Pontifícia Universidade Católica
do Rio de Janeiro

Sonia Fiol González

Heuristics for data point selection for labeling in Semi-Supervised and Active Learning contexts

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. Hélio Côrtes Vieira Lopes
Co-Advisor: Prof. Cassio Freitas Pereira de Almeida

Rio de Janeiro
February 2021

Pontifícia Universidade Católica
do Rio de Janeiro

Sonia Fiol González

Heuristics for data point selection for labeling in Semi-Supervised and Active Learning contexts

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. Hélio Côrtes Vieira Lopes
Advisor
Departamento de Informática – PUC-Rio

Prof. Cassio Freitas Pereira de Almeida
Co-Advisor
ENCE

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

Prof. Vinícius da Silva
Departamento de Informática – PUC-Rio

Prof. Luiz Carlos Pacheco Rodrigues Velho
IMPA

Prof^a. Jéssica Quintanilha Kubrusly
UFF

Prof. Alex Laier Bordignon
UFF

Rio de Janeiro, February the 3rd, 2021

Sonia Fiol González

The author graduated in Computer Science from the University of Havana in 2012. In 2016, she obtained her master's degree in Computer Science at PUC-Rio. She has an interest in Data Science, Machine Learning, and Information Visualization.

Bibliographic Data

Fiol González, Sonia

Heuristics for data point selection for labeling in Semi-Supervised and Active Learning contexts / Sonia Fiol González; advisor: Hélio Côrtes Vieira Lopes; co-advisor: Cassio Freitas Pereira de Almeida. – 2021.

98 f: il. color. ; 30 cm

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

Inclui bibliografia

1. keywordpre – Teses. 2. keywordpre – Teses. 3. Seleção de pontos. 4. Matriz de co-associação. 5. Propagação de rótulos. 6. Aprendizado semi-supervisionado. 7. Aprendizado activo. I. Côrtes Vieira Lopes, Hélio. II. Freitas Pereira de Almeida, Cassio. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. IV. Título.

CDD: 004

Acknowledgments

In the first place, to my advisors, Hélio Lopes and Cassio Almeida, for the opportunity, confidence, creativity, passion, and dedication to the research and the valuable teachings through this period. To professor Marcus Poggi for professional advice.

To Jefry, for his infinite patience and understanding in this period filled with tension. For his innumerable advice in order to finish the thesis on time.

To my family, in particular, my parents and my sister, for the love, affection, and unconditional support through these four years of sacrifice. To my best friend Delvia, for her constant concern and advice even being far away.

To my colleagues for providing me an exceptional environment and spirit. Also for their friendship and will to contribute to this work. Especially to Ariane for her friendship and for spending her precious time reviewing this work in detail.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. Finally, I would like to thank Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), for partially financing this research under grant #142349/2016-0.

For you all, my sincere thank you!

Abstract

Fiol González, Sonia; Côrtes Vieira Lopes, Hélio (Advisor); Freitas Pereira de Almeida, Cassio (Co-Advisor). **Heuristics for data point selection for labeling in Semi-Supervised and Active Learning contexts**. Rio de Janeiro, 2021. 98p. Tese de Doutorado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Supervised learning is, today, the branch of Machine Learning central to most business disruption. The approach relies on having amounts of labeled data large enough to learn functions with the required approximation. However, labeled data may be expensive, to obtain or to construct through a labeling process. Semi-supervised learning (SSL) strives to label accurately data from small amounts of labeled data and the use of unsupervised learning techniques. One labeling technique is label propagation. We use specifically the Consensus rate-based label propagation (CRLP) in this work. A consensus function is central to the propagation. A possible consensus function is a co-association matrix that estimates the probability of data points i and j belong to the same group. In this work, we observe that the co-association matrix has valuable information embedded in it. When no data is labeled, it is common to choose with a uniform probability randomly, the data to manually label, from which the propagation proceeds. This work addresses the problem of selecting a fixed-size set of data points to label (manually), to improve the label propagation algorithm's accuracy. Three selection techniques, based on stochastic sampling principles, are proposed: Stratified Sampling (SP), Probability (P), and Stratified Sampling - Probability (SSP). They are all based on the information embedded in the co-association matrix. Experiments were carried out on 15 benchmark sets and showed exciting results. Not only because they provide a more balanced selection when compared to a uniform random selection, but also improved the accuracy results of a label propagation method. These strategies were also tested inside an active learning process in a different context, also achieving good results.

Keywords

Data point selection; Co-association matrix; Label propagation; Semi-supervised learning; Active learning.

Resumo

Fiol González, Sonia; Côrtes Vieira Lopes, Hélio; Freitas Pereira de Almeida, Cassio. **Heurísticas para seleção de pontos para serem anotados no contexto de Aprendizado Semi-Supervisionado e Ativo**. Rio de Janeiro, 2021. 98p. Tese de Doutorado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

O aprendizado supervisionado é, hoje, o ramo do aprendizado de máquina central para a maioria das inovações nos negócios. A abordagem depende de ter grandes quantidades de dados rotulados, suficiente para ajustar funções com a precisão necessária. No entanto, pode ser caro obter dados rotulados ou criar os rótulos através de um processo de anotação. O aprendizado semi-supervisionado (SSL) é usado para rotular com precisão os dados a partir de pequenas quantidades de dados rotulados utilizando técnicas de aprendizado não supervisionado. Uma técnica de rotulagem é a propagação de rótulos. Neste trabalho, usamos especificamente o algoritmo *Consensus rate-based label propagation* (CRLP). Este algoritmo depende do uma função de consenso para a propagação. Uma possível função de consenso é a matriz de co-associação que estima a probabilidade dos pontos i e j pertencem ao mesmo grupo. Neste trabalho, observamos que a matriz de co-associação contém informações valiosas para tratar esse tipo de problema. Quando nenhum dado está rotulado, é comum escolher aleatoriamente, com probabilidade uniforme, os dados a serem rotulados manualmente, a partir dos quais a propagação procede. Este trabalho aborda o problema de seleção de um conjunto de tamanho fixo de dados para serem rotulados manualmente que propiciem uma melhor precisão no algoritmo de propagação de rótulos. Três técnicas de seleção, baseadas em princípios de amostragem estocástica, são propostas: *Stratified Sampling* (SS), *Probability* (P), and *Stratified Sampling - Probability* (SSP). Eles são todos baseados nas informações embutidas na matriz de co-associação. Os experimentos foram realizados em 15 conjuntos de benchmarks e mostraram resultados muito interessantes. Não só, porque eles fornecem uma seleção mais equilibrada quando comparados a uma seleção aleatória, mas também melhoram os resultados de precisão na propagação de rótulos. Em outro contexto, essas estratégias também foram testadas dentro de um processo de aprendizagem ativa, obtendo também bons resultados.

Palavras-chave

Seleção de pontos; Matriz de co-associação; Propagação de rótulos; Aprendizado semi-supervisionado; Aprendizado activo.

Table of Contents

1 Introduction 1
1.1 Motivation 3
1.2 Problem 3
1.3 Goal 3
1.4 Methodology 4
1.5 Contributions 4
1.6 Document organization 4

2 Background 6
2.1 Label Propagation Algorithm 6
2.2 Active Learning Algorithms 8
2.3 Summary 10

3 Related works 12
3.1 Ensemble clustering 12
3.2 Ensemble Clustering + Label Propagation 14
3.3 Active learning + SSL 17
3.4 Discussion 19

4 Selection strategies for labeling 21
4.1 Stratified sampling strategy 21
4.2 Probability sampling strategy 25
4.3 Stratified Sampling with Probability selection strategy 28

5 Algorithms 33
5.1 Implementation of SS, P and SSP 33
5.2 Step 2 of CRLP algorithm 36
5.3 Alpha tuning 37
5.4 Active Learning 37
5.4.1 Active Learning process with uncertainty 38
5.4.2 Active Learning process with CM update 39
5.4.3 Active Learning process with uncertainty and CM update 42

6 Experimental study 44
6.1 Benchmark 44
6.2 Procedure 44
6.3 Selection strategies experiment 46
6.3.1 Selection strategies with replacement 46
6.3.2 Selection strategies without replacement 49
6.4 Label Propagation experiment 52
6.4.1 Selection strategies applied to CRLP algorithm with replacement 53
6.4.2 Selection strategies and CRLP algorithm without replacement 55
6.4.3 Summary of LP experiment 55
6.5 Active Learning experiment 59

6.5.1 Active Learning process with Uncertainty 59
6.5.2 Active Learning process with CM update 59
6.5.3 Active Learning process with Uncertainty and CM update 66
6.5.4 Summary of Active Learning experiment 73
6.6 Discussion 74

7 Considerations about selection strategies 75
7.1 Similarity Matrix vs Co-association Matrix 75
7.2 Optimal solution analysis 77

8 Conclusion 81
8.1 Contributions 81
8.2 Future works 82

Bibliography 82

A Appendices 89
A.1 Appendix 1 90
A.2 Appendix 2 91
A.3 Appendix 3 93
A.4 Appendix 4 94
A.5 Appendix 5 95
A.6 Appendix 6 96
A.7 Appendix 7 97

List of Figures

Figure 1.1 Co-association matrix and Similarity matrix for the Wine dataset. 2

Figure 1.2 Activities carried out in this work. The main stages are in orange. In light gray are the activities and experiments related to the selection strategies, and the CM's creation. Finally, dark gray activities are related to the label propagation algorithm after applying the selection strategies and the active learning process. 5

Figure 2.1 Flow diagram of the Pool-Based AL Scenario (image taken and adapted from (Kumar and Gupta, 2020)). 9

Figure 4.1 Stratified Sampling strategy diagram to define the initial labeled data points. 22

Figure 4.2 Example of Stratified Sampling strategy in a toy dataset. 23

Figure 4.3 Example of Stratified Sampling limitation. 25

Figure 4.4 Probability sampling strategy diagram to define the initial labeled data points. 26

Figure 4.5 Example of Probability sampling strategy in a toy dataset. 27

Figure 4.6 Stratified Sampling Probability diagram to define the initial labeled data points. 30

Figure 4.7 Example of SS-Probability strategy in a toy dataset. 31

Figure 5.1 Flow diagram of Active Learning process. 38

Figure 5.2 Flow of Active Learning process with CM update. 40

Figure 6.1 Mean Jensen Shannon Divergence curves of the Random, Probability, Stratified Sample and Stratified Sample Probability algorithms with replacement when applied to 15 datasets. 47

Figure 6.2 Standard Deviation Jensen Shannon Divergence curves of the Random, Probability, Stratified Sample and Stratified Sample Probability algorithms with replacement when applied to 15 datasets. 48

Figure 6.3 Mean Jensen Shannon Divergence curves of the Random, Probability, Stratified Sample and Stratified Sample Probability algorithms without replacement when applied to 15 datasets. 50

Figure 6.4 Standard Deviation Jensen Shannon Divergence curves of the Random, Probability, Stratified Sample and Stratified Sample Probability algorithms without replacement when applied to 15 datasets. 51

Figure 6.5 Classification mean accuracy curves for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms with replacement in 15 datasets. 53

Figure 6.6 Classification standard deviation accuracy curves for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms with replacement in 15 datasets. 54

Figure 6.7 Classification mean accuracy curves for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms without replacement in 15 datasets. 56

Figure 6.8 Classification standard deviation accuracy curves for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms without replacement in 15 datasets. 57

Figure 6.9 Classification accuracy curves of the all simulations for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms by dataset. 58

Figure 6.10 Classification mean accuracy curves for the Random, Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with uncertainty in 15 datasets. 60

Figure 6.11 Classification standard deviation accuracy curves for the Random, Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with uncertainty in 15 datasets. 61

Figure 6.12 Classification mean accuracy curves for the Random, Random with Uncertainty, Stratified Sample Probability and Stratified Sample Probability with Uncertainty algorithms for the label propagation without replacement and active learning process with uncertainty in 15 datasets. 62

Figure 6.13 Classification standard deviation accuracy curves for the Random, Random with Uncertainty, Stratified Sample Probability and Stratified Sample Probability with Uncertainty algorithms for the label propagation without replacement and active learning process with uncertainty in 15 datasets. 63

Figure 6.14 Classification mean accuracy curves for the Random, Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with CM update in 15 datasets. 64

Figure 6.15 Classification standard deviation accuracy curves for the Random, Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with CM update in 15 datasets. 65

Figure 6.16 Classification mean accuracy curves for the Random, Random with CM update, Stratified Sample Probability and Stratified Sample Probability with CM update algorithms for the label propagation without replacement and active learning process with CM update in 15 datasets. 67

Figure 6.17 Classification standard deviation accuracy curves for the Random, Random with CM update, Stratified Sample Probability and Stratified Sample Probability with CM update algorithms for the label propagation without replacement and active learning process with CM update in 15 datasets. 68

Figure 6.18 Classification mean accuracy curves for the Random, Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with uncertainty and CM update in 15 datasets. 69

Figure 6.19 Classification standard deviation accuracy curves for the Random, Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with uncertainty and CM update in 15 datasets. 70

Figure 6.20 Classification mean accuracy curves for the Random, Random with Uncertainty and CM update, Stratified Sample Probability and Stratified Sample Probability with Uncertainty and CM update algorithms for the label propagation without replacement and active learning process with uncertainty and CM update in 15 datasets. 71

Figure 6.21 Classification standard deviation accuracy curves for the Random, Random with Uncertainty and CM update, Stratified Sample Probability and Stratified Sample Probability with Uncertainty and CM update algorithms for the label propagation without replacement and active learning process with uncertainty and CM update in 15 datasets. 72

Figure 6.22 Classification accuracy curves of the 100 simulations for the Random, Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process by dataset. 73

Figure 7.1 Classification mean and standard deviation accuracy curves for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms without replacement using the SM in 14 datasets. 76

Figure 7.2 Classification mean accuracy curves for the Optimal, Random and Stratified Sample Probability algorithms without replacement when applied to Leukemia, Lymphoma and Armstrong datasets. 78

Figure A.1 Classification accuracy curves of the 100 simualtions for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms with replacement in 15 datasets. 91

Figure A.2 Classification accuracy curves of the 100 simualtions for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms with replacement in 15 datasets (continued). 92

Figure A.3 Classification accuracy curves of the 100 simulations for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms without replacement in 15 datasets. 93
Figure A.4 Classification accuracy curves of the 100 simulations for the Random,Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with uncertainty in 15 datasets. 94
Figure A.5 Classification accuracy curves of the 100 simulations for the Random, Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with CM update in 15 datasets. 95
Figure A.6 Classification accuracy curves of the 100 simulations for the Random,Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with uncertainty and CM update in 15 datasets. 96
Figure A.7 Co-association matrix of Leukemia, Lymphoma, Armstrong, Wine, Chen, Seeds, Congressional, Vertebral and Breast Cancer 1 datasets. 97
Figure A.8 Co-association matrix of Synthetic, Balance Scale, Urban, Breast Cancer 2 and Segmentation datasets. 98

List of Tables

Table 6.1 Overview of datasets. 45

Table 6.2 Hyperparameters for the CRLP blind algorithms using Random, Stratified Sampling, Probability and Stratified Sampling with Probability as selected initial data points. 52

Table 7.1 Mean accuracy and Standard Deviation by number of labeled data points for best solution found, SSP and Uniform Random strategies. 80

Table A.1 Median accuracy with Median Absolute Deviation by k for Blind Uniform Random CRLP algorithm. 90

List of Abbreviations

AL – Active Learning

CM – Co-association matrix

CRLP – Consensus rate-based label propagation algorithm

EC – Ensemble Clustering

IA – Artificial Intelligence

JS – Jensen-Shannon

LP – Label Propagation

ML – Machine Learning

P – Probability sampling strategy

QS – Query System

SM – Similarity Matrix

SS – Stratified Sampling strategy

SSP – Stratified Sampling with Probability selection strategy

SSL – Semi-supervised Learning

1
Introduction

Machine Learning is playing an essential role in the transformation of industry and society (Molnar, 2020). Supervised learning is today, the branch of Machine Learning central to most business disruption. The approach relies on having amounts of labeled data large enough to learn functions with the required approximation. However, labeled data may be expensive to obtain or to construct through a labeling process.

Within the area of unsupervised learning, researchers recognize *ensemble clustering* as a useful technique (Huang et al., 2019). This technique aims to combine multiple partitions (clustering results) of the same data set into a final partition. One of its main steps is creating a consensus function that reflects the similarity between two data points. An example of a consensus function is the so-called Co-association Matrix (CM) (Fred and Jain, 2005), where the position (i, j) contains the probability of the data point i and the data point j be in the same group considering the multiple partitions.

Figure 1.1 shows an example of a CM and a Similarity Matrix (SM) for the Wine dataset (Lichman et al., 2013) with three classes. Both matrices are sorted by class. In the similarity matrix, the values are very similar, so it is difficult to differentiate the blocks. In (Fiol-González et al., 2019), the authors define it as a block: "rectangular shape formed around the main diagonal in the heat map containing the elements belonging to the same cluster". In the CM, these blocks are better defined, identifying three groups with frequency values closer to 1, although the center block seems more confusing. Both matrices present regions of confusion, however, the CM has less noise than the SM, and the CM matrix contains more valuable information about the dataset. So, it is very evident that the CM has much information embedded in its content. In this work, we will explore this fact.

The task of getting labeled data is expensive and time-consuming (Berikov et al., 2017). It is possible to learn from data when one combines unlabeled and a small amount of labeled data (Berikov et al., 2017). This situation is precisely the Semi-Supervised Learning (SSL) area of study. SSL algorithms have two premises: the first is that data points next to each other commonly belong to the same class, and the second is that data points at

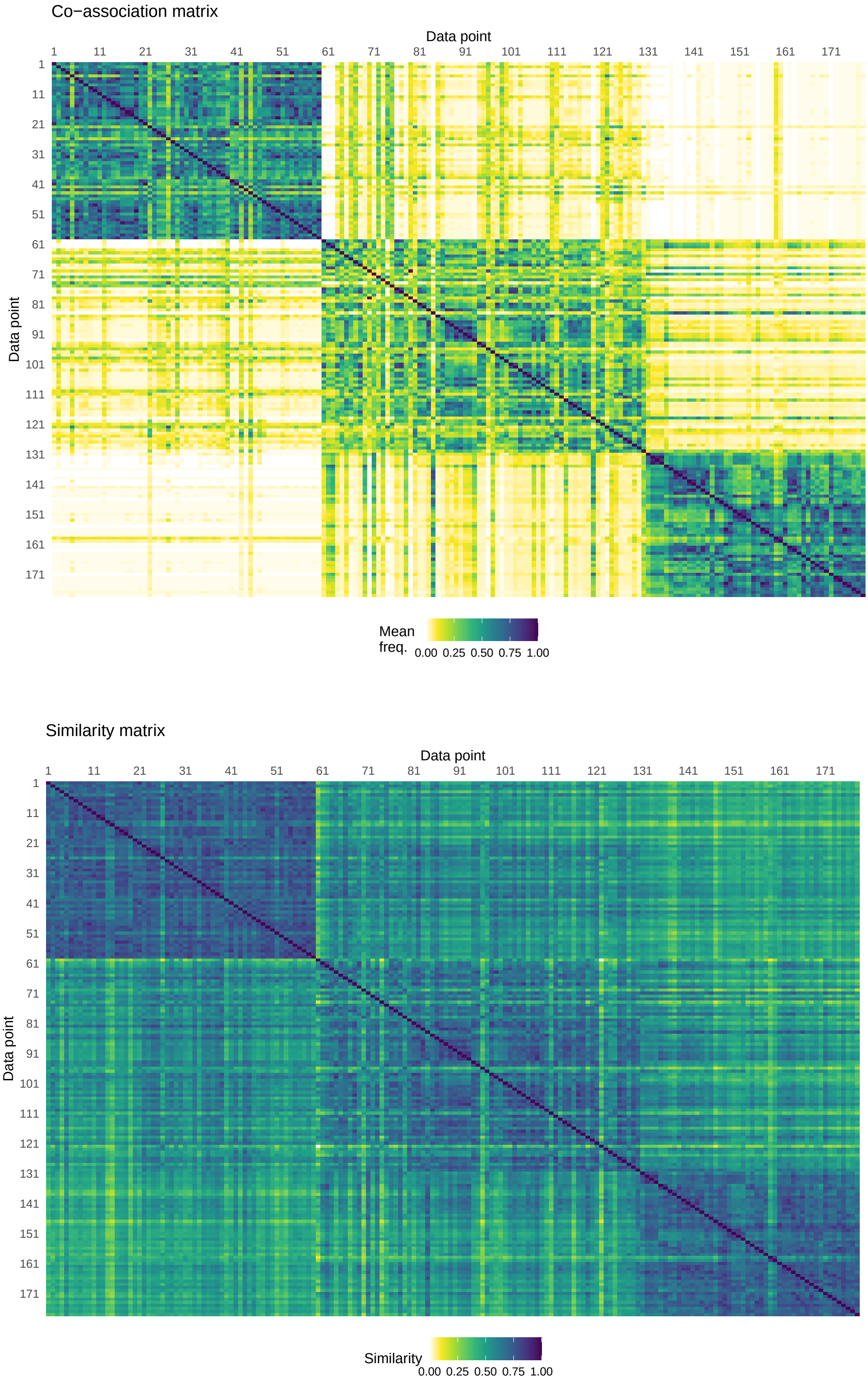


Figure 1.1: Co-association matrix and Similarity matrix for the Wine dataset.

the same structure (cluster) commonly belong to the same class (Zhou et al., 2004a).

Label Propagation (LP) algorithms form a class of traditional algorithm of this area, and they aim to propagate the known information of the annotated data points to their neighbors iteratively until convergence (Zhu et al., 2003a; Zhou et al., 2004a).

1.1 Motivation

LP algorithms still require labeled data. The labeled data selection is carried out randomly in the literature, taking into account prior knowledge of the data set classes (Zhu and Ghahramani, 2002; Zhu et al., 2003a; Zhou et al., 2004a; Wang and Zhang, 2007; Yu and Kim, 2018). Given this premise, applying LP algorithms to a real-life problem is complicated because the class to which each point belongs is unknown. At most, it could be known how many classes/groups the analyzed domain has.

1.2 Problem

This work addresses the problem of selecting a fixed-size set of data points to label, aiming to improve the label propagation algorithm's accuracy.

1.3 Goal

This work aims to improve the selection of the initial data point set. To guide our work, based on our goal, we propose the following research questions:

- **RQ1:** How can we ensure that the data points class selection represents the real class distribution?
- **RQ2:** Is it possible to build a data point selection strategy from the co-association matrix to improve the accuracy of the Consensus Rate-based Label Propagation (CRLP) method?
- **RQ3:** Within the Active Learning framework, is it possible to integrate initial point selection strategies to improve classification accuracy?

1.4 Methodology

To answer the research questions of this work, we adopt the following procedure. First, we reviewed the literature on Ensemble Clustering, Semi-supervised learning, and Active Learning. In the Ensemble clustering area, we specifically search how to obtain the consensus function through the co-association matrix, while in the semi-supervised area, we search for label propagation algorithms. In the Active Learning area, we explore the traditional query systems. Finally, we search for papers merging previous areas. Based on the literature review and the identified problem, we developed our proposed solution that consists of three different data point selection strategies. We validate our proposal through three quantitative experiments applied to 15 datasets from the literature. We show that the proposed solution improves the classification of the CRLP algorithm. Figure 1.2 illustrates the sequence of activities carried out in this work.

1.5 Contributions

We propose three data points selection strategies for labeling using stochastic sampling principles based on a co-association matrix. This strategies are Stratified Sampling selection strategy (SS), Probability Sampling selection strategy (P) and Stratified Sampling Probability selection strategy (SSP). Our strategies results, show an improvement in the classification results in semi-supervised and active learning contexts.

1.6 Document organization

The remaining of this document is structured as follows: Chapter 2 briefly presents important background concepts, and Chapter 3 describes the related works on ensemble clustering algorithms, label propagation algorithms, and ensemble clustering with label propagation algorithms. Chapter 5 presents the proposal algorithms. Chapter 4 presents Stratified Sampling, Probability, and Stratified Sampling - Probability strategies. Chapter 5 presents the proposal algorithms. Chapter 6 describes some experiments in benchmark data sets. Chapter 7 presents a study about the selection of Co-association matrix and conclusions are given in Chapter 8.

