Pontifícia Universidade Católica
do Rio de Janeiro

Daniel Lemes Gribel

A Model-based Framework for Semi-supervised Clustering and Community Detection

Tese de Doutorado

Thesis presented to the Programa de Pós–graduação em Informática of PUC-Rio in partial fulfillment of the requirements for the degree of Doutor em Ciências – Informática.

Advisor : Prof. Thibaut Victor Gaston Vidal
Co-advisor: Prof. Michel Gendreau

Rio de Janeiro
July 2021

Pontifícia Universidade Católica do Rio de Janeiro

Daniel Lemes Gribel

A Model-based Framework for Semi-supervised Clustering and Community Detection

Thesis presented to the Programa de Pós–graduação em Informática of PUC-Rio in partial fulfillment of the requirements for the degree of Doutor em Ciências – Informática. Approved by the Examination Committee:

Prof. Thibaut Victor Gaston Vidal
Advisor
Departamento de Informática – PUC-Rio

Prof. Michel Gendreau
Co-advisor
École Polytechnique de Montréal

Prof. Marco Serpa Molinaro
Departamento de Informática – PUC-Rio

Prof. Marcus Vinicius Soledade Poggi de Aragão
Departamento de Informática – PUC-Rio

Prof. Daniel Aloise
École Polytechnique de Montréal

Prof. Emilio Carrizosa Priego
Universidad de Sevilla

Rio de Janeiro, July 26th, 2021

Daniel Lemes Gribel

Daniel Lemes Gribel holds a M.Sc. (2017) from the Department of Informatics at the Pontifical Catholic University of Rio de Janeiro (PUC-Rio) and a B.Sc. in Information Systems (2014) from the Federal University of the State of Rio de Janeiro (UNIRIO), Brazil. He held a scholarship from the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) during the doctorate. For one year, he was a visiting student at the École Polytechnique de Montréal / CIRRELT, and received a scholarship from the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES).

Bibliographic data

Gribel, Daniel

A Model-based Framework for Semi-supervised Clustering and Community Detection / Daniel Lemes Gribel; advisor: Thibaut Victor Gaston Vidal; co-advisor: Michel Gendreau. – 2021.

103 f: il. color. ; 30 cm

Tese (doutorado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática, 2021.

Inclui bibliografia

1. Informática – Teses. 2. Otimização. 3. Aprendizado de máquina. 4. Mineração de dados. 5. Agrupamento. 6. Agrupamento semissupervisionado. 7. Modelos de blocos estocásticos. 8. Detecção de comunidades. I. Vidal, Thibaut. II. Gendreau, Michel. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. IV. Título.

CDD: 004

To my parents, for their support
and encouragement.

Acknowledgments

First, I would like to thank my advisors for the amazing discussions and the great opportunities.
I am also very grateful to my colleagues at PUC-Rio and CIRRELT for the friendship and inspiring conversations.
Next, I would like to thank CNPq and CAPES for the valuable funding.
Last but not least, I would like to thank my family for all their support, given in so many dimensions.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001.

Abstract

Gribel, Daniel; Vidal, Thibaut (Advisor); Gendreau, Michel (Co-Advisor). **A Model-based Framework for Semi-supervised Clustering and Community Detection**. Rio de Janeiro, 2021. 103p. Tese de Doutorado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

In model-based clustering, we aim to separate data samples into meaningful groups by optimizing the fit of some observed data to a mathematical model. The recent adoption of model-based clustering has allowed practitioners to model complex patterns in data and explore a wide range of applications. This thesis investigates model-driven approaches for community detection and semi-supervised clustering by adopting a maximum-likelihood perspective. We first focus on exploiting constrained optimization techniques to present a new model for community detection with stochastic block models (SBMs). We show that the proposed constrained formulation reveals communities structurally different from those obtained with classical community detection models. We then study a setting where inaccurate annotations are provided as *must-link* and *cannot-link* relations, and propose a novel semi-supervised clustering model. Our experimental analysis shows that incorporating partial supervision and appropriately encoding prior user knowledge significantly enhance clustering performance. Finally, we examine the problem of semi-supervised clustering in the presence of unreliable class labels. We focus on the case where groups of untrustworthy annotators deliberately misclassify data samples and propose a model to handle such incorrect statements.

Keywords

Optimization; Machine learning; Data mining; Clustering; Semi-supervised clustering; Stochastick block models; Community detection.

Resumo

Gribel, Daniel; Vidal, Thibaut; Gendreau, Michel. **Um Framework Baseado em Modelo para Clusterização Semissupervisionada e Detecção de Comunidades**. Rio de Janeiro, 2021. 103p. Tese de Doutorado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Em clusterização baseada em modelos, o objetivo é separar amostras de dados em grupos significativos, otimizando a aderência dos dados observados a um modelo matemático. A recente adoção de clusterização baseada em modelos tem permitido a profissionais e usuários mapearem padrões complexos nos dados e explorarem uma ampla variedade de aplicações. Esta tese investiga abordagens orientadas a modelos para detecção de comunidades e para o estudo de clusterização semissupervisionada, adotando uma perspectiva baseada em máxima verossimilhança. Focamos primeiramente na exploração de técnicas de otimização com restrições para apresentar um novo modelo de detecção de comunidades por meio de modelos de blocos estocásticos (SBMs). Mostramos que a formulação com restrições revela comunidades estruturalmente diferentes daquelas obtidas com modelos clássicos. Em seguida, estudamos um cenário onde anotações imprecisas são fornecidas na forma de relações *must-link* e *cannot-link*, e propomos um modelo de clusterização semissupervisionado. Nossa análise experimental mostra que a incorporação de supervisão parcial e de conhecimento prévio melhoram significativamente os agrupamentos. Por fim, examinamos o problema de clusterização semissupervisionada na presença de rótulos de classe não confiáveis. Investigamos o caso em que grupos de anotadores deliberadamente classificam incorretamente as amostras de dados e propomos um modelo para lidar com tais anotações incorretas.

Palavras-chave

Otimização; Aprendizado de máquina; Mineração de dados; Agrupamento; Agrupamento semissupervisionado; Modelos de blocos estocásticos; Detecção de comunidades.

Table of contents

List of figures

List of tables

List of Abreviations

AC-DC-SBM – Assortative-constrained degree-corrected stochastic block model

CI – Centroid index

DC-SBM – Degree-corrected stochastic block model

EM – Expectation-maximization

GMM – Gaussian mixture model

KL – Kullback–Leibler

MSSC – Minimum sum-of-squares clustering

NMI – Normalized mutual information

PPM – Planted partition model

SBM – Stochastic block model

1
Introduction

1.1
Context and Background

Data clustering is the general technique of finding meaningful groups of data samples according to some criteria. For example, one may intend to group movies based only on their characteristics or features. There are different ways of cataloging them, and we do it according to some criteria (for example, by genre, director, or decade). Each criterion represents a different perspective regarding the data samples we have at hand. However, no matter the criteria we choose in clustering, we rely only on the data's features. Clustering is then the unsupervised branch of machine learning, where the goal is to recognize meaningful patterns in data.

As discussed in Anderberg [4], the task of grouping data samples exists for a long time, being one of the most primitive activities of humans. Bouveyron et al. [13] affirm that the first record of a systematic grouping is the biological taxonomy system of Carl Linnaeus, in his *Systema Naturae* (1735) and subsequent works. Linnaeus cataloged species from the natural world, dividing them into animal, plant, and mineral kingdoms. He also designed specific taxonomies within each kingdom, as the 24 classes defined for plants in the *Systema Sexuale*.

Clustering differs from ordinary taxonomy systems in that it employs systematic numerical methods for identifying meaningful groups [13]. In the early development of cluster analysis, most clustering techniques were ad hoc approaches applied to similarity measures extracted for each pair of observed samples. Over time, practitioners adopted model-based clustering, which considers probabilistic formulations. The adoption of model-based clustering has allowed practitioners to model complex patterns in data and explore a wide range of applications, including medical imaging, criminal activity detection, market segmentation, community detection on social networks, among others [71]. Although the clustering practice has an end in itself, we can also use it as a step within supervised learning. For example, we can use a cluster representative sample to infer missing feature values or to perform data compression. During the training of supervised models, it is common to use a clustering solution to estimate the parameters of classification models.

In model-based clustering, we aim to find meaningful groups of samples by optimizing the fit of an observed data to some mathematical model. A common way of estimating the model parameters, and the one adopted in this research, is maximizing the likelihood function. Typically, we assume that a mixture of probability distributions generates the data. The probability distributions can be Gaussian, Poisson, Bernoulli, Exponential, among others, chosen according to our knowledge regarding the data's generative process.

It is important to note that data representation in clustering tasks may vary widely. The information coming along with data samples may be numerical or categorical features describing the samples' general characteristics. In other situations, no feature set describes the data, but relational information states that some connection exists between two samples. It is also possible that some similarity measure between pairs of samples defines the dataset. Therefore, depending on the type of data we have, we assume an appropriate generative model. For example, mixtures of Gaussian distributions are widely used to model continuous features. For relational data, mixtures of discrete distributions such as Poisson or Bernoulli are commonly adopted to model the connections between data samples. In this case, we assume that a probability distribution describes the connectivity pattern between two clusters, including each cluster internally.

1.2 Scope and Focus

The scope of this thesis is the study of model-driven approaches for data clustering, in which we investigate community detection and semi-supervised clustering tasks. We first focus on the exploitation of constrained optimization to present a new model for community detection. We then proceed towards the proposition of semi-supervised learning models, in which we investigate a particular setting where inaccurate and untrustworthy annotations are introduced in data as side information.

We dedicate the first part of this thesis to investigating the use of stochastic block models (SBMs) [34, 52] for community detection. SBMs are probabilistic models widely used for graph partitioning. They are a natural modeling choice for community detection when only relational data is available. We discuss the advantages and drawbacks of using ordinary SBMs for community detection and highlight the conditions in which the classic model may lead to undesired outcomes. On top of this analysis, we propose model extensions for SBMs that encode prior knowledge about the desired outcome by appending a set of constraints to the original model.

In the second part of this thesis, we handle SBMs in a way that differs from the conventional use of the model to introduce a semi-supervised clustering model. We consider continuous features describing the data samples and use SBMs to model pairwise annotations. This general setting corresponds to the case in which *experts* provide supervision in the form of *must-link* and *cannot-link* annotations. We investigate the benefits of incorporating such relational supervision into the minimum sum-of-squares clustering model and analyze the impact of having inaccurate relational information, i.e., when the *experts* provide annotations that present some error. Moreover, we extend the model to integrate prior knowledge regarding the annotations' accuracy and discuss the circumstances in which the use of this knowledge is beneficial.

Finally, we investigate the problem of mitigating the presence of noisy annotations in classification tasks. More precisely, we explore semi-supervised techniques for detecting groups of untrustworthy annotators. We focus on the case where unreliable groups of annotators deliberately misclassify data samples, and propose a semi-supervised model to detect the inclination of such groups.

1.3 Objectives and Contributions

The flexibility of models like SBMs can be helpful in many situations, but it can also lead to undesirable outcomes in some cases. Particularly, ordinary SBMs may converge towards solutions that do not meet the assortativity requirements, i.e., when the probability of connections within communities is higher than between communities. Especially in sparse and lightly assortative networks, ordinary SBMs may converge towards non-assortative solutions. Therefore, the first main objective of this research is to study forms of imposing constraints to SBMs that meet the requirements of assortativity and assess the impact of such constraints in community detection.

In the context of clustering with partial supervision, several models have recently been proposed. Semi-supervised clustering is an active research area that received much attention due to the real-world demands for suitable models and algorithms for clustering in the presence of scarce and weak annotations. Although fundamental advances happened in semi-supervised learning in the last years, most existing algorithms are unable to handle inaccurate supervision. Additionally, most of the existing approaches usually do not directly derive from a model-based perspective. Therefore, the second main objective of this study is to fill this gap and provide principled probabilistic models for semi-supervised clustering formulations capable of accommodating inaccurate and unreliable annotations.

In sum, this research's general goal is to explore the flexibility of model-driven approaches and develop novel methodologies along with algorithmic solutions to make progress in semi-supervised clustering and community detection. In light of the main objectives, the principal contributions of this work are the following:

- We present a variant of the SBM as a model extension for improving community detection in assortative networks, which consists of incorporating constraints that accounts for prior user knowledge and permits attaining assortative structures that are not obtained with ordinary SBMs;
- By discussing the practical implications of imposing assortativity constraints in networks, we highlight the regimes in which such restrictions contribute to improving the detection of communities;
- We introduce a novel semi-supervised clustering model, in which we couple Gaussian mixtures and SBMs to model data features and annotations in the form of *must-link* and *cannot-link* constraints;
- We demonstrate that our semi-supervised model is capable of accommodating inaccurate annotations, and discuss the circumstances in which the introduction of prior information regarding the annotations accuracy improves clustering performance;
- In order to handle noisy labels, we introduce a methodology to detect groups of untrustworthy annotations by jointly considering labels and data features;
- We provide efficient algorithms for the assortative SBM and the semi-supervised clustering models;
- Through extensive computational experiments using synthetic and real-world datasets, we show that the assortative SBM and the semi-supervised clustering models lead to significant performance improvements in terms of clustering evaluation metrics.

1.4 Thesis structure

This manuscript is organized as follows: Chapter 2 presents the assortative-constrained SBM. We cover the necessary background of SBMs along with a review of relevant research on this topic. We then introduce the proposed model and a solution approach. Chapter 3 is dedicated to clustering with partial supervision. We state the essential theoretical background and discuss previous works, and then present our semi-supervised model along with an algorithmic solution. Chapter 4 presents a semi-supervised methodology for detecting groups of untrustworthy annotators. Finally, Chapter 5 presents the conclusions and future perspectives.

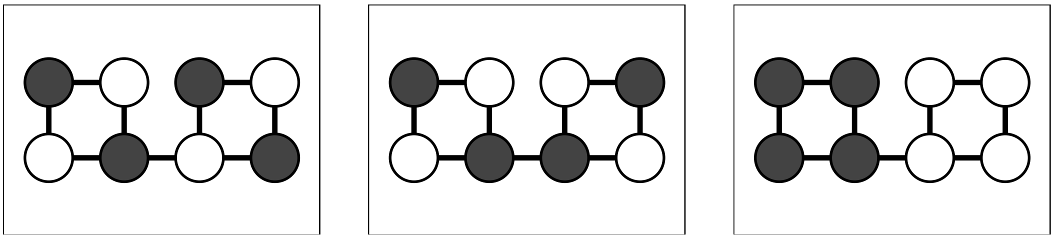
2
Assortative-Constrained Stochastic Block Models

2.1
Introduction

Community detection methods hold a central place in machine learning, with an extensive range of applications related to sociological behavior, protein interactions, image segmentation, and gene expressions analysis [1]. In most of these applications, the actual classes of the nodes in the network are unknown, but pairwise relations between nodes are exploited to identify communities.

Fitting the parameters of a stochastic block model (SBM) [34, 52] to a given graph is a prominent way of searching for communities. The canonical SBM assumes that each node belongs to one block (representing a community) and that the expected number of edges between two nodes depends only on the blocks to which they belong. Thus, the model only assumes that nodes within each block are statistically equivalent in their connectivity patterns. Several variations of the standard SBM were also introduced to overcome some of its limitations. The degree-corrected SBM (DC-SBM) introduced by Karrer and Newman [36], in particular, allows non-uniform node degree distributions, making block modeling more representative of real-world networks.

Broadly speaking, a solution for community detection (represented as a partition of the node set into communities) is *assortative* when connections within communities are more frequent than in between communities, it is *disassortative* when connections within communities are less frequent than in between communities, and finally it is *non-assortative* if no such relation exists among all communities. SBM-based community detection approaches are agnostic to the assortativity of their solutions. They allow to search for solutions with a pre-defined number of communities, and can indifferently model assortative and disassortative structures. This modeling capability can be viewed as an asset but also as a weakness. Indeed, SBMs are often used in contexts in which users expect assortative solutions. In the most dramatic situations, non-assortative solutions might go under the radar and lead to mistakes of interpretation. In other cases, non-assortative solutions with a better likelihood may substitute the assortative solutions which were originally sought (Figure 2.1). This later situation is especially prevalent in case studies involving sparse graphs, or with lightly assortative structures which challenge detection algorithms.



(a) Optimal solution $\log \mathcal{L} = -2.7616$

(b) 2^{nd} best solution $\log \mathcal{L} = -5.3193$

(c) 3^{rd} best solution $\log \mathcal{L} = -5.9012$

Figure 2.1: The three best solutions in a small example case with two communities. The two best solutions in terms of maximum likelihood are disassortative, whereas the third (c) is assortative.

In this chapter, we propose a variant of the DC-SBM which includes user knowledge about assortativity. We incorporate this information by setting assortativity constraints on the DC-SBM parameter set. Indeed, if the user expects an assortative solution due to the characteristics of the application case, it is plausible to guide the convergence of the model via additional constraints. We show that the resulting constrained likelihood-maximization model can be solved efficiently with an iterative method based on local-search and interior-point algorithms. Our computational experiments show that the assortativity constraints prevent the search from converging towards spurious non-assortative local minima, especially in sparse networks. These constraints also contribute to identifying different solution structures in application cases related to the analysis of the brain cortex. The key contributions of this chapter are, therefore, the following.

1) We introduce a DC-SBM variant which incorporates assortativity constraints to represent prior user knowledge;

2) We propose an efficient solution approach based on local optimization and interior-point algorithms for this model;

3) Through extensive computational experiments, we discuss the practical implications of this constrained model and identify the regimes in which it contributes to improve community detection practice.

2.2 Related Works

SBMs are commonly used to extract meaningful information from complex networks. The classical SBM is also a natural modeling choice for community detection [1] and a generalization of modularity maximization [51]. The surveys of Abbe [1] and Lee and Wilkinson [40] discuss key results regarding recovery requirements and solution algorithms. Different types of algorithms can be used to fit SBMs, based on Markov Chain Monte Carlo (MCMC) approaches [47, 52, 56], variational inference [2, 68], belief propagation [19], spectral clustering [41, 58, 60], and semidefinite programming [14, 15], among others.

To date, few works have considered the possibility of incorporating prior information on assortativity. Moore et al. [48] studied an SBM in which the edge probabilities within and in between communities follow a Beta prior. The hyperparameters defining the Beta distributions drive the degree of assortativity in the graph. Yet, according to their experiments, these priors dominate only in small or sparse datasets, otherwise they tend to wash out.

The Assortative Mixed Membership SBM (a-MMSB) introduced by Gopalan et al. [29] considers soft node-to-community assignments and includes a latent parameter describing community strength, representing how tightly nodes are connected within each group. Edges are assumed to be drawn from a Bernoulli distribution centered around the community strength if the nodes belong to the same group. Otherwise, the distribution is centered around a small value. A variational inference approach is used to fit the model. Li et al. [42] pursued this research line by proposing a scalable MCMC method using a stochastic gradient algorithm for posterior inference in the a-MMSB.

Lu and Szymanski [45] finally proposed a regularized variant of the DC-SBM, using a prior to regularize the observed in-degree ratio of each node. In practice, this adaptation turns out to penalize high-degree nodes with many connections to other communities. The new parameter is adjusted to control the assortativity level, and a MCMC algorithm is used to infer the block assignments.

The aforementioned models aim to better fit assortative networks, but they are either dependent on ad-hoc parameters which are difficult to scale [29, 45], or of limited effect for larger graphs [48]. In light of these works, we decided to explore a different approach, which consists in guiding the search towards assortative structures via constraints in the SBM parameters. To fit our model, we propose effective algorithms for the resulting constrained maximum-likelihood optimization problem.

2.3 DC-SBM: Background and Notations

In its most fundamental form, the DC-SBM considers N nodes allocated to K groups. We assume that the number of edges between a pair of nodes (i, j) depends only on the groups to which the nodes belong and on their degrees [51]. Finding the latent membership of nodes corresponds to finding the block-model parameters that best fit the observed graph [1]. For an observed adjacency matrix $\boldsymbol{A} \in \mathbb{N}^{N,N}$ representing a graph with m (possibly weighted) edges, the log-likelihood function of the DC-SBM is calculated as [36]:

$$\log P(\boldsymbol{A}|\boldsymbol{\Omega}, \boldsymbol{Z}) = \frac{1}{2}\sum_{rs}^{K}\sum_{ij}^{N}\left(A_{ij}\log(\omega_{rs}) - \frac{k_i k_j}{2m}\omega_{rs}\right) z_{ir} z_{js}, \tag{2-1}$$

in which we consider the special case that the probability of selecting node i to form an edge is constrained to be $k_i/\sqrt{2m}$, where k_i is the degree of node i. Variables $\boldsymbol{Z} \in \{0,1\}^{N,K}$ represent the binary community assignments, in such a way that $z_{ir} = 1$ indicates that node i is assigned to group r. $\boldsymbol{\Omega}$ is a symmetric $K \times K$ edge probability matrix. Each element ω_{rs} of $\boldsymbol{\Omega}$ corresponds to the expected number of edges between any two points in groups r and s. The expected number of edges between nodes i and j is $\frac{k_i k_j}{2m}\omega_{rs}$, for $z_{ir} = 1$ and $z_{js} = 1$.

In the DC-SBM model, we aim to find the parameters $\boldsymbol{Z}$ and $\boldsymbol{\Omega}$ that maximize the likelihood (2-1). If we fix the assignment $\boldsymbol{Z}$, then the (unconstrained) maximum-likelihood for each parameter ω_{rs} can be estimated by differentiation:

$$\hat{\omega}_{rs} = \frac{2m \cdot m_{rs}}{\kappa_r \kappa_s}, \tag{2-2}$$

where $m_{rs} = \sum_{ij}^{N} A_{ij} z_{ir} z_{js}$ is the number of edges between groups r and s, and $\kappa_r = \sum_{i}^{N} k_i z_{ir}$ is the sum of the degrees of nodes in group r. If we substitute $\hat{\omega}_{rs}$ in Equation (2-1), we obtain the following log-likelihood function (see the appendix A.1 for calculations):

$$\log P(\boldsymbol{A}|\boldsymbol{Z}) = \frac{1}{2}\sum_{rs}^{K}\sum_{ij}^{N}\left(A_{ij}\log\left(\frac{m_{rs}}{\kappa_r \kappa_s}\right)\right) z_{ir} z_{js}, \tag{2-3}$$

in which we dropped the terms that do not involve $\boldsymbol{Z}$.

2.3.1 Planted Partition Model and Modularity

The Planted Partition Model (PPM) is a special case of the standard SBM with only two parameters describing the blocks: $\omega_{rs} = \omega_{\text{IN}}$ if $r = s$, and $\omega_{rs} = \omega_{\text{OUT}}$ if $r \neq s$. Newman [51] shows that maximizing the likelihood of the PPM is equivalent to maximizing modularity. Modularity optimization maximizes the difference between the observed graph and a random graph where edges are reinserted randomly and the degrees of each node is preserved. As a consequence, it results in maximizing the number of edges within groups, leading to assortative solutions. However, modularity maximization is also subject to strong limitations: beyond its inability to define the number K of communities, the model assumes that all communities have similar statistical properties [51]. This is a major issue when the distribution of edges between the blocks varies significantly.

2.4 Assortative-Constrained SBM

We now introduce the assortative-constrained degree-corrected SBM (AC-DC-SBM) along with an efficient algorithm to fit it by maximum likelihood. Following Amini et al. [3], two main notions of assortativity can be distinguished for block models:

Strong assortativity. All diagonal terms of $\mathbf{\Omega}$ are greater or equal than all off-diagonal terms:

$$\omega_{qq} \geq \omega_{rs} \quad \forall\, q, r, s \in \{1, \ldots, K\}, r \neq s. \tag{2-4}$$

Weak assortativity. Each diagonal term of $\mathbf{\Omega}$ is greater or equal than the other terms in its row:

$$\omega_{qq} \geq \omega_{qs} \quad \forall\, q, s \in \{1, \ldots, K\}. \tag{2-5}$$

Other types of assortativity constraints may be considered with simple adaptations of our algorithm, e.g., imposing a lower bound on the number of blocks satisfying Condition (2-5). In this study, we will use the strongest definition of assortativity based on Condition (2-4). With these constraints, the log-likelihood maximization model becomes:

$$\max_{\boldsymbol{\Omega},\boldsymbol{Z},\lambda} \quad \frac{1}{2}\sum_{rs}^{K}\sum_{ij}^{N}\left(A_{ij}\log(\omega_{rs}) - \frac{k_i k_j}{2m}\omega_{rs}\right) z_{ir}z_{js} \tag{2-6a}$$

$$\text{s.t.} \quad \omega_{qq} \geq \lambda \quad \forall q \in \{1,\dots,K\} \tag{2-6b}$$

$$\omega_{rs} \leq \lambda \quad \forall r,s \in \{1,\dots,K\}, r \neq s \tag{2-6c}$$

$$\omega_{rs} \geq 0 \quad \forall r,s \in \{1,\dots,K\}, \tag{2-6d}$$

where λ represents a continuous variable acting as a threshold.

It is important to note that the assortativity constraints only apply on the block-model parameters $\boldsymbol{\Omega}$. This does not completely eliminate the possibility of a disassortative partition as represented by $\boldsymbol{Z}$, but strongly penalizes its log-likelihood in comparison to other assortative solutions.

2.4.1 Likelihood Maximization

We introduce an iterative algorithm to solve (2-6a–2-6d). This algorithm starts with a random initial solution and proceeds by iteratively evaluating each possible relocation of a node to a different community. Each such relocation is only applied if its application combined with an optimal update of $\boldsymbol{\Omega}$ results into an improvement of the likelihood. As such, the evaluation of each relocation may require the solution of a small constrained convex optimization subproblem with K^2 variables and constraints to find an optimal $\boldsymbol{\Omega}$ for the new partition. For the classical DC-SBM, the optimal $\boldsymbol{\Omega}$ is simply obtained via Equation (2-2). This is however, no longer true for the AC-DC-SBM due to the assortativity constraints. As described in Algorithm 1, the overhead associated to this operation can be mitigated by combining two techniques:

(i) an incremental move evaluation approach, using the log-likelihood of the unconstrained subproblem (Lines 9–10) to filter relocation candidates (Line 11), and possibly keeping this solution if it naturally satisfies the assortativity constraints (Lines 12–13);

(ii) an efficient interior point solver for Problem (2-7a–2-7d), only used if the relocation candidate was not filtered out due to the previous conditions (Lines 14–19).

We use the interior point algorithm of Domahidi et al. [23] for the solution of each subproblem. When the partition is fixed, the constrained maximization

Algorithm 1 Likelihood maximization algorithm

Input: Adjacency matrix: $\boldsymbol{A}$, Number of communities: K
Output: Network partition: $\boldsymbol{Z}$
Initialize a random partition $\boldsymbol{Z}$
Find $\boldsymbol{\Omega}$ by solving (2-7a–2-7d) for partition $\boldsymbol{Z}$
Evaluate log-likelihood: $L \leftarrow \log P(\boldsymbol{A}|\boldsymbol{\Omega}, \boldsymbol{Z})$
repeat
  for each $i \in \{1, \ldots, N\}$ and $r \in \{1, \ldots, K\}$ **do**
    Consider partition $\boldsymbol{Z}^{\mathrm{R}}$ obtained from $\boldsymbol{Z}$ by relocating node i to community r
    Find $\boldsymbol{\Omega}'$ maximizing $\log P(\boldsymbol{A}|\boldsymbol{\Omega}', \boldsymbol{Z}^{\mathrm{R}})$
    Evaluate log-likelihood: $L' \leftarrow \log P(\boldsymbol{A}|\boldsymbol{\Omega}', \boldsymbol{Z}^{\mathrm{R}})$
    if $L' > L$ **then**
      if $\boldsymbol{\Omega}'$ satisfies (2-7b–2-7d) **then**
        Apply: $\boldsymbol{\Omega} \leftarrow \boldsymbol{\Omega}'$, $\boldsymbol{Z} \leftarrow \boldsymbol{Z}^{\mathrm{R}}$, $L \leftarrow L'$
      else
        Find $\boldsymbol{\Omega}''$ by solving (2-7a–2-7d) for partition $\boldsymbol{Z}^{\mathrm{R}}$
        $L'' \leftarrow \log P(\boldsymbol{A}|\boldsymbol{\Omega}'', \boldsymbol{Z}^{\mathrm{R}})$
        if $L'' > L$ **then**
          Apply: $\boldsymbol{\Omega} \leftarrow \boldsymbol{\Omega}''$, $\boldsymbol{Z} \leftarrow \boldsymbol{Z}^{\mathrm{R}}$, $L \leftarrow L''$
        end if
      end if
    end if
  end for
until No improving relocation has been identified

subproblem takes the following form:

$$\max_{\boldsymbol{\Omega},\lambda} \quad \frac{1}{2}\sum_{rs}^{K} \left(m_{rs}\log(\omega_{rs}) - T_{rs}\omega_{rs}\right) \tag{2-7a}$$

$$\text{s.t.} \quad \omega_{qq} \geq \lambda \quad \forall q \in \{1, \ldots, K\} \tag{2-7b}$$

$$\omega_{rs} \leq \lambda \quad \forall r, s \in \{1, \ldots, K\}, r \neq s \tag{2-7c}$$

$$\omega_{rs} \geq 0 \quad \forall r, s \in \{1, \ldots, K\}, \tag{2-7d}$$

where m_{rs} represents the number of edges between communities r and s according to the fixed partition and $T_{rs} = \ (\sum_t^K m_{rt} \sum_t^K m_{st})/2m$.

2.5 Empirical Studies

We conduct extensive computational experiments on synthetic and real datasets to analyze three aspects of the proposed assortative-constrained DC-SBM (AC-DC-SBM). Firstly, we wish to know under which conditions the assortativity constraints help to converge to desirable partitions. Secondly, we compare the AC-DC-SBM, the standard DC-SBM and the modularity maximization model in terms of community detection performance. Finally, we apply the AC-DC-SBM to graphs representing brain cortex data, highlighting structures which were not previously detected before and discuss the implications of the different models.

The algorithms presented in this chapter were implemented in Julia (version 1.0.5) and the source code is available at `http://github.com/danielgribel/AssortativeSBM`.

2.5.1 Networks Generated From a PPM

The standard DC-SBM usually finds assortative solutions for assortative networks with a sufficient amount of information. However, it can be trapped into spurious non-assortative local minima on sparse or lightly assortative networks. To limit the number of factors, we conduct this first analysis on datasets generated by a simple PPM (Section 2.3.1) with $K = 4$ blocks, $N = 100$ nodes, an average degree of 16, and different ratio values for $\omega_{\text{OUT}}/\omega_{\text{IN}}$ representing different assortativity levels. Our goal is to evaluate in which regimes the assortativity constraints are meaningful. Figure 2.2 therefore depicts the performance of the standard DC-SBM and of the proposed AC-DC-SBM in terms of normalized mutual information (NMI) [38]. NMI is an entropy-based score that measures the mutual dependence between two random variables. In clustering, NMI quantifies the "amount of information" that can be extracted from one partition by observing another partition. It is an adjusted measure that bounds the result between 0 (no mutual dependence) and 1 (same clustering). For each dataset and model, we report the results of 100 independent runs from different initial solutions. These results are represented as box plots, where the whiskers extend to 1.5 times the interquartile range. Table A.1 in the appendix presents the detailed results for networks generated from PPMs.

For the datasets of Figure 2.2, detectability is known to be possible for values of $\omega_{\text{OUT}}/\omega_{\text{IN}}$ smaller than ≈ 0.4 (see [20]). As expected, as the ratio $\omega_{\text{OUT}}/\omega_{\text{IN}}$ increases beyond 0.4, both models are unable to recover the communities. In contrast, when this ratio diminishes below 0.4, the performance of both methods improves, highlighting a phase transition towards a regime where partial recovery is possible. As visible in these experiments, the transition of AC-DC-SBM occurs before that of the standard DC-SBM. For example, when $\omega_{\text{OUT}}/\omega_{\text{IN}} = 0.25$, AC-DC-SBM achieves an average NMI of 0.55, compared to 0.34 for DC-SBM. Similarly, when $\omega_{\text{OUT}}/\omega_{\text{IN}} = 0.1$, AC-DC-SBM achieves near-perfect recovery on a much larger proportion of the runs. As such, it appears that the assortativity constraints are useful to guide likelihood maximization algorithms in challenging datasets located within the phase transition regime.

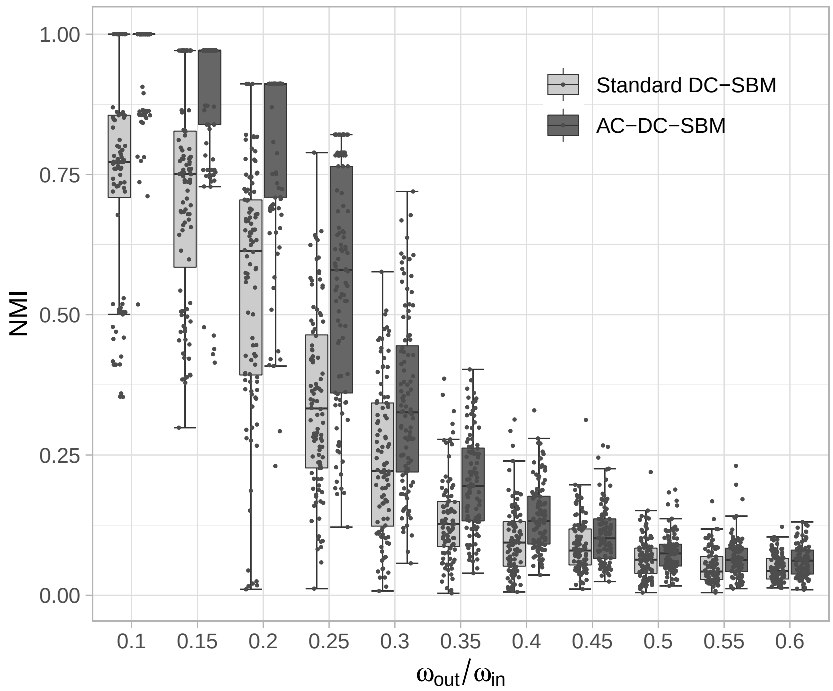


Figure 2.2: Performance of DC-SBM and AC-DC-SBM on networks generated from PPMs with varying degree of assortativity.

Datasets ($\omega_{\text{OUT}}/\omega_{\text{IN}}$)	Average CPU time (s)		Solutions submitted to the constrained subproblem (%)
	Standard DC-SBM	AC-DC-SBM	AC-DC-SBM
0.1	0.03	0.44	8.16
0.2	0.04	0.52	6.81
0.3	0.04	0.66	8.03
0.4	0.04	0.76	9.95
0.5	0.04	0.94	13.17
0.6	0.04	1.01	13.94

Table 2.1: Performance of DC-SBM and AC-DC-SBM in terms of computational effort.

Table 2.1 compares the computational effort needed to solve the standard DC-SBM and the AC-DC-SBM for the same datasets. The last column reports the percentage of relocation evaluations (from Lines 7–22 of Algorithm 1) that required the solution of the constrained continuous optimization subproblem (2-7a–2-7d) over the mixture parameters (i.e., that entered Line 15). As the networks become less assortative, the optimization algorithm relies more frequently on the solution of the constrained convex problem, leading to a sensible increase of computational time.

2.5.2
Networks Generated From SBMs

We now repeat the previous experiment on general SBMs, characterized by a larger number of parameters. To compare the results of the DC-SBM and AC-DC-SBM, we generate 50 synthetic datasets with $N = 100$ nodes and $K = 4$ blocks. For each dataset, the $\mathbf{\Omega}$ parameters are uniformly sampled in the following intervals:

$$\omega_{rr} \in \; [0.45, 0.55] \qquad \forall\, r \in \{1, \ldots, K\} \tag{2-8}$$

$$\omega_{rs} \in \; [0, 0.4] \qquad \forall\, r, s \in \{1, \ldots, K\}, r \neq s. \tag{2-9}$$

Each node is allocated to one of the four blocks with equal probability. Then, for each node pair (i, j), a number of edges is generated from a Poisson distribution centered in ω_{rs}, where r and s represent the blocks of i and j.

Figure 2.3 compares the NMI obtained with the standard DC-SBM and the proposed AC-DC-SBM on these networks. For each network and model, we conduct 50 independent runs from different initial solutions and report the results as boxplots. AC-DC-SBM obtains on 49 out of 50 datasets a better or equal median NMI than DC-SBM. DC-SBM appears to be very sensitive to low-quality local minima. This behavior is particularly visible on the first six datasets presented in the figure. A pairwise Wilcoxon test comparing the average NMI of both methods over the 50 datasets confirms the statistical significance of this difference of performance (with $p = 3.9 \times 10^{-10}$). Table A.2 in the appendix presents the detailed results for networks generated from SBMs.

In a second part of this analysis, we filter the set of solutions produced by the methods to focus on the top 10% in terms of likelihood for each dataset. This corresponds to a typical use case in which multiple independent runs are performed to avoid local minima. Figure 2.4 displays the relative difference between the NMI of the 10% top solutions of the AC-DC-SBM and those of the standard DC-SBM. For the sake of completeness, we repeat the same analysis with the modularity-maximization algorithm. As visible in these results, the best AC-DC-SBM solutions still outperform those of the two other approaches on most datasets. The statistical significance of these observations is confirmed by pairwise Wilcoxon tests (with $p = 2.4 \times 10^{-5}$ and $p = 5.1 \times 10^{-6}$ for DC-SBM and modularity maximization, respectively). Table A.3 in the appendix

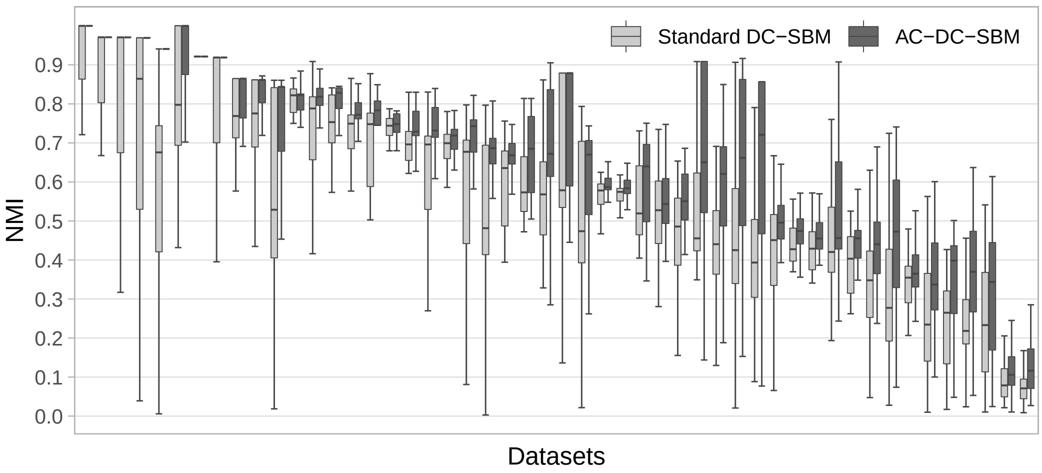


Figure 2.3: Performance of DC-SBM and AC-DC-SBM on networks generated from general SBMs. The results are ordered by median NMI.

presents the detailed results on the top 10% solutions for networks generated from SBMs.

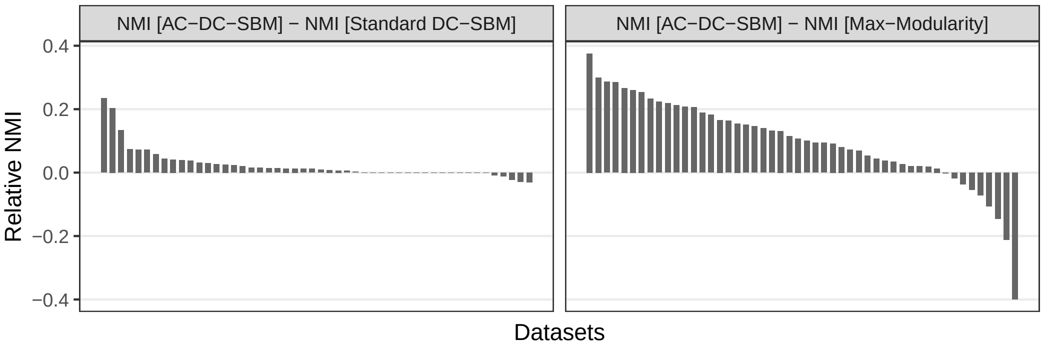


Figure 2.4: Relative NMI between the AC-DC-SBM and the standard DC-SBM (left) and modularity-maximization (right). Analysis based on the top 10% best solutions for each dataset.

Figure 2.5 finally compares the number of assortative communities found by AC-DC-SBM and DC-SBM. The standard DC-SBM produces much fewer assortative communities on average (2.43 compared to 3.76). As discussed earlier in this chapter, AC-DC-SBM only enforces constraints on the block-model parameters $\mathbf{\Omega}$, and therefore does not systematically guarantee assortative partitions. Yet, non-assortative partitions are heavily affected from a likelihood perspective and therefore generally avoided. Finally, remark that modularity maximization always produces assortative solutions, but its equivalence to the PPM (with only two parameters driving the distribution of the edges) limits its ability to fit more general SBMs. Among these alternatives, AC-DC-SBM appears to find a trade-off between insufficient and excessive expressiveness.

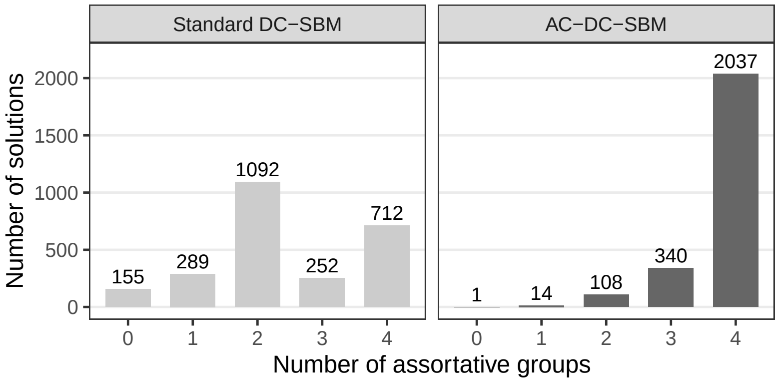


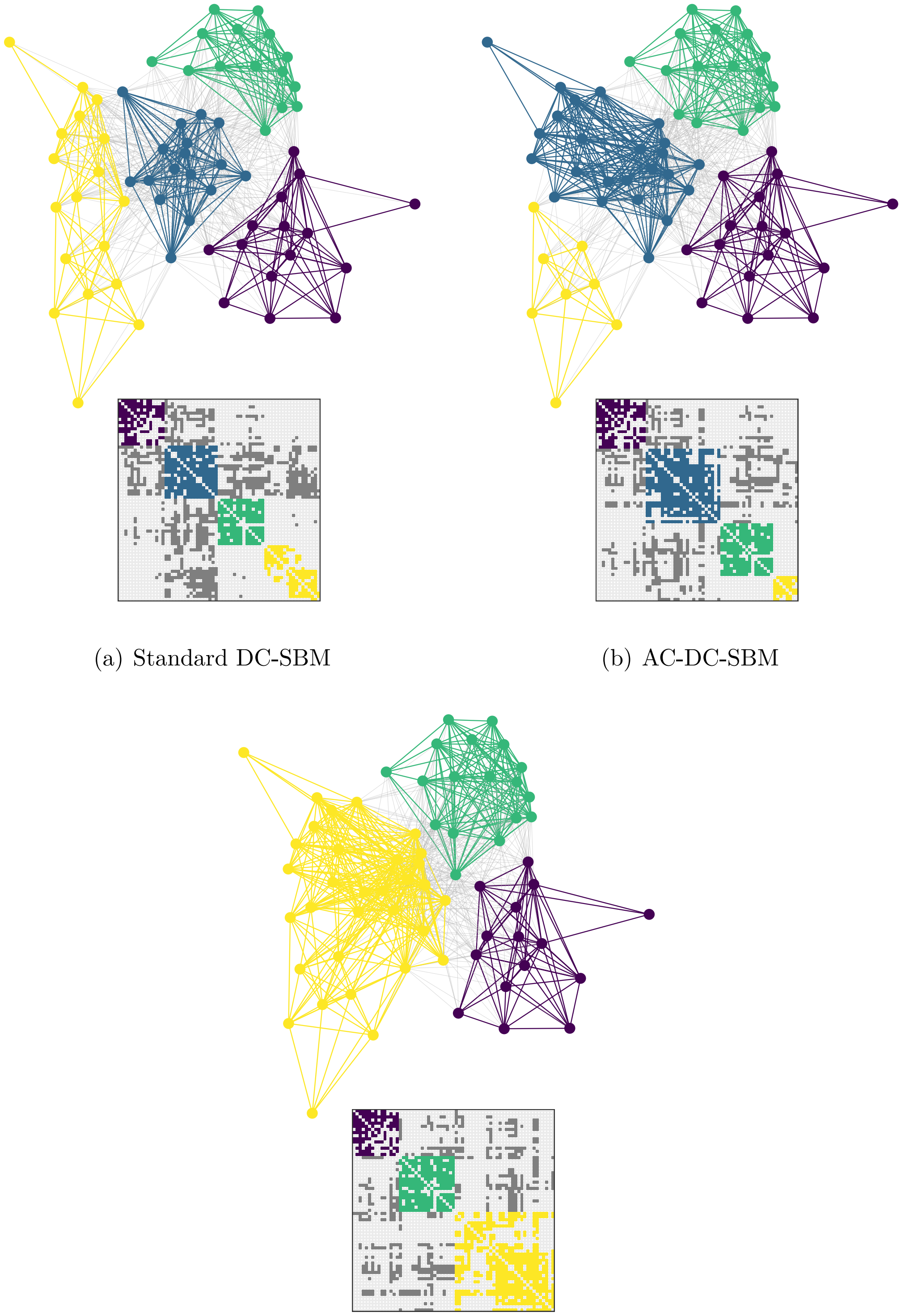
Figure 2.5: Distribution of the number of assortative communities found by AC-DC-SBM and DC-SBM on networks generated from SBMs.

