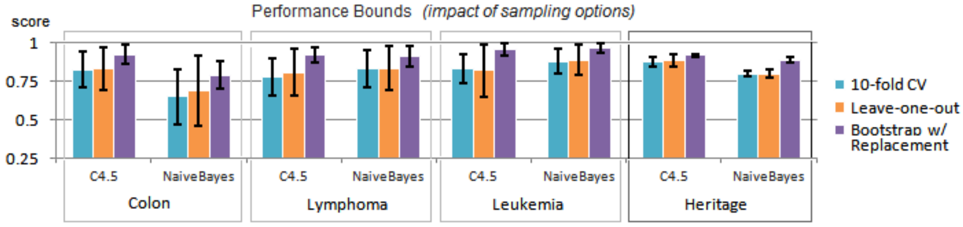


Figure 1.4: Performance guarantees from real datasets with varying $\frac{n}{m}$ degree for two classifiers (C4.5 [542] and Naive Bayes [356]) tested under different sampling options (cross-validation, leave-one-out and bootstrap).

A view on the significance/feasibility of the inferred performance guarantees from real and synthetic high-dimensional data is respectively provided in Table 1.5 and Figure 1.5. Different methods were adopted to compute the significance (p-value) associated with a collection of error estimates. These methods basically compute a p-value by comparing the collected error estimates against estimates provided by loose settings where: *1)* the target model is learned from permuted data; *2)* a *null* classifier is learned from the original data; and *3)* the target model is learned from *null* data (preservation of global conditional regularities). The null classifier simply averages a class-conditional values per feature using the training observations and computes the mode of classes, where each class is associated with the expected class per feature based on the observed values per feature for a given testing observation. We also considered the setting proposed by Mukherjee et al. [471] (Eq.1.2). Comparisons are given by one-tailed t-tests. For this analysis, we compared the significance of the learned C4.5, Naive Bayes and support vector machines (SVM) models for real datasets and averaged their values for synthetic datasets. A major observation can be retrieved: p-values are not highly significant ($\ll$1%) when n<m, meaning that the performance of the learned models is not significantly better than very loose learners. Again, this observation underlines the importance of carefully framing assessments of models learned from high-dimensional data contexts. Additionally, different significance views can result in quite different p-values, which stresses the need to choose an appropriate robust basis to validate the collected estimates. The tests based on comparisons against null data are the most conservative, while the counts performed under Eq.1.2 (permutations) are not sensitive to distances among error mismatches and can easily lead to biased results.

To further understand the root of the variability associated with the performance of models learned from high-dimensional datasets, Figure 1.6 provides its decomposition in two components: bias and variance. Bias provides

	Colon			Leukemia			Heritage		
	C4.5	NBayes	SVM	C4.5	NBayes	SVM	C4.5	NBayes	SVM
Comparison Against Permutated Data	1.5%	41.3%	1.2%	0.6%	0.1%	0.2%	~0%	~0%	~0%
Comparison Against Null Model	1.1%	32.2%	1.2%	0.1%	0.1%	0.1%	~0%	~0%	~0%
Comparison Against Null Dataset	15.2%	60.3%	9.3%	9.7%	12.0%	7.2%	1.3%	3.8%	1.7%
Permutations Density Function (Eq.1.2)	14.0%	36.0%	8.4%	8.4%	1.2%	0.8%	0.0%	0.4%	0.0%

Table 1.5: Significance of the collected error estimates of models learned from real datasets using improvement p-values. p-values are computed by comparing the target models vs. a baseline classification models, and error estimates collected from the original dataset vs. a permuted dataset or null dataset (where basic regularities are preserved).

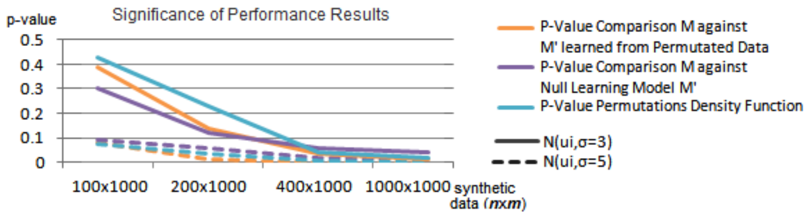


Figure 1.5: Significance views on the error estimates collected by classification models from m>n synthetic datasets under easy N(u_i,σ=3) and moderate N(u_i,σ=5) settings against loose baseline settings.

a view on how the expected error deviates across folds for the target dataset, pinpointing the propensity of a model to underfit data. Variance provides a view on how the model behavior differs across distinct training folds, measuring the propensity of a model to overfit data. We can observe that the bias component is slightly higher than the variance component. This is understandable, since the assessed classification model is a decision tree (C4.5 [542]), and thus decisions are inferred from small regions of the original data space (small subsets of features from the overall high-dimensional set of features). Interestingly, we observe that the higher $\frac{m}{n}$ ratio is, the higher the bias/variance ratio. The sum of these components decrease for an increased number of observations, n, and it also depends on the nature of the conditional distributions of the dataset, as it is shown by synthetic data under class-conditional Gaussian assumption with small-to-large overlapping areas under the probability density curves. The focus on each one of these components is also critical to study the impact of the capacity error associated with the learned model (see Figure 1.2).

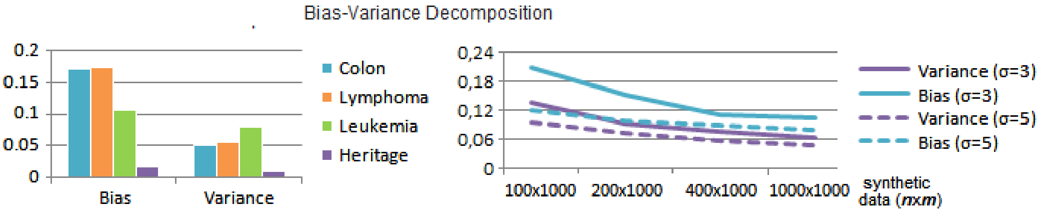


Figure 1.6: Decomposition of the performance variability from real and synthetic data using C4.5: understanding the model capacity (*variance* component) and the model error (*bias* component).

Imbalanced Multi-class Data. The importance of selecting adequate performance views to retrieve realistic guarantees is shown in Figure 1.7. This is still an underestimated problem that needs to be addressed for both: *1)* balanced datasets where the class-conditional distributions differs in complexity (see the sensitivity associated with the classes from Colon and Leukemia datasets in Figure 1.7a), and *2)* for imbalanced datasets, where the representativity of each class can hamper the learning task even if the complexity of the class-conditional distributions is similar (see the sensitivity of the classes from datasets with different degrees of imbalance Figure 1.7b). In this analysis, we considered sensitivity as a simplistic motivational metric, however many other loss functions hold intrinsic properties of interest to derive particular implications from the performance of the target models. The chosen performance view not only impacts the expected true error, but the variability of the error as it is well-demonstrated in Figure 1.7a. This impacts both the inferred bounds and the number of significant comparisons.

Performance Guarantees from Flexible Data Settings. To understand how performance guarantees vary across different data settings for a specific model, we computed C4.5 performance bounds from multiple synthetic data

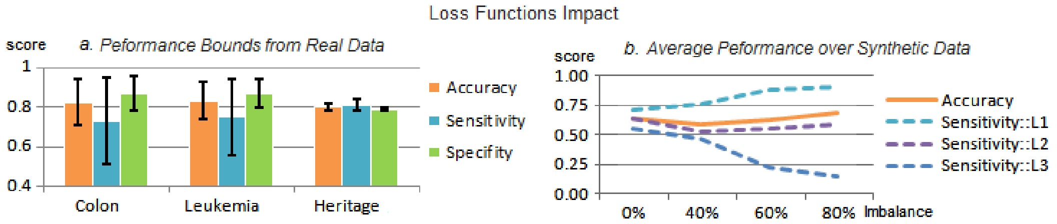
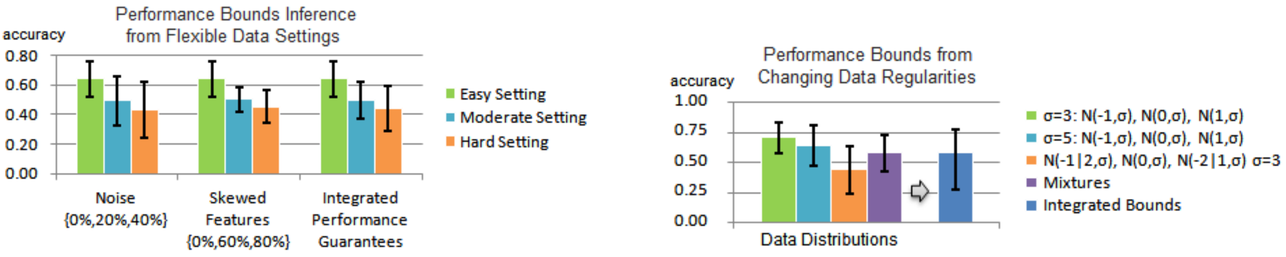


Figure 1.7: Impact of adopting alternative loss functions on the: *a*) performance variability of real datasets, and *b*) true performance of synthetic datasets (n=200 and m=500) with varying degrees of imbalance among classes.

settings with varying degree of planted noise and skewed features. This analysis is provided in Figure 1.8a. Generalizing bounds from datasets with different learning complexity may result in very loose bounds and, therefore, should be avoided. In fact, planting noise and skewing features not only increases the expected error but also its variability. Still, generalizations are possible when the differences between collections of error estimates is not high. In these cases, collections can be joint to compute new confidence intervals. When the goal is to compare sets of models, superiority relations can be tested for each setting under relaxed significance levels, and outputted if the same relation appears across all settings. In our experimental study, only few superiority relations between C4.5 and Naive Bayes using the Friedman-test under loose levels of significance (10%).

Figure 1.8b assesses the impact of using different conditional distributions for the inference of general performance guarantees for C4.5. Understandably, the expected error increases when the overlapping area between conditional distributions is higher or when a particular class is described by a mixture of distributions. Combining such hard settings with more easy settings gives rise to loose performance bounds and to a residual number of significant superiority relations between models. Still, this assessment is required to validate and weight the increasing number data-independent implications of performance from the recent studies.



(a) Varying degree of planted noise (deviations as a percentage of domain values) and skewed features (percentage of total features).

(b) Varying class-conditional distributions (assumptions described in Table 1.4).

Figure 1.8: Inference of performance guarantees in a (n=200,m=500)-space with varying regularities.

Discussion. In this chapter, we synthesized critical principles to bound and compare the performance of models learned from high-dimensional data. First, we surveyed and provide empirical evidence for the challenges related with this task for data where size is comparable or lower than dimensionality (n<m). This task is critical as many scientific implications have been derived from classification models whose differences in performance against permuted or *null* data is not significant. Also, the distance of the estimated confidence bounds is considerably high in these data contexts, leading to the absence of statistically significant superiority relations from the application of Friedman comparisons.

Second, motivated by these challenges, we have shown the importance of adopting robust statistical principles to test the feasibility of the collected estimates. Different tests for computing significance levels have been proposed, each one providing different levels of conservatism, which can be used to validate and weight the increasing number of implications derived from these models.

Third, we compared alternative ways of bounding and comparing performance, including different sampling schema, loss functions and statistical tests. In particular, we used initial empirical evidence to show how different estimators can bias the true error or smooth its variability. An alternative to the inference of performance guarantees from estimates is to approximate the true performance from the properties of the learned models. For this latter line of research two strategies can be followed. A first strategy is to retrieve guarantees from the learned parameters from multivariate models that preserve the original dimensionality [644, 200]. A second strategy is to understand the significance and discriminative power of the selected regions from the original space whenever a form of dimensionality reduction is applied before or during the learning process [589, 343].

Fourth, understanding the source of variability of the performance of the models is critical in these spaces as this variability can be either related with the overfitting aspect of the models or with the learning complexity associated with the properties of the model and data. The variability of performance can, thus, be further decomposed in variance and bias. While the variance captures the differences on the behavior of the model across samples from the target population, the bias captures the learning error within these samples. These components are thus critical to understand and refine classifiers' behavior.

Fifth, the impact of varying data regularities on the performance guarantees was also assessed by generating data with varying $\frac{n}{m}$ ratio, (conditional) distributions, noise, class imbalance, and uninformative features. In particular, we observed that inferring general bounds and comparisons from flexible data settings is possible, but tends to originate loose guarantees when mixing data with distinct regularities. In these cases, a summarization and comprehensive presentation of the guarantees should be available. In particular, when bounding performance, a regression for the variations of performance can be alternatively provided.

1.5 Summary of Contributions and Implications

Motivated by the challenges of learning from high-dimensional data, this chapter establishes a solid foundation on how to assess the performance guarantees of classification models. For this end, we surveyed contributions on how to bound and compare the performance of models (as a function of data size and/or dimensionality) from distinct research streams. A taxonomy to understand their major challenges was proposed. To answer the identified challenges, a set of principles was proposed. These principles offer a solid foundation to define adequate estimators of the true performance. In particular, three major types of estimators were considered – estimators learned 1) from error estimates, 2) from a direct analysis of the learned model, and 3) from a direct analysis of the input data. Complementary, we proposed adequate statistical tests to bound and compare performance, as well as adequate performance views with desirable smoothing factors in the presence of probabilistic outcomes. Furthermore, we extend these principles to infer performance guarantees from multiple parameterizations and flexible data settings where the underlying global and local regularities can vary. Finally, we show how the observed error can be decomposed to study the generalization error of the classifier.

Experimental results support the relevance of these principles. They provide initial empirical evidence for the importance of computing adequate significance views to adjust the statistical power when bounding and comparing the performance of models, of selecting adequate error estimators, of inferring guarantees from flexible data settings, and of decomposing the error to gain further insights on its sources of variability.

This work opens an important door for understanding, bounding and comparing the performance of classification models. The guarantees of performance inferred under the proposed methodology can be used to weight and validate the increasing number of implications derived from learning tasks over high-dimensional data, and to correctly measure the impact of dimensionality reduction procedures. In this context, the proposed methodology is applied to assess the contributions of this dissertation and as the means to validate the target hypothesis.

2

Performance Guarantees of Local Descriptive Models

The proposed principles to assess the performance of classification models cannot be directly applied towards descriptive models as there are not yet consensual estimators of their true performance. This prevents the analysis of their true performance and the interpretation of their generalization ability. The commonly applied loss functions are biased towards specific methods and are often associated with incomplete performance views, and there is no ground truth to evaluate descriptive models from real data. These challenges are observed when assessing local descriptive models learned from both tabular data and structured data.