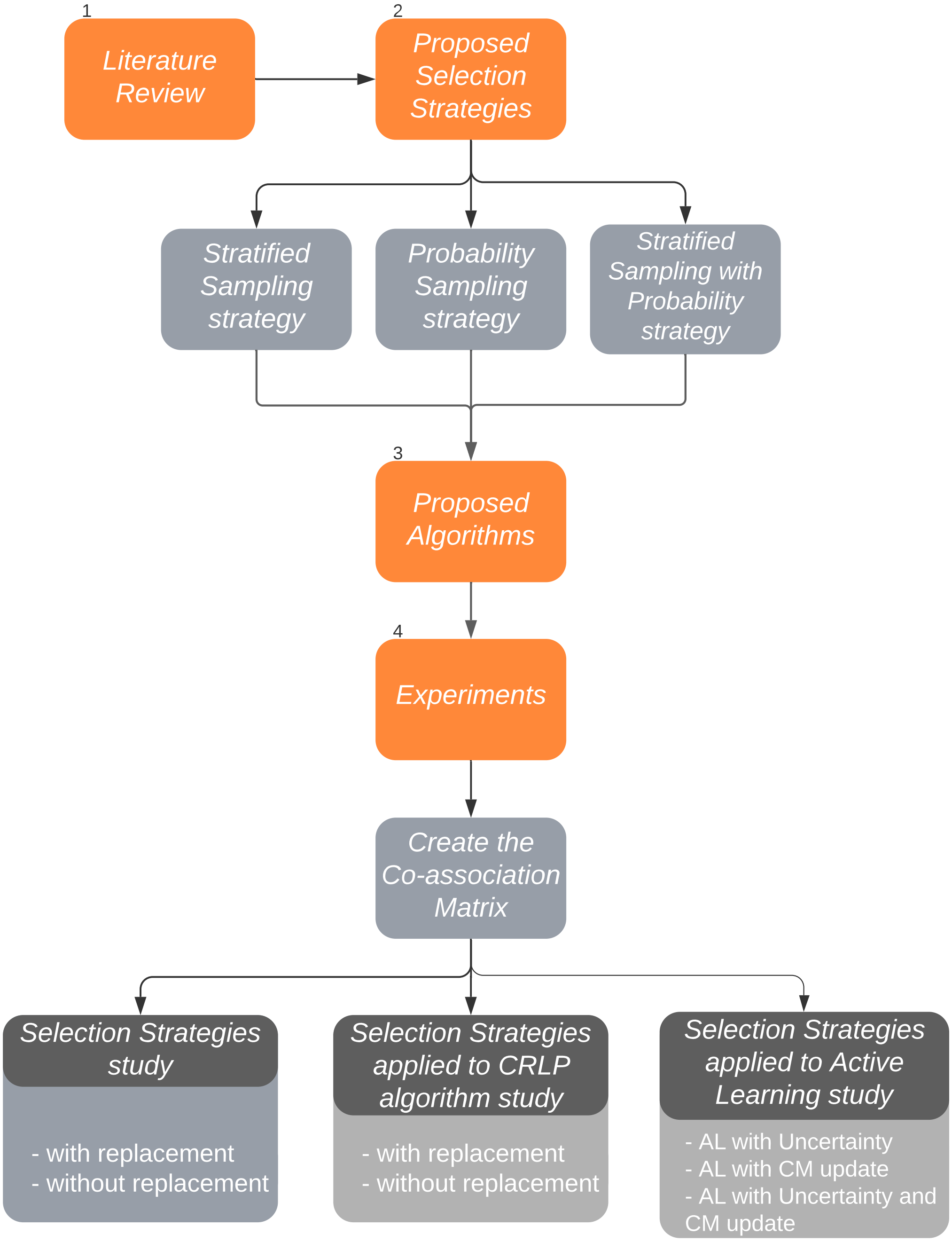

1
Literature Review
2
Proposed Selection Strategies
Stratified Sampling strategy
Probability Sampling strategy
Stratified Sampling with Probability strategy
3
Proposed Algorithms
4
Experiments
Create the Co-association Matrix
Selection Strategies study
- with replacement
- without replacement
Selection Strategies applied to CRLP algorithm study
- with replacement
- without replacement
Selection Strategies applied to Active Learning study
- AL with Uncertainty
- AL with CM update
- AL with Uncertainty and CM update

Figure 1.2: Activities carried out in this work. The main stages are in orange. In light gray are the activities and experiments related to the selection strategies, and the CM's creation. Finally, dark gray activities are related to the label propagation algorithm after applying the selection strategies and the active learning process.

2
Background

In this chapter, we present the main concepts and algorithms that support our work. In the Section 2.1 we describe the main LP algorithms. In the Section 2.2 we present the fundamental ideas about AL and traditional Query Systems. In the Section 2.3 we carry out a summary and further considerations.

2.1
Label Propagation Algorithm

Semi-supervised Learning (SSL) algorithms assume that nearby data points belong to the same class and that similar data points also belong to the same class Wang and Zhang (2007). SSL creates the model by joining the labeled and unlabeled data (Settles, 2009). In algorithms based on graphs, the unlabeled data points are classified by propagating the label through the data points. In this work, we used the Consensus Rate Label Propagation (CRLP) algorithm (Yu and Kim, 2018).

The CRLP algorithm aims to propagate the weighted graph labels $G =< V, E, CM >$ of V vertices and E edges, where the weight of the edges is defined in the co-association matrix CM. The algorithm contains three steps. Firstly, the algorithm creates CM by combining multiple clustering results. Secondly, It randomly selects a set of data points to be labeled and propagate the label to the unlabeled data points with the information from CM. Finally, the algorithm classifies new observations based on multiple clustering and labels propagation results. CRLP algorithm selects a fixed number of data points per class randomly (Yu and Kim, 2018).

Algorithm 1 shows how the classification process is carried out through propagation. The algorithm receives as input parameters, *dataset* that represents the analyzed dataset, Y_0 that specifies the set of annotated data points, B the specified number of partitions to obtain the CM, and a scalar α that ranges in the interval $[0, 1)$ is a learning rate. We explain each step of the algorithm in more detail below.

In the first step to creating CM, B partitions are obtained by randomly varying the number of groups (line 5), always choosing half of the variables (line 6) and using the K-Means (line 7) as the clustering algorithm. The K_i function

Algorithm 1 Consensus rate-based label propagation (Yu and Kim, 2018)

```
1: procedure CRLP($dataset, Y_0, B, \alpha$)
2:     $n \leftarrow$ number of data points in $dataset$
3:     $p \leftarrow$ number of attributes in $dataset$
4:     while i in {0 .. B} do
5:         $k \leftarrow$ random between $\{2..\sqrt{n}\}$
6:         $w \leftarrow$ random $p/2$ attributes from $dataset$
7:         $K_i \leftarrow$ K-Means($k, dataset[:, w]$)
8:     end while
9:     $I_i^{l,k} = \begin{cases} 1 & if K_i(o_l) = K_i(o_k) \\ 0 & otherwise \end{cases}$
10:    $CM \leftarrow \frac{1}{B} \sum_{i=1}^{B} I_i$
11:    $\forall x \in \{1,..,N\}\ CM[x,x] \leftarrow 0.$
12:    Compute the diagonal degree matrix D by $D_{ii} \leftarrow \sum_j CM_{ij}$
13:    Compute the normalized graph Laplacian $L \leftarrow D^{-1/2} CM D^{-1/2}$
14:    Initialize $Y\hat{}(0) \leftarrow Y_0$
15:    Iterate $Y^{(t+1)} \leftarrow \alpha L Y^{(t)} + (1-\alpha) Y^{(0)}$ until convergence to $Y^{(\infty)}$
16:    Label data point $dp_i$ with $\underset{j}{argmax}\ \hat{y_{ij}}^{(\infty)}$
17:    Finally, classify the labels of new observations with the smoothness
       function obtain previously.
18: end procedure
```

verify if a data point belongs to the i cluster. The B partitions are summarized in CM on lines 9 and 10. Then, from the CM and Y_0, the algorithm performs a label propagation process to obtain a classification model. In the second step, zero is assigned on the diagonal of the CM (line 11), the diagonal matrix is computed from the CM (line 12) and normalized using the Laplacian graph (line 13). In each iteration (line 15), each data point receives the information from its neighbors ($\alpha L Y^{(t)}$) and also retains part of the initial information ($(1-\alpha)Y^{(0)}$) depending on the α value. The α parameter specifies the relative amount of information taken from the neighbors and retain from the initial information. When the method converges (line 16), each data point in $Y^{(t)}$ contains the degree of belonging to each class. Each data point will receive the class that contains the highest probability value. In step 3 (line 17), the new observations (testing set) will be classified. Yu and Kim Yu and Kim (2018) divide the dataset into training and testing, so it is necessary to create a classification model using only the training set and then apply the created model to the testing set. In our case, we did not perform this division of the dataset because we use the general knowledge of the entire dataset.

From the CRLP algorithm, we used the first and second steps (lines 2 to 16) and included a data point selection strategy based on the CM. We have chosen this LP algorithm to be the one to test our sampling strategies since it also uses the CM as a basis.

2.2 Active Learning Algorithms

Active Learning (AL) is a sub-area of Machine Learning. It is a framework that allows you to automatically select the most informative data points to be annotated manually (Yin et al., 2019). With this, the AL paradigm reduces the time/effort/cost of annotation in the training dataset (Tomanek and Hahn, 2009; Settles, 2009). AL reduces the number of instances that must be labeled to achieve reasonable accuracy. The objective is to maximize the classification's accuracy given a cost function where the cost is associated with the acquisition or annotation of a data point (Aggarwal, 2015).

The framework has two main components, the Query System (QS) and the Oracle. The Query System is in charge of exploring the data points and returning those that are the most informative. It is considered an essential task in the process of active learning (Yin et al., 2019). The Oracle returns the annotation of each data point suggested by the QS.

In AL, there are three main types of working scenarios: Membership Query Synthesis Based AL Scenario, Stream-Based AL Scenario, and Pool-Based AL Scenario Kumar and Gupta (2020). The Pool-Based AL Scenario is when the learner has access to the set of unlabeled data before starting the learning process. The following process represents it. The stage receives a fixed unlabeled data set, and in each process iteration, an instance is selected to be annotated by the oracle. The oracle knows the actual label of this instance, and a new model is generated based on all the annotated data. This process repeats until it meets the stop conditions (Baram et al., 2004). This type of scenario is the most common among literature papers. Figure 2.1 shows a diagram representing the AL loop for this scenario. The Membership Query Synthesis Based AL Scenario is also known as selective sampling. At each stage, the learner generates an instance of the input data space and requests its annotation from the oracle. This scenario is an example of Pool-Based AL, where all data points in the input domain represent the pool. The Stream-Based AL Scenario learner receives a stream of unlabeled data. In each trial, we get an instance from the stream, and the learner decides whether to label it down or not. For more information on the scenarios, see (Settles, 2009; Kumar and Gupta, 2020) where the authors carry out an extensive review of the active learning literature.

Every AL process starts from an initial set of annotated data points. Next, we obtain a learning model from the previous set. Then this model

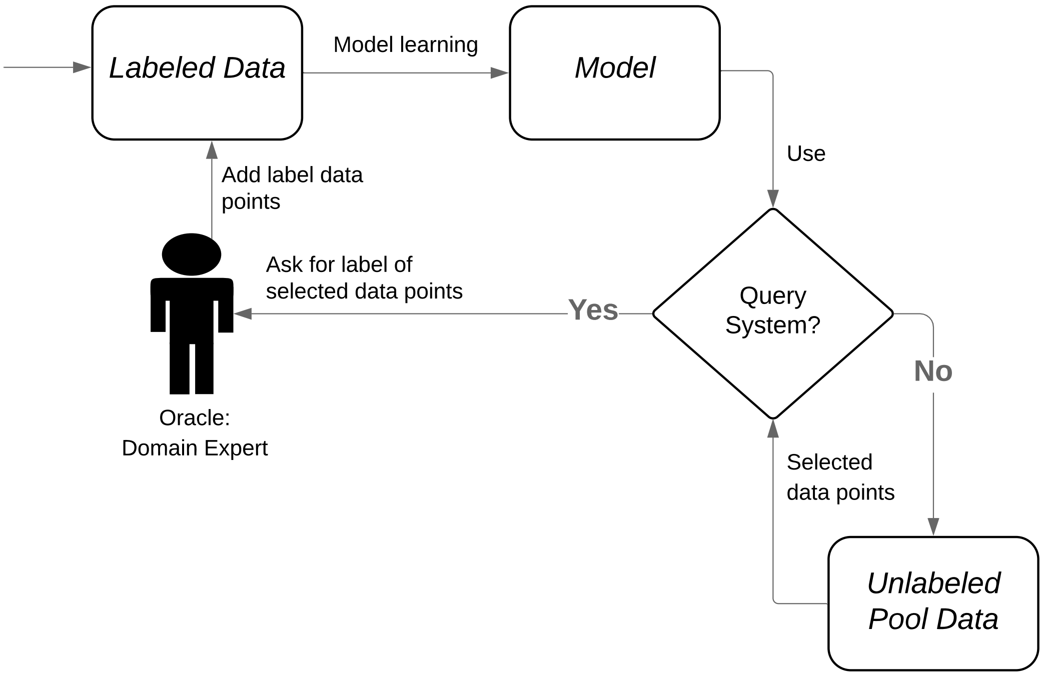


Figure 2.1: Flow diagram of the Pool-Based AL Scenario (image taken and adapted from (Kumar and Gupta, 2020)).

is used to evaluate the unlabeled data points that are in the pool. In the Query System, we define the conditions for a data point to be selected. If the data point selected from the unlabeled set does not meet the QS conditions, it chooses another data point from the pool. Otherwise, the domain expert labels the datapoint and adds to the set of labeled data points until it meets a predefined stop condition or until the pool is empty.

Not all data points are equally informative. Therefore it is essential to choose the most relevant data points through the query system. There are several types of Query Systems to achieve this goal such as Uncertainty Sampling, Query by Committee, Support Vector Machines (SVM) Based Approach, density-weighted method, Expected error reduction, Variance reduction, and Expected model change. Depending on the type of query instance, we can divide these strategies into three categories: informative-based, representative-based, and the combination of the previous two (informative and representative-based) (Kumar and Gupta, 2020). In (Kumar and Gupta, 2020), the authors present a well-structured QS hierarchy divided by scenario type and task type.

Uncertainty Sampling is the most straightforward and widely used strategy in the literature. It selects the data points with the highest uncertainty over its annotation and generally uses probabilistic models to determine the lowest certainty datapoints. Entropy commonly measures uncertainty. Query by

Committee is a more theoretical approach that involves maintaining a committee of learning models. These models are trained with labeled data, and each model votes on a candidate to be analyzed. Finally, we choose the data point that generates the most significant discrepancy between the models. Support Vector Machines (SVM) Based Approach uses the SVM model as the basis for selecting the data points to be annotated. The SVM technique is known for creating margins to separate training data. With these margins, it is possible to select the unlabeled data points closest to the margins. The points closest to the margins are the most ambiguous data points, and their annotation would help create more accurate models. Expected model change selects the instance that will generate the current model's most significant change if its annotation is known. This technique is common in gradient-based models because the impact caused on the model is estimated through the gradient. Expected error reduction aims to select the data point that most reduces the model error. Instead of measuring how the model varies (Expected model change), it reduces the error of the change. This strategy is the most computationally expensive. Variance reduction is also an approach to reduce generalized error indirectly through minimization of the output variance. In some cases, this variance has a closed-form solution. The main idea of the Density-weighted method is to mix the most uncertain points with the most informative. The most informative data points are considered those found in dense or homogeneous areas according to the distribution of the points (Settles, 2009; Kumar and Gupta, 2020).

Another possible classification of AL is given by the number of data points selected at each moment. From this point of view, they are divided into myopic active learning and batch active learning (Yang and Loog, 2019). A QS algorithm is myopic active learning when, in each iteration, it selects only one data point. Examples of this category are Uncertainty Sampling, Query by Committee, Error reduction, to cite some. A QS algorithm is classified as batch active learning when it simultaneously selects a group of data points from the unlabeled pool data (Yang and Loog, 2019; Das et al., 2020).

2.3 Summary

In summary, Semi-supervised learning and Active learning address the same problem but from different points of view. Semi-supervised Learning exploits the knowledge that the model thinks it learned about the unlabeled data (Settles, 2009). That is, the algorithm creates the model by joining the labeled and unlabeled data. In algorithms based on graphs, the unlabeled data points are classified by propagating the label of the labeled data. Active learning is a framework that explores unknown data points. The objective is to create a learning model from the labeled data points (Settles, 2009). It has a mechanism for selecting unlabeled data points through the query system. It obtains the real classification through an oracle to later train the model again. In our case, we combine both approaches. In other words, the model inside the active learning loop is a semi-supervised method that receives both labeled and unlabeled data.

3
Related works

This chapter presents an overview of the current works in the literature related to our work. Section 3.1 presents a literature review on Ensemble Clustering methods for creating the Co-association matrix. Section 3.2 presents works that combine Ensemble Clustering with Label Propagation while the section 3.3 combines Active Learning with Semi-supervised Learning. Finally section 3.4 we summarize the state of the art and complement it with our considerations.

3.1
Ensemble clustering

The ensemble clustering technique Strehl and Ghosh (2002); Fern and Brodley (2003); Topchy et al. (2004); Fred and Jain (2005); Wang et al. (2009); Vega-Pons and Ruiz-Shulcloper (2011); Xu and Tian (2015); Huang et al. (2015) aims to combine multiple weak base clustering results into a final partition. They all demonstrate that it is a relevant problem and present new upcoming challenges. Therefore, it is necessary to solve a correspondence problem.

Different strategies generate ensemble clustering, such as applying different clustering algorithms, using different initial parameters, or selecting a different subset of features. The primary step in this technique is to create a consensus function. A particular type of consensus function is the Co-association Matrix, also known as the Consensus Matrix.

There are several approaches to create the consensus function on clustering ensemble techniques. Among them, we could cite the co-association matrix. For example, the Cluster-based Similarity Partitioning Algorithm (CSPA), proposed in Strehl and Ghosh (2002), analyze element relationships to generate a co-association matrix, and finally, apply a clustering method. Another approach based on the co-association technique is the Evidence Accumulation matrix (EAC) method Fred and Jain (2005). The EAC applies Average Link (EAC-AL) and Single Link (EAC-SL) to extract the consensus clustering from a clustering ensemble. In Huang et al. (2015), the authors present an extension of the EAC called Weighted Evidence Accumulation matrix (WEAC). This extension penalizes low-quality clusters and assigns weights to each base clustering to generate the consensus partition. Iam-On et al. Iam-On et al. (2008)

present a link-based method that enhances the co-association matrix by addressing the relationship between partitions. The authors applied a linked network model and analyzed the similarity among clusters. The proposed method was tested on ten datasets (real and artificial datasets) and three benchmark measures. In Wang et al. (2009), the authors present the Probability Accumulation (PA) method. The authors take into consideration the cluster sizes of original clustering to generate a new correlation matrix. The Ensemble Clustering Matrix Completion (ECMC) method Yi et al. (2012) construct a partially observed matrix where each entry has an uncertainty value associated. The algorithm filters the most uncertain entries and then complete the matrix to fill the unobserved data. The Robust Spectral Ensemble Clustering (RSEC) Tao et al. (2016) captures various noise of the co-association matrix by applying a low-rank constraint. The proposed work split the co-association matrix into two matrices: a matrix with the underlying cluster structure and a matrix with the noise. Finally, the author applied a spectral clustering algorithm to find the final partition. Another strategy based on the ensemble-driven cluster uncertainty estimation and local weighting co-association matrix was presented in Huang et al. (2018). The authors introduce an ensemble-driven cluster validity measure and use the entropy to calculate the cluster uncertainty. In Fiol-Gonzalez et al. (2018), a novel committee-based clustering method was proposed. The method contains three steps: Firstly, create the ensemble through clustering and feature selection algorithms. Secondly, summarize the multiple partitions into a co-association matrix taking into account each data point's silhouette coefficient on each based partition. Finally, it applies a clustering method to generate the final partition. In (Huang et al., 2019), the authors proposed the Ultra-Scalable Spectral Clustering (U-SPEC) and ultra-scalable ensemble clustering (U-SENC). In U-SPEC, a hybrid representative selection strategy and a fast approximation method for K -nearest representatives are proposed to construct a sparse affinity sub-matrix. In Zhong et al. (2019), the authors filter the co-association matrix to obtain more accurate clustering results. The authors remove evidence with low occurrence frequency and use normalized cut to generate multiple partitions. In He and Huang (2019), the author presents a meta-Cluster based consensus cluster with local weighting and random walking (MC^3LR). The MC^3LR constructs a similarity graph, explores high order structural information, and estimates each base clustering's reliability through the Ensemble-driven cluster index (ECI). All previous related works focus on generating CM though different approaches. However, there is still active research on new techniques.

We adopted the procedure used in (Yu and Kim, 2018) to obtain the

CM. The authors created an ensemble with 100 members (partitions) by using the K-means algorithm, varying the number of groups between 2 and $\sqrt{Nb.instances}$ randomly, and selecting randomly $Nb.Attributes/2$ attributes. The K-means algorithm is executed ten times to generate each member, and the best result is chosen. The centroids are determined by using the K-means++ algorithm.

The simulation matrix related to the member k, where k varies between 1 and 100, is defined using Equation 3-1.

$$S_k[i,j] = \begin{cases} 1, & \text{if the } i^{th} \text{ data point was in the same cluster that } j^{th} \text{ data point} \\ 0, & \text{otherwise} \end{cases} \tag{3-1}$$

Finally, the co-association matrix is defined by the normalized sum of all simulations as in Equation 3-2.

$$CM = \frac{1}{100}\sum_{k=1}^{100} S_k \tag{3-2}$$

As a result, the CM contains in the entry (i,j) the probability that the elements dp_i and dp_j belong to the same cluster. Also, we have a square, symmetric and normalized matrix.

3.2 Ensemble Clustering + Label Propagation

In (Zhu et al., 2003a) it is proposed a semi-supervised method based on Gaussian random field model. This method creates a weighted graph where the vertex represent the data points, and the edges represent the distances. Therefore the authors use this graph to formulate the learning problem, and it is closely related to Spectral Graph Theory, Random Walk, and Electric Networks. This paper looks forward to combining labeled and unlabeled data effectively. The author performs ten trials, and for each trial, the algorithm selects between 20 and 100 random points, computes a weight matrix, and propagate the labels. All the results in this paper lie on a single dataset (MNIST). This paper relies on random algorithms to select the data point to be labeled. The authors relate a 95% accuracy with 20 labeled data points. However, the authors focus on two classes (1 and 2) out of ten, uses random strategies to choose the labeled data points, and provides mean accuracy of all results.

This article (De Sousa, 2015) proposes an overview of the Gaussian Fields and Harmonic Functions (GFHF) algorithm, considering and answering questions regarding the convergence analysis, scalability, active learning, its regularization framework, out-of-sample extension, and active learning. The author highlights active learning technique's relevance to take advantage of a weighted graph generated from the dataset using the GFHF algorithm.

The authors in (Zhou et al., 2004a) state that the semi-supervised learning focuses designing a sufficiently smooth classification function to adapt itself according to the data points structured revealed by the labeled and unlabeled data points. In this paper, the author presents a simple algorithm to obtain this kind of objective. In this article, the authors use the error rate metric. The authors were inspired by the Spreading Activation Network and Diffusion Kernels. The algorithm's core idea is to propagate the label to their neighbors depending on certain conditions until a global stop condition is met. The data points were represented in a similarity matrix. The experiments were carried on three datasets. The first one was a toy dataset as a proof of concept. The second dataset was a subset of the MNIST dataset. The authors filter four classes [1,2,3 and 4] out of ten, summing up to 3874 data points. The final reported result was the mean error over 100 trials, and the samples to be labeled must have at least one member of each class. The third dataset was a text classification (20-newsgroup) containing 3970 documents divided into four classes. In conclusion, the authors demonstrate the effective use of unlabeled data points in the used datasets.

In (Ovelgönne and Geyer-Schulz, 2012), the idea is to train several weak graph clustering and then combine them to create a more robust clustering. The authors combine multiple classifications and clustering results in order to improve prediction accuracy.

In (Zhang et al., 2014), the authors combine multiple classifications and clustering results in order to improve the prediction accuracy. Firstly, the algorithm applies several clustering algorithms and combines the results in a similarity graph. This graph can represent the internal data points relation. Once the graph is built, the author applies Zhou's semi-supervised learning algorithm (Zhou et al., 2004a) and define a bipartite graph between the labeled and unlabeled data points. This bipartite graph can modify the label propagation step with an alpha trade-off parameter. The authors claim that the proposed approach can improve the results over traditional semi-supervised algorithms. The experiments were performed on three datasets. For each dataset, the algorithm runs over 50 times on random partitions of the data. The authors describe how the proposed algorithm obtains better results than existing alternatives by incorporating portions of the propagated labeled objects.

The authors in (Yu et al., 2016) propose a Semi-Supervised Classification using Multiple Clusterings (SSCMC). The algorithm creates a projection of the original samples into random subspaces and applies the clustering algorithms on the projected data points.

In (Yu and Kim, 2018), the authors propose the Consensus rate-based label propagation algorithm. The algorithm creates a consensus matrix from multiple clusterings. Then apply a Label Propagation method by selecting random points to annotate. The propagation algorithm used is the LGC (Zhou et al., 2004a) using the consensus matrix as a weighted matrix. In this work, we used the CRLP algorithm because it is the only one that uses the CM.

The authors in (Berikov et al., 2017) proposed a semi-supervised classification using a combination of ensemble clustering and kernel-based learning. The algorithm has two steps. First, they create a weighted average co-association matrix using multiple clusterings. Then, they use the labeled data to contact a decision function.

The work (Forestier and Wemmert, 2016) focus on cases where the labeled data points are limited and introduces a method combining supervised and unsupervised learning called Semi-supervised learning enhanced by multiple clusterings (SLEMC). This work aims to generate new variables used to enrich the input data, generate clusterings on labeled and unlabeled data, and group data points by maximizing intracluster similarity and intercluster dissimilarity metrics. In having a labeled data point inside a cluster, all the data points receive the same class. However, this is not guaranteed in real life. Therefore, the authors applied combinations of multiple clusterings to avoid this problem.

In (Livieris, 2019), the author presents a new semi-supervised method based on an ensemble approach. Firstly, the author combines a set of well-known individual predictors such as self-training, co-training, and tree-training. The idea behind the scenes is to discover hidden information on the unlabeled data points. Finally, a committed-based ensemble receives the formerly algorithm's outputs to generate a consensus through a maximum probability-based voting scheme. The two primary steps are selection and combination. The experiments were carried over 40 benchmark datasets and varying the training radius in 10%, 20%, 30%, and 40% number of examples. The authors compare the performance against learning algorithms available in the literature. The results of the new algorithm show an improvement compared to traditional semi-supervised learning.

In (Kim and Cho, 2019), the authors demonstrate how the use of semi-supervised learning techniques can strengthens the boundaries of the decision algorithms. Moreover, the authors state that label propagation can effectively learn similar features intra-class. The authors use label propagation and transductive support vector machine to label the unlabeled data points and the Dempster-Shafer theory to determine whether a data point should be annotated or not. During the experiments, the author labeled up to the 20% of the data point of the Lending Club dataset. The authors conclude that the proposed ensemble outperforms the traditional algorithm from the literature.

The paper (Berikov and Litvinenko, 2019) proposes a method combining graph Laplacian regularization and cluster ensemble techniques. To reuse memory and accelerate calculations, the author uses a low rank-decomposition of the similarity matrix.