2.5.3 Brain Cortex Networks

Many real-world networks are known to present assortative structures, e.g., in applications to module or community detection in brain cortex networks, protein-protein interaction, and metabolic networks [16, 35, 37, 59]. We analyze in this section the case of the "cats cortex network", which is known to have an assortative structure and is divided into four main functional areas: visual, auditory, frontolimbic, and somatosensory-motor duties [39]. The network is obtained from the cortico-cortical connectivity pattern described by Scannell et al. [61], based on 1139 cortico-cortical connections and 65 cortical areas. As in most community detection tasks, the ground truth in this network is not available. In fact, there is no unique "correct" partitioning [55], but different algorithms can allow to highlight different underlying structures.

Figure 2.6 reports the communities found with the standard DC-SBM, the AC-DC-SBM and modularity maximization models on this dataset. For each model, we performed 100 optimization runs and registered the best solution (in terms of likelihood or modularity).

The best solution obtained with the standard DC-SBM is visibly non-assortative. The minimum value found along the $\mathbf{\Omega}$ diagonal is 1.5060, whereas the maximum value in the off-diagonal is 1.9050. The size of each group is similar, and one disassortative community acts as a "hub" for edges that flow between groups. In contrast, the partition produced by the AC-DC-SBM satisfies the strong assortativity conditions. The minimum value of the $\mathbf{\Omega}$ diagonal is 2.0196, and the maximum value in the off-diagonal is 1.7152. This solution includes communities of different sizes with edges which are more evenly distributed between groups. Two mutually-disconnected community pairs are also identified (green-yellow and purple-yellow). Finally, the



(a) Standard DC-SBM

(b) AC-DC-SBM

(c) Modularity maximization model

Figure 2.6: The best among 100 network partitions found by different models in the cats cortex network.

modularity-maximization approach leads to the most assortative partitioning of this network. Yet, since the model does not take K into consideration, this partitioning contains only three groups, contrasting with the four functional areas which were originally expected.

2.6 Concluding Remarks

Assortativity constraints arise as a natural approach to guide maximum-likelihood algorithms away from spurious local minima on networks which have a presupposed assortative structure. In this chapter, we have shown that these constraints can be effectively handled with tailored local optimization and interior point methods. Our experiments show that the resulting AC-DC-SBM significantly outperforms unconstrained community detection methods in lightly assortative graphs, especially in regimes which are close to the detectability threshold. In these circumstances, the classic SBM has a strong tendency to converge towards non-assortative solutions, while the modularity maximization model does not generalize well to graphs in which the number of edges between groups widely varies. On the practical example of a brain cortex network, the proposed AC-DC-SBM reveals drastically different community structures which were not identified by other algorithms.

The research perspectives related to this work are numerous. We recommend to further evaluate the impact of assortativity constraints on known phase transitions and thresholds. We also recommend to investigate different algorithmic paradigms to improve the solution of this constrained maximum likelihood formulation, and to pursue the study of the AC-DC-SBM in a wider range of application contexts.

3
Semi-supervised Clustering with Inaccurate Pairwise Annotations

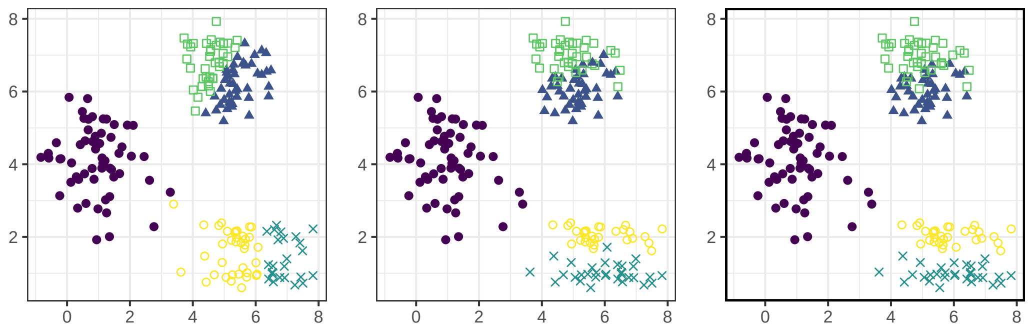
3.1
Introduction

Data clustering aims at systematically grouping a set of data samples such that samples with similar features are placed within the same cluster, whereas samples with a certain degree of separability are allocated to different clusters. Although clustering is an unsupervised learning task, situations exist in which partial annotations are given with the dataset [62], leading to semi-supervised models.

In particular, relational information in the form of pairwise constraints are regularly used: *must-link* constraints state that a pair of data samples must belong to the same cluster, whereas *cannot-link* constraints separate pairs of data samples into different groups. Relational information is usually provided by domain experts who introduce such semi-supervision in domains where it is difficult, time-consuming, or expensive to measure the actual classes accurately [8, 9]. Incorporating relational supervision can bring significant benefits. Figure 3.1, for example, compares clustering solutions obtained without and with semi-supervised learning, on a dataset with 200 samples and 600 random pairwise annotations. In this example, the relational information guides the clustering algorithm out of a local minimum of the unsupervised model toward a solution close to the ground-truth.

The present chapter focuses on the use of relational information in clustering. We consider a regime in which *experts* or automated procedures provide pairwise annotations indicating whether pairs of observations belong to the same group or not. This regime presents two notable characteristics: First, the annotators are not entirely accurate, so the relational information is given with some level of trust. Second, they have a limited work capacity, so only a small amount of pairwise relational information is available.

Some previous works focused on semi-supervised clustering settings with relational information, especially on variants of the minimum sum-of-squares clustering (MSSC) model with additional pairwise constraints [8, 10, 57, 66].



(a) Unsupervised clustering (b) Pairwise-clustering (c) Ground-truth

Figure 3.1: Different partitions in a mixture of spherical and ellipsoidal Gaussians.

The K-means algorithm [32] is a well-known local optimizer of this formulation, and successive improvements of this solution method have been proposed over the years [30, 44, 53, 65]. However, most of these studies incorporate pairwise information in classical search algorithms such as K-means through additional ad hoc constraints or soft penalty factors. By doing so, these approaches lack a probabilistic interpretation and may fail in the presence of noisy and scarce supervision due to erroneous binding constraints. Similarly, soft penalties depend largely on parameter choices that adequately balance the value of the clustering objective function and the number of violated constraints.

To cope with these issues, we introduce a maximum-likelihood approach for a generative model that assumes that data samples are generated by spherical Gaussian distributions. The *must-link* and *cannot-link* constraints occur between a pair of data samples with probabilities that depend only on the groups that contain the samples. To model the presence of *must-link* and *cannot-link* relations, we assume graphs generated by stochastic block models (SBMs) and integrate prior beliefs to represent possible knowledge of the experts' accuracy. We further propose efficient solution techniques for this model based on the HG-means approach [30], a state-of-the-art algorithm for the MSSC model that enhances the classical K-means approach through successive restarts from promising starting points obtained by recombination.

Finally, we conduct extensive computational experiments by applying the proposed model to synthetic and real-world datasets to measure how relational information affects clustering. We show that pairwise annotations can significantly improve clustering performance, even when given only a small amount of imperfect supervision. Incorporating pairwise annotations can also reveal clustering structures not detected by unsupervised approaches, as demonstrated on a real-world dataset. Finally, we show that incorporating prior knowledge regarding the experts' accuracy further guides the clustering process toward more accurate partitioning.

3.2
Related Works and Background

Several clustering formulations have been proposed to exploit pairwise information, e.g., based on expectation-maximization (EM) [9, 63], spectral clustering [43, 67], or affinity propagation [6, 27].

Some previous works have adapted the MSSC objective function to incorporate pairwise constraints. Wagstaff et al. [66] proposed a variant of the K-means algorithm that imposes that no constraint is violated. However, such a model may fail to find a feasible solution. Basu et al. [8] and Hiep et al. [33] included a penalty term that is either uniform or proportional to the distance between samples in the dataset. Bilenko et al. [10] studied the MSSC with pairwise constraints and proposed a metric-adaptive penalty factor according to which the penalty of a violated *must-link* is greater for two distant samples than for two close samples. An analogous notion holds for *cannot-links*. Pelleg and Baras [57] also explored an extension of K-means in which the violated pairwise constraints are tentatively solved by moving a cluster's centroid to change the regions of the feature space covered by the clusters and thereby satisfy the constraints.

Bai et al. [7] included supervision from different sources (pairwise constraints, positive labeling, and negative labeling) in a pairwise relational matrix representation. For the resulting optimization problem, the authors proposed eigenvalue decomposition methods that jointly maximize within-cluster similarity and the consensus among the different supervision. Shental et al. [63] modified the Gaussian mixture model (GMM) likelihood to incorporate *must-link* and *cannot-link* constraints and designed an EM algorithm with tailored update rules to handle these constraints. *Must-link* constraints are handled by collapsing data samples through transitive closure, whereas *cannot-links* are described through Markov networks. However, erroneous pairwise relations can strongly affect the results of the algorithm.

All these approaches are adaptations of the MSSC and GMM formulations to cluster data samples with additional pairwise constraints. Thus, the relational information is incorporated into the formulation to find a partition (e.g., by using the violation of pairwise constraints as penalty factors). An alternative way to jointly consider the data features and the relational information is to model the observed data from a probabilistic perspective. According to this perspective, the features and pairwise relations are assumed to come from a generative model, which is fit to the data.

SBMs [34, 52] are general classes of random graph models commonly used to detect clusters based only on relational information. When such graphs have some structure, fitting the parameters of an SBM to empirical graphs is widely adopted to reveal blocks (clusters). In the canonical form of SBMs, the expected number of edges between two samples is determined solely by the blocks to which they belong. In this way, samples within each block are statistically equivalent in terms of their connectivity patterns. SBMs are regularly used to recover meaningful information from complex graphs and are also a natural modeling choice for community detection. The surveys of Abbe [1] and of Lee and Wilkinson [40] discuss key concepts and solution approaches in stochastic block modeling. Different types of algorithms can be used to fit SBMs based on Markov chain Monte Carlo approaches [47, 52, 56], variational inference [2, 68], belief propagation [19], spectral clustering [41, 58, 60], or semidefinite programming [14, 15], among others.

Previous studies have proposed to couple SBMs with data features (also referred to as meta-data). Stanley et al. [64] presented a probabilistic model that combines relational information and data features within a "soft membership" formulation. In the derived model, SBM probabilities define the graph connectivity, and Gaussian parameters describe the features. The authors employ EM algorithms to maximize the resulting likelihood function. Although EM works well for estimating the Gaussian parameters, the computation of the conditional distributions to get the assignment probabilities is not tractable with SBMs [18]. They therefore use a variational approach that optimizes a lower bound of the SBM likelihood function.

Contisciani et al. [17] introduced a probabilistic model for community detection in multi-layer graphs, combining sample features with relational information, where the sample features are categorical. Each category has a probability of being observed in a community, while an SBM variant serves to model relational information. Thus, the proposed model includes two independent likelihood functions and assumes conditional independence of the observed features and networks. Given that each likelihood may differ in magnitude, the authors propose using a weight—tuned by cross-validation—that inclines the model toward one of the formulations. As a consequence, this approach diverges from a maximum-likelihood perspective.

The techniques described above represent fundamental advances in semi-supervised models and methods. However, they either involve hard constraints and cannot handle imprecise annotations, or they depend on soft penalty factors that are difficult to calibrate. When noisy and inaccurate information is part of the problem, adopting a principled probabilistic model is advantageous. In what follows, we fill this gap and propose a maximum-likelihood approach that incorporates experts' annotations.

3.3 Proposed Model

In the pairwise-constrained clustering problem, we are given a set $\boldsymbol{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ with N data samples in $\mathbb{R}^D$ along with a symmetric adjacency matrix $\boldsymbol{A} \in \mathbb{N}^{N,N}$ representing some relational information between the data samples, where the entry A_{ij} indicates the number of existing connections (edges) between data samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. Typically, pairwise constraints express some hard association. For example, they indicate whether two samples must be assigned to the same cluster or to different clusters. We then aim to partition the data samples into K disjoint clusters $\mathcal{C} = \{C_1, \ldots, C_K\}$ with the goal of optimizing a given clustering criterion. One way to formalize this problem is to define a likelihood function and fit this function's parameters to the observed data.

Gaussian Mixture Model. The Gaussian Mixture Model (GMM) is a widely used probabilistic model that assumes data samples generated by a finite number of Gaussian distributions. The model parameters are the mean points and the covariance matrices of each cluster, and the assignment of samples to clusters is a latent variable. In this work, we explore the hard-membership version of the GMM, which assumes that each data sample is assigned to exactly one cluster, so that the latent assignment variable becomes binary. It is well known that maximizing the likelihood of the hard-membership GMM (also referred to as the MSSC) approximates the ordinary GMM, and algorithms such as K-means act as a variational expectation-maximization in the GMM [46]. The log-likelihood function for the hard-membership GMM can be calculated as per Bishop [11]:

$$\log P(\boldsymbol{X}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{Z}) = \sum_{i}^{N} \sum_{r}^{K} z_{ir} \log \mathcal{N}(\boldsymbol{x}_i|\boldsymbol{\mu}_r, \boldsymbol{\Sigma}_r), \tag{3-1}$$

where $\boldsymbol{Z} \in \mathbb{R}^{N,K}$ is the binary cluster indicator such that each entry $z_{ir} \in \{0, 1\}$ takes the value 1 if and only if sample i belongs to cluster r, so $\sum_r^K z_{ir} = 1 \; \forall \, i \in \{1, \ldots, N\}$. The variables $\boldsymbol{\mu} = \{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K\}$ and $\boldsymbol{\Sigma} = \{\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_K\}$ contain the means and covariances of the Gaussian components, respectively, and

$\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{\Sigma})$ refers to the multivariate Gaussian probability density function:

$$\mathcal{N}(\boldsymbol{x}_i|\boldsymbol{\mu}_r,\boldsymbol{\Sigma}_r)=\frac{e^{-\frac{1}{2}(\boldsymbol{x}_i-\boldsymbol{\mu}_r)^{\mathbf{T}}\boldsymbol{\Sigma}_r^{-1}(\boldsymbol{x}_i-\boldsymbol{\mu}_r)}}{(2\pi)^{\frac{D}{2}}\det(\boldsymbol{\Sigma}_r)^{\frac{1}{2}}}. \tag{3-2}$$

For the special case of spherical GMMs, $\boldsymbol{\Sigma}_r=\sigma_r^2\mathbf{I}\ \forall\ r\in\{1,\ldots,K\}$, and the log-likelihood function may be expressed as

$$\begin{aligned}\log P(\boldsymbol{X}|\boldsymbol{\mu},\boldsymbol{\sigma},\boldsymbol{Z})&=\sum_i^N\sum_r^K z_{ir}\log\left(\frac{e^{-\|\boldsymbol{x}_i-\boldsymbol{\mu}_r\|^2/2\sigma_r^2}}{(2\pi)^{D/2}\sigma_r^D}\right)\\ =&\sum_i^N\sum_r^K z_{ir}\left(-\frac{\|\boldsymbol{x}_i-\boldsymbol{\mu}_r\|^2}{2\sigma_r^2}-\frac{D}{2}\log(2\pi)-D\log(\sigma_r)\right).\end{aligned} \tag{3-3}$$

When the assignments are fixed, as seen in Bishop [11], the maximum of this log-likelihood function occurs when

$$\hat{\boldsymbol{\mu}}_r=\frac{\sum_i^N z_{ir}\boldsymbol{x}_i}{\sum_i^N z_{ir}} \tag{3-4}$$

and

$$\hat{\sigma}_r^2=\frac{\sum_i^N z_{ir}\|\boldsymbol{x}_i-\boldsymbol{\mu}_r\|^2}{D\sum_i^N z_{ir}}. \tag{3-5}$$

Therefore, suppressing the constant $D\log(2\pi)$ and rearranging terms, we obtain:

$$\log P(\boldsymbol{X}|\boldsymbol{Z})\propto-\frac{1}{2}\sum_i^N\sum_r^K z_{ir}\left(\frac{\|\boldsymbol{x}_i-\hat{\boldsymbol{\mu}}_r\|^2}{\hat{\sigma}_r^2}-2D\log(\hat{\sigma}_r)\right). \tag{3-6}$$

Stochastic Block Models. The likelihood function of a GMM is a well-known clustering formulation when data samples have continuous features. To incorporate pairwise constraints into our semi-supervised setting, we now briefly review SBMs, a family of probabilistic models used to detect structure in graphs, and then proceed toward a unified formulation that considers both feature-based samples and relational information.

In its most fundamental form, an SBM considers N data samples and K groups, where each sample is assigned to one group. Then, we assume undirected edges placed between two samples at random with expected value ω_{rs} that depends only on groups r and s to which the data samples belong [51]. Fitting an SBM corresponds to finding the latent membership of data samples and the block-model parameters $\boldsymbol{\Omega}$ that best fit an observed graph [1], where $\boldsymbol{\Omega}$ is a $K\times K$ symmetric matrix with entries ω_{rs}. For an observed adjacency matrix $\boldsymbol{A} \in \mathbb{N}^{N,N}$ representing a graph with m possibly weighted edges, the log-likelihood function of the SBM can be expressed as per Karrer and Newman [36] (see the appendix B.1 for calculations):

$$\log P(\boldsymbol{A}|\boldsymbol{\Omega}, \boldsymbol{Z}) = \frac{1}{2} \sum_{rs}^{K} \sum_{ij}^{N} \left(A_{ij} \log(\omega_{rs}) - \omega_{rs} \right) z_{ir} z_{js}, \tag{3-7}$$

where parameters $\boldsymbol{Z}$ and $\boldsymbol{\Omega}$ are the latent variables, with $\boldsymbol{Z} \in \mathbb{R}^{N,K}$ being the binary cluster indicator, and ω_{rs} an entry of $\boldsymbol{\Omega}$ representing the expected number of edges between two samples in clusters r and s. Note that we opt for the ordinary SBM formulation, which differs from the assortative DC-SBM presented in Chapter 2 with node degree correction. We will see later that, as we consider sparse graphs, the introduction of degree correction has virtually no effect under this scenario. Therefore, we opt for the simpler formulation.

If we fix the assignment $\boldsymbol{Z}$ in Equation (3-7), then the maximum-likelihood values of ω_{rs} can be found by differentiation:

$$\hat{\omega}_{rs} = \frac{m_{rs}}{n_r n_s}, \tag{3-8}$$

where $m_{rs} = \sum_{ij}^{N} A_{ij} z_{ir} z_{js}$ is the number of edges between clusters r and s, and $n_r = \sum_{i}^{N} z_{ir}$ is the number of samples in cluster r. Using the closed form of $\boldsymbol{\Omega}$ from Equation (3-8), the log-likelihood of the SBM can be rewritten as (see the appendix B.1 for calculations)

$$\begin{aligned} \log P(\boldsymbol{A}|\boldsymbol{Z}) &= \frac{1}{2} \sum_{rs}^{K} \sum_{ij}^{N} \left(A_{ij} \log(\hat{\omega}_{rs}) - \hat{\omega}_{rs} \right) z_{ir} z_{js} \\ &= \frac{1}{2} \sum_{rs}^{K} \sum_{ij}^{N} \left(A_{ij} \log\left(\frac{m_{rs}}{n_r n_s} \right) \right) z_{ir} z_{js}. \end{aligned} \tag{3-9}$$

3.3.1 Experts' Annotations Setting

Our proposed generative model considers a set $\boldsymbol{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ of N samples in $\mathbb{R}^D$, along with two independent graphs $\boldsymbol{A}^+$ and $\boldsymbol{A}^-$ that represent the *must-link* and *cannot-link* relations in the form of adjacency matrices. These annotations are produced by experts on a subset of sample pairs. We remind that we now need two matrices for encoding the *must-link* and *cannot-link* relations. Therefore, we must fit two SBMs instead of one, as described earlier in the general pairwise-constrained clustering problem. The complete generative process can be described as follows:

- For each $i \in \{1, \ldots, N\}$:
  - With the probability $1/K$ that each Gaussian component get picked, select component $r \in \{1, \ldots, K\}$ and set $\hat{y}_i = r$ as the ground-truth;
  - Generate a D-dimensional sample $\boldsymbol{x}_i$ from component r:

$$\boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{\mu}_r, \sigma_r^2). \tag{3-10}$$

- For each sample pair $(\boldsymbol{x}_i, \boldsymbol{x}_j)$ selected independently and with uniform probability, an expert labels the pair as a *must-link* or a *cannot-link* relation according to a Bernoulli distribution, which is defined based on the groups to which the samples belong:

$$\begin{cases} A^+_{ij} = \text{Bernoulli}(p_{\hat{y}_i \hat{y}_j}), \\ A^-_{ij} = 1 - A^+_{ij}, \end{cases} \tag{3-11}$$

  in which $p_{rs} \in [0, 1]$ is the probability of marking a pair of samples as a *must-link* given that the samples belong to groups r and s. Analogously, $1 - p_{rs}$ is the probability of marking a pair of samples in groups r and s as a *cannot-link*. Typically, $p_{rr} \geq p_{rs}$ when $r \neq s$.

Assuming that m annotations are generated independently and uniformly between sample pairs, the expected number of *must-link* edges between an arbitrary pair of samples from groups r and s is βp_{rs}, with $\beta = 2m/[N(N+1)]$. Here, we consider the sampling with replacement, which allows the presence of self-edges. This condition simplifies the calculations and has only a marginal impact on the model. Similarly, the expected number of *cannot-link* edges is $\beta(1 - p_{rs})$. We can thus model the experts' annotations setting by using two stochastic block models with matrices $\boldsymbol{\Omega}^+$ and $\boldsymbol{\Omega}^-$ for the *must-link* and *cannot-link* graphs, respectively. In this case, $\omega^+_{rs} \sim \beta p_{rs}$ and $\omega^-_{rs} \sim \beta(1 - p_{rs})$. Since multiple experts provide annotations with replacement, we obtain Poisson-distributed matrices $\boldsymbol{\Omega}^+$ and $\boldsymbol{\Omega}^-$. Note, however, that the two graphs produced are not independent because a "failure" in a Bernoulli trial generates an edge in the *cannot-link* graph. Nonetheless, independence holds between annotations because of sample pair selections with replacement, such that we can reasonably approximate the experts' annotations by two independent SBMs with parameters $\boldsymbol{\Omega}^+$ and $\boldsymbol{\Omega}^-$ for *must-link* and *cannot-link* relations, respectively:

$$P(\boldsymbol{X}, \boldsymbol{A}^{+}, \boldsymbol{A}^{-}|\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\Omega}, \boldsymbol{Z}) = P(\boldsymbol{X}|\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{Z}) \times P(\boldsymbol{A}^{+}|\boldsymbol{\Omega}^{+}, \boldsymbol{Z}) \times P(\boldsymbol{A}^{-}|\boldsymbol{\Omega}^{-}, \boldsymbol{Z}). \tag{3-12}$$

Hereinafter, we consider $\mathcal{L}(\cdot) = \log P(\cdot)$ to refer to a log-likelihood function, $\boldsymbol{A} = \{\boldsymbol{A}^{+}, \boldsymbol{A}^{-}\}$ to represent the *must-link* and *cannot-link* graphs, and $\boldsymbol{\Omega} = \{\boldsymbol{\Omega}^{+}, \boldsymbol{\Omega}^{-}\}$ to represent the two SBM matrices. Thus, the resulting log-likelihood function, which considers two independent SBM graphs, is

$$\begin{aligned}\mathcal{L}(\boldsymbol{X}, \boldsymbol{A}|\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\Omega}, \boldsymbol{Z}) \propto & -\sum_{i}^{N}\sum_{r}^{K}\left(\frac{\|\boldsymbol{x}_i - \boldsymbol{\mu}_r\|^2}{\sigma_r^2} + 2D\log(\sigma_r)\right) z_{ir} \\ & + \sum_{rs}^{K}\sum_{ij}^{N}\left(A_{ij}^{+}\log(\omega_{rs}^{+}) - \omega_{rs}^{+}\right) z_{ir} z_{js} \\ & + \sum_{rs}^{K}\sum_{ij}^{N}\left(A_{ij}^{-}\log(\omega_{rs}^{-}) - \omega_{rs}^{-}\right) z_{ir} z_{js},\end{aligned} \tag{3-13}$$

where we removed the constant $\frac{1}{2}$ in front of all terms. The variables $\boldsymbol{\mu}_r$ and σ_r are obtained from Equations (3-4) and (3-5), respectively, and ω_{rs}^{+} and ω_{rs}^{-} are obtained from Equation (3-8). As a consequence, we can write this log-likelihood as

$$\mathcal{L}(\boldsymbol{X}, \boldsymbol{A}|\boldsymbol{Z}) = \log P(\boldsymbol{X}|\boldsymbol{Z}) + \log P(\boldsymbol{A}^{+}|\boldsymbol{Z}) + \log P(\boldsymbol{A}^{-}|\boldsymbol{Z}). \tag{3-14}$$

3.3.2 *Prior* Knowledge of Experts' Accuracy

Although the SBMs are used to infer partitions of any structure, it is common in practice to have an estimate of the experts' accuracy. In some circumstances, we may reasonably assume to have pre-evaluated the experts' accuracy before the annotation procedure. Consequently, we propose an extension of model (3-13) that incorporates a *prior* belief regarding the accuracy of annotations. We first consider the maximum posterior estimate of parameters $\boldsymbol{\Omega}$ and $\boldsymbol{Z}$ in the SBM:

$$P(\boldsymbol{\Omega}, \boldsymbol{Z}|\boldsymbol{A}) \propto P(\boldsymbol{A}|\boldsymbol{\Omega}, \boldsymbol{Z})P(\boldsymbol{\Omega}, \boldsymbol{Z}), \tag{3-15}$$

where the joint prior distribution is

$$P(\boldsymbol{\Omega}, \boldsymbol{Z}) = P(\boldsymbol{\Omega}|\boldsymbol{Z})P(\boldsymbol{Z}), \tag{3-16}$$

and we assume that $P(\boldsymbol{Z})$ has the same probability for any assignment $\boldsymbol{Z}$ and thus is treated as a constant. As in Peixoto [56], we opt for the following form of a prior function:

$$\begin{aligned} P(\boldsymbol{\Omega}|\boldsymbol{Z}) &= \prod_{r\leq s} \lambda_{rs}(\boldsymbol{Z},p)\ e^{-\lambda_{rs}(\boldsymbol{Z},p)\omega_{rs}} \\ &= \prod_{rs} \left(\lambda_{rs}(\boldsymbol{Z},p)\ e^{-\lambda_{rs}(\boldsymbol{Z},p)\omega_{rs}}\right)^{\frac{1}{2}(1+\delta_{rs})}, \end{aligned} \tag{3-17}$$

where δ_{rs} is the Kronecker delta, $\lambda_{rs}(\boldsymbol{Z},p)$ is the rate parameter in the exponential distribution, and p is the experts' accuracy for any pair of clusters r and s, such that $p = 1.0$ represents totally-accurate experts. We remind that $1/\lambda_{rs}(\boldsymbol{Z},p)$ is the expected (mean) value in the exponential distribution and therefore represents the prior expected number of edges between a pair of samples in clusters r and s. Although the experts' accuracy is fixed, the values we choose for our priors depends on $\boldsymbol{Z}$, since the assignment choices impact the size of the clusters and, therefore, the expected total number of annotations. This dependence occurs because SBMs have two sets of parameters, making our prior distribution conditioned on $\boldsymbol{Z}$. For the sake of brevity, we will use the short form $\lambda_{rs} = \lambda_{rs}(\boldsymbol{Z},p)$ in the remainder of this section. Since we have two graphs, we use λ^+_{rs} and λ^-_{rs} to refer to our priors in the *must-link* and *cannot-link* graphs, respectively. Suitable values for λ^+_{rs} and λ^-_{rs} are discussed later in this section. This leads to the following posterior distribution:

$$\begin{aligned} P(\boldsymbol{\Omega},\boldsymbol{Z}|\boldsymbol{A}) \propto P(\boldsymbol{A}^+|\boldsymbol{\Omega}^+,\boldsymbol{Z}) \prod_{rs}\left(\lambda^+_{rs}e^{-\lambda^+_{rs}\omega^+_{rs}}\right)^{\frac{1}{2}(1+\delta_{rs})} \\ \times P(\boldsymbol{A}^-|\boldsymbol{\Omega}^-,\boldsymbol{Z}) \prod_{rs}\left(\lambda^-_{rs}e^{-\lambda^-_{rs}\omega^-_{rs}}\right)^{\frac{1}{2}(1+\delta_{rs})}, \end{aligned} \tag{3-18}$$

and therefore to the following log-posterior with the observed features $\boldsymbol{X}$:

$$\begin{aligned} \mathcal{L}(\boldsymbol{\mu},\boldsymbol{\sigma},\boldsymbol{\Omega},\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{A}) \propto -\sum_i^N\sum_r^K\left(\frac{\|\boldsymbol{x}_i-\boldsymbol{\mu}_r\|^2}{\sigma_r^2}+2D\log(\sigma_r)\right)z_{ir} \\ +\sum_{rs}\sum_{ij}\left(A^+_{ij}\log(\omega^+_{rs})-\omega^+_{rs}\right)z_{ir}z_{js} \\ +\sum_{rs}\sum_{ij}\left(A^-_{ij}\log(\omega^-_{rs})-\omega^-_{rs}\right)z_{ir}z_{js} \\ +\sum_r \log(\lambda^+_{rr}\lambda^-_{rr}) - \lambda^+_{rr}\omega^+_{rr} - \lambda^-_{rr}\omega^-_{rr} \\ +\sum_{rs} \log(\lambda^+_{rs}\lambda^-_{rs}) - \lambda^+_{rs}\omega^+_{rs} - \lambda^-_{rs}\omega^-_{rs}. \end{aligned} \tag{3-19}$$

In Equation (3-19), we used a constant prior for the mixture of Gaussians, and therefore only take the likelihood into account. The last two summations come from the exponential priors. The optimal value of ω_{rs} in the posterior log-likelihood can then be estimated by differentiation:

$$\hat{\omega}_{rs} = \begin{cases} m_{rs}/(n_r n_s + 2\lambda_{rs}) & \text{if } r = s \\ m_{rs}/(n_r n_s + \lambda_{rs}) & \text{otherwise,} \end{cases} \tag{3-20}$$

where we substitute $\hat{\omega}_{rs}$ with the corresponding parameter ω^+_{rs} or ω^-_{rs}, depending on the graph (likewise for m_{rs} and λ_{rs}).

Parametrization of the priors. We estimate λ^+_{rs} and λ^-_{rs} according to our prior beliefs regarding the expert's proficiency p and the assignment $\boldsymbol{Z}$. In the *must-link* graph, λ^+_{rs} can be expressed as:

$$\lambda^+_{rs} = \begin{cases} 1/f^+_{\text{IN}}(\boldsymbol{Z}, p) & \text{if } r = s \\ 1/f^+_{\text{OUT}}(\boldsymbol{Z}, p) & \text{otherwise,} \end{cases} \tag{3-21}$$

whereas, analogously, λ^-_{rs} is expressed as:

$$\lambda^-_{rs} = \begin{cases} 1/f^-_{\text{IN}}(\boldsymbol{Z}, p) & \text{if } r = s \\ 1/f^-_{\text{OUT}}(\boldsymbol{Z}, p) & \text{otherwise.} \end{cases} \tag{3-22}$$

The functions $f^+_{\text{IN}}(\boldsymbol{Z}, p)$ and $f^-_{\text{IN}}(\boldsymbol{Z}, p)$ represent a prior knowledge of the expected number of edges between two samples in the same group, for the *must-link* and *cannot-link* graphs, respectively. Similarly, $f^+_{\text{OUT}}(\boldsymbol{Z}, p)$ and $f^-_{\text{OUT}}(\boldsymbol{Z}, p)$ represent the prior expected number of edges between two samples in different groups, for the two graphs. In our priors estimation, we expect that the annotation mistakes occur in the same proportion among the clusters. Namely, every cluster, or pair of clusters, has a priori the same percentage of error. Due to the experts' annotations setting, the following relationship holds between $f^+_{\text{IN}}(\boldsymbol{Z}, p)$ and $f^+_{\text{OUT}}(\boldsymbol{Z}, p)$ in the *must-link* graph (see the appendix B.2 for the demonstration):

$$f^+_{\text{IN}}(\boldsymbol{Z}, p) = \frac{p}{1-p} f^+_{\text{OUT}}(\boldsymbol{Z}, p). \tag{3-23}$$

Analogously, for the *cannot-link* graph, we have:

$$f^-_{\text{IN}}(\boldsymbol{Z}, p) = \frac{1-p}{p} f^-_{\text{OUT}}(\boldsymbol{Z}, p). \tag{3-24}$$

The number of pairs within and between groups given by $\boldsymbol{Z}$, and the number of *must-link* annotations m^+ and *cannot-link* annotations m^- also lead to the following relations:

$$f^+_{\text{IN}}(\boldsymbol{Z},p)\sum_r \frac{n_r(n_r+1)}{2} + f^+_{\text{OUT}}(\boldsymbol{Z},p)\sum_{r<s} n_r n_s = m^+, \tag{3-25}$$

$$f^-_{\text{IN}}(\boldsymbol{Z},p)\sum_r \frac{n_r(n_r+1)}{2} + f^-_{\text{OUT}}(\boldsymbol{Z},p)\sum_{r<s} n_r n_s = m^-, \tag{3-26}$$

where $n_r = \sum_i z_{ir}$ is the number of samples in group r. Then, combining Equations (3-23–3-26) leads to:

$$f^+_{\text{IN}}(\boldsymbol{Z},p) = m^+ \Big/ \left(\sum_r \frac{n_r(n_r+1)}{2} + \frac{(1-p)}{p}\sum_{r<s} n_r n_s\right), \tag{3-27}$$

$$f^-_{\text{IN}}(\boldsymbol{Z},p) = m^- \Big/ \left(\sum_r \frac{n_r(n_r+1)}{2} + \frac{p}{(1-p)}\sum_{r<s} n_r n_s\right). \tag{3-28}$$

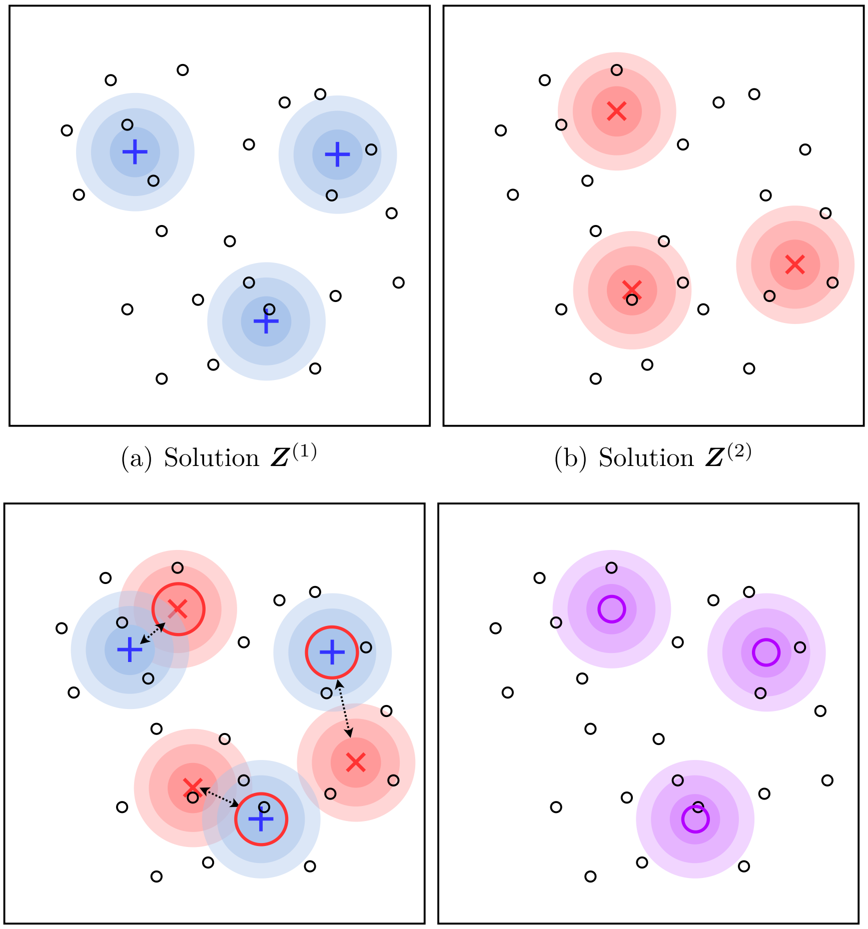
3.4 Solution Approach

To solve model (3-19), we adapt the hybrid genetic search of Gribel and Vidal [30], which has demonstrated state-of-the-art performance on the minimum-sum-of-squares clustering problem. As summarized in Algorithm 2, the method begins with a set of Π_1 initial solutions obtained by using the K-means algorithm starting from different centers, followed by local search. After this initialization phase, the algorithm iteratively generates new solutions via three successive steps: crossover, mutation, and local search. Upon attaining the maximum population size Π_2, the best Π_1 solutions in terms of log-likelihood are preserved to ensure elitism and selection pressure, and the remaining solutions are discarded. The algorithm terminates after a fixed number of iterations. The remainder of this section details each operator.

Crossover. The algorithm selects two random parent solutions $\boldsymbol{Z}^{(1)}$ and $\boldsymbol{Z}^{(2)}$ in the population and applies a crossover to them to create a new solution. This operator works as follows (see Figure 3.2):

– **Step 1**. It first solves a bipartite matching problem to pair up the centers of the two solutions. Let $G = (\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{E})$ be a complete bipartite graph in which the vertex set $\boldsymbol{U} = (\boldsymbol{u}_1, \dots, \boldsymbol{u}_K)$ represents the centers of solution $\boldsymbol{Z}^{(1)}$ and $\boldsymbol{V} = (\boldsymbol{v}_1, \dots, \boldsymbol{v}_K)$ represents the centers of solution $\boldsymbol{Z}^{(2)}$. Each edge $(\boldsymbol{u}_i, \boldsymbol{v}_j) \in \boldsymbol{E}$, for $i \in 1, \dots, K$ and $j \in 1, \dots, K$ represents a possible association of center i from solution $\boldsymbol{Z}^{(1)}$ with center j from solution $\boldsymbol{Z}^{(2)}$. The minimum-cost bipartite matching problem is then solved in graph G by considering the weights of the edges in $\boldsymbol{E}$ as the squared Euclidean distance between the vertices in $\boldsymbol{V}$ and $\boldsymbol{U}$.

- **Step 2**. For each pair obtained in the previous step, the crossover randomly selects one of the two centers with equal probability. This effectively recombines the centers of both parents.
- **Step 3**. Once the new centers are generated, each sample $\boldsymbol{x}_i$ is assigned to the closest center in terms of Euclidean distance.



(a) Solution $\boldsymbol{Z}^{(1)}$ (b) Solution $\boldsymbol{Z}^{(2)}$

(c) Assignment and random selection (d) Resulting solution

Figure 3.2: Crossover based on exact centers matching.

Mutation. The mutation operator follows the crossover. Its goal is to introduce randomness into the solutions and permit a broader exploration of the search space. We use a special case of the mutation scheme described in Gribel and Vidal [30] in which all samples have an equal chance of being selected as the new center:

1. Select one center for removal with uniform probability.
2. Select a random sample and create a new center at its position.
3. Re-assign each sample to the closest center.

Local Search. The solution generated by the previous steps serves as a starting point for a two-phase local search (Algorithms 3 and 4) that iterates until converging:

1. The algorithm iteratively evaluates each possible relocation of an annotated sample (i.e., a sample involved in at least one pairwise annotation) to a different cluster. Each relocation is applied if it improves the likelihood (see Algorithm 3).
2. Next, the unannotated samples are assigned to their closest cluster, as determined by the distance to the cluster center. The parameters of the Gaussians are then updated based on the new assignments. These two steps are iterated until convergence to a local optimum, making this step of the local search equivalent to a K-means algorithm applied to the unannotated samples (see Algorithm 4).

For notational simplicity, Algorithms 3 and 4 cover the case of log-likelihood maximization (model without priors). Still, the algorithms work analogously for the log-posterior maximization, with the priors updated depending on the given $\boldsymbol{Z}$, according to Equations (3-23)–(3-28).