This chapter addresses these challenges for local descriptive models, which for tabular data are flexibly given by biclustering models and for more structured data are given by triclustering and cascade models. As a result, four major contributions are provided. First, we extend the previous principles towards descriptive settings, enabling the possibility to bound and compare of local descriptive models. Second, state-of-the-art loss functions for the assessment of local descriptors from tabular data contexts are compared and new principles for their adequate application are proposed. Third, these principles are extended for structured data contexts and new loss functions proposed for an adequate assessment of triclustering and cascade models. Finally, the principles retrieved under these contributions are consistently integrated within a robust evaluation methodology.

These four contributions are respectively described in *Sections 2.1-2.4*, and their implications synthesized in *Section 2.5*. The empirical evidence of their relevance is provided throughout *Books III* and *IV* along with the assessment of new methods to learn local descriptive models. Figure 2.1 provides a structured view on the challenges and proposed contributions to robustly assess descriptive models.

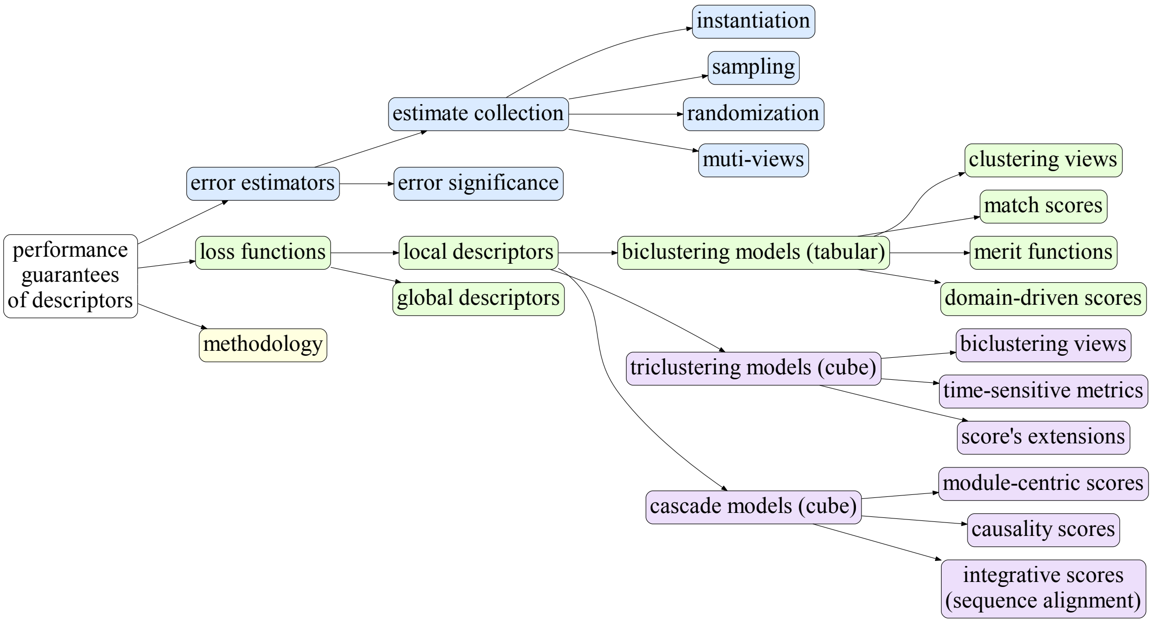


Figure 2.1: Challenges and contributions for inferring performance guarantees from descriptive models.

2.1 Bounding and Comparing Descriptive Models

Revisiting the structural concepts introduced in *Section 1.2*, a *descriptive model* ($|C|=1$) either globally or locally approximates P_X regularities. In particular, a local descriptive model is a composition of regions given by subsets of observations, features and (possibly) time points. Given a tabular dataset with $\mathbf{X}$ observations and $\mathbf{Y}$ features, a bicluster defines region ($\mathbf{I}_i \subseteq \mathbf{X}, \mathbf{J}_i \subseteq \mathbf{Y}$) satisfying a specific homogeneity criteria (Def.I-1.10). Given more structured data with $\mathbf{X}$ observations, $\mathbf{Y}$ features and $\mathbf{T}$ time points, a tricluster defines a region ($\mathbf{I}_i \subseteq \mathbf{X}, \mathbf{J}_i \subseteq \mathbf{Y}, \mathbf{K}_i \subseteq \mathbf{T}$) satisfying a specific homogeneity (Def.I-1.11). When contiguity of $\mathbf{K}_i$ time points is assumed, a set of triclusters can be meaningfully related through a cascade (Def.I-1.12).

Although the use of sampling principles to bound and compare the performance from error estimates is a natural option for classification models, there are not yet contributions that show whether performance guarantees can be also inferred for descriptive models. This analysis is critical to evaluate the generalization error of descriptive models since they are equally susceptible to either overfit or underfit the high-dimensional data.

For this aim, the proposed loss functions in previous chapter need to be replaced according to the properties of the target model. A synthesis of the commonly used performance metrics for global descriptive models, regression models, and biclustering and triclustering models is provided in Table 2.1. The evaluation of local descriptive models can either be made in the presence or absence of planted biclusters/triclusters, $\mathcal{H}$. Similarly, global descriptive models that return a mixture of distributions that approximate the population from which the sample was retrieved, $P_{\mathbf{X}|\mathbf{Y}} \sim \pi$, can be evaluated in the presence and absence of the underlying true regularities. A detailed discussion of the listed metrics of local descriptors is provided in the following section.

Understandably, the application of smoothing factors is no longer meaningful in the context of regression models (as the output is already a numeric quantity), although can be applied in the context of descriptive models with probabilistic outputs. Illustrating, a biclustering model with a probabilistic output is defined by a set of membership vectors, where each vector corresponds to a single bicluster by expressing the probability of each row $\mathbf{x}_i$ and column $\mathbf{y}_j$ in the matrix being included in the bicluster ($\mathbf{x}_i \in \mathbf{I}$ and $\mathbf{y}_j \in \mathbf{J}$).

Model	Performance views
Classifier	Accuracy (percentage of samples correctly classified); area under receiver-operating curve (AUC); metrics derived from (multi-class) confusion matrices, including sensitivity, specificity and F-Measure.
Regression	Normalized root mean squared error, ϵ-insensitive, Huber, among others.
Biclustering (hidden regions)	Entropy [27, 577]; clustering F-measure (and its recall and precision terms); Jaccard-based scores [536]; fabia consensus [324]; relative non-intersecting area (RNAI) [91]; subspace clustering error [516].
Biclustering (no hidden regions)	Merit functions (as long as not biased towards the learning method), such as squared residue and Pearson's correlation [503, 134]; assessment of the over-representation of terms from knowledge bases [429].
Triclustering	Biclustering views, dedicated merit functions (3D residues and hyperplanes' comparison) [281, 12], time-sensitive similarities [176].
Cascade	Module-centric metrics (from biclustering and triclustering views); causal-centric metrics (fan-in, fan-out and cascade errors [437]); integrative scores such as sequence similarity and Bayesian sensitivity and specificity [339].
Global descriptor	Fit of the learned model towards the observed data, similarity between the approximated and known regularities.

Table 2.1: Performance views to estimate the true error of decision and descriptive models.

Given these loss functions, the proposed assessment becomes applicable to descriptive models under the following principles. First, in the presence of synthetic data with hidden biclusters/triclusters/regularities, multiple data instances for each set of data parameterizations are generated. This is done by generating instance with a preserved number of observations n, features m, and both local and global regularities. In this way multiple error estimates become available, turning the inference of performance bounds, as well as the study of variance and bias components, possible. We suggest the generation of ≥30 data instances, a minimum threshold for the application of statistical principles dependent on a Gaussian assumption.

Second, in the presence of real data (absence of hidden biclusters/triclusters/regularities), four strategies can be considered. Error estimates can be collected from: 1) multiple subsamples of the target dataset (where testing observations are either discarded or used to control different sources of error [670, 579]); 2) multiple synthetic datasets generated using the approximated (global and local) regularities than the given dataset; 3) similar real datasets whenever available; or 4) multiple performance views associated, for instance, with different loss functions. Understandably, each option has an inherent drawback. First option neglects part of the data and can lead to redundant errors. The methods for approximate-and-generate data or randomize data often neglect specificities of interest and may show some bias towards the assessed approaches. Similar real data may not always be available, and even within the same data domain can show distinct regularities (evidenced, for instance, from expression data collected for different tumors [219]). Finally, the error estimates collected under the last option are not conceptually consistent. As a result of these observations, for the purpose of inferring performance guarantees we either suggest the upfront specification of multiple real datasets or, the replacement of this analysis, by the previous estimators based on more objective error estimates collected from synthetic data.

In the context of the collection of error estimates and similarly to classification models, it is relevant to test their feasibility in order to guarantee that estimators rely on significant error estimates only. This can done against: 1) a null learning function (by using simplistic methods such as BiMax for biclustering [536], multivariate Gaussians for global descriptors [356] and triCluster for triclustering [710]), 2) null datasets (by permuting or randomizing data [252, 219]), or 3) the guarantees of statistical significance of the assessed descriptive model. Due to the complexity and relevance of this later option, robust statistical tests for this end are proposed in *Book V*.

Some implications of extending the assessments in the previous chapter towards descriptive models include the possibility to: draw comprehensive results from multiple assessments; decompose the generalization error to further assess the properties of these models; and rely on the previously proposed (non-biased) estimators and statistical tests for bounding and comparing models under different levels of conservatism. Conservatism is given by the confidence and statistical power when bounding performance and by the selected test and applied correction when comparing descriptive models.

2.2 Performance Views for Biclustering Models (Local Descriptors in Tabular Data)

The definition of conclusive loss views associated with biclustering models is challenged by three major issues. First, a large variety of loss functions and synthetic datasets have been proposed with many being biased to the specificities of a particular method. The behavior of biclustering methods are mainly defined by an underlying merit function. Merit functions are used to guide the search and provide a simple means to affect the structure, coherency and quality of biclusters. An illustrative merit function is the variance of the values in the bicluster. In this context, biases are observed when a variant of the adopted merit function is used to assess the biclustering method. Second, there is no ground truth to describe the biclusters observed in real data. Third, despite the efforts in developing standard assessment methodologies [475, 536], they only cover specific aspects of performance, leading to wrong assumptions regarding the performance of the target methods.

Similarly to decision models, assessing biclustering models on both synthetic and real data is essential to obtain a complete view on the strengths and weaknesses of a given method. In synthetic data, a set of biclusters $\mathcal{H} = \{\mathbf{H}_1, ..\mathbf{H}_g\}$ (referred as hidden or true biclusters) is typically planted. Objective metrics can be formulated since an approximate solution is known a priori, including clustering metrics, metrics based on the relative non-intersecting area [91, 516], and match scores [536, 324]. In the absence of hidden biclusters, only subjective metrics can be formulated. Merit functions can be applied as long as they are not biased towards the merit functions used within the approaches under comparison. Complementarily, domain-driven scores can be computed using the groups of rows and columns in biclustering solutions retrieved from real datasets against annotations extracted from (biomedical) knowledge bases, semantic sources or bibliographic databases [440, 620]. Below, we provide further detail on the properties on this set of performance views.

Clustering Metrics. A bicluster $(\mathbf{I},\mathbf{J})$ can be seen as the intersection of two cluster sets: rows' cluster $\cup_{i\in\mathbf{I}}\mathbf{x}_i$ and columns' cluster $\cup_{j\in\mathbf{J}}\mathbf{y}_j$. In this context, clustering metrics are applied to one dimension at a time (rows or columns). Typical objective functions in clustering aim high *intra-cluster similarity* (overall values on columns/rows within a bicluster are similar across rows/columns) and low *inter-cluster similarity* (values on columns/rows differ between biclusters). *Purity* is a simple and transparent metric that combines these similarities. However, a high purity can be easily achieved under a high number of clusters (in particular, is maximal if there is one bicluster for each row/column). Metrics addressing this problem can be derived from mutual-information theory [27, 577]. In particular, we propose the use of entropy (Eq.2.1). Entropy is centered on two major principles: *i)* a found cluster should mainly contain rows from a single hidden cluster; and *ii)* hidden clusters should not be partitioned or merged across found clusters [475]. The overall quality of a biclustering model is given by the average of all found biclusters $\mathbf{B}_j \in \mathcal{B}$ weighted by the number of rows/columns per cluster (Eq.2.2). We normalize the maximal entropy such that results are bounded by 0 (perfect) and 1 (low quality).

$$E(\mathbf{B}_j) = -\Sigma_{i=1} P(\mathbf{H}_i|\mathbf{B}_j) log(P(\mathbf{H}_i|\mathbf{B}_j)), \quad with\ P(\mathbf{H}_i|\mathbf{B}_j) = \frac{|I_i \cap I_j|}{|I_j|} \tag{2.1}$$

$$E(\mathcal{B}) = \frac{\Sigma_{j=1}^{|\mathbf{B}_j|} |I_j| E(B_j)}{log|\mathcal{H}| \times \Sigma_{j=1}^{|\mathbf{B}_j|} |I_j|} \tag{2.2}$$

An alternative is to view clustering as a series of decisions. Rand index measures the percentage of decisions that are correct (that is, a form of accuracy that penalizes both false positive and false negative decisions). In particular, we propose the use of F-measure to evaluate how well the hidden biclusters are represented [28, 474]. The underlying principle is that biclusters should cover many rows/columns of a particular hidden cluster but few rows/columns from other hidden clusters. To preserve this principle, $I_m(H_j)$ is adopted, meaning the union of all rows/columns from the biclusters is mapped to the closer hidden cluster (formally, $\forall_z \frac{|I_{B_i} \cap I_{H_j}|}{|I_{H_j}|} \geq \frac{|I_{B_i} \cap I_{H_z}|}{|I_{H_z}|}$). This idea can be formulated with the terms recall and precision. A high recall corresponds to a high coverage of rows/columns from H_j, while a high precision denotes a low coverage of rows/columns from other clusters (Eq.2.3). The harmonic mean of precision and recall is the F-measure. The average over the F-measured values for all hidden clusters $\{H_1, .., H_m\}$ gives the solution F-measure (Eq.2.4).

$$recall(H_j) = \frac{|I_{H_j} \cap I_m(H_j)|}{|I_{H_j}|}, \quad precision(H_j) = \frac{|I_{H_j} \cap I_m(H_j)|}{|I_m(H_j)|} \tag{2.3}$$

$$F - measure = \frac{1}{m} \Sigma_{j=1}^{m} \frac{2 \times recall(H_j) \times precision(H_j)}{recall(H_j) + precision(H_j)} \tag{2.4}$$

A final possible metric is to rely on the accuracy from cluster-based classification [474, 100]. The idea is to predict the hidden cluster of a row/column using the found biclusters. Each observation $\mathbf{x}_i$ is thus represented as a bit-vector of length m where position j equals 1 if $\mathbf{x}_i \in \mathbf{B}_j$ and 0 otherwise. For quality measurement, a classifier is built (using biclustering solutions) and evaluated according to the principles proposed in the previous chapter.