3.3 Active learning + SSL

The recent acceptance of combining semi-supervised learning and active learning is highlighted in (Yin et al., 2019). Several are the works that combine SSL and AL in tiny contexts such as text classification (Zhu et al., 2003b), language context (Tur et al., 2005; Tomanek and Hahn, 2009), image classification (Zhu et al., 2003b; Zhou et al., 2004b; Long et al., 2008; Yang and Loog, 2019), industrial context (Yin et al., 2019), among others.

In (Zhu et al., 2003b), the authors combine the Semi-supervised Learning and the Active Learning paradigm. This effect can be achieved by combining the Gaussian Random Fields and Harmonic Energy Minimization Function. The idea is to apply a greedy strategy to select the unlabeled data points which minimize the estimated expected classification error (risk) of a harmonic energy minimization function. The authors demonstrate how the proposed framework leads to a more accurate selection of unlabeled data points than previous strategies, such as selecting the data points with maximum label ambiguity. The experiments were performed into synthetic datasets. Also, the algorithm was tested in the handwritten digits recognition and the text classification problems. The results show the proposed active learning algorithm's effectiveness compared to the SVM Most Uncertain and Most Uncertain Query.

The authors in (Zhou et al., 2004b) propose the Semi-supervised Active Image Retrieval (Ssair) Algorithm. This algorithm combines semi-supervised learning and the active learning mechanism. The goal is to exploit the underlying structure of the unlabeled data points to improve image retrieval performance. The algorithm uses the labeled data points to create two different learners. Each learner produced a confidence rank for each image, and the most relevant/irrelevant images are labeled and used as training examples in the next algorithm step. The experiments were performed in a 2000 image sample from the COREL database. There were selected 100 images divided into 20 classes. According to the authors, the results show how the proposed algorithm improves the retrieval performance.

In (Tur et al., 2005), the authors propose two algorithms for spoken language understanding inspired by certainty-based active learning. The first algorithm increases the training dataset with automatically labeled classes for the unlabeled instances, while the second one increases the dataset with a weighted combination of human-labeled classes and automatically labeled classes. These algorithms' goal is to reduce the amount of data that need to be labeled and take advantage of the already labeled instances to predict the unlabeled data points. These algorithms use a Boosting strategy to learn the labeled data points classification and reflect on the unlabeled ones. The authors state that this algorithm is prepared to receive a constant data flow instead of a fixed length dataset. According to the authors, this behavior better reflects a real-world scenario on the spoken language understanding problem. The experiments were carried out on the "How may I help you?" dataset from AT&T. The training process was performed ten times with different training and testing sets ,and finally, the authors reported the mean classification error rate. The authors demonstrate that the combined use of semi-supervised learning and active learning can speed up the learned model's convergence while bypassing inaccurate problems caused by unbalanced data.

In (Long et al., 2008), the authors combine Semi-supervised learning and Active Leaning techniques. They rely on a pool-based active learning approach. This pool-based approach is commonly composed of a learning strategy and a sampling strategy. The authors used graph-based label propagation as the base classifier and used the expected data points entropy to select the data points to be labeled. The sampling strategy tends to select the data point with the maximum expected entropy reduction. The experiments were performed in four datasets, and each dataset was randomly divided into ten equal partitions. One set was used as a testing set, and the remaining as the training sets in each turn, . The training set was also divided into labeled and unlabeled. The labeled data points set starts with one randomly selected instance, and in each iteration, the algorithm sample a single instance to be labeled according to a myopic strategy. The results illustrate a positive balance compared to other sampling methods, such as Random Sampling, Query-by-Committee, and Uncertainty sampling. Finally, the authors claim that the proposed algorithm can handle multi-class learning problems.

In (Tomanek and Hahn, 2009), the authors present and active learning approach where the human annotator has to label the most uncertain subsequences among the selected sequence. To achieve this goal, the authors combine active learning and self-learning in a semi-supervised algorithm. This algorithm uses the Conditional Random Fields as the base sequence classifier and bootstrapping to avoid poor human annotations on critical regions. The experiments were performed in two different domains: the general language newspaper domain and the sub-language biology domain. The results illustrate how the proposed approach can definitively outperform supervised learning approaches. The authors conclude that the algorithm can effectively present sentences useful to the learning task.

In (Park and Kim, 2019), the authors present an active semi-supervised learning algorithm through the combination of multiple sample criteria into a Laplacian kernel. Also, the authors take advantage of the Self-Organizing Map (SOM) in a clustering process. This clustering process provides useful information such as the centroids, the number of labeled samples in each group, and the clusters' size. The idea of the algorithm is to minimize the variance in a Laplacian regularized least squares regression model. The experiments were performed in 10 datasets selected from the UCI online repository. The results show the effectiveness of the proposed algorithm.

The paper (Yin et al., 2019) proposes an active semi-supervised learning method based on the Fisher Discriminant Analysis (ALsemiFDA) model applied to the industrial fault classification task. This algorithm received labeled and unlabeled data points. Firstly, one has to train an FDA with the labeled data points and predict the unlabeled data points after that. Secondly, the algorithm calculated the predicted data points's entropy and labeled the maximum entropy instance with domain experts. Finally, the human-labeled data points are switched into the labeled set and repeat the process until the stop conditions meet.

The authors use four UCI datasets and the Tennessee Eastman Process dataset. The UIC datasets were divided into 70%-30% for training and testing. The training set was divided into 30%-70% for labeled and unlabeled. The Tennessee Eastman Process dataset was divided the same way with an 80%-20% rate. The author concludes that ALsemiFDA algorithm's application on the previously mentioned datasets visually proves the correctness of the idea and the algorithm's effectiveness.

3.4 Discussion

In these label propagation and active learning papers, the selection of data points to be labeled is carried out randomly and maintaining a balance between the classes (Zhu and Ghahramani, 2002; Zhu et al., 2003a; Zhou et al., 2004a; Wang and Zhang, 2007; Yu and Kim, 2018; Yang and Loog, 2019). In some cases, they select a fixed number of data points per class randomly (Yu and Kim, 2018). If the selection does not have all the classes representatives, the selection process continues until all classes have at least one representative data point (Zhu and Ghahramani, 2002; Zhu et al., 2003a; Zhou et al., 2004a).

In the recent active learning article (Yang and Loog, 2019), the problem of selecting the initial set of labeled data points is very well summarized. It also states that this selection of data points has not been widely addressed in the literature, adopting the same random strategy commonly used, selecting a fixed number of elements per class like in (Baram et al., 2004).

Making the selection is only possible in benchmark data sets, but not in data sets corresponding to real-life problems. So, our work comes to give a contribution to this task. Also, it has a simulation process, which presupposes that the dataset is annotated. These simulations of propagation algorithms have an good mean accuracy, which in most cases is the metric used to report the results (Yu et al., 2016; Zhang et al., 2014; Yu and Kim, 2018; Livieris, 2019; Kim and Cho, 2019; Zhang et al., 2020). However, the results do not show a dispersion measure, and the comparison is made only with the central value.

We propose data point selection strategies to substitute the random selection to spend as few resources as possible on the data annotation task. In the next chapter, we will present three heuristics based on stochastic sampling from the co-association matrix's implicit knowledge.

4
Selection strategies for labeling

In this chapter, we present three different strategies for selecting data points to be labeled. They are based on the information that is embedded in the CM. In Section 4.1, we present the Stratified sampling strategy that randomly selects data points with uniform probability based on their importance. In Section 4.2, we present the Probability sampling strategy that randomly selects data points with a probability calculated from the CM information. Section 4.3 presents the Stratified Sampling with Probability selection strategy that combines the two previous strategies.

4.1
Stratified sampling strategy

The Stratified sampling strategy (SS) selects data points based on their importance in a stratified sampling process, which is a method of sampling from a set that can be partitioned into subsets. The idea of stratification is to get data points that have different connection levels in the CM. To do so, we first compute the data point importance $I(dp_i)$, which is given by the sum of all the probabilities related to the i^{th} data point in the CM. Then, it is normalized by the total sum of the CM. In resume, the data point importance is defined in Equation 4-1 as follows:

$$I(dp_i) = \frac{\sum_{j=1}^{n} CM[i,j]}{\sum_{k=1}^{n} \sum_{j=1}^{n} CM[k,j]} \tag{4-1}$$

where n is the number of data points and $\forall x \in \{1,..,n\}$ $CM[x,x] = 0$.

The diagram in Figure 4.1 presents an overview of the proposed method. With the importance of each data point in hand, we put them in a vector, named IO, according to decreasing order of importance. In a sequence, we divide this ordered importance vector into equal-sized strata to have the chance of choice data points from different importance regions. The number of strata is always equal to nr, the number of data points to be selected. At this step, a data point is randomly selected in each stratum. Therefore, we will have nr selected data points with different degrees of importance. This strategy can be considered a non-deterministic method based on stratified sampling.

Figure 4.2 shows an example of how the initial data points are selected

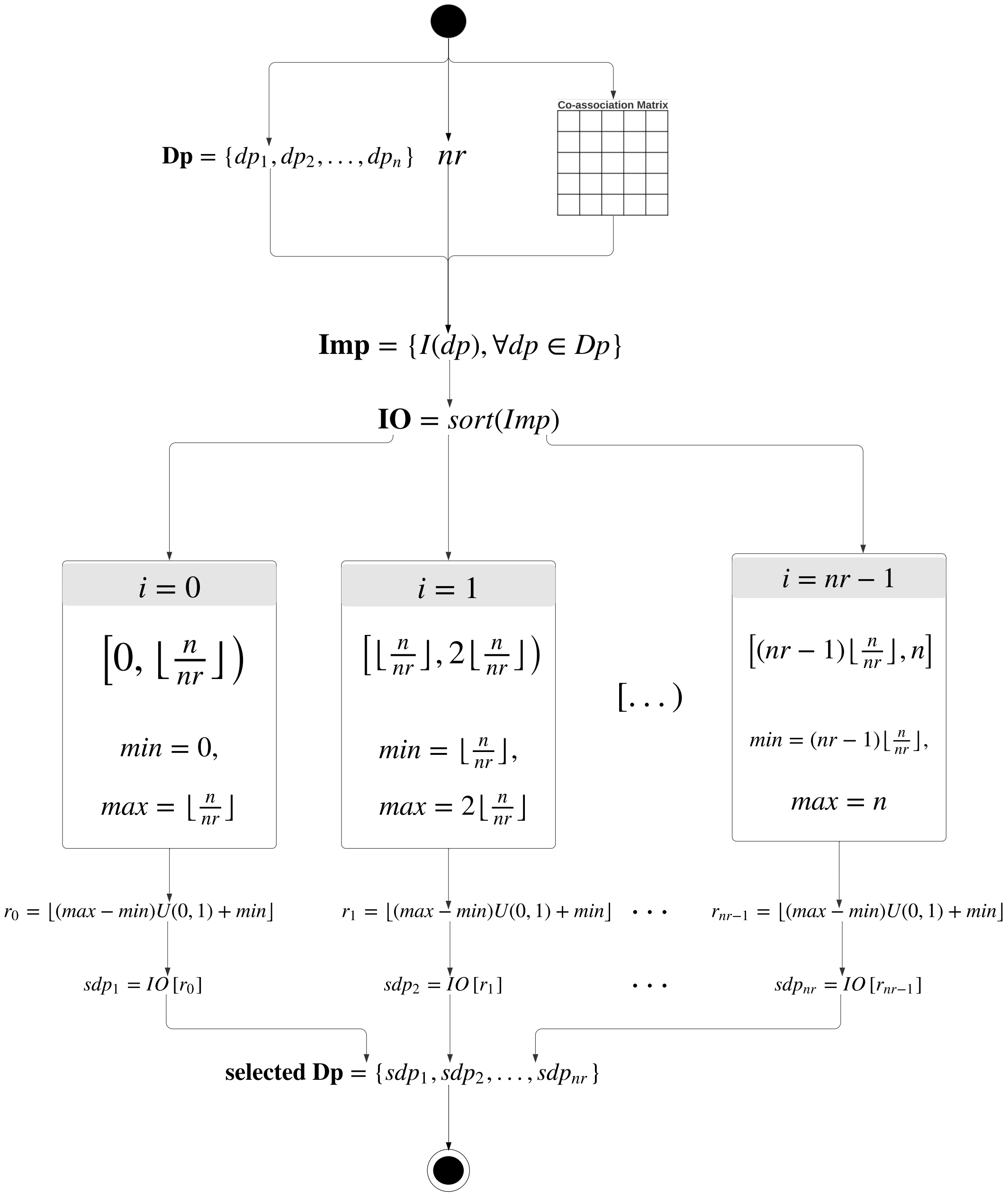

$\mathbf{Dp} = \{dp_1, dp_2, \ldots, dp_n\}$
nr
Co-association Matrix
$\mathbf{Imp} = \{I(dp), \forall dp \in Dp\}$
$\mathbf{IO} = sort(Imp)$
$i = 0$
$[0, \lfloor \frac{n}{nr} \rfloor)$
$min = 0,$
$max = \lfloor \frac{n}{nr} \rfloor$
$i = 1$
$[\lfloor \frac{n}{nr} \rfloor, 2\lfloor \frac{n}{nr} \rfloor)$
$min = \lfloor \frac{n}{nr} \rfloor,$
$max = 2\lfloor \frac{n}{nr} \rfloor$
$[\ldots)$
$i = nr - 1$
$[(nr-1)\lfloor \frac{n}{nr} \rfloor, n]$
$min = (nr-1)\lfloor \frac{n}{nr} \rfloor,$
$max = n$
$r_0 = \lfloor (max - min)U(0,1) + min \rfloor$
$r_1 = \lfloor (max - min)U(0,1) + min \rfloor$
$\cdots$
$r_{nr-1} = \lfloor (max - min)U(0,1) + min \rfloor$
$sdp_1 = IO[r_0]$
$sdp_2 = IO[r_1]$
$\cdots$
$sdp_{nr} = IO[r_{nr-1}]$
$\textbf{selected Dp} = \{sdp_1, sdp_2, \ldots, sdp_{nr}\}$

Figure 4.1: Stratified Sampling strategy diagram to define the initial labeled data points.

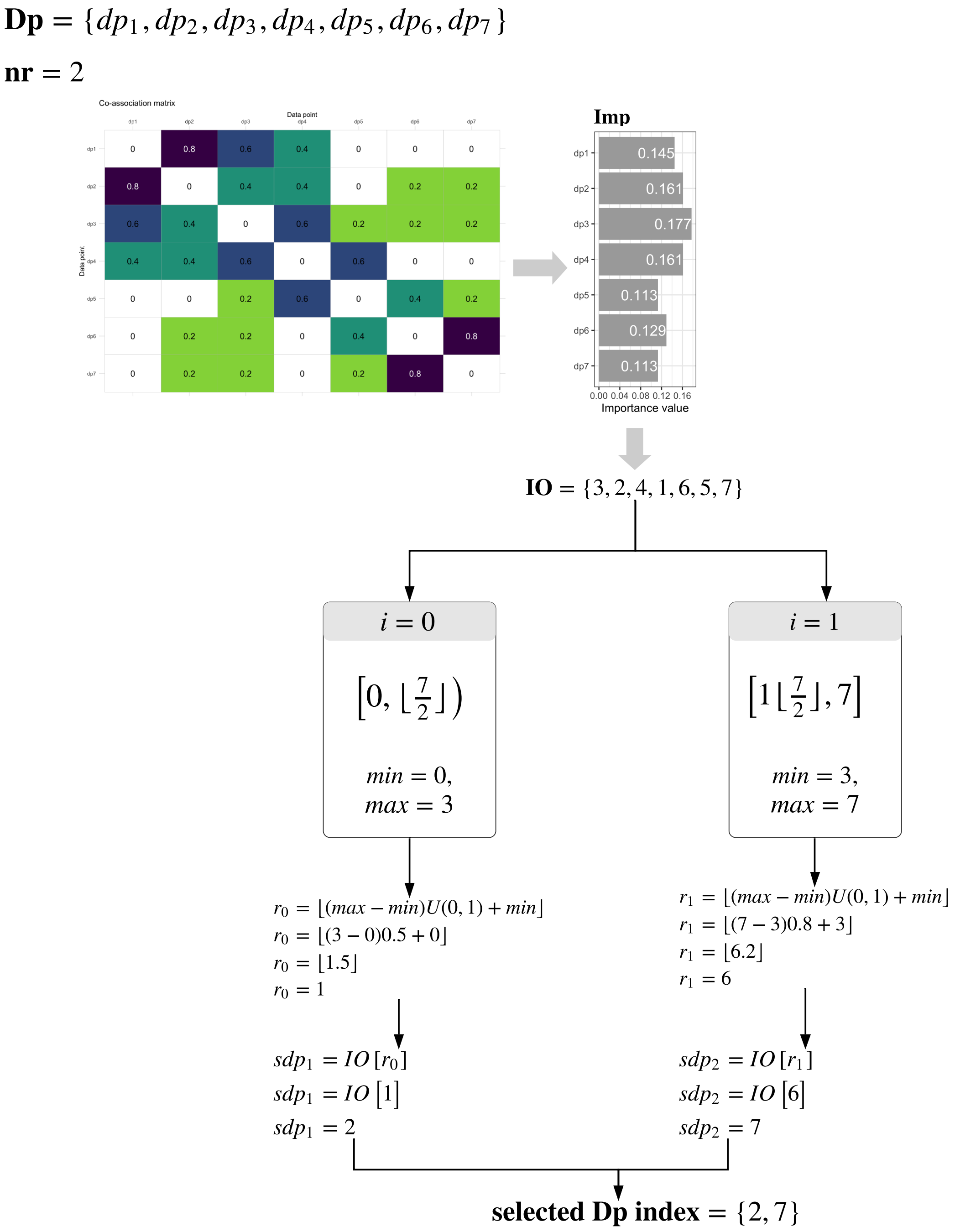

$\mathbf{Dp} = \{dp_1, dp_2, dp_3, dp_4, dp_5, dp_6, dp_7\}$
$\mathbf{nr} = 2$
Co-association matrix
Data point
dp1 dp2 dp3 dp4 dp5 dp6 dp7
Imp
dp1 0.145
dp2 0.161
dp3 0.177
dp4 0.161
dp5 0.113
dp6 0.129
dp7 0.113
0.00 0.04 0.08 0.12 0.16
Importance value
$\mathbf{IO} = \{3, 2, 4, 1, 6, 5, 7\}$
$i = 0$
$\left[0, \lfloor \frac{7}{2} \rfloor\right)$
$min = 0,$
$max = 3$
$i = 1$
$\left[1 \lfloor \frac{7}{2} \rfloor, 7\right]$
$min = 3,$
$max = 7$
$r_0 = \lfloor (max - min)U(0,1) + min \rfloor$
$r_0 = \lfloor (3-0)0.5 + 0 \rfloor$
$r_0 = \lfloor 1.5 \rfloor$
$r_0 = 1$
$r_1 = \lfloor (max - min)U(0,1) + min \rfloor$
$r_1 = \lfloor (7-3)0.8 + 3 \rfloor$
$r_1 = \lfloor 6.2 \rfloor$
$r_1 = 6$
$sdp_1 = IO[r_0]$
$sdp_1 = IO[1]$
$sdp_1 = 2$
$sdp_2 = IO[r_1]$
$sdp_2 = IO[6]$
$sdp_2 = 7$
$\mathbf{selected\ Dp\ index} = \{2, 7\}$

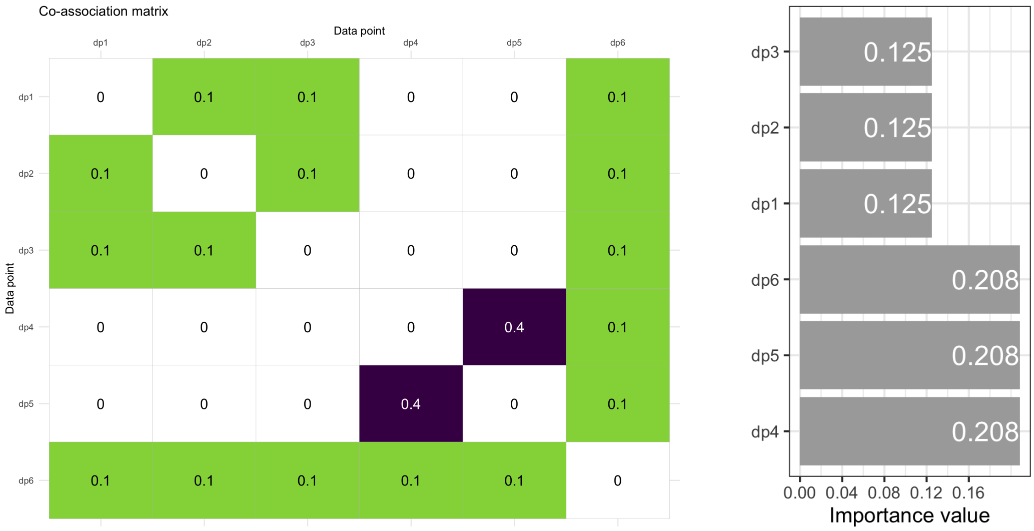
Figure 4.2: Example of Stratified Sampling strategy in a toy dataset.

in a toy dataset. In this example, we have a dataset extracted from (Yu and Kim, 2018). It is made up of seven data points and two classes. Data points dp_1, dp_2, dp_3 and dp_4 belong to the first class (class #1) and data points dp_5, dp_6 and dp_7 belong to the second class (class #2). In addition, there is a co-association matrix showing, for example, that data points dp_1 and dp_2 have the strongest connections, while data points dp_3 and dp_7 have the weakest connections. It could be considered that strong relationships have a similarity value closer to one, and weak relationships have a similarity value closer to zero. The idea is to obtain two representative data points ($nr = 2$). Ideally, one data point is selected from each class to ensure that the propagation process is carried out from a balanced labeled subset. Initially, the importance of each data point is calculated, being this [0.145, 0.161, 0.177, 0.161, 0.113, 0.129, 0.113] and this vector is sorted in decreasing order IO = $[dp_3, dp_2, dp_4, dp_1, dp_6, dp_5, dp_7]$. Then the vector IO is divided into two strata. The first stratum ($[dp_3, dp_2, dp_4]$) comprises the data points that have the highest importance values, and all belong to class #1. The second stratum ($[dp_1, dp_6, dp_5, dp_7]$) have data points from both classes. One data point from each stratum is chosen randomly. The dp_2 data point was sorted from the first stratum, while data point dp_7 was extracted from the second stratum.

CM is fundamental in our strategy. In such a way, if this matrix has noise, the importance indicator will be affected, and this is a limitation that must be addressed. Figure 4.3 illustrates an example of this problem. The dataset comprises six data points like the connections shown in the CM. The importance value for each data point is also observed. In the case of the data point dp_6, it has a weak connection with other data points. In other words, in the multiple partitions made, the dp_6 was always partitioned with different data points. The data points dp_4 and dp_5 are the ones with the strongest connections and, therefore, a greater importance.

In the example, it is possible to notice that the importance of the dp_6 data point is equal to the importance of the dp_4 and dp_5. However, dp_6 is an uncertain data point.

There are several different situations with the same importance problem. Data points changing groups in the multiple partitions are candidates to be defined as confusing or uncertain data points. On the contrary, data points that were together most of the time are candidates to be defined as clear or certain data points.



	dp1	dp2	dp3	dp4	dp5	dp6
dp1	0	0.1	0.1	0	0	0.1
dp2	0.1	0	0.1	0	0	0.1
dp3	0.1	0.1	0	0	0	0.1
dp4	0	0	0	0	0.4	0.1
dp5	0	0	0	0.4	0	0.1
dp6	0.1	0.1	0.1	0.1	0.1	0

Figure 4.3: Example of Stratified Sampling limitation.

4.2 Probability sampling strategy

The Probability sampling strategy (P) defines a probability model that will be used to select a data point on the dataset to label. The idea is to select data points that are in distant regions due to the premise that differently labeled data points are distant. Once one data point is selected, the probability of the other data points to be selected will be proportional to the distance to this one. To define the distance between two data points, we transform the CM into a distance matrix using the Equation 4-2.

$$dm_{i,j} = 1 - CM_{i,j}. \tag{4-2}$$

where i and j are data points.