Algorithm 2 Hybrid-Genetic Search

Input: Feature data: $\boldsymbol{X}$, Adjacency matrices: $\boldsymbol{A}$, Annotated samples: $\mathcal{A}$, Unannotated samples: $\mathcal{U}$, Number of clusters: K, Parameters: Π_1 and Π_2
$\mathcal{S} \leftarrow$ Set with Π_1 initial solutions
repeat
    $\boldsymbol{Z}^{(1)}, \boldsymbol{Z}^{(2)} \leftarrow$ Random solutions from $\mathcal{S}$
    $\boldsymbol{Z} \leftarrow$ Crossover($\boldsymbol{Z}^{(1)}$, $\boldsymbol{Z}^{(2)}$)
    $\boldsymbol{Z'} \leftarrow$ Mutation($\boldsymbol{Z}$)
    Algorithm 3: FitAnnotated($\boldsymbol{X}$, $\boldsymbol{A}$, $\boldsymbol{Z}'$, $\mathcal{A}$)
    Algorithm 4: FitUnannotated($\boldsymbol{X}$, $\boldsymbol{A}$, $\boldsymbol{Z}'$, $\mathcal{U}$)
    Add solution $\boldsymbol{Z}'$ to $\mathcal{S}$
    if $|\mathcal{S}| = \Pi_2$ **then**
        $\mathcal{S} \leftarrow$ Select the best Π_1 solutions
    end if
until Maximum number of iterations is attained
return Best solution $\boldsymbol{Z}^*$ found

Algorithm 3 FitAnnotated: Relocation of Annotated Samples

Input: Feature data: $\boldsymbol{X}$, Adjacency matrices: $\boldsymbol{A}$, Current solution: $\boldsymbol{Z}$, Annotated samples: $\mathcal{A}$
Find parameters $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$, $\boldsymbol{\Omega}^{+}$ and $\boldsymbol{\Omega}^{-}$ maximizing $\mathcal{L}(\boldsymbol{X}, \boldsymbol{A}|\boldsymbol{Z})$
Evaluate log-likelihood with the estimated parameters:
$\mathcal{Q} \leftarrow \mathcal{L}(\boldsymbol{X}, \boldsymbol{A}|\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\Omega}, \boldsymbol{Z})$
repeat
    for $i \in \mathcal{A}$ and $r \in \{1, \ldots, K\}$ in random order **do**
        Consider solution $\boldsymbol{Z}^{\mathrm{R}}$ obtained from $\boldsymbol{Z}$ by relocating sample i to cluster r
        Find parameters $\hat{\boldsymbol{\mu}}$, $\hat{\boldsymbol{\sigma}}$, $\hat{\boldsymbol{\Omega}}^{+}$ and $\hat{\boldsymbol{\Omega}}^{-}$ maximizing $\mathcal{L}(\boldsymbol{X}, \boldsymbol{A}|\boldsymbol{Z}^{\mathrm{R}})$
        Evaluate log-likelihood with the estimated parameters:
        $\mathcal{Q}' \leftarrow \mathcal{L}(\boldsymbol{X}, \boldsymbol{A}|\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{\Omega}}, \boldsymbol{Z}^{\mathrm{R}})$
        if $\mathcal{Q}' > \mathcal{Q}$ **then**
            Apply: $\boldsymbol{\mu} \leftarrow \hat{\boldsymbol{\mu}}$, $\boldsymbol{\sigma} \leftarrow \hat{\boldsymbol{\sigma}}$, $\boldsymbol{\Omega}^{+} \leftarrow \hat{\boldsymbol{\Omega}}^{+}$, $\boldsymbol{\Omega}^{-} \leftarrow \hat{\boldsymbol{\Omega}}^{-}$,
            $\boldsymbol{Z} \leftarrow \boldsymbol{Z}^{\mathrm{R}}$, $\mathcal{Q} \leftarrow \mathcal{Q}'$
        end if
    end for
until No improving relocation has been identified

Algorithm 4 FitUnannotated: Assignment of Unannotated Samples

Input: Feature data: $\boldsymbol{X}$, Adjacency matrices: $\boldsymbol{A}$, Current solution: $\boldsymbol{Z}$, Unannotated samples: $\mathcal{U}$
repeat
    for $i \in \mathcal{U}$ **do**
        $y_i \leftarrow \min_r \left\|\boldsymbol{x}_i - \boldsymbol{\mu}_r\right\|^2$
        Update $\boldsymbol{Z}$ with the new assignment
    end for
    for $r \in \{1, \ldots, K\}$ **do**
        $\boldsymbol{\mu}_r \leftarrow \sum_i^N z_{ir}\boldsymbol{x}_i / \sum_i^N z_{ir}$
        $\sigma_r^2 \leftarrow \sum_i^N \sum_r^K z_{ir} \left\|\boldsymbol{x}_i - \boldsymbol{\mu}_r\right\|^2 / \left(2D \sum_i^N z_{ir}\right)$
    end for
until No change in the solution has been identified
Update log-likelihood with the estimated parameters and the current value of $\boldsymbol{\Omega}$:
$\mathcal{Q} \leftarrow \mathcal{L}(\boldsymbol{X}, \boldsymbol{A}|\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\Omega}, \boldsymbol{Z})$

3.5 Computational Experiments

We conducted computational experiments to investigate two main effects. First, we analyze how the incorporation of relational information affects the performance of the method on datasets that match the ideal conditions of the model (i.e., mixtures of spherical Gaussians). We evaluate the performance of the proposed semi-supervised models as a function of the quality and amount of the information provided and analyze the impact of incorporating prior beliefs. Second, we assess how the model performs on more challenging real data not likely to be generated from spherical Gaussian mixtures. We evaluate the extent to which the model generalizes to treat these cases and discuss some of its limitations.

All algorithms were implemented in Julia (version 1.0.5). The source code is available at `http://github.com/danielgribel/SSC-IPA`.

3.5.1 Evaluation Metrics

We consider three evaluation metrics in our experimental setting: normalized mutual information (NMI) [38], an entropy-based measure to compare two partitions from the sample-group memberships; the Kullback–Leibler (KL) divergence between two Gaussians mixtures using the matching-based approximation of Goldberger et al. [28]; and the centroid index (CI) [25], a discrete measure of the number of different cluster locations between two clustering solutions. A CI of zero indicates that the given partition matches the ground-truth structure. These metrics reflect different aspects of the solutions: NMI compares the partitions (membership variables) with the ground-truth, the KL divergence compares the continuous Gaussian parameters, and the CI is based on the coordinates of the solution centers.

3.5.2 Performance for Mixtures of Spherical Gaussians

In our first set of experiments, we analyze the general performance of Algorithm 2 applied to synthetic datasets that meet ideal conditions (i.e., mixtures of spherical Gaussians).

To generate these datasets, we use overlapping mixtures in which each group has its own dispersion. More precisely, for each group r, we create a D-dimensional mean $\boldsymbol{\mu}_r$ by sampling uniformly over the range $[-1, 1]$. For the dispersion of each group, we sample σ_r^2 uniformly from the range $[0, 5]$. Each data sample is then generated with probability $1/K$ from group r according to the Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_r, \sigma_r^2)$.

Finally, the edges of graphs $\boldsymbol{A}^+$ and $\boldsymbol{A}^-$ are randomly generated via Equation (3-11). We create datasets with different experts' accuracies by defining a parameter $p \in \{0.8, 0.9, 1.0\}$ and setting $p_{rr} = p$ for all r and $p_{rs} = 1 - p$ for all $r \neq s$. In all datasets, the number of samples is set to $N = 200$, the feature-space dimension is set to $D = 10$, and the number of clusters is selected from the set $K \in \{2, 4, 6\}$. For each value of K, we generate 50 Gaussian mixtures, leading to 150 datasets. We define the number of total annotations m (including both *must-links* and *cannot-links*) as a proportion of the number of samples N, in which $m \in \{0,\ N/2,\ N,\ 1.5N,\ 2N,\ \ldots,\ 4N\}$. This experimental setup includes 3750 cases overall, considering all 150 datasets and the possible values of p and m.

Tables 3.1–3.3 report the performance of both proposed models for $p \in \{0.8, 0.9, 1.0\}$. All results correspond to the best log-likelihood solution found after 50 repetitions of Algorithm 2.

Table 3.1 reports the NMI and KL divergence performance for $p = 0.8$, i.e., when we expect a mistake rate of 20% for the experts. For $K = 2$, we observe that the pairwise annotations have a positive impact on clustering

	K = 2		K = 4		K = 6	
Priors:	×	✓	×	✓	×	✓
	NMI					
$m = 0$	0.4808		0.4358		0.4003	
$m = 100$	0.5136	0.5160	0.4323	0.4373	0.3834	0.3878
$m = 200$	0.5497	0.5536	0.4319	0.4436	0.3878	0.3943
$m = 300$	0.5880	0.5894	0.4369	0.4468	0.3870	0.3917
$m = 400$	0.6676	0.6549	0.4687	0.4697	0.3976	0.4064
$m = 500$	0.7260	0.7210	0.4895	0.4865	0.3907	0.4063
$m = 600$	0.7974	0.7978	0.4994	0.5030	0.4006	0.4201
$m = 700$	0.8328	0.8346	0.5296	0.5375	0.4114	0.4216
$m = 800$	0.8616	0.8623	0.5369	0.5479	0.4181	0.4322
	KL divergence					
$m = 0$	0.0998		0.3980		0.7529	
$m = 100$	0.0657	0.0733	0.3414	0.3421	0.7157	0.6671
$m = 200$	0.0528	0.0520	0.3300	0.3062	0.6652	0.6327
$m = 300$	0.0396	0.0383	0.3188	0.3089	0.6093	0.6121
$m = 400$	0.0285	0.0327	0.2614	0.2563	0.5903	0.5790
$m = 500$	0.0154	0.0152	0.2380	0.2365	0.6013	0.5550
$m = 600$	0.0075	0.0077	0.2206	0.2167	0.5637	0.5276
$m = 700$	0.0060	0.0054	0.2092	0.1757	0.5563	0.5413
$m = 800$	0.0037	0.0036	0.2102	0.1679	0.5404	0.4985

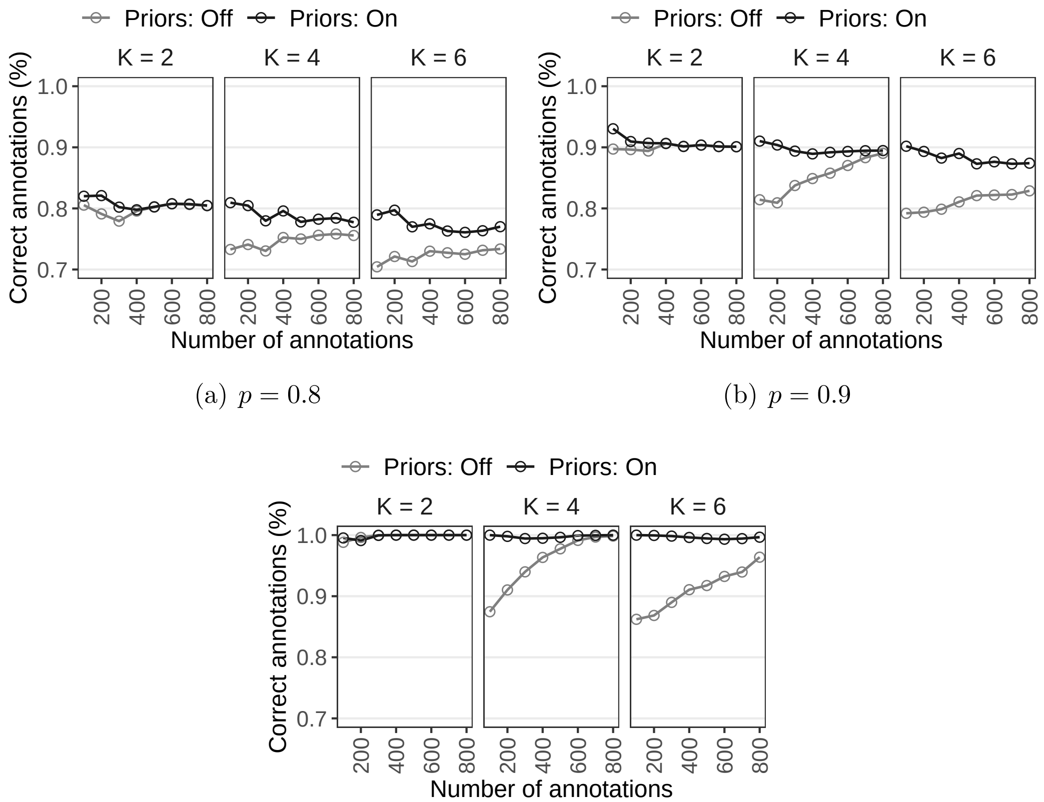
Table 3.1: Average NMI and KL divergence on synthetic datasets, for $p = 0.8$.

	K = 2		K = 4		K = 6	
Priors:	×	✓	×	✓	×	✓
	NMI					
$m = 0$	0.4808		0.4358		0.4003	
$m = 100$	0.5461	0.5483	0.4513	0.4607	0.3930	0.4034
$m = 200$	0.6515	0.6601	0.4672	0.4840	0.4048	0.4089
$m = 300$	0.7546	0.7618	0.4998	0.5140	0.4046	0.4118
$m = 400$	0.8603	0.8678	0.5369	0.5529	0.4248	0.4485
$m = 500$	0.9045	0.9017	0.5998	0.6085	0.4385	0.4562
$m = 600$	0.9381	0.9387	0.6416	0.6694	0.4731	0.4908
$m = 700$	0.9664	0.9659	0.7107	0.7302	0.4768	0.5035
$m = 800$	0.9722	0.9728	0.7608	0.7831	0.4991	0.5274
	KL divergence					
$m = 0$	0.0998		0.3980		0.7529	
$m = 100$	0.0541	0.0537	0.3291	0.2906	0.6637	0.6433
$m = 200$	0.0304	0.0263	0.2951	0.2702	0.6311	0.6043
$m = 300$	0.0117	0.0095	0.2531	0.2157	0.6225	0.5669
$m = 400$	0.0049	0.0041	0.2155	0.1785	0.5545	0.4835
$m = 500$	0.0020	0.0020	0.1635	0.1362	0.5284	0.4684
$m = 600$	0.0012	0.0012	0.1314	0.1004	0.4753	0.4098
$m = 700$	0.0007	0.0007	0.0825	0.0617	0.4823	0.4022
$m = 800$	0.0006	0.0005	0.0562	0.0492	0.4518	0.3476

Table 3.2: Average NMI and KL divergence on synthetic datasets, for $p = 0.9$.

	K = 2		K = 4		K = 6	
Priors:	×	✓	×	✓	×	✓
	NMI					
$m = 0$	0.4808		0.4358		0.4003	
$m = 100$	0.6444	0.5559	0.4683	0.4706	0.4019	0.4156
$m = 200$	0.8140	0.7578	0.5195	0.5218	0.4228	0.4551
$m = 300$	0.9402	0.9311	0.6128	0.6016	0.4642	0.4820
$m = 400$	0.9746	0.9746	0.7075	0.7145	0.4926	0.5270
$m = 500$	0.9936	0.9936	0.8089	0.8289	0.5409	0.5891
$m = 600$	0.9976	0.9976	0.8830	0.9130	0.6120	0.6508
$m = 700$	0.9976	0.9976	0.9375	0.9490	0.6602	0.7322
$m = 800$	1.0000	1.0000	0.9678	0.9749	0.7509	0.8038
	KL divergence					
$m = 0$	0.0998		0.3980		0.7529	
$m = 100$	0.0293	0.0401	0.3213	0.2713	0.6843	0.6002
$m = 200$	0.0078	0.0086	0.2741	0.1939	0.6060	0.4648
$m = 300$	0.0012	0.0015	0.1703	0.1145	0.5468	0.3874
$m = 400$	0.0004	0.0004	0.1039	0.0616	0.4818	0.3122
$m = 500$	0.0001	0.0001	0.0602	0.0238	0.4167	0.2388
$m = 600$	0.0000	0.0000	0.0277	0.0086	0.3182	0.1666
$m = 700$	0.0001	0.0001	0.0110	0.0056	0.2838	0.1110
$m = 800$	0.0000	0.0000	0.0065	0.0026	0.1848	0.0740

Table 3.3: Average NMI and KL divergence on synthetic datasets, for $p = 1.0$.



(a) $p = 0.8$

(b) $p = 0.9$

(c) $p = 1.0$

Figure 3.3: Percentage of correct annotations given by the two proposed models in synthetic datasets.

performance, even with a small amount of information. For 100 annotations, the use of pairwise information leads to an average NMI of approximately 0.51, against 0.48 for the unsupervised model ($m = 0$). For datasets with four clusters, a significant performance enhancement occurs only for $m \geq 400$. With six clusters, the incorporation of pairwise annotations has less impact. Moreover, the NMI slightly decreases for small values of m despite the improved KL divergence. Finally, the inclusion of priors does not lead to large performance differences in this setting. The most visible impact occurs when $K = 4$ and $K = 6$, but only when m is sufficiently large.

Table 3.2 presents the same set of experiments for $p = 0.9$. Since the annotation accuracy is higher than in the previous case, the resulting graphs are more structured. Consequently, semi-supervision translates into a larger gain of performance over the unsupervised model. For $K = 2$ and $m = 100$, the semi-supervised models present an average NMI of approximately 0.55. In the case with two clusters, the proposed models achieve a near-perfect recovery when m is large. Finally, the use of prior information regarding the experts' accuracy had a more significant impact than in the previous case with $p = 0.8$.

As expected and seen in Table 3.3, the difference in performance between the semi-supervised and unsupervised approaches becomes evident in a regime with perfect annotations $p = 1.0$. With $K = 2$ and $m = 100$ annotations, we obtain an average NMI of more than 0.64 without priors. As the number of annotations grows, the semi-supervised solution converges toward the ground-truth, effectively attaining it when $m = 800$ and $K = 2$. Still, when $p = 1.0$, the model with priors suffers from numerical instability because f_{IN}^{-} and f_{OUT}^{+} drop to zero. To circumvent this issue, we use $p = 1 - 10^{-6}$ as an approximation. Despite this adjustment, the penalties represented by $\boldsymbol{\lambda}$ may still remain quite large such that, for small values of K and sparse graphs with many unannotated samples, the priors tend to dominate the other terms in the objective function. This diminishes the impact of the Gaussians terms in the objective and leads to more frequent misallocations of unannotated samples. We therefore recommend using the formulation without priors in these circumstances or even using simple constraints when the experts' annotations are perfect. In the other circumstances, the model with priors generally performs better.

Figure 3.3 presents the average percentage of correct annotations according to the partitions obtained with the two models. When we incorporate the prior beliefs, this quantity becomes close to the real number of correct annotations for $K = 4$ and $K = 6$. Conversely, the model without priors requires more information to approximate the real number of mistakes even when the experts' accuracy is high ($p = 0.9$ and $p = 1.0$). This behavior stems from the fact that ordinary SBMs can recover any connectivity pattern, which may be an issue in sparse graphs with little structure [31].

Finally, Figures 3.4–3.6 compare the CI obtained with the two proposed models and the unsupervised model for $K = 6$ and different values of m. In Figure 3.4, for $p = 0.8$, no significant difference appears between the three models, although the semi-supervised models present more datasets with CI = 0 (same ground-truth structure) and CI = 1 (one center diverging from the ground-truth center locations). Figure 3.5 compares the CI when $p = 0.9$. For 200 annotations, 37 out of 50 datasets have CI = 0 or CI = 1 without prior information, whereas 36 cases are reported with priors. The unsupervised model, however, presents only 29 datasets with CI = 0 or 1. Finally, Figure 3.6 shows the CI distribution with perfect annotation accuracy. In this case, differences between the two proposed models are more significant, notably when more information is provided.

3.5.3 Performance on Real-World Benchmarks

This section considers datasets that are assumed not to be generated by spherical Gaussian distributions. The goal is to show whether the introduction of pairwise information leads to partitions with a different structure from those obtained with unsupervised clustering in challenging datasets that do not fit the original assumptions of the model. For this analysis, we consider eight real datasets from the UCI machine learning repository [24] with continuous multi-feature data and available ground-truth information. Table 3.4 summarizes these datasets in terms of size and number of clusters.

Dataset	N	D	K
Diabetes	145	5	3
Iris	150	4	3
Wine	178	13	3
Thyroid	215	5	3
Vertebral	310	6	3
E. coli	336	7	8
Breast-Cancer	683	9	2
Pendigits-389	2157	16	3

Table 3.4: UCI datasets.

Figures 3.7 to 3.9 show the performance of the two models in terms of NMI for $p \in \{0.8,\ 0.9,\ 1.0\}$ and $m \in \{N/2,\ N,\ 1.5N\}$. For each combination of a dataset and values of p and m, we generate ten different graphs. We run 50 repetitions of Algorithm 2 on each case and register the NMI for the solution with the best log-likelihood. Then, we measure the difference of (i.e., relative) NMI between each of the proposed semi-supervised models and the baseline model without supervision. We represent those values as boxplots, in which the whiskers extend to 1.5 times the interquartile range.

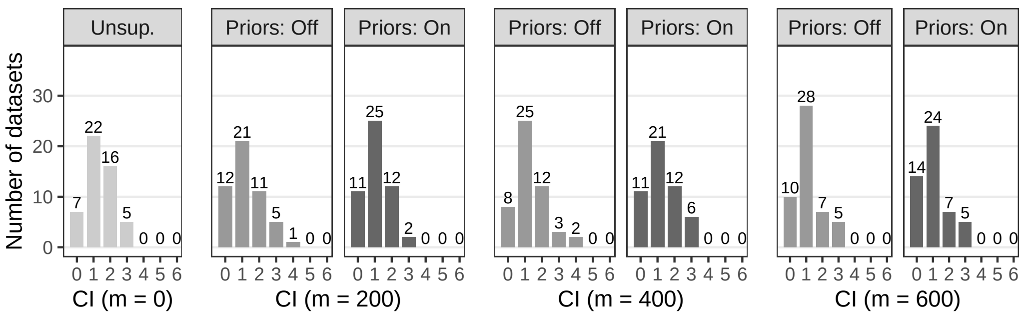


Figure 3.4: CI of 50 Gaussian mixtures for $m = \{0, 200, 400, 600\}$, and $p = 0.8$.

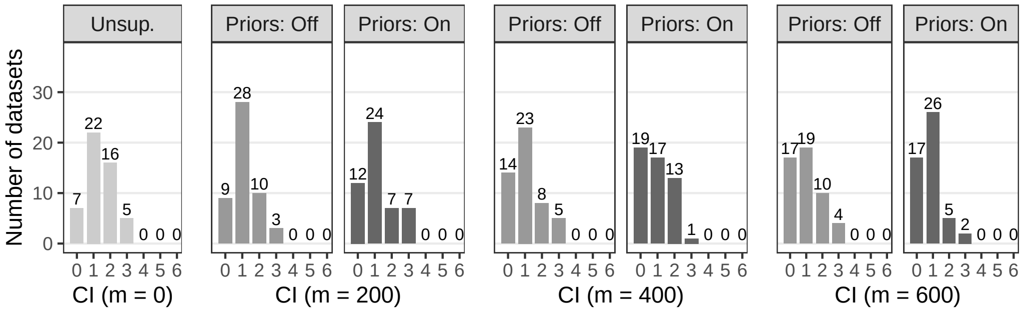


Figure 3.5: CI of 50 Gaussian mixtures for $m = \{0, 200, 400, 600\}$, and $p = 0.9$.

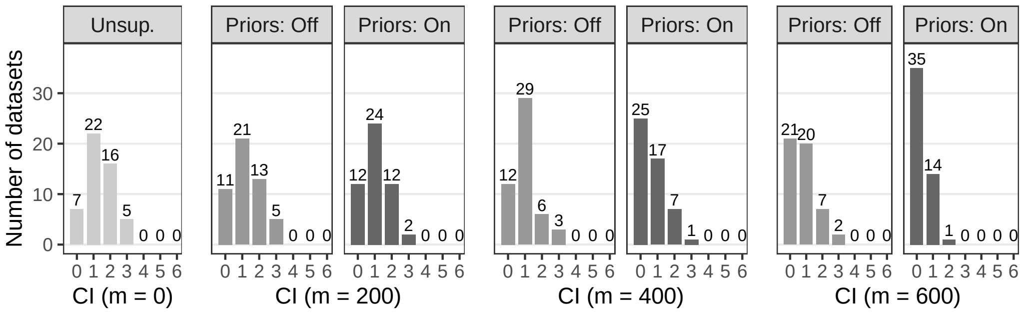


Figure 3.6: CI of 50 Gaussian mixtures for $m = \{0, 200, 400, 600\}$, and $p = 1.0$.

Figure 3.7 presents the relative NMI for $p = 0.8$, i.e., considering annotations that are quite inaccurate. In at least two out of eight datasets, the NMI improves when pairwise information is considered. For the remaining

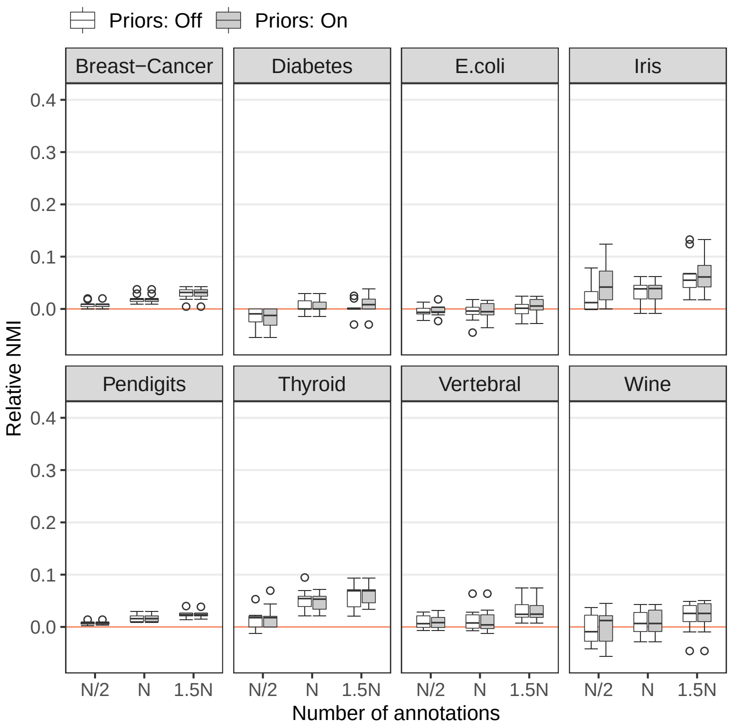


Figure 3.7: Relative NMI between the proposed models and the unsupervised model in UCI datasets ($p = 0.8$).

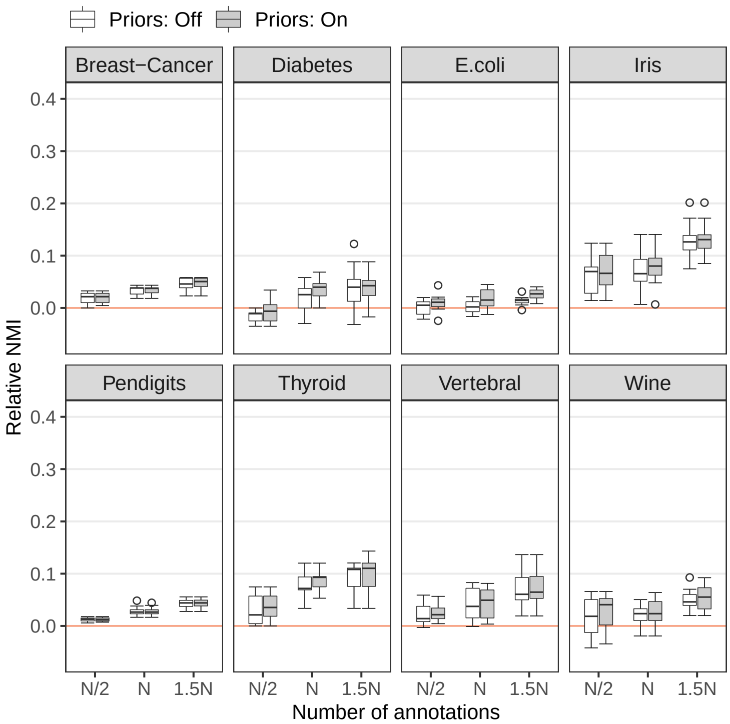


Figure 3.8: Relative NMI between the proposed models and the unsupervised model in UCI datasets ($p = 0.9$).

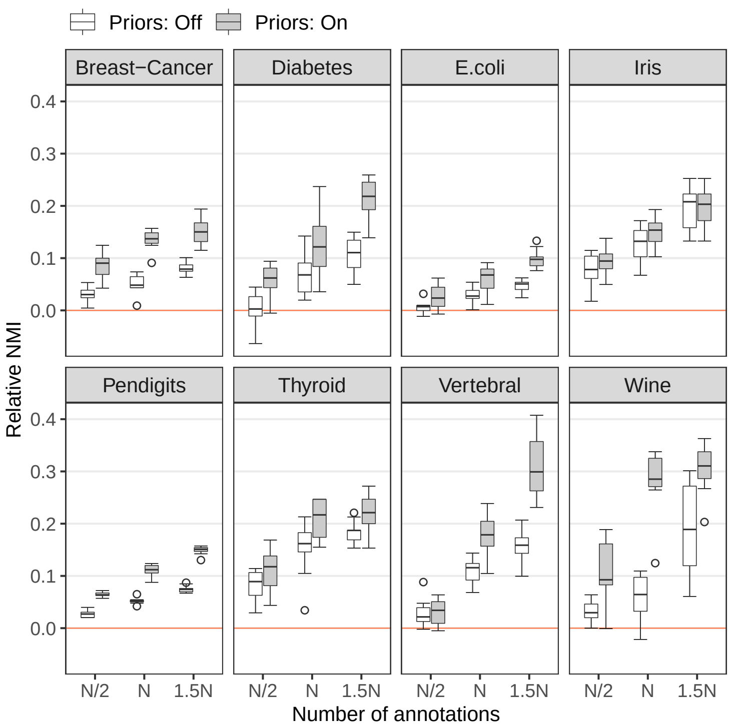


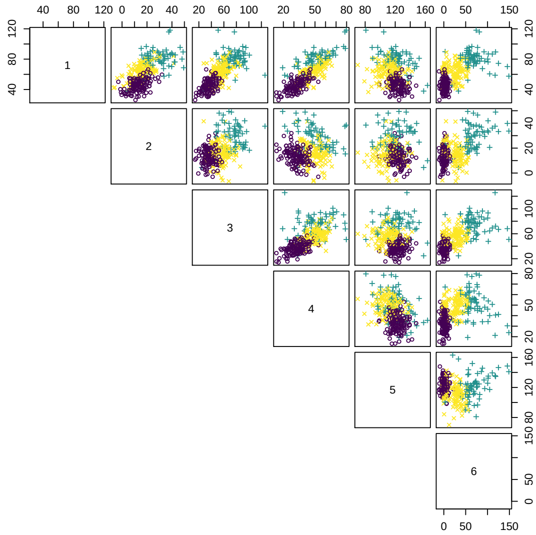
Figure 3.9: Relative NMI between the proposed models and the unsupervised model in UCI datasets ($p = 1.0$).

datasets, no significant improvement occurs. Additionally, we did not observe significant differences between the two proposed models (with or without priors) for $p = 0.8$. In general, prior knowledge led to the same solutions that are obtained without priors.

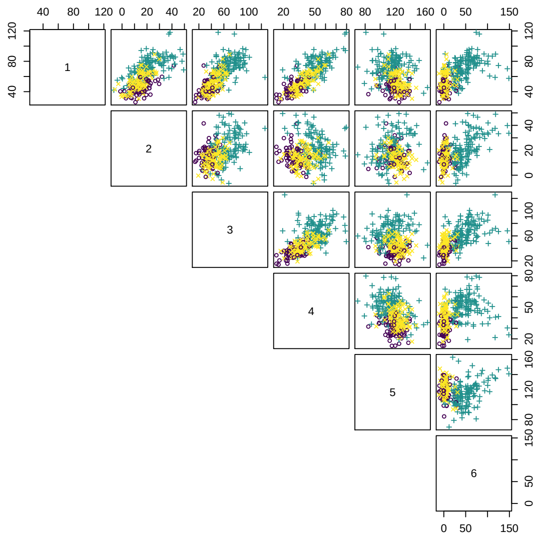
Figure 3.8 reports the relative NMI for graphs with only 10% annotation errors ($p = 0.9$). In these conditions, the NMI obtained in the datasets *Diabetes*, *E. coli*, and *Wine* also increase visibly upon adding pairwise information. Although the two proposed models perform quite similarly, in all test instances, the model with priors performs the same or better than the model without priors.

In Figure 3.9, with perfect annotations, the median relative NMI is positive in all datasets and for all values of m. We consider again $p = 1 - 10^{-6}$ for the priors estimation. A notable difference now appears between the two proposed models. The most expressive difference appears for $m = 1.5N$ (average NMI of 0.8828 with prior information versus 0.8074 without priors). The results reveal that, given trusted supervision, attaching prior beliefs significantly boosts performance even if the available supervision is limited and the datasets do not fit the original assumptions.

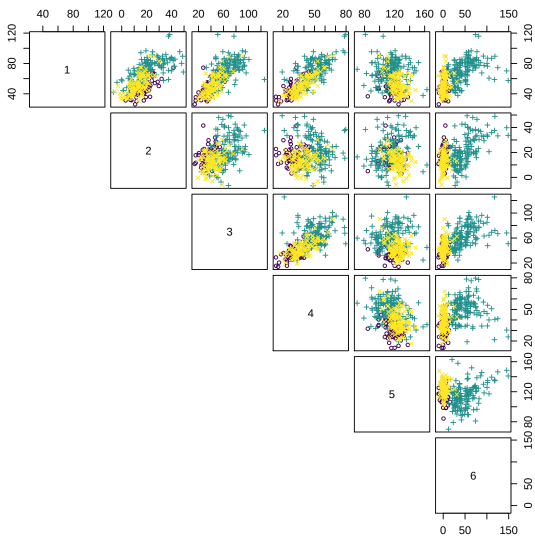
Vertebral column dataset. Figure 3.10 presents the solutions obtained for the *Vertebral* dataset [24] with unsupervised clustering and the semi-supervised model with priors, along with the ground-truth solution. The *Vertebral* dataset contains six biomechanical measures used to classify orthopedic patients into three classes: normal, disk hernia, and *spondilolysthesis*. Each diagonal in the figure represents one feature of the dataset, and each upper square presents the feature values in pairs. We consider $m = N = 310$ pairwise annotations with no errors. In these conditions, the unsupervised model obtains CI = 1, whereas the use of pairwise annotations leads to a CI of zero. Without side information, unsupervised clustering naturally tends to retrieve partitions with separable clusters because it relies only on the available features. The introduction of pairwise information can reveal hidden structures, especially if some clusters significantly overlap, which is the case for the *Vertebral* dataset. The yellow ("$\times$" crosses) and purple (circles) clusters considerably overlap, and the unsupervised formulation does not capture this characteristic. Adding pairwise annotations can guide the clustering process toward solutions that differ structurally from those obtained with an unsupervised model. Beyond repairing the membership of annotated samples originally misallocated by unsupervised methods, the semi-supervision also reveals partitions with markedly distinct structures.



(a) Unsupervised clustering



(b) Pairwise supervision ($m = N$)



(c) Ground-truth solution

Figure 3.10: Different clustering structures found in the *Vertebral* dataset.

3.6 Concluding Remarks

Side information in the form of pairwise annotations can be a powerful tool to improve clustering performance. In this chapter, we used SBMs to model *must-link* and *cannot-link* annotations and integrated them into the minimum sum-of-squares clustering model. We provided efficient learning algorithms and demonstrated that incorporating pairwise information can significantly improve clustering performance, even if the annotations are provided in a small volume and with mistakes. Moreover, for both synthetic and real-world datasets, we have shown that prior knowledge of annotation accuracy can be harnessed to further improve clustering. In challenging cases in which groups overlap substantially and the observed data do not fit the model's general premises, the adoption of pairwise information can be decisive to reveal hidden structures.

This work provides a font of promising research perspectives. Firstly, we suggest exploring other algorithmic approaches for the proposed models along with specific applications such as facial image recognition and video object classification, as discussed in Basu et al. [9]. Moreover, we suggest other methodological extensions that naturally fit in a semi-supervised framework. Further improvements could also be achieved using active learning to select samples for annotation, or through label propagation techniques [9, 70, 72]. Finally, this model could be further generalized to consider experts with different accuracy levels and support other probability distributions, increasing its generality and flexibility.

4
Semi-supervised Learning with Untrustworthy Labelers

4.1
Context and Background

The presence of noisy labels is usually a challenge faced in supervised learning. Training data frequently contain errors in their annotations, typically caused by human mistakes in translating information [50]. According to the survey of Frénay and Verleysen [26], the sources of label noise are various. It can result from the insufficient amount of information provided to the annotator about the data sample or the lack of knowledge the annotator has. Since obtaining accurate labels is often expensive and time-consuming, asking non-experts to provide annotations is very common. For example, annotations obtained in a "crowdsourcing" manner made it possible to quickly increase the size of annotated datasets, with the drawback of introducing noisy information. In addition, the degree of subjectivity of classes in some tasks and data encoding and communication problems are also frequent sources of label noise.

As exposed by Frénay and Verleysen [26], the consequences of label noise are numerous. It may decrease classification performance, require the increase of model complexity, or distort the proportion of observed classes. Thus, different strategies have been proposed to handle noisy labels (see, for example, [5, 12, 21, 22, 49, 54, 69]). In general, two main approaches are used to treat incorrect annotations [26]. First, some techniques directly encode noisy labels and treat such inaccuracies during training. A second approach verifies the correctness of the labels through filtering techniques. In this case, incorrect labels are identified and re-labeled (or removed) before training. These two main approaches have given rise to a relatively recent discussion in machine learning communities, in which two views for this problem emerged: a model-centric and a data-centric perspective.

In Chapter 3, we showed that different levels of inaccuracies lead to different values of NMI, assuming the same learning model. With incorrect annotations, we normally need more information to achieve the same performance seen with totally accurate data. From a data-centric perspective, there is a trade-off between the accuracy and the amount of supervision provided.

Once we repair the incorrect labels, we can use whatever model we desire. On the other hand, a model-centric approach handles the annotations inaccuracies within a mathematical formulation describing our knowledge about the problem, keeping the data we observe as it is.

In this chapter, we propose an approach to handle label noise from a model-based perspective. In some situations, we can explore particular structures from the observed labels, especially when groups of annotators deliberately provide incorrect annotations. This scenario usually connects to real problems where annotators engage in polarized situations, such as classifying content as negative or positive and even as reliable or fake. However, these scenarios regularly extrapolate the binary case, and more than two classes may appear. When the untrustworthy annotations have some structure, for example, due to the annotators' political preference or motivation, we can take advantage of these patterns and use this knowledge to identify groups of untrustworthy labelers. We do this in a semi-supervised manner, in which we use the provided labels as side information within a clustering model. After fitting the model, we obtain a partition of the samples, which can be further mapped to classes.

4.2 Proposed Model

We now present the semi-supervised clustering model in which we estimate the probability that a group of annotators change the label of a sample based on its class. We assume Gaussian-distributed data samples along with labels provided by L groups of annotators. The considered generative model is the following:

- For each $i \in \{1, \ldots, N\}$:
  - Pick a Gaussian component $r \in \{1, \ldots, K\}$ with probability $1/K$, and set $\hat{y}_i = r$ as the ground-truth;
  - Generate a D-dimensional sample $\boldsymbol{x}_i$ from component r:

$$\boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{\mu}_r, \sigma_r^2) \tag{4-1}$$

  - Ask an expert of group $j \in \{1, \ldots, L\}$ to label sample $\boldsymbol{x}_i$:

$$y_i \sim \text{Multinomial}(\theta_{j\hat{y}_i}), \tag{4-2}$$

  where $\theta_{jr} = \{\theta_{jr1}, \ldots, \theta_{jrK}\}$ is the vector encoding the labelling behaviour of annotators in group j. Here, θ_{jrs} represents the probability that an annotator belonging to group j labels a sample belonging to class r as a sample of class s, and hence $\sum_s \theta_{jrs} = 1$ for each pair of group j and class r.

We consider the binary indicator g_{ij}, with $g_{ij} = 1$ if sample $\boldsymbol{x}_i$ is labelled by an expert of group j, and $g_{ij} = 0$ otherwise. Therefore, the multinomial distribution of Equation (4-2) can be expressed as the probability a sample i is labelled as class s:

$$P(y_i = s) = \prod_{j=1}^{L} \theta_{j\hat{y}_i s}^{g_{ij}}. \tag{4-3}$$

We now present the set of model parameters and data used in the presented annotation setting:

- N: Number of data samples;
- L: Number of groups of annotators;
- K: Number of classes;
- g_{ij}: Binary indicator stating that sample i was annotated by group j (given information);
- y_{is}: Binary indicator stating that sample i was labelled as class s (given information);
- n_j: Number of annotations given by group j (given information);
- z_{ir}: Binary class indicator stating that sample i is assigned to class r (model parameter);
- θ_{jrs}: Probability that an annotator belonging to group j labels a sample belonging to class r as a sample of class s (model parameter).

The likelihood of observing a set of annotations $\boldsymbol{Y}$, given probabilities $\boldsymbol{\Theta}$ and an assignment choice $\boldsymbol{Z}$ is therefore

$$P(\boldsymbol{Y}|\boldsymbol{\Theta}, \boldsymbol{Z}) = \prod_{i}^{N}\prod_{j}^{L}\prod_{r}^{K}\prod_{s}^{K} \theta_{jrs}^{g_{ij} z_{ir} y_{is}} = \prod_{j}^{L}\prod_{r}^{K}\prod_{s}^{K} \theta_{jrs}^{m_{jrs}}, \tag{4-4}$$

where $m_{jrs} = \sum_i^N z_{ir} y_{is} g_{ij}$ is the number of samples assigned to class r, which were labelled as class s by an annotator of group j. As θ_{jrs} represents probabilities, $\sum_s \theta_{jrs} = 1$. Thus, the log-likelihood is:

$$\log P(\boldsymbol{Y}|\boldsymbol{\Theta}, \boldsymbol{Z}) = \sum_{j}^{L}\sum_{r}^{K}\sum_{s}^{K} m_{jrs} \log(\theta_{jrs}) \tag{4-5a}$$

$$\text{s.t.} : \sum_{s} \theta_{jrs} = 1, \quad \forall jr. \tag{4-5b}$$

Adding the Lagrangian variables $\boldsymbol{\lambda}$ leads to the following unconstrained formulation

$$\begin{aligned}\log P(\mathbf{Y}|\boldsymbol{\Theta}, \boldsymbol{Z}, \boldsymbol{\Lambda}) &= \sum_{j}^{L}\sum_{r}^{K}\sum_{s}^{K} m_{jrs}\log(\theta_{jrs}) + \sum_{j}^{L}\sum_{r}^{K}\lambda_{jr}\left(\sum_{s}^{K}\theta_{jrs} - 1\right) \\ &= \sum_{j}^{L}\sum_{r}^{K}\sum_{s}^{K} m_{jrs}\log(\theta_{jrs}) + \lambda_{jr}\theta_{jrs} - \sum_{j}^{L}\sum_{r}^{K}\lambda_{jr},\end{aligned} \tag{4-6}$$

for which we can find the maximum value of θ_{jrs} by derivation:

$$\hat{\theta}_{jrs} = -\frac{m_{jrs}}{\lambda_{jr}}. \tag{4-7}$$

From Equations (4-5b) and (4-7), we can estimate λ_{jr}:

$$\lambda_{jr} = -\sum_{s}^{K} m_{jrs}. \tag{4-8}$$

And from Equations (4-7) and (4-8), we re-write the log-likelihood:

$$\begin{aligned}\log(\boldsymbol{Y}|\boldsymbol{Z}) &= \sum_{j}^{L}\sum_{r}^{K}\sum_{s}^{K} m_{jrs}\log\left(\frac{m_{jrs}}{\sum_s m_{jrs}}\right) - m_{jrs} + \sum_{j}^{L}\sum_{r}^{K}\sum_{s}^{K} m_{jrs} \\ &= \sum_{j}^{L}\sum_{r}^{K}\sum_{s}^{K} m_{jrs}\log\left(\frac{m_{jrs}}{\sum_s m_{jrs}}\right).\end{aligned} \tag{4-9}$$

Finally, coupling the mixture of Gaussians formulation with model (4-9), we obtain the following log-likelihood after a few arrangements

$$\begin{aligned}\mathcal{L}(\boldsymbol{X}, \boldsymbol{Y}|\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\Theta}, \boldsymbol{Z}) \propto& -\frac{1}{2}\sum_{i}^{N}\sum_{r}^{K}\left(\frac{\|\boldsymbol{x}_i - \boldsymbol{\mu}_r\|^2}{\sigma_r^2} + 2D\log(\sigma_r)\right) z_{ir} \\ &+ \sum_{i}^{N}\sum_{j}^{L}\sum_{r}^{K}\sum_{s}^{K}\log(\theta_{jrs}) g_{ij} y_{is} z_{ir},\end{aligned} \tag{4-10}$$

where variables $\boldsymbol{\mu}_r$ and σ_r are obtained from Equations (3-4) and (3-5), respectively, and θ_{jrs} is obtained from Equation (4-7). Alternatively, we can write this log-likelihood as

$$\mathcal{L}(\boldsymbol{X}, \boldsymbol{Y}|\boldsymbol{Z}) = \log P(\boldsymbol{X}|\boldsymbol{Z}) + \log P(\boldsymbol{Y}|\boldsymbol{Z}), \tag{4-11}$$

where $\log P(\boldsymbol{X}|\boldsymbol{Z})$ is obtained from Equation (3-6), and $\log P(\boldsymbol{Y}|\boldsymbol{Z})$ is obtained from Equation (4-9).