Match Scores. Alternative loss functions have been proposed with the goal of simultaneously assessing both of the dimensions of a biclustering model. A commonly used quality metric for this end is the *relative non-intersecting area* (RNIA) [91]. RNIA[1] measures to what extent the elements a_{ij} in hidden biclusters are covered by the found biclusters. The problem of RNIA metric is that one cannot distinguish if several found biclusters cover a hidden bicluster or exactly one found bicluster matches the hidden bicluster. Therefore, an extended version of RNIA, *clustering error* (CE) [516], was proposed to map each found bicluster to at most one hidden bicluster and each hidden bicluster to at most one found bicluster. In this way, CE penalizes solutions where similar biclusters are

[1] Expressed as $(U - I)/U$, where U is the union of elements in a hidden or found bicluster, and I is the intersection of elements (elements in both a hidden and a found bicluster). The more equal I and U, the better the solution quality.

found with either erroneously excluded or included rows and/or columns. To gain further insight into the obtained results, the CE metric should be decomposed in two components: U (uncovered portion of a hidden bicluster) and E (portion of the found bicluster that was not implanted), possibly plotted within a two-dimensional chart.

An alternative metric is to rely on Jaccard-based match scores (MS) [536] to assess the similarity of $\mathcal{B}$ and $\mathcal{H}$ directly from the Jaccard index (Eq.2.5). $MS(\mathcal{B},\mathcal{H})$ defines the extent to which found biclusters cover the hidden biclusters (completeness), while $MS(\mathcal{H},\mathcal{B})$ reflects how well hidden biclusters are recovered (precision). $\mathcal{H} = \mathcal{B}$ is only achieved when both scores are optimum. Although these scores has been largely applied for biclustering solution, they are only able to evaluate the accuracy of one dimension at a time (rows or columns), similarly to clustering metrics.

$$MS(\mathcal{B},\mathcal{H}) = \frac{1}{|\mathcal{B}|}\Sigma_{(I_1,J_1)\in\mathcal{B}} max_{(I_2,J_2)\in\mathcal{H}} \frac{|I_1 \cap I_2|}{|I_1 \cup I_2|} \tag{2.5}$$

However, since MS scores are not sensitive to the number of biclusters in both sets, Hochreiter et al. [324] introduced a consensus (FC) by computing similarities between the pairs of closest biclusters between $\mathcal{B}$ and $\mathcal{H}$. As the FC score is divided by the number of biclusters of the larger set, it penalizes large biclustering solutions. Assuming that S_1 and S_2 are, respectively, the larger and smaller set of biclusters from $\{\mathcal{B},\mathcal{H}\}$, and MP contains the assigned pairs using the Munkres method based on the extent of overlapping areas [476], then FC is given by Eq.2.6.

$$FC(\mathcal{B},\mathcal{H}) = \frac{1}{|\mathcal{S}_1|}\Sigma_{((I_1,J_1)\in\mathcal{S}_1,(I_2,J_2)\in\mathcal{S}_2)\in MP} \frac{|I_1 \cap I_2| \times |J_1 \cap J_2|}{|I_1| \times |J_1| + |I_2| \times |J_2| - |I_1 \cap I_2| \times |J_1 \cap J_2|} \tag{2.6}$$

Basics 2.1 Biclustering models learned from univariate time series: differences and similarities

The paradigmatic case of biclustering, the discovery of correlated subsets of observations and columns/features, is applicable to real-valued matrices. In expression data, one dimension is given by genes and another by conditions (samples retrieved from different methods, stimuli, environmental contexts, tissues, organs or individuals). When the input data is given by a set of univariate time series, a specialization of this original biclustering task assuming a contiguity constraints on the subsets of columns/time points is commonly targeted. In expression data, conditions now correspond to time-points for transcriptional activity along time [427, 151]. Naturally, when this is the case new similarity metrics for subsets of time points can be made available to compare a found bicluster against a hidden bicluster. Previous studies [176, 551] survey metrics to compare time series, effectly weighting temporal misalignments such as time lags. Similarly, similarity metrics between time series can be used in these contexts to compose new merit functions, since they can be used to assess the homogeneity between pairs of observations within a found bicluster.

Merit Functions. Merit functions have been proposed both with the goal to guide the biclustering task and to evaluate the quality of solutions. Common merit functions rely on alternative forms of co-variance between both rows and columns of a particular bicluster. Merit functions can be applied in the absence of knowledge regarding the hidden biclusters. Merit functions either define the homogeneity of each bicluster (intra-bicluster homogeneity) or of the output set of biclusters (inter-bicluster homogeneity), allowing some biclusters to deviate from the expected homogeneity as long as the overall criterion is preserved. In this way, merit functions can be used to provide both local and global performance views of a biclustering model.

To illustrate some of these functions with simplicity, we denote a_{iJ}, a_{Ij} and a_{IJ} values as, respectively, the mean of the values of $\mathbf{x}_i$ observation ($\frac{1}{|J|}\Sigma_{j\in J}a_{ij}$), of $\mathbf{y}_j$ feature ($\frac{1}{|I|}\Sigma_{i\in I}a_{ij}$) and of all elements in the bicluster ($\frac{1}{|I||J|}\Sigma_{i\in I,j\in J}a_{ij}$). Illustrative examples of merit functions include: the commonly adopted mean squared residue (MSR) [134], optimally tuned for the assessment of a constant biclusters (Eq.2.7), MSR extensions [690] to accommodate additive and multiplicative forms of coherency), and the Pearson's correlation coefficent (PCC) [91] able to evaluate more flexible forms of coherency (Eq.2.8). An extensive comparison of merit functions is provided by Orzechowski [503].

$$MSR = \frac{1}{nm}\Sigma_{i\in I}\Sigma_{j\in J}\mu_{ij}^2, \quad with\ \mu_{ij} = a_{ij} - a_{iJ} - a_{Ij} + a_{IJ} \tag{2.7}$$

$$PCC = \frac{\Sigma_{j\in J}(a_{ij} - a_{iJ})(a_{Ij} - a_{IJ})}{\sqrt{\Sigma_{j\in J}(a_{ij} - a_{iJ})^2 \Sigma_{j\in J}(a_{Ij} - a_{IJ})^2}} \tag{2.8}$$

Merit functions can be used to assess a given biclustering model directly or, alternatively, to guide the extraction of a new set of biclusters seen as the set of hidden biclusters and the given model assessed using the previously introduced matching scores. In this second option, the merit function needs to be applied within an unconstrained optimization problem, which can be solved using, for instance, a Quasi-Newton method or a stochastic approach under tight convergence criteria [485].

Understandably, some key drawbacks associated with the application of merit function to assess a biclustering model can be identified. First, the selection of merit functions for the assessment of biclustering method can be subjected to large biases when the same or similar merit functions are used to guide the search. Second, merit functions are focused on specific aspects of homogeneity of a bicluster (or set of biclusters), often disregarding alternative (yet coherent) forms of homogeneity. For this reason, and since different merit functions can provide radically distinct performance views, multiple merit functions should be ideally combined for more fair assessments. Nevertheless, the application of multiple merit functions increases the assessments' complexity. Third, merit functions may benefit biclustering models with small biclusters as they easily show high homogeneity levels. Some proposed merit functions in literature compensate this undesirable effect by weighting the size of biclusters in order to benefit the discovery of larger biclusters [503]. However, this is insufficient to guarantee the significance of biclusters and, in some cases, often leads to the opposite (undesirable) effect of favoring large biclusters with weak homogeneity, thus increasing the risk of false positive assessments.

Domain-driven Metrics. Although real data is critical to reveal the true performance of a learning method on a specific domain and to avoid biases from data generation procedures, only subjective loss functions can be formulated in this context since there is no ground truth to describe the hidden biclusters. As a result, the existing assessments either target the homogeneity and quality of the discovered biclusters (by using the introduced merit functions) or, alternatively, the domain relevance of a biclustering model against background knowledge (by formulating domain-driven scores). For this second purpose, labels L (annotations) can be extracted from (biomedical) knowledge bases, semantic sources or bibliographic databases [458, 16]. These labels are simply clusters, either associated with a group of rows $\cup_{i \in I} \mathbf{x}_i$ or columns $\cup_{j \in J} \mathbf{y}_j$. Understandably, rows (or columns) can have none or multiple labels associated. Although existing methodologies to retrieve annotations are more proeminent for biomedical data domains and less complete and reliable for certain social data domains, domain-driven assessments are consensually considered to be critical to validate the performance of biclustering methods.

In the presence of annotations, a common option is to determine a p-value for each bicluster, by testing an hypergeometric hypothesis against these labels (see Table 2.2). The p-value, often referred as a measure of enrichment, can be additionally tuned to account for family-wise errors, as well as other alternatives to control type I errors (false positive enrichments). The *family-wise error rate* (FWER) denotes the probability of accepting at least one false positive annotation (flagging a non-significant label as significant) from the set of tested annotations for a particular bicluster [64]. For this end, a conservative correction satisfying FWER is the Bonferroni adjustment, which bounds the α risk of false positive discoveries when assessing s annotations by the corrected α/s level for each test. Varying corrections have been applied for this end with different thresholds of significance [500, 578]. Alternatives to the hypergeometric test, including Fisher's exact test, have been also proposed [672].

	$\in L$	$\notin L$	
♯rows in the bicluster	n_{11}	n_{12}	n_{1+}
♯rows not in the bicluster	n_{21}	n_{22}	n_{2+}
	n_{+1}	n_{+2}	n

Table 2.2: Hypergeometric testing of biclusters against external labels. Assessment is done by testing the hypothesis that n_{11} is sufficiently high by using a hypergeometric assumption to compute the p-value: $\Sigma_{x \leq n_{11}} P(N_{11}{=}x|n_{+1}, n, n_{1+})$ where $P(N_{11}{=}n_{11}|n_{+1}, n, n_{+1}){=}\binom{n_{+1}}{n_{11}} \times \binom{n_{+2}}{n_{12}} / \binom{n}{n_{1+}}$.

Pointers 2.2 Annotations in biological domains

In biological domains, the hypergeometric score has been used to investigate whether the discovered sets of molecular entities (genes, proteins, metabolites and molecular complexes) show significantly enriched annotations from different data sources. These annotations are typically given by Gene Ontology terms (http://www.geneontology.org), well-studied transcription factors (http://

genie.weizmann.ac.il), protein-protein interaction (http://dip.doe-mbi.ucla.edu) and metabolic pathways (http://www.genome.jp/kegg/).

Discussion. Despite the maturity of clustering metrics, they have been primarily proposed to assess clustering models, and thus show specific drawbacks for the assessment of biclustering models, particularly when these models are associated with biclusters with a high number of overlapping rows or columns. To surpass these problems, the CE metric can be considered, although it does not provide further detail on whether the observed mismatches between the discovered and hidden set biclusters is either due to poor coverage or precision. Although the FC metric has been arguably defended, it suffers from two drawbacks. First, FC strongly penalizes solutions with a high number of (similar) biclusters. This type of solutions are commonly found in exhaustive biclustering methods where sets of similar biclusters (associated with a single hidden bicluster) are discovered due to background noise. Second, the FC score favors the assessment of biclustering methods that required a pre-fixed number of biclusters (commonly comparable with the number of hidden biclusters). The MS scores are not sensitive to these problem and decompose the accuracy of the model in terms of its coverage and precision. However, they are separately applied over rows and columns, failing to provide an integrative view. To address this issue, we propose an extension of the MS scores (Eq.2.9), referred as revised match scores (RMS), a square-root of the overlapping area between the closest biclusters in order to compute accuracy scores comparable with MS.

$$\mathbf{RMS}(\mathcal{B},\mathcal{H}) = \frac{1}{|\mathcal{B}|}\Sigma_{(I_1,J_1)\in\mathcal{B}} max_{(I_2,J_2)\in\mathcal{H}} \sqrt{\frac{|I_1\cap I_2|}{|I_1\cup I_2|}\frac{|J_1\cap J_2|}{|J_1\cup J_2|}} \tag{2.9}$$

These objective metrics should be complemented with subjective metrics when assessing biclustering models on real data. Due to the biases and incompleteness associated with performance views gathered from merit functions, we suggest their replacement by domain-driven scores whenever possible.

2.3 Performance Views for Local Descriptors learned from Structured Data

Depending on the input data structure, different local descriptors can be considered. In line with the scope of our thesis (see *Section I-1.1*), on of our primary goals is placed on learning flexible models from cube data (multivariate time series). In line with the needs to robustly assess these different types of models, the following subsections compare loss functions for the adequate learning of triclustering and cascade models.

2.3.1 Triclustering Models

Informative regions from real-valued cube data (Def.I-1.4) can be flexibly approximated with triclustering models. These models are often learned for longitudinal experiments in which observations are evaluated under a set of conditions at several time points. In these data contexts, the triclustering task aims to discover subsets of observations, features and time points (referred as triclusters) with high homogeneity and statistical significance (Def.I-1.11). Similarly to biclustering models, triclustering models can be assessed in the presence and absence of hidden triclusters depending on whether synthetic or real data is available. As the study of triclustering in this thesis is seen as an intermediary step for the delivery of cascade models, the provided state-of-the-art evaluation metrics are covered with more brevity.