The diagram in Figure 4.4 shows the main steps of the Probability strategy. Initially, the Probability algorithm receives the input parameters: the data points, the number of desired representatives, and the CM. From the CM, we compute the matrix of distances between each pair of data points, the initial vector of probabilities that starts with equal probability ($1/N$) for each data point, where N represents the total number of data points. Next, the empty set *selected Dp* is created to store the data points that will be randomly selected. The second block corresponds to the selection of points. Then we choose a data point and make a random selection with the probability stored in *current_probability*. The selected data point is removed from the data point list to not be selected again in the next iterations. Furthermore, the selected data point is added to the set *selected Dp*. Then the probabilities update step is performed. In this step the selected points are used to update the vector of

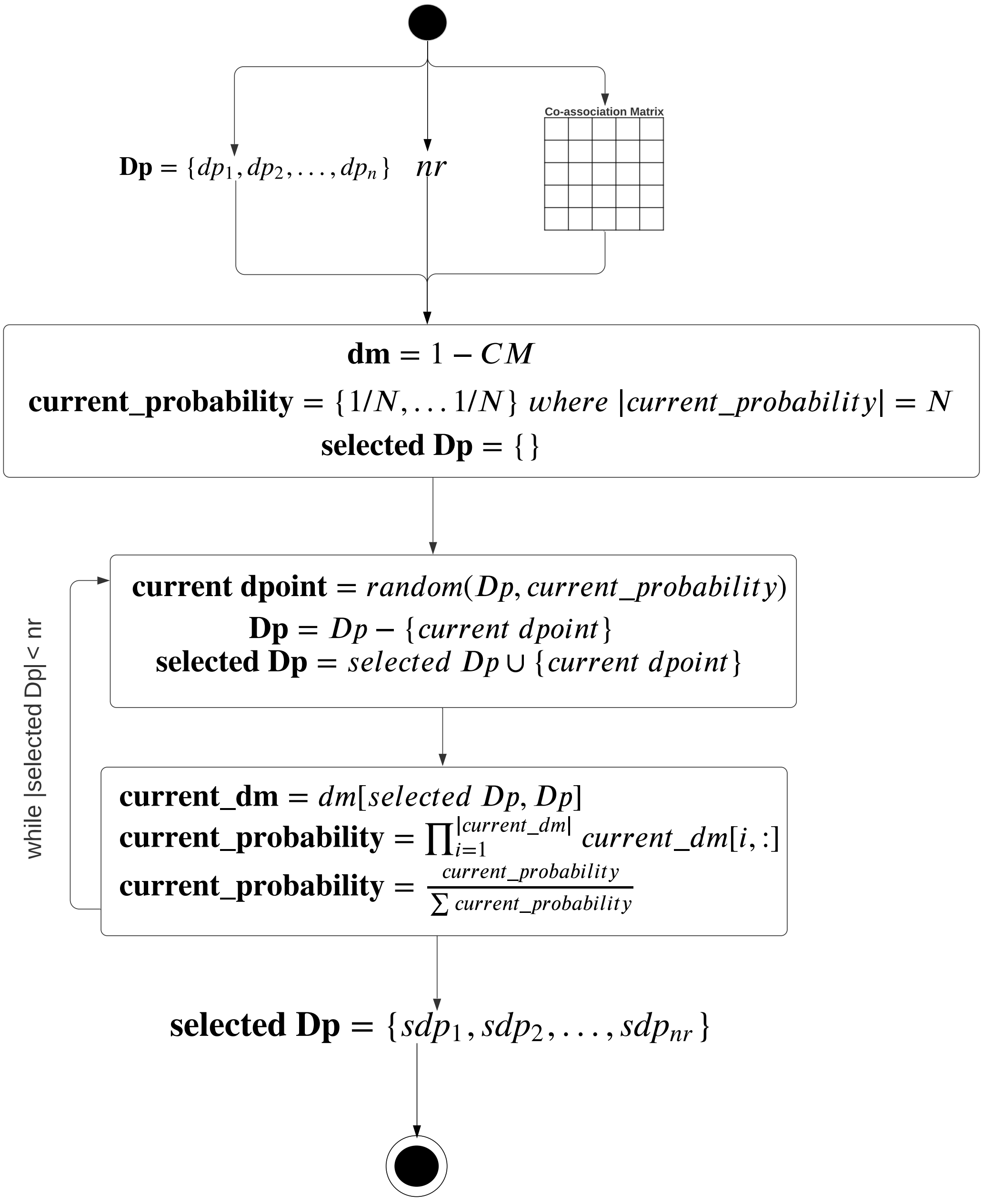
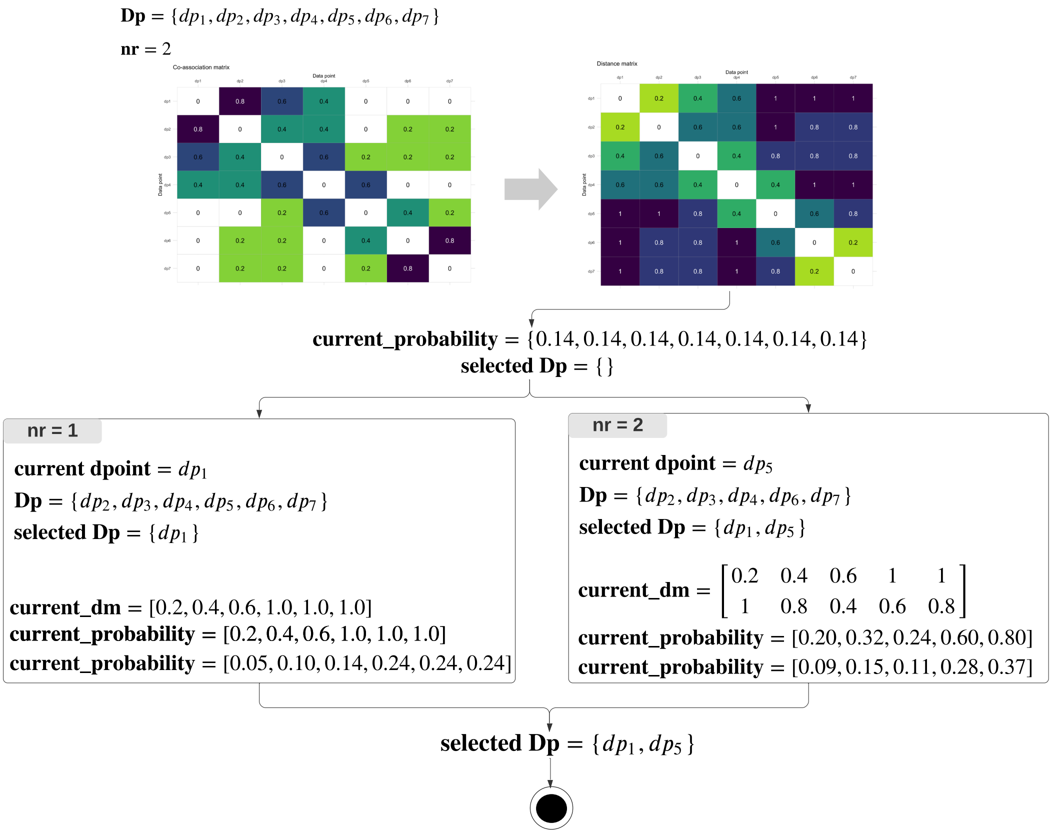

Co-association Matrix
Dp = $\{dp_1, dp_2, \ldots, dp_n\}$
nr
dm = $1 - CM$
current_probability = $\{1/N, \ldots 1/N\}$ *where* $|current\_probability| = N$
selected Dp = $\{\}$
current dpoint = $random(Dp, current\_probability)$
Dp = $Dp - \{current\ dpoint\}$
selected Dp = $selected\ Dp \cup \{current\ dpoint\}$
while |selected Dp|< nr
current_dm = $dm[selected\ Dp, Dp]$
current_probability = $\prod_{i=1}^{|current\_dm|} current\_dm[i,:]$
current_probability = $\frac{current\_probability}{\sum current\_probability}$
selected Dp = $\{sdp_1, sdp_2, \ldots, sdp_{nr}\}$

Figure 4.4: Probability sampling strategy diagram to define the initial labeled data points.

current_probability following the Equation 4-3. In this step, the main idea is to filter the distance matrix by the selected data points and apply a multiplication per column to obtain the probability that the next data point will be drawn and normalize this probability. These last two steps (selection and update) are performed until the desired nr data points have been selected. Finally, we return the set *selected Dp* with the selected data points.

$$\textbf{current\_probability} = \prod_{i=1}^{|current\_dm|} current\_dm[i,:], \tag{4-3}$$

when $current\_dm = dm[selected\ Dp, Dp]$.



Dp = $\{dp_1, dp_2, dp_3, dp_4, dp_5, dp_6, dp_7\}$

nr = 2

Co-association matrix

Data point	dp1	dp2	dp3	dp4	dp5	dp6	dp7
dp1	0	0.8	0.6	0.4	0	0	0
dp2	0.8	0	0.4	0.4	0	0.2	0.2
dp3	0.6	0.4	0	0.6	0.2	0.2	0.2
dp4	0.4	0.4	0.6	0	0.6	0	0
dp5	0	0	0.2	0.6	0	0.4	0.2
dp6	0	0.2	0.2	0	0.4	0	0.8
dp7	0	0.2	0.2	0	0.2	0.8	0

Distance matrix

Data point	dp1	dp2	dp3	dp4	dp5	dp6	dp7
dp1	0	0.2	0.4	0.6	1	1	1
dp2	0.2	0	0.6	0.6	1	0.8	0.8
dp3	0.4	0.6	0	0.4	0.8	0.8	0.8
dp4	0.6	0.6	0.4	0	0.4	1	1
dp5	1	1	0.8	0.4	0	0.6	0.8
dp6	1	0.8	0.8	1	0.6	0	0.2
dp7	1	0.8	0.8	1	0.8	0.2	0

current_probability = $\{0.14, 0.14, 0.14, 0.14, 0.14, 0.14, 0.14\}$
selected Dp = $\{\}$

nr = 1

current dpoint = dp_1
Dp = $\{dp_2, dp_3, dp_4, dp_5, dp_6, dp_7\}$
selected Dp = $\{dp_1\}$

current_dm = $[0.2, 0.4, 0.6, 1.0, 1.0, 1.0]$
current_probability = $[0.2, 0.4, 0.6, 1.0, 1.0, 1.0]$
current_probability = $[0.05, 0.10, 0.14, 0.24, 0.24, 0.24]$

nr = 2

current dpoint = dp_5
Dp = $\{dp_2, dp_3, dp_4, dp_6, dp_7\}$
selected Dp = $\{dp_1, dp_5\}$

current_dm = $\begin{bmatrix} 0.2 & 0.4 & 0.6 & 1 & 1 \\ 1 & 0.8 & 0.4 & 0.6 & 0.8 \end{bmatrix}$
current_probability = $[0.20, 0.32, 0.24, 0.60, 0.80]$
current_probability = $[0.09, 0.15, 0.11, 0.28, 0.37]$

selected Dp = $\{dp_1, dp_5\}$

Figure 4.5: Example of Probability sampling strategy in a toy dataset.

Figure 4.5 shows an example using the Probability strategy in the same toy dataset. In this dm, for example, we see that dp_1 has a maximum distance from the data points dp_5, dp_6, and dp_7. As this algorithm prioritizes data points that are distant in the random selection, the data points dp_1, dp_5, dp_6 and $dp7$ are strong candidates to be selected. Next, the probability is initialized for all data points, this is $1/7 = 0.14$ approximately, and the empty set is created where the selected data points will be stored.

When the number of representatives is 1 ($nr = 1$), a data point with probability 0.14 is randomly drawn for all data points. As a result, dp_1 was obtained, added to the resulting set, and then eliminated from the data points list to avoid being drawn again. Then we calculate the vector of *current_probability* with the values of the row corresponding to dp_1 in dm. We only select the data point dp_1, so it is unnecessary to perform the multiplication by column. Next, we normalize *current_probability* so that the sum of all the elements is 1. This is the probability with which the next data point will be drawn.

When the number of representatives is 2 ($nr = 2$), from the remaining data points $(dp_2, dp_3, dp_4, dp_5, dp_6, dp_7)$ we randomly draw a new data point with probabilities $[0.05, 0.10, 0.14, 0.24, 0.24, 0.24]$, resulting in dp_5 being chosen. The data point dp_5 is removed from the set of data points and added to the set of resulting data points. Then the probability vector is updated for the next draw. Note that now dm filtered by dp_1 and dp_5 is a matrix with two rows and five columns. When we perform the multiplication by column we obtain as *current_probability* the vector with values $[0.20, 0.32, 0.24, 0.60, 0.80]$. Then, we normalize the vector and we get $[0.09, 0.15, 0.11, 0.28, 0.37]$. The latter will be the probability vector to choose the third data point. Since we only want two data points, the selection process ends and we return the representatives dp_1 and dp_5 to be labeled.

The data point dp_1 belongs to class #1 while the data point dp_5 belongs to class #2. One can observe that the initial set of data points will be balanced since it has both classes representatives, so when we perform the label propagation process, it favors a better classification.

4.3
Stratified Sampling with Probability selection strategy

The Stratified Sampling with Probability selection strategy (SSP) combines the two previous strategies to randomly select data points within each stratum with a given probability. In the SS strategy, the data points are randomly selected in each stratum with equal probability, while SSP applies the probability strategy inside the stratum.

The diagram in Figure 4.6 shows the main steps of the SSP algorithm. Given the input parameters: the data points, the number of desired representatives, and the CM, we start the procedure with the initialization step of the variables that will be used, such as the creation of the distance matrix between any pair of data points using the Equation 4-2. From the CM, we obtain the importance vector using the Equation 4-1, and it is ordered in descending order. Also, as part of the initialization, we calculate the initial probability to choose the first point. In this case, all data points have the same probability of being selected. Also, we define the set of ids of the strata that will be explored. In the next step, we randomly select a stratum that will be processed, as explained in the following steps. To process a stratum, it is first removed from the strata set to not be explored again in the next iterations. Based on the selected strata's id, a lower bound and an upper bound are calculated to define the strata. Then the importance vector is filtered using the bounds to obtain the data points corresponding to the strata. Next, we filter the probability vector to keep only the data points of the selected strata. Then a data point is randomly chosen within the strata with the filtered probability. We add the selected data point to the set of *selected Dp*. The next step focuses on updating the probability vector. To do so, we firstly filter the distance matrix by the selected data points, perform a multiplication per column to obtain the probability with which the next data point will be drawn (within the next strata drawn) and then normalize this probability. These last three steps are carried out until nr data points are selected. Finally, the set *selected Dp* with the selected data points is returned.

Figure 4.7 shows how the initial data points are selected in the same dataset used in the two previous subsections. We initially calculate the distance matrix from the CM. Furthermore, we calculate the importance of each data point being [0.145, 0.161, 0.177, 0.161, 0.113, 0.129, 0.113] and we order this vector decreasingly remaining as follows: IO $= [dp_3, dp_2, dp_4, dp_1, dp_6, dp_5, dp_7]$. The initial probability of each data point is $1/7 = 0.14$. We only want two representatives, to divide the importance vector into two strata with ids 0 and 1 and create the empty set where the selected data points will be stored. Both strata have the same probability of being drawn. We randomly draw a stratum and obtain strata one. We update the vector of strata eliminating strata one so that it will not be drawn again. We calculate the lower and upper bound of strata one, which is [3,7], and select the data points dp_1, dp_6, dp_5 and dp_7 belonging to strata one. Randomly, we select a data point with the probabilities [0.14, 0.14, 0.14, 0.14] and we obtained the data point dp_7. We add the data point dp_7 to the set of selected representatives and update the probability vector as being dm filtered by dp_7. Then, we normalize the vector of probabilities and obtain [0.22, 0.17, 0.17, 0.22, 0.17, 0.04, 0].

In the next step, we chose a random stratum. For example, we obtained stratum 0, which was the only one that remained to be explored. Then we calculate the lower and upper bounds of stratum 0 with values [0,3). From these values we filter the importance vector to obtain the data points of this stratum (dp_3, dp_2, dp_4) and the probability vector $(0.17, 0.17, 0.22)$. Then we randomly select the data point dp_4. We add the point dp_4 to the set of selected representatives. Next, we filter dm by dp_7 and dp_4, obtaining a matrix of two

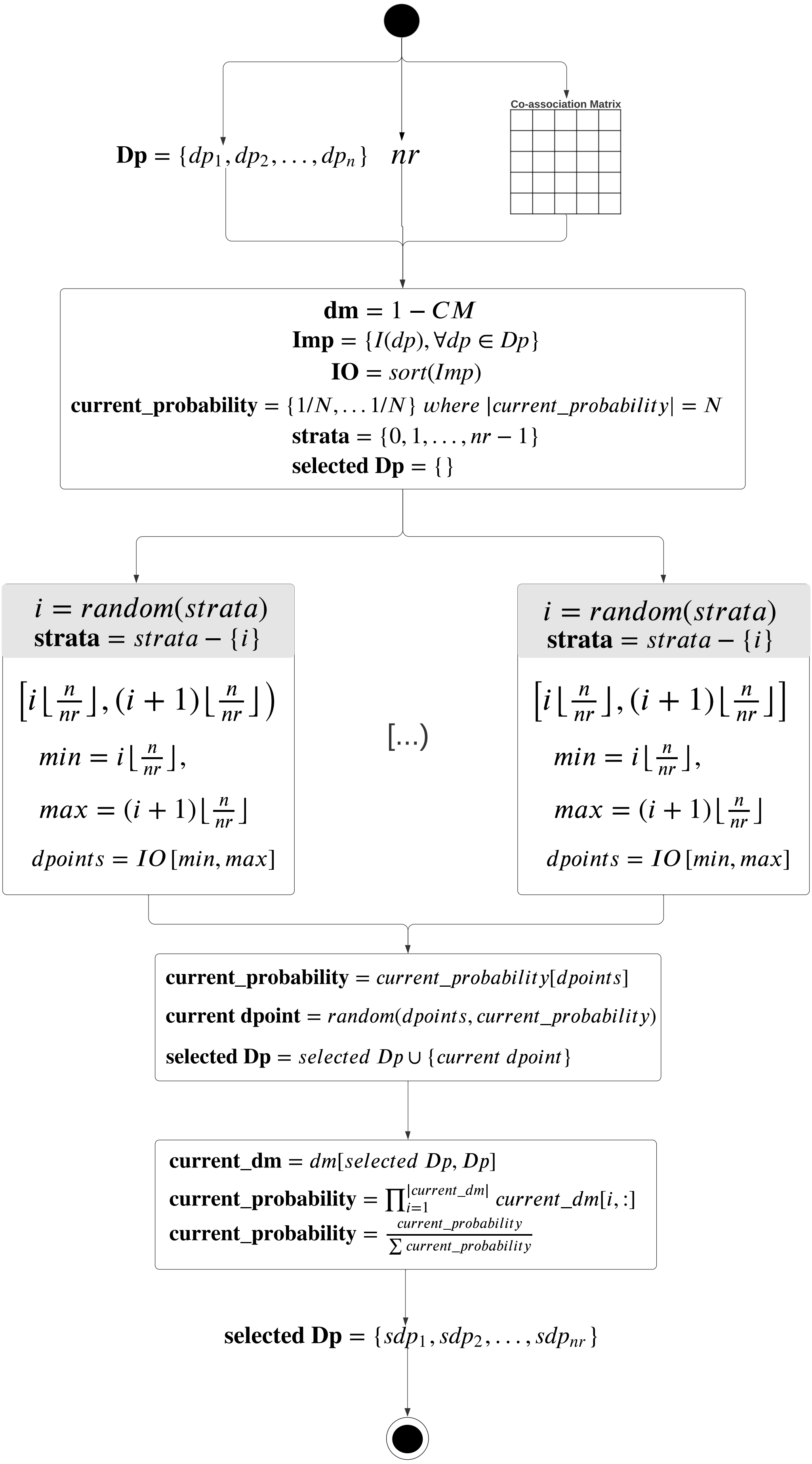
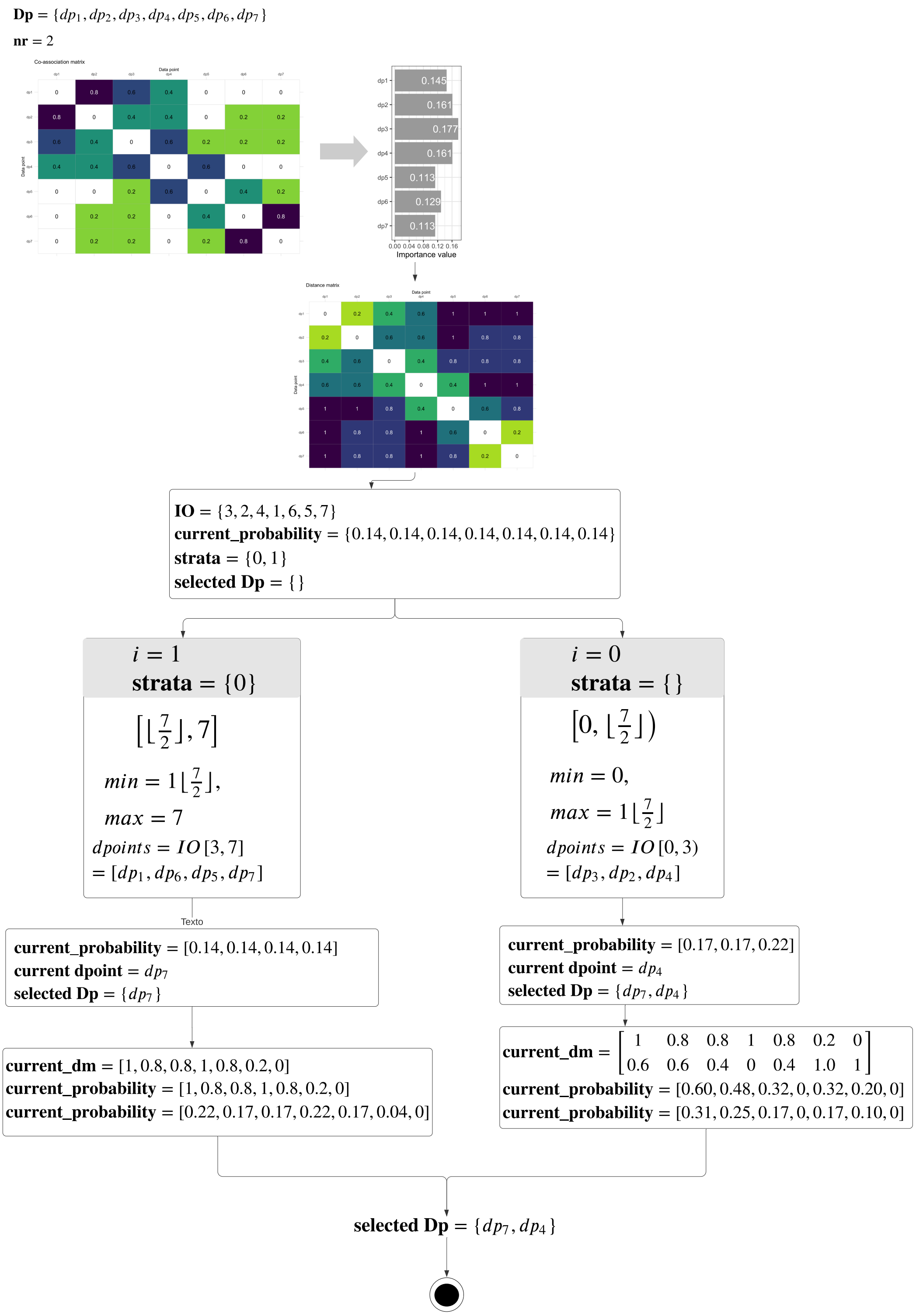

$\mathbf{Dp} = \{dp_1, dp_2, \ldots, dp_n\}$
nr
Co-association Matrix
$\mathbf{dm} = 1 - CM$
$\mathbf{Imp} = \{I(dp), \forall dp \in Dp\}$
$\mathbf{IO} = sort(Imp)$
$\mathbf{current\_probability} = \{1/N, \ldots 1/N\}\ where\ |current\_probability| = N$
$\mathbf{strata} = \{0, 1, \ldots, nr - 1\}$
$\mathbf{selected\ Dp} = \{\}$
$i = random(strata)$
$\mathbf{strata} = strata - \{i\}$
$\left[i\lfloor \frac{n}{nr} \rfloor, (i+1)\lfloor \frac{n}{nr} \rfloor\right)$
$min = i\lfloor \frac{n}{nr} \rfloor,$
$max = (i+1)\lfloor \frac{n}{nr} \rfloor$
$dpoints = IO[min, max]$
[...)
$i = random(strata)$
$\mathbf{strata} = strata - \{i\}$
$\left[i\lfloor \frac{n}{nr} \rfloor, (i+1)\lfloor \frac{n}{nr} \rfloor\right]$
$min = i\lfloor \frac{n}{nr} \rfloor,$
$max = (i+1)\lfloor \frac{n}{nr} \rfloor$
$dpoints = IO[min, max]$
$\mathbf{current\_probability} = current\_probability[dpoints]$
$\mathbf{current\ dpoint} = random(dpoints, current\_probability)$
$\mathbf{selected\ Dp} = selected\ Dp \cup \{current\ dpoint\}$
$\mathbf{current\_dm} = dm[selected\ Dp, Dp]$
$\mathbf{current\_probability} = \prod_{i=1}^{|current\_dm|} current\_dm[i, :]$
$\mathbf{current\_probability} = \frac{current\_probability}{\sum current\_probability}$
$\mathbf{selected\ Dp} = \{sdp_1, sdp_2, \ldots, sdp_{nr}\}$

Figure 4.6: Stratified Sampling Probability diagram to define the initial labeled data points.