4.3 Methodology and Experiments

We adapted the hybrid genetic search described in Algorithm 2 to solve model (4-10). Instead of updating the SBM parameters in Algorithm 3, we use Equations (4-7) and (4-8) to update the parameter $\boldsymbol{\Theta}$, which represents the annotations probabilities given by groups of annotators. Algorithms 5 and 6 describe the general structure of the proposed method.

Algorithm 5 Hybrid-Genetic Search: Untrustworthy Labelers

Input: Feature data: $\boldsymbol{X}$, Labels: $\boldsymbol{Y}$, Matrix indicating which group labeled each sample: $\boldsymbol{G}$, Number of clusters: K, Number of groups of annotators: L, Parameters: Π_1 and Π_2
$\mathcal{S} \leftarrow$ Set with Π_1 initial solutions
repeat
  $\boldsymbol{Z}^{(1)}, \boldsymbol{Z}^{(2)} \leftarrow$ Random solutions from $\mathcal{S}$
  $\boldsymbol{Z} \leftarrow$ Crossover($\boldsymbol{Z}^{(1)}$, $\boldsymbol{Z}^{(2)}$)
  $\boldsymbol{Z'} \leftarrow$ Mutation($\boldsymbol{Z}$)
  Algorithm 6: FitSamples($\boldsymbol{X}$, $\boldsymbol{Y}$, $\boldsymbol{Z}'$)
  Add solution $\boldsymbol{Z'}$ to $\mathcal{S}$
  if $|\mathcal{S}| = \Pi_2$ **then**
    $\mathcal{S} \leftarrow$ Select the best Π_1 solutions
  end if
until Maximum number of iterations is attained
return Best solution $\boldsymbol{Z}^*$ found

Algorithm 6 FitSamples: Untrustworthy Labelers

Input: Feature data: $\boldsymbol{X}$, Labels: $\boldsymbol{Y}$, Current solution: $\boldsymbol{Z}$
Find parameters $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$, $\boldsymbol{\Theta}$ maximizing $\mathcal{L}(\boldsymbol{X}, \boldsymbol{Y}|\boldsymbol{Z})$
Evaluate log-likelihood with the estimated parameters:
$\mathcal{Q} \leftarrow \mathcal{L}(\boldsymbol{X}, \boldsymbol{Y}|\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\Theta}, \boldsymbol{Z})$
repeat
  for $i \in \boldsymbol{X}$ and $r \in \{1, \ldots, K\}$ in random order **do**
    Consider solution $\boldsymbol{Z}^{\mathrm{R}}$ obtained from $\boldsymbol{Z}$ by relocating sample i to cluster r
    Find parameters $\hat{\boldsymbol{\mu}}$, $\hat{\boldsymbol{\sigma}}$, $\hat{\boldsymbol{\Theta}}$ maximizing $\mathcal{L}(\boldsymbol{X}, \boldsymbol{Y}|\boldsymbol{Z}^{\mathrm{R}})$
    Evaluate log-likelihood with the estimated parameters:
    $\mathcal{Q}' \leftarrow \mathcal{L}(\boldsymbol{X}, \boldsymbol{Y}|\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{\Theta}}, \boldsymbol{Z}^{\mathrm{R}})$
    if $\mathcal{Q}' > \mathcal{Q}$ **then**
      Apply: $\boldsymbol{\mu} \leftarrow \hat{\boldsymbol{\mu}}$, $\boldsymbol{\sigma} \leftarrow \hat{\boldsymbol{\sigma}}$, $\boldsymbol{\Theta} \leftarrow \hat{\boldsymbol{\Theta}}$, $\boldsymbol{Z} \leftarrow \boldsymbol{Z}^{\mathrm{R}}$, $\mathcal{Q} \leftarrow \mathcal{Q}'$
    end if
  end for
until No improving relocation has been identified

We conducted computational experiments to analyze how different amounts of untrustworthy annotations affect clustering performance in terms of NMI. We compare the performance of the proposed semi-supervised model under different sets of annotations with unsupervised clustering. Our computational experiments consider datasets with $K = 3$ classes and $L = 3$ groups of annotators. We analyze the performance of the proposed model in synthetic datasets and the *Vertebral* UCI benchmark. In the following sections, we describe the parameters used to generate the synthetic data and the scenarios we consider for the groups of annotators. The algorithm was implemented in Julia (version 1.0.5).

Data generation. We generate synthetic datasets with $N = 500$ samples as in Section 3.5.2, in which we consider overlapping Gaussian distributions with each cluster having its own dispersion. More precisely, each class r has a D-dimensional mean $\boldsymbol{\mu}_r$ that is sampled uniformly over the range $[-2, 2]$. For the variance of each class, we sample σ_r^2 uniformly from the range $[0, 5]$. In the sequence, each data sample is generated with probability $1/K$ from group r according to the Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_r, \sigma_r^2)$.

Labels generation. Our experimental setting considers $L = 3$ groups of annotators. Besides, we assume a scenario with one group of reliable annotators and two untrustworthy groups. We consider that the group represented by $j = 1$ is trustworthy, without loss of generality. On the other hand, the remaining two groups ($j = 2$ and $j = 3$) correspond to untrustworthy groups. We remind that the parameter θ_{jrs} represents the probability an annotator in group j labels a sample belonging to class r as a sample of class s. In our generative process, we consider the ground-truth matrices $\hat{\Theta}_1$, $\hat{\Theta}_2$ and $\hat{\Theta}_3$, which generate the labels according to the following pattern:

$$\hat{\Theta}_1 = \begin{pmatrix} p & q & q \\ q & p & q \\ q & q & p \end{pmatrix}, \quad \hat{\Theta}_2 = \begin{pmatrix} q & q & p \\ q & p & q \\ p & q & q \end{pmatrix}, \quad \hat{\Theta}_3 = \begin{pmatrix} q & p & q \\ p & q & q \\ q & q & p \end{pmatrix},$$

where $p \in [0.5, 1.0]$ represents the annotators' accuracy, and $q = (1 - p)/2$. Thus, p and q represent the classes transitions. The entries $p \geq 0.5$ in the diagonal of $\hat{\Theta}_1$ characterize the trustworthy group with accuracy p. On the other hand, $\hat{\Theta}_2$ represents an untrustworthy group that normally provides correct annotations for samples of class 2 but deliberately changes the labels of the other classes. Similarly, $\hat{\Theta}_3$ represents the behavior of a group that generally provides correct annotations for samples of class 3. It is worth noticing that the untrustworthy groups present an annotation pattern when they deliberately change a label, which is expressed by the value of q. In the specified setting, the untrustworthy group is less likely to change the label of a sample to the class it normally provides correct annotations. Finally, we consider that the number of annotations n_j given by each group is pre-defined. Then, for each $j \in \{1, \ldots, L\}$, we repeatedly request an annotator belonging to the group to label a random data sample that was not labeled yet, until reaching n_j. We consider that all samples are labeled once, and thus $\sum_{j=1}^{L} n_j = N$.

4.3.1 Clustering Performance

In our experimental set-up, we analyze the impact of untrustworthy annotations by considering different values for the size n_1 of the trustworthy group. We consider equally-sized untrustworthy groups, with $n_2 = n_3 = (N - n_1)/2$. In addition, we consider $p = 0.9$, which means that the trustworthy group is 90% accurate for every class. At the same time, for $j \in \{2, 3\}$, the j-th group is 90% accurate in labeling samples of class j. However, these groups have a 90% probability of changing the label of a sample that belongs to a class other than j. For analyzing the proposed model, we consider four different annotations sets:

- $\boldsymbol{S}^{(\varnothing)}$: set with no annotation;
- $\boldsymbol{S}^{(\mathrm{T})}$: set containing only the annotations of the trustworthy group 1;
- $\boldsymbol{S}^{(\mathrm{U})}$: set containing only the annotations of the untrustworthy groups 2 and 3;
- $\boldsymbol{S}^{(\mathrm{ALL})}$: set with all annotations, given by the three groups.

Therefore, $|\boldsymbol{S}^{(\varnothing)}| = 0$, $|\boldsymbol{S}^{(\mathrm{T})}| = n_1$, $|\boldsymbol{S}^{(\mathrm{U})}| = (N - n_1)/2$, and $|\boldsymbol{S}^{(\mathrm{ALL})}| = N$.

Table 4.1 reports the average NMI on 50 synthetic datasets as a function of the size of the trustworthy group. For small values of n_1, we observe that the solutions obtained with the annotation set $\boldsymbol{S}^{(\mathrm{U})}$ present a better NMI than the ones obtained with $\boldsymbol{S}^{(\mathrm{T})}$, due to the abundance of annotations given by the untrustworthy groups. As expected, when n_1 increases, we see that using the annotations given by the trustworthy group is beneficial. In general, we observe that even when we consider only the set of untrustworthy annotators, we can still explore the patterns provided by them and leverage clustering performance. Therefore, we highlight considering such groups, even if they are predominantly untrustworthy. Moreover, we usually do not know a priori which groups are trustworthy in practical situations.

In Table 4.2, we present the NMI obtained in the *Vertebral* UCI dataset [24], which has three classes. We report the average NMI obtained with ten independent runs of Algorithm 5. We again present the performance as a function of the size of the trustworthy group for different annotations sets, with $p = 0.9$ and equally-sized untrustworthy groups. We use the same scheme of annotators as in the results previously presented for the synthetic datasets. We observe that for small values of n_1, the proposed model obtains NMI similar to that attained with no supervision, when considering the set $\boldsymbol{S}^{(\mathrm{T})}$. When a sufficient amount of untrustworthy annotations is provided, we observe a significant increase in NMI, especially for $n_2 + n_3 = 0.9N$. In addition, it is worth considering both trustworthy and untrustworthy groups of annotators.

Size of n_1	Size of $n_2 + n_3$	Average NMI			
		$\boldsymbol{S}^{(\varnothing)}$	$\boldsymbol{S}^{(\mathrm{T})}$	$\boldsymbol{S}^{(\mathrm{U})}$	$\boldsymbol{S}^{(\mathrm{ALL})}$
$0.1N$	$0.9N$	0.7533	0.7631	0.8538	0.8683
$0.2N$	$0.8N$		0.7726	0.8401	0.8661
$0.3N$	$0.7N$		0.7815	0.8311	0.8657
$0.4N$	$0.6N$		0.7865	0.8264	0.8686
$0.5N$	$0.5N$		0.7988	0.8137	0.8710

Table 4.1: Synthetic datasets: Average NMI as a function of the size of the trustworthy group for different annotations sets.

Size of n_1	Size of $n_2 + n_3$	Average NMI			
		$\boldsymbol{S}^{(\varnothing)}$	$\boldsymbol{S}^{(\mathrm{T})}$	$\boldsymbol{S}^{(\mathrm{U})}$	$\boldsymbol{S}^{(\mathrm{ALL})}$
$0.1N$	$0.9N$	0.4146	0.4180	0.4906	0.5008
$0.2N$	$0.8N$		0.4218	0.4677	0.4905
$0.3N$	$0.7N$		0.4258	0.4610	0.5029
$0.4N$	$0.6N$		0.4357	0.4484	0.4905
$0.5N$	$0.5N$		0.4306	0.4369	0.5002

Table 4.2: *Vertebral* dataset: NMI obtained as a function of the size of the trustworthy group for different annotations sets.

4.4 Concluding Remarks

The presence of incorrect labels originated from untrustworthy sources can significantly hurt the quality of predictions. Recent discussions in the machine learning community introduce data-driven and model-driven approaches to handle noisy annotations. One alternative consists of repairing incorrect labels and then using existing models for prediction. Incorrect labeling, however, follows some patterns in certain situations. For example, in polarized scenarios, an annotator may have a motivation (for example, political) to change the label of observations belonging to particular classes. In these cases, one approach consists of modeling the behavior of annotator groups and then deriving appropriate classes.

In this chapter, we studied the scenario in which groups of untrustworthy annotators provide data labels and focused on techniques for clustering in the presence of such annotations. We proposed a model to estimate the probabilities that a group of annotators has on switching the label of a given observation to a different class. We adopted a maximum-likelihood approach to handle untrustworthy annotations in a semi-supervised manner. The proposed model, however, is limited to finding a clustering solution for the data samples. A natural extension for this work consists of deriving a set of new labels from the resulting partition. This way, we could further use the derived labels for training a classifier. However, a clustering analysis can still help extract underlying classes before the training phase of a supervised model. Finally, we recommend further investigation of the proposed model in practical applications with different numbers of classes.

5
Conclusions and Perspectives

Clustering is a systematic way of identifying meaningful groups in data through numerical methods. The adoption of model-based approaches for data clustering has recently received increasing attention and has allowed the modeling of complex patterns and relations. Especially when noisy and inaccurate information is present in data, the use of principled probabilistic models has numerous advantages. They highlight the necessary premises and point out when and why a method works properly or inadequately [13].

In this research, we investigated model-based approaches for clustering-related problems. More precisely, we introduced novel models along with tailored solution methods for community detection and semi-supervised learning. The general scheme that we adopted in this work considers expressing the generative model responsible for generating the data and adopting maximum-likelihood estimation methods. In addition, we explored the use of prior user knowledge to enhance clustering performance via the introduction of parameters constraints and the incorporation of prior distributions. We revealed that such prior knowledge is fundamental to achieving accurate solutions.

The study of assortative networks, or networks with modular structures, is present in many domains, and we proposed extending the widely used degree-corrected SBM to account for assortative solutions. Moreover, we demonstrated that a constrained optimization approach could guide algorithms to find assortative structures. We then progressed towards the study of a semi-supervised setting in which supervision in the form of *must-link* and *cannot-link* annotations is provided by domain experts. We used SBMs to model such supervision, considering the presence of inaccurate annotations. We demonstrated that coupling a mixture of Gaussians with pairwise annotations significantly impacts clustering performance even when supervision is scarce and inaccurate. Finally, we examined the problem of learning in the presence of class labels given by untrustworthy groups of annotators. We introduced a semi-supervised model for handling such incorrect labels, which depends on the data features and the patterns of the given annotations.

This research presents a range of possibilities for future work and perspectives. First, we suggest exploring different algorithms for the proposed models. With the advent of large datasets, we should give special attention to scalability. The current bottleneck of the proposed methods in this work is the local search procedure, which may be prohibitive for large datasets. Therefore, there is room for improving computational efficiency by avoiding unpromising solutions and for developing alternative optimizers. Second, we recommend harnessing the flexibility of the proposed models to derive tailored formulations, which can meet domain requirements not covered by the general models. For example, we can consider different probability distributions to describe the data features in our semi-supervised model and extend the assortativity rules for community detection. These customized extensions would consequently increase the range of possible applications. Finally, we recommend expanding the experimental analysis and further investigating the practical implications of considering untrustworthy annotations. Moreover, we suggest extending the proposed methodology to derive a new set of adjusted labels and assess the impact of labeling repair in classification problems.

Bibliography

[1] ABBE, E. (2017). **Community detection and stochastic block models: Recent developments**. *The Journal of Machine Learning Research*, 18(1):6446–6531.

[2] AIROLDI, E. M., BLEI, D. M., FIENBERG, S. E., AND XING, E. P. (2008). **Mixed membership stochastic blockmodels**. *Journal of Machine Learning Research*, 9(Sep):1981–2014.

[3] AMINI, A. A., LEVINA, E., ET AL. (2018). **On semidefinite relaxations for the block model**. *The Annals of Statistics*, 46(1):149–179.

[4] ANDERBERG, M. R. (2014). **Cluster analysis for applications: probability and mathematical statistics: a series of monographs and textbooks**, volume 19. Academic press.

[5] ARAZO, E., ORTEGO, D., ALBERT, P., O'CONNOR, N., AND MCGUINNESS, K. (2019). **Unsupervised label noise modeling and loss correction**. In *International Conference on Machine Learning*, pages 312–321. PMLR.

[6] ARZENO, N. M. AND VIKALO, H. (2015). **Semi-Supervised Affinity Propagation with Soft Instance-Level Constraints**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(5):1041–1052.

[7] BAI, L., LIANG, J., AND CAO, F. (2020). **Semi-supervised clustering with constraints of different types from multiple information sources**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

[8] BASU, S., BANERJEE, A., AND MOONEY, R. J. (2004). **Active semi-supervision for pairwise constrained clustering**. In *Proceedings of the 2004 SIAM international conference on data mining*, pages 333–344. SIAM.

[9] BASU, S., DAVIDSON, I., AND WAGSTAFF, K. (2008). **Constrained clustering: Advances in algorithms, theory, and applications**. CRC Press.

[10] BILENKO, M., BASU, S., AND MOONEY, R. J. (2004). **Integrating constraints and metric learning in semi-supervised clustering**. In

Proceedings of the twenty-first international conference on Machine learning, page 11. ACM.

[11] BISHOP, C. M. (2006). **Pattern recognition and machine learning**. springer.

[12] BOOTKRAJANG, J. AND KABÁN, A. (2012). **Label-noise robust logistic regression and its applications**. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 143–158. Springer.

[13] BOUVEYRON, C., CELEUX, G., MURPHY, T. B., AND RAFTERY, A. E. (2019). **Model-based clustering and classification for data science: with applications in R**, volume 50. Cambridge University Press.

[14] CAI, T. T., LI, X., ET AL. (2015). **Robust and computationally feasible community detection in the presence of arbitrary outlier nodes**. *The Annals of Statistics*, 43(3):1027–1059.

[15] CHEN, Y., SANGHAVI, S., AND XU, H. (2012). **Clustering sparse graphs**. In *Advances in Neural Information Processing Systems*, pages 2204–2212.

[16] CHEN, Z. J., HE, Y., ROSA-NETO, P., GERMANN, J., AND EVANS, A. C. (2008). **Revealing modular architecture of human brain structural networks by using cortical thickness from MRI**. *Cerebral Cortex*, 18(10):2374–2381.

[17] CONTISCIANI, M., POWER, E., AND DE BACCO, C. (2020). **Community detection with node attributes in multilayer networks**. *arXiv preprint arXiv:2004.09160*.

[18] DAUDIN, J.-J., PICARD, F., AND ROBIN, S. (2008). **A mixture model for random graphs**. *Statistics and Computing*, 18(2):173–183.

[19] DECELLE, A., KRZAKALA, F., MOORE, C., AND ZDEBOROVÁ, L. (2011a). **Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications**. *Physical Review E*, 84(6):066106.

[20] DECELLE, A., KRZAKALA, F., MOORE, C., AND ZDEBOROVÁ, L. (2011b). **Inference and phase transitions in the detection of modules in sparse networks**. *Physical Review Letters*, 107(6):065701.

[21] DGANI, Y., GREENSPAN, H., AND GOLDBERGER, J. (2018). **Training a neural network based on unreliable human annotation of medical images**. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 39–42. IEEE.

[22] DING, Y., WANG, L., FAN, D., AND GONG, B. (2018). **A semi-supervised two-stage approach to learning from noisy labels**. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1215–1224. IEEE.

[23] DOMAHIDI, A., CHU, E., AND BOYD, S. (2013). **ECOS: An SOCP solver for embedded systems**. In *2013 European Control Conference (ECC)*, pages 3071–3076. IEEE.

[24] DUA, D. AND GRAFF, C. (2017). **UCI Machine Learning Repository**.

[25] FRÄNTI, P., REZAEI, M., AND ZHAO, Q. (2014). **Centroid index: cluster level similarity measure**. *Pattern Recognition*, 47(9):3034–3045.

[26] FRÉNAY, B. AND VERLEYSEN, M. (2013). **Classification in the presence of label noise: a survey**. *IEEE transactions on neural networks and learning systems*, 25(5):845–869.

[27] GIVONI, I. AND FREY, B. (2009). **Semi-supervised affinity propagation with instance-level constraints**. In *Artificial intelligence and statistics*, pages 161–168. PMLR.

[28] GOLDBERGER, J., GORDON, S., GREENSPAN, H., ET AL. (2003). **An Efficient Image Similarity Measure Based on Approximations of KL-Divergence Between Two Gaussian Mixtures.** In *ICCV*, volume 3, pages 487–493.

[29] GOPALAN, P. K., GERRISH, S., FREEDMAN, M., BLEI, D. M., AND MIMNO, D. M. (2012). **Scalable inference of overlapping communities**. In *Advances in Neural Information Processing Systems*, pages 2249–2257.

[30] GRIBEL, D. AND VIDAL, T. (2019). **HG-means: A scalable hybrid genetic algorithm for minimum sum-of-squares clustering**. *Pattern Recognition*, 88:569–583.

[31] GRIBEL, D., VIDAL, T., AND GENDREAU, M. (2021). **Assortative-constrained stochastic block models**. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 6212–6218. IEEE.

[32] HARTIGAN, J. A. AND WONG, M. A. (1979). **Algorithm AS 136: A k-means clustering algorithm**. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108.

[33] HIEP, T. K., DUC, N. M., AND TRUNG, B. Q. (2016). **Local search approach for the pairwise constrained clustering problem**. In *Proceedings of the Seventh Symposium on Information and Communication Technology*, pages 115–122. ACM.

[34] HOLLAND, P. W., LASKEY, K. B., AND LEINHARDT, S. (1983). **Stochastic blockmodels: First steps**. *Social Networks*, 5(2):109–137.

[35] HUSS, M. AND HOLME, P. (2007). **Currency and commodity metabolites: their identification and relation to the modularity of metabolic networks**. *IET Systems Biology*, 1(5):280–285.

[36] KARRER, B. AND NEWMAN, M. E. (2011). **Stochastic blockmodels and community structure in networks**. *Physical Review E*, 83(1):016107.

[37] KREIMER, A., BORENSTEIN, E., GOPHNA, U., AND RUPPIN, E. (2008). **The evolution of modularity in bacterial metabolic networks**. *Proceedings of the National Academy of Sciences*, 105(19):6976–6981.

[38] KVALSETH, T. O. (1987). **Entropy and correlation: Some comments**. *IEEE Transactions on Systems, Man, and Cybernetics*, 17(3):517–519.

[39] LAMEU, E. L., BORGES, F. S., BORGES, R. R., IAROSZ, K. C., CALDAS, I. L., BATISTA, A. M., VIANA, R. L., AND KURTHS, J. (2016). **Suppression of phase synchronisation in network based on cat's brain**. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 26(4):043107.

[40] LEE, C. AND WILKINSON, D. J. (2019). **A review of stochastic block models and extensions for graph clustering**. *Applied Network Science*, 4(1):122.

[41] LEI, J., RINALDO, A., ET AL. (2015). **Consistency of spectral clustering in stochastic block models**. *The Annals of Statistics*, 43(1):215–237.

[42] LI, W., AHN, S., AND WELLING, M. (2016). **Scalable MCMC for mixed membership stochastic blockmodels**. In *Artificial Intelligence and Statistics*, pages 723–731.

[43] LI, Z., LIU, J., AND TANG, X. (2009). **Constrained clustering via spectral regularization**. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 421–428. IEEE.

[44] LIKAS, A., VLASSIS, N., AND VERBEEK, J. J. (2003). **The global k-means clustering algorithm.** *Pattern recognition*, 36(2):451–461.

[45] LU, X. AND SZYMANSKI, B. K. (2019). **Regularized Stochastic Block Model for robust community detection in complex networks.** *arXiv preprint arXiv:1903.11751*.

[46] LÜCKE, J. AND FORSTER, D. (2019). **k-means as a variational EM approximation of Gaussian mixture models.** *Pattern Recognition Letters*, 125:349–356.

[47] MCDAID, A. F., MURPHY, T. B., FRIEL, N., AND HURLEY, N. J. (2013). **Improved Bayesian inference for the stochastic block model with application to large networks.** *Computational Statistics & Data Analysis*, 60:12–31.

[48] MOORE, C., YAN, X., ZHU, Y., ROUQUIER, J.-B., AND LANE, T. (2011). **Active learning for node classification in assortative and disassortative networks.** In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data mining*, pages 841–849. ACM.

[49] NATARAJAN, N., DHILLON, I. S., RAVIKUMAR, P., AND TEWARI, A. (2013). **Learning with noisy labels.** In *NIPS*, volume 26, pages 1196–1204.

[50] NETTLETON, D. F., ORRIOLS-PUIG, A., AND FORNELLS, A. (2010). **A study of the effect of different types of noise on the precision of supervised learning techniques.** *Artificial intelligence review*, 33(4):275–306.

[51] NEWMAN, M. E. (2016). **Equivalence between modularity optimization and maximum likelihood methods for community detection.** *Physical Review E*, 94(5):052315.

[52] NOWICKI, K. AND SNIJDERS, T. A. B. (2001). **Estimation and prediction for stochastic blockstructures.** *Journal of the American Statistical Association*, 96(455):1077–1087.

[53] ORDIN, B. AND BAGIROV, A. M. (2015). **A heuristic algorithm for solving the minimum sum-of-squares clustering problems.** *Journal of Global Optimization*, 61(2):341–361.

[54] ORTEGO, D., ARAZO, E., ALBERT, P., O'CONNOR, N. E., AND MCGUINNESS, K. (2021). **Towards robust learning with different label noise**

distributions. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 7020–7027. IEEE.

[55] PEEL, L., LARREMORE, D. B., AND CLAUSET, A. (2017). **The ground truth about metadata and community detection in networks**. *Science Advances*, 3(5):e1602548.

[56] PEIXOTO, T. P. (2019). **Bayesian stochastic blockmodeling**. *Advances in Network Clustering and Blockmodeling*, pages 289–332.

[57] PELLEG, D. AND BARAS, D. (2007). **K-means with large and noisy constraint sets**. In *European Conference on Machine Learning*, pages 674–682. Springer.

[58] QIN, T. AND ROHE, K. (2013). **Regularized spectral clustering under the degree-corrected stochastic blockmodel**. In *Advances in Neural Information Processing Systems*, pages 3120–3128.

[59] RAVASZ, E., SOMERA, A. L., MONGRU, D. A., OLTVAI, Z. N., AND BARABÁSI, A.-L. (2002). **Hierarchical organization of modularity in metabolic networks**. *Science*, 297(5586):1551–1555.

[60] ROHE, K., CHATTERJEE, S., YU, B., ET AL. (2011). **Spectral clustering and the high-dimensional stochastic blockmodel**. *The Annals of Statistics*, 39(4):1878–1915.

[61] SCANNELL, J. W., BLAKEMORE, C., AND YOUNG, M. P. (1995). **Analysis of connectivity in the cat cerebral cortex**. *Journal of Neuroscience*, 15(2):1463–1483.

[62] SCHWENKER, F. AND TRENTIN, E. (2014). **Pattern classification and clustering: A review of partially supervised learning approaches**. *Pattern Recognition Letters*, 37:4–14.

[63] SHENTAL, N., BAR-HILLEL, A., HERTZ, T., AND WEINSHALL, D. (2004). **Computing Gaussian mixture models with EM using equivalence constraints**. *Advances in neural information processing systems*, 16(8):465–472.

[64] STANLEY, N., BONACCI, T., KWITT, R., NIETHAMMER, M., AND MUCHA, P. J. (2019). **Stochastic block models with multiple continuous attributes**. *Applied Network Science*, 4(1):1–22.

[65] VASSILVITSKII, S. AND ARTHUR, D. (2006). **k-means++: The advantages of careful seeding**. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035.

[66] WAGSTAFF, K., CARDIE, C., ROGERS, S., SCHRÖDL, S., ET AL. (2001). **Constrained k-means clustering with background knowledge**. In *Icml*, volume 1, pages 577–584.

[67] WANG, X., QIAN, B., AND DAVIDSON, I. (2014). **On constrained spectral clustering and its applications**. *Data Mining and Knowledge Discovery*, 28(1):1–30.

[68] WANG, Y. R., BICKEL, P. J., ET AL. (2017). **Likelihood-based model selection for stochastic block models**. *The Annals of Statistics*, 45(2):500–528.

[69] WU, S., XIA, X., LIU, T., HAN, B., GONG, M., WANG, N., LIU, H., AND NIU, G. (2021). **Class2simi: A noise reduction perspective on learning with noisy labels**. ICML.

[70] XIONG, S., AZIMI, J., AND FERN, X. Z. (2013). **Active learning of constraints for semi-supervised clustering**. *IEEE Transactions on Knowledge and Data Engineering*, 26(1):43–54.

[71] XU, R. AND WUNSCH, D. (2005). **Survey of clustering algorithms**. *IEEE Transactions on neural networks*, 16(3):645–678.

[72] ZHU, X. J. (2005). **Semi-supervised learning literature survey**.

A
Supplement to “Assortative-Constrained Stochastic Block Models”

A.1
Degree-corrected SBM Formulation

As stated by Karrer and Newman [36], the degree-corrected SBM likelihood is defined as follows:

$$P(\boldsymbol{A}|\boldsymbol{\Omega}, \boldsymbol{y}) = \prod_{i}^{N} \frac{\left(\frac{1}{2}\frac{k_i^2}{2m}\omega_{y_i y_i}\right)^{\frac{A_{ii}}{2}}}{(\frac{1}{2}A_{ii})!} e^{-\frac{k_i^2}{2m}\frac{\omega_{y_i y_i}}{2}} \times \prod_{i<j}^{N} \frac{\left(\frac{k_i k_j}{2m}\omega_{y_i y_j}\right)^{A_{ij}}}{A_{ij}!} e^{-\frac{k_i k_j}{2m}\omega_{y_i y_j}}, \tag{A-1}$$

where $y_i \in \{1, \ldots, K\}$ is the cluster for which the sample $\boldsymbol{x}_i$ is assigned, and $\frac{k_i k_j}{2m}\omega_{y_i y_j}$ is the expected number of edges between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. Taking the logarithm, we end-up with the following log-likelihood function:

$$\begin{aligned}
\mathcal{L}(\boldsymbol{A}|\boldsymbol{\Omega}, \boldsymbol{y}) = & \sum_{i}^{N} \frac{A_{ii}}{2} \log\left(\frac{k_i^2}{4m}\omega_{y_i y_i}\right) - \frac{k_i^2}{4m}\omega_{y_i y_i} - \log\left(\frac{1}{2}A_{ii}\right)! + \\
& \sum_{i<j}^{N} A_{ij} \log\left(\frac{k_i k_j}{2m}\omega_{y_i y_j}\right) - \frac{k_i k_j}{2m}\omega_{y_i y_j} - \log(A_{ij})! \\
= & \sum_{i}^{N} \frac{1}{2}A_{ii} \log\left(\frac{k_i^2}{4m}\right) + \frac{1}{2}A_{ii}\log(\omega_{y_i y_i}) - \frac{1}{2}\frac{k_i^2}{2m}\omega_{y_i y_i} - \log\left(\frac{1}{2}A_{ii}\right)! + \\
& \sum_{i \neq j}^{N} \frac{1}{2}A_{ij} \log\left(\frac{k_i k_j}{2m}\right) + \frac{1}{2}A_{ij}\log(\omega_{y_i y_j}) - \frac{1}{2}\frac{k_i k_j}{2m}\omega_{y_i y_j} - \frac{1}{2}\log(A_{ij})!.
\end{aligned} \tag{A-2}$$

The terms $\frac{1}{2}A_{ii}\log\left(\frac{k_i^2}{4m}\right)$, $\log\left(\frac{1}{2}A_{ii}\right)!$, $\frac{1}{2}A_{ij}\log\left(\frac{k_i k_j}{2m}\right)$, and $\frac{1}{2}\log(A_{ij})!$ do not depend on parameters $\boldsymbol{\Omega}$ and $\boldsymbol{y}$, and can be neglected. Thus, the DC-SBM log-likelihood can be written as:

$$\begin{aligned}\mathcal{L}(\boldsymbol{A}|\boldsymbol{\Omega},\boldsymbol{y}) &= \sum_{i}^{N}\frac{1}{2}A_{ii}\log(\omega_{y_iy_i}) - \frac{1}{2}\frac{k_i^2}{2m}\omega_{y_iy_i} + \sum_{i\neq j}^{N}\frac{1}{2}A_{ij}\log(\omega_{y_iy_j}) - \frac{1}{2}\frac{k_ik_j}{2m}\omega_{y_iy_j}\\ &= \frac{1}{2}\sum_{ij}^{N}A_{ij}\log(\omega_{y_iy_j}) - \frac{k_ik_j}{2m}\omega_{y_iy_j}.\end{aligned} \tag{A-3}$$

By replacing the variable y by the binary cluster indicator $\boldsymbol{Z}\in\{0,1\}^{N,K}$, we have:

$$\begin{aligned}\mathcal{L}(\boldsymbol{A}|\boldsymbol{\Omega},\boldsymbol{Z}) &= \frac{1}{2}\sum_{rs}^{K}\sum_{ij}^{N}\left(A_{ij}\log(\omega_{rs}) - \frac{k_ik_j}{2m}\omega_{rs}\right)z_{ir}z_{js}\\ &= \frac{1}{2}\sum_{rs}^{K}\sum_{ij}^{N}z_{ir}z_{js}A_{ij}\log(\omega_{rs}) - \frac{1}{4m}\sum_{rs}^{K}\sum_{ij}^{N}z_{ir}z_{js}k_ik_j\omega_{rs}\\ &= \frac{1}{2}\sum_{rs}^{K}\log(\omega_{rs})\left(\sum_{ij}^{N}z_{ir}z_{js}A_{ij}\right) - \frac{1}{4m}\sum_{rs}^{K}\omega_{rs}\left(\sum_{ij}^{N}z_{ir}z_{js}k_ik_j\right)\\ &= \frac{1}{2}\sum_{rs}^{K}\log(\omega_{rs})m_{rs} - \frac{1}{4m}\sum_{rs}^{K}\omega_{rs}\left(\sum_{i}^{N}z_{ir}k_i\sum_{j}^{N}z_{js}k_j\right)\\ &= \frac{1}{2}\sum_{rs}^{K}\log(\omega_{rs})m_{rs} - \frac{1}{4m}\sum_{rs}^{K}\omega_{rs}\kappa_r\kappa_s\\ &= \frac{1}{2}\sum_{rs}^{K}m_{rs}\log(\omega_{rs}) - \frac{\kappa_r\kappa_s}{2m}\omega_{rs},\end{aligned} \tag{A-4}$$

where k_i is the degree of node i, $m_{rs} = \sum_{ij}^{N}A_{ij}z_{ir}z_{js}$ is the number of edges between clusters r and s, and $\kappa_r = \sum_{s}^{K}m_{rs} = \sum_{i}^{N}k_iz_{ir}$ is the sum of the degrees of nodes in group r. Taking the derivative regarding ω_{rs} and setting Equation (A-4) to zero, we have:

$$\hat{\omega}_{rs} = \frac{2m\cdot m_{rs}}{\kappa_r\kappa_s}. \tag{A-5}$$

Substituting $\hat{\omega}_{rs}$ in Equation (A-4), we have:

$$\begin{aligned}\mathcal{L}(\boldsymbol{A}|\boldsymbol{Z}) &= \frac{1}{2}\sum_{rs}^{K}m_{rs}\log(2m) + m_{rs}\log\left(\frac{m_{rs}}{\kappa_r\kappa_s}\right) - m_{rs}\\ &= \frac{1}{2}\sum_{rs}^{K}m_{rs}\log\left(\frac{m_{rs}}{\kappa_r\kappa_s}\right) + \frac{1}{2}\sum_{rs}^{K}m_{rs}(\log(2m)-1)\\ &= \frac{1}{2}\sum_{rs}^{K}m_{rs}\log\left(\frac{m_{rs}}{\kappa_r\kappa_s}\right) + C,\end{aligned} \tag{A-6}$$

where $C = m(\log(2m) - 1)$ is constant and can be dropped from the log-likelihood function. Re-writing Equation (A-6) in terms of the pair of samples i and j, we have:

$$\begin{aligned}\mathcal{L}(\boldsymbol{A}|\boldsymbol{Z}) &= \frac{1}{2}\sum_{rs}^{K}\left(\sum_{ij}^{N} A_{ij} z_{ir} z_{js}\right)\log\left(\frac{m_{rs}}{\kappa_r\kappa_s}\right) \\ &= \frac{1}{2}\sum_{rs}^{K}\sum_{ij}^{N}\left(A_{ij}\log\left(\frac{m_{rs}}{\kappa_r\kappa_s}\right)\right) z_{ir} z_{js}.\end{aligned} \tag{A-7}$$

A.2 Detailed Results on Networks Generated from a PPM

Table A.1 presents the results in terms of NMI performance and computational efforts obtained with the DC-SBM and AC-DC-SBM models on networks generated from PPMs.

$\omega_{\mathrm{OUT}}/\omega_{\mathrm{IN}}$	Avg. NMI		Median solution NMI		Best solution NMI		Avg. Time (s)	
	DC-SBM	AC-DC-SBM	DC-SBM	AC-DC-SBM	DC-SBM	AC-DC-SBM	DC-SBM	AC-DC-SBM
0.10	0.7490	**0.9595**	0.7721	**1.0000**	**1.0000**	**1.0000**	0.0341	0.4395
0.15	0.7162	**0.8961**	0.7504	**0.9708**	**0.9708**	**0.9708**	0.0370	0.5273
0.20	0.5470	**0.8121**	0.6135	**0.9115**	**0.9115**	**0.9115**	0.0386	0.5157
0.25	0.3449	**0.5505**	0.3330	**0.5798**	0.7890	**0.8211**	0.0414	0.6264
0.30	0.2420	**0.3412**	0.2219	**0.3260**	0.5074	**0.7196**	0.0397	0.6594
0.35	0.1343	**0.2002**	0.1266	**0.1947**	**0.3281**	0.3208	0.0354	0.7323
0.40	0.0980	**0.1392**	0.0939	**0.1323**	0.1399	**0.2369**	0.0389	0.7551
0.45	0.0891	**0.1084**	0.0798	**0.1015**	0.1048	**0.1566**	0.0398	0.8789
0.50	0.0653	**0.0770**	0.0635	**0.0746**	**0.1019**	0.0860	0.0368	0.9390
0.55	0.0521	**0.0663**	0.0423	**0.0628**	**0.0692**	0.0565	0.0374	0.9961
0.60	0.0486	**0.0616**	0.0433	**0.0618**	**0.0817**	0.0793	0.0395	1.0087

Table A.1: Performance of DC-SBM and AC-DC-SBM on networks generated from PPMs with varying degree of assortativity.

A.3
Detailed Results on Networks Generated from SBMs

Tables A.2 and A.3 present the NMI performance obtained with the modularity maximization, DC-SBM, and AC-DC-SBM models. We present the results on 50 synthetic networks generated from SBMs. Table A.2 considers the average NMI of 50 independent runs in each dataset, whereas Table A.3 reports the NMI obtained on the 10% best solutions in terms of log-likelihood value.

Dataset	Average NMI			Median solution NMI			Best solution NMI		
	Max-Mod	DC-SBM	AC-DC-SBM	Max-Mod	DC-SBM	AC-DC-SBM	Max-Mod	DC-SBM	AC-DC-SBM
A001	0.5753	0.7411	0.9248	0.5653	0.8644	0.9692	0.8547	0.9692	0.9692
A002	0.4683	0.4354	0.5662	0.4629	0.4405	0.6206	0.5393	0.8096	0.8497
A003	0.1319	0.0759	0.1237	0.1294	0.0710	0.1163	0.2513	0.1335	0.2061
A004	0.7191	0.7158	0.7390	0.7202	0.7441	0.7484	0.9619	0.7372	0.7474
A005	0.4682	0.4967	0.6865	0.4768	0.4555	0.6503	0.5870	0.9085	0.9085
A006	0.2923	0.2369	0.3506	0.2925	0.2648	0.3982	0.3923	0.3982	0.4028
A007	0.7312	0.7890	0.8108	0.7083	0.8221	0.8221	0.8968	0.8255	0.8255
A008	0.6068	0.5836	0.6730	0.6095	0.5733	0.6851	0.7587	0.8139	0.8139
A009	0.3839	0.5326	0.5846	0.3982	0.5781	0.5867	0.4045	0.5908	0.5810
A010	0.7113	0.5585	0.7048	0.7123	0.5682	0.6722	0.8772	0.8596	0.8596
A011	0.4654	0.6876	0.7467	0.4634	0.6963	0.7286	0.5251	0.8300	0.8300
A012	0.4794	0.4662	0.6580	0.4755	0.4252	0.6618	0.5775	0.8178	0.8178
A013	0.5157	0.6304	0.7418	0.5219	0.5786	0.8788	0.6021	0.8788	0.8788
A014	0.4090	0.6824	0.7045	0.4061	0.6993	0.7188	0.4496	0.7191	0.7191
A015	0.3521	0.3185	0.4214	0.3616	0.3479	0.4402	0.3512	0.4394	0.6899
A016	0.4259	0.2400	0.3005	0.4320	0.2331	0.3438	0.5654	0.5272	0.6138
A017	0.7163	0.7763	0.8213	0.7139	0.7690	0.8647	0.7045	0.8647	0.8647
A018	0.6863	0.8326	0.9465	0.6912	0.9707	0.9707	0.7598	0.9707	0.9707
A019	0.4795	0.4666	0.5347	0.4721	0.4857	0.5508	0.5784	0.6128	0.6128
A020	0.4271	0.4030	0.4909	0.4287	0.4509	0.4955	0.4824	0.5086	0.4981
A021	0.6032	0.7490	0.8116	0.5923	0.7533	0.8282	0.7347	0.8412	0.8412
A022	0.5842	0.5884	0.6562	0.6045	0.6356	0.6684	0.6624	0.7473	0.7166
A023	0.3945	0.4401	0.5213	0.4020	0.4204	0.4562	0.5007	0.8705	0.9075
A024	0.2849	0.2500	0.3592	0.2887	0.2344	0.3370	0.4372	0.5094	0.7258
A025	0.4535	0.4131	0.4647	0.4445	0.4276	0.4741	0.5495	0.4977	0.5543
A026	0.6880	0.7901	0.8813	0.6960	0.9189	0.9189	0.7042	0.9189	0.9189
A027	0.3367	0.3736	0.4387	0.3488	0.4032	0.4556	0.4446	0.4877	0.5119
A028	0.7736	0.8842	0.9411	0.7570	0.9708	0.9708	0.8986	0.9708	0.9708
A029	0.4755	0.5423	0.5977	0.5028	0.5193	0.6400	0.5843	0.7084	0.7326
A030	0.5236	0.5026	0.6132	0.5596	0.4738	0.6699	0.6276	0.7061	0.7061
A031	0.4275	0.3095	0.4638	0.4436	0.2773	0.4726	0.6838	0.5965	0.7214
A032	0.4332	0.3769	0.4600	0.4399	0.4290	0.4552	0.5438	0.4929	0.5696
A033	0.5290	0.5754	0.7670	0.5228	0.5288	0.8437	0.6421	0.8211	0.8211
A034	0.3047	0.2468	0.3701	0.3099	0.2182	0.3698	0.4407	0.6369	0.6369
A035	0.5459	0.5748	0.6974	0.5479	0.6773	0.7431	0.7068	0.7926	0.7431
A036	0.8283	0.9175	0.9868	0.8553	1.0000	1.0000	0.8553	1.0000	1.0000
A037	0.1246	0.0871	0.1171	0.1203	0.0786	0.1057	0.2045	0.1298	0.1797
A038	0.5430	0.6129	0.7146	0.5728	0.6960	0.7317	0.6289	0.7121	0.8395
A039	0.8143	0.6968	0.7440	0.7829	0.7481	0.7836	1.0000	0.7571	0.8001
A040	0.5225	0.4193	0.6255	0.5429	0.3934	0.7207	0.6481	0.8567	0.8567
A041	0.6846	0.5637	0.5855	0.6788	0.5746	0.5839	1.0000	0.5809	0.6213
A042	0.6556	0.7259	0.8167	0.6641	0.7756	0.8615	0.7408	0.8615	0.8615
A043	0.8229	0.8822	0.9089	0.8391	0.9213	0.9213	0.9081	0.9213	0.9213
A044	0.3442	0.3081	0.3579	0.3581	0.3547	0.3649	0.3514	0.3453	0.4317
A045	0.5816	0.8116	0.9356	0.5875	0.7977	1.0000	0.6966	1.0000	1.0000
A046	0.4676	0.5134	0.5494	0.4548	0.5275	0.5433	0.5251	0.6309	0.7474
A047	0.6439	0.5237	0.6584	0.6652	0.4814	0.6866	0.7942	0.7316	0.6765
A048	0.7097	0.7069	0.7741	0.6891	0.7494	0.7723	0.9317	0.8182	0.8182
A049	0.6773	0.6062	0.8997	0.6694	0.6758	0.9409	0.8096	0.9409	0.9411
A050	0.6983	0.7212	0.8139	0.7313	0.7884	0.8183	0.7514	0.8183	0.7891

Table A.2: Performance of modularity maximization, DC-SBM and AC-DC-SBM on networks generated from general SBMs.