Objective Metrics. A tricluster can be seen as a bicluster with sustained homogeneity across a subset of overall time points. As such, and due to the additional fact a tricluster may show temporal misalignments, the commonly applied loss functions simply evaluate the ability to recover a given subset of observations and features. Under this assumption, all the previous objective metrics proposed for biclustering models can be directly applied for this end, including clustering metrics and alternative match scores. However, by disregarding the time dimension, there is no way to validate whether a triclustering algorithm can effectively recover the time points of planted triclusters and correctly model temporal misalignments. This need is particularly relevant when cascades of similar triclusters are observed along time. In this context, although clustering metrics (such as entropy, recall and precision) and Jaccard-based match scores can be applied along the three dimensions of a tricluster, they lead to voluminous results. To surpass this drawback, new similarity metrics have been proposed [263], including an extension of the clustering F-measure (referred as the E4SC metric) and their recall-precision components [273]. An extension of the provided RMS, referred as RMS3 (Eq.2.10), is proposed to offer a more adequate penalization of non-matched regions between triclusters.

$$\mathbf{RMS3}(\mathcal{B},\mathcal{H}) = \frac{1}{|\mathcal{B}|}\Sigma_{(I_1,J_1,K_1)\in\mathcal{B}} max_{(I_2,J_2,K_2)\in\mathcal{H}} \sqrt[3]{\frac{|I_1 \cap I_2|}{|I_1 \cup I_2|}\frac{|J_1 \cap J_2|}{|J_1 \cup J_2|}\frac{|K_1 \cap K_2|}{|K_1 \cup K_2|}} \tag{2.10}$$

Subjective Metrics. In the absence of knowledge regarding the local regularities of a given dataset, three major groups of loss functions can be identified: simple statistics (including tricluster's coverage and fluctuations [710]), merit functions, and domain-driven scores.

Merit functions for biclustering have been extended for triclustering [280]. An illustrative example is the extension of MSR, referred as mean square residue 3D [281]. By seeing each observation in a cube as a multivariate time series, the task of defining adequate merit functions for triclustering can be seen as the task of finding similarity measures between the observations (defining a real-valued matrix) of a given tricluster, which can be given by Euclidean distances, Pearson correlation coefficients, and residue-based functions. Yet the majority of these similarity measures ignore the fact that triclusters may be inferred from regions with temporal misalignments across observations, which may lead to non-trivial forms of coherency across observations [12]. For this end, merit functions can be formulated based on the proximity between pairwise observations from a given tricluster, where an observation is defined by a matrix (subset of features and time points). An illustrative measure for this end is planar mean residue similarity (PMRS) [12]. PMRS ranges from 0 (no correlation) to 1 (perfect correlation). Assuming a_{iJK} to be the average value for observation $\mathbf{x}_i$, Eq.2.11 illustrates this metric for $\mathbf{x}_1$ and $\mathbf{x}_2$ observations.

$$PMRS\,(\mathbf{x}_1,\mathbf{x}_2) = \frac{\sum_{j=1}^{m}\sum_{k=1}^{p}|(a_{1jk}-a_{1JK})-(a_{2jk}-a_{2JK})|}{2\times max(\sum_{j=1}^{m}\sum_{k=1}^{p}|a_{1jk}-a_{1JK}|,\sum_{j=1}^{m}\sum_{k=1}^{p}|a_{2jk}-a_{2JK}|)} \tag{2.11}$$

Understandably, the given *domain-scores* for biclustering remain valid for the triclustering task. Meaningful labels extracted from knowledge repositories, semantic sources and literature can be used to tag observations and (possibly) features of a cube. The analysis of their enrichment for each tricluster can then be used to validate and characterize the relevance of a triclustering model for a given domain.

2.3.2 Cascade Models

A cascade is a composition of modules (Def.I-1.12) related through temporal dependencies, either parallel or sequential. The modules within a cascade correspond to triclusters with contiguous time points and supported by an identical subset of observations. As the task of learning such cascades from multivariate time series is novel, there are not yet contributions on how to evaluate cascade models.

Basics 2.3 Cascades in biological domains

In biological domains, cascades can be learned from multiple gene expression time series with m multivariate order corresponding to the number of genes. In this context, a cascade is defined by a chain of regulatory modules with temporal dependencies, capturing the responses of downstream regulatory genes to concentrations of transcription factors produced by upstream regulatory genes. Cascades are initiated when cells are faced with a new condition, such as starvation [483], infection [484], stress [238], drug inception [567] or change in health state [117]. A (regulatory) cascade can be seen as a specialization of a gene regulatory network. A gene regulatory network is a model of the progression of transcriptional regulatory states in time defined by a set of regulatory interactions between groups of genes and of their transcription factors.

Understandably, the surveyed metrics for triclustering models [310, 263], as well as alternative similarity metrics for frequent motifs and other temporal patterns possibly accounting for misalignments [176] can be consider to evaluate the discovered modules within a cascade. However, despite the maturity of these metrics, they are not prepared to handle a matched pair of cascades with a distinct number of modules. Furthermore, they cannot assess the adequacy of the modeled dependencies between modules. In order to answer these challenges, we propose new metrics to assess cascades in the presence of synthetic or real data.

Metrics for Synthetic Data. Assuming an underlying ground truth given by planted cascades, similarity metrics sensitive to the correctness of dependencies between modules within a cascade are required for effective assessments. Below, we survey two major groups of loss functions and propose a new metric. First, three loss functions to assess distinct types of (regulatory) causality between (regulatory) modules were initially proposed by Marbach et al. [437] to measure the ability to learn distinct relations between the building blocks of networks (termed network motifs [456, 583]). These loss functions measure the: 1) fan-out error to assess co-regulation of multiple outputs ($\mathbf{B}_1 \Rightarrow \{\mathbf{B}_2,\mathbf{B}_3\}$), 2) fan-in error to assess co-regulation from multiple inputs ($\{\mathbf{B}_2,\mathbf{B}_3\} \Rightarrow \mathbf{B}_1$) and 3) cascade error to assess the ability to identify shortcuts, occurring when either indirect regulation ($\mathbf{B}_1 \Rightarrow \mathbf{B}_2 \Rightarrow \mathbf{B}_3$) or non-existing causality ($\mathbf{B}_1 \nRightarrow \mathbf{B}_3$) is misinterpreted as direct regulation ($G_1 \Rightarrow G_3$).

Second, additional relevant metrics have been proposed [452, 438], including sensitivity and specificity views of the learned regulatory cascades [339]. Nevertheless, as they were proposed in the context of Bayesian learning functions, cannot be easily generalized for alternative learners.

Let the set of hidden and discovered cascades be $\mathcal{H}$ and $\mathcal{R}$, respectively. To combine the advantages from triclustering score and the previous loss functions, we propose the use of a new integrative match score, referred as cascades match score (CMS), to measure the ability to cover all of the hidden set of cascades $CMS(\mathcal{H},\mathcal{R})$ (completeness) and to retrieve only the set of hidden cascades $MS(\mathcal{R},\mathcal{H})$ (precision). Assuming that $MS(R_1,R_2)$ measures the matching between two cascades, the target CMS score is given in Eq.2.12. A simplistic matching between cascades can be given by seeing the set of modules from R_1 and R_2 ($\mathcal{B}_1$ and $\mathcal{B}_2$ respectively) as two sets of triclusters and then testing their Jaccard-based differences $MS(\mathcal{B}_1,\mathcal{B}_2)$ (Eq.2.10), subspace clustering error, or Fabia consensus $FC(\mathcal{B}_1,\mathcal{B}_2)$ (Eq.2.6).

$$CMS(\mathcal{R},\mathcal{H}) = \frac{1}{|\mathcal{R}|}\Sigma_{R_1\in\mathcal{R}} max_{R_2\in\mathcal{H}} MS(R_1,R_2) \tag{2.12}$$

However, to be able to test causality, a cascade needs to be seen as a sequence of modules with well-defined precedences. Illustrating, consider that data is given by expression time series, where observations correspond to samples, assume a cascade R with three modules $\{\mathbf{B}_1,\mathbf{B}_2,\mathbf{B}_3\}$ sharing $\mathbf{I}$ observations, occurring during intervals of time points (possibly) described by the probability functions $\{\delta_1,\delta_2,\delta_3\}$, containing a compact number of features ($\mathbf{J}_1{=}\{y_{11},y_{12}\},\mathbf{J}_2{=}\{y_21\},\mathbf{J}_3{=}\{y_{31},t_{32}\}$), and related through constraints $\{\mathbf{B}_1,\mathbf{B}_2\}$ and ($\mathbf{B}_1{\Rightarrow}\mathbf{B}_2$). The sequential representation of R, seq_R, is $seq_R{=}(y_{11}y_{12}y_{21})(y_{31}y_{32})$, a sequence of itemsets where each itemset is a set of features' identifiers. In gene expression time series, observations and features are respectively given by samples and genes. The focus here is on finding groups of genes with coherent regulatory behavior on a minimum subset of features. In this context, the sequential representation defines frequent precedences and occurrences among coherently expressed genes. Based on this sequential representation, we propose a new score, referred as causal-cascade match score (C2MS), whose computation is given by Algorithm 2. C2MS is inspired on principles for aligning (biologic) multivariate sequences [436], thus offering a robust way to compare the similarity between the sequential representations of two cascades.

By accounting for the alignment of both co-occurrences and precedences, this score meaningfully combines fan-in, fan-out and cascade errors. Additionally, completeness can be seen as a measure of sensitivity when mscore (line 6) is given by $(G_i \cap G_j)/G_i$ and of specificity when the mscore is given by $(G_i \cap G_j)/G_j$.

Algorithm 2: *C2MS*(R_1, R_2): match score between two cascades based on sequence alignments.

```
index ← 0;
result ← 0;
foreach $J_i \in seq_{R_1}$ /*$J_i$ is the $i^{th}$ itemset of $seq_{R_1}$ */ do
    max ← -1;
    foreach $J_j \in seq_{R_2}$ starting at index do
        mscore ← $\frac{J_i \cap J_j}{J_i \cup J_j}$;
        if $mscore \leq max$ then break;
        max ← mscore;
    index ← j;
    result ← result + max;
result ← result / $|R_1|$; /* divided by the number of modules */
```

Finally, this score can be further enriched to weight temporal mismatches between cascades: mscore$\times P(\delta_i \cap \delta_j)$, where $P(\delta_i \cap \delta_j)$ measures the similarity between the distributions of time points.

Metrics for Real Data. The domain relevance of a cascade can be assessed using three major groups of metrics. First, merit functions to evaluate a cascade can be given by averaging the homogeneity of its modules, where the homogeneity of a module is given by a tricluster's merit function. The homogeneity of modules can be weighted by the size and positioning of the modules to calibrate overall expectations regarding the relevance of a cascade.

Second, by assessing individually each module against knowledge bases with functional enrichment analyzes, and by validating the dependency of temporally-related modules against literature. In biological domains, this latter task is critical to verify the causality of regulatory behavior between two modules. Causality can be tested based on the overlapping degree of the transcription factors that regulate each one of these modules.

Third, whenever possible, the learned models can be assessed against stable cascades inferred from different data-based and knowledge-based sources. In biological domains, this can be accomplished by testing the learned cascades against stable gene regulatory networks (see *Basics 2.3*), assuming that these networks are not used as background knowledge to guide the learning target task.

Additional lessons can be retrieved from existing efforts to compare methods aiming to learn variants of the proposed cascades (mainly from univariate time series) [455, 39, 439, 604, 118]. Despite the utility of the covered views, since domain knowledge is still largely incomplete and prone to biases, results from real data should be always complemented with assessments in synthetic data using the proposed CMS metric (Eq.2.12 parameterized with C2MS score given in Algorithm 2).

2.4 Assessment Methodology

We propose a methodology for assessing biclustering, triclustering and cascade learning methods according to three major decision axes. The first axis concerns the target data. Multiple synthetic data should be generate with varying size, dimensionality and global regularities, and with planted regions with varying structure, coherency, noise and overlapping degree. Real data should be additionally collected in line with the purpose of the target methods. The second axis concerns the selected methods and parameterizations to establish comparisons. The compared methods may rely on distinct assumptions that lead to a different number, size, positioning of biclusters/triclusters/cascades/patterns, yet the results should be always analyzed in the context of their assumptions. Illustrating, the comparison of two biclustering methods should take into consideration that one method is, for instance, purposefully focused on modeling biclusters with constant values with tolerance to noise, while the other method aim to purposefully learn biclusters with flexible coherencies without tolerance to noise. Finally, the third axis concerns the performance estimators.

For this aim, the estimators proposed in *Section 2.1* should be parameterized with the loss functions discussed in the *Sections 2.2* and *2.3* to infer robust performance guarantees. The surveyed and proposed loss functions target different aspects of accuracy and domain relevance, and should be naturally complemented with estimators of computational time and memory efficiency. In particular, the study of phase transitions for different loss functions and data assumptions is particularly relevant to reveal the true behavior of the proposed/compared methods. A result of the exploration of these axes for robust assessments, is the proliferation of results. To minimize this danger, the proposed principles in the previous chapter to bound and compare performance from multiple settings can be considered.

The proposed assessment methodology offers robust performance guarantees that can be easily extended to accommodate further contributions, including: new principles to generate synthetic data (*Chapter 3*), and new loss functions to specifically measure the statistical significance of the target models (*Book V*).

2.5 Concluding Note

This chapter compared state-of-the-art loss functions for local descriptive models learned from both tabular and structured data contexts, and placed a set of principles (together with the proposal of new loss functions) to guarantee robust and informative performance views.

The assessment methodology proposed in the previous chapter was here extended towards descriptive models under an adequate sampling method to collect estimates. As a result, principles such as the introduced statistical tests, smoothing factors, and inference of guarantees from flexible data settings became applicable to descriptive models, including biclustering, triclustering and cascade models, as well as global descriptors.

3
Synthetic Data Generation for Robust Assessments

Hundreds of algorithms in the fields of biclustering, pattern mining, network analysis, multivariate data analysis and triclustering have been proposed in the last few years with the goal of learning local descriptive models for biomedical and social data analysis. Each algorithm shows unique specificities that can turn its solutions significantly different from the solutions outputted by peer algorithms aiming to solve the same task. In this context, a robust assessment of their performance is required. Although the previous chapter proposed a set of evaluation metrics towards this end, there are not yet consensual data benchmarks for their evaluation [536, 503, 91, 475, 516, 202]. Real data is insufficient for their assessment since there is no ground truth to evaluate the recall and precision of the discovered regions (e.g. false positive discoveries are not penalized). In fact, assessments based on homogeneity criteria and domain-driven relevance (such as enrichment p-values against terms from knowledge bases) may not be indicative of their true performance [503, 310]. These facts prevent a clear understanding of the pros and cons of existing methods.

Furthermore, although the use of synthetic data is critical to provide objective performance views, existing data benchmarks[1] suffer from some of the following four major problems: 1) biases towards specific approaches (often the case when a variant of the optimized merit function is used to plant regions) [503, 59, 324]; 2) restrictions associated with the positioning of the regions of interest (fixed number, non-variable size and non-overlapping conditioning), size and dimensionality of the dataset, distribution of background values, and form and degree of noise [536, 324]; 3) absence of planted regions with flexible coherency [310, 202]; and 4) no plaid effects (meaningful composition of regions on their overlapping areas) [324, 536, 303]. Adding to these observations, there is a scarsity of data benchmarks for the analysis of (high-dimensional) multivariate time series and multi-sets of events with local regularities.