$\mathbf{Dp} = \{dp_1, dp_2, dp_3, dp_4, dp_5, dp_6, dp_7\}$

$\mathbf{nr} = 2$

Co-association matrix

Data point	dp1	dp2	dp3	dp4	dp5	dp6	dp7
dp1	0	0.8	0.6	0.4	0	0	0
dp2	0.8	0	0.4	0.4	0	0.2	0.2
dp3	0.6	0.4	0	0.6	0.2	0.2	0.2
dp4	0.4	0.4	0.6	0	0.6	0	0
dp5	0	0	0.2	0.6	0	0.4	0.2
dp6	0	0.2	0.2	0	0.4	0	0.8
dp7	0	0.2	0.2	0	0.2	0.8	0

Distance matrix

Data point	dp1	dp2	dp3	dp4	dp5	dp6	dp7
dp1	0	0.2	0.4	0.6	1	1	1
dp2	0.2	0	0.6	0.6	1	0.8	0.8
dp3	0.4	0.6	0	0.4	0.8	0.8	0.8
dp4	0.6	0.6	0.4	0	0.4	1	1
dp5	1	1	0.8	0.4	0	0.6	0.8
dp6	1	0.8	0.8	1	0.6	0	0.2
dp7	1	0.8	0.8	1	0.8	0.2	0

$\mathbf{IO} = \{3, 2, 4, 1, 6, 5, 7\}$
$\mathbf{current\_probability} = \{0.14, 0.14, 0.14, 0.14, 0.14, 0.14, 0.14\}$
$\mathbf{strata} = \{0, 1\}$
$\mathbf{selected\ Dp} = \{\}$

$i = 1$
$\mathbf{strata} = \{0\}$

$$[\lfloor \tfrac{7}{2} \rfloor, 7]$$

$min = 1\lfloor \frac{7}{2} \rfloor,$
$max = 7$
$dpoints = IO[3, 7]$
$= [dp_1, dp_6, dp_5, dp_7]$

$\mathbf{current\_probability} = [0.14, 0.14, 0.14, 0.14]$
$\mathbf{current\ dpoint} = dp_7$
$\mathbf{selected\ Dp} = \{dp_7\}$

$\mathbf{current\_dm} = [1, 0.8, 0.8, 1, 0.8, 0.2, 0]$
$\mathbf{current\_probability} = [1, 0.8, 0.8, 1, 0.8, 0.2, 0]$
$\mathbf{current\_probability} = [0.22, 0.17, 0.17, 0.22, 0.17, 0.04, 0]$

$i = 0$
$\mathbf{strata} = \{\}$

$$[0, \lfloor \tfrac{7}{2} \rfloor)$$

$min = 0,$
$max = 1\lfloor \frac{7}{2} \rfloor$
$dpoints = IO[0, 3)$
$= [dp_3, dp_2, dp_4]$

$\mathbf{current\_probability} = [0.17, 0.17, 0.22]$
$\mathbf{current\ dpoint} = dp_4$
$\mathbf{selected\ Dp} = \{dp_7, dp_4\}$

$$\mathbf{current\_dm} = \begin{bmatrix} 1 & 0.8 & 0.8 & 1 & 0.8 & 0.2 & 0 \\ 0.6 & 0.6 & 0.4 & 0 & 0.4 & 1.0 & 1 \end{bmatrix}$$
$\mathbf{current\_probability} = [0.60, 0.48, 0.32, 0, 0.32, 0.20, 0]$
$\mathbf{current\_probability} = [0.31, 0.25, 0.17, 0, 0.17, 0.10, 0]$

$\mathbf{selected\ Dp} = \{dp_7, dp_4\}$

Figure 4.7: Example of SS-Probability strategy in a toy dataset.

rows and seven columns. When performing the multiplication by columns, we obtain as *current_probability*, the vector with values [0.60, 0.48, 0.32, 0, 0.32, 0.20, 0]. Then, we normalize the vector and obtain [0.31, 0.25, 0.17, 0, 0.17, 0.10, 0]. The latter will be the vector of probabilities to choose the third data point if necessary. Finally, the selected representatives are returned dp_7 in stratum one and dp_4 in stratum zero. The data point dp_7 belongs to class #2, while the data point dp_4 belongs to class #1. In this example, one could observe that the initial set of data points is balanced, favoring a better classification in the label propagation method.

5 Algorithms

In this chapter, we present an implementation of the proposed data point selection strategies in Section 5.1. We show in Section 5.2 how to transform the selected data points to be used in step 2 of the CRLP algorithm because we made the label propagation using this step. Finally, in Section 5.3, we show the procedure adopted to obtain the learning rate α parameter value.

5.1 Implementation of SS, P and SSP

Algorithm 2 Stratified Sampling strategy to define the initial labeled data points

```
 1: procedure SS($CM, nr$)
 2:     importance ← $I(CM)$ using Equation 4-1
 3:     $sort\_index \leftarrow order(importance)$
 4:     k ← $N/nr$
 5:     $i \leftarrow 0$
 6:     $selected\_index \leftarrow []$
 7:     while $i \leq nr - 1$ do
 8:         $min\_index \leftarrow \lfloor i * k \rfloor$
 9:         $max\_index \leftarrow \lfloor (i+1) * k - 1 \rfloor$
10:         $pu \leftarrow U(0,1)$
11:         $p \leftarrow pu * (max\_index - min\_index) + min\_index$
12:         $selected\_index \leftarrow selected\_index \cup sort\_index[p]$
13:         $i \leftarrow i + 1$
14:     end while
15:     return $selected\_index$
16: end procedure
```

The SS strategy steps are summarized in Algorithm 2, which receives as input the parameters CM and the desired number of representatives (nr). Initially, in line 2, the importance of each data point is calculated through Equation 4-1. The resulting importance vector (importance) is ordered in decreasingly, as indicated in line 3. The ordered importance vector will be divided into *nr* intervals with *k* elements. Each interval is defined by a min value (line 8) and a max value (line 9). For each interval, a number between zero and one is randomly chosen (line 10) and scaled between the min and max values as shown on line 11. This result is added to the list *selected_index* (line 12) and returned at the end of the algorithm (line 15). The selected data points in *selected_index* are then annotated.

Algorithm 3 Probability strategy to define the initial labeled data points

```
 1: procedure PROBABILITY($CM, nr$)
 2:     $N \leftarrow$ number of data points
 3:     $dm_{i,j} = 1 - CM_{i,j}$ (Equation 4-2)
 4:     $points \leftarrow$ list of size $N$
 5:     $current\_probability \leftarrow$ list of size $N$ with $1/N$ values
 6:     $selected\_dp \leftarrow []$
 7:     $i \leftarrow 0$
 8:     while $i \leq nr$ do
 9:         $current\_point \leftarrow$ random select a data point from points with $current\_probability$
10:         $selected\_dp.add(current\_point)$
11:         $points.remove(current\_point)$
12:         $selected\_dm \leftarrow dm[selected\_dp, points]$
13:         $current\_probability \leftarrow$ multiply by column $selected\_dm$
14:         $current\_probability \leftarrow current\_probability/sum(current\_probability)$
15:         $i \leftarrow i + 1$
16:     end while
17:     return $selected\_dp$
18: end procedure
```

Algorithm 3 summarizes the steps of P strategy. It receives as input parameters the CM and the desired number of representatives (nr). In the beginning, all data points have the same probability of being selected (line 5). This probability varies as more data points are selected. The first data point is randomly selected with equal probability, while for the second data point, the probability of being drawn is the distance from the first *dp* drawn to all remaining data points (the row of the *dp* drawn in the *dm*), privileging the most distant data points. When there are more than two data points selected, the probability of selecting the next data point is the product between the distances of the selected data points (rows of the distance matrix indexed by the previous data points - line 12). For example, after having two data points drawn, it must be far from the data points already drawn when choosing a third data point, which implies independence. For this purpose, we make the product of the distances of the data points already drawn. This procedure is performed from lines 7 to 14.

Algorithm 4 Stratified Sampling with Probability strategy to define the initial labeled data points

```
 1: procedure SS PROBABILITY($CM, nr$)
 2:     $N \leftarrow$ number of data points
 3:     $dm_{i,j} = 1 - CM_{i,j}$ (Equation 4-2)
 4:     importance $\leftarrow I(CM)$ using Equation 4-1
 5:     $sort\_index \leftarrow order(importance)$
 6:     k $\leftarrow N/nr$
 7:     $j \leftarrow 0$
 8:     $selected\_index \leftarrow []$
 9:     $strata \leftarrow$ list of size $N$
10:     $current\_probability \leftarrow$ list of size $N$ with $1/N$ values
11:     $selected\_dp \leftarrow []$
12:     while $j \leq nr$ do
13:         $i \leftarrow$ random select a stratum from $stratas$
14:         $strata.remove(i)$
15:         $min\_index \leftarrow \lfloor i * k \rfloor$
16:         $max\_index \leftarrow \lfloor (i+1) * k - 1 \rfloor$
17:         $points \leftarrow sort\_index[min\_index : max\_index]$
18:         $points\_probabilities \leftarrow current\_probability[points]$
19:         $points\_probabilities \leftarrow points\_probabilities/sum(points\_probabilities)$
20:         $current\_point \leftarrow$ random select a data point from points with $points\_probabilities$
21:         $selected\_dp.add(current\_point)$
22:         $selected\_dm \leftarrow dm[selected\_dp, :]$
23:         $current\_probability \leftarrow$ multiply by row $selected\_dm$
24:         $j \leftarrow j + 1$
25:     end while
26:     return $selected\_dp$
27: end procedure
```

The SSP strategy steps are summarized in Algorithm 4, receives as input parameters the CM and the desired number of representatives (nr). This algorithm is a hybrid of the Algorithm 3 (Probability) and the Algorithm 4 (SS). In this case, the data points' importance vector is divided into strata and select a data point for each stratum. Unlike SS, data points have different probabilities of being drawn. In this variant of the algorithm, the strata are selected randomly and not sequentially as in SS. For example, if we have ten strata, one is randomly selected and removed from the strata list, so data points from those strata will not be selected again (lines 13 and 14). Within the stratum, a data point is randomly selected (line 20). The probability of selecting this data point is reduced only to the data points that belong to the selected stratum (The probability vector is obtained in the same way as in the Probability strategy).

The next function shows how we obtain the set of data points to be manually labeled. The Random selection selects data points randomly without replacement and with a uniform probability density. The other algorithms are detailed above.

```
function INITIALDATAPOINTS(CM, nr, strategy_name)
    $data\_points \leftarrow 1 : nrow(CM)$
    if $strategy\_name == random$ then
        $Y_0 = Random(data\_points, nr)$
    else if $strategy\_name == stratified\_sampling$ then
        $Y_0 = SS(CM, nr)$
    else if $strategy\_name == probability$ then
        $Y_0 = P(CM, nr)$
    else
        $Y_0 = SSP(CM, nr)$
    end if
    return $Y_0$
end function
```

5.2 Step 2 of CRLP algorithm

The second step of the CRLP algorithm is summarized in Algorithm 1, lines 11 to 16. We obtain the Y_0 value using the previously defined function *InitialDataPoints*. The selected data points label can be assigned by a domain expert or come from a benchmark dataset. This vector is then transformed into a matrix Y_0. Y_0 is a $n \times m$ matrix where n represents the number of data points in the dataset and m the number of classes. $Y_0[i, j]$ is define as:

$$Y_0[i,j] = \begin{cases} 1, & \text{if the } i^{th} \text{ element was labeled in the } j^{th} \text{ class} \\ 0, & \text{otherwise} \end{cases}$$

The label propagation process can be executed to classify all data points in the dataset. The algorithm performs a label propagation process from the CM and Y_0 to obtain a classification model.

5.3 Alpha tuning

We need to find the α value parameter of the CRLP algorithm. This parameter is the learning rate and specifies the relative amount of information kept from the neighbors and the initial information for each data point. To find the α value, we carried out 30 simulations of the CRLP algorithm with Uniform Random selection varying the alpha in $\{0.2, 0.4, 0.6, 0.8\}$ and 100 iterations. The number of representatives data points varied between $\{nb.class, nb.class + 1, ..., 5 * nb.class\}$.

Finally, we reported the Mean and Mad (Median Absolute Deviation) Accuracy by Alpha. The α was selected with Max's mean accuracy, and in the cases where several the α values returned the same accuracy, we selected the one closer to 0.5. In this case, α value is used to balance the propagation algorithm learning rate and we selected the one closer to 0.5 to keep the balance in the formula, and not bias one formula member. We consider our α chosen to be conservative.

5.4 Active Learning

The scenario for our AL is Pool-based sampling. Figure 5.1 shows the flow of activities of our AL process. We first create the CM, and then we apply the selection strategies to form our initial set of selected data points. The domain expert labels the initial set of data points, and finally, we get the set of labeled data points. We used the CRLP algorithm as a learning model, and the parameter α was the same one defined in Section 5.3. The AL loop was executed five times, and in each iteration, the Query System selects $nb.class$ data points through our selection strategies and the random version. This new set of data points is labeled and added to the labeled data points set, and so on until five iterations are reached. We divided the AL approaches into three algorithms. Firstly, we use the entropy to add data points with greater uncertainty to the selection. Secondly, we update the CM with the oracle's information and finally merge both ideas.

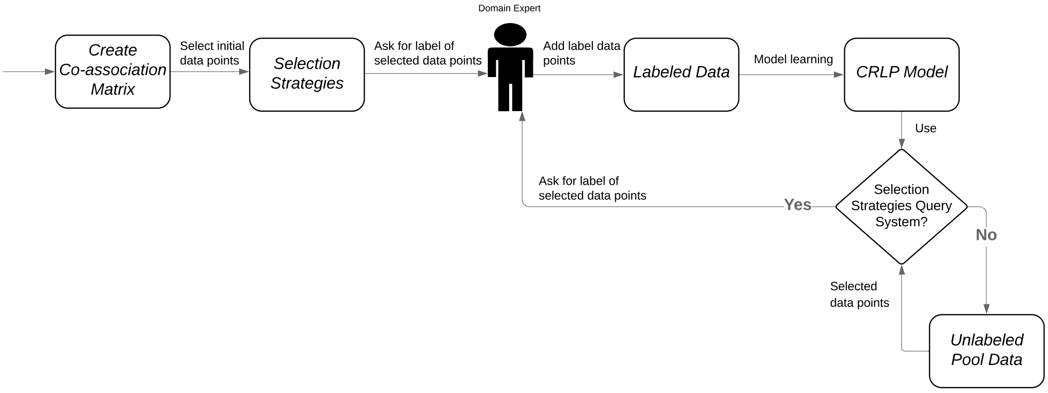

Domain Expert
Create Co-association Matrix
Select initial data points
Selection Strategies
Ask for label of selected data points
Add label data points
Labeled Data
Model learning
CRLP Model
Use
Selection Strategies Query System?
Yes
No
Ask for label of selected data points
Selected data points
Unlabeled Pool Data

Figure 5.1: Flow diagram of Active Learning process.

5.4.1 Active Learning process with uncertainty

The first stage adds the most uncertain data point in each iteration to the set of selected data points, in addition to the data points recommended by the strategies. Following the same diagram as Figure 5.1, we add to the Query System a strategy based on the calculation of uncertainty for each data point of the pool and thus improve the previous results. We calculate the uncertainty of each data point through entropy. We selected the data points closest to 0.5.

The Query System Random Sampling is the most common baseline used to compare AL strategies. This strategy randomly selects data points from the unlabeled pool (Ramirez-Loaiza et al., 2017). In our case, we use the Uniform Random Uncertainty algorithm, which consists of initially making a random selection of data points without considering the classes' knowledge. Then, as part of the QS, we calculate the point's uncertainty obtained from the CRLP algorithm. We select the most uncertain *nb_class* data points to be labeled by the oracle at each iteration.

In the case of the proposed selection strategies, we choose the *nb_class* data points through Probability, SS, or SS-Probability in each case. The data points are selected as follows: the first data point is selected through entropy, and the remaining *nb_class-1* data points are complemented with the proposed selection strategies as detailed in Algorithm 5.

Algorithm 5 Active Learning with uncertainty

```
procedure AL_UNCERTAINTY($CM, nr, max\_iter, strategy\_name, \alpha$)
    $dp \leftarrow \{1, 2, 3, ..., |CM|\}$
    $selected\_dp \leftarrow INITIALDATAPOINTS(CM, nr, strategy\_name)$
    $unlabeled\_dp \leftarrow dp - selected\_dp$
    for $i = 0$ to $max\_iter$ do
        if $|unlabeled\_dp| == 0$ then
            $break$
        end if
        $y\_pred\_prob \leftarrow step2\_CRLP(CM, selected\_dp.labels, \alpha)$
        for $j = 0$ to $|dp|$ do
            $y\_pred[j] \leftarrow index\_of(max(y\_pred\_prob[j]))$
        end for
        $dp\_entropy \leftarrow entropy(y\_pred\_prob)$
        $sorted\_dp\_entropy \leftarrow sort(dp\_entropy, decreasing = TRUE)$
        $unlabeled\_sorted\_dp\_entropy \leftarrow sorted\_dp\_entropy - selected\_dp$
        if $strategy\_name == random$ then
            $max\_entropy\_dp \leftarrow unlabeled\_sorted\_dp\_entropy[1 : nr]$
            $selected\_dp \leftarrow selected\_dp \cup max\_entropy\_dp$
        else
            $max\_entropy\_dp \leftarrow unlabeled\_sorted\_dp\_entropy[1]$
            $selected\_dp \leftarrow selected\_dp \cup max\_entropy\_dp$
            $current\_dp \leftarrow INITIALDATAPOINTS(CM, nr - 1, strategy\_name)$ where not in $selected\_dp$
            $selected\_dp \leftarrow selected\_dp \cup current\_dp$
        end if
        $unlabeled\_dp \leftarrow dp - selected\_dp$
    end for
    return $selected\_dp$
end procedure
```

5.4.2 Active Learning process with CM update

In the second stage, we add the step of updating the CM, as shown in Figure 5.2. This step aims to reflect in the CM the dataset acquired knowledge in each iteration through the oracle.

Let *sdp* be the set of data points selected to be labeled by the oracle

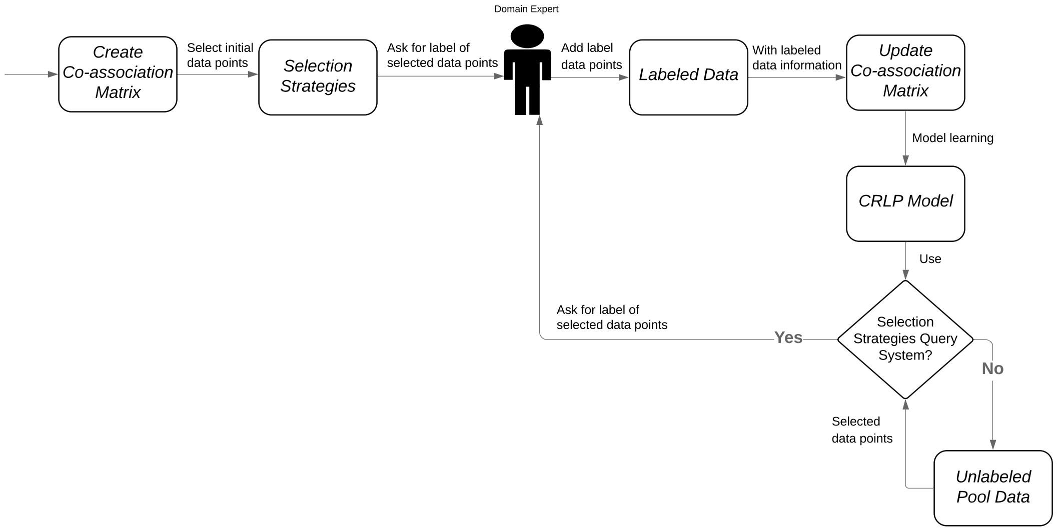

Domain Expert
Create Co-association Matrix
Select initial data points
Selection Strategies
Ask for label of selected data points
Add label data points
Labeled Data
With labeled data information
Update Co-association Matrix
Model learning
CRLP Model
Use
Selection Strategies Query System?
Yes
Ask for label of selected data points
No
Selected data points
Unlabeled Pool Data

Figure 5.2: Flow of Active Learning process with CM update.

and Y(sdp) be the class of each of these data points. The update of the CM is carried out as follows:

$$CM_{i,j} = \begin{cases} 1 & \text{if } Y(sdp_i) = Y(sdp_j) \\ 0 & otherwise \end{cases} \tag{5-1}$$

In other words, the probability obtained from the ensemble of each selected data points pair is updated with a value of 0 if the pair of data points belong to different classes and with a value of 1 if they both belong to the same class as detailed in Algorithm 6. We hypothesize that in this way, the CM will gradually eliminate possible noise. In this way, the CM is better reflecting the knowledge acquired in each iteration of the loop.

It is essential to clarify that the CM's update influences the selection strategies and the model used to perform the classification. Algorithm 7 shows how we merge the AL loop with the CM update.

Algorithm 6 Update CM

```
procedure UPDATE_CM($CM, selected\_dp$)
    for $n$ in $selected\_dp$ do
        for $m$ in $selected\_dp$ do
            if $n \neq m$ then
                if $n.label == m.label$ then
                    $CM[n, m] \leftarrow 1$
                else
                    $CM[n, m] \leftarrow 0$
                end if
            end if
        end for
    end for
end procedure
```

Algorithm 7 Active Learning with CM update

```
procedure AL_CM($CM, nr, max\_iter, strategy\_name, \alpha$)
    $dp \leftarrow \{1, 2, 3, ..., |CM|\}$
    $selected\_dp \leftarrow INITIALDATAPOINTS(CM, nr, strategy\_name)$
    $UPDATE\_CM(CM, selected\_dp)$
    $unlabeled\_dp \leftarrow dp - selected\_dp$
    for $i = 0$ to $max\_iter$ do
        if $|unlabeled\_dp| == 0$ then
            $break$
        end if
        $y\_pred\_prob \leftarrow step2\_CRLP(CM, selected\_dp.labels, \alpha)$
        for $j = 0$ to $|dp|$ do
            $y\_pred[j] \leftarrow index\_of(max(y\_pred\_prob[j]))$
        end for
        $current\_dp \leftarrow INITIALDATAPOINTS(CM, nr, strategy\_name)$
where not in $selected\_dp$
        $selected\_dp \leftarrow selected\_dp \cup current\_dp$
        $UPDATE\_CM(CM, selected\_dp)$
        $unlabeled\_dp \leftarrow dp - selected\_dp$
    end for
    return $selected\_dp$
end procedure
```

5.4.3 Active Learning process with uncertainty and CM update

Finally, the third stage combines the previous ones as it considers both the selection of the most uncertain data points and the CM's updating. As part of the QS in each iteration, we maintain the selection strategies and the selection of the most uncertain data point. In addition, in each iteration, we update the CM as shown in Algorithm 8.

Algorithm 8 Active Learning with uncertainty and CM update

```
procedure AL_UNCERTAINTY_CM($CM, nr, max\_iter, strategy\_name, \alpha$)
    $dp \leftarrow \{1, 2, 3, ..., |CM|\}$
    $selected\_dp \leftarrow INITIALDATAPOINTS(CM, nr, strategy\_name)$
    $UPDATE\_CM(CM, selected\_dp)$
    $unlabeled\_dp \leftarrow dp - selected\_dp$
    for $i = 0$ to $max\_iter$ do
        if $|unlabeled\_dp| == 0$ then
            break
        end if
        $y\_pred\_prob \leftarrow step2\_CRLP(CM, selected\_dp.labels, \alpha)$
        for $j = 0$ to $|dp|$ do
            $y\_pred[j] \leftarrow index\_of(max(y\_pred\_prob[j]))$
        end for
        $dp\_entropy \leftarrow entropy(y\_pred\_prob)$
        $sorted\_dp\_entropy \leftarrow sort(dp\_entropy, decreasing = TRUE)$
        $unlabeled\_sorted\_dp\_entropy \leftarrow sorted\_dp\_entropy - selected\_dp$
        if $strategy\_name == random$ then
            $max\_entropy\_dp \leftarrow unlabeled\_sorted\_dp\_entropy[1 : nr]$
            $selected\_dp \leftarrow selected\_dp \cup max\_entropy\_dp$
        else
            $max\_entropy\_dp \leftarrow unlabeled\_sorted\_dp\_entropy[1]$
            $selected\_dp \leftarrow selected\_dp \cup max\_entropy\_dp$
            $current\_dp \leftarrow INITIALDATAPOINTS(CM, nr - 1, strategy\_name)$ where not in $selected\_dp$
            $selected\_dp \leftarrow selected\_dp \cup current\_dp$
        end if
        $UPDATE\_CM(CM, selected\_dp)$
        $unlabeled\_dp \leftarrow dp - selected\_dp$
    end for
    return $selected\_dp$
end procedure
```

6
Experimental study

This chapter describes how we evaluate our proposal, present the results obtained, and discuss their implications to the LP and AL areas. In Section 6.1 we present the datasets used in our experiments, and a general explanation about three executed experiments and used metrics(subsection 6.2). First, to know if the real distribution of the classes in the selection is maintained, we compared in Section 6.3 the results of Selection strategies with the Random selection. Second, in Section 6.4, we compare the random selection and the selection strategies with the CRLP algorithm. Finally, Section 6.5 applies the selection strategies in the area of Active Learning.

6.1
Benchmark

In our experiments, we used 15 well-known datasets, 14 are used in (Yu and Kim, 2018), and the 15th is the MNIST test dataset (Chollet et al., 2015). Most of them are available in the UCI Machine Learning Repository (Lichman et al., 2013), such as Wine, Seeds, Congressional Voting, Vertebral, Breast Cancer 1 and 2, Synthetic control chart, Balance Scale, Urban Land Cover, and Segmentation Image. The remaining datasets can be found in the following sources: Leukemia in (Boulesteix et al., 2018), Lymphoma in (Chung et al., 2019), Armstrong in (Armstrong et al., 2002), Chen 2002 in (Chen et al., 2002). Table 6.1 gives details of these datasets. Only Congressional and Balance Scale datasets have categorical attributes, and the remaining of them have continuous attributes. Notice that, in the majority of the datasets, the classes are unbalanced. The visual representation of CM for 14/15 datasets is presented in Appendix A.7.

6.2
Procedure

Three experiments were carried out. We run 100 simulations in each experiment. The first experiment compared if Uniform Random selection with the Selection strategies (with and without replacement) maintained the real distribution of the classes in the selection. The second experiment was to compare

Table 6.1: Overview of datasets.

	Dataset	Nb.Instances	Nb.Attribute	Nb.Class	Balanced
1	Leukemia	38	3051	2	No
2	Lymphoma	62	4026	3	No
3	Armstrong	72	2194	3	No
4	Wine	178	13	3	No
5	Chen 2002	179	85	2	No
6	Seeds	210	7	3	Yes
7	Congressional	232	16	2	No
8	Vertebral	310	6	3	No
9	Breast Cancer 1	569	30	2	No
10	Synthetic	600	60	6	Yes
11	Balance Scale	625	4	3	No
12	Urban	675	147	9	No
13	Breast Cancer 2	683	9	2	No
14	Segmentation	2310	19	7	Yes
15	Mnist Test	10000	784	10	No

the CRLP algorithm's accuracy with the different selection strategies and the LP process (with and without replacement). The third experiment was to compare the accuracy of the CRLP algorithm in the AL process.