Dataset	Average NMI			Median solution NMI			Best solution NMI		
	Max-Mod	DC-SBM	AC-DC-SBM	Max-Mod	DC-SBM	AC-DC-SBM	Max-Mod	DC-SBM	AC-DC-SBM
A001	0.7021	0.9692	0.9692	0.6646	0.9692	0.9692	0.8547	0.9692	0.9692
A002	0.5305	0.7248	0.7365	0.5393	0.6911	0.7280	0.5393	0.8096	0.8497
A003	0.2016	0.1255	0.1649	0.2141	0.1212	0.1320	0.2513	0.1335	0.2061
A004	0.9005	0.7480	0.7542	0.8595	0.7389	0.7474	0.9619	0.7372	0.7474
A005	0.5332	0.9085	0.9085	0.5334	0.9085	0.9085	0.5870	0.9085	0.9085
A006	0.3255	0.3893	0.4262	0.3299	0.3982	0.4365	0.3923	0.3982	0.4028
A007	0.8968	0.8255	0.8255	0.8968	0.8255	0.8255	0.8968	0.8255	0.8255
A008	0.7454	0.7414	0.8139	0.7504	0.7468	0.8139	0.7587	0.8139	0.8139
A009	0.4164	0.5684	0.5825	0.4194	0.5701	0.5810	0.4045	0.5908	0.5810
A010	0.8772	0.8508	0.8596	0.8772	0.8596	0.8596	0.8772	0.8596	0.8596
A011	0.5434	0.8069	0.8300	0.5337	0.8300	0.8300	0.5251	0.8300	0.8300
A012	0.6475	0.8456	0.8711	0.6607	0.8178	0.9066	0.5775	0.8178	0.8178
A013	0.6182	0.8788	0.8788	0.6021	0.8788	0.8788	0.6021	0.8788	0.8788
A014	0.4648	0.7413	0.7191	0.4570	0.7191	0.7191	0.4496	0.7191	0.7191
A015	0.4455	0.4440	0.6787	0.4505	0.4634	0.6899	0.3512	0.4394	0.6899
A016	0.5905	0.4626	0.5356	0.5820	0.4764	0.5161	0.5654	0.5272	0.6138
A017	0.7010	0.8647	0.8647	0.7045	0.8647	0.8647	0.7045	0.8647	0.8647
A018	0.7522	0.9707	0.9707	0.7598	0.9707	0.9707	0.7598	0.9707	0.9707
A019	0.5739	0.6116	0.6271	0.5784	0.6128	0.6302	0.5784	0.6128	0.6128
A020	0.4153	0.5268	0.4961	0.3935	0.5086	0.4909	0.4824	0.5086	0.4981
A021	0.7269	0.8412	0.8412	0.7347	0.8412	0.8412	0.7347	0.8412	0.8412
A022	0.6511	0.7062	0.6944	0.6624	0.7054	0.7054	0.6624	0.7473	0.7166
A023	0.4993	0.7579	0.7839	0.5007	0.7602	0.7105	0.5007	0.8705	0.9075
A024	0.4225	0.4307	0.6347	0.4186	0.4303	0.5824	0.4372	0.5094	0.7258
A025	0.5034	0.4993	0.5293	0.5192	0.4977	0.5398	0.5495	0.4977	0.5543
A026	0.7724	0.9189	0.9189	0.7980	0.9189	0.9189	0.7042	0.9189	0.9189
A027	0.3980	0.4713	0.4923	0.3771	0.4877	0.4860	0.4446	0.4877	0.5119
A028	0.8986	0.9708	0.9708	0.8986	0.9708	0.9708	0.8986	0.9708	0.9708
A029	0.5464	0.6825	0.6975	0.5329	0.6823	0.7083	0.5843	0.7084	0.7326
A030	0.6035	0.7108	0.7108	0.6046	0.7061	0.7061	0.6276	0.7061	0.7061
A031	0.6556	0.6022	0.6756	0.6738	0.5965	0.7214	0.6838	0.5965	0.7214
A032	0.4637	0.4695	0.4837	0.4505	0.4698	0.4655	0.5438	0.4929	0.5696
A033	0.6450	0.8459	0.8527	0.6421	0.8437	0.8606	0.6421	0.8211	0.8211
A034	0.4091	0.4647	0.5980	0.4025	0.5276	0.5834	0.4407	0.6369	0.6369
A035	0.6546	0.7361	0.7489	0.6834	0.7153	0.7431	0.7068	0.7926	0.7431
A036	0.8601	1.0000	1.0000	0.8553	1.0000	1.0000	0.8553	1.0000	1.0000
A037	0.1412	0.1180	0.1592	0.1137	0.1222	0.1627	0.2045	0.1298	0.1797
A038	0.6316	0.7286	0.7866	0.6325	0.7110	0.8141	0.6289	0.7121	0.8395
A039	1.0000	0.7552	0.7872	1.0000	0.7533	0.7844	1.0000	0.7571	0.8001
A040	0.6744	0.8435	0.8567	0.6705	0.8567	0.8567	0.6481	0.8567	0.8567
A041	1.0000	0.5913	0.5997	1.0000	0.5839	0.5966	1.0000	0.5809	0.6213
A042	0.7694	0.8615	0.8615	0.7553	0.8615	0.8615	0.7408	0.8615	0.8615
A043	0.9081	0.9213	0.9213	0.9081	0.9213	0.9213	0.9081	0.9213	0.9213
A044	0.3609	0.3949	0.3979	0.3637	0.3849	0.3669	0.3514	0.3453	0.4317
A045	0.7003	1.0000	1.0000	0.6966	1.0000	1.0000	0.6966	1.0000	1.0000
A046	0.5355	0.6760	0.6675	0.5255	0.6852	0.6361	0.5251	0.6309	0.7474
A047	0.7439	0.6975	0.7417	0.7942	0.7007	0.7155	0.7942	0.7316	0.6765
A048	0.9016	0.7829	0.7951	0.9099	0.7718	0.8052	0.9317	0.8182	0.8182
A049	0.8096	0.9409	0.9411	0.8096	0.9409	0.9411	0.8096	0.9409	0.9411
A050	0.7551	0.8183	0.7891	0.7576	0.8183	0.7891	0.7514	0.8183	0.7891

Table A.3: Performance of modularity maximization, DC-SBM and AC-DC-SBM on networks generated from general SBMs (the 10% best solutions for each dataset).

B
Supplement to "Semi-supervised Clustering with Inaccurate Pairwise Annotations"

B.1
Ordinary SBM Formulation

The SBM likelihood for Poisson-distributed edges is defined as follows [36]:

$$P(\boldsymbol{A}|\boldsymbol{\Omega}, \boldsymbol{y}) = \prod_{i}^{N} \frac{\left(\frac{1}{2}\omega_{y_i y_i}\right)^{\frac{A_{ii}}{2}}}{(\frac{1}{2}A_{ii})!} e^{-\frac{\omega_{y_i y_i}}{2}} \times \prod_{i<j}^{N} \frac{\left(\omega_{y_i y_j}\right)^{A_{ij}}}{A_{ij}!} e^{-\omega_{y_i y_j}}, \tag{B-1}$$

where $y_i \in \{1, \ldots, K\}$ is the cluster for which the sample $\boldsymbol{x}_i$ is assigned, and $\omega_{y_i y_j}$ is the expected number of edges between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. Taking the logarithm, we have the following log-likelihood function:

$$\begin{aligned}
\mathcal{L}(\boldsymbol{A}|\boldsymbol{\Omega}, \boldsymbol{y}) =& \sum_{i}^{N} \frac{A_{ii}}{2} \log\left(\frac{\omega_{y_i y_i}}{2}\right) - \frac{1}{2}\omega_{y_i y_i} - \log\left(\frac{1}{2}A_{ii}\right)! + \\
& \sum_{i<j}^{N} A_{ij} \log\left(\omega_{y_i y_j}\right) - \omega_{y_i y_j} - \log(A_{ij})! \\
=& \sum_{i}^{N} \frac{1}{2}A_{ii} \log\left(\frac{1}{2}\right) + \frac{1}{2}A_{ii} \log(\omega_{y_i y_i}) - \frac{1}{2}\omega_{y_i y_i} - \log\left(\frac{1}{2}A_{ii}\right)! + \\
& \sum_{i\neq j}^{N} \frac{1}{2}A_{ij} \log(\omega_{y_i y_j}) - \frac{1}{2}\omega_{y_i y_j} - \frac{1}{2}\log(A_{ij})!.
\end{aligned} \tag{B-2}$$

The terms $\frac{1}{2}A_{ii}\log\left(\frac{1}{2}\right)$, $\log\left(\frac{1}{2}A_{ii}\right)!$, and $\frac{1}{2}\log(A_{ij})!$ do not depend on parameters $\boldsymbol{\Omega}$ and $\boldsymbol{y}$, and can be neglected. Thus, the SBM log-likelihood can be written as:

$$\mathcal{L}(\boldsymbol{A}|\boldsymbol{\Omega}, \boldsymbol{y}) = \frac{1}{2}\sum_{ij}^{N} A_{ij} \log\left(\omega_{y_i y_j}\right) - \omega_{y_i y_j}. \tag{B-3}$$

If we replace the membership variable $\boldsymbol{y}$ by the binary clustering indicator

$\boldsymbol{Z} \in \{0,1\}^{N,K}$, we have:

$$\begin{aligned}
\mathcal{L}(\boldsymbol{A}|\boldsymbol{\Omega},\boldsymbol{Z}) =& \frac{1}{2}\sum_{rs}^{N}\sum_{ij}^{N}\left(A_{ij}\log\left(\omega_{rs}\right)-\omega_{rs}\right) z_{ir}z_{js} \\
=& \frac{1}{2}\sum_{rs}^{N}\sum_{ij}^{N} z_{ir}z_{js}A_{ij}\log\left(\omega_{rs}\right) - \frac{1}{2}\sum_{rs}^{N}\sum_{ij}^{N} z_{ir}z_{js}\omega_{rs} \\
=& \frac{1}{2}\sum_{rs}^{N}\log\left(\omega_{rs}\right)\left(\sum_{ij}^{N} z_{ir}z_{js}A_{ij}\right) - \frac{1}{2}\sum_{rs}^{N}\omega_{rs}\left(\sum_{ij}^{N} z_{ir}z_{js}\right) \\
=& \frac{1}{2}\sum_{rs}^{N}\log\left(\omega_{rs}\right)m_{rs} - \frac{1}{2}\sum_{rs}^{N}\omega_{rs}\left(\sum_{i}^{N} z_{ir}\right)\left(\sum_{j}^{N} z_{js}\right) \\
=& \frac{1}{2}\sum_{rs}^{N} m_{rs}\log\left(\omega_{rs}\right) - n_r n_s\omega_{rs}.
\end{aligned} \tag{B-4}$$

The maximum value of ω_{rs} can be obtained by derivation

$$\hat{\omega}_{rs} = \frac{m_{rs}}{n_r n_s}, \tag{B-5}$$

and plugged into $\mathcal{L}(\boldsymbol{A}|\boldsymbol{\Omega},\boldsymbol{Z})$, leading to:

$$\begin{aligned}
\mathcal{L}(\boldsymbol{A}|\boldsymbol{Z}) =& \frac{1}{2}\sum_{rs}^{N} m_{rs}\log\left(\frac{m_{rs}}{n_r n_r}\right) - m_{rs} \\
=& \frac{1}{2}\sum_{rs}^{N} m_{rs}\log\left(\frac{m_{rs}}{n_r n_r}\right) - \frac{1}{2}\sum_{rs}^{N} m_{rs}.
\end{aligned} \tag{B-6}$$

As $m = \frac{1}{2}\sum_{rs}^{N} m_{rs}$ is a constant, the log-likelihood can be simplified to:

$$\begin{aligned}
\mathcal{L}(\boldsymbol{A}|\boldsymbol{Z}) =& \frac{1}{2}\sum_{rs}^{N} m_{rs}\log\left(\frac{m_{rs}}{n_r n_r}\right) \\
=& \frac{1}{2}\sum_{rs}^{K}\sum_{ij}^{N}\left(A_{ij}\log\left(\frac{m_{rs}}{n_r n_s}\right)\right) z_{ir}z_{js}.
\end{aligned} \tag{B-7}$$

B.2
The Relationship Between Within-clusters and Between-clusters Priors

Let $E_{\text{IN}}(\boldsymbol{Z}) = \sum_{r=1}^{K}(n_r(n_r+1))/2$ be the number of pairs in the dataset for which the two samples are in the same group, for a given $\boldsymbol{Z}$. Likewise, let $E_{\text{OUT}}(\boldsymbol{Z}) = \sum_{r<s}^{K} n_r n_s$ be the number of pairs with samples in different groups. Therefore, the expected number of correct *must-link* edges (same-group

samples correctly annotated) for a given $\boldsymbol{Z}$ and p is

$$\frac{2E_{\text{IN}}(\boldsymbol{Z})}{N(N+1)}mp, \tag{B-8}$$

and the expected number of erroneous *must-link* edges (different-group samples wrongly annotated) is:

$$\frac{2E_{\text{OUT}}(\boldsymbol{Z})}{N(N+1)}m(1-p). \tag{B-9}$$

Similarly, the expected number of correct *cannot-link* edges (different-group samples correctly annotated) for a given $\boldsymbol{Z}$ and p is

$$\frac{2E_{\text{OUT}}(\boldsymbol{Z})}{N(N+1)}mp, \tag{B-10}$$

and the expected number of erroneous *cannot-link* edges (same-group samples wrongly annotated) is:

$$\frac{2E_{\text{IN}}(\boldsymbol{Z})}{N(N+1)}m(1-p). \tag{B-11}$$

Thus, the expected number of *must-links* for a pair of samples in the same cluster, and the expected number of *cannot-links* for a pair of samples in different clusters is:

$$\left[\frac{2E_{\text{IN}}(\boldsymbol{Z})}{N(N+1)}mp\right] / E_{\text{IN}}(\boldsymbol{Z}) = \left[\frac{2E_{\text{OUT}}(\boldsymbol{Z})}{N(N+1)}mp\right] / E_{\text{OUT}}(\boldsymbol{Z}) = \frac{2mp}{N(N+1)}. \tag{B-12}$$

Similarly, the expected number of *cannot-links* for a pair of samples in the same cluster, and the expected number of *must-links* for a pair of samples in different clusters is:

$$\left[\frac{2E_{\text{OUT}}(\boldsymbol{Z})}{N(N+1)}m(1-p)\right] / E_{\text{OUT}}(\boldsymbol{Z}) = \left[\frac{2E_{\text{IN}}(\boldsymbol{Z})}{N(N+1)}m(1-p)\right] / E_{\text{IN}}(\boldsymbol{Z}) = \frac{2m(1-p)}{N(N+1)}. \tag{B-13}$$

Therefore, from Equations (B-12) and (B-13), we can state the following relation between $f^{+}_{\text{IN}}(\boldsymbol{Z}, p)$ and $f^{+}_{\text{OUT}}(\boldsymbol{Z}, p)$ in the *must-link* graph:

$$f^{+}_{\text{IN}}(\boldsymbol{Z}, p) = \frac{p}{1-p} f^{+}_{\text{OUT}}(\boldsymbol{Z}, p). \tag{B-14}$$

Analogously, for the *cannot-link* graph, we have:

$$f^{-}_{\mathrm{IN}}(\boldsymbol{Z}, p) = \frac{1-p}{p} f^{-}_{\mathrm{OUT}}(\boldsymbol{Z}, p). \tag{B-15}$$

B.3 Detailed Results on Mixtures of Gaussians

Tables B.1 to B.9 present the NMI on different mixtures of Gaussians obtained with the proposed semi-supervised clustering model. We report the results obtained with and without prior distributions. Each table consider a different combination of K and p, where $K \in \{2, 4, 6\}$, and $p \in \{0.8, 0.9, 1.0\}$.

Dataset	$m=0$	$m=100$		$m=200$		$m=300$		$m=400$		$m=500$		$m=600$		$m=700$		$m=800$	
Priors:		×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓
G001	0.9191	0.8890	0.8890	0.9191	0.9598	0.9598	0.9598	0.9191	0.9191	0.9598	0.9598	0.9598	0.9598	0.9191	0.9191	0.9598	0.9598
G002	0.1171	0.1493	0.1701	0.2139	0.2215	0.0418	0.2116	0.3596	0.3027	0.1226	0.1045	0.6382	0.6382	0.6676	0.6676	0.7282	0.7282
G003	0.0047	0.0192	0.0001	0.0862	0.0463	0.3801	0.2947	0.1316	0.1752	0.4576	0.4228	0.7596	0.7151	0.7456	0.7456	0.6725	0.6725
G004	0.3348	0.3317	0.3475	0.5682	0.5682	0.5369	0.5369	0.7576	0.7576	0.7596	0.7596	0.7104	0.7104	0.8165	0.8165	0.7356	0.7356
G005	0.7568	0.6776	0.6673	0.8048	0.8048	0.8319	0.8319	0.8081	0.8081	0.8573	0.8573	0.8627	0.8627	0.9599	0.9599	0.8627	0.8627
G006	0.0799	0.1598	0.1636	0.0040	0.0000	0.3010	0.3246	0.4882	0.4882	0.6348	0.6012	0.7459	0.7196	0.6924	0.6760	0.7955	0.7955
G007	0.2400	0.2451	0.2755	0.3901	0.3901	0.3905	0.5099	0.5642	0.5642	0.5472	0.4887	0.4921	0.4921	0.7577	0.7577	0.7651	0.7651
G008	0.5288	0.5617	0.5617	0.6338	0.6315	0.6916	0.6916	1.0000	1.0000	0.7587	0.7587	0.8314	0.8314	1.0000	1.0000	0.9600	0.9600
G009	0.1548	0.0473	0.0038	0.3123	0.3508	0.2561	0.2193	0.3690	0.3690	0.6045	0.6045	0.5443	0.6199	0.5084	0.5084	0.7853	0.8068
G010	0.1029	0.0862	0.0762	0.1074	0.1733	0.2457	0.3085	0.1927	0.2171	0.4318	0.4291	0.5231	0.5665	0.6956	0.6956	0.9029	0.9029
G011	0.8321	0.9291	0.9291	0.9291	0.9291	0.9291	0.9291	0.9597	0.9597	0.9596	0.9596	0.9597	0.9597	0.9291	0.9291	0.9596	0.9596
G012	0.4037	0.4194	0.4665	0.4704	0.4565	0.6830	0.6663	0.6930	0.6930	0.7356	0.7356	0.7576	0.7576	0.7817	0.7817	0.8626	0.8626
G013	0.4239	0.4874	0.4764	0.4900	0.4824	0.4786	0.4704	0.7104	0.7104	0.7668	0.7668	0.8383	0.8383	0.7834	0.7834	0.8052	0.8052
G014	0.0492	0.0465	0.0465	0.0260	0.0427	0.0147	0.1076	0.0158	0.0158	0.3422	0.2132	0.6383	0.5472	0.6931	0.6930	0.5561	0.6278
G015	0.9297	1.0000	1.0000	0.9598	0.9598	0.9598	0.9598	1.0000	1.0000	0.9598	0.9598	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G016	0.8056	0.7595	0.7595	0.8321	0.8082	0.8082	0.8082	0.9295	0.9295	0.8585	0.8585	0.9192	0.9192	0.8889	0.8889	0.9597	0.9597
G017	0.8158	0.8889	0.8889	0.8890	0.8890	0.8890	0.8890	0.8889	0.8889	0.9597	0.9597	0.9597	0.9597	0.9597	0.9597	1.0000	1.0000
G018	0.5513	0.6186	0.6186	0.7194	0.7194	0.7135	0.7135	0.7596	0.7596	0.8890	0.8890	0.8890	0.8890	0.8153	0.8153	0.8626	0.8626
G019	0.5513	0.6159	0.6351	0.6554	0.6351	0.7008	0.7008	0.7858	0.7858	0.8082	0.8082	0.8890	0.8890	0.9597	0.9597	0.8889	0.8889
G020	0.3998	0.5280	0.5280	0.4665	0.4543	0.6216	0.6774	0.7596	0.7596	0.6351	0.6119	0.8321	0.8321	0.7353	0.7353	0.9190	0.9190
G021	0.1379	0.2534	0.3199	0.2491	0.2781	0.3350	0.3308	0.5025	0.5025	0.6276	0.6276	0.6276	0.6276	0.8056	0.8056	0.8624	0.8081
G022	0.1716	0.1873	0.2305	0.1591	0.1577	0.2130	0.1730	0.3117	0.2793	0.6889	0.6889	0.7789	0.7789	0.8585	0.8585	0.8071	0.8071
G023	0.7194	0.7596	0.7596	0.8160	0.8160	0.7817	0.7817	0.8167	0.8167	0.9294	0.9294	0.9192	0.9192	0.9192	0.8625	0.9296	0.9296
G024	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G025	0.1045	0.1672	0.1800	0.0140	0.1812	0.0593	0.0304	0.3665	0.3450	0.5335	0.5335	0.6549	0.6549	0.8082	0.8082	0.7657	0.7657
G026	0.3733	0.3247	0.3434	0.6928	0.6527	0.6014	0.6014	0.7353	0.7353	0.7465	0.7390	0.8082	0.8082	0.7761	0.7761	0.7862	0.7862
G027	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G028	0.0574	0.1215	0.1790	0.1133	0.1658	0.0025	0.0858	0.3351	0.0619	0.6351	0.5557	0.5358	0.5358	0.6533	0.6533	0.7755	0.7755
G029	0.8170	0.8170	0.8170	0.8578	0.8578	0.8798	0.8798	0.8798	0.8798	0.9599	0.9599	0.8890	0.8890	0.9297	0.9297	0.8890	0.8890
G030	0.1391	0.1899	0.2222	0.2290	0.2202	0.4018	0.3433	0.7015	0.6348	0.5232	0.4793	0.7150	0.7150	0.7277	0.7277	0.8083	0.8083
G031	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G032	0.0290	0.0046	0.0133	0.0011	0.0237	0.1189	0.0142	0.0042	0.0006	0.1651	0.1270	0.3901	0.4071	0.2995	0.4864	0.7758	0.7758
G033	0.0960	0.1313	0.1251	0.1992	0.2200	0.4053	0.3536	0.3822	0.3822	0.3556	0.3934	0.7135	0.7135	0.7389	0.7389	0.8321	0.8321
G034	0.1017	0.3822	0.3777	0.4851	0.3543	0.5806	0.4998	0.5157	0.4102	0.5487	0.5513	0.6060	0.6060	0.8785	0.8785	0.8626	0.8626
G035	0.9191	0.9191	0.9191	0.9599	0.9599	0.9191	0.9191	0.9599	0.9599	0.9599	0.9599	0.9599	0.9599	1.0000	1.0000	1.0000	1.0000
G036	0.3317	0.4429	0.4174	0.4864	0.5023	0.5023	0.5023	0.5909	0.5909	0.7595	0.7595	0.7357	0.7357	0.7649	0.8081	0.7151	0.7151
G037	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.9596	0.9596	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G038	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G039	0.3448	0.3859	0.4126	0.4839	0.4416	0.5316	0.5316	0.5648	0.5648	0.6919	0.7589	0.6595	0.7011	0.7193	0.7193	0.7810	0.7810
G040	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G041	0.2420	0.3215	0.3215	0.2900	0.1864	0.5157	0.5672	0.5157	0.5157	0.6717	0.6717	0.7805	0.7589	0.6940	0.6940	0.7135	0.7135
G042	1.0000	0.9600	0.9600	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G043	0.1959	0.2900	0.2518	0.3891	0.4563	0.3442	0.3550	0.4418	0.4418	0.5090	0.5090	0.6842	0.6842	0.8565	0.8565	0.8583	0.8583
G044	0.5019	0.4502	0.4887	0.3247	0.3745	0.4921	0.4791	0.6308	0.6308	0.6677	0.6677	0.8890	0.8890	0.8890	0.8890	0.8082	0.8082
G045	0.0648	0.1375	0.0746	0.2087	0.1076	0.2404	0.2404	0.2695	0.2642	0.5853	0.6010	0.5511	0.5511	0.6774	0.6774	0.7353	0.7353
G046	0.9032	0.9190	0.9190	0.9297	0.9297	0.8627	0.8627	1.0000	1.0000	0.8052	0.8052	0.9190	0.9190	0.9299	0.9299	0.9032	0.9032
G047	0.0899	0.1309	0.0869	0.1901	0.3009	0.2321	0.1733	0.5005	0.4316	0.3474	0.4430	0.7389	0.7389	0.7817	0.7817	0.6957	0.6957
G048	0.4469	0.4864	0.4864	0.5801	0.6176	0.5216	0.5094	0.7863	0.7391	0.8565	0.8565	0.8083	0.8083	0.7597	0.7597	0.8890	0.8890
G049	0.4836	0.6209	0.6052	0.5909	0.5909	0.6174	0.6174	0.7349	0.7147	0.7565	0.7565	0.7565	0.7810	0.8627	0.7962	0.9036	0.9036
G050	0.7656	0.8165	0.7862	0.7596	0.7596	0.8056	0.8056	0.7817	0.7817	0.9295	0.9293	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Average	**0.4808**	**0.5136**	**0.5160**	**0.5497**	**0.5536**	**0.5880**	**0.5894**	**0.6676**	**0.6549**	**0.7260**	**0.7210**	**0.7974**	**0.7978**	**0.8328**	**0.8346**	**0.8616**	**0.8623**

Table B.1: NMI on mixtures of Gaussians ($K=2$ and $p=0.8$).

Dataset	$m = 0$	$m = 100$		$m = 200$		$m = 300$		$m = 400$		$m = 500$		$m = 600$		$m = 700$		$m = 800$	
Priors:		×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓
G001	0.9191	0.8625	0.8625	0.9598	0.9598	1.0000	1.0000	0.9598	0.9598	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G002	0.1171	0.2041	0.2138	0.2974	0.2871	0.3293	0.3293	0.8318	0.8318	0.9600	0.9600	0.9599	0.9599	0.9038	0.9038	0.9300	0.9300
G003	0.0047	0.2051	0.1223	0.0003	0.0659	0.5978	0.6533	0.6931	0.6737	0.7861	0.7861	0.8889	0.8889	0.9192	0.9192	1.0000	1.0000
G004	0.3348	0.5023	0.4429	0.6077	0.6437	0.7388	0.7388	0.8165	0.8165	0.8624	0.8624	0.9597	0.9597	1.0000	1.0000	0.9597	0.9597
G005	0.7568	0.8167	0.8167	0.8048	0.8318	0.9035	0.9035	0.8582	0.8582	0.8889	0.8889	0.8889	0.8889	1.0000	1.0000	0.9599	0.9599
G006	0.0799	0.1687	0.1763	0.3676	0.4021	0.5755	0.6116	0.7354	0.7354	0.8583	0.8583	0.8321	0.8321	0.9599	0.9599	0.9598	0.9598
G007	0.2400	0.4300	0.3905	0.5514	0.5486	0.6771	0.6771	0.7388	0.7388	0.7596	0.7596	0.9291	0.9291	0.8889	0.8625	0.9291	0.9291
G008	0.5288	0.5972	0.6338	0.7559	0.7559	0.9300	0.9300	0.9600	0.9600	0.8888	0.8888	0.9301	0.9301	1.0000	1.0000	1.0000	1.0000
G009	0.1548	0.0023	0.0010	0.3508	0.4520	0.6527	0.6703	0.7375	0.7651	0.9302	0.9302	0.8302	0.8560	0.9302	0.9302	0.9302	0.9302
G010	0.1029	0.1203	0.1381	0.3123	0.3406	0.4469	0.4162	0.7957	0.8321	0.8321	0.8625	0.8056	0.8056	0.9596	0.9596	1.0000	1.0000
G011	0.8321	0.8889	0.8889	0.8889	0.8889	1.0000	1.0000	0.9597	0.9597	0.9192	0.9192	1.0000	1.0000	1.0000	1.0000	0.9596	0.9596
G012	0.4037	0.4444	0.5901	0.6383	0.6383	0.7950	0.7950	0.8082	0.8082	0.8890	0.8890	0.8890	0.8890	0.9032	0.9032	1.0000	1.0000
G013	0.4239	0.5672	0.4824	0.6510	0.6510	0.7175	0.7175	0.9600	0.9600	0.9600	0.9600	0.9600	0.9600	0.8805	0.8805	1.0000	1.0000
G014	0.0492	0.0264	0.0291	0.1995	0.1967	0.0062	0.1819	0.7150	0.6355	0.7596	0.7596	0.9295	0.9295	1.0000	1.0000	0.9597	0.9597
G015	0.9297	1.0000	0.9598	1.0000	1.0000	0.9295	0.9295	1.0000	1.0000	0.9598	0.9598	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G016	0.8056	0.8056	0.8056	0.9295	0.9295	0.9597	0.9597	1.0000	1.0000	0.9293	0.9293	0.9597	0.9597	1.0000	1.0000	1.0000	1.0000
G017	0.8158	0.8889	0.8889	0.9597	0.9597	0.9597	0.9597	0.9597	0.9597	0.9597	0.9597	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G018	0.5513	0.6504	0.7063	0.7597	0.7198	0.7817	0.7817	0.8890	0.8890	1.0000	1.0000	0.9598	0.9598	0.9294	0.9294	1.0000	1.0000
G019	0.5513	0.6930	0.6930	0.7197	0.7014	0.8382	0.8382	0.9295	0.9295	0.8321	0.8321	0.9192	0.9192	0.9597	0.9597	1.0000	1.0000
G020	0.3998	0.5280	0.4921	0.7098	0.7098	0.6958	0.6737	0.8582	0.8582	0.8582	0.8582	0.9297	0.9297	1.0000	1.0000	1.0000	1.0000
G021	0.1379	0.3786	0.4054	0.5643	0.5907	0.8081	0.8081	0.7150	0.8359	0.8056	0.8056	0.8624	0.8624	0.9596	0.9596	0.9596	0.9596
G022	0.1716	0.2227	0.2213	0.3384	0.3870	0.5409	0.5278	0.7109	0.7109	0.9179	0.9179	0.9179	0.9179	0.9600	0.9600	0.9041	0.9041
G023	0.7194	0.8055	0.7859	0.7859	0.7859	0.8384	0.8890	0.9192	0.9192	1.0000	1.0000	0.9192	0.9192	0.9597	0.9597	1.0000	1.0000
G024	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G025	0.1045	0.1800	0.2370	0.3602	0.4153	0.7231	0.7231	0.6713	0.6713	0.8318	0.8318	0.7657	0.7657	0.9599	0.9599	0.9599	0.9599
G026	0.3733	0.4164	0.4164	0.8389	0.8627	0.7814	0.7814	0.9036	0.9036	0.9599	0.9599	0.9032	0.9032	1.0000	1.0000	0.9190	0.9190
G027	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G028	0.0574	0.1898	0.1653	0.2618	0.3406	0.5513	0.6340	0.6615	0.7649	0.9192	0.9192	0.9597	0.9597	0.9597	0.9597	0.9293	0.9293
G029	0.8170	0.8389	0.8389	0.8890	0.8890	0.9599	0.9599	1.0000	1.0000	0.9599	0.9599	0.9599	0.9599	1.0000	1.0000	0.9297	0.9297
G030	0.1391	0.1756	0.2475	0.3770	0.5232	0.4852	0.5157	0.8890	0.8890	0.7813	0.7813	0.8627	0.8627	0.9298	0.9298	0.9599	0.9599
G031	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G032	0.0290	0.0267	0.0133	0.1330	0.0197	0.1733	0.2783	0.3210	0.4346	0.7231	0.6215	0.9192	0.9192	0.9192	0.9192	0.9596	0.9596
G033	0.0960	0.1929	0.2228	0.4344	0.4344	0.4565	0.4290	0.7391	0.7391	0.7817	0.7817	0.9296	0.9296	0.9192	0.9192	1.0000	1.0000
G034	0.1017	0.4910	0.4844	0.4423	0.5336	0.8167	0.7576	0.8160	0.8160	0.8890	0.8890	0.8890	0.8890	0.9597	0.9597	0.9296	0.9296
G035	0.9191	0.9191	0.9191	0.9599	0.9599	1.0000	1.0000	1.0000	1.0000	0.9599	0.9599	0.9599	0.9599	1.0000	1.0000	1.0000	1.0000
G036	0.3317	0.4726	0.4594	0.6930	0.6930	0.7357	0.7357	0.7462	0.7462	0.8624	0.8624	0.9295	0.9295	0.9597	0.9597	0.9597	0.9597
G037	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G038	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G039	0.3448	0.4441	0.4441	0.6595	0.6430	0.6519	0.6519	0.7193	0.7385	0.8317	0.8317	0.8389	0.8389	1.0000	1.0000	1.0000	1.0000
G040	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G041	0.2420	0.2697	0.2657	0.5623	0.5623	0.6365	0.6365	0.8624	0.8624	0.8626	0.8314	1.0000	1.0000	0.9301	0.9301	0.9302	0.9302
G042	1.0000	0.9600	0.9600	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G043	0.1959	0.2491	0.2582	0.4500	0.4046	0.6350	0.6350	0.8083	0.8083	0.8169	0.8169	0.8797	0.8797	1.0000	1.0000	0.9598	0.9598
G044	0.5019	0.4727	0.5464	0.7130	0.6178	0.6529	0.6529	0.9297	0.9297	0.8890	0.8890	0.9599	0.9599	1.0000	1.0000	1.0000	1.0000
G045	0.0648	0.1509	0.1591	0.2535	0.2622	0.6556	0.6746	0.7465	0.7595	0.8627	0.8627	0.9190	0.9190	0.8320	0.8320	0.8582	0.8890
G046	0.9032	0.9599	0.9599	0.9297	0.9297	0.8582	0.8582	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.9599	0.9599
G047	0.0899	0.1125	0.1125	0.2307	0.2582	0.6014	0.5643	0.8384	0.8384	0.7767	0.7391	1.0000	1.0000	0.9192	0.9192	0.9027	0.9027
G048	0.4469	0.5755	0.5557	0.6556	0.6556	0.7816	0.7597	0.9298	0.9298	1.0000	1.0000	1.0000	1.0000	0.9599	0.9599	1.0000	1.0000
G049	0.4836	0.5845	0.5845	0.7198	0.6956	0.9600	0.9600	0.9600	0.9600	0.9600	0.9600	0.9600	0.9600	0.9600	0.9600	1.0000	1.0000
G050	0.7656	0.8165	0.8165	0.8585	0.8585	0.8890	0.8890	0.9192	0.9597	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Average	**0.4808**	**0.5461**	**0.5483**	**0.6515**	**0.6601**	**0.7546**	**0.7618**	**0.8603**	**0.8678**	**0.9045**	**0.9017**	**0.9381**	**0.9387**	**0.9664**	**0.9659**	**0.9722**	**0.9728**

Table B.2: NMI on mixtures of Gaussians ($K = 2$ and $p = 0.9$).

Dataset	$m = 0$	$m = 100$		$m = 200$		$m = 300$		$m = 400$		$m = 500$		$m = 600$		$m = 700$		$m = 800$	
Priors:		×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓
G001	0.9191	0.9191	0.9598	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G002	0.1171	0.2026	0.1561	0.6150	0.3871	0.9600	0.8889	0.9599	0.9599	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G003	0.0047	0.0514	0.2429	0.4291	0.4469	0.7651	0.7651	0.8889	0.8889	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G004	0.3348	0.6278	0.5440	0.7595	0.8321	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G005	0.7568	0.8364	0.8364	0.9599	0.9599	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G006	0.0799	0.3576	0.2454	0.6958	0.5315	0.8321	0.8321	0.9599	0.9599	0.9191	0.9191	1.0000	1.0000	0.9599	0.9599	1.0000	1.0000
G007	0.2400	0.6598	0.5155	0.8568	0.7596	0.9597	0.9597	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G008	0.5288	0.8314	0.8314	0.8314	0.8314	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G009	0.1548	0.4682	0.4203	0.8560	0.7448	0.9177	0.9177	0.9177	0.9177	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G010	0.1029	0.2491	0.1446	0.6217	0.4596	0.9294	0.6931	0.9597	0.9597	1.0000	1.0000	0.9596	0.9596	1.0000	1.0000	1.0000	1.0000
G011	0.8321	0.9596	0.9596	0.9596	0.9596	0.9596	0.9596	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G012	0.4037	0.6555	0.4194	0.8321	0.4904	0.9296	0.9296	0.9597	0.9597	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G013	0.4239	0.6510	0.5048	0.8381	0.7095	0.9599	0.9599	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G014	0.0492	0.2427	0.1594	0.3475	0.2988	0.9597	0.9192	0.8890	0.8890	1.0000	1.0000	1.0000	1.0000	0.9597	0.9597	1.0000	1.0000
G015	0.9297	0.9598	0.9598	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G016	0.8056	0.9295	0.9295	0.9597	0.9597	1.0000	1.0000	1.0000	1.0000	0.9597	0.9597	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G017	0.8158	0.9597	0.9597	0.9026	0.9026	0.9597	0.9597	0.9597	0.9597	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G018	0.5513	0.8349	0.6052	0.8890	0.7597	0.9296	0.9296	0.9597	0.9597	0.9597	0.9597	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G019	0.5513	0.7357	0.5702	0.9295	0.9295	0.9597	0.9597	1.0000	1.0000	0.9597	0.9597	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G020	0.3998	0.6837	0.6155	0.8791	0.8165	0.9190	0.9190	1.0000	1.0000	1.0000	1.0000	0.9599	0.9599	1.0000	1.0000	1.0000	1.0000
G021	0.1379	0.3917	0.1220	0.7816	0.7816	0.8384	0.8384	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G022	0.1716	0.3359	0.1513	0.6676	0.6745	0.9600	0.9600	0.9179	0.9179	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G023	0.7194	0.8585	0.8890	0.8890	0.8890	0.9192	0.9192	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G024	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G025	0.1045	0.3294	0.1944	0.5335	0.5848	0.8580	0.8580	0.9599	0.9599	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G026	0.3733	0.5160	0.4197	1.0000	0.8320	0.9599	0.9599	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G027	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G028	0.0574	0.2873	0.1924	0.5643	0.4430	0.9293	0.9293	0.9597	0.9597	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G029	0.8170	0.8582	0.7814	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G030	0.1391	0.4014	0.3327	0.7147	0.6716	0.8890	0.8890	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G031	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G032	0.0290	0.1330	0.0004	0.2189	0.2752	0.7858	0.7763	0.9192	0.9192	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G033	0.0960	0.4316	0.1661	0.6532	0.3307	0.8384	0.8384	0.9597	0.9597	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G034	0.1017	0.6014	0.3486	0.7862	0.7584	0.9296	0.9296	0.9294	0.9294	0.9598	0.9598	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G035	0.9191	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G036	0.3317	0.6666	0.6666	0.8158	0.7008	0.9026	0.9026	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G037	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G038	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G039	0.3448	0.4356	0.4006	0.7344	0.7141	0.9600	0.9600	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G040	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G041	0.2420	0.4340	0.3840	0.8079	0.6951	0.9600	0.9600	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G042	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G043	0.1959	0.4596	0.1408	0.6381	0.5230	0.9298	0.8053	0.9298	0.9298	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G044	0.5019	0.5666	0.1365	0.7863	0.7465	0.9190	0.9190	0.9599	0.9599	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G045	0.0648	0.2078	0.1178	0.6928	0.4702	0.9032	0.9297	0.9297	0.9297	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G046	0.9032	0.9599	0.7972	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G047	0.0899	0.2814	0.1135	0.6269	0.5193	0.7389	0.7389	0.8890	0.8890	0.9598	0.9598	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G048	0.4469	0.6738	0.5415	0.9191	0.7963	0.8890	0.8890	1.0000	1.0000	0.9599	0.9599	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G049	0.4836	0.7195	0.5706	0.8170	0.8170	1.0000	1.0000	0.9600	0.9600	1.0000	1.0000	1.0000	1.0000	0.9600	0.9600	1.0000	1.0000
G050	0.7656	0.8570	0.7462	0.8890	0.8890	0.9597	0.9597	0.9597	0.9597	1.0000	1.0000	0.9597	0.9597	1.0000	1.0000	1.0000	1.0000
Average	**0.4808**	**0.6444**	**0.5559**	**0.8140**	**0.7578**	**0.9402**	**0.9311**	**0.9746**	**0.9746**	**0.9936**	**0.9936**	**0.9976**	**0.9976**	**0.9976**	**0.9976**	**1.0000**	**1.0000**

Table B.3: NMI on mixtures of Gaussians ($K = 2$ and $p = 1.0$).