When moving from descriptive to classification tasks, similar limitations are observed. Although the assessment of classifiers is less subjective in the presence of labeled real data, two major challenges remain. First, there is an inherent difficulty to systematically explore how the algorithm behavior for data with varying regularities. Second, although some synthetic data benchmarks were proposed (surveyed in *Section II-1.3.1*) to surpass this problem, there is an overall focus on global regularities and therefore a lack of data benchmarks able to plant local regularities.

To surpass these limitations and integrate recent dispersed contributions for synthetic data generation, we propose three major algorithms able to effectively generate tabular, structured and labeled data. These generators provide the unique possibility to easily parameterize the properties of these data contexts, enabling assessments that tackle the biases and restrictions of existing methods. As such, this chapter proposes four major contributions:

- new generator of matrix and network data, BiGen (Bicluster data Generator), offering the unique possibility to flexibly parameterize: data size, dimensionality and regularities global and the planted biclusters according to their structure (number, size, positioning, overlapping effects) and homogeneity (coherency assumption,

[1] http://bmi.osu.edu/hpc/software/bibench/bibench.html
http://www.tik.ee.ethz.ch/sop/bimax/
http://www.bioinf.jku.at/software/fabia/benchmark.html

coherency strength and quality) without biases towards the specificities of existing methods;

- first generator of three-way time series, referred as CascadeGen (Cascade data Generator), able to adequately plant complex and diverse cascades of responses with temporal and structural misalignments across observations and varying forms of homogeneity;
- new generator of labeled (tabular and structured) data, referred L2Gen (LabeLed data Generator), introducing the possibility to: parameterize the number of classes and their imbalance, combine global and local class-conditional regularities, and define their discriminative power;
- generation procedures that guarantee that the computational complexity is approximately linear on the size of the matrix size and number of planted regularities, making simulation of arbitrarily large datasets scalable;
- default data benchmarks that preserve the statistical properties of experimental biological data.

Accordingly, this chapter is divided in three major sections. *Sections 3.1*, *3.2* and *3.3* respectively describe BiGen, CascadeGen and L2Gen, the proposed generators of biclustering data, multivariate time series data and labeled data with local regularities. Empirical evidence from the application of these generators and associated implications are synthesized in *Section 3.4*.

Disclosure Note. As some of the concepts provided here will be expanded throughout *Books III-VI*, this chapter can be complementarily viewed as a structured introduction to the challenging nature of learning local models from tabular and structured data with varying complexities.

3.1 Generation of Synthetic Biclustering Data

In this section, we tackle the problem of adequately generating synthetic data for revealing the true performance of biclustering algorithms. For this aim, *Section 3.1.1* provides the background on the properties of biclustering models. *Section 3.1.2* surveys previous efforts for generating biclustering data and their limitations. *Section 3.1.3* introduces BiGen, describing its generation procedures to affect . Finally, *Sections 3.1.4* and *3.1.6* show how BiGen can be used to generate data benchmarks and to guarantee the soundness of the generated data.

3.1.1 Background

Biclustering allows the discovery of regions of interest from real-valued matrices, each region defining a subset of observations (rows) showing a coherent pattern on a subset of the overall features (columns). According to Def.I-1.10, given a matrix with $\mathbf{X}=\{\mathbf{x}_1,..,\mathbf{x}_n\}$ rows, $\mathbf{Y}=\{\mathbf{y}_1,..,\mathbf{y}_m\}$ columns, and elements $a_{ij} \in \mathbb{R}$ relating i^{th} row and j^{th} column: a biclustering model is a set of biclusters $\mathcal{B}=\{\mathbf{B}_1,..,\mathbf{B}_s\}$, where each *bicluster* $\mathbf{B}_k=(\mathbf{I}_k \subseteq \mathbf{X}, \mathbf{J}_k \subseteq \mathbf{Y})$. satisfies specific *homogeneity* and *significance* criteria. These concepts were instantiated in *Basics I-1.8*.

In the absence of restrictions to the biclustering task, the **homogeneity** criteria determines the structure, coherency and quality of biclustering solutions. The homogeneity of a biclustering model is commonly guaranteed through a merit function. An illustrative merit function is the variance of the values in the bicluster. Merit functions are used to guide the search and provide a simple means to affect the structure, coherency and quality of biclusters.

The **structure** of a biclustering solution is essentially defined by the number, size and positioning of biclusters. Fig.3.1 illustrates varying structures of biclusters. Flexible structures are characterized by an arbitrary-high set of (possibly overlapping) biclusters. The number of biclusters can either be fixed, parameterized [134, 324], dynamically defined as function of data properties [311], or variable [616].

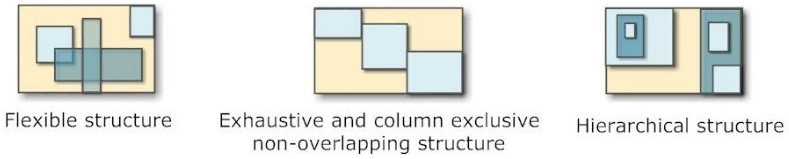


Figure 3.1: Biclustering models with varying types of structures.

The **coherency** of a bicluster is defined by the observed correlation of values (**coherency assumption**) and by the allowed deviation from expectations (**coherency strength**). A bicluster can have coherency of values across its rows, columns or overall elements, where the values typically follow constant, additive, multiplicative and symmetric models [429]. More flexible coherence assumptions are given by order-preserving and plaid models [429]. Defs.3.1 and 3.2 formalize these coherency assumptions, and Def.3.3 introduces the complementary notion of coherency strength. *Bascis 3.1* illustrates alternative forms of coherency of biclusters.

Def. 3.1 Let the elements in a bicluster $a_{ij} \in (\mathbf{I}, \mathbf{J})$ have *coherency across rows* given by $a_{ij} = c_j + \gamma_i + \eta_{ij}$, where c_j is the expected value for column j, γ_i is the adjustment for row i, and η_{ij} is the noise factor.

Def. 3.2 The γ factors define the coherency assumption: **constant** when γ=0, **multiplicative** if a_{ij} is better described by $c_j\gamma_i + \eta_{ij}$, and **additive** otherwise. **Symmetries** can be accommodated on rows, $a_{ij} \times c_i$ where $c_i \in$ {1,-1}. **Order**-preserving assumption is verified when the values of rows induce the same linear ordering across columns. A **plaid** assumption considers the cumulative effect of the contributions from multiple biclusters on areas where their rows and columns overlap.

Given the illustrative additive bicluster $(\mathbf{I},\mathbf{J})$=({$\mathbf{x}_1, \mathbf{x}_2$},{$\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3$}) in $\mathbb{N}_0^+$ with coherency on rows, where $(\mathbf{x}_1, \mathbf{J})$= {1, 3, 2} and $(\mathbf{x}_2, \mathbf{J})$={3, 4, 2}. This bicluster can be described by $a_{ij} = c_j + \gamma_i$ with the pattern φ={c_1=0,c_2=2, c_3=1}, supported by two rows with additive factors γ_1=1 and γ_2=3. Additional instantiations of the introduced coherency assumptions are further provided in *Basics 3.1*.

Basics 3.1 Illustrating Non-Constant Coherency Assumptions

Given the discrete tabular dataset $\mathbf{A}_1$, Figures 3.2 and 3.3 identify biclsuters with non-constant coherency assumptions with at least two supporting observations and non noise. The multiplicative bicluster (green) has pattern φ={c_2=-1,c_3=2,c_5=-1,c_7=1} and adjustments {γ_1=1,γ_2=2,γ_4=2}. The additive bicluster (yellow) has pattern φ={c_2=-4,c_3=2,c_5=-4,c_7=0} and adjustments {γ_2=2,γ_3=0,γ_4=2}. The symmetric bicluster (blue) has φ={c_2=3,c_3=0,c_5=-1,c_6=2,c_7=-1} and symmetry on $\mathbf{x}_6$. Finally, the order-preserving (purple) is characterized by the permutation $\mathbf{y}_4 \leq \mathbf{y}_5 \leq \mathbf{y}_7 \leq \mathbf{y}_3 \leq \mathbf{y}_6 \leq \mathbf{y}_2$ and has a symmetry on $\mathbf{x}_6$.

Figure 3.2: Illustrative multiplicative (green) and symmetric (blue) biclusters in $\mathbf{A}_1$

	$\mathbf{y}_1$	$\mathbf{y}_2$	$\mathbf{y}_3$	$\mathbf{y}_5$	$\mathbf{y}_7$	$\mathbf{y}_4$	$\mathbf{y}_6$	class
$\mathbf{x}_1$	3	**-1**	**2**	**-1**	**1**	3	1	c_1
$\mathbf{x}_2$	1	**-2**	**4**	**-2**	**2**	0	0	c_1
$\mathbf{x}_4$	-3	**-2**	**4**	**-2**	**2**	2	-1	c_1
$\mathbf{x}_3$	1	-4	2	-4	0	-3	0	c_1
$\mathbf{x}_5$	1	**3**	**0**	**-1**	**-1**	-2	**2**	c_2
$\mathbf{x}_6$	-3	**-3**	**0**	**1**	**1**	4	**-2**	c_2

Figure 3.3: Illustrative additive (yellow) and order-preserving (purple) biclusters in $\mathbf{A}_1$

	$\mathbf{y}_1$	$\mathbf{y}_2$	$\mathbf{y}_3$	$\mathbf{y}_5$	$\mathbf{y}_7$	$\mathbf{y}_4$	$\mathbf{y}_6$	class
$\mathbf{x}_1$	3	-1	2	-1	1	3	1	c_1
$\mathbf{x}_2$	1	**-2**	**4**	**-2**	**2**	0	0	c_1
$\mathbf{x}_4$	-3	**-2**	**4**	**-2**	**2**	2	-1	c_1
$\mathbf{x}_3$	1	**-4**	**2**	**-4**	**0**	-3	0	c_1
$\mathbf{x}_5$	1	**3**	**0**	**-1**	**-1**	**-2**	**2**	c_2
$\mathbf{x}_6$	-3	**-3**	**0**	**1**	**1**	**4**	**-2**	c_2

Def. 3.3 Let $\overset{\frown}{A}$ be the amplitude of the range of values in a dataset $\mathbf{A}$. Given a dataset $\mathbf{A}$, the coherency strength is a range $\delta \in [0, A]$, such that $a_{ij} = k_j + \gamma_i + \eta_{ij}$ where $\eta_{ij} \in [-\delta/2, \delta/2]$.

The **quality** of a set of biclusters is defined by the type and amount of accommodated noise.

The statistical **significance** of a bicluster determines its deviation from expectations. Since significance is essentially a function of the size and coherency of the bicluster, we exclude this aspect from this section for the sake of simplicity. *Book V* further elaborates on this issue and revises BiGen to guarantee the plantation of significant biclusters.

Resulting from these observations, a real-valued matrix with local regularities can be described by a (multivariate) distribution of background values and a set of biclusters defined by their structure, coherency, quality and significance.

3.1.2 Related Work

Many studies place important principles for data generation [503, 91, 475, 516, 472]. However, the generated datasets are often undisclosed. Some of the few available biclustering data benchmarks include the ones provided with BiBench[2] [202], BiMax[3] [536], Fabia[4] [324], ISA [342] and BicPAM[5] [310] tools. BiMax provides biclustering data with varying degree of noise and overlapping, although it fails to explore the size of data, the number and shape of biclusters, and their coherency. Fabia data benchmarks explore varying coherencies, although they have a fixed number of rows, columns and biclusters, where the planted biclusters show heavy tail properties and no overlapping areas. BicPAM [310], and more recently BicSPAM[6] [311] and BiP[7] [303], make available data with planted biclusters with varying number, size, coherency and noise of biclusters. However, similarly to the previous benchmarks, no generator is made available and some of the synthetic settings do not resemble biological data. Contrasting, the data settings proposed by Gu and Liu [270] and by Segat et al. [573] force a proximity with real expression data by, respectively, approximating its regularities or relying on sampling methods. However, only a few parameters can be varied in both settings.

To our knowledge, the most complete generator of synthetic biclustering data up to date is BiBench [202]. BiBench allows the parameterization of the data size and number of biclusters, and the possibility to generate biclusters with shift and scale factors, deviations from expected values and simplistic overlapping conditions. Despite its relevance, BiBench suffers from some limitations. First, BiBench assumes biclusters to have approximated constant values across columns (in other words, each row have a unique expected value). In the context of gene expression, this means that a gene (row) in a bicluster needs to be up-regulated, undifferentiated or down-regulated across all conditions (columns) in the bicluster. When considering the transpose matrix, this results in the imposition that two genes (columns) in a bicluster need to have the same level of expression. Understandably, this prevents relevant scenarios where genes may show different yet correlated levels of expression. Therefore, it is desirable to allow non-constant values across columns with constant, shift, and scaling factors on rows. Second, BiBench does not consider specific types of coherency, including constant biclusters with symmetries on rows (to model putative regulatory modules with both activation and repression mechanisms) and order-preserving coherencies (to model regulatory modules with highly flexible yet coherent patterns of expression). Third, it does not provide the possibility to flexibly vary the shape of biclusters (biclusters with varying number of rows and columns) within a dataset. Fourth, BiBench does not include the possibility to control the coherency strength of the values in a bicluster since its background values are generated using Gaussian distributions. Fifth, the absence of a graphical interface and visualization mechanism for the generated data prevents its usability. Some other limitations, include the absence of options to: generate biclusters with contiguous columns for time series data analysis; generate symbolic data; plant noisy and missing elements; and create meaningul plaid structures where the properties associated with the overlapping areas are flexibly controlled.

3.1.3 Generating Biclustering Data

This section describes BiGen according the major generation procedures to control the properties of the planted set of biclusters according to their shape and size, coherency, positioning, and quality.