In all of the experiments, a simulation process was carried out varying the desired number of representatives. In the experiment with replacement, the number of selected representatives varies continuously between nb.class and 5*nb.class in each dataset $\{nb.class, nb.class + 1, ..., 5 * nb.class\}$ while in the case without replacement goes from $\{nb.class, 2 * nb.class, ..., 5 * nb.class\}$. We have a premise that we do not know the class of data points a priori. Such fact differs from the paper that present the CRLP algorithm Yu and Kim (2018). In the experiment with replacement, the same data point for a different number of representants can be drawn more than once while in the case without replacement, a multiple of the number of classes is always chosen as the number of representatives to have the same number of data points per class. However, as the selection is blind, we cannot guarantee it. In this case, only the strata that have not yet been drawn are sampled, thus maintaining the previously selected data points.

The terms *number of desired representatives*, *nb.class*, *number of labeled observations* and *number of queried samples* refer to the number of selected data points.

The metrics to assess the first experiment were Jensen-Shannon (JS) Divergence, reporting Mean and Standard Deviation (Sd), while in the second and third experiments, we also used Mean and Sd to report the accuracy results.

The JS Divergence (Lin, 1991) measure the similarity between two mass probability functions, which its square root can be considered a distance metric between probability mass functions. Also, the smaller the divergence value, the better is the result. The mass probability function of the selected data point's classes generated by the proposed selection strategies should be similar to the mass probability function of the classes in the original dataset. The JS-Divergence is given by Equation 6-1:

$$D^{(J)}(P||Q) = H^{(S)}\left[\frac{P+Q}{2}\right] - \frac{H^{(S)}[P]}{2} - \frac{H^{(S)}[Q]}{2}, \tag{6-1}$$

where P and Q are mass probability functions, and $H^{(S)}$ (Shannon and Weaver, 1963) is the Shannon entropy, which is given by:

$$H^{(S)}[P] = -\sum_{j=1}^{N} p_j \ln(p_j).$$

We implement the selection strategies and the CRLP algorithm in the language `Pyhton` (Van Rossum and Drake, 2009). To calculate the accuracy in each simulation we use the metric implemented in the `scikit-learn package` (Pedregosa et al., 2011). We calculate the Shannon Jensen divergence by applying the Jennsen Shannon distance's square root defined in the `scipy package` (Virtanen et al., 2020). We execute the experiments on a computer with an Intel Core i7 processor, 3.30 GHz, and 64 Gb of RAM.

6.3 Selection strategies experiment

In this section, we present the first experiment that compares the Uniform Random selection with the proposed selection strategies (with and without replacement). Here, we look to the aspect that the selected set of data points maintains the original dataset's classes distribution. The subsection 6.3.1 presents the results obtained with the Selection strategies with replacement and the subsection 6.3.2 presents the results obtained with the Selection strategies without replacement.

6.3.1 Selection strategies with replacement

Figure 6.1 and Figure 6.2 shows the results. In most of the datasets, the Uniform Random strategy is observed in red with a higher mean divergence and a higher standard deviation. This means that it would not choose Uniform Random, but any other proposed selection methods in a general way. Further-

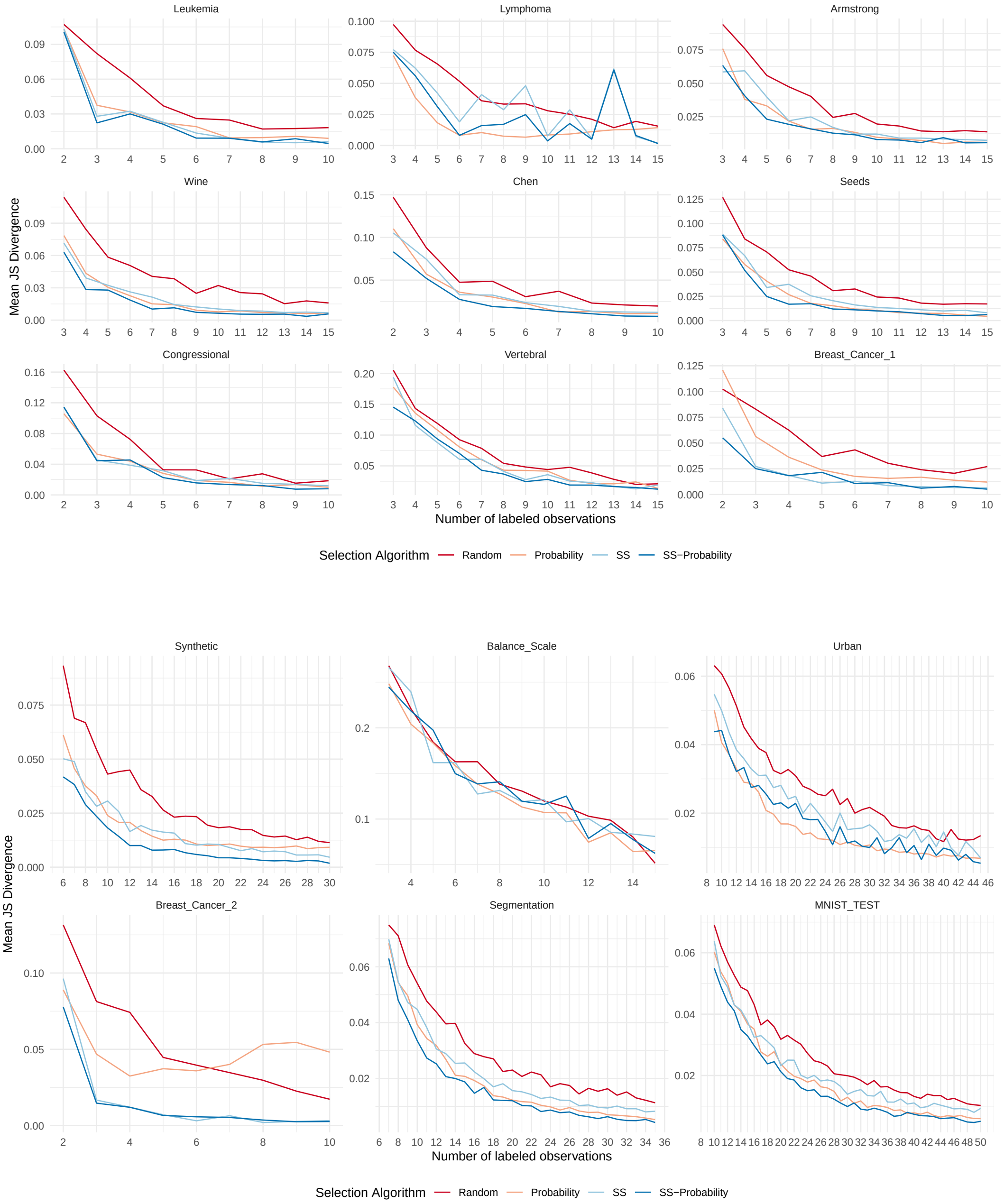


Figure 6.1: Mean Jensen Shannon Divergence curves of the Random, Probability, Stratified Sample and Stratified Sample Probability algorithms with replacement when applied to 15 datasets.

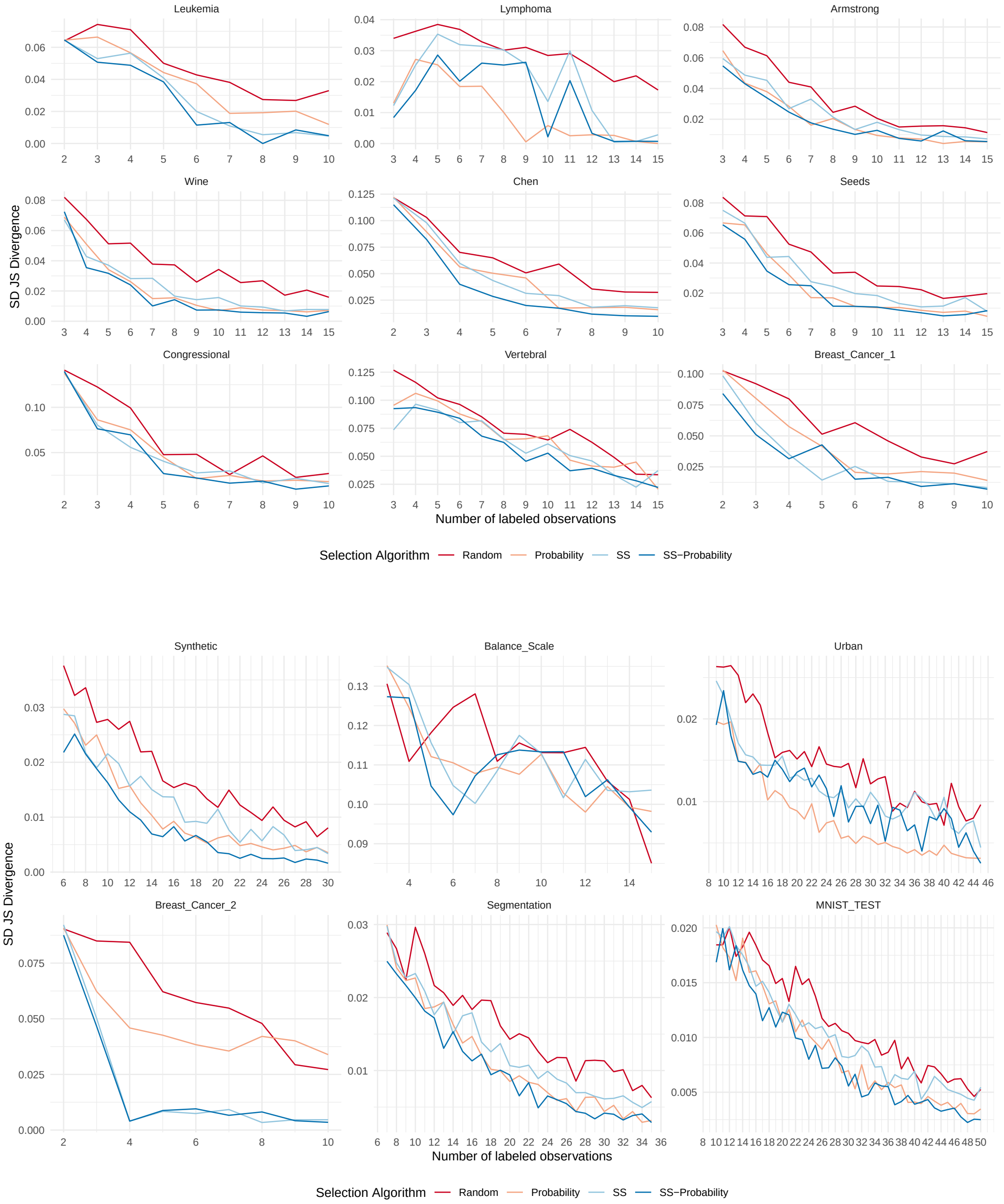


Figure 6.2: Standard Deviation Jensen Shannon Divergence curves of the Random, Probability, Stratified Sample and Stratified Sample Probability algorithms with replacement when applied to 15 datasets.

more, we observe that most of the times the rest of the selection algorithms have similar mean divergence values with a tendency to be less between the SS and SS-Probability methods. This suggests a positive effect of the SS method. In many situations, the SS-Probability has a simple gain over the SS method. On average, these selection strategies respect the original classes' distribution more than the Uniform Random selection in red. Based on the observations, Probability, SS, and SS-Probability favor a more appropriate selection. However, this is not the case for the Lymphoma and Balance Scale datasets.

In the case of the Lymphoma dataset, which has 62 data points divided into three classes, initially the mean divergence of Probability, SS and SS-Probability remain smaller than in the Random method (Figure 6.1). However, from representative number six, it begins to increase, having a critical peak with 13 representative data points in the SS and SS-Probability methods. The peak may be due to one missing class representative in the selection. For example, in the case of SS, dividing the ordered importance vector by 13, the first nine strata correspond to data points only of class #1, strata ten and eleven only have data points of class #3, and the data points from strata twelve and thirteen, mostly belong to class #2, but have one representative from class #1 and another from class #3. In this case, elements of class #2 may never be sampled, favoring these high divergence values. However, the Probability method performs much better than Uniform Random, SS, and SS-Probability. In general, it obtains lower values of mean divergence and standard deviation. In the case of the Lymphoma dataset, the strata affected the performance of the selection strategy.

The Balance Scale dataset is naturally unbalanced. Of the 625 data points divided into three classes, only 49 belong to class #1, 288 belong to class #2, and 288 belong to class #3. The number of representatives seven has the highest peak in divergence (Figure 6.1). When dividing the ordered vector of importance into seven strata, the 49 data points of class #1 were diluted between these strata, but in less quantity than the remaining data points. So it may be that the first class is never drawn, negatively impacting the divergence in the SS and SS-Probability methods. In the Probability strategy, the mean divergence is less than the Uniform Random in 1415 representatives.

6.3.2
Selection strategies without replacement

Figure 6.3 and Figure 6.4 show the results. In general, this selection approach without replacement and the selection of multiples of the *nb.class* generates smooth curves. Furthermore, in most datasets, the mean JS divergence

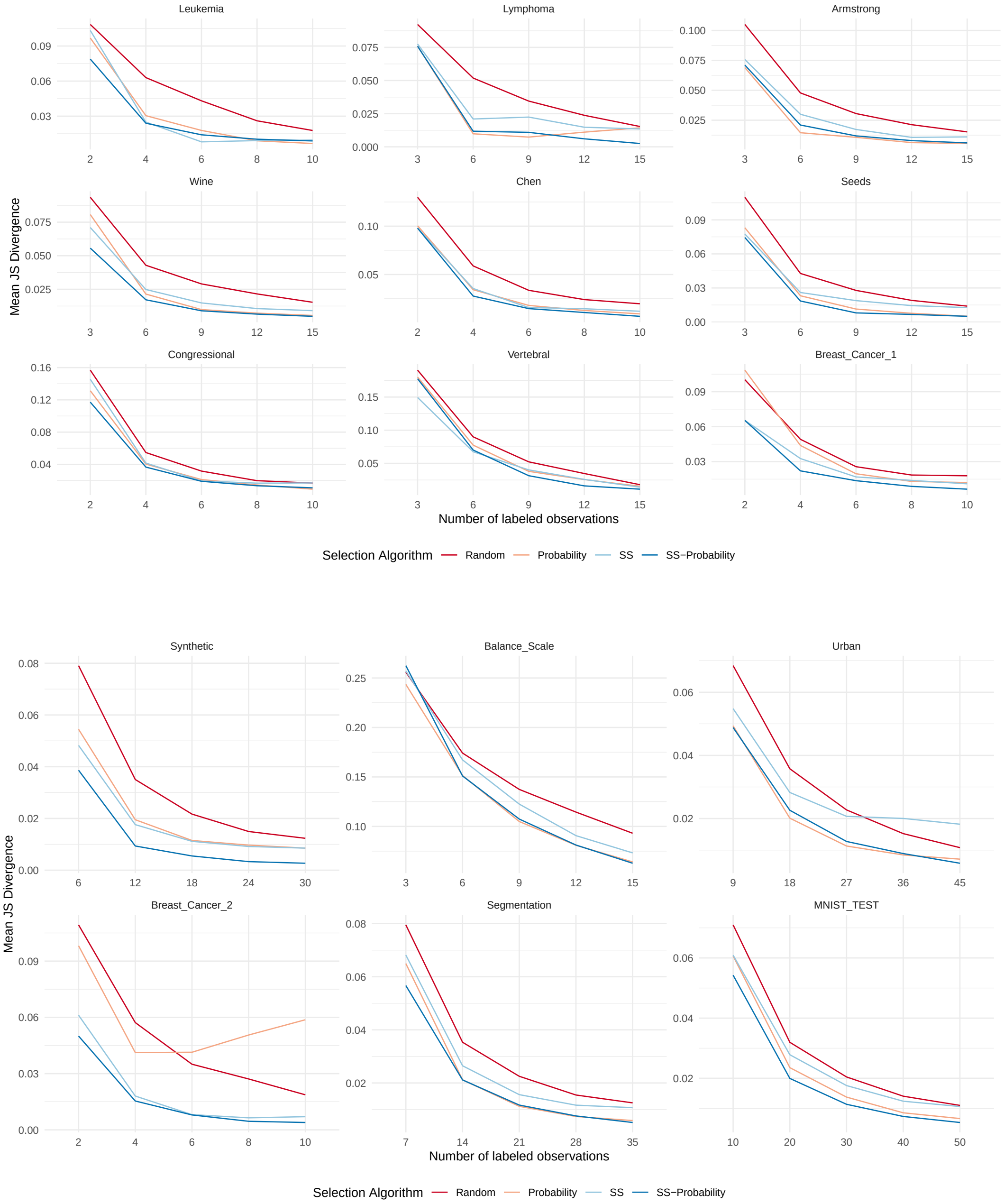


Figure 6.3: Mean Jensen Shannon Divergence curves of the Random, Probability, Stratified Sample and Stratified Sample Probability algorithms without replacement when applied to 15 datasets.

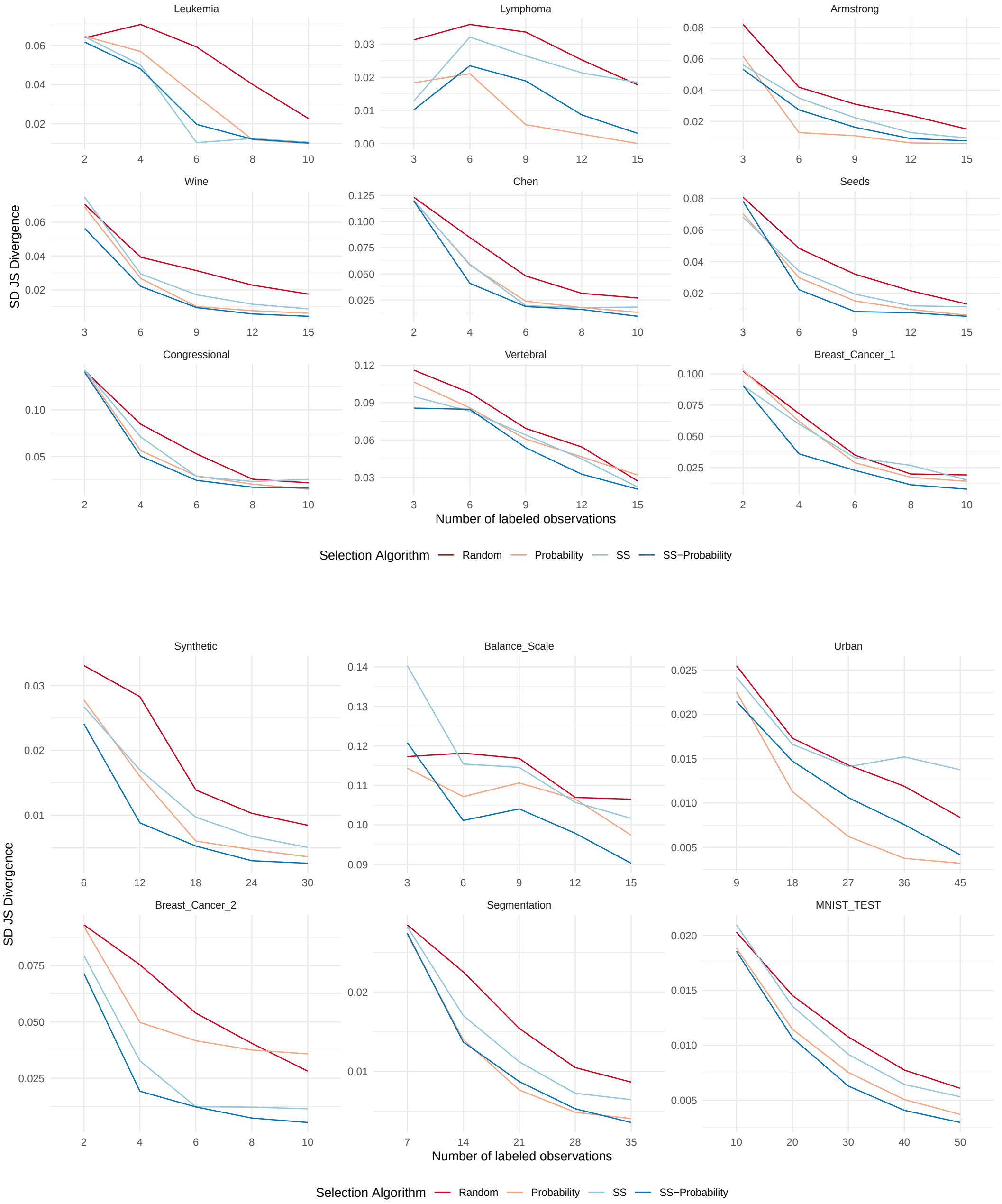


Figure 6.4: Standard Deviation Jensen Shannon Divergence curves of the Random, Probability, Stratified Sample and Stratified Sample Probability algorithms without replacement when applied to 15 datasets.

from the P, SS, and SSP strategies tend to be less than the mean divergence from the Uniform Random strategy. In contrast to this behavior, the standard deviation of the selection strategies in this approach tends to be less than the Uniform Random method's standard deviation.

In summary, these results and visualizations verify that, on average, the P, SS, and SSP strategies generate the mass probability functions more similar to the real mass probability function of the classes in the hole dataset when compared to the Uniform Random selection is more suitable than random selection. The analysis described above helps us to answer RQ1.

6.4 Label Propagation experiment

We run a second experiment to compare the CRLP algorithm's accuracy considering the different selection strategies with and without replacement. In Table 6.2 it is possible to see the α value for each dataset obtained through the procedure described in Section 5.3. Appendix A.1 presents more detailed information on α selection. These α values will be used in this experiment for all selection strategies.

Table 6.2: Hyperparameters for the CRLP blind algorithms using Random, Stratified Sampling, Probability and Stratified Sampling with Probability as selected initial data points.

		(Random and Selection strategies)
	Dataset	Blind_CRLP
1	Leukemia	$\alpha = 0.4$
2	Lymphoma	$\alpha = 0.4$
3	Armstrong	$\alpha = 0.2$
4	Wine	$\alpha = 0.2$
5	Chen 2002	$\alpha = 0.2$
6	Seeds	$\alpha = 0.2$
7	Congressional	$\alpha = 0.4$
8	Vertebral	$\alpha = 0.4$
9	Breast Cancer 1	$\alpha = 0.4$
10	Synthetic	$\alpha = 0.2$
11	Balance Scale	$\alpha = 0.2$
12	Urban	$\alpha = 0.2$
13	Breast Cancer 2	$\alpha = 0.4$
14	Segmentation	$\alpha = 0.2$
15	Mnist Test	$\alpha = 0.4$

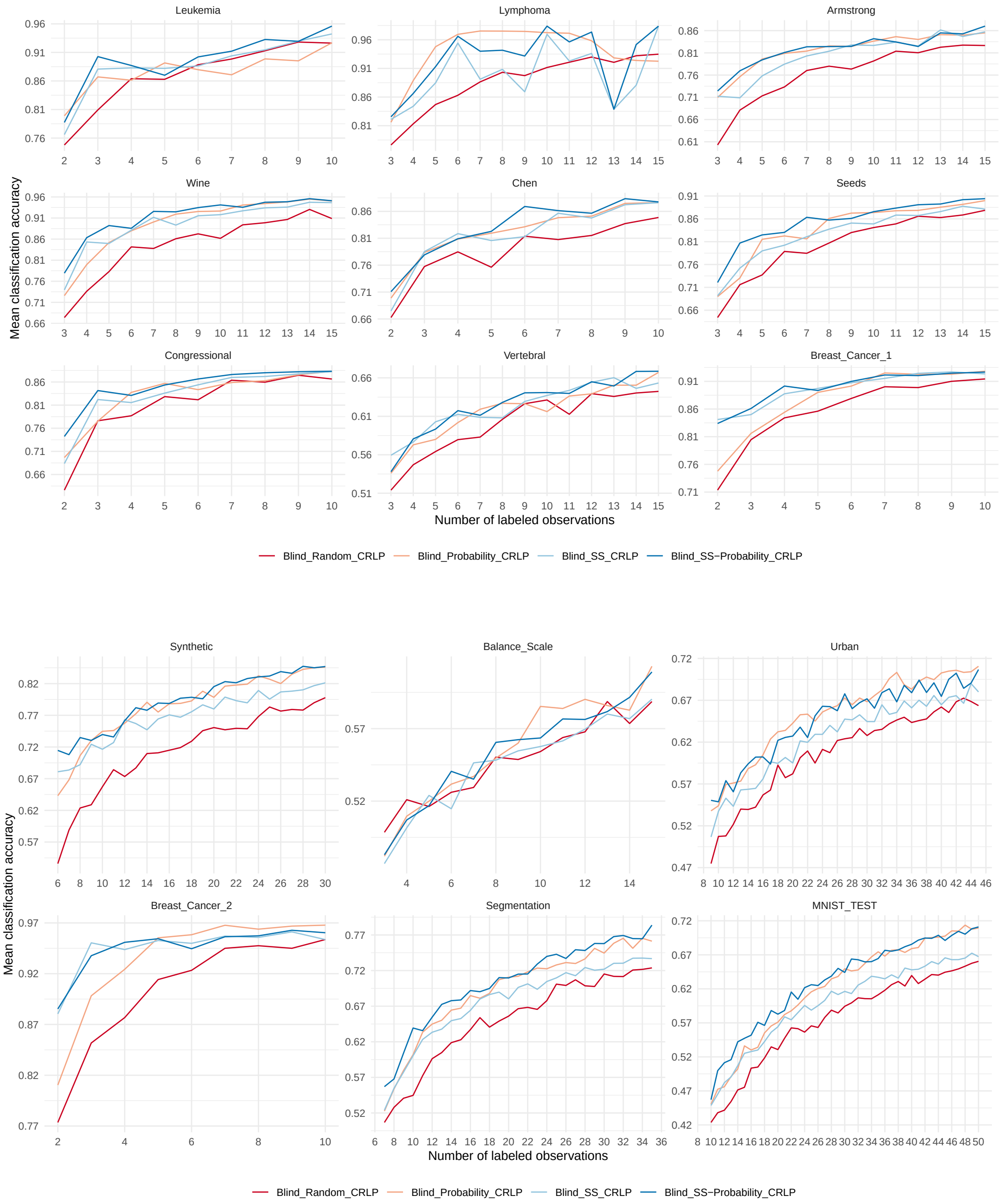


Figure 6.5: Classification mean accuracy curves for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms with replacement in 15 datasets.