Dataset	$m = 0$	$m = 100$		$m = 200$		$m = 300$		$m = 400$		$m = 500$		$m = 600$		$m = 700$		$m = 800$	
Priors:		×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓
G001	0.4144	0.4181	0.4109	0.4205	0.4637	0.4026	0.4822	0.3924	0.4037	0.4127	0.4230	0.4185	0.4375	0.6057	0.6746	0.4860	0.4859
G002	0.2457	0.2589	0.2626	0.2381	0.2672	0.2416	0.2295	0.2799	0.2841	0.2850	0.2738	0.3226	0.3157	0.3373	0.3211	0.3625	0.2622
G003	0.1126	0.1183	0.1471	0.1296	0.1311	0.1119	0.1013	0.1910	0.1452	0.1403	0.0758	0.1227	0.2016	0.2176	0.3529	0.1520	0.2787
G004	0.2148	0.1698	0.1530	0.1952	0.2295	0.2064	0.1460	0.2051	0.2401	0.2542	0.2264	0.2402	0.2265	0.2201	0.2824	0.2548	0.3225
G005	0.5446	0.4190	0.3872	0.5128	0.4548	0.4133	0.4655	0.5253	0.5101	0.5308	0.5312	0.6213	0.5881	0.7034	0.7034	0.6850	0.6597
G006	0.3409	0.3335	0.2956	0.3753	0.3305	0.3455	0.3583	0.3581	0.3922	0.4281	0.3864	0.3509	0.3724	0.4412	0.5021	0.4575	0.4543
G007	0.2050	0.1613	0.2227	0.2291	0.2356	0.2494	0.1806	0.2816	0.2577	0.2107	0.1666	0.1946	0.2786	0.3816	0.3337	0.3904	0.3959
G008	0.7044	0.7451	0.7101	0.7543	0.7476	0.6944	0.7023	0.7623	0.6740	0.7420	0.7516	0.7913	0.7918	0.7378	0.7236	0.8577	0.8577
G009	0.2826	0.2923	0.2683	0.1365	0.1219	0.2247	0.2027	0.1967	0.2395	0.2925	0.2877	0.3165	0.3455	0.3902	0.3973	0.3392	0.4172
G010	0.0732	0.0649	0.0660	0.0871	0.1205	0.0638	0.0472	0.0724	0.0595	0.0336	0.1332	0.0652	0.1034	0.0852	0.1283	0.0936	0.1073
G011	0.4330	0.3863	0.5123	0.4237	0.4398	0.4431	0.5446	0.5663	0.5589	0.5440	0.5475	0.5864	0.5845	0.6026	0.5909	0.6031	0.6100
G012	0.3975	0.3983	0.4183	0.3659	0.3966	0.4026	0.4123	0.4848	0.4930	0.3916	0.5241	0.4374	0.4999	0.4877	0.5239	0.5252	0.5195
G013	0.4421	0.5308	0.5467	0.5899	0.5897	0.4302	0.4239	0.5477	0.5863	0.5556	0.5842	0.5675	0.5197	0.5598	0.5598	0.7577	0.7577
G014	0.1979	0.1615	0.1756	0.2297	0.2326	0.1778	0.1974	0.2379	0.2382	0.1851	0.2097	0.3064	0.2866	0.2950	0.2811	0.3044	0.2914
G015	0.5286	0.4953	0.5021	0.4899	0.4910	0.4073	0.4769	0.5232	0.5191	0.6925	0.6262	0.5300	0.5311	0.5933	0.6359	0.6085	0.5528
G016	0.5404	0.5292	0.5763	0.5234	0.5158	0.6591	0.6508	0.6082	0.5957	0.5746	0.5706	0.6241	0.6141	0.6167	0.6167	0.7175	0.7141
G017	0.5044	0.4494	0.4866	0.4564	0.4263	0.4926	0.5212	0.5356	0.5118	0.5298	0.5309	0.5853	0.5870	0.5359	0.5721	0.5862	0.6032
G018	0.6567	0.6787	0.6787	0.6659	0.6703	0.7331	0.7331	0.7129	0.7129	0.7775	0.7775	0.7828	0.7828	0.7676	0.7744	0.8375	0.8375
G019	0.5739	0.5682	0.5639	0.5562	0.5610	0.6387	0.6110	0.5860	0.6160	0.6341	0.6341	0.6983	0.6892	0.7087	0.7087	0.7163	0.6843
G020	0.3986	0.4713	0.5079	0.4120	0.4418	0.4089	0.4169	0.4814	0.4658	0.4400	0.4368	0.6413	0.6131	0.5484	0.5484	0.5878	0.5878
G021	0.1462	0.1078	0.1337	0.1373	0.1559	0.1092	0.1317	0.0770	0.1070	0.2264	0.2342	0.1774	0.0736	0.2254	0.1900	0.2251	0.2313
G022	0.5075	0.5013	0.4963	0.4777	0.4706	0.4830	0.5301	0.4884	0.4966	0.5893	0.5599	0.5758	0.6407	0.6373	0.6363	0.5490	0.5778
G023	0.7202	0.7717	0.7770	0.7212	0.8092	0.7794	0.7856	0.7784	0.7733	0.7826	0.7492	0.8114	0.7560	0.7856	0.8710	0.8583	0.8583
G024	0.4620	0.4651	0.4782	0.4685	0.4472	0.3771	0.3558	0.5362	0.5434	0.5357	0.5960	0.5165	0.5165	0.5117	0.5125	0.5735	0.5346
G025	0.1199	0.1215	0.1072	0.1329	0.0973	0.1674	0.1767	0.1560	0.1270	0.2361	0.1968	0.1733	0.1472	0.2036	0.1504	0.1782	0.2112
G026	0.2123	0.1886	0.2092	0.2499	0.2575	0.2043	0.1740	0.1532	0.1842	0.2814	0.2801	0.2858	0.2803	0.3877	0.2896	0.2912	0.2663
G027	0.8737	0.8649	0.8649	0.8664	0.8732	0.9089	0.9089	0.9217	0.9217	0.9067	0.9067	0.9177	0.9162	0.9226	0.9226	0.8981	0.8981
G028	0.2158	0.2040	0.1973	0.2146	0.2371	0.1231	0.1532	0.2645	0.2474	0.1841	0.2123	0.1873	0.2042	0.3956	0.3703	0.3499	0.3301
G029	0.5848	0.5630	0.5530	0.5575	0.5575	0.6035	0.5949	0.7606	0.7606	0.6658	0.6706	0.6938	0.7002	0.7214	0.7214	0.7186	0.6974
G030	0.1211	0.1095	0.1088	0.0486	0.2152	0.0887	0.0986	0.1337	0.1408	0.1199	0.1043	0.1536	0.1021	0.1460	0.1466	0.0839	0.1196
G031	0.8008	0.7371	0.7371	0.8222	0.8222	0.8449	0.8384	0.8309	0.8240	0.8722	0.8721	0.9194	0.9194	0.8664	0.8664	0.8498	0.8972
G032	0.4488	0.3955	0.3930	0.4012	0.3940	0.4201	0.4369	0.4430	0.4428	0.4239	0.4712	0.4794	0.4788	0.5069	0.5225	0.4706	0.5292
G033	0.2066	0.1693	0.1845	0.1900	0.2029	0.1417	0.1819	0.2469	0.2361	0.2131	0.2170	0.2304	0.3277	0.2402	0.3054	0.2025	0.3352
G034	0.2876	0.2631	0.3352	0.3088	0.3102	0.3272	0.3263	0.3409	0.3422	0.3996	0.3526	0.3328	0.3300	0.4115	0.4491	0.3929	0.3775
G035	0.8777	0.8839	0.8839	0.8911	0.8911	0.8768	0.8644	0.8832	0.8832	0.9178	0.9178	0.9119	0.9119	0.8861	0.8768	0.8656	0.8540
G036	0.4761	0.4796	0.5135	0.5110	0.5273	0.5766	0.5603	0.5731	0.5281	0.6047	0.5097	0.6658	0.6428	0.7104	0.7086	0.7986	0.7986
G037	0.4246	0.4677	0.3667	0.3881	0.3955	0.3790	0.3668	0.4153	0.4515	0.4281	0.4066	0.5291	0.5143	0.4641	0.5314	0.6312	0.5518
G038	0.7008	0.7269	0.7182	0.7138	0.7153	0.6973	0.7227	0.7707	0.7742	0.7598	0.7645	0.7415	0.7671	0.7930	0.7893	0.8922	0.8922
G039	0.1636	0.1940	0.1927	0.1567	0.2276	0.2031	0.2179	0.2603	0.2569	0.3283	0.3004	0.2457	0.1843	0.2770	0.2444	0.2799	0.2601
G040	0.8282	0.8146	0.8224	0.7841	0.7841	0.8451	0.8451	0.8747	0.8647	0.8621	0.8621	0.8619	0.8619	0.8556	0.8428	0.8624	0.8624
G041	0.3310	0.3736	0.3419	0.3292	0.3483	0.3380	0.3777	0.3234	0.4074	0.4556	0.4039	0.3608	0.3805	0.3753	0.3886	0.4281	0.3359
G042	0.7381	0.7458	0.7461	0.7458	0.7266	0.7371	0.7649	0.7798	0.7749	0.7881	0.7948	0.7382	0.7664	0.8148	0.8045	0.7337	0.7298
G043	0.6354	0.6521	0.6521	0.6831	0.6749	0.7080	0.7145	0.5824	0.6120	0.7046	0.7046	0.6892	0.7627	0.6656	0.6656	0.7835	0.7835
G044	0.1132	0.1119	0.1308	0.1055	0.1355	0.1455	0.1720	0.0963	0.1459	0.0690	0.1255	0.1740	0.1186	0.1653	0.1717	0.1500	0.1757
G045	0.4977	0.4933	0.4943	0.4706	0.5400	0.4550	0.4716	0.4854	0.4552	0.5376	0.5406	0.6238	0.5763	0.6456	0.6358	0.4801	0.6654
G046	0.6418	0.6562	0.6550	0.6654	0.6627	0.6720	0.6688	0.6464	0.6464	0.6885	0.6872	0.6605	0.6751	0.6956	0.7860	0.7194	0.7733
G047	0.4763	0.4749	0.4769	0.4943	0.4690	0.4995	0.5063	0.5587	0.4990	0.5215	0.5055	0.5037	0.5341	0.5080	0.5156	0.5260	0.6360
G048	0.4873	0.5219	0.5382	0.4034	0.4786	0.5586	0.5771	0.6337	0.6614	0.6478	0.6175	0.6171	0.6524	0.6898	0.6389	0.6622	0.6831
G049	0.3532	0.3821	0.3550	0.3356	0.3973	0.3421	0.4169	0.3368	0.3647	0.4639	0.4639	0.3777	0.4137	0.4549	0.4694	0.4152	0.4617
G050	0.5260	0.5235	0.5105	0.5283	0.4877	0.4861	0.4964	0.5333	0.5098	0.6082	0.6005	0.6111	0.6246	0.6812	0.6197	0.6546	0.6692
Average	**0.4358**	**0.4323**	**0.4373**	**0.4319**	**0.4436**	**0.4369**	**0.4468**	**0.4687**	**0.4697**	**0.4895**	**0.4865**	**0.4994**	**0.5030**	**0.5296**	**0.5375**	**0.5369**	**0.5479**

Table B.4: NMI on mixtures of Gaussians ($K = 4$ and $p = 0.8$).

Dataset	$m=0$	$m=100$		$m=200$		$m=300$		$m=400$		$m=500$		$m=600$		$m=700$		$m=800$	
Priors:		×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓
G001	0.4144	0.4003	0.4128	0.4363	0.4870	0.4550	0.5308	0.4495	0.4742	0.6045	0.5674	0.6697	0.6927	0.8591	0.8816	0.9251	0.9529
G002	0.2457	0.2729	0.2837	0.2540	0.3401	0.3531	0.3583	0.3648	0.3995	0.4434	0.5370	0.3924	0.5181	0.7535	0.7535	0.6634	0.6634
G003	0.1126	0.0908	0.2454	0.0851	0.1612	0.0958	0.2023	0.2235	0.2751	0.2297	0.1902	0.2802	0.3762	0.5712	0.6512	0.6402	0.5610
G004	0.2148	0.2011	0.1785	0.3021	0.3329	0.2767	0.2970	0.2972	0.3193	0.4116	0.4670	0.5895	0.6483	0.7885	0.7885	0.4400	0.6011
G005	0.5446	0.3966	0.5269	0.5703	0.5958	0.6211	0.6256	0.6027	0.6546	0.6313	0.6523	0.6843	0.7303	0.7964	0.8058	0.8892	0.8892
G006	0.3409	0.3596	0.3816	0.3213	0.3603	0.3859	0.3515	0.3434	0.5245	0.4373	0.5407	0.5972	0.6676	0.6472	0.5958	0.7429	0.7571
G007	0.2050	0.1698	0.1650	0.2118	0.2709	0.2498	0.2503	0.3521	0.3823	0.3482	0.2589	0.3310	0.4118	0.4313	0.6389	0.4630	0.4728
G008	0.7044	0.7603	0.7390	0.7917	0.7938	0.7810	0.7509	0.8093	0.8357	0.8271	0.8271	0.8948	0.8948	0.8866	0.8684	0.9467	0.9467
G009	0.2826	0.2260	0.2673	0.3098	0.1458	0.4216	0.4342	0.3874	0.3322	0.5963	0.5788	0.4768	0.4958	0.5566	0.6339	0.6789	0.6484
G010	0.0732	0.0513	0.0577	0.0913	0.0886	0.0725	0.1437	0.1287	0.0626	0.0987	0.1238	0.1115	0.1349	0.1161	0.1604	0.1378	0.1414
G011	0.4330	0.5099	0.5489	0.4929	0.5134	0.5882	0.6187	0.7077	0.6687	0.6991	0.6693	0.7549	0.7617	0.7315	0.8479	0.8609	0.8609
G012	0.3975	0.4507	0.4430	0.4087	0.4856	0.6088	0.5808	0.5070	0.5679	0.6682	0.6622	0.7687	0.7657	0.8147	0.7525	0.8682	0.8682
G013	0.4421	0.5127	0.5138	0.6092	0.6127	0.6043	0.6036	0.7437	0.6977	0.7355	0.7105	0.7568	0.7568	0.7557	0.7626	0.8823	0.8823
G014	0.1979	0.1787	0.2003	0.2271	0.2566	0.1852	0.2368	0.3305	0.3151	0.3138	0.3493	0.5077	0.4972	0.5512	0.6660	0.6705	0.6586
G015	0.5286	0.4992	0.5811	0.4895	0.4878	0.5020	0.5472	0.6345	0.6170	0.6953	0.7586	0.6313	0.6387	0.7073	0.7732	0.8244	0.8232
G016	0.5404	0.5795	0.6089	0.5319	0.5345	0.6859	0.6859	0.6121	0.6245	0.6789	0.6989	0.7309	0.7478	0.7727	0.8064	0.8618	0.9402
G017	0.5044	0.5560	0.4999	0.4915	0.5223	0.5679	0.5595	0.6105	0.6328	0.7767	0.8232	0.7602	0.7170	0.7981	0.7813	0.8382	0.8204
G018	0.6567	0.6872	0.6979	0.7161	0.7258	0.7565	0.7799	0.7910	0.7661	0.8639	0.8639	0.8354	0.8354	0.8882	0.8882	0.9418	0.9418
G019	0.5739	0.6085	0.5566	0.6096	0.6511	0.6598	0.6598	0.7466	0.7587	0.7452	0.7393	0.8547	0.8547	0.8055	0.7859	0.9056	0.9056
G020	0.3986	0.5115	0.4873	0.5250	0.5438	0.5362	0.6006	0.6281	0.6281	0.6182	0.6382	0.7249	0.7249	0.7302	0.7302	0.7768	0.7768
G021	0.1462	0.1583	0.1787	0.1665	0.1618	0.2066	0.2499	0.1517	0.2354	0.2853	0.2713	0.2484	0.3189	0.3785	0.5355	0.3902	0.4638
G022	0.5075	0.5051	0.5132	0.4991	0.4999	0.4564	0.5299	0.5041	0.5146	0.6145	0.7115	0.5827	0.6110	0.6920	0.7684	0.5880	0.7799
G023	0.7202	0.7590	0.7590	0.8500	0.8599	0.8987	0.8901	0.8328	0.8328	0.8550	0.8592	0.9290	0.9490	0.9005	0.8882	0.8645	0.8645
G024	0.4620	0.4751	0.4403	0.4584	0.5099	0.5074	0.5224	0.5961	0.6096	0.6804	0.6367	0.6095	0.5997	0.8383	0.7889	0.8055	0.8427
G025	0.1199	0.1505	0.1359	0.1385	0.1645	0.2046	0.2288	0.2144	0.2204	0.2811	0.2889	0.2046	0.2902	0.3387	0.5189	0.5725	0.5831
G026	0.2123	0.2079	0.1993	0.2258	0.2573	0.2321	0.2729	0.2787	0.2336	0.4131	0.3539	0.3651	0.3544	0.4424	0.5442	0.5670	0.6402
G027	0.8737	0.8673	0.8673	0.8883	0.8883	0.9089	0.9089	0.9217	0.9217	0.9405	0.9405	0.8981	0.9067	0.9696	0.9696	0.9513	0.9693
G028	0.2158	0.2664	0.2426	0.2153	0.2139	0.2895	0.2452	0.2241	0.3417	0.4008	0.4353	0.3928	0.5422	0.5497	0.5464	0.6103	0.6529
G029	0.5848	0.5429	0.5462	0.6277	0.6092	0.5678	0.5825	0.8414	0.8414	0.8318	0.8365	0.8440	0.8440	0.8599	0.8599	0.8827	0.8827
G030	0.1211	0.1120	0.1393	0.1001	0.1885	0.1208	0.1568	0.1464	0.1657	0.2230	0.3728	0.3057	0.4044	0.3496	0.3857	0.4606	0.4835
G031	0.8008	0.8133	0.8133	0.8370	0.8370	0.8512	0.8519	0.8668	0.8668	0.9007	0.9007	0.9279	0.9279	0.8969	0.8681	0.9587	0.9587
G032	0.4488	0.4212	0.4269	0.4111	0.4068	0.4488	0.4192	0.4504	0.4391	0.4170	0.3974	0.5998	0.6115	0.5857	0.7058	0.8603	0.8629
G033	0.2066	0.1876	0.1942	0.2864	0.3105	0.1550	0.2349	0.3383	0.3676	0.3745	0.3683	0.5260	0.6025	0.4869	0.4454	0.4934	0.5385
G034	0.2876	0.3173	0.2757	0.3685	0.4410	0.3173	0.3660	0.3792	0.4069	0.5363	0.5477	0.5502	0.5923	0.7138	0.7532	0.8742	0.9012
G035	0.8777	0.9089	0.8991	0.9089	0.9200	0.8861	0.9197	0.9392	0.9219	0.9522	0.9645	0.9522	0.9522	0.9325	0.9325	0.9238	0.9238
G036	0.4761	0.5430	0.5949	0.6014	0.5285	0.6211	0.6388	0.6955	0.7572	0.7668	0.6707	0.8948	0.8948	0.7812	0.7698	0.9350	0.9350
G037	0.4246	0.4405	0.4609	0.4168	0.4422	0.4608	0.4693	0.4316	0.4993	0.5796	0.6278	0.7047	0.7543	0.7627	0.7627	0.7334	0.7800
G038	0.7008	0.7248	0.7391	0.7086	0.7152	0.7590	0.7463	0.8336	0.8336	0.8580	0.8580	0.9099	0.8717	0.9277	0.9277	0.9337	0.9337
G039	0.1636	0.2420	0.3008	0.2099	0.2586	0.3030	0.2963	0.2773	0.3701	0.3647	0.3109	0.3316	0.3613	0.5412	0.4620	0.6219	0.7958
G040	0.8282	0.8129	0.8129	0.8356	0.8549	0.8417	0.8417	0.8866	0.8866	0.9347	0.9709	0.9512	0.9340	0.9509	0.9509	0.9527	0.9527
G041	0.3310	0.3767	0.3987	0.3455	0.3588	0.4322	0.3961	0.5138	0.5089	0.5538	0.5809	0.7017	0.7488	0.6734	0.5099	0.6724	0.6556
G042	0.7381	0.7492	0.7449	0.7500	0.7556	0.7379	0.7575	0.8376	0.8269	0.8067	0.8299	0.8803	0.9229	0.8803	0.8893	0.8819	0.8819
G043	0.6354	0.6336	0.6336	0.6575	0.6575	0.7033	0.7312	0.7293	0.7186	0.7924	0.7924	0.8499	0.8499	0.8907	0.8907	0.9640	0.9511
G044	0.1132	0.1766	0.1135	0.2241	0.1540	0.1258	0.0801	0.0995	0.1471	0.1883	0.1876	0.2787	0.3612	0.5155	0.5233	0.5150	0.6056
G045	0.4977	0.5206	0.5531	0.5131	0.5330	0.6265	0.6099	0.5690	0.5737	0.6759	0.7118	0.7693	0.7664	0.8027	0.8077	0.8431	0.8431
G046	0.6418	0.6658	0.6954	0.6534	0.6466	0.7487	0.7487	0.6690	0.7360	0.7101	0.7365	0.7891	0.8015	0.8173	0.8219	0.9015	0.9015
G047	0.4763	0.4890	0.5025	0.4901	0.4974	0.5233	0.5211	0.4986	0.5310	0.5404	0.5438	0.5498	0.6277	0.7454	0.7300	0.8414	0.8636
G048	0.4873	0.5373	0.5835	0.5843	0.5448	0.6336	0.6507	0.6447	0.6447	0.7906	0.7906	0.7549	0.7628	0.8745	0.8745	0.8463	0.8984
G049	0.3532	0.3799	0.3755	0.3602	0.4790	0.4491	0.4432	0.5323	0.4935	0.5986	0.6033	0.5818	0.5941	0.6472	0.6906	0.7726	0.7968
G050	0.5260	0.5598	0.5009	0.5533	0.5973	0.5639	0.5736	0.6720	0.6604	0.7014	0.6701	0.8424	0.8424	0.8289	0.8179	0.8620	0.8993
Average	**0.4358**	**0.4513**	**0.4607**	**0.4672**	**0.4840**	**0.4998**	**0.5140**	**0.5369**	**0.5529**	**0.5998**	**0.6085**	**0.6416**	**0.6694**	**0.7107**	**0.7302**	**0.7608**	**0.7831**

Table B.5: NMI on mixtures of Gaussians ($K = 4$ and $p = 0.9$).

Dataset	$m = 0$	$m = 100$		$m = 200$		$m = 300$		$m = 400$		$m = 500$		$m = 600$		$m = 700$		$m = 800$	
Priors:		×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓
G001	0.4144	0.4263	0.4960	0.4268	0.5124	0.6269	0.7229	0.7570	0.7867	0.8406	0.8406	0.9353	0.9479	0.9048	0.9048	0.9416	0.9416
G002	0.2457	0.2652	0.2479	0.2783	0.3294	0.4278	0.2637	0.6946	0.6812	0.8098	0.7114	0.8577	0.8577	0.8692	0.8692	0.9656	0.9656
G003	0.1126	0.1795	0.1914	0.2395	0.2731	0.3553	0.2997	0.3684	0.4173	0.6245	0.6989	0.8290	0.9119	0.9475	0.9475	0.9171	0.9171
G004	0.2148	0.2212	0.2390	0.2853	0.2349	0.4678	0.2932	0.6195	0.5148	0.8472	0.7293	0.7983	0.8027	0.9646	0.9646	0.9825	0.9825
G005	0.5446	0.6473	0.5098	0.5541	0.5494	0.7767	0.7299	0.8636	0.7281	0.9520	0.9520	0.9142	0.9142	1.0000	1.0000	0.9698	0.9698
G006	0.3409	0.3766	0.3868	0.4371	0.3783	0.4838	0.3942	0.5471	0.6009	0.7929	0.7570	0.7905	0.8089	0.9467	0.9467	1.0000	1.0000
G007	0.2050	0.2961	0.2562	0.3341	0.2972	0.3774	0.3946	0.4898	0.5075	0.5582	0.6397	0.7742	0.7198	0.7918	0.8090	0.9825	0.9825
G008	0.7044	0.7714	0.7387	0.8390	0.8390	0.8703	0.8609	0.9647	0.9467	0.9647	0.9647	0.9465	0.9465	1.0000	1.0000	0.9826	0.9826
G009	0.2826	0.2364	0.2866	0.3159	0.3756	0.5231	0.4478	0.5425	0.5374	0.7336	0.6546	0.8748	0.9107	0.9232	0.9232	0.9653	0.9653
G010	0.0732	0.0899	0.0810	0.0863	0.1258	0.1461	0.2561	0.1319	0.2034	0.2342	0.3146	0.3273	0.7600	0.3656	0.6171	0.5849	0.8465
G011	0.4330	0.5555	0.5766	0.7295	0.6388	0.7826	0.7178	0.8802	0.8804	0.8985	0.9175	0.9467	0.9467	0.9644	0.9644	0.9466	0.9825
G012	0.3975	0.4144	0.4138	0.5780	0.4543	0.6564	0.5156	0.8238	0.7026	0.8841	0.9145	0.8992	0.8992	1.0000	1.0000	0.9830	0.9830
G013	0.4421	0.5084	0.6028	0.6477	0.5504	0.7038	0.7248	0.8805	0.8759	0.8882	0.8930	0.9525	0.9525	0.9642	0.9828	0.9539	0.9539
G014	0.1979	0.1356	0.2038	0.3033	0.3399	0.4844	0.5179	0.5405	0.5661	0.6766	0.7408	0.8047	0.9293	0.9121	0.9121	0.9163	0.9472
G015	0.5286	0.5147	0.5948	0.5124	0.5152	0.6287	0.5739	0.6936	0.7075	0.9048	0.9048	0.9121	0.9475	0.9178	0.9178	0.9654	0.9654
G016	0.5404	0.6033	0.6104	0.6756	0.6510	0.7411	0.7971	0.7756	0.8829	0.8880	0.9289	0.9472	0.9826	0.9651	0.9651	0.9826	0.9826
G017	0.5044	0.5610	0.5165	0.6286	0.5538	0.7258	0.7280	0.8254	0.7966	0.8805	0.8805	0.9469	0.9469	0.9659	0.9659	1.0000	1.0000
G018	0.6567	0.7017	0.6770	0.7761	0.7775	0.8239	0.8376	0.8598	0.8809	0.8997	0.9521	0.9650	0.9825	0.9823	0.9823	1.0000	1.0000
G019	0.5739	0.6634	0.6750	0.7467	0.7901	0.8233	0.6984	0.8468	0.8755	0.9470	0.9470	0.9830	0.9830	0.9350	0.9350	0.9820	0.9820
G020	0.3986	0.5250	0.4321	0.6271	0.5341	0.7417	0.6918	0.8031	0.8548	0.8692	0.8865	0.9174	0.9174	0.8925	0.9689	1.0000	1.0000
G021	0.1462	0.1174	0.1400	0.2137	0.2162	0.2579	0.2019	0.2456	0.4601	0.5751	0.6528	0.7587	0.7597	0.8766	0.8766	0.9699	0.9699
G022	0.5075	0.4828	0.6009	0.5193	0.6417	0.6303	0.6119	0.6189	0.6298	0.7323	0.7153	0.8921	0.9472	0.9655	0.9836	0.9247	0.9522
G023	0.7202	0.7806	0.8000	0.8482	0.8460	0.8525	0.8584	0.9048	0.9561	0.9545	0.9545	0.9838	0.9838	0.9837	0.9837	0.9838	0.9838
G024	0.4620	0.4451	0.4071	0.5596	0.6523	0.6470	0.6870	0.6813	0.6937	0.8776	0.8950	0.9338	0.9516	0.9654	0.9654	0.9707	0.9707
G025	0.1199	0.1370	0.1395	0.1608	0.2144	0.3404	0.3502	0.3158	0.4293	0.4620	0.6383	0.5686	0.8146	0.7736	0.9401	0.9654	0.9654
G026	0.2123	0.2118	0.1850	0.3329	0.3007	0.3051	0.3787	0.4490	0.4346	0.7586	0.7258	0.8445	0.9047	0.9827	0.9827	0.8817	0.8817
G027	0.8737	0.8968	0.9179	0.9067	0.9067	0.9588	0.9588	0.9513	1.0000	0.9822	0.9822	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
G028	0.2158	0.2159	0.1875	0.2936	0.3208	0.3795	0.5064	0.6499	0.4852	0.6363	0.7370	0.8194	0.8517	0.8886	0.9168	0.9824	0.9824
G029	0.5848	0.6485	0.5424	0.6471	0.6608	0.8211	0.7389	0.9227	0.9227	0.8986	0.9073	0.9830	0.9830	0.9821	0.9821	1.0000	1.0000
G030	0.1211	0.0825	0.0875	0.1508	0.2283	0.2586	0.3076	0.2686	0.3204	0.6430	0.6034	0.7564	0.7587	0.9161	0.9161	0.9828	0.9828
G031	0.8008	0.7583	0.8449	0.8414	0.8414	0.9102	0.9102	1.0000	1.0000	0.9643	0.9643	0.9822	0.9822	0.9822	0.9822	1.0000	1.0000
G032	0.4488	0.4249	0.4458	0.5052	0.4711	0.4629	0.5431	0.6123	0.6628	0.6806	0.7902	0.6886	0.8756	0.9818	0.9818	0.9827	0.9827
G033	0.2066	0.1729	0.2205	0.3153	0.3587	0.4717	0.4502	0.4825	0.4062	0.7903	0.7261	0.7855	0.7719	0.9697	0.9697	0.9537	0.9537
G034	0.2876	0.3433	0.3369	0.3585	0.3149	0.5438	0.4956	0.6457	0.6742	0.8416	0.8377	0.9469	0.9171	0.9120	0.9120	0.9820	0.9820
G035	0.8777	0.9114	0.8836	0.9523	0.9088	0.9827	0.9827	0.9826	0.9826	0.9645	0.9645	0.9702	0.9702	1.0000	1.0000	1.0000	1.0000
G036	0.4761	0.6396	0.6285	0.5345	0.5625	0.6404	0.6304	0.8017	0.7874	0.8694	0.8694	0.9477	0.9477	0.9824	0.9824	1.0000	1.0000
G037	0.4246	0.4237	0.4624	0.4219	0.4363	0.4761	0.4922	0.7682	0.9001	0.9013	0.8708	0.8672	0.8672	0.9457	0.9466	0.9821	0.9821
G038	0.7008	0.7050	0.7783	0.7598	0.7752	0.8176	0.8591	0.9583	0.9583	0.9692	0.9692	0.9821	0.9821	0.9821	0.9821	0.9642	0.9642
G039	0.1636	0.2168	0.2337	0.1834	0.2686	0.3180	0.2743	0.4945	0.5927	0.5282	0.7146	0.7802	0.8285	0.9171	0.9171	0.9821	0.9821
G040	0.8282	0.8736	0.8694	0.9153	0.9043	0.9466	0.9024	0.9688	0.9688	0.9819	0.9819	0.9829	0.9829	1.0000	1.0000	1.0000	1.0000
G041	0.3310	0.4167	0.3308	0.4047	0.4585	0.4695	0.4462	0.6462	0.6198	0.8520	0.8483	0.8681	0.9220	0.9233	0.9233	0.9646	0.9646
G042	0.7381	0.7618	0.7611	0.7431	0.7710	0.8160	0.8595	0.8893	0.8893	0.9517	0.9517	0.9825	0.9825	0.9825	0.9825	1.0000	1.0000
G043	0.6354	0.6587	0.6611	0.6333	0.6710	0.8126	0.7993	0.9150	0.9150	0.9333	0.9333	0.9821	0.9821	1.0000	1.0000	1.0000	1.0000
G044	0.1132	0.1292	0.1569	0.1211	0.2141	0.1875	0.3265	0.5458	0.4222	0.3674	0.6841	0.6758	0.7873	0.9474	0.9474	1.0000	1.0000
G045	0.4977	0.5160	0.5233	0.5360	0.6088	0.7282	0.6584	0.8169	0.8442	0.8929	0.9108	0.9822	0.9822	1.0000	1.0000	0.9826	0.9826
G046	0.6418	0.7123	0.6917	0.7157	0.7353	0.7919	0.7696	0.8917	0.8547	0.9122	0.9122	0.9825	0.9649	0.9825	0.9825	1.0000	1.0000
G047	0.4763	0.5058	0.5412	0.5126	0.4887	0.5786	0.6501	0.6927	0.6819	0.8473	0.8228	0.9823	0.9823	0.9230	0.9230	0.9642	0.9642
G048	0.4873	0.5290	0.5060	0.6327	0.5545	0.7103	0.6889	0.7996	0.7604	0.8077	0.8577	0.9826	0.9826	0.9654	0.9654	1.0000	1.0000
G049	0.3532	0.4051	0.3626	0.5237	0.4466	0.6007	0.5552	0.7218	0.7227	0.8058	0.8280	0.9831	0.9831	0.9831	0.9831	0.9646	0.9646
G050	0.5260	0.6009	0.5472	0.7116	0.5940	0.7577	0.7220	0.8252	0.8019	0.9694	0.9694	0.9341	0.9341	0.9459	0.9459	0.9829	0.9829
Average	**0.4358**	**0.4683**	**0.4706**	**0.5195**	**0.5218**	**0.6128**	**0.6016**	**0.7075**	**0.7145**	**0.8089**	**0.8289**	**0.8830**	**0.9130**	**0.9375**	**0.9490**	**0.9678**	**0.9749**

Table B.6: NMI on mixtures of Gaussians ($K = 4$ and $p = 1.0$).

Dataset	$m = 0$	$m = 100$		$m = 200$		$m = 300$		$m = 400$		$m = 500$		$m = 600$		$m = 700$		$m = 800$	
Priors:		×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓
G001	0.3500	0.2640	0.3107	0.2928	0.2917	0.2755	0.2893	0.2897	0.2895	0.3273	0.3329	0.2798	0.2748	0.3809	0.4049	0.2845	0.2653
G002	0.3279	0.2803	0.3383	0.3061	0.3391	0.3250	0.3356	0.3247	0.3771	0.3268	0.3433	0.3371	0.3768	0.3622	0.3441	0.3661	0.3214
G003	0.2371	0.1939	0.1697	0.2270	0.2054	0.1722	0.2391	0.1611	0.1871	0.2565	0.3007	0.2076	0.1805	0.1998	0.2065	0.2279	0.2364
G004	0.2737	0.2362	0.1831	0.2959	0.2863	0.2821	0.2816	0.3160	0.3419	0.3045	0.2983	0.2433	0.2999	0.2907	0.2512	0.2617	0.2882
G005	0.3630	0.3490	0.4218	0.3649	0.3692	0.4048	0.3564	0.3098	0.3311	0.4000	0.3968	0.4417	0.4227	0.4432	0.3961	0.4314	0.4481
G006	0.2552	0.2292	0.2494	0.2086	0.2109	0.2165	0.2131	0.2156	0.2154	0.1928	0.2340	0.2109	0.2127	0.2776	0.2720	0.2583	0.2758
G007	0.2091	0.2088	0.2166	0.1965	0.2105	0.1626	0.2340	0.2143	0.2184	0.2275	0.2684	0.2147	0.2897	0.2779	0.2457	0.2340	0.2267
G008	0.4421	0.4361	0.4463	0.4386	0.4060	0.4507	0.4194	0.4349	0.3995	0.3737	0.4445	0.4528	0.4767	0.3643	0.3883	0.4831	0.5062
G009	0.3342	0.2470	0.2408	0.2287	0.2611	0.2415	0.2598	0.2765	0.3153	0.2556	0.2234	0.2712	0.2933	0.2159	0.2822	0.3406	0.3129
G010	0.1064	0.0945	0.0916	0.0783	0.0931	0.0955	0.0981	0.0911	0.1169	0.0958	0.0875	0.1018	0.1527	0.0850	0.0817	0.0801	0.0751
G011	0.2636	0.2443	0.2516	0.2550	0.2789	0.2958	0.2522	0.2863	0.3316	0.2719	0.2645	0.2839	0.2908	0.2611	0.2745	0.3063	0.2755
G012	0.5674	0.5480	0.5542	0.5575	0.5255	0.5068	0.5657	0.5676	0.5351	0.5503	0.6217	0.6084	0.6304	0.5820	0.5644	0.5907	0.6077
G013	0.5288	0.5173	0.5094	0.5678	0.6035	0.5309	0.5335	0.6299	0.5673	0.5519	0.5307	0.4766	0.4720	0.5447	0.5761	0.5424	0.5362
G014	0.5401	0.5367	0.5288	0.5106	0.5404	0.5570	0.5437	0.5414	0.5442	0.5220	0.5169	0.5709	0.5555	0.6078	0.6095	0.5759	0.5759
G015	0.5948	0.5229	0.5353	0.5222	0.5346	0.5461	0.5360	0.5589	0.5739	0.6588	0.6487	0.5515	0.6116	0.6034	0.6107	0.5811	0.6041
G016	0.4419	0.4445	0.4896	0.4791	0.5241	0.5137	0.5041	0.4822	0.5098	0.4528	0.4802	0.5343	0.5538	0.4892	0.4851	0.5191	0.5533
G017	0.6353	0.6302	0.6485	0.6505	0.6754	0.5866	0.6158	0.5840	0.6480	0.6141	0.6605	0.6364	0.6305	0.6834	0.7010	0.6645	0.6020
G018	0.6943	0.6745	0.6727	0.6569	0.6715	0.6493	0.6570	0.7115	0.6678	0.5662	0.5842	0.6579	0.7012	0.7218	0.7564	0.7265	0.7310
G019	0.2640	0.2978	0.2695	0.3304	0.2836	0.2399	0.2994	0.2636	0.2982	0.3168	0.3614	0.2920	0.3622	0.2788	0.3289	0.2906	0.4162
G020	0.2724	0.2831	0.2806	0.2958	0.2746	0.2555	0.2859	0.3684	0.3491	0.2509	0.2921	0.2357	0.3300	0.3408	0.3780	0.3661	0.4225
G021	0.2231	0.2505	0.2417	0.2218	0.2476	0.2897	0.2623	0.2334	0.2771	0.2342	0.2238	0.2186	0.2236	0.2521	0.2601	0.2242	0.3163
G022	0.4551	0.3747	0.4480	0.4388	0.4052	0.3884	0.4057	0.4389	0.4281	0.4327	0.4519	0.4626	0.4649	0.4443	0.4276	0.4503	0.4490
G023	0.4808	0.4438	0.4313	0.4598	0.4737	0.4745	0.4328	0.4263	0.4456	0.4187	0.5008	0.5681	0.5587	0.4285	0.4812	0.4616	0.4257
G024	0.4848	0.4709	0.4524	0.5122	0.4874	0.4532	0.4890	0.4985	0.5066	0.4842	0.4978	0.5084	0.5158	0.4946	0.4611	0.5034	0.4637
G025	0.2050	0.2010	0.1900	0.2329	0.2059	0.1986	0.1862	0.2298	0.2175	0.2305	0.2415	0.2406	0.2864	0.2528	0.2658	0.2759	0.2689
G026	0.1613	0.1718	0.1493	0.1816	0.2030	0.1793	0.1509	0.1923	0.2023	0.1952	0.1593	0.1750	0.1714	0.1664	0.1761	0.1427	0.2260
G027	0.5966	0.6355	0.6400	0.6279	0.5968	0.6076	0.6338	0.6399	0.6360	0.6109	0.6780	0.6029	0.6130	0.6955	0.6738	0.7520	0.7361
G028	0.0845	0.0843	0.0875	0.0903	0.1494	0.1097	0.1390	0.1220	0.1198	0.0845	0.1084	0.1325	0.1743	0.1146	0.0984	0.0667	0.1561
G029	0.5188	0.5253	0.5341	0.4784	0.5095	0.4820	0.5093	0.5637	0.5467	0.5763	0.5926	0.4524	0.5655	0.5076	0.4872	0.4982	0.6125
G030	0.1393	0.1136	0.1124	0.0820	0.1060	0.1120	0.0843	0.1352	0.1522	0.1522	0.1019	0.0731	0.1229	0.1164	0.1051	0.1860	0.1442
G031	0.6774	0.6142	0.5696	0.5892	0.6302	0.6290	0.6353	0.6321	0.6277	0.6343	0.6541	0.6077	0.6100	0.6249	0.6212	0.6999	0.7224
G032	0.2869	0.2336	0.2516	0.2304	0.2305	0.2333	0.2887	0.2845	0.2772	0.2593	0.2970	0.3175	0.3631	0.3130	0.3202	0.3557	0.4078
G033	0.2098	0.2007	0.2083	0.1946	0.1989	0.2108	0.1988	0.2273	0.1959	0.2190	0.2943	0.2735	0.2413	0.1875	0.2769	0.2013	0.1764
G034	0.2090	0.2173	0.2241	0.1908	0.2083	0.2271	0.1926	0.2118	0.2278	0.1787	0.1647	0.2357	0.2663	0.2426	0.2833	0.2057	0.3283
G035	0.7883	0.7650	0.7668	0.7658	0.7734	0.7805	0.7394	0.7312	0.7488	0.7751	0.7751	0.8377	0.8377	0.8249	0.8435	0.7713	0.7905
G036	0.4121	0.3891	0.4029	0.3998	0.4144	0.3626	0.4297	0.3676	0.3778	0.4058	0.3669	0.4280	0.4699	0.3699	0.4314	0.4219	0.4409
G037	0.5305	0.5362	0.5283	0.5797	0.5569	0.5712	0.5473	0.5237	0.5454	0.5324	0.5356	0.5502	0.5642	0.5720	0.5957	0.5439	0.5475
G038	0.6247	0.6091	0.5984	0.6248	0.6339	0.5952	0.5888	0.6226	0.6816	0.6017	0.6319	0.6328	0.6285	0.6953	0.6648	0.6807	0.6272
G039	0.1737	0.1258	0.1212	0.1170	0.1265	0.1712	0.1865	0.1500	0.1464	0.1021	0.1327	0.1474	0.1541	0.1349	0.1649	0.1306	0.1768
G040	0.6177	0.6397	0.5840	0.6147	0.6341	0.6373	0.6296	0.6271	0.7007	0.6027	0.5823	0.5946	0.6681	0.6003	0.6408	0.6411	0.6387
G041	0.5115	0.4290	0.5016	0.4805	0.4307	0.5137	0.5203	0.5729	0.5303	0.5266	0.4713	0.4855	0.5342	0.5012	0.5030	0.5535	0.5478
G042	0.6329	0.6321	0.6331	0.6205	0.6118	0.6133	0.6612	0.6191	0.6291	0.6724	0.6509	0.6594	0.6681	0.6786	0.6211	0.5797	0.6426
G043	0.5838	0.5487	0.5914	0.5685	0.5602	0.5486	0.5670	0.5716	0.6584	0.5494	0.5579	0.5861	0.5616	0.5550	0.5641	0.6667	0.6481
G044	0.2463	0.2304	0.2431	0.2235	0.2047	0.2203	0.2095	0.2342	0.2335	0.1956	0.2509	0.2039	0.1661	0.2069	0.2445	0.2327	0.3016
G045	0.5782	0.5634	0.5801	0.5564	0.5796	0.5678	0.5701	0.5534	0.5470	0.5282	0.5795	0.5836	0.5481	0.6003	0.5930	0.6048	0.5530
G046	0.4533	0.4762	0.4458	0.4520	0.4319	0.4296	0.3982	0.4556	0.4040	0.3802	0.4531	0.3916	0.4118	0.4252	0.4620	0.4597	0.4635
G047	0.3520	0.3656	0.3559	0.3184	0.3397	0.3465	0.3284	0.3853	0.4105	0.3582	0.3599	0.3263	0.3288	0.3922	0.3625	0.3695	0.4031
G048	0.3604	0.3692	0.3628	0.3303	0.3714	0.3658	0.3446	0.3815	0.4003	0.3811	0.4191	0.3914	0.3825	0.3964	0.4499	0.3118	0.3554
G049	0.2585	0.2720	0.2937	0.3464	0.3043	0.3111	0.3026	0.2858	0.2616	0.2765	0.2740	0.2915	0.3443	0.2829	0.3657	0.3054	0.3193
G050	0.4589	0.4432	0.4303	0.3982	0.5025	0.4135	0.4322	0.3328	0.3660	0.4215	0.4150	0.4444	0.4416	0.4839	0.4779	0.4791	0.4417
Average	**0.4003**	**0.3834**	**0.3878**	**0.3878**	**0.3943**	**0.3870**	**0.3917**	**0.3976**	**0.4064**	**0.3907**	**0.4063**	**0.4006**	**0.4201**	**0.4114**	**0.4216**	**0.4181**	**0.4322**

Table B.7: NMI on mixtures of Gaussians ($K = 6$ and $p = 0.8$).