Core Parameters. BiGen is able to generate datasets with varying size, underlying distributions of (either real or symbolic) values, number and shape of biclusters. Key user-definable parameters include:

- data size ($|\mathbf{X}| \times |\mathbf{Y}|$) and background distribution of values (Gaussian and Uniform). The selection of a Nor-

[2] http://bmi.osu.edu/hpc/software/bibench/bibench.html
[3] http://www.tik.ee.ethz.ch/sop/bimax/
[4] http://www.bioinf.jku.at/software/fabia/benchmark.html
[5] web.ist.utl.pt/rmch/software/bicpam
[6] web.ist.utl.pt/rmch/software/bicspam
[7] web.ist.utl.pt/rmch/software/bip

mal distribution can be used to mimic expression data (where the majority of elements is non-differentially expressed) and network data (where the majority of interactions has low/residual weight);

- range of real-values $\widehat{A}$ (and eligible coherency strength δ) or, alternatively, number of ordinal symbols $|\mathcal{L}|$;
- the indication whether symmetries are allowed. Selecting symmetries are, for instance, useful to separate activation and repression mechanisms in biological data, profitability profiles in financial data, preferences in social networks and collaborative filtering data;
- number of biclusters;
- shape and size of biclusters: distribution of the number of rows and columns. Uniform distributions are made available to plant biclusters with dissimilar shapes (varying number of rows and columns). Alternatively, Gaussian distributions are made available for a more conservative variation of the number of rows on the generated set of biclusters;

Coherency Criteria. BiGen allows the parameterization of the coherency assumption (Def.3.2) and strength (Def.3.1) of biclusters. In real-valued data the *coherency strength*, δ, can be specified as a percentage of the inputted range of values. Contrasting, in symbolic data, the number of symbols implicitly defines the coherency strength (given $\mathbf{A}$, $\delta = \widehat{A}\ /|\mathcal{L}|$).

BiGen supports the generation of biclusters according to a wide-variety of *coherency assumptions*, including constant, additive, multiplicative, symmetric, order-preserving and plaid assumption (Def.3.2). In particular, the selection of additive, multiplicative and symmetric factors per bicluster and their distribution across rows (columns) assumes an underlying uniform distribution on the possible factors and rows (columns). Illustrating, consider a bicluster with 6 rows and 3 columns, multiplicative coherency on rows and possible factors $\gamma_i \in \{1,2,3\}$. In this scenario, the planted bicluster has in average 3 pairs of (randomly selected) rows with $\gamma_i{=}1$, $\gamma_i{=}2$ and $\gamma_i{=}3$ factors.

Furthermore, BiGen guarantees that the range of additive and multiplicative factors allowed for a given set of biclusters are uniformly distributed. Illustrating, consider a symbolic dataset with 6 symbols ($|\mathcal{L}| \in \{0..5\}$) and a biclustering model with 10 biclusters following an additive assumption with coherency across rows. Given a bicluster $(\mathbf{I},\mathbf{J})$ with 20 rows with pattern $\{0,4,3\}$ in the absence of noise ($\eta_{ij}{=}0$), this implies that only two additive factors can be planted for all 20 rows ($\{0,4,3\}$ and $\{1,5,4\}$), $\gamma_i \in \{0,1\}$. Since BiGen guarantees an adequate distribution of the allowed additive factors for the generated biclusters, a planted set 10 biclusters would in average include 2 biclusters where $\gamma_i \in \{0,..,5\}$, 2 biclusters where $\gamma_i \in \{0,..,4\}$, and so forth. Figure 3.4 illustrates two additive biclusters from a planted set with 10 additive biclusters where $\widehat{A}{=}4$ and $\delta{=}20\%$.

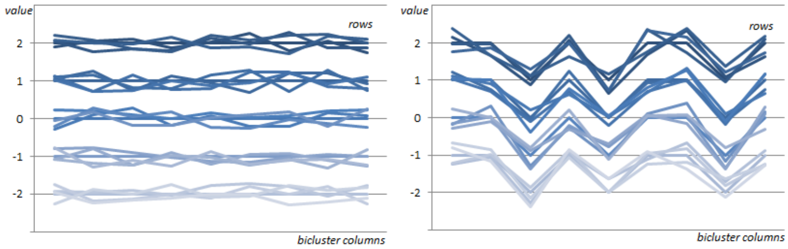


Figure 3.4: Illustrative additive biclusters with coherency on rows, δ=20%, varying ranges of adjustments and approximately uniform assignments of factors per row, $\mathbf{I} \sim U(20,40)$, $\mathbf{J} \sim U(8,20)$.

Order-preserving biclusters are planted by generating a set of vectors with arbitrary-degree of value distances, only the orderings are preserved across the vectors. These orderings can show a parameterizable degree of co-occurrences versus precedences based on the inputted coherency strength (see *Chapter 6* for further details).

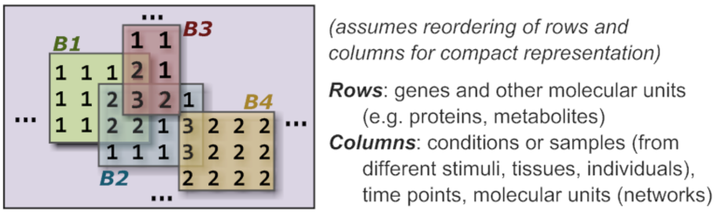
When constant or symmetric models are selected, BiGen additionally gives the possibility to select whether only *differential values* are relevant or not. When selecting differential values only, BiGen generates biclusters with values in the first and fifth quintiles of domain values in the presence of symmetries, and in the last third of domain values in the absence of symmetries.

Finally, BiGen allows the possibility of generating biclusters with contiguous columns (or rows). This is a necessary requirement to evaluate biclustering algorithms developed for the analysis of *time series* data [427].

Plaid Structure. To control the properties associated with the positioning of biclusters and meaningful modeling of their overlapping areas, BiGen introduces the possibility of planting plaid structures. These structures are particularly important for replicating the regularities of biological data and social networks [303]. A biclustering model following a plaid structure is a composition of K biclusters (layers), where θ_{ijk} specifies the contribution of each bicluster according to the following model of the matrix $\mathbf{A}$:

$$a_{ij} = \mu_0 + \sum_{k=0}^{K} \theta_{ijk}\rho_{ik}\kappa_{jk} \;\mid \theta_{ijk} = \mu_k + \alpha_{ik} + \beta_{jk} + \eta_{ijk}, \tag{3.1}$$

where ρ_{ik} and κ_{jk} are binary values defining, respectively, the membership of row i and column j in bicluster k. In this context, two biclusters with overlapping rows and columns are often referred as interacting biclusters. Figure 3.5 shows an illustrative additive plaid structure.



(assumes reordering of rows and columns for compact representation)

Rows: genes and other molecular units (e.g. proteins, metabolites)

Columns: conditions or samples (from different stimuli, tissues, individuals), time points, molecular units (networks)

Figure 3.5: Illustrative additive composition of constant biclusters.

To support the generation of flexible plaid structures, we provide the first systematic attempt to identify and formalize its major variables. For this aim, we propose the following four new concepts.

First, average number of interactions per bicluster: $\kappa = \frac{1}{K}\Sigma_{i=1}^{K}\left(\Sigma_{j=1}^{K} f(\mathbf{B}_i, \mathbf{B}_j)\right)$, where $f(\mathbf{B}_i, \mathbf{B}_j) = 1$ if $\mathbf{B}_i \cap \mathbf{B}_j \neq 0$, and 0 otherwise. This concept can be further extended to guarantee that subsets biclusters interact with each other but not with biclusters outside the group, $(\mathbf{B}_1 \cap \mathbf{B}_2 \neq 0 \wedge \mathbf{B}_1 \cap \mathbf{B}_3 = 0) \Rightarrow \mathbf{B}_2 \cap \mathbf{B}_3 = 0$. Illustrating, a solution with 20 biclusters and $\kappa \sim N(5, 0)$ has 4 subgroups with an average of 5 biclusters interacting with each other. When this number matches the total number of biclusters, all biclusters influence each other forming a complex model of interactions. When $K\%\kappa \neq 0$, BiGen plants K/κ groups of $\lfloor\kappa\rfloor$ biclusters with overlapping areas and an additional group with $K\%\kappa$ biclusters.

Second, distribution of the number of layers within a set of interacting biclusters: $(\kappa - 2)\phi + 2$ where $\phi \in [0, 1]$. Accordingly, when $\phi \sim N(100\%, 0)$, this implies that the observed plaid effects are a composition of the contributions from all the κ biclusters and $\phi \sim N(0\%, 0)$ means that plaid effects derive from pairwise overlaps.

Third, composition function of the underlying contributions. According to (3.1), the observed coherency on the overlapping areas is given by an additive composition function (see Figure 3.5), meaning that the value of an element in the matrix is either given by the background value or by the sum of the values of the biclusters that contain that element. Multiplicative and interpoled functions are also available. The multiplicative function, $\Pi_{k=0}^{K}\theta_{ijk}\rho_{ik}\kappa_{jk}$ can be used for settings where each contribution has a catalyzing effect on the value, while the interpoled function, $(\sum_{k=0}^{K}\rho_{ik}\kappa_{jk})^{-1}\sum_{k=0}^{K}\theta_{ijk}\rho_{ik}\kappa_{jk}$ (average of contributions) is well-suited for settings where the overlapping areas are neither characterized by cumulative nor scaling effects.

Fourth, average weight per additional contribution, ν (by replacing θ_{ijk} with $\nu\theta_{ijk}$ in (3.1)), and allowed noise on the composed value, ϵ (by replacing θ_{ijk} with $(\theta_{ijk} \pm \epsilon)$ in (3.1)). Weights are critical to model non-linear cumulative effects and their value can be assigned based on the number and contributions of the overlapping biclusters. Illustrating, considering ν=0.8 and an element with contributions $\{\mu_1, \mu_2, \mu_3\}$ from three biclusters. In this context, the expected value for such element would be bounded by $\sum_{i=1}^{3}\nu\mu_i \pm \epsilon$ and $\sum_{i=1}^{3}\mu_i \pm \epsilon$.

As such, BiGen makes available the following key additional parameters to compose plaid structures:

- overlapping degree between biclusters (Gaussian distribution) as an expected percentage of: 1) overlapping rows per bicluster and, 2) overlapping columns per bicluster;
- average number of interacting biclusters, κ, and the distribution of the overlapping areas, ϕ;
- function for composing contributions, f;
- average weight per additional contribution, ν, and allowed noise on the composed value, ϵ.

Noise. BiGen allows the parameterization of the quality of the planted biclusters and data background values by introducing the possibility to:

- introduce parameterizable percentage of noisy elements in the matrix;
- create deviations on the generated values according to a Uniform distribution given the allowed differences (default setting) or Gaussian distribution given the standard deviation. Illustrating, a Uniform deviation of 20% for a dataset with $\widehat{A}=2$ implies noise factor $\eta_{ij} \sim [-0.4, 0.4]$;
- plant a parameterizable degree of missing values either randomly distribution across all elements or with deviations towards specific subsets of columns and rows.

3.1.4 Data Benchmarks with BiGen

BiGen makes available default settings for the prompt generation of data benchmarks, each with parameterizations targeting a specific aspect of performance: data properties, number and size of biclusters, allowed coherencies, positioning/locality, or degree of noise. With the exception of the aspect under variation, the remaining aspects are associated with parameterizations driven from experimental expression data. Table 3.1 describes these settings.

The provided data settings assume a high-dimensionality and coherency across columns. In expression data, coherency is commonly assumed across genes (features/columns) and the presence of labels is frequently associated with samples or patients (observations/rows). Nevertheless, when ignoring labels, the data settings from Table 3.1 can be equally rewritten by transposing matrices and biclusters. Illustrating, the (n-100,m-1000)-space with planted biclusters with column-based coherency and $|\mathbf{I}| \sim U(10,12)$ and $|\mathbf{J}| \sim U(40,60)$ can be rewritten as a (n-100,m-1000)-space with planted biclusters with row-based coherency and $|\mathbf{I}| \sim U(40,60)$ and $|\mathbf{J}| \sim U(10,12)$.

The selected number of rows and columns per bicluster follows a Uniform distribution using the ranges in Table 3.1. By default, we allow for overlapping biclusters according to a plaid structure (10% of overlapping elements) and noise factors (up to 20% of the range of values), which can difficult the recovery of the planted biclusters.

♯columns×♯rows (♯rows×♯columns) assuming coherency on rows (columns)	100×40	500×70	1000×100	2000×200
Number of hidden biclusters	3	5	10	20
Number of rows (columns) in biclusters	[6,8]	[8,10]	[10,12]	[12,15]
Number of columns (rows) in biclusters (constant γ=1, non-constant $\gamma \in [1,2]$)	[14×γ,20×γ]	[25×γ,40×γ]	[40×γ,60×γ]	[60×γ,80×γ]
Area of biclusters	8.9%×γ	4.2%×γ	5.5%×γ	4.7%×γ

Settings for the 1000×100 data space (default settings in bold):

- Coherency strength δ={5%, 10%, 15%, **20%**, 33%} (or symbols $|L|$={20, 10, 7, **5**, 3})
- Values' deviation in {0, δ/2, δ, 2δ}, and degree of noisy and missings elements in {0%, **2%**, 5%, 10%}
- Overlapping degree (rows and columns) θ={0, 0.1, **0.2**, 0.4}, composition f={**sum**, product, weighted} with incremental weight ν={**1**, 0.7, 0.4} and noise ϵ={**0.1**, 0.2}
- Number of interactions per bicluster κ={0.5K, **0.3K**, 0.1K} with ϕ={1, **0.8**, 0.5}
- Coherencies={**constant**, additive, multiplicative, symmetric, order-preserving, plaid} with {**no**,yes} differential values and **no** contiguous columns

Table 3.1: BiGen parameters for the generation of default data benchmarks.

Due to the inherent higher complexity of learning biclusters under an order-preserving assumption and the observed efficiency bottlenecks from existing biclustering algorithms aiming to solve this task, BiGen provides a variant of Table 3.1 settings when the order-preserving coherency is assumed. As such, the 100×50, 500×10, 1000×200 and 2000×400 settings (with 3, 5, 10 and 20 biclusters) are respectively replaced by 100×30, 500×50, 1000×75 and 2000×100 settings (with 2, 3, 5 and 8 biclusters).

Significance Criteria. The provided data settings in Table 3.1 guarantee that the planted biclusters are statistically significance. In other words, given a specific data setting, the probability that a non-planted bicluster (with similar number of rows and columns and the same coherency assumption and orientation) emerges from background values is very low (see *Book V* and Tables 2.2-2.3 for further details).