6.4.1 Selection strategies applied to CRLP algorithm with replacement

This subsection presents the results obtained with the Selection strategies applied to the CRLP algorithm with replacement. Figure 6.5 and Figure 6.6 show the results.

It is observed that, on average, higher accuracy is obtained through the P and SSP selection strategies rather than making the selection randomly.

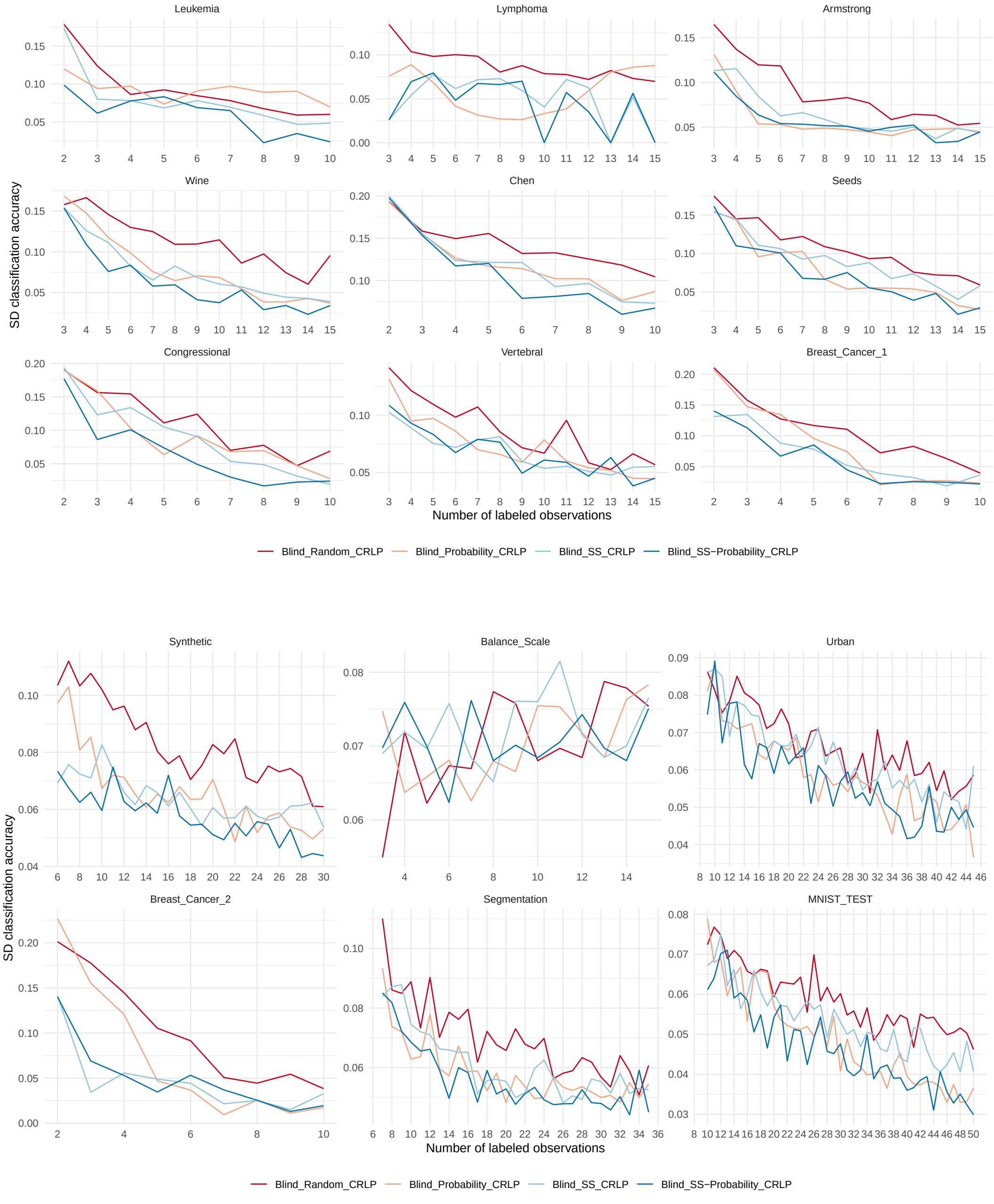


Figure 6.6: Classification standard deviation accuracy curves for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms with replacement in 15 datasets.

This example indicates that the P and SSP strategies are more accurate than the Uniform Random selection strategy in most datasets. A minor standard deviation shows that our selection strategies are consistent in the 100 simulations. Again, SS and SSP do not perform well on Lymphoma and Balance Scale datasets (Figure 6.5), but it was expected due to the selection problems explained in the first experiment.

Appendix A.2 show the accuracy dispersion over 100 simulations. On the mean the selection strategies gets a higher accuracy value with an equal or better dispersion on most of the datasets.

6.4.2 Selection strategies and CRLP algorithm without replacement

This subsection presents the results obtained with the Selection strategies applied to the CRLP algorithm without replacement. Figure 6.7 and Figure 6.8 show the obtained results.

The curves are smoother than in the experiment with replacement. We can see that, on average, label propagation using Uniform Random selection (red color) generates lower accuracy values and higher standard deviation than our strategies. In the Lymphoma and Balance Scale datasets, the CRLP with Random selection has better performance than SS_CRLP (light blue) and SS-Probability_CRLP (dark blue); but the Probability_CRLP (orange color) obtained higher values of accuracy. Our strategies show consistency with the selection since on average, the standard deviation is less than the Uniform Random standard deviation. The results demonstrate an evident impact on selecting the initial data points when applying the CRLP algorithm.

Appendix A.3 shows the accuracy dispersion over all simulations. On the mean, the proposed selection strategies get a higher accuracy value on most of the datasets. As the number of representatives increases, the dispersion in the Uniform Random strategy is greater than the dispersion in the selection strategies. SS-Probability appears to be a good candidate as an initial data point selection algorithm. These results help us to answer our RQ2.

6.4.3 Summary of LP experiment

Figure 6.9 presents a summary of the second experiment performed. For each dataset, all the simulations carried out are shown without distinguishing between the number of representatives. Results show the SSP dominates the other selection procedures, also presenting smaller dispersion.

By experimentation, we know that we obtain a greater accuracy with a

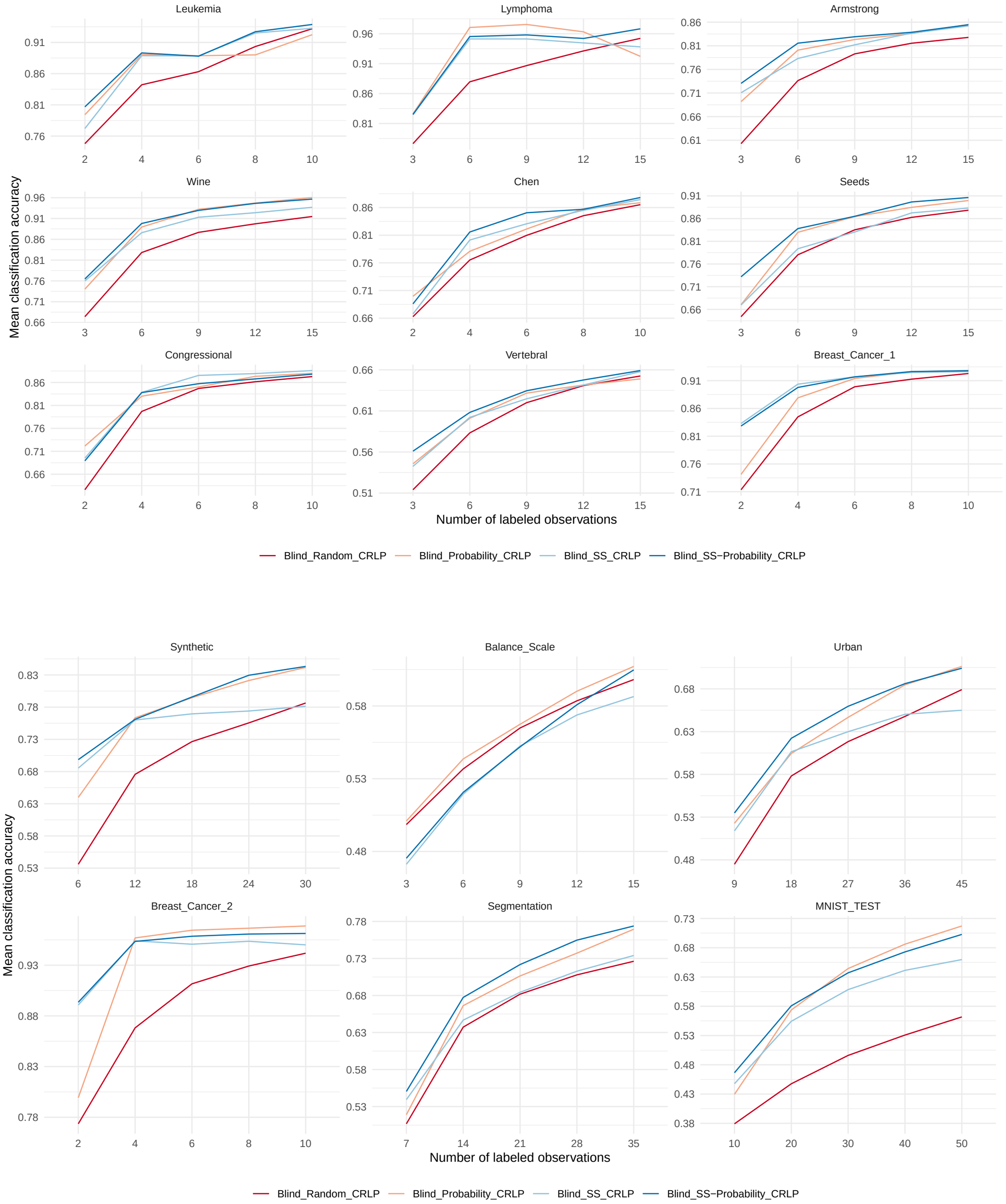


Figure 6.7: Classification mean accuracy curves for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms without replacement in 15 datasets.

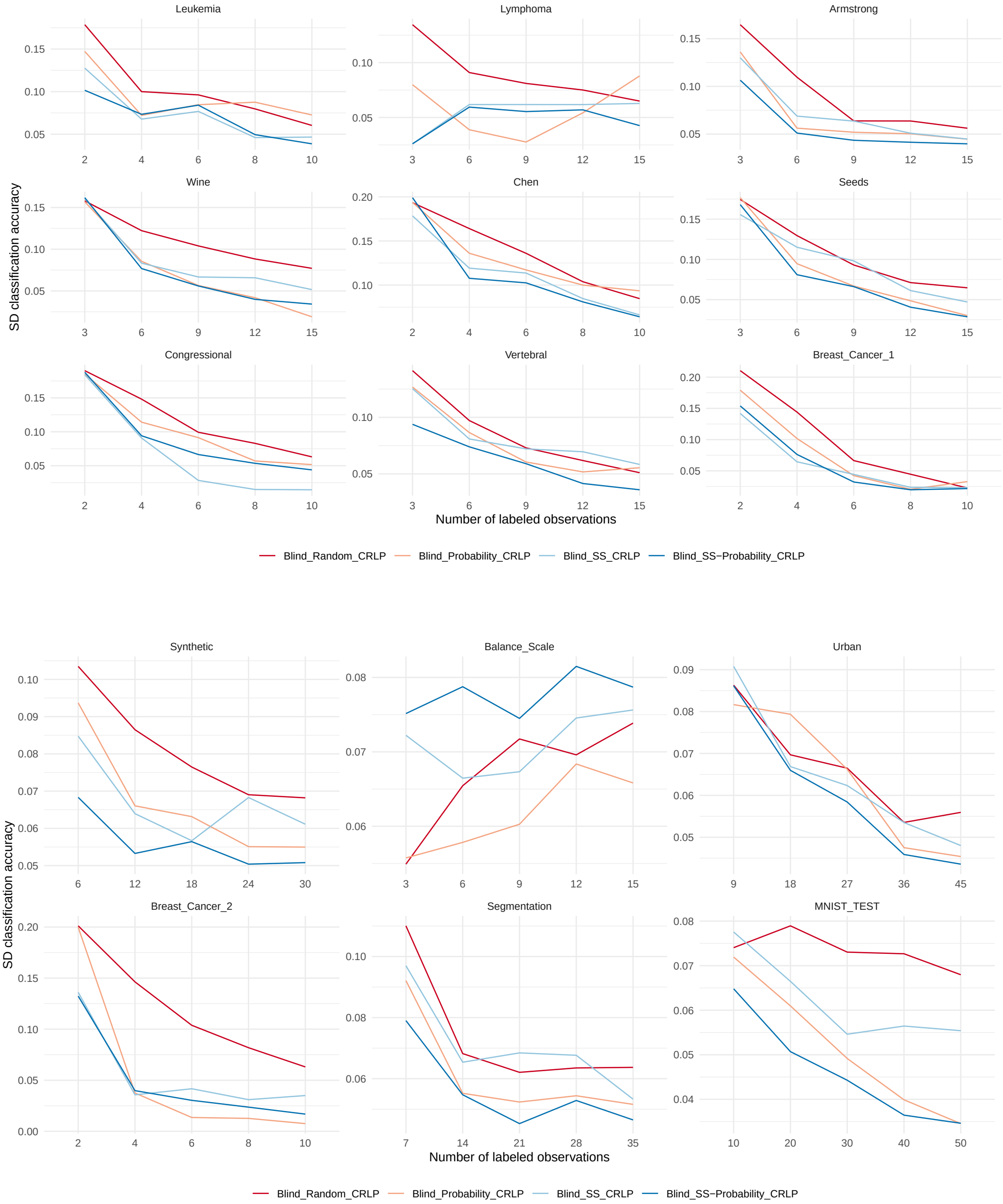


Figure 6.8: Classification standard deviation accuracy curves for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms without replacement in 15 datasets.

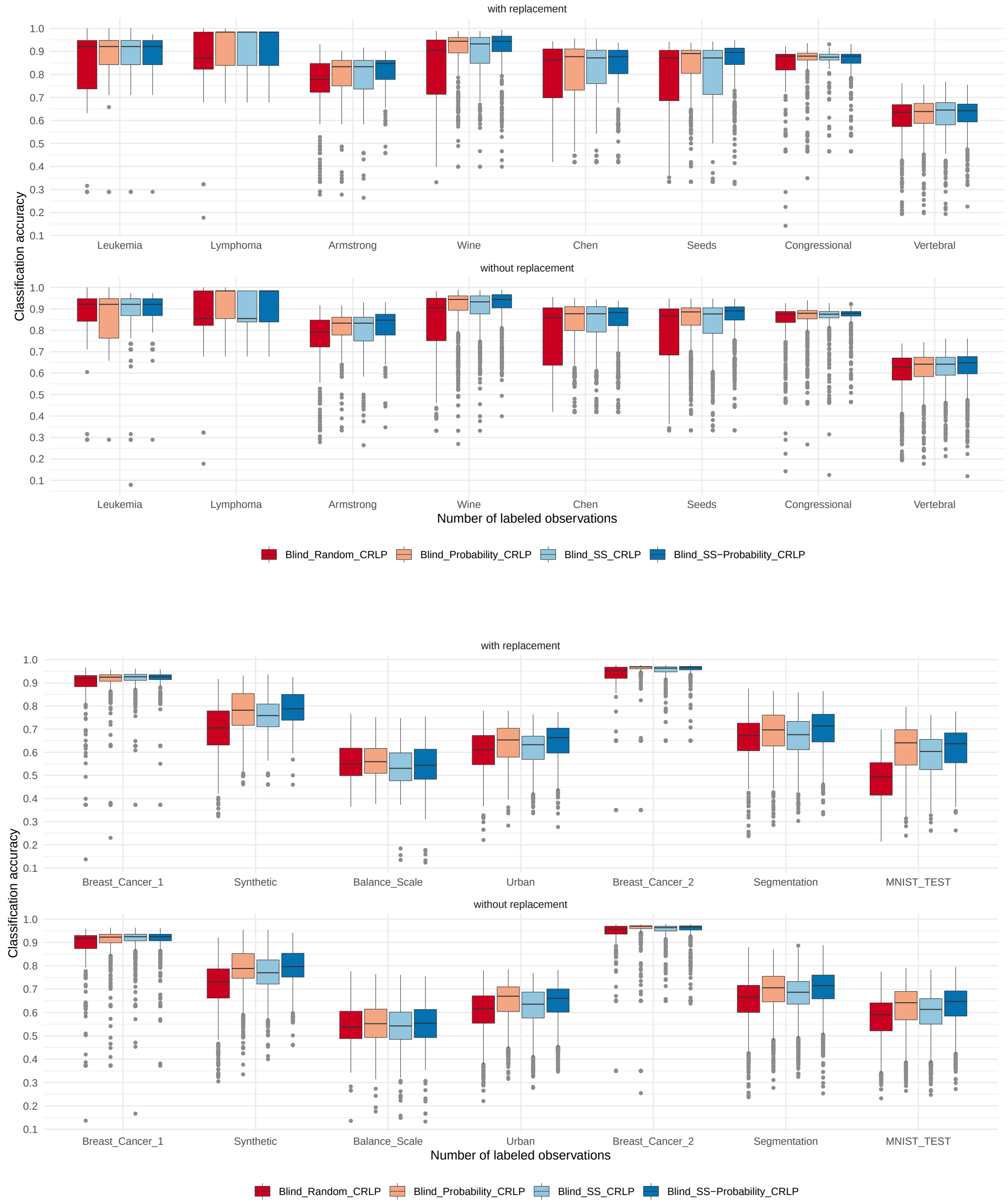


Figure 6.9: Classification accuracy curves of the all simulations for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms by dataset.

smaller dispersion. So if we execute the algorithm only once, we will have less risk of selecting non-representative data points. In practice, when we have higher accuracy and a smaller dispersion, we are reducing the risk of the selection process. Therefore, our strategies make the data point selection for the labeling process more robust.

6.5 Active Learning experiment

In our third experiment, the idea is to apply the selection strategies in the area of Active Learning. The previous LP without replacement experiment consists of evaluating the proposed selection strategies and comparing them with Uniform Random selection, which is currently the traditional way. So this can be seen as an AL process.

6.5.1 Active Learning process with Uncertainty

Figures 6.10, 6.11 and Appendix A.4 show the obtained results. The Query System Random Sampling does not use uncertainty. Therefore, we leave the selection random for comparison purposes only. According to the observed results, the behavior of the strategies is maintained in most of the datasets. When adding uncertainty behavior to the traditional random selection, we observed that the performance worsened in most of the datasets. In other words, we had low average accuracy and high standard deviation, worse than the Uniform Random selection.

The SS-Probability strategy affects both the LP and AL processes. Therefore, an important question is: did it bring any gain to add uncertainty in the selection in terms of performance? Figures 6.12 and 6.13 show a comparison between the experiment of label propagation without replacement (Random and SS-Probability) and active learning process with uncertainty (Random_with_Uncertainty and SS-Probability_with_Uncertainty). In most datasets, there seems to be no positive impact of adding data point selection based on uncertainty. Not even in the traditional random method.

6.5.2 Active Learning process with CM update

Figures 6.14, 6.15 and Appendix A.5 show the obtained results. Given the observed results, the differences between the strategies remain similar to the previous experiment. Updating the CM intuitively seems to be the right decision. Only it did not bring benefits.

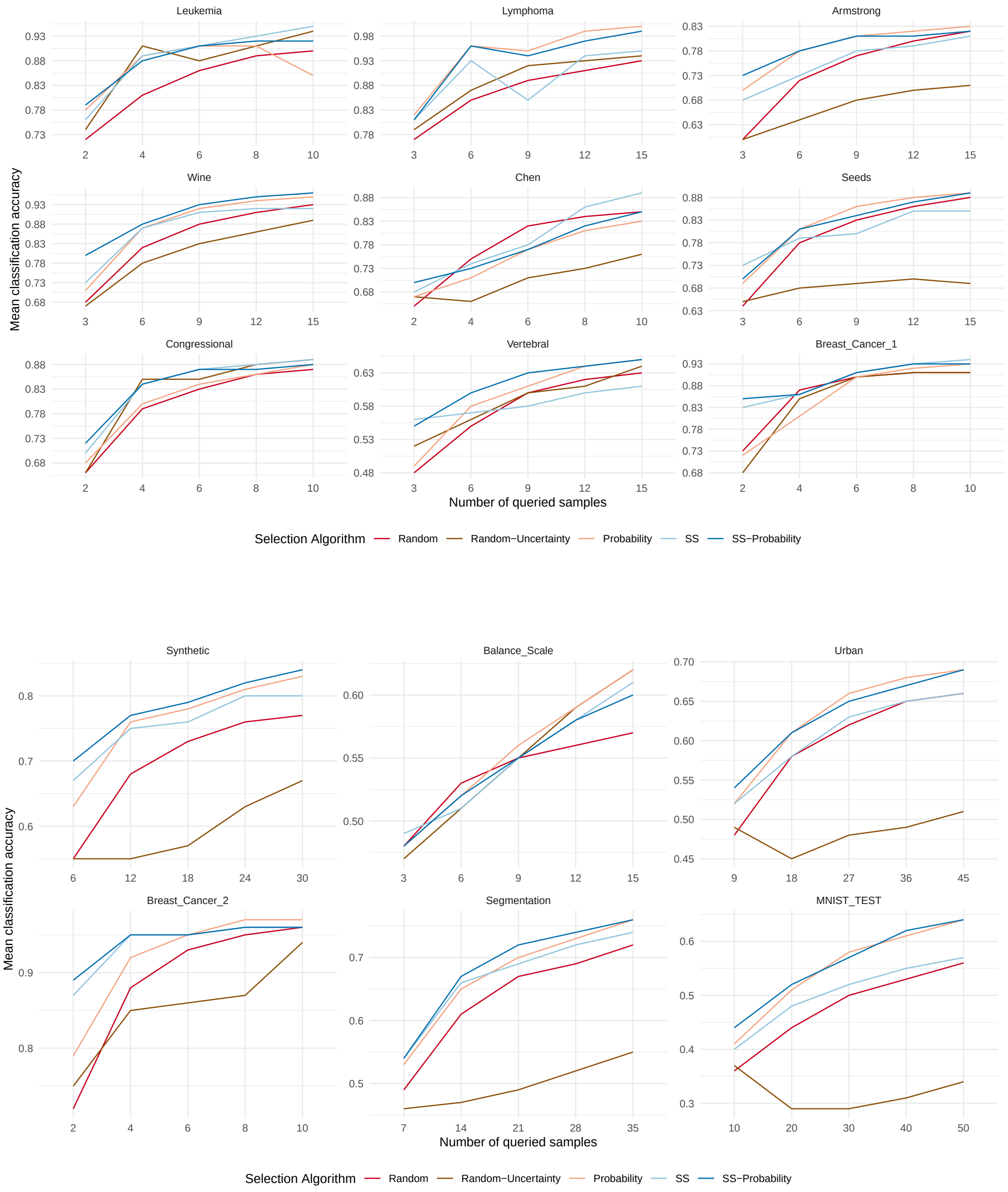


Figure 6.10: Classification mean accuracy curves for the Random, Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with uncertainty in 15 datasets.

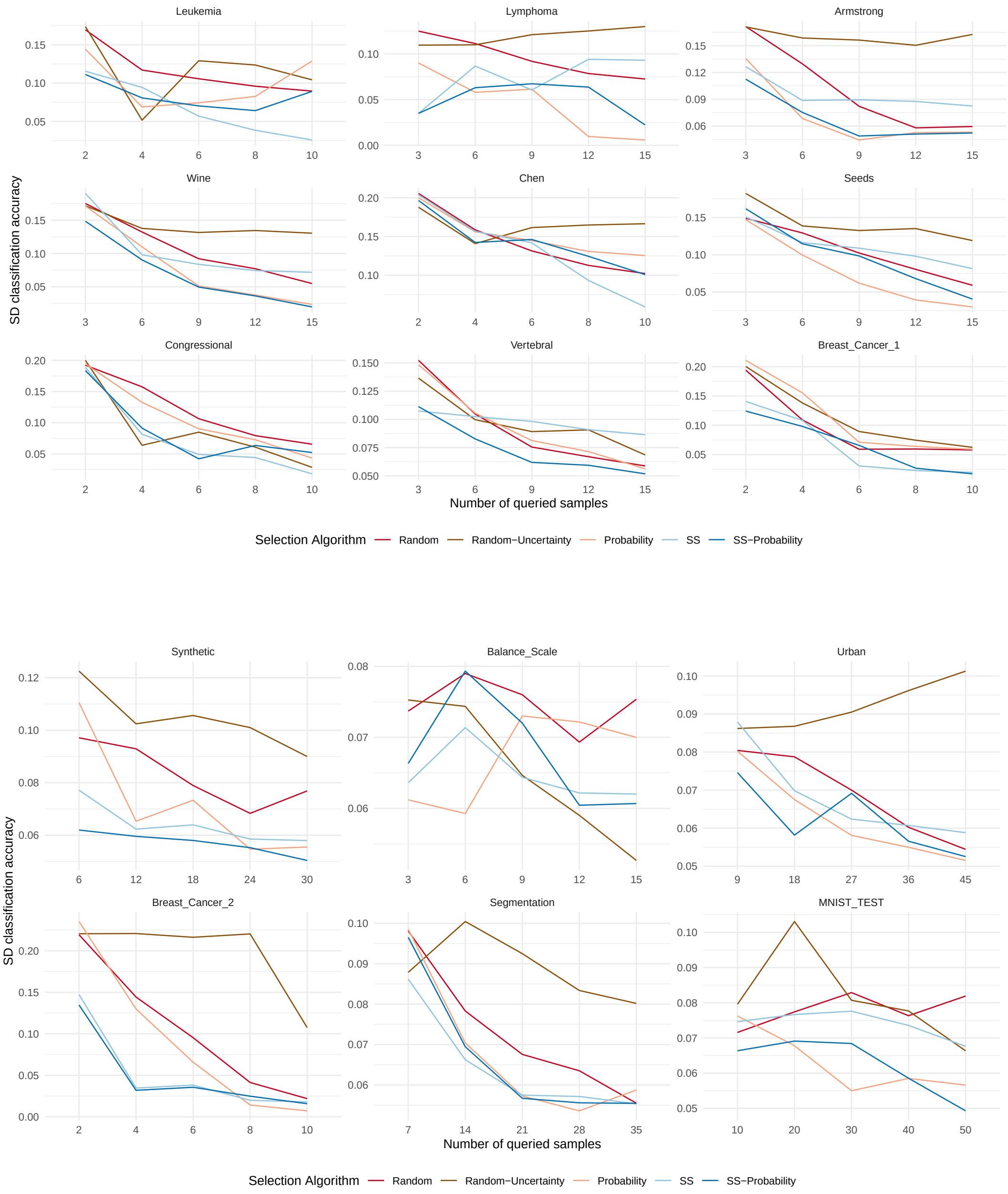


Figure 6.11: Classification standard deviation accuracy curves for the Random, Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with uncertainty in 15 datasets.