Dataset	$m = 0$	$m = 100$		$m = 200$		$m = 300$		$m = 400$		$m = 500$		$m = 600$		$m = 700$		$m = 800$	
Priors:		×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓
G001	0.3500	0.2917	0.2931	0.2945	0.2804	0.2796	0.3181	0.3280	0.3658	0.4056	0.3425	0.2796	0.3247	0.4273	0.4795	0.4100	0.4810
G002	0.3279	0.2946	0.2987	0.3263	0.3663	0.4031	0.3599	0.3501	0.4293	0.3776	0.3463	0.4660	0.4628	0.4498	0.4143	0.3946	0.4264
G003	0.2371	0.1842	0.2317	0.2071	0.1889	0.2296	0.2317	0.2229	0.2008	0.2688	0.2928	0.3107	0.2835	0.2510	0.2600	0.3283	0.3109
G004	0.2737	0.2431	0.2650	0.2875	0.2531	0.2601	0.2745	0.3657	0.3235	0.3222	0.3454	0.3701	0.3678	0.4409	0.4678	0.3099	0.3423
G005	0.3630	0.3907	0.3700	0.3786	0.4085	0.4045	0.3647	0.3693	0.3941	0.4527	0.4441	0.4803	0.5017	0.5349	0.4988	0.5306	0.5906
G006	0.2552	0.2329	0.2317	0.1651	0.1989	0.2759	0.2213	0.2611	0.2384	0.2814	0.2608	0.3370	0.3597	0.3128	0.2917	0.3656	0.3279
G007	0.2091	0.2349	0.2279	0.2006	0.2267	0.2686	0.2820	0.2694	0.2678	0.2393	0.3621	0.2953	0.3126	0.4035	0.3610	0.2961	0.4442
G008	0.4421	0.4747	0.4505	0.4112	0.4183	0.4176	0.4160	0.4308	0.4602	0.4766	0.4342	0.4260	0.4868	0.5022	0.4976	0.4832	0.4852
G009	0.3342	0.2631	0.2764	0.2382	0.2527	0.2683	0.2408	0.2636	0.3217	0.2362	0.3178	0.3443	0.3256	0.3273	0.4662	0.4021	0.4099
G010	0.1064	0.1185	0.1399	0.0810	0.0979	0.0980	0.1018	0.0632	0.0897	0.0943	0.1506	0.1159	0.1175	0.0719	0.0992	0.1238	0.1250
G011	0.2636	0.2974	0.2768	0.2642	0.3216	0.2938	0.3133	0.3041	0.3402	0.2922	0.3199	0.3794	0.3648	0.2995	0.3517	0.4400	0.4791
G012	0.5674	0.5676	0.5607	0.5973	0.6054	0.5727	0.5582	0.6137	0.5278	0.6628	0.6887	0.5943	0.7037	0.6341	0.6821	0.6902	0.6755
G013	0.5288	0.4957	0.5134	0.6052	0.5737	0.6101	0.6096	0.6289	0.6734	0.6874	0.7259	0.6661	0.6485	0.6708	0.6834	0.7631	0.7496
G014	0.5401	0.5227	0.5502	0.5588	0.5547	0.5482	0.5752	0.5276	0.5812	0.5272	0.5492	0.5478	0.5970	0.6248	0.7368	0.6644	0.6826
G015	0.5948	0.5906	0.5799	0.5195	0.5808	0.5208	0.5914	0.5960	0.6665	0.6821	0.7283	0.6147	0.5799	0.7187	0.6896	0.6315	0.7293
G016	0.4419	0.4926	0.4324	0.4700	0.4784	0.5155	0.5306	0.5568	0.5693	0.4705	0.4721	0.5704	0.6259	0.5634	0.5463	0.6339	0.6109
G017	0.6353	0.6227	0.6338	0.6891	0.6399	0.6126	0.6038	0.6883	0.6677	0.6928	0.7255	0.6808	0.7505	0.7070	0.6821	0.7120	0.7495
G018	0.6943	0.6697	0.7240	0.6948	0.6983	0.6652	0.6826	0.7472	0.7374	0.7418	0.7704	0.7380	0.7456	0.7679	0.8281	0.8435	0.8435
G019	0.2640	0.2712	0.3061	0.3306	0.3157	0.2631	0.2898	0.2785	0.3597	0.3669	0.3394	0.4172	0.5175	0.3660	0.3924	0.3460	0.3888
G020	0.2724	0.3128	0.3220	0.3297	0.3350	0.2899	0.2939	0.3204	0.3873	0.4177	0.4087	0.4050	0.4502	0.4881	0.4865	0.4574	0.5033
G021	0.2231	0.2201	0.2209	0.2708	0.2711	0.2127	0.2754	0.2557	0.2445	0.3342	0.2985	0.3017	0.3129	0.3007	0.3874	0.3663	0.3603
G022	0.4551	0.4003	0.4822	0.4240	0.4353	0.4271	0.4769	0.4419	0.4895	0.4715	0.5195	0.5270	0.5498	0.5427	0.5655	0.5633	0.5097
G023	0.4808	0.4682	0.4902	0.5048	0.4961	0.4427	0.4358	0.5111	0.5083	0.4340	0.5744	0.4929	0.6148	0.4810	0.4939	0.5073	0.6604
G024	0.4848	0.4505	0.5016	0.5178	0.5087	0.4647	0.4927	0.4980	0.5203	0.4831	0.4866	0.5329	0.5400	0.5090	0.4992	0.5044	0.5555
G025	0.2050	0.2131	0.2054	0.2648	0.2273	0.2218	0.2180	0.2575	0.2560	0.2659	0.2723	0.2684	0.2836	0.2552	0.2732	0.3328	0.2997
G026	0.1613	0.1436	0.1693	0.1979	0.2225	0.1934	0.1889	0.2046	0.2034	0.2210	0.2583	0.1855	0.1470	0.2434	0.1791	0.2519	0.2687
G027	0.5966	0.6464	0.6690	0.5959	0.6463	0.6386	0.6801	0.6825	0.7395	0.7159	0.7159	0.6538	0.7150	0.7706	0.8125	0.7287	0.7200
G028	0.0845	0.0729	0.1085	0.0897	0.1067	0.0961	0.1380	0.0923	0.1079	0.1360	0.1124	0.1394	0.1703	0.1070	0.1451	0.1412	0.2158
G029	0.5188	0.5033	0.5245	0.5323	0.5655	0.5144	0.4912	0.5005	0.5479	0.4909	0.4994	0.6223	0.6196	0.5817	0.6077	0.5837	0.6547
G030	0.1393	0.1696	0.1661	0.1016	0.0919	0.1204	0.1058	0.1358	0.1817	0.1003	0.1622	0.1813	0.1877	0.1430	0.1726	0.1832	0.1714
G031	0.6774	0.5974	0.5875	0.6181	0.6391	0.6095	0.6434	0.6681	0.6785	0.7288	0.7477	0.7233	0.7361	0.7339	0.7544	0.7618	0.7799
G032	0.2869	0.2846	0.2812	0.2339	0.2527	0.2583	0.3322	0.2470	0.3300	0.2638	0.3173	0.3791	0.4464	0.3563	0.4247	0.4627	0.5456
G033	0.2098	0.2240	0.2964	0.2458	0.2482	0.2030	0.2025	0.2544	0.2486	0.2629	0.3060	0.3421	0.3304	0.3879	0.3235	0.2849	0.3877
G034	0.2090	0.2238	0.2578	0.2170	0.2316	0.2766	0.2516	0.3013	0.2807	0.2994	0.2818	0.2248	0.3842	0.3539	0.4045	0.3747	0.4291
G035	0.7883	0.7731	0.7828	0.8029	0.8081	0.8176	0.8335	0.8119	0.8194	0.8131	0.8195	0.8186	0.8635	0.8114	0.8566	0.8190	0.8337
G036	0.4121	0.4280	0.4639	0.4070	0.3991	0.4003	0.4248	0.4855	0.4742	0.4385	0.4896	0.5352	0.5707	0.5909	0.6048	0.4823	0.5192
G037	0.5305	0.5133	0.5397	0.5479	0.5437	0.5179	0.5762	0.5661	0.6291	0.6060	0.6517	0.7114	0.6679	0.5828	0.6662	0.6068	0.6531
G038	0.6247	0.6301	0.6121	0.6219	0.6528	0.6643	0.6216	0.6091	0.6769	0.6571	0.7024	0.6658	0.6637	0.7648	0.7591	0.8209	0.8291
G039	0.1737	0.1092	0.1190	0.1491	0.2034	0.1219	0.1298	0.1659	0.1656	0.1238	0.1393	0.1423	0.1467	0.2132	0.2301	0.1809	0.1662
G040	0.6177	0.6108	0.5933	0.6354	0.6491	0.6584	0.6500	0.7229	0.7749	0.6869	0.6789	0.6755	0.6675	0.6790	0.7299	0.7075	0.6812
G041	0.5115	0.5177	0.4660	0.5764	0.4871	0.5228	0.5637	0.5715	0.6550	0.5408	0.5255	0.5768	0.6075	0.5034	0.5678	0.6923	0.6889
G042	0.6329	0.6328	0.6487	0.6456	0.6404	0.6385	0.6736	0.6528	0.6431	0.6671	0.6666	0.7835	0.7933	0.6743	0.7495	0.7453	0.7854
G043	0.5838	0.6041	0.5248	0.5906	0.5510	0.5831	0.5483	0.6556	0.6916	0.5950	0.5871	0.7535	0.7403	0.5893	0.6335	0.7618	0.7290
G044	0.2463	0.2269	0.2403	0.1986	0.1876	0.2555	0.2132	0.2324	0.2588	0.2878	0.2827	0.2804	0.3114	0.2872	0.4044	0.3281	0.3813
G045	0.5782	0.5697	0.5916	0.5590	0.5439	0.6112	0.6042	0.5647	0.5480	0.5611	0.5702	0.6636	0.5818	0.5953	0.6608	0.6900	0.6793
G046	0.4533	0.4628	0.4620	0.4216	0.4138	0.4651	0.4556	0.4462	0.4287	0.4676	0.4337	0.4895	0.4984	0.4724	0.4817	0.4998	0.4914
G047	0.3520	0.3137	0.3880	0.3429	0.3406	0.3733	0.3585	0.3799	0.4117	0.3674	0.3654	0.3875	0.3516	0.3920	0.4167	0.4275	0.5264
G048	0.3604	0.3390	0.3518	0.4341	0.4989	0.3330	0.3368	0.4094	0.4923	0.3986	0.4504	0.5123	0.4859	0.4924	0.4934	0.4534	0.5095
G049	0.2585	0.2855	0.2952	0.3567	0.3429	0.3104	0.2957	0.2462	0.3481	0.3511	0.4054	0.3783	0.3953	0.3441	0.4087	0.3313	0.4478
G050	0.4589	0.4461	0.4480	0.4846	0.4448	0.4781	0.5128	0.4820	0.4695	0.4595	0.4633	0.6650	0.6290	0.5175	0.5521	0.5355	0.5346
Average	**0.4003**	**0.3930**	**0.4034**	**0.4048**	**0.4089**	**0.4046**	**0.4118**	**0.4248**	**0.4485**	**0.4385**	**0.4562**	**0.4731**	**0.4908**	**0.4768**	**0.5035**	**0.4991**	**0.5274**

Table B.8: NMI on mixtures of Gaussians ($K = 6$ and $p = 0.9$).

Dataset	$m = 0$	$m = 100$		$m = 200$		$m = 300$		$m = 400$		$m = 500$		$m = 600$		$m = 700$		$m = 800$	
Priors:		×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓
G001	0.3500	0.3035	0.3274	0.2996	0.3481	0.3119	0.3015	0.3944	0.4909	0.4050	0.5077	0.4776	0.4681	0.5170	0.5897	0.5592	0.7445
G002	0.3279	0.3456	0.3912	0.3498	0.4055	0.4118	0.5277	0.4499	0.4054	0.5048	0.5674	0.5142	0.5557	0.6400	0.7035	0.7824	0.7547
G003	0.2371	0.1977	0.2659	0.2846	0.2841	0.2635	0.3647	0.3023	0.3386	0.2516	0.3900	0.3858	0.4410	0.5953	0.6329	0.5919	0.6667
G004	0.2737	0.2981	0.2855	0.2820	0.3352	0.3205	0.4189	0.4256	0.4104	0.4350	0.4882	0.4891	0.5040	0.6344	0.7357	0.6868	0.7097
G005	0.3630	0.3738	0.4071	0.4093	0.4874	0.5159	0.4143	0.5193	0.5414	0.5628	0.5769	0.5989	0.6905	0.5906	0.8096	0.6991	0.8877
G006	0.2552	0.2202	0.2209	0.2228	0.2686	0.3005	0.3017	0.2620	0.2977	0.3579	0.3774	0.4302	0.5066	0.3756	0.4645	0.5906	0.7289
G007	0.2091	0.2179	0.2259	0.2712	0.2634	0.3376	0.2572	0.3081	0.3236	0.4183	0.3886	0.4905	0.5436	0.4454	0.6295	0.6386	0.6961
G008	0.4421	0.4242	0.4291	0.4617	0.4337	0.4671	0.4428	0.4827	0.5686	0.4660	0.4982	0.4990	0.5546	0.5301	0.7826	0.6952	0.7770
G009	0.3342	0.3337	0.3065	0.2757	0.2829	0.3760	0.3047	0.2859	0.3147	0.4468	0.4438	0.5522	0.5276	0.5743	0.6846	0.7813	0.7731
G010	0.1064	0.0943	0.1017	0.0936	0.1601	0.1332	0.1589	0.1003	0.1915	0.1240	0.2150	0.2789	0.4585	0.1156	0.3917	0.3213	0.4092
G011	0.2636	0.2765	0.2713	0.3093	0.3520	0.3178	0.3699	0.2969	0.4856	0.3889	0.4727	0.6592	0.6198	0.6605	0.7140	0.7754	0.8582
G012	0.5674	0.5761	0.6138	0.5864	0.6065	0.6591	0.6967	0.6823	0.7640	0.6810	0.7308	0.7965	0.8193	0.8555	0.7955	0.8865	0.8844
G013	0.5288	0.5242	0.5220	0.5680	0.6067	0.6754	0.6743	0.7414	0.7085	0.7228	0.7585	0.8732	0.8659	0.9212	0.9083	0.9200	0.9200
G014	0.5401	0.5245	0.5466	0.5591	0.5625	0.5727	0.6138	0.5925	0.6646	0.6575	0.6794	0.7971	0.7919	0.7666	0.8374	0.9499	0.9499
G015	0.5948	0.5810	0.5456	0.6549	0.7034	0.6904	0.6599	0.7861	0.8048	0.7746	0.7840	0.8517	0.8752	0.8414	0.9059	0.9757	0.9880
G016	0.4419	0.4239	0.4946	0.5279	0.5401	0.5585	0.5407	0.5915	0.6740	0.7181	0.6326	0.7529	0.7909	0.7720	0.8541	0.8679	0.9092
G017	0.6353	0.5810	0.6474	0.6782	0.7095	0.6869	0.7030	0.7186	0.7694	0.7840	0.7706	0.8153	0.8546	0.9249	0.9514	0.9053	0.9512
G018	0.6943	0.6901	0.7042	0.7265	0.7658	0.7628	0.7379	0.8613	0.9078	0.8886	0.9166	0.9292	0.9375	0.9409	0.9621	0.9013	0.9191
G019	0.2640	0.3061	0.3380	0.2651	0.3622	0.3417	0.3412	0.3246	0.4161	0.4288	0.5175	0.4981	0.5602	0.4093	0.6587	0.7695	0.6744
G020	0.2724	0.2762	0.3205	0.2886	0.3966	0.4479	0.4073	0.4995	0.5344	0.5499	0.4942	0.5650	0.5449	0.7581	0.7839	0.7198	0.8594
G021	0.2231	0.2144	0.2429	0.2603	0.2826	0.3481	0.3511	0.2567	0.3440	0.2801	0.4108	0.3436	0.4624	0.4588	0.5169	0.4070	0.5645
G022	0.4551	0.4393	0.4632	0.4623	0.5597	0.4972	0.5640	0.5113	0.5772	0.5441	0.5903	0.6189	0.7543	0.6148	0.6848	0.7028	0.8699
G023	0.4808	0.4721	0.5156	0.4796	0.4883	0.5350	0.5992	0.5254	0.7120	0.6474	0.7426	0.7878	0.8036	0.8856	0.8588	0.8925	0.8925
G024	0.4848	0.4371	0.4819	0.5455	0.5557	0.5220	0.5961	0.5505	0.5816	0.5981	0.7199	0.6705	0.7501	0.6480	0.7200	0.7922	0.7888
G025	0.2050	0.2332	0.2150	0.2044	0.1832	0.2380	0.3644	0.3309	0.2756	0.3123	0.3632	0.3253	0.4865	0.3380	0.4396	0.4992	0.6555
G026	0.1613	0.1983	0.1927	0.2119	0.2464	0.1962	0.2943	0.2039	0.3229	0.2928	0.3704	0.3144	0.3835	0.2272	0.4097	0.4046	0.5103
G027	0.5966	0.6475	0.6064	0.6147	0.7183	0.7075	0.7102	0.7731	0.7946	0.8404	0.8611	0.8864	0.8733	0.9220	0.9220	0.9755	0.9755
G028	0.0845	0.1021	0.1065	0.1010	0.1669	0.1244	0.1833	0.1593	0.2903	0.1508	0.2233	0.2548	0.2781	0.2981	0.5479	0.2428	0.4283
G029	0.5188	0.5583	0.5828	0.5639	0.5623	0.6120	0.6014	0.6397	0.6275	0.7415	0.7257	0.8501	0.8003	0.8729	0.8967	0.8619	0.9035
G030	0.1393	0.1619	0.1631	0.0982	0.1267	0.1086	0.2251	0.1593	0.2427	0.2013	0.3464	0.1730	0.2596	0.2585	0.3809	0.4820	0.4756
G031	0.6774	0.7010	0.6565	0.7141	0.6887	0.7527	0.7209	0.7382	0.7177	0.8194	0.8648	0.8241	0.8936	0.9294	0.9166	0.9294	0.9211
G032	0.2869	0.2523	0.2690	0.3138	0.4145	0.3637	0.2885	0.3709	0.4010	0.4871	0.5682	0.4131	0.5097	0.7448	0.7698	0.7288	0.8568
G033	0.2098	0.2059	0.2437	0.2289	0.2655	0.3240	0.2979	0.2651	0.3176	0.3477	0.5115	0.5098	0.4520	0.6103	0.5396	0.8497	0.7290
G034	0.2090	0.2488	0.2402	0.2448	0.2731	0.2750	0.3193	0.3842	0.3579	0.3400	0.4139	0.4954	0.5808	0.4332	0.5846	0.7806	0.8687
G035	0.7883	0.8014	0.7985	0.8366	0.8276	0.8261	0.8420	0.8075	0.8053	0.8131	0.9406	0.8763	0.8808	0.8787	0.8993	0.9144	0.9433
G036	0.4121	0.4123	0.4061	0.4374	0.4912	0.5023	0.4776	0.5142	0.4919	0.5507	0.5409	0.7496	0.5923	0.7927	0.7603	0.8410	0.8358
G037	0.5305	0.5503	0.5818	0.5777	0.5816	0.6286	0.6193	0.6519	0.5984	0.7338	0.8018	0.8526	0.8022	0.8259	0.7340	0.9624	0.9330
G038	0.6247	0.6079	0.6476	0.6383	0.6654	0.6320	0.6711	0.7582	0.7540	0.7995	0.8265	0.8653	0.8459	0.8618	0.8982	0.9595	0.9595
G039	0.1737	0.1537	0.1764	0.1591	0.2356	0.1835	0.1768	0.2042	0.2265	0.2394	0.3296	0.2623	0.4448	0.3137	0.4530	0.3735	0.5004
G040	0.6177	0.6126	0.6308	0.6534	0.6529	0.6677	0.6558	0.6560	0.7102	0.7602	0.7895	0.8528	0.8782	0.8690	0.8773	1.0000	1.0000
G041	0.5115	0.5576	0.5458	0.5552	0.5809	0.6491	0.5695	0.6157	0.5821	0.6368	0.7039	0.7289	0.8022	0.8602	0.8473	0.9677	0.9878
G042	0.6329	0.6364	0.6280	0.6961	0.7176	0.6511	0.6474	0.6951	0.6968	0.7924	0.7756	0.8714	0.8170	0.8027	0.9396	0.9254	0.9254
G043	0.5838	0.5593	0.5497	0.6046	0.6801	0.7483	0.7413	0.7229	0.7497	0.7173	0.7302	0.8394	0.8775	0.9153	0.9280	0.8782	0.9049
G044	0.2463	0.2681	0.2096	0.2138	0.2561	0.2216	0.3206	0.3854	0.2450	0.4802	0.4435	0.4687	0.4331	0.6059	0.5876	0.4741	0.6557
G045	0.5782	0.5797	0.5786	0.5527	0.5922	0.5483	0.6661	0.5887	0.6730	0.6052	0.6842	0.5907	0.8422	0.6377	0.8760	0.9084	0.9629
G046	0.4533	0.4536	0.4442	0.4665	0.4262	0.4964	0.4951	0.5137	0.4967	0.5328	0.6155	0.5324	0.5472	0.6804	0.7978	0.7936	0.8531
G047	0.3520	0.3734	0.3646	0.3533	0.3913	0.3655	0.4928	0.4150	0.4658	0.3910	0.5559	0.5961	0.6354	0.6127	0.6819	0.6088	0.7934
G048	0.3604	0.4045	0.4346	0.3847	0.4127	0.4418	0.4747	0.6061	0.6197	0.5964	0.6123	0.6074	0.6323	0.7595	0.7598	0.8438	0.8434
G049	0.2585	0.3160	0.3621	0.3579	0.3270	0.3809	0.3885	0.4055	0.4598	0.5482	0.5352	0.5785	0.6392	0.7597	0.8182	0.7142	0.8028
G050	0.4589	0.3672	0.4750	0.4964	0.5108	0.5124	0.5070	0.5949	0.5995	0.6804	0.6451	0.6063	0.6706	0.7258	0.7705	0.8133	0.7895
Average	**0.4003**	**0.4019**	**0.4156**	**0.4228**	**0.4551**	**0.4642**	**0.4820**	**0.4926**	**0.5270**	**0.5409**	**0.5891**	**0.6120**	**0.6508**	**0.6602**	**0.7322**	**0.7509**	**0.8038**

Table B.9: NMI on mixtures of Gaussians ($K = 6$ and $p = 1.0$).

B.4
Detailed Results on UCI Datasets

Tables B.10 to B.17 present the general clustering performance obtained with the proposed semi-supervised model on UCI datasets. We report the results regarding NMI, KL divergence, and CI, along with the computational time obtained with the proposed solution method. Each table presents the performance obtained with and without prior distributions for different values of p.

	NMI						KL divergence						CI						Time (s)					
	$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$	
Priors:	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓
$m = 0$	0.6565						0.4845						0						7.83					
$m = N/2$	0.6471	0.6316	0.6471	0.6649	0.7011	0.7169	0.5083	0.3471	0.5083	0.2696	0.2644	0.1503	0	0	0	0	0	0	11.18	8.84	9.72	7.99	8.27	6.65
	0.6565	0.6565	0.6285	0.6285	0.6565	0.7472	0.4845	0.4845	0.2155	0.2155	0.4845	0.1271	0	0	0	0	0	0	8.24	8.11	7.81	7.37	7.84	7.21
	0.6215	0.6215	0.6215	0.6215	0.6215	0.6837	0.3783	0.3783	0.3783	0.3783	0.3785	0.0945	0	0	0	0	0	0	6.87	6.97	6.93	6.96	6.68	6.51
	0.6471	0.6565	0.6471	0.6565	0.6565	0.6982	0.5083	0.4845	0.5083	0.4845	0.4845	0.3854	0	0	0	0	0	0	8.52	7.90	7.12	6.65	6.66	7.60
	0.6019	0.6019	0.6285	0.6285	0.6886	0.7402	0.2188	0.2188	0.2079	0.2079	0.2102	0.1372	0	0	0	0	0	0	7.75	7.53	6.70	7.10	7.63	7.11
	0.6267	0.6267	0.6415	0.6415	0.6421	0.7055	0.2162	0.2162	0.2205	0.2205	0.3251	0.1716	0	0	0	0	0	0	8.22	9.45	8.24	8.20	7.89	7.53
	0.6565	0.6565	0.6565	0.6565	0.6661	0.7201	0.4845	0.4845	0.4845	0.4845	0.4447	0.1430	0	0	0	0	0	0	9.67	8.87	9.17	9.36	8.60	6.89
	0.6565	0.6249	0.6442	0.6442	0.5930	0.6513	0.4845	0.4612	0.3965	0.3965	0.3851	0.1521	0	0	0	0	0	0	10.00	9.44	7.10	7.26	7.97	7.40
	0.6565	0.6565	0.6565	0.6908	0.6885	0.7293	0.4845	0.4845	0.4845	0.2204	0.2743	0.1670	0	0	0	0	0	0	9.41	7.82	6.71	7.95	8.00	7.34
	0.6471	0.6565	0.6471	0.6661	0.6619	0.7506	0.5083	0.4845	0.5083	0.4447	0.4839	0.1011	0	0	0	0	0	0	8.02	9.37	10.75	9.37	7.76	7.25
Average	**0.6417**	**0.6389**	**0.6419**	**0.6499**	**0.6576**	**0.7143**	**0.4276**	**0.4044**	**0.3913**	**0.3322**	**0.3735**	**0.1629**	**0**	**0**	**0**	**0**	**0**	**0**	**8.79**	**8.43**	**8.03**	**7.82**	**7.73**	**7.15**
$m = N$	0.6565	0.6565	0.7063	0.7063	0.7146	0.6922	0.4845	0.4845	0.2027	0.2027	0.2443	0.0532	0	0	0	0	0	0	9.24	9.16	9.35	9.32	8.46	8.11
	0.6773	0.6649	0.7146	0.7011	0.6908	0.7482	0.2574	0.2693	0.2487	0.2598	0.2411	0.1470	0	0	0	0	0	0	8.46	8.67	8.49	8.87	10.07	8.79
	0.6565	0.6565	0.6565	0.6977	0.7616	0.8397	0.4845	0.4845	0.4845	0.3494	0.2168	0.0407	0	0	0	0	0	0	10.40	9.62	8.78	8.33	9.16	8.63
	0.6816	0.6816	0.6949	0.6949	0.7480	0.7961	0.1941	0.1941	0.1907	0.1907	0.2513	0.0793	0	0	0	0	0	0	8.58	8.91	9.35	9.63	9.33	7.85
	0.6421	0.6421	0.6773	0.6773	0.7343	0.7148	0.3251	0.3251	0.2488	0.2488	0.2748	0.3630	0	0	0	0	0	0	8.54	8.94	9.08	9.13	9.93	9.22
	0.6859	0.6859	0.6900	0.7251	0.7988	0.8935	0.1681	0.1681	0.1972	0.2023	0.2053	0.0735	0	0	0	0	0	0	8.33	8.85	8.53	8.86	9.83	9.44
	0.6565	0.6711	0.6565	0.7039	0.6773	0.7380	0.4845	0.4597	0.4845	0.2004	0.2556	0.1559	0	0	0	0	0	0	9.52	9.14	9.15	9.37	9.26	9.19
	0.6442	0.6442	0.6867	0.6867	0.7448	0.8020	0.3882	0.3882	0.3811	0.3811	0.1941	0.0805	0	0	0	0	0	0	11.31	10.57	8.80	8.63	9.16	8.68
	0.6565	0.6565	0.6267	0.6577	0.6949	0.7600	0.4845	0.4845	0.2288	0.2078	0.1868	0.0356	0	0	0	0	0	0	10.41	9.88	9.43	9.23	8.38	9.19
	0.6565	0.6565	0.6565	0.6565	0.6762	0.8226	0.4845	0.4845	0.4845	0.4845	0.4162	0.0884	0	0	0	0	0	0	12.57	14.03	11.86	12.41	12.04	9.27
Average	**0.6614**	**0.6616**	**0.6766**	**0.6907**	**0.7241**	**0.7807**	**0.3755**	**0.3743**	**0.3152**	**0.2728**	**0.2486**	**0.1117**	**0**	**0**	**0**	**0**	**0**	**0**	**9.74**	**9.78**	**9.28**	**9.38**	**9.56**	**8.84**
$m = 1.5N$	0.6565	0.6885	0.7790	0.7126	0.7988	0.9157	0.4845	0.2846	0.2186	0.1456	0.2053	0.0324	0	0	0	0	0	0	10.21	9.29	9.21	8.95	11.08	10.75
	0.6267	0.6267	0.6908	0.7072	0.7251	0.7955	0.2318	0.2318	0.2366	0.2208	0.2046	0.0468	0	0	0	0	0	0	9.89	10.02	9.51	10.92	10.84	9.80
	0.6565	0.6565	0.7118	0.7006	0.8062	0.8579	0.4845	0.4845	0.3138	0.1664	0.1627	0.0529	0	0	0	0	0	0	9.49	9.80	8.91	9.17	9.40	10.11
	0.6816	0.6949	0.6949	0.6949	0.7353	0.8988	0.1941	0.1929	0.1907	0.1907	0.2677	0.0307	0	0	0	0	0	0	9.02	9.57	9.66	9.57	10.35	9.97
	0.6584	0.6584	0.7448	0.7448	0.7671	0.8042	0.1997	0.1997	0.1948	0.1948	0.1760	0.0838	0	0	0	0	0	0	8.64	8.85	9.47	9.15	10.03	10.48
	0.6766	0.6753	0.7096	0.7096	0.7671	0.8627	0.3098	0.1974	0.1922	0.1922	0.2012	0.1009	0	0	0	0	0	0	9.74	9.31	8.68	9.11	10.47	9.76
	0.6565	0.6711	0.6249	0.6394	0.7063	0.8464	0.4845	0.4597	0.4573	0.4329	0.2009	0.0301	0	0	0	0	0	0	9.51	10.12	9.54	9.53	10.71	9.56
	0.6565	0.6565	0.6977	0.6977	0.7671	0.9029	0.4845	0.4845	0.3537	0.3537	0.1966	0.0263	0	0	0	0	0	0	9.69	9.41	10.29	10.79	10.22	9.51
	0.6565	0.6753	0.6621	0.6753	0.7997	0.9029	0.4845	0.1976	0.1951	0.1976	0.1404	0.0240	0	0	0	0	0	0	10.01	10.04	9.91	10.35	10.75	9.64
	0.6565	0.6565	0.6565	0.6565	0.7482	0.8869	0.4845	0.4845	0.4845	0.4845	0.1750	0.0179	0	0	0	0	0	0	10.73	11.37	12.67	12.07	9.54	10.19
Average	**0.6582**	**0.6660**	**0.6972**	**0.6939**	**0.7621**	**0.8674**	**0.3842**	**0.3217**	**0.2837**	**0.2579**	**0.1930**	**0.0446**	**0**	**0**	**0**	**0**	**0**	**0**	**9.69**	**9.78**	**9.79**	**9.96**	**10.34**	**9.98**

Table B.10: Detailed performance in the *Diabetes* dataset.

	NMI						KL divergence						CI						Time (s)					
	$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$	
Priors:	$\times$	$\checkmark$	$\times$	$\checkmark$	$\times$	$\checkmark$	$\times$	$\checkmark$	$\times$	$\checkmark$	$\times$	$\checkmark$	$\times$	$\checkmark$	$\times$	$\checkmark$	$\times$	$\checkmark$	$\times$	$\checkmark$	$\times$	$\checkmark$	$\times$	$\checkmark$
$m = 0$	0.7475						0.1436						0						3.65					
$m = N/2$	0.7650	0.7927	0.8555	0.8555	0.8049	0.7971	0.1545	0.1439	0.0693	0.0693	0.0525	0.0353	0	0	0	0	0	0	7.44	7.09	6.92	7.29	6.95	6.75
	0.7462	0.8027	0.7616	0.8079	0.8196	0.8324	0.0974	0.0375	0.1045	0.0428	0.0354	0.0183	0	0	0	0	0	0	6.46	6.62	6.78	7.27	8.21	7.12
	0.7650	0.8714	0.8714	0.8714	0.7650	0.8196	0.1545	0.0334	0.0334	0.0334	0.1545	0.0324	0	0	0	0	0	0	7.01	6.93	7.09	7.71	6.07	6.90
	0.7475	0.7475	0.7872	0.7900	0.8049	0.8465	0.1436	0.1436	0.0592	0.0348	0.0576	0.0195	0	0	0	0	0	0	7.01	6.95	6.47	6.87	7.12	6.63
	0.7543	0.7650	0.7718	0.7616	0.8259	0.8467	0.1176	0.1545	0.1268	0.0993	0.0659	0.0286	0	0	0	0	0	0	7.06	7.00	7.88	7.51	6.64	6.83
	0.8256	0.8256	0.8148	0.7971	0.8298	0.8585	0.0415	0.0415	0.0534	0.0378	0.1293	0.0228	0	0	0	0	0	0	7.34	7.38	6.71	7.87	7.28	6.72
	0.7475	0.7475	0.8196	0.8196	0.8256	0.8256	0.1436	0.1436	0.0247	0.0247	0.0375	0.0375	0	0	0	0	0	0	6.82	6.60	6.63	7.92	7.15	7.20
	0.8259	0.8259	0.8259	0.8259	0.8585	0.8374	0.0694	0.0694	0.0694	0.0694	0.0294	0.0253	0	0	0	0	0	0	6.49	7.00	7.16	7.07	7.35	6.57
	0.7475	0.7650	0.7650	0.7650	0.8585	0.8855	0.1436	0.1545	0.1545	0.1545	0.0393	0.0156	0	0	0	0	0	0	7.32	6.93	6.85	7.32	6.49	6.71
	0.7859	0.7859	0.8259	0.8585	0.8623	0.8623	0.1673	0.1673	0.0643	0.0273	0.0163	0.0163	0	0	0	0	0	0	7.26	7.70	7.41	7.75	7.50	6.55
Average	**0.7710**	**0.7929**	**0.8099**	**0.8153**	**0.8255**	**0.8412**	**0.1233**	**0.1089**	**0.0760**	**0.0593**	**0.0618**	**0.0252**	**0**	**0**	**0**	**0**	**0**	**0**	**7.02**	**7.02**	**6.99**	**7.46**	**7.08**	**6.80**
$m = N$	0.7791	0.7791	0.8224	0.8298	0.9011	0.9011	0.1073	0.1073	0.1461	0.1181	0.0289	0.0289	0	0	0	0	0	0	7.57	8.59	8.43	9.09	8.85	8.08
	0.8094	0.8094	0.8094	0.8463	0.9193	0.9405	0.2102	0.2102	0.2102	0.0916	0.0079	0.0008	0	0	0	0	0	0	8.45	8.70	7.67	8.19	8.10	8.69
	0.7859	0.7871	0.8080	0.8080	0.8502	0.8644	0.1673	0.0877	0.0967	0.0967	0.0229	0.0081	0	0	0	0	0	0	7.55	7.97	9.39	9.15	9.02	9.19
	0.7587	0.7587	0.7543	0.7543	0.8148	0.8764	0.1778	0.1778	0.1114	0.1114	0.0523	0.0382	0	0	0	0	0	0	9.35	9.87	8.38	9.81	10.50	8.44
	0.7927	0.7927	0.8168	0.8168	0.8502	0.8502	0.1384	0.1384	0.0207	0.0207	0.0213	0.0213	0	0	0	0	0	0	8.45	8.73	7.64	7.99	9.03	8.05
	0.8094	0.8094	0.8882	0.8882	0.8714	0.9311	0.2102	0.2102	0.0440	0.0440	0.0258	0.0092	0	0	0	0	0	0	8.19	8.31	8.23	9.43	9.43	8.17
	0.7390	0.7390	0.7956	0.7956	0.8256	0.9013	0.1088	0.1088	0.0602	0.0602	0.0412	0.0058	0	0	0	0	0	0	9.10	8.88	10.17	9.81	8.35	8.47
	0.7859	0.7859	0.8465	0.8465	0.8882	0.8882	0.1677	0.1677	0.0130	0.0130	0.0337	0.0337	0	0	0	0	0	0	8.38	8.88	8.53	8.17	8.25	8.89
	0.7927	0.7927	0.8467	0.8324	0.8981	0.9193	0.1380	0.1380	0.0420	0.0220	0.0085	0.0033	0	0	0	0	0	0	9.67	9.35	9.78	9.55	8.03	8.31
	0.7625	0.7625	0.7859	0.8256	0.9193	0.9013	0.0646	0.0646	0.1704	0.0416	0.0064	0.0040	0	0	0	0	0	0	7.43	7.51	7.72	8.01	9.29	9.75
Average	**0.7815**	**0.7817**	**0.8174**	**0.8244**	**0.8738**	**0.8974**	**0.1490**	**0.1411**	**0.0915**	**0.0619**	**0.0249**	**0.0153**	**0**	**0**	**0**	**0**	**0**	**0**	**8.41**	**8.68**	**8.59**	**8.92**	**8.89**	**8.60**
$m = 1.5N$	0.7791	0.7791	0.8764	0.8764	0.8802	0.8802	0.1073	0.1073	0.0393	0.0393	0.0053	0.0053	0	0	0	0	0	0	9.26	9.48	9.45	9.40	9.29	9.86
	0.8094	0.8094	0.8882	0.8882	0.9193	0.9193	0.2102	0.2102	0.0424	0.0424	0.0031	0.0031	0	0	0	0	0	0	7.94	8.27	8.74	10.69	9.41	9.23
	0.8714	0.8714	0.9193	0.9193	1.0000	1.0000	0.0302	0.0302	0.0053	0.0053	0.0000	0.0000	0	0	0	0	0	0	8.56	8.85	10.65	9.68	9.22	9.09
	0.7650	0.7650	0.8585	0.8585	0.9703	0.9703	0.1544	0.1544	0.0403	0.0403	0.0013	0.0013	0	0	0	0	0	0	8.40	8.50	9.79	9.41	10.36	9.81
	0.8156	0.8156	0.8224	0.8855	0.9703	0.9703	0.1827	0.1827	0.1588	0.0146	0.0006	0.0006	0	0	0	0	0	0	9.28	9.11	8.32	8.12	9.79	10.51
	0.7971	0.7971	0.8714	0.8714	0.9013	0.9193	0.0443	0.0443	0.0216	0.0216	0.0092	0.0021	0	0	0	0	0	0	8.98	9.08	10.45	10.22	9.79	9.16
	0.7872	0.7872	0.8324	0.8324	0.9703	0.9311	0.0450	0.0450	0.0150	0.0150	0.0046	0.0096	0	0	0	0	0	0	9.39	10.54	9.94	10.28	10.34	9.20
	0.8802	0.8802	0.9490	0.9490	0.9013	0.9013	0.0130	0.0130	0.0066	0.0066	0.0071	0.0071	0	0	0	0	0	0	9.11	9.43	9.84	9.86	9.38	9.50
	0.7927	0.8359	0.8585	0.8585	1.0000	1.0000	0.1380	0.0569	0.0339	0.0339	0.0000	0.0000	0	0	0	0	0	0	8.72	9.01	9.11	8.90	8.83	9.00
	0.8079	0.8079	0.8802	0.8802	0.9405	0.9703	0.0335	0.0335	0.0087	0.0087	0.0058	0.0013	0	0	0	0	0	0	8.25	9.98	8.79	9.22	9.28	9.73
Average	**0.8106**	**0.8149**	**0.8756**	**0.8819**	**0.9454**	**0.9462**	**0.0959**	**0.0878**	**0.0372**	**0.0228**	**0.0037**	**0.0030**	**0**	**0**	**0**	**0**	**0**	**0**	**8.79**	**9.23**	**9.51**	**9.58**	**9.57**	**9.51**

Table B.11: Detailed performance in the *Iris* dataset.

	NMI						KL divergence						CI						Time (s)					
	$p=0.8$		$p=0.9$		$p=1.0$		$p=0.8$		$p=0.9$		$p=1.0$		$p=0.8$		$p=0.9$		$p=1.0$		$p=0.8$		$p=0.9$		$p=1.0$	
Priors:	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓
$m=0$	0.6371						0.9043						0						5.92					
$m=N/2$	0.6557	0.6469	0.6557	0.6800	0.6909	0.8248	0.7394	0.7425	0.7394	0.6854	0.6766	0.1076	0	0	0	0	0	0	10.91	11.41	10.83	11.14	11.57	10.09
	0.6667	0.6667	0.6920	0.6920	0.6706	0.7231	0.7395	0.7395	0.7516	0.7516	0.8482	0.4780	0	0	0	0	0	0	10.49	10.98	10.29	10.84	10.44	10.40
	0.5965	0.5808	0.6204	0.6285	0.6371	0.8258	1.0122	1.0557	0.9566	0.9115	0.8755	0.0668	0	0	0	0	0	0	10.94	10.57	11.60	10.88	10.17	11.04
	0.6087	0.6087	0.6130	0.6028	0.6477	0.7315	1.0949	1.0949	1.0014	1.0066	0.9272	0.4387	0	0	0	0	0	0	10.92	11.30	10.53	10.46	10.78	11.48
	0.6130	0.6130	0.6555	0.6709	0.6618	0.6898	0.9857	0.9857	0.8065	0.7882	0.8686	0.5683	0	0	0	0	0	0	11.43	11.45	10.76	11.40	10.89	10.70
	0.6237	0.6518	0.6371	0.6756	0.6706	0.6362	0.9151	0.7249	0.9043	0.6676	0.8520	0.7858	0	0	0	0	0	0	10.83	10.41	10.30	10.30	10.31	10.41
	0.6322	0.6602	0.7030	0.7030	0.7009	0.8053	0.9918	0.8171	0.7640	0.7640	0.6666	0.0745	0	0	0	0	0	0	11.23	11.69	10.87	11.03	11.22	10.44
	0.6741	0.6822	0.6741	0.6822	0.6555	0.7189	0.8452	0.7725	0.8452	0.7725	0.8065	0.5191	0	0	0	0	0	0	11.27	10.90	11.86	11.80	10.28	10.29
	0.5952	0.5952	0.5952	0.6028	0.6875	0.7279	1.0702	1.0702	1.0702	0.9928	0.6862	0.3928	0	0	0	0	0	0	11.86	11.54	11.85	10.86	10.46	10.48
	0.6608	0.6539	0.7009	0.6971	0.6628	0.7776	1.0539	1.1282	0.6791	0.5876	0.9286	0.2343	0	0	0	0	0	0	10.59	10.52	11.66	10.23	11.60	10.33
Average	**0.6327**	**0.6359**	**0.6547**	**0.6635**	**0.6685**	**0.7461**	**0.9448**	**0.9131**	**0.8518**	**0.7928**	**0.8136**	**0.3666**	**0**	**0**	**0**	**0**	**0**	**0**	**11.05**	**11.08**	**11.06**	**10.89**	**10.77**	**10.57**
$m=N$	0.6413	0.6413	0.6516	0.6516	0.7407	0.9032	0.7709	0.7709	0.7592	0.7592	0.5171	0.0071	0	0	0	0	0	0	13.62	13.54	13.82	13.62	12.65	11.90
	0.6309	0.6309	0.6276	0.6440	0.6155	0.9016	0.8176	0.8176	0.8326	0.9240	0.8618	0.0289	0	0	0	0	0	0	13.72	14.48	12.89	12.94	12.11	12.41
	0.6474	0.6652	0.6628	0.6628	0.6652	0.9276	0.8783	0.7139	0.9286	0.9286	0.7139	0.0199	0	0	0	0	0	0	12.50	11.83	12.25	12.70	13.48	12.97
	0.6800	0.6800	0.6725	0.6725	0.7208	0.7616	0.7769	0.7769	0.7792	0.7792	0.5689	0.4566	0	0	0	0	0	0	12.32	12.75	12.55	12.10	11.95	12.85
	0.6461	0.6461	0.6461	0.6461	0.7465	0.9738	0.8324	0.8324	0.8324	0.8324	0.4401	0.0117	0	0	0	0	0	0	12.55	12.34	12.29	12.51	12.85	12.37
	0.6788	0.6788	0.6875	0.6875	0.7389	0.9748	0.7897	0.7897	0.6958	0.6958	0.6844	0.0025	0	0	0	0	0	0	12.73	11.94	13.23	13.75	13.64	13.82
	0.6214	0.6214	0.6180	0.6180	0.7155	0.9223	0.9290	0.9290	0.8167	0.8167	0.7378	0.0089	0	0	0	0	0	0	13.34	13.26	12.04	12.32	12.73	13.08
	0.6706	0.6706	0.6585	0.7009	0.6564	0.9222	0.8520	0.8520	0.8585	0.6669	0.7029	0.0097	0	0	0	0	0	0	14.17	14.10	13.48	12.90	12.49	12.76
	0.6274	0.6274	0.6859	0.6971	0.6822	0.9738	0.7764	0.7764	0.6574	0.6356	0.7823	0.0015	0	0	0	0	0	0	11.29	13.13	12.01	12.91	12.23	13.41
	0.6087	0.6087	0.6628	0.6585	0.6875	0.9223	1.1193	1.1193	0.9109	0.8306	0.6862	0.0245	0	0	0	0	0	0	12.63	13.11	12.52	13.55	12.15	12.17
Average	**0.6453**	**0.6470**	**0.6573**	**0.6639**	**0.6969**	**0.9183**	**0.8543**	**0.8378**	**0.8071**	**0.7869**	**0.6695**	**0.0571**	**0**	**0**	**0**	**0**	**0**	**0**	**12.89**	**13.05**	**12.71**	**12.93**	**12.63**	**12.77**
$m=1.5N$	0.6756	0.6756	0.6972	0.6972	0.7968	1.0000	0.6676	0.6676	0.6476	0.6476	0.5515	0.0000	0	0	0	0	0	0	13.48	13.92	13.56	12.97	13.30	14.77
	0.6569	0.6569	0.6786	0.7065	0.8353	0.9747	0.7299	0.7299	0.6351	0.5976	0.4245	0.0142	0	0	0	0	0	0	14.19	15.32	13.65	13.89	15.05	13.46
	0.6467	0.6467	0.6652	0.6652	0.9384	0.9476	0.7398	0.7398	0.7139	0.7139	0.0348	0.0081	0	0	0	0	0	0	13.64	14.45	13.04	13.29	12.99	14.10
	0.6488	0.6488	0.7299	0.7115	0.8168	0.9042	0.7846	0.7846	0.6351	0.6566	0.4645	0.0848	0	0	0	0	0	0	13.23	12.94	12.80	12.81	13.80	12.74
	0.6276	0.6276	0.7074	0.7164	0.9130	1.0000	0.9055	0.9055	0.7371	0.6447	0.0375	0.0000	0	0	0	0	0	0	13.65	13.67	13.09	13.36	12.96	14.36
	0.6788	0.6875	0.6875	0.6875	0.7416	0.8403	0.7897	0.6958	0.6958	0.6958	0.6271	0.1738	0	0	0	0	0	0	14.35	13.50	14.70	14.08	13.29	13.58
	0.6690	0.6690	0.6756	0.6667	0.7432	0.9476	0.7785	0.7785	0.7305	0.7843	0.6202	0.0200	0	0	0	0	0	0	13.17	13.65	12.83	13.49	15.46	14.18
	0.6860	0.6860	0.6973	0.7294	0.6978	0.9294	0.8392	0.8392	0.8179	0.6356	0.8289	0.0150	0	0	0	0	0	0	13.78	13.72	15.66	13.82	12.21	14.46
	0.5911	0.5911	0.6569	0.6569	0.9125	0.9747	1.1613	1.1613	0.7756	0.7756	0.0519	0.0142	0	0	0	0	0	0	13.10	13.03	13.04	12.77	13.23	14.98
	0.6836	0.6836	0.6788	0.6788	0.8984	0.9214	0.8030	0.8030	0.7245	0.7245	0.0523	0.0020	0	0	0	0	0	0	12.76	14.03	13.46	13.69	13.12	15.19
Average	**0.6564**	**0.6573**	**0.6874**	**0.6916**	**0.8294**	**0.9440**	**0.8199**	**0.8105**	**0.7113**	**0.6876**	**0.3693**	**0.0332**	**0**	**0**	**0**	**0**	**0**	**0**	**13.54**	**13.82**	**13.58**	**13.42**	**13.54**	**14.18**

Table B.12: Detailed performance in the *Wine* dataset.