An important observation is that there is a higher likelihood for background values to form a non-planted bicluster if the underlying coherency strength is looser and/or the underlying coherency assumption is flexible. Illustrating, the formation of large spurious biclusters (false positives) is more probable for data with two symbols (δ=50%) than for data with 10 symbols (δ=10%). Complementarily, order-preserving biclusters are more flexible than additive and multiplicative biclusters, which in turn are more flexible than constant biclusters. In this context, in order to guarantee that biclusters with more flexible coherency assumptions are statistically significance, BiGen weights their size by a factor $\gamma \geq 1$ based on their degree flexibility. From empirical evidence, the following weights are provided: order-preserving (γ=2), additive (γ=1.5 with symmetries and γ=1.4 otherwise), multiplicative (γ=1.3), constant (γ=1.2 with symmetries and γ=1 otherwise).

Performance Guarantees. For each of the listed settings we suggest the instantiation of 40 matrices: 20 real-valued and 20 symbolic. And for each of these sets, we suggest the variation of the generated background values by considering both Uniform and Gaussian distributions, together with the possibility of using symmetries by either allowing or avoiding negatives ranges of values. Illustrating for the symbolic case, 10 matrices are generated with background values following a Uniform distribution (5 respecting U(1,$|\mathcal{L}|$) and 5 respecting U(-$\frac{|\mathcal{L}|}{2}$,$\frac{|\mathcal{L}|}{2}$)) and 10 matrices with background values generated according to a Gaussian distribution (5 respecting N($\frac{|\mathcal{L}|}{2}$,$\frac{|\mathcal{L}|}{6}$) and 5 respecting N(0,$\frac{|\mathcal{L}|}{6}$)).

Efficiency Limits. BiGen also includes on its data benchmarks an additional set of data settings for the purpose of testing efficiency limits. This is done by: 1) planting 10 biclusters to occupy 5% of the elements from the input data space (following a Gaussian distribution on the number of rows and columns) assuming a column-based coherency, 2) fixing a dimensionality of m=10.000 (magnitude of the human genome), and by 3) varying the total number of rows n of the generated data matrix to assess the scalability of the algorithms.

3.1.5 Generating Sparse and Network Data

According to Def.I-1.3, network data is typically given by a weighted graph defined by a set nodes $\mathbf{X}$=$\{\mathbf{x}_1,..,\mathbf{x}_n\}$ (observations and features) where $\mathbf{x}_i \in \mathbb{R}^n$ and $a_{ij} \in \mathbb{R}$ interactions between nodes $\mathbf{x}_i$ and $\mathbf{x}_j$. Contrasting, with tabular data, network data is typically characterized by an inherently high sparsity due to the high number of missing interactions a_{ij}. Although a parameterizable number of missing elements satisfying certain locality constraints could be planted under the previously introduced settings, the generated datasets would no longer resemble properties of real-world data. For this reason, we propose a generator of network with default settings respecting the commonly observed topological statistics of biological networks. The major variables of this generator are:

- number of nodes, density (average fraction of overall interactions per node) and distribution of their connectivity (in order to guarantee the emergence of nodes with above average connectivity to simulate hubs);
- distribution of the weight of interactions for real networks (Poisson/Uniform assignment of (non-)strictly positive ranges of values revealing the association strength of two nodes) and of labels for symbolic networks (Uniform/Gaussian assignment of labels revealing node function such as activation/repression);
- number, size (Uniform distribution on the number of nodes), shape (imbalance on the size of the disjoint sets of each subgraph), overlapping, coherency assumption (dense, constant, symmetric, plaid, order-preserving)

and coherency strength of the planted biclusters;

- degree of noisy interactions (from 0% to 20%).

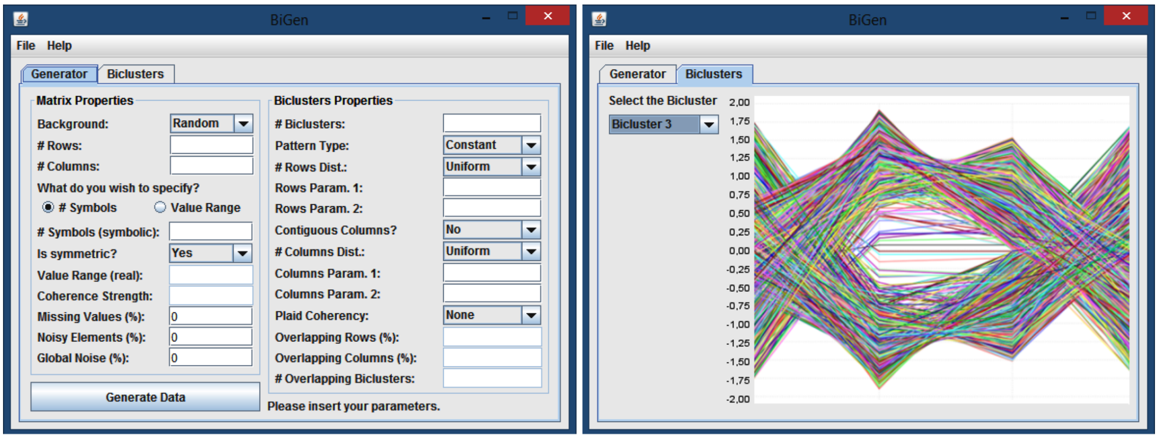
Table 3.2 describes the proposed variations of some of these parameters for the assessment of learning methods for network data analysis. These variables assume that the generated network is homogeneous (single type of nodes). Nevertheless, the generator of network data also introduces the possibility to generate heterogeneous network through the specification of the size and connectivity between two distinct sets of nodes.

	Network nodes (10% density)					Network density (2000 nodes)			
	200	500	1000	2000	10000	1%	5%	10%	25%
♯ Hidden modules	5	10	15	20	30	3	5	10	20
♯ Nodes per module	[20,30]	[30,40]	[40,50]	[50,70]	[100,140]	[50,70]	[50,70]	[50,70]	[50,70]
% Interactions in modules	19,5%	12,2%	7,6%	4,5%	1,1%	22,5%	9,0%	4,5%	2,3%

Table 3.2: Suggested synthetic benchmarks for network data analysis (compact view).

3.1.6 Software Description

BiGen software offers both graphical and programmatic interfaces. Figure 3.6 provides an illustrative snapshot of the graphical interface of BiGen. As illustrated, BiGen provides a parametric environment to easily control the properties and complexity of generated biclustering data. This is done without hampering the usability of BiGen software, since both default settings and seeds for the generation of benchmark data are made available.



Matrix Properties

Background: Random
Rows:
Columns:
What do you wish to specify?
● # Symbols ○ Value Range
Symbols (symbolic):
Is symmetric? Yes
Value Range (real):
Coherence Strength:
Missing Values (%): 0
Noisy Elements (%): 0
Global Noise (%): 0

Biclusters Properties

Biclusters:
Pattern Type: Constant
Rows Dist.: Uniform
Rows Param. 1:
Rows Param. 2:
Contiguous Columns? No
Columns Dist.: Uniform
Columns Param. 1:
Columns Param. 2:
Plaid Coherency: None
Overlapping Rows (%):
Overlapping Columns (%):
Overlapping Biclusters:

Please insert your parameters.

Select the Bicluster: Bicluster 3

Figure 3.6: Snapshots of BiGen: data generation and visualization.

BiGen's graphical interface offers three additional benefits. First, it guarantees the soundness of the generation process by either disabling options when certain parameters are selected (e.g. discovery of symmetric models for a strictly positive range of values) or by displaying messages of error (e.g. an error message is displayed when additive models with symmetries are used over discrete data with an even number of items). Second, after the data is generated, BiGen provides the possibility to visualize the generated data (heatmap) and the graphical representation of biclusters. In Figure 3.6, a bicluster following a multiplicative assumption and symmetries is illustrated. Finally, BiGen supports the exportation of generated data according to a wide-variety of formats.

Alternatively to the graphical interface, BiGen makes available a programmatic interface that can be used to extend the generation procedures to support, for instance, the generation of discriminative biclusters in the presence of labeled data. Understandably, an in-depth description of the properties associated with the BiGen software is out of the scope of this dissertation, and thus we redirect the reader to its tool tutorial published in web.tecnico.ulisboa.pt/rmch/software/bigen.

3.2 Generation of Structured Data

When the previous tabular datasets are seen as occurrences/snapshots along a third dimension (the temporal axis), temporal data structures need to be generated. In particular, we consider two types of data structures: 1) three-way time series, where observations are described by a multivariate time series with m features and p time points; and 2) multi-sets of events, where observations are described by an arbitrary-number of timestamped events (possibly) associated with distinct attributes (event types). In fact, as shown throughout *Book IV*, many other data structures, such as relational and multi-dimensional structures, can be mapped into one of the two previous data formats.

For the aim of adequately generating structured data, we first visit their common forms of local regularities and then propose effective procedures for their plantation. Relevant regions in three-way time series are given by triclustering and cascade models (Defs.I-1.11 and I-1.12), while in multi-sets of event (Def.I-1.6) are given by arrangements of informative events.

3.2.1 Generation of Three-Way Time Series with Planted Cascades

A three-way time series (Def.I-1.4) is a three-dimensional matrix defined by a set of observations, features and time points. As formalized in Def.I-1.12, a *cascade* R is a composition of triclusters (modules) sharing a significant number of observations $\{\mathbf{B}_1, .., \mathbf{B}_l\}$ through either parallel $\{\mathbf{B}_i, \mathbf{B}_j\}$ or sequential $\mathbf{B}_i \Rightarrow \mathbf{B}_j$ relations. Illustrating, in the presence of expression three-way time series, putative regulatory modules are associated with regions given by triclusters, and their meaningful composition reveals regulatory responses given by cascades. Given a three-way time series $\mathbf{A}$, the task of *modeling cascades* aims to learn a set of cascades $\{R_1, .., R_s\}$ satisfying specific criteria of homogeneity and statistical significance. An illustrative cascade model was provided in *Basics I-1.9*. A broader formal view of this task, together with an in-depth view of its applicability is provided in *Chapter IV-1*.

Despite the apparent formulation simplicity of a cascade model, the combinatorial complexity of this learning task is significantly higher than the biclustering task and further challenged by the presence of stochastic uncertainties, including:

- the possible presence of temporal shifts and scales [261] per module, together with the need to allow for flexible structures and coherencies. Figure 3.7 (purples boxes) illustrates these conditions;
- the inherent diversity of cascades and the possibility that paths within a cascade can rapidly branch, possibly leading to structural divergences between observations;
- the presence of temporal misalignments between observations. Figure 3.7 (red boxes) shows how these misalignments can be either associated with the duration or time distance between modules of a cascade.

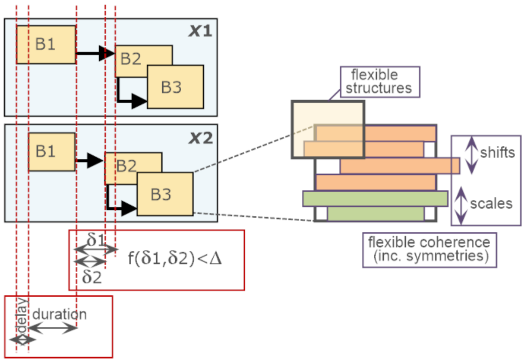


Figure 3.7: Inherent complexities associated with cascades typically observed in three-way time series.

Although synthetic 3TS data benchmarks are available in literature [101, 469, 188, 459, 282, 400, 557, 637], they are insufficient to robustly assess computational methods aiming to model cascades due to three major reasons. First, they are not able to plant flexible structures of modules with parameterizable homogeneity and significance criteria. The few attempts to plant motifs, triclusters and other patterns place restrictive constraints on the generation procedures. Second, and a natural implication from the previous observation, there are not yet synthetic data with planted local regularities given by cascade models. Finally, the available benchmarks do not offer the possibility to flexibly control the parameters controlling the generation of multivariate time series.

Below, we explore the parameters to affect both the core properties and stochasticity of three-way time series, and use them to generate data benchmarks resembling the regularities of gene expression time series.

Core Parameters. To assess the performance of methods able to answer the target task of learning cascades from 3TS, we allow the parameterization of the following set of major variables:

- number of observations ($|\mathbf{X}|$), features ($|\mathbf{Y}|$) and time points ($|\mathbf{T}|$);
- distribution of background values: either Uniform or Gaussian distributions together with smooth factors across contiguous time points to better reflect the pace of the change of values over time;
- number of planted of cascades s (number of distinct elicited frequent responses), as well as a distribution of their support/representativity (cascades may be elicited by subsets observations with different size);
- structural dependencies: distribution of the size (number of features and time points) and duration of the modules within each cascade. Uniform distributions of the size and duration of modules are proposed by default to guarantee the diversity of regulatory behavior;
- temporal dependencies: distribution of 1) the number of modules per cascade (Uniform distributions are suggested to test the ability to recover cascades with varying structures), 2) the type of dependencies per cascade, and 3) temporal distances associated with the dependencies;
- regulatory coherency strength ($|\mathcal{L}|$), assumption and presence of symmetries;
- degree and type of planted noise (planted deviation on the observed values or replacement by random values), and amount of missings.

Handling Stochasticity. For the adequate simulation of the inherent stochasticity of cascades from 3TS, CascadeGen allows for the plantation of bifurcated responses and introduces the possibility to plant different forms of temporal and structural misalignments with varying extent.

Bifurcations. CascadeGen supports a parameterized distribution of the number of fan-out relations within a cascade. Bifurcations are planted on the last occurring modules. Illustrating, consider a cascade with 4 precedences and 2 bifurcations. This setting implies the presence of 8 modules with the following sequential relations: $\{\mathbf{B}_1 \Rightarrow \mathbf{B}_2 \Rightarrow \{\mathbf{B}_3, \mathbf{B}_4\}, \mathbf{B}_3 \Rightarrow \{\mathbf{B}_5, \mathbf{B}_6\}, \mathbf{B}_4 \Rightarrow \{\mathbf{B}_7, \mathbf{B}_8\}\}$. CascadeGen guarantees the soundness of these assignments.

Furthermore, CascadeGen allows for the selection of the desirable type of bifurcation. There are two major types of bifurcations: 1) elicitation of parallel modules (no significant changes on the supporting observations), and 2) elicitation of mutually exclusive modules (implying a separation of the cascade's observations for each one of the bifurcated paths).

Temporal Misalignments. CascadeGen permits a parameterized definition of the form and extent of temporal misalignments (changes in the occurrence and duration of a module for different observations). To describe these generation procedures, consider that although a set of time points $\mathbf{K}$ associated with a given module, $\mathbf{B}=(\mathbf{I},\mathbf{J},\mathbf{K})$, may not necessarily be coherently satisfied by every observation in $\mathbf{I}$ (see Figure 3.7), $\mathbf{K}$ should either be soundly inferred or, alternatively, well-approximated by an aleatory variable. Illustrating, given a three-way gene expression time series, consider the presence of a module $(\mathbf{I},\mathbf{J},\mathbf{K})$ with a set of genes (features), $\mathbf{J} \subseteq \mathbf{Y}$, having coherent expression on $[t_2, t_5]$ time points for the $x_1 \in \mathbf{I}$ sample and coherent expression on $[t_3, t_7]$ time points for the $x_2 \in \mathbf{I}$ sample. In this context, the planted temporal misalignments should not prevent the retrieval of the true planted module, $(\mathbf{I},\mathbf{J},\mathbf{K} \sim U(t_2, t_7))$. For these settings, to guarantee that the inferred set of time points do not penalize assessments, the default (II-2.12) matching score (parameterized with the score described in Alg.II-2) guarantees a sound evaluation of the modeled cascades against the planted cascades.

Three major levels of difficulty can be considered for the plantation of misalignments: 1) modules with fixed time points across observations (absence of misalignments); 2) Uniform and Gaussian selection/assignment of time points per module, and 3) arbitrary-high misalignments yet respecting sequential constraints. Although the second setting (depending on the parameters of the selected distributions) may not preserve the sequential constraints between modules for some observations of a planted cascade, a generated cascade in this setting is in general more easily recovered than a cascade planted in the third setting due to the presence of average expectations on the temporal occurrence of modules.

Structural Misalignments. Finally, CascadeGen supports the Gaussian assignment of temporal shifts and/or scales on the features of a single module. Illustrating, consider a regulatory module (defined by a subset of samples **I**, genes **J**, and time points **T**) from a three-way gene expression time series. In this context, two genes (features) in **J** are co-regulated, yet their regulation may be affected by kinetic delays on their activation and/or repression. Under the Gaussian assumption for the plantation of regulatory shifts/scales, we guarantee the possibility to stably infer the well-defined set of planted/true time points **T**.

Generating Data Benchmarks with CascadeGen. The previously introduced parameters were explored to generate benchmark data. Table 3.1 describes the provided benchmarks. We varied the size of 3TS (maintaining the proportion between observations, time points and genes commonly observed in expression data). For each setting from Table 3.1, we created 30 instances with background values generated according to a Gaussian distribution, N($\frac{|\mathcal{L}|}{2}$,$\frac{|\mathcal{L}|}{6}$), with smooth factors between temporally contiguous elements.

$\|A\|\times\|G\|\times\|T\|$	10×100×10	20×150×15	30×200×20	40×300×25	50×500×30
♯Cascades	2	3	3	4	4
♯Precedences	[2,4]	[3,4]	[3,5]	[4,5]	[4,6]
♯Cooccurrences	[8,10]	[10,14]	[14,20]	[18,25]	[20,30]

Variants for fixed 1000×100 setting (default in bold)
Support of cascades: fixed $\|\Phi_R\|\in$\{0.2,0.4,0.6,0.8,1\} or varying (by default **N(0.5,1)**)
♯Modules per cascade $\|R\|\in$\{2,3,4,5,7,10\}, ♯Genes per module |**J**|∈\{10,15,20,30,50\}
Coherency strength $\mathcal{L}$ ∈\{\{0,..,3\},**\{-2,..,2\}**,\{0,..,6\},\{-4,..,4\}\} under \{constant,**symmetric**\} model
Temporal misalignments: fixed time, stochastic (varying variance), and arbitrary with respected orders (default)
Number of bifurcations \{0,**1**,2,3\} with \{yes,**no**\} exclusive paths
% shifted/scaled genes per module \{**0**,0.2,0.4,0.7\} with % of deviation on time points \{0,**0.1**,0.2,0.3\}
Noisy elements \{0,**0.02**,0.05,0.1,0.2\}

Table 3.3: Data benchmarks: properties of the generated three-way time series.

For the aim of assessing the computational efficiency, additional data benchmarks are available in CascadeGen based on the variation of the number of rows (|**X**|∈{10,20,50,100,200,1000}), columns (|**Y**|∈{100,200,500, 1000,2000,10000,20000}) and time points (|**T**|∈{10,20,50,100,200,1000}) of 3TS, as well as the number of cascades, modules per cascades (precedences) and features per module (cooccurrences).

3.2.2 Generation of Collections of Events with Informative Arrangements

According to Def.I-1.6, a multi-set of events is a set of n observations $\mathbf{x}_i \in \mathcal{X}$, where each observation is a set of events and each event is a tuple (y_j, a_{ijk}, t_{ijk}), where $y_j \in \mathcal{Y}_j$ is the type of event (feature), a_{ijk} is its value and t_{ijk} the timestamp. Regions of interest from high-dimensional multi-sets of events are commonly defined by arrangements of informative events. An arrangement essentially defines a set of events with preserved temporal constraints for a subset of observations. Despite the availability of few synthetic sequences of events with planted orderings of events (referred as episodes) [391], they are neither able to generate distinct types of events nor flexibly control the properties of data and of the planted arrangements. As such, we propose a new generator based on the extension of the IBM Generator tool[8], originally prepared to generate itemset sequences with planted sequential patterns.

Interestingly, an itemset sequence (Def.I-1.5) can be seen as a sequence of event-sets, and a sequential pattern as an arrangement of events satisfying ordering constraints. The mean and deviation of the number of planted (maximal) sequential patterns, itemsets per pattern, items per itemset can be used to easily parameterize the properties of the planted arrangements. According to the principles to be discussed in *Book V*, the generator further verifies if the parameterization of these variables guarantees the statistical significance of the arrangements.

Although the IBM Generator can be applied for this end, three challenges prevent the generation of multi-sets of events resembling the regularities of real-world databases such as repositories of health records or financial decisions. First, an itemset sequence does not consider varying temporal distances between events. For this aim, each itemset should be annotated with a time interval, and each event (item) from an itemset annotated with a timestamp within the assigned interval. The length of time intervals and the time distance between them can be easily controlled through the parameters of Uniform/Gaussian assumptions[9].

Second, in order to adequately generate observations with varying degree of sparsity, IBM Generator can be parameterized with strong deviations on the expected number of items per itemset and itemsets per sequence. Alternatively, events (items) not taking part in a planted sequential pattern can be removed or added. In this way, observations can be generated with distinct amounts of events, more in line with real-world regularities (e.g. contrasting levels of clinical activity between individuals).

Third, additional extensions need to be accommodated to create non-identically distributed attributes. For this purpose, the alphabet of itemset sequences from the IBM Generator can be partitioned into subsets of items, each subset with a possibly unique amount and frequency of items (to simulate categoric attributes with different cardinalities) and with a possibly assigned cost table and imputable deviations (to simulate numeric attributes).

A final consideration relates with the need to adequately parameterize the IBM Generator tool for the generation of data benchmarks resembling the regularities of real-world data. As this parameterization highly differs for different domains (e.g. repositories of health records, financial decisions or web-actions show distinct regularities), it should be done case by case according to user expectations and statistics from the target real-world data[10].

3.3 Generation of Labeled Data with Planted Local Regularities

Observations from tabular or structured data can be labeled. In this context, the criteria of relevance becomes not only focused on the homogeneity and statistical significance of a target local descriptive model, but also on its discriminative power in order to adequately describe class-conditional regularities and support the classification of new observations. As such, we below motivate the need for new synthetic data and extend the previously proposed generators towards labeled data contexts.

Related Work. Throughout *Section II-1.3.1*, we surveyed simulation studies aiming to generate high-dimensional labeled data [669, 276, 333, 200]. In particular, their primary focus was placed on the plantation of global regularities. Towards this end, alternative class-conditional distributions to generate data were surveyed, with their discriminative properties being planted through differences on: 1) the parameters of the probability functions, such as the mean and/or standard deviation of class-conditional multivariate Gaussian assumptions, 2) the type and composition of probability functions (including non-trivial mixtures), and 3) the properties of their covariance-matrices. Furthermore, principles were proposed to plant varying types and degree of noise, skew features (to simulate the *apparent* non-discriminative power of a large extent of features from high-dimensional data), and introduce different sources variability.

[8] http://www.cs.loyola.edu/~cgiannel/assoc_gen.html

[9] Illustrating, $U(0,1)$ means that the overall time duration is equally distributed per itemset (absence of time intervals), while $U(0,0.5)$ implies a similar duration between the duration of an itemset and the temporal between them. Additional Uniform or Gaussian distributions are used to codify the deviations on the expected durations.

[10] Illustrative parameterizations of the extended IBM Generator Tool are provided in *Book IV*

Despite the relevance of these contributions, as well as from alternative studies producing additional data benchmarks [451], they suffer from two major problems. First, they either neglect the largely recognized presence of local regularities in real-world data (Table I-1.1) or weakly explore them by planting simplistic local forms of dependency between features. Understandably, such correlation factors are insufficient to model the flexibility of forms associated with local regularities described in the previous sections. Second, their principles are only applicable to tabular data.

Furthermore, although recent contributions to perform discriminative pattern mining and biclustering generate synthetic data [497, 130], they suffer from the listed limitations in *Section 3.1.2* (biased restrictions on the allowed structure, coherency and quality of the planted biclusters) and cannot be used to flexibly parameterize the discriminative power of a region, generate data with imbalance on the classes and plant global regularities.

Solution: Generating Labeled Data. To address the surveyed challenges, we extend the previously proposed generators of class-conditional observations towards multi-class data. In this context, our goal is to effectively plant discriminative regions. Given a set of classes $\mathcal{C}$, the set of discriminative regions is often referred as an *associative model* (or discriminative local descriptive model), a composition of decision rules, where each rule $\mathbf{B} \Rightarrow C$ maps a discriminative region of interest $\mathbf{B}$ (either a bicluster, tricluster, cascade, sequential pattern or arrangement) into a set of labels $C \subset \mathcal{C}$.

In this context, we propose L2Gen (LabeLed data Generator) to generate labeled data with local regularities (discriminative regions according to an associative model) and/or global regularities (class-conditional multivariate distributions). Some of the additional parameters that can be varied against the baseline procedures provided by BiGen and CascadeGen include:

- number of classes $|\mathcal{C}|$ and level of imbalance between classes (0% implies an equal distribution of observations per class, and 50% implies that the class with fewest observations has half of the observations of the class with most observations and the remaining classes are uniformly distributed between these ranges);
- mixture of multivariate distributions to describe class-conditional data observations (according to the principles provided in *Section II-1.3.1*);
- the type and distribution of discriminative power across the planted regions of a given associative model;
- imbalance on the number of discriminative regions per class to test the ability to adequately learn associative models without biased decisions towards specific classes.

By default, the type of discriminative power is planted according to the confidence of a given decision rule: fraction of class-conditional observations with a label appearing in the rule's consequent. In this context, if a discriminative power of 80% is inputted for a given region $\mathbf{B}$ associated with a rule $\mathbf{B} \Rightarrow C$, this implies that 80% of its observations have a label in C and the remaining 20% have label in $\mathcal{C} \backslash C$ (where $\mathcal{C}$ is the set of all labels). In order to mimic the properties of real-world data, an Uniform or Gaussian distribution of the discriminative power can be specified in order to guarantee that different regions within a single associative model may show different levels of discriminative power.

By combining these parameters with the possibilities supported by BiGen and CascadeGen, L2Gen enables an environment to flexibly control the regularities underlying data and easily detect weaknesses and strengths in the performance of both local and global classifiers.

Table 3.4 describes the proposed synthetic data benchmarks generated with L2Gen (see also Tables 3.1, 3.2 and 3.3 for the default values of the excluded parameters). Similarly to previous data benchmarks, the generation of at least 30 data instances is suggested to study the performance of classifiers for a given data setting and its changes when varying a given parameter (assuming remaining default parameterizations).

Tabular data (|**X**|×|**Y**|)	100×100	400×400	1000×1000	2000×5000
Number of biclusters	6	9	12	15
Bicluster length |**J**|	[5,10]	[10,20]	[15,30]	[25,50]
Absolute support |**I**|	[20,50]	[100,200]	[250,500]	[500,1000]

3TS data (|**X**|×|**Y**|×|**T**|)	20×100×10	40×150×15	60×200×20	80×300×25	100×500×30
♯Cascades	6	9	9	12	12
♯Precedences	[2,4]	[3,4]	[3,5]	[4,5]	[4,6]
♯Cooccurrences	[8,10]	[10,14]	[14,20]	[18,25]	[20,30]
♯Observations (support)	15	30	40	50	60

Additional parameters (on top of Tables 3.1, 3.2 and 3.3 variables) with *default* values in bold):

Number of classes {2,**3**,5} with {0,**0.25**,0.5} imbalance degree;
Discriminative power $\mu(\phi)$={**90%**,80%,70%,60%} and $\sigma(\phi)$={0%,2%,**5%**,10%}
Overlapping degree {0,**0.1**,0.2,0.5} with plaid effect f={**sum**,product}

Table 3.4: Data benchmarks: properties of the generated labeled data.

3.4 Summary of Contributions and Implications

Following the growing popularity of local descriptive models and the explosion of available algorithms for their learning, the use of adequate synthetic data for their evaluation becomes critical to gain precise views about their true performance. BiGen, CascadeGen and L2Gen methods are proposed to generate data with planted regions with parameterizable flexible structure, coherency and quality. These methods offer a highly parameterizable environment to study the strengths and weaknesses of a wide-set of algorithms for biclustering, pattern mining, network analysis, triclustering, motif discovery, cascade discovery, modeling of event-sets and classification. Yet, default parameterizations are provided to generate data with regularities according to biological data domains for standardized and fair assessments of upcoming contributions on these fields of research.

The relevance of BiGen is of particular importance since there is a lack of understanding of the true performance of biclustering algorithms as a function of the coherency strength, positioning and quality of biclusters. Additionally, the relevance of CascadeGen and L2Gen is driven by the lack of available data benchmarks to respectively model: varying forms of local regularities (such as given by coherent modules and their connectivity) from structured data, and discriminative regions with varying properties.

Empirical Evidence. Experimental results provided throughout *Books III, IV* and *VI* provide substantial empirical evidence that support the relevance of BiGen, CascadeGen and L2Gen to unravel key aspects of the performance of assessed algorithms.

Concluding Note. As a result of these observations, we expect to see assessments for the evaluation of state-of-the-art and upcoming learning methods grounded on the principles proposed in this chapter (principles to generate data) and *Chapters 1* and *2* (principles to bound and compare performance). These assessments are essential to gain a complete understanding of their behavior in high-dimensional data contexts, and possibly guide their improvement.