	NMI						KL divergence						CI						Time (s)					
	$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$	
Priors:	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓
$m = 0$	0.7282						0.0287						0						6.87					
$m = N/2$	0.7459	0.7459	0.7459	0.7459	0.8029	0.8384	0.0238	0.0238	0.0238	0.0238	0.0223	0.0121	0	0	0	0	0	0	14.02	15.12	13.60	13.09	14.56	12.83
	0.7158	0.7282	0.7282	0.7653	0.8118	0.8448	0.0343	0.0272	0.0292	0.0205	0.0476	0.0498	0	0	0	0	0	0	15.47	15.61	14.38	13.72	14.84	12.36
	0.7504	0.7721	0.8029	0.8029	0.8228	0.8704	0.0210	0.0217	0.0223	0.0223	0.0246	0.0053	0	0	0	0	0	0	14.03	15.78	17.53	16.70	15.58	13.13
	0.7282	0.7282	0.7282	0.7282	0.7575	0.7907	0.0287	0.0287	0.0287	0.0287	0.0398	0.0453	0	0	0	0	0	0	15.18	17.04	15.37	14.36	13.38	12.36
	0.7459	0.7459	0.7494	0.7619	0.7871	0.8000	0.0253	0.0253	0.0293	0.0256	0.0271	0.0123	0	0	0	0	0	0	12.43	13.53	15.53	17.33	13.87	11.33
	0.7282	0.7282	0.7860	0.7860	0.8424	0.8544	0.0275	0.0275	0.0433	0.0433	0.0310	0.0126	0	0	0	0	0	0	14.79	15.37	14.91	15.78	15.38	12.24
	0.7282	0.7282	0.7282	0.7282	0.8384	0.8969	0.0287	0.0287	0.0287	0.0287	0.0194	0.0070	0	0	0	0	0	0	15.60	15.22	15.49	15.79	14.06	12.29
	0.7813	0.7976	0.7836	0.7836	0.8350	0.8815	0.0219	0.0222	0.0257	0.0257	0.0292	0.0105	0	0	0	0	0	0	15.35	15.51	15.57	14.75	14.97	12.13
	0.7459	0.7459	0.7976	0.7976	0.8339	0.8468	0.0253	0.0253	0.0225	0.0225	0.0301	0.0324	0	0	0	0	0	0	15.36	16.98	15.72	14.89	16.78	13.60
	0.7494	0.7494	0.7494	0.7494	0.7718	0.7718	0.0309	0.0309	0.0309	0.0309	0.0299	0.0299	0	0	0	0	0	0	15.78	16.28	15.45	16.18	14.51	11.33
Average	**0.7419**	**0.7470**	**0.7599**	**0.7649**	**0.8104**	**0.8396**	**0.0267**	**0.0261**	**0.0284**	**0.0272**	**0.0301**	**0.0217**	**0**	**0**	**0**	**0**	**0**	**0**	**14.80**	**15.64**	**15.36**	**15.26**	**14.79**	**12.36**
$m = N$	0.7836	0.7619	0.8228	0.8228	0.8973	0.9410	0.0276	0.0229	0.0267	0.0267	0.0247	0.0107	0	0	0	0	0	0	20.75	19.74	19.13	19.54	18.82	15.37
	0.7871	0.7871	0.8000	0.8216	0.8903	0.9741	0.0299	0.0299	0.0266	0.0246	0.0095	0.0012	0	0	0	0	0	0	19.66	20.96	18.90	18.47	16.94	14.75
	0.7977	0.7871	0.8433	0.8209	0.9412	0.9749	0.0374	0.0270	0.0351	0.0303	0.0066	0.0012	0	0	0	0	0	0	17.96	18.51	17.82	20.68	15.86	14.98
	0.7619	0.7619	0.7976	0.7976	0.8330	0.8832	0.0275	0.0275	0.0218	0.0218	0.0231	0.0111	0	0	0	0	0	0	19.38	19.39	16.61	19.00	17.23	14.41
	0.7813	0.7813	0.8000	0.8228	0.9309	0.8982	0.0236	0.0236	0.0252	0.0252	0.0257	0.0344	0	0	0	0	0	0	20.27	20.39	17.10	17.08	18.13	15.77
	0.7860	0.7860	0.7971	0.7971	0.8819	0.9750	0.0464	0.0464	0.0383	0.0383	0.0295	0.0095	0	0	0	0	0	0	17.85	18.59	18.56	16.89	17.49	16.15
	0.8228	0.8000	0.8484	0.8484	0.9156	0.9136	0.0267	0.0245	0.0268	0.0268	0.0325	0.0211	0	0	0	0	0	0	19.76	19.92	17.66	16.83	17.00	16.84
	0.7625	0.7625	0.8195	0.8195	0.8715	0.9750	0.0272	0.0272	0.0335	0.0335	0.0264	0.0095	0	0	0	0	0	0	16.64	17.41	17.67	17.81	14.98	14.69
	0.7813	0.7813	0.7813	0.7813	0.7625	0.8967	0.0236	0.0236	0.0236	0.0236	0.0257	0.0123	0	0	0	0	0	0	20.56	19.10	20.42	19.87	17.72	14.25
	0.7494	0.7494	0.7619	0.8279	0.8897	0.9491	0.0291	0.0291	0.0256	0.0164	0.0475	0.0121	0	0	0	0	0	0	19.17	19.72	20.90	20.27	20.43	15.09
Average	**0.7814**	**0.7759**	**0.8072**	**0.8160**	**0.8814**	**0.9381**	**0.0299**	**0.0282**	**0.0283**	**0.0267**	**0.0251**	**0.0123**	**0**	**0**	**0**	**0**	**0**	**0**	**19.20**	**19.37**	**18.48**	**18.64**	**17.46**	**15.23**
$m = 1.5N$	0.7976	0.7976	0.8225	0.8225	0.8973	0.9750	0.0225	0.0225	0.0229	0.0229	0.0249	0.0095	0	0	0	0	0	0	21.01	21.34	20.29	20.86	19.19	16.70
	0.8216	0.8216	0.8487	0.8487	0.9156	1.0000	0.0246	0.0246	0.0281	0.0281	0.0307	0.0000	0	0	0	0	0	0	16.17	17.67	19.80	21.50	21.78	19.14
	0.7488	0.7721	0.8377	0.8715	0.9156	0.9576	0.0327	0.0215	0.0229	0.0264	0.0323	0.0179	0	0	0	0	0	0	18.60	18.76	20.78	20.78	21.06	17.75
	0.7619	0.7619	0.7619	0.7619	0.8972	0.9240	0.0275	0.0275	0.0275	0.0275	0.0215	0.0058	0	0	0	0	0	0	19.64	21.08	17.94	18.71	19.75	18.11
	0.7813	0.7813	0.8393	0.8393	0.9410	0.9410	0.0236	0.0236	0.0375	0.0375	0.0107	0.0107	0	0	0	0	0	0	21.63	22.19	20.97	20.88	18.68	18.08
	0.7971	0.7971	0.8350	0.8484	0.8815	0.8815	0.0377	0.0377	0.0336	0.0268	0.0089	0.0089	0	0	0	0	0	0	20.08	18.43	19.67	18.16	21.06	17.59
	0.8000	0.8000	0.8484	0.8484	0.9491	1.0000	0.0245	0.0245	0.0268	0.0268	0.0170	0.0000	0	0	0	0	0	0	21.26	21.22	19.76	20.64	18.81	18.28
	0.7976	0.7976	0.7976	0.7976	0.9153	0.9155	0.0222	0.0222	0.0222	0.0222	0.0104	0.0015	0	0	0	0	0	0	22.17	22.18	20.17	18.13	23.33	16.49
	0.8000	0.8000	0.8377	0.8377	0.9153	0.9410	0.0260	0.0260	0.0229	0.0229	0.0173	0.0110	0	0	0	0	0	0	18.70	21.47	20.23	20.99	20.89	17.14
	0.7619	0.7619	0.7871	0.7871	0.8972	0.9749	0.0256	0.0256	0.0271	0.0271	0.0203	0.0010	0	0	0	0	0	0	19.83	20.82	19.38	21.55	21.33	17.84
Average	**0.7868**	**0.7891**	**0.8216**	**0.8263**	**0.9125**	**0.9511**	**0.0267**	**0.0256**	**0.0272**	**0.0268**	**0.0194**	**0.0066**	**0**	**0**	**0**	**0**	**0**	**0**	**19.91**	**20.52**	**19.90**	**20.22**	**20.59**	**17.71**

Table B.13: Detailed performance in the *Thyroid* dataset.

	NMI						KL divergence						CI						Time (s)					
	$p=0.8$		$p=0.9$		$p=1.0$		$p=0.8$		$p=0.9$		$p=1.0$		$p=0.8$		$p=0.9$		$p=1.0$		$p=0.8$		$p=0.9$		$p=1.0$	
Priors:	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓
$m=0$	0.4146						1.4761						1						14.20					
$m=N/2$	0.4249	0.4282	0.4282	0.4415	0.4623	0.4636	1.4391	1.4155	1.4155	1.3598	1.2939	0.9949	1	1	1	1	1	1	27.47	26.89	29.37	31.31	23.72	27.06
	0.4168	0.4135	0.4218	0.4280	0.4240	0.4096	1.4668	1.4796	1.4696	1.4689	1.4445	1.2361	1	1	1	1	1	1	27.64	31.72	26.22	31.63	32.45	26.42
	0.4388	0.4344	0.4349	0.4304	0.4395	0.4260	1.3891	1.4191	1.4019	1.4321	1.4134	1.2092	1	1	1	1	1	1	29.50	27.94	28.32	33.89	24.01	24.80
	0.4270	0.4270	0.4292	0.4189	0.4328	0.4569	1.4468	1.4468	1.4613	1.4588	1.4816	1.0463	1	1	1	1	1	1	29.07	27.86	26.14	25.62	27.57	26.94
	0.4125	0.4125	0.4115	0.4220	0.4125	0.4658	1.4855	1.4855	1.4878	1.4592	1.4855	0.6260	1	1	1	1	1	1	26.50	27.78	28.65	27.46	28.32	26.71
	0.4077	0.4077	0.4263	0.4435	0.4263	0.4770	1.4951	1.4951	1.4248	1.4333	1.4192	0.4422	1	1	1	1	1	0	25.19	27.63	26.29	25.67	23.30	26.70
	0.4406	0.4461	0.4579	0.4579	0.4556	0.4226	1.3796	1.3987	1.3624	1.3624	1.3782	0.9675	1	1	1	1	1	1	27.16	25.77	23.26	27.85	32.05	28.43
	0.4136	0.4136	0.4718	0.4713	0.4306	0.4406	1.4767	1.4767	1.2590	1.2004	1.4871	0.5736	1	1	1	1	1	1	24.26	26.38	25.56	24.63	25.92	27.68
	0.4431	0.4431	0.4736	0.4507	0.5028	0.4231	1.4264	1.4264	1.3078	1.3878	1.1933	1.1923	1	1	1	1	1	1	26.14	28.22	22.84	25.42	29.41	29.61
	0.4146	0.4187	0.4177	0.4309	0.4482	0.4783	1.4653	1.4606	1.4539	1.4862	1.4720	0.9263	1	1	1	1	1	1	29.16	27.49	27.23	27.62	24.30	27.30
Average	**0.4240**	**0.4245**	**0.4373**	**0.4395**	**0.4435**	**0.4464**	**1.4470**	**1.4504**	**1.4044**	**1.4049**	**1.4069**	**0.9214**	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	**0.9**	**27.21**	**27.77**	**26.39**	**28.11**	**27.11**	**27.17**
$m=N$	0.4136	0.4136	0.4197	0.4197	0.4903	0.6532	1.4767	1.4767	1.4603	1.4603	1.1553	0.0433	1	1	1	1	1	0	31.72	31.70	30.50	35.65	34.66	34.36
	0.4294	0.4158	0.4298	0.4605	0.5446	0.5694	1.4605	1.4653	1.4872	1.3609	1.0387	0.2292	1	1	1	1	1	0	35.68	34.77	31.71	28.30	33.41	38.54
	0.4428	0.4428	0.4926	0.4926	0.5032	0.6077	1.4065	1.4065	1.3109	1.3109	1.1342	0.2403	1	1	1	1	1	0	31.53	40.03	41.16	35.14	31.29	40.24
	0.4213	0.4213	0.4851	0.4724	0.5583	0.6183	1.4416	1.4416	1.1790	1.2227	1.0757	0.1775	1	1	1	1	1	0	34.11	27.80	34.61	37.75	36.86	44.05
	0.4406	0.4467	0.4640	0.4640	0.5365	0.5788	1.3650	1.3622	1.2856	1.2856	1.0301	0.3104	1	1	1	1	1	0	32.22	37.16	35.71	32.67	32.35	32.24
	0.4073	0.4022	0.4401	0.4638	0.5171	0.5322	1.4915	1.5116	1.4425	1.0143	1.0296	0.0781	1	1	1	1	1	0	31.32	29.70	30.57	28.26	32.93	37.91
	0.4077	0.4077	0.4136	0.4182	0.5391	0.6194	1.4989	1.4989	1.4874	1.5008	1.1379	0.0472	1	1	1	1	1	0	32.83	33.20	30.48	31.52	37.89	36.38
	0.4231	0.4231	0.4872	0.4872	0.5286	0.5781	1.4399	1.4399	1.2141	1.2141	1.1055	0.2284	1	1	1	1	1	0	32.40	31.74	28.39	33.73	32.10	38.51
	0.4783	0.4783	0.4975	0.4961	0.4829	0.5193	1.2582	1.2582	1.1448	1.1511	1.1924	0.5395	1	1	1	1	1	1	32.44	31.74	32.77	27.38	26.17	45.71
	0.4117	0.4106	0.4310	0.4182	0.5320	0.6288	1.4865	1.4892	1.4574	1.4909	1.1469	0.1796	1	1	1	1	1	0	32.57	37.23	32.87	33.38	29.05	36.71
Average	**0.4276**	**0.4262**	**0.4561**	**0.4593**	**0.5233**	**0.5905**	**1.4325**	**1.4350**	**1.3469**	**1.3012**	**1.1046**	**0.2074**	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	**0.1**	**32.68**	**33.51**	**32.88**	**32.38**	**32.67**	**38.47**
$m=1.5N$	0.4355	0.4366	0.4752	0.4752	0.5811	0.7204	1.4474	1.4456	1.2627	1.2627	1.0958	0.0572	1	1	1	1	1	0	34.30	38.16	31.33	34.66	38.32	41.70
	0.4219	0.4219	0.5510	0.5510	0.5522	0.7943	1.4865	1.4865	1.1169	1.1169	1.0723	0.0085	1	1	1	1	1	0	34.66	35.37	30.87	35.01	32.81	42.22
	0.4533	0.4467	0.4796	0.4796	0.5718	0.7178	1.3492	1.3618	1.2684	1.2684	1.0628	0.0439	1	1	1	1	1	0	40.78	40.92	33.97	32.99	29.61	40.94
	0.4325	0.4325	0.4752	0.4647	0.5750	0.6757	1.4414	1.4414	1.2326	1.2351	0.9853	0.0747	1	1	1	1	1	0	31.48	33.46	34.10	34.29	33.69	39.26
	0.4415	0.4415	0.4711	0.4790	0.6133	0.7097	1.3602	1.3602	1.2816	1.2637	0.9740	0.0240	1	1	1	1	1	0	32.97	33.77	33.28	32.63	33.12	41.87
	0.4892	0.4892	0.5191	0.5191	0.5572	0.6823	1.1249	1.1249	1.0823	1.0823	1.0634	0.1320	1	1	1	1	1	0	34.76	34.12	29.56	30.21	35.03	40.79
	0.4292	0.4292	0.4612	0.4621	0.5140	0.6456	1.4626	1.4626	1.3429	1.3392	1.0826	0.2784	1	1	1	1	1	0	37.35	36.11	31.15	34.36	33.98	37.70
	0.4585	0.4585	0.4625	0.4904	0.5598	0.7890	1.0742	1.0742	1.1632	1.0893	1.0945	0.0157	1	1	1	1	1	0	29.64	29.77	32.23	32.73	33.50	39.35
	0.4892	0.4892	0.5164	0.5158	0.5897	0.8221	1.1342	1.1342	1.2011	1.2014	1.0443	0.0139	1	1	1	1	1	0	35.99	36.95	29.53	28.28	31.62	36.96
	0.4360	0.4338	0.4338	0.4338	0.6215	0.6746	1.4736	1.4786	1.4786	1.4786	0.9839	0.0601	1	1	1	1	1	0	41.18	36.65	33.27	33.83	33.47	39.37
Average	**0.4487**	**0.4479**	**0.4845**	**0.4871**	**0.5736**	**0.7232**	**1.3354**	**1.3370**	**1.2430**	**1.2338**	**1.0459**	**0.0708**	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	**0.0**	**35.31**	**35.53**	**31.93**	**32.90**	**33.52**	**40.02**

Table B.14: Detailed performance in the *Vertebral* dataset.

	NMI						KL divergence						CI						Time (s)					
	$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$	
Priors:	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓
$m = 0$	0.6925						1.4806						3						25.07					
$m = N/2$	0.6938	0.7108	0.7125	0.7358	0.6905	0.7025	1.4580	1.3642	1.2019	1.0666	1.5288	1.0176	3	3	3	3	3	3	75.02	81.80	76.67	73.20	86.56	80.76
	0.6850	0.6859	0.6779	0.7055	0.7023	0.6855	1.4948	1.4581	1.5617	1.2627	1.2667	1.1698	3	3	3	3	3	3	87.28	85.47	77.90	78.23	82.36	81.41
	0.6822	0.6849	0.6883	0.7133	0.6994	0.7142	1.5776	1.5652	1.5192	1.1667	1.2869	0.9799	3	3	3	3	3	2	76.77	84.08	77.74	77.92	85.69	77.75
	0.6705	0.6694	0.7062	0.6680	0.6811	0.7544	1.4170	1.4367	1.2321	1.4177	1.8462	0.6812	3	3	3	3	4	3	74.86	77.87	79.57	77.09	79.05	74.68
	0.7055	0.6962	0.7033	0.7111	0.6975	0.7180	1.4822	1.4630	1.2103	1.1662	1.4053	0.7711	3	3	3	3	3	2	77.34	85.52	80.34	80.65	89.60	90.73
	0.6968	0.6955	0.6768	0.7058	0.6847	0.7386	1.3376	1.3396	1.3825	1.4303	1.4362	0.7147	3	3	3	3	3	2	80.22	76.09	76.39	71.18	79.62	83.15
	0.6866	0.6942	0.6999	0.7002	0.7000	0.6952	1.2533	1.3349	1.2965	1.4334	1.2359	0.9430	3	3	3	3	3	3	81.61	82.28	81.04	80.40	91.15	85.49
	0.6801	0.6812	0.6711	0.7015	0.7000	0.7463	1.1141	1.1154	1.3313	1.1211	1.4007	0.8797	3	3	3	3	3	2	83.11	82.82	78.49	81.41	86.74	80.52
	0.6934	0.6869	0.6957	0.6935	0.7243	0.6994	1.3874	1.4192	1.4889	1.4406	1.0576	0.8449	3	3	3	3	2	2	79.10	83.06	76.30	83.20	86.38	85.05
	0.6855	0.6866	0.7047	0.6904	0.7024	0.7317	1.1749	1.1788	1.2433	1.1792	1.2465	0.8566	3	3	3	3	3	3	81.77	81.12	79.98	79.31	89.30	80.42
Average	**0.6879**	**0.6892**	**0.6936**	**0.7025**	**0.6982**	**0.7186**	**1.3697**	**1.3675**	**1.3468**	**1.2685**	**1.3711**	**0.8859**	**3.0**	**3.0**	**3.0**	**3.0**	**3.0**	**2.5**	**79.71**	**82.01**	**78.44**	**78.26**	**85.65**	**82.00**
$m = N$	0.6961	0.7045	0.6881	0.7374	0.6937	0.7839	1.2931	1.4545	1.5246	1.0705	1.0902	0.4840	3	3	4	3	3	2	102.54	97.71	95.06	96.23	106.53	100.08
	0.6806	0.6831	0.7075	0.7282	0.7219	0.7828	1.3058	1.3316	1.2748	1.2473	1.1612	0.2981	3	3	3	3	2	2	105.00	104.91	111.50	106.34	112.95	100.79
	0.6870	0.6896	0.6944	0.6966	0.7180	0.7451	1.2781	1.2372	1.2866	1.1476	1.1059	0.5182	3	3	3	3	2	2	98.42	95.92	100.52	96.43	104.80	115.87
	0.7105	0.7090	0.7139	0.7248	0.7373	0.7528	1.0820	1.0835	1.2004	1.1592	1.1548	0.6173	3	3	3	3	3	3	93.14	100.06	100.22	100.48	110.69	114.75
	0.6711	0.6743	0.6846	0.6801	0.7036	0.7039	1.4705	1.2959	1.2500	1.1162	1.3097	0.7538	3	3	3	2	3	3	97.15	100.02	94.98	98.77	108.39	113.85
	0.6473	0.6566	0.6946	0.7139	0.7178	0.7316	1.5762	1.5340	1.2268	1.0429	1.2505	0.6275	3	3	3	3	3	3	99.01	105.21	104.46	96.50	112.40	112.81
	0.6900	0.6848	0.6762	0.6965	0.7146	0.7253	1.2821	1.2831	1.4354	1.4383	1.2301	0.6694	3	3	3	3	3	3	101.83	108.94	102.58	103.71	107.82	110.98
	0.6949	0.7080	0.7102	0.7338	0.7311	0.7726	1.3054	1.2289	1.5520	1.2034	1.1145	0.6032	3	3	3	3	3	2	101.11	98.41	105.23	101.94	101.75	93.58
	0.6976	0.6978	0.6957	0.7015	0.7295	0.7691	1.3776	1.3786	1.3643	1.3451	1.2603	0.5128	3	3	3	3	3	2	108.16	107.61	112.64	104.49	106.44	102.48
	0.6865	0.6805	0.6818	0.6826	0.7465	0.7677	1.2972	1.3057	1.4123	1.4290	1.0912	0.7364	3	3	3	3	3	3	98.13	97.73	96.65	95.08	113.38	99.27
Average	**0.6862**	**0.6888**	**0.6947**	**0.7095**	**0.7214**	**0.7535**	**1.3268**	**1.3133**	**1.3527**	**1.2200**	**1.1768**	**0.5821**	**3.0**	**3.0**	**3.1**	**2.9**	**2.8**	**2.5**	**100.45**	**101.65**	**102.38**	**100.00**	**108.52**	**106.45**
$m = 1.5N$	0.6920	0.7110	0.6885	0.7330	0.7367	0.7962	1.2253	1.2022	1.3020	1.1487	1.0564	0.2164	2	3	3	3	2	2	101.79	113.02	102.13	108.72	101.27	103.39
	0.6805	0.6924	0.7074	0.7203	0.7547	0.8147	1.1970	1.1798	1.1906	1.1990	1.1419	0.4498	3	3	3	3	2	2	105.64	101.67	112.46	113.83	104.80	93.61
	0.6930	0.7096	0.7068	0.7226	0.7464	0.7908	1.5470	1.2166	1.3288	1.2863	1.0767	0.4731	3	3	3	3	3	3	106.56	103.89	109.16	109.15	100.58	103.49
	0.6951	0.6953	0.7238	0.7293	0.7441	0.7686	1.2519	1.2536	1.1685	1.1510	1.1198	0.9286	3	3	3	3	2	3	98.00	102.40	100.65	99.82	112.30	107.49
	0.6710	0.6646	0.7035	0.7007	0.7311	0.7897	1.4812	1.5430	1.1013	1.1645	1.2080	0.4779	3	3	3	3	3	2	103.67	104.53	100.20	99.26	117.74	116.18
	0.6641	0.6701	0.6983	0.7186	0.7167	0.8258	1.5580	1.5272	1.4431	1.1062	1.2469	0.2725	3	3	3	2	3	2	107.46	111.78	116.13	115.36	112.04	109.56
	0.7169	0.7167	0.7123	0.7170	0.7422	0.7776	1.2298	1.2314	1.1901	1.0670	1.0595	0.3456	3	3	3	3	2	2	115.58	110.55	112.28	114.71	108.45	105.84
	0.7033	0.7132	0.7079	0.7282	0.7462	0.7793	1.1358	1.0681	1.2131	1.0762	1.1076	0.7842	3	3	3	3	2	3	108.53	107.29	110.15	103.81	102.45	99.98
	0.6947	0.6899	0.7021	0.7089	0.7240	0.7712	1.3975	1.4043	1.3183	1.3468	1.1285	0.7157	3	3	3	3	3	2	109.75	128.90	103.27	102.34	114.29	99.74
	0.7045	0.7009	0.7102	0.7100	0.7490	0.7908	1.1971	1.1592	1.0634	1.0633	1.3021	0.6038	3	3	2	2	3	2	107.81	126.80	117.06	122.11	108.96	108.27
Average	**0.6915**	**0.6964**	**0.7061**	**0.7189**	**0.7391**	**0.7905**	**1.3221**	**1.2785**	**1.2319**	**1.1609**	**1.1447**	**0.5268**	**2.9**	**3.0**	**2.9**	**2.8**	**2.5**	**2.3**	**106.48**	**111.08**	**108.35**	**108.91**	**108.29**	**104.76**

Table B.15: Detailed performance in the *E. coli* dataset.

	NMI						KL divergence						CI						Time (s)					
	$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$	
Priors:	✗	✓	✗	✓	✗	✓	✗	✓	✗	✓	✗	✓	✗	✓	✗	✓	✗	✓	✗	✓	✗	✓	✗	✓
$m = 0$	0.7340						0.5936						0						14.29					
$m = N/2$	0.7541	0.7541	0.7541	0.7541	0.7541	0.8586	0.5921	0.5921	0.5921	0.5921	0.5921	0.2721	0	0	0	0	0	0	70.62	70.85	70.63	72.39	65.58	61.06
	0.7524	0.7431	0.7571	0.7571	0.7873	0.8238	0.5399	0.5663	0.5255	0.5255	0.4910	0.3506	0	0	0	0	0	0	68.60	64.44	61.45	65.16	63.14	66.92
	0.7431	0.7431	0.7632	0.7632	0.7775	0.8350	0.5661	0.5661	0.5633	0.5633	0.5146	0.3315	0	0	0	0	0	0	67.17	72.79	66.28	73.99	63.86	66.18
	0.7385	0.7385	0.7431	0.7431	0.7667	0.7766	0.5792	0.5792	0.5661	0.5661	0.4941	0.4428	0	0	0	0	0	0	65.33	67.77	70.07	65.69	66.68	71.11
	0.7340	0.7340	0.7431	0.7431	0.7619	0.8308	0.5936	0.5936	0.5663	0.5663	0.5083	0.2912	0	0	0	0	0	0	68.86	67.71	68.13	69.72	64.49	69.16
	0.7431	0.7431	0.7632	0.7632	0.7727	0.8407	0.5661	0.5661	0.5658	0.5658	0.5331	0.2872	0	0	0	0	0	0	70.34	67.00	64.91	66.61	67.93	69.53
	0.7431	0.7431	0.7586	0.7586	0.7571	0.8250	0.5662	0.5662	0.5788	0.5788	0.5255	0.3184	0	0	0	0	0	0	63.73	63.45	67.57	64.37	69.98	62.65
	0.7340	0.7340	0.7340	0.7385	0.7385	0.7973	0.5936	0.5936	0.5936	0.5762	0.5792	0.3738	0	0	0	0	0	0	69.80	65.25	59.98	62.34	65.64	61.39
	0.7385	0.7385	0.7477	0.7477	0.7619	0.8192	0.5802	0.5802	0.5495	0.5495	0.5027	0.3241	0	0	0	0	0	0	65.05	71.44	58.30	60.29	63.50	67.41
	0.7431	0.7431	0.7667	0.7667	0.7727	0.7973	0.5639	0.5639	0.4908	0.4908	0.5329	0.4488	0	0	0	0	0	0	65.96	65.54	60.83	61.00	69.39	65.16
Average	**0.7424**	**0.7415**	**0.7531**	**0.7535**	**0.7650**	**0.8204**	**0.5741**	**0.5767**	**0.5592**	**0.5574**	**0.5274**	**0.3441**	**0**	**0**	**0**	**0**	**0**	**0**	**67.55**	**67.62**	**64.82**	**66.16**	**66.02**	**66.06**
$m = N$	0.7431	0.7431	0.7667	0.7667	0.8025	0.8842	0.5662	0.5662	0.4942	0.4942	0.4447	0.1983	0	0	0	0	0	0	88.72	88.58	81.68	79.49	81.74	77.01
	0.7524	0.7524	0.7571	0.7619	0.7823	0.8711	0.5362	0.5362	0.5241	0.5094	0.5057	0.2104	0	0	0	0	0	0	89.10	86.38	75.75	76.86	89.59	83.34
	0.7524	0.7524	0.7716	0.7716	0.7775	0.8711	0.5399	0.5399	0.4791	0.4791	0.5212	0.2201	0	0	0	0	0	0	88.06	86.72	79.78	82.82	79.68	83.76
	0.7524	0.7524	0.7524	0.7524	0.7431	0.8250	0.5372	0.5372	0.5372	0.5372	0.5661	0.2901	0	0	0	0	0	0	81.87	83.16	89.07	90.52	82.88	82.71
	0.7477	0.7477	0.7727	0.7727	0.7823	0.8586	0.5453	0.5453	0.5314	0.5314	0.5009	0.2844	0	0	0	0	0	0	93.31	90.42	89.93	88.63	74.49	70.29
	0.7541	0.7541	0.7727	0.7727	0.8077	0.8842	0.5921	0.5921	0.5358	0.5358	0.4318	0.1961	0	0	0	0	0	0	87.53	88.77	80.88	83.57	85.52	74.65
	0.7716	0.7716	0.7766	0.7766	0.7868	0.8648	0.4787	0.4787	0.4628	0.4628	0.4310	0.2405	0	0	0	0	0	0	82.71	87.99	74.87	78.82	77.98	89.41
	0.7632	0.7632	0.7775	0.7775	0.7775	0.8776	0.5658	0.5658	0.5222	0.5222	0.5222	0.1993	0	0	0	0	0	0	85.06	89.78	82.60	87.34	80.32	79.72
	0.7477	0.7477	0.7727	0.7716	0.8026	0.8617	0.5512	0.5512	0.5307	0.4752	0.3896	0.2076	0	0	0	0	0	0	93.40	86.24	79.91	86.80	78.17	79.72
	0.7495	0.7495	0.7586	0.7586	0.7775	0.8910	0.6027	0.6027	0.5759	0.5759	0.5178	0.1584	0	0	0	0	0	0	87.41	87.43	90.98	91.96	86.46	78.64
Average	**0.7534**	**0.7534**	**0.7679**	**0.7682**	**0.7840**	**0.8689**	**0.5515**	**0.5515**	**0.5193**	**0.5123**	**0.4831**	**0.2205**	**0**	**0**	**0**	**0**	**0**	**0**	**87.72**	**87.55**	**82.55**	**84.68**	**81.68**	**79.93**
$m = 1.5N$	0.7727	0.7727	0.7775	0.7873	0.8025	0.9051	0.5323	0.5323	0.5192	0.4864	0.4447	0.1325	0	0	0	0	0	0	94.52	94.35	91.63	94.63	94.49	81.66
	0.7385	0.7385	0.7667	0.7817	0.8077	0.8617	0.5792	0.5792	0.4942	0.4451	0.4318	0.2025	0	0	0	0	0	0	96.48	97.49	87.45	88.14	82.37	94.43
	0.7766	0.7766	0.7920	0.7920	0.8130	0.8490	0.4648	0.4648	0.4209	0.4209	0.4083	0.2494	0	0	0	0	0	0	90.47	108.15	89.29	90.37	91.37	80.63
	0.7524	0.7524	0.7571	0.7571	0.8350	0.8910	0.5372	0.5372	0.5230	0.5230	0.3499	0.1640	0	0	0	0	0	0	99.11	109.85	97.23	97.23	87.14	84.65
	0.7571	0.7571	0.7727	0.7727	0.8238	0.9280	0.5198	0.5198	0.5314	0.5314	0.3846	0.0806	0	0	0	0	0	0	97.51	95.76	96.23	97.25	83.90	86.41
	0.7679	0.7679	0.7727	0.7727	0.8350	0.8648	0.5520	0.5520	0.5358	0.5358	0.3486	0.2635	0	0	0	0	0	0	97.47	110.15	91.61	90.56	83.45	81.53
	0.7716	0.7716	0.7920	0.7920	0.7973	0.9051	0.4787	0.4787	0.4198	0.4198	0.4072	0.1312	0	0	0	0	0	0	116.33	101.00	86.87	87.23	87.02	83.69
	0.7679	0.7679	0.7823	0.7823	0.8130	0.8684	0.5501	0.5501	0.5065	0.5065	0.4144	0.1961	0	0	0	0	0	0	107.33	103.90	97.11	89.42	87.51	88.19
	0.7619	0.7619	0.7868	0.7868	0.8130	0.8842	0.5061	0.5061	0.4336	0.4336	0.4152	0.1830	0	0	0	0	0	0	93.98	105.41	91.76	93.52	86.76	98.49
	0.7632	0.7632	0.7923	0.7923	0.8130	0.8842	0.5597	0.5597	0.4740	0.4740	0.4166	0.1978	0	0	0	0	0	0	90.65	91.44	86.91	91.67	91.40	81.60
Average	**0.7630**	**0.7630**	**0.7792**	**0.7817**	**0.8153**	**0.8842**	**0.5280**	**0.5280**	**0.4858**	**0.4777**	**0.4021**	**0.1801**	**0**	**0**	**0**	**0**	**0**	**0**	**98.39**	**101.75**	**91.61**	**92.00**	**87.54**	**86.13**

Table B.16: Detailed performance in the *Breast-Cancer* dataset.

	NMI						KL divergence						CI						Time (s)					
	$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$		$p = 0.8$		$p = 0.9$		$p = 1.0$	
Priors:	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓	×	✓
$m = 0$	0.8071						0.2458						0						136.86					
$m = N/2$	0.8105	0.8105	0.8141	0.8154	0.8381	0.8706	0.0968	0.0968	0.0933	0.0927	0.0758	0.0511	0	0	0	0	0	0	1,147.68	968.75	895.83	879.80	769.63	789.27
	0.8151	0.8151	0.8198	0.8198	0.8273	0.8642	0.0983	0.0983	0.0915	0.0915	0.0746	0.0366	0	0	0	0	0	0	1,083.89	884.08	1,042.28	875.87	811.59	735.89
	0.8155	0.8155	0.8249	0.8249	0.8282	0.8760	0.0948	0.0948	0.0808	0.0808	0.0715	0.0302	0	0	0	0	0	0	986.05	875.10	1,000.33	820.75	713.97	650.07
	0.8157	0.8157	0.8227	0.8227	0.8272	0.8696	0.0942	0.0942	0.0894	0.0894	0.0756	0.0364	0	0	0	0	0	0	981.10	888.37	857.55	827.94	844.70	727.19
	0.8211	0.8211	0.8211	0.8223	0.8360	0.8697	0.0879	0.0879	0.0879	0.0858	0.0679	0.0366	0	0	0	0	0	0	891.40	829.38	937.85	794.32	795.41	699.30
	0.8120	0.8120	0.8165	0.8165	0.8469	0.8750	0.0969	0.0969	0.0914	0.0914	0.0542	0.0321	0	0	0	0	0	0	977.22	834.05	907.75	948.50	760.09	678.24
	0.8095	0.8107	0.8129	0.8144	0.8272	0.8658	0.0954	0.0949	0.0924	0.0921	0.0717	0.0489	0	0	0	0	0	0	927.10	886.35	864.47	836.26	851.53	806.77
	0.8140	0.8140	0.8201	0.8188	0.8323	0.8791	0.0913	0.0913	0.0868	0.0871	0.0762	0.0464	0	0	0	0	0	0	833.60	821.50	878.22	837.32	758.99	744.93
	0.8143	0.8143	0.8229	0.8243	0.8363	0.8709	0.0929	0.0929	0.0887	0.0883	0.0721	0.0460	0	0	0	0	0	0	872.88	864.70	788.52	842.31	750.74	826.89
	0.8166	0.8166	0.8177	0.8165	0.8394	0.8732	0.0968	0.0968	0.0941	0.0945	0.0709	0.0369	0	0	0	0	0	0	941.72	871.58	914.40	785.72	723.39	677.66
Average	**0.8144**	**0.8146**	**0.8193**	**0.8196**	**0.8339**	**0.8714**	**0.0945**	**0.0945**	**0.0896**	**0.0894**	**0.0711**	**0.0401**	**0**	**0**	**0**	**0**	**0**	**0**	**964.26**	**872.39**	**908.72**	**844.88**	**778.00**	**733.62**
$m = N$	0.8273	0.8273	0.8297	0.8297	0.8594	0.9210	0.0799	0.0799	0.0751	0.0751	0.0492	0.0131	0	0	0	0	0	0	1,082.26	1,070.89	1,032.30	1,004.04	1,007.58	933.28
	0.8368	0.8368	0.8555	0.8515	0.8568	0.8948	0.0740	0.0740	0.0535	0.0536	0.0540	0.0207	0	0	0	0	0	0	1,099.53	1,124.89	992.18	925.61	833.62	999.78
	0.8285	0.8298	0.8453	0.8466	0.8721	0.9309	0.0786	0.0782	0.0646	0.0629	0.0452	0.0132	0	0	0	0	0	0	1,367.16	1,089.95	955.75	1,056.47	1,047.89	959.94
	0.8169	0.8169	0.8339	0.8339	0.8615	0.9302	0.0913	0.0913	0.0716	0.0716	0.0496	0.0081	0	0	0	0	0	0	1,292.67	1,141.17	1,081.45	1,091.68	879.37	820.24
	0.8189	0.8189	0.8335	0.8335	0.8589	0.9134	0.0841	0.0841	0.0718	0.0718	0.0473	0.0193	0	0	0	0	0	0	1,179.33	1,137.65	1,137.28	1,115.54	972.46	919.42
	0.8286	0.8286	0.8387	0.8387	0.8598	0.9105	0.0780	0.0780	0.0678	0.0678	0.0468	0.0214	0	0	0	0	0	0	1,177.06	1,072.42	907.24	949.83	1,048.83	873.13
	0.8222	0.8222	0.8291	0.8291	0.8550	0.9124	0.0820	0.0820	0.0802	0.0802	0.0530	0.0220	0	0	0	0	0	0	1,245.55	1,214.27	1,127.14	1,090.42	900.98	909.36
	0.8237	0.8237	0.8366	0.8366	0.8608	0.9290	0.0814	0.0814	0.0695	0.0695	0.0492	0.0212	0	0	0	0	0	0	1,120.35	1,149.51	962.89	967.74	1,018.33	851.69
	0.8157	0.8157	0.8237	0.8237	0.8491	0.9191	0.0902	0.0902	0.0815	0.0815	0.0665	0.0209	0	0	0	0	0	0	1,095.01	1,093.04	1,083.12	1,097.12	1,021.49	964.79
	0.8160	0.8160	0.8326	0.8326	0.8593	0.9189	0.0941	0.0941	0.0746	0.0746	0.0490	0.0193	0	0	0	0	0	0	1,244.51	1,087.72	1,066.62	1,050.29	1,098.34	972.27
Average	**0.8235**	**0.8236**	**0.8359**	**0.8356**	**0.8593**	**0.9180**	**0.0834**	**0.0833**	**0.0710**	**0.0709**	**0.0510**	**0.0179**	**0**	**0**	**0**	**0**	**0**	**0**	**1,190.34**	**1,118.15**	**1,034.60**	**1,034.87**	**982.89**	**920.39**
$m = 1.5N$	0.8282	0.8282	0.8412	0.8412	0.8753	0.9616	0.0758	0.0758	0.0652	0.0652	0.0409	0.0022	0	0	0	0	0	0	1,114.94	1,116.14	1,017.47	1,005.44	1,130.70	1,008.66
	0.8470	0.8456	0.8582	0.8585	0.8759	0.9566	0.0620	0.0619	0.0512	0.0507	0.0422	0.0023	0	0	0	0	0	0	1,039.17	1,132.11	1,044.43	945.33	1,015.03	1,025.47
	0.8323	0.8323	0.8627	0.8627	0.8833	0.9374	0.0746	0.0746	0.0467	0.0467	0.0379	0.0095	0	0	0	0	0	0	1,153.93	1,283.01	1,159.88	1,133.55	1,053.58	921.24
	0.8248	0.8248	0.8527	0.8527	0.8942	0.9609	0.0797	0.0797	0.0558	0.0558	0.0409	0.0066	0	0	0	0	0	0	1,108.92	1,168.06	1,138.38	1,049.06	884.60	1,063.41
	0.8336	0.8336	0.8440	0.8453	0.8809	0.9494	0.0726	0.0726	0.0626	0.0622	0.0392	0.0076	0	0	0	0	0	0	1,258.09	1,246.51	1,101.60	1,169.54	1,081.20	999.14
	0.8335	0.8335	0.8563	0.8563	0.8919	0.9614	0.0722	0.0722	0.0517	0.0517	0.0417	0.0023	0	0	0	0	0	0	1,122.13	1,147.32	1,030.58	994.14	941.51	983.53
	0.8283	0.8283	0.8497	0.8497	0.8741	0.9532	0.0755	0.0755	0.0583	0.0583	0.0405	0.0097	0	0	0	0	0	0	1,089.25	1,131.40	985.67	1,137.31	874.14	1,078.77
	0.8317	0.8317	0.8553	0.8566	0.8825	0.9594	0.0771	0.0771	0.0561	0.0556	0.0391	0.0036	0	0	0	0	0	0	1,188.11	1,184.96	1,051.48	1,146.00	1,103.00	1,002.75
	0.8288	0.8288	0.8349	0.8349	0.8816	0.9562	0.0799	0.0799	0.0706	0.0706	0.0405	0.0043	0	0	0	0	0	0	1,118.71	1,186.23	1,116.95	1,134.86	997.38	1,047.48
	0.8209	0.8220	0.8457	0.8457	0.8794	0.9644	0.0891	0.0882	0.0656	0.0656	0.0407	0.0025	0	0	0	0	0	0	1,162.19	1,144.60	1,068.67	1,182.44	914.92	945.32
Average	**0.8309**	**0.8309**	**0.8501**	**0.8504**	**0.8819**	**0.9561**	**0.0759**	**0.0758**	**0.0584**	**0.0582**	**0.0404**	**0.0051**	**0**	**0**	**0**	**0**	**0**	**0**	**1,135.54**	**1,174.03**	**1,071.51**	**1,089.77**	**999.61**	**1,007.58**

Table B.17: Detailed performance in the *Pendigits-389* dataset.