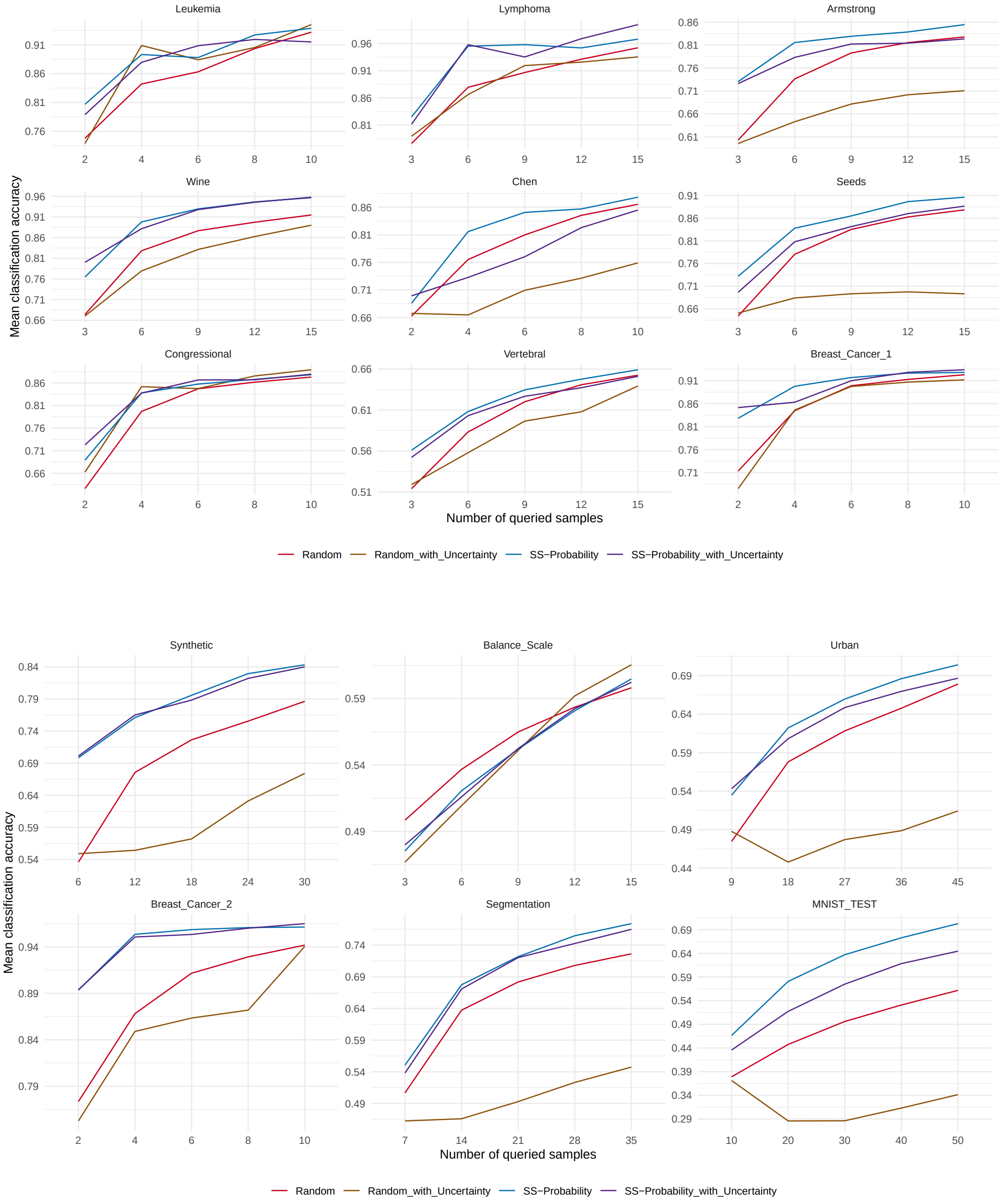


Figure 6.12: Classification mean accuracy curves for the Random, Random with Uncertainty, Stratified Sample Probability and Stratified Sample Probability with Uncertainty algorithms for the label propagation without replacement and active learning process with uncertainty in 15 datasets.

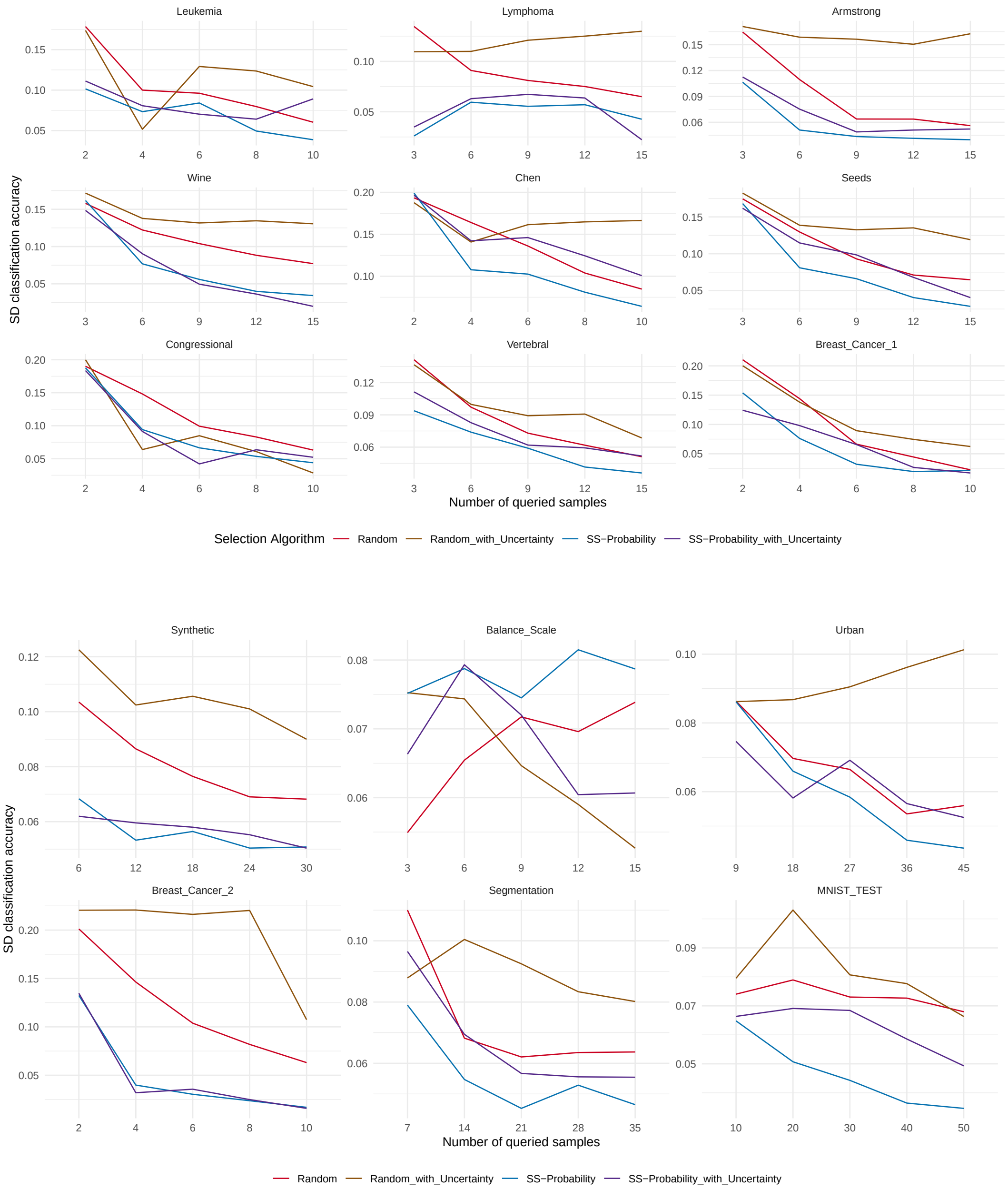


Figure 6.13: Classification standard deviation accuracy curves for the Random, Random with Uncertainty, Stratified Sample Probability and Stratified Sample Probability with Uncertainty algorithms for the label propagation without replacement and active learning process with uncertainty in 15 datasets.

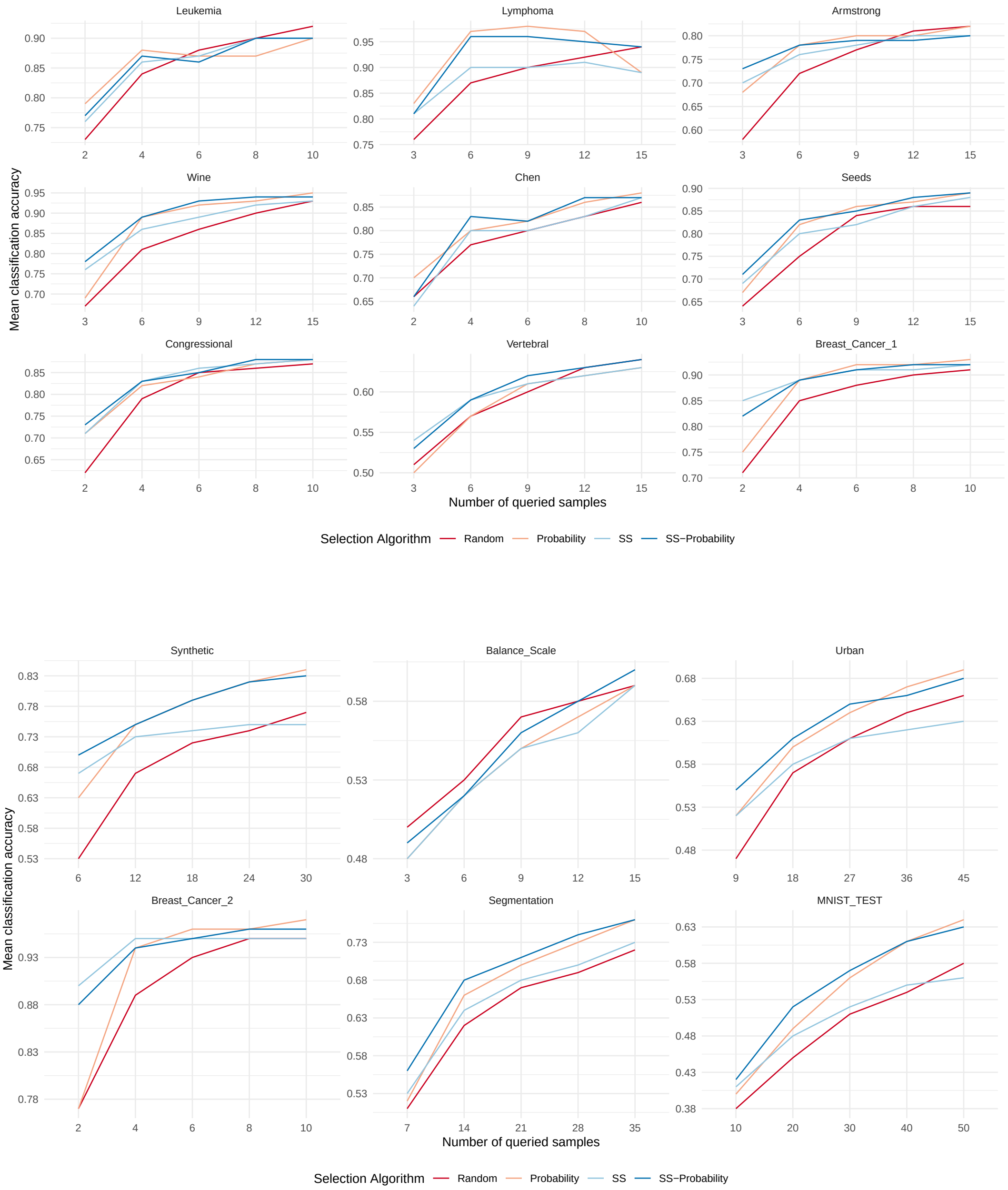


Figure 6.14: Classification mean accuracy curves for the Random, Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with CM update in 15 datasets.

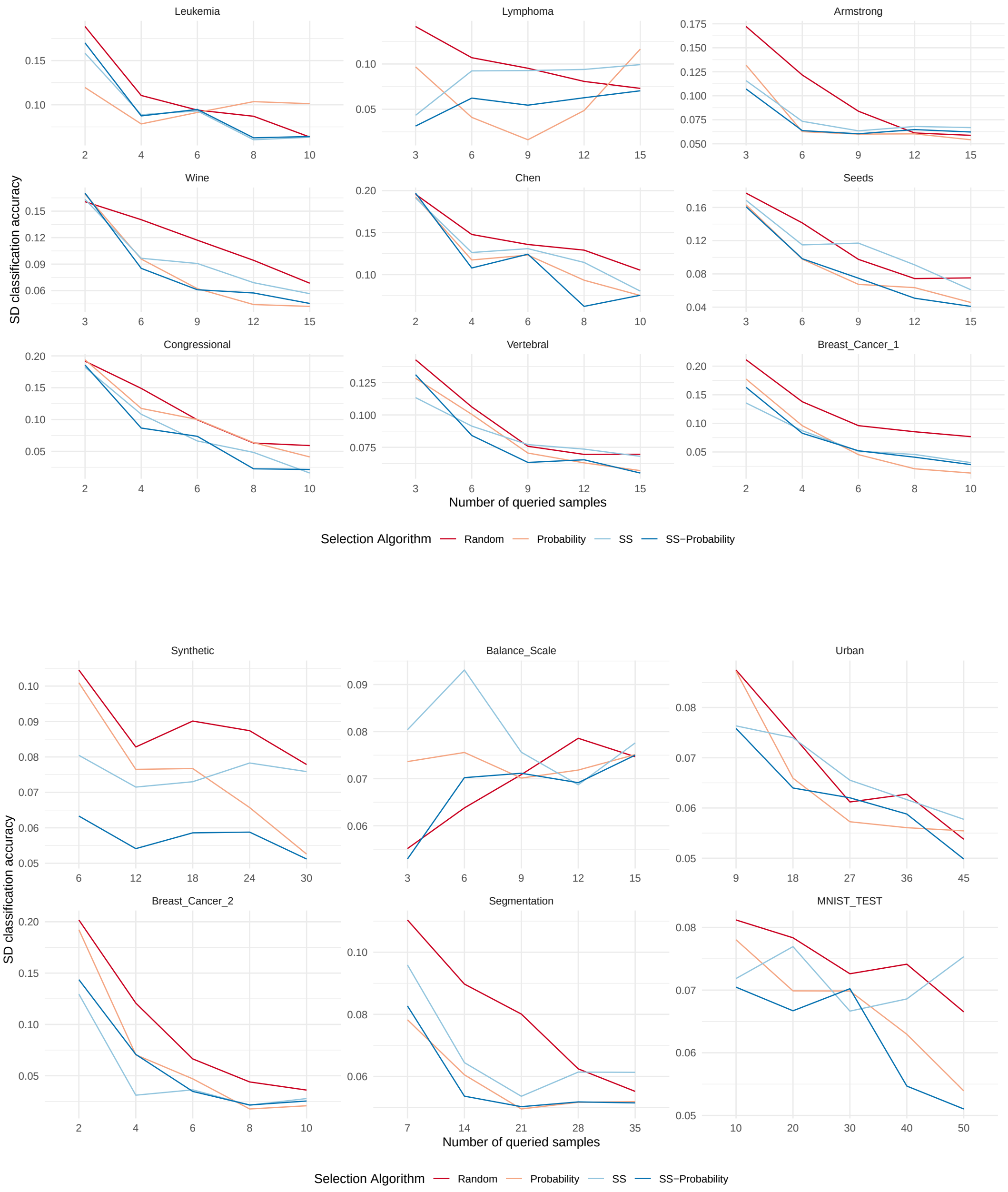


Figure 6.15: Classification standard deviation accuracy curves for the Random, Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with CM update in 15 datasets.

Could it be that updating the CM brings any improvement in the SS-Probability strategy in terms of performance? Figures 6.16 and 6.17 show a comparison between the label propagation experiment without replacement (Random and SS-Probability) and active learning process with CM update (Random_with_CM_update and SS-Probability_with_CM_update). In most of the datasets both SS-Probability with CM update strategy was similar to the SS-Probability strategy. So everything indicates that it had a small impact on mean accuracy. Therefore, we would not advise this variant since it is necessary to recalculate the importance vector each time.

6.5.3 Active Learning process with Uncertainty and CM update

Figures 6.18, 6.19 and Appendix A.6 show the obtained results. As in the previous results, the strategies maintain the performance in most of the datasets.

Figures 6.20 and 6.21 show a comparison between two experiments: label propagation without replacement and active learning process with CM update. In most of the datasets, there seems to be no positive impact of updating the CM with uncertainties, not even in the traditional Random strategy.

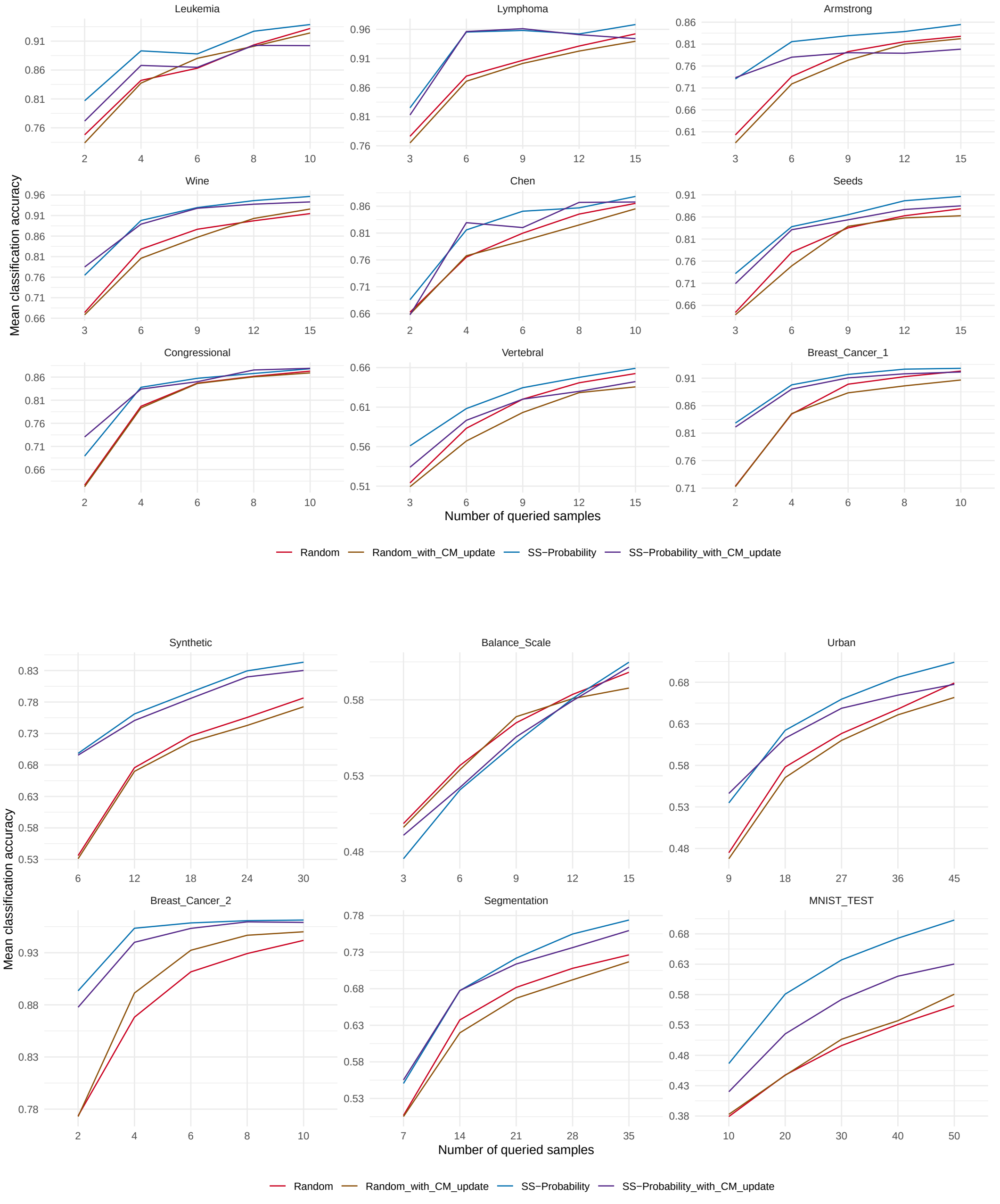


Figure 6.16: Classification mean accuracy curves for the Random, Random with CM update, Stratified Sample Probability and Stratified Sample Probability with CM update algorithms for the label propagation without replacement and active learning process with CM update in 15 datasets.

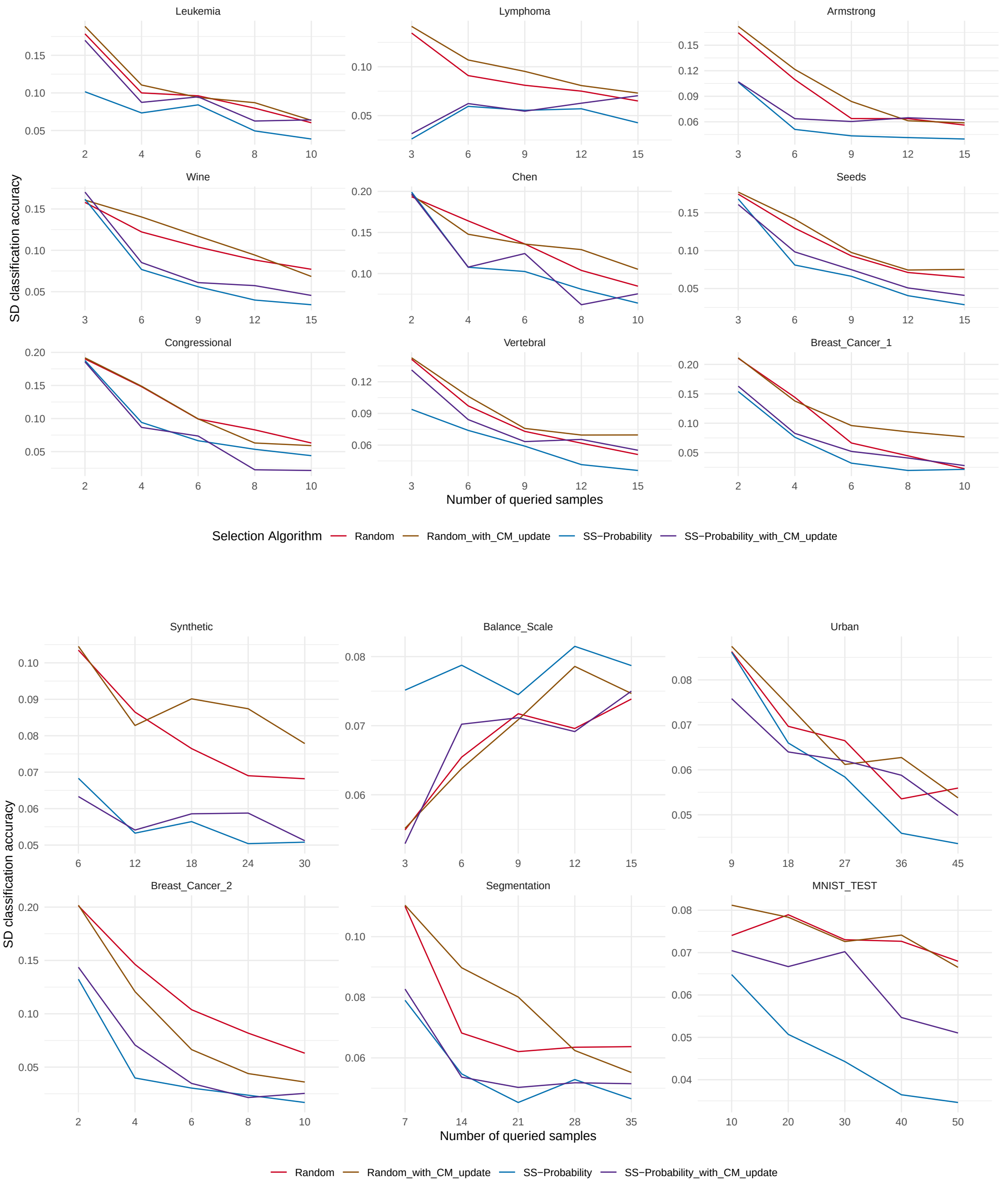


Figure 6.17: Classification standard deviation accuracy curves for the Random, Random with CM update, Stratified Sample Probability and Stratified Sample Probability with CM update algorithms for the label propagation without replacement and active learning process with CM update in 15 datasets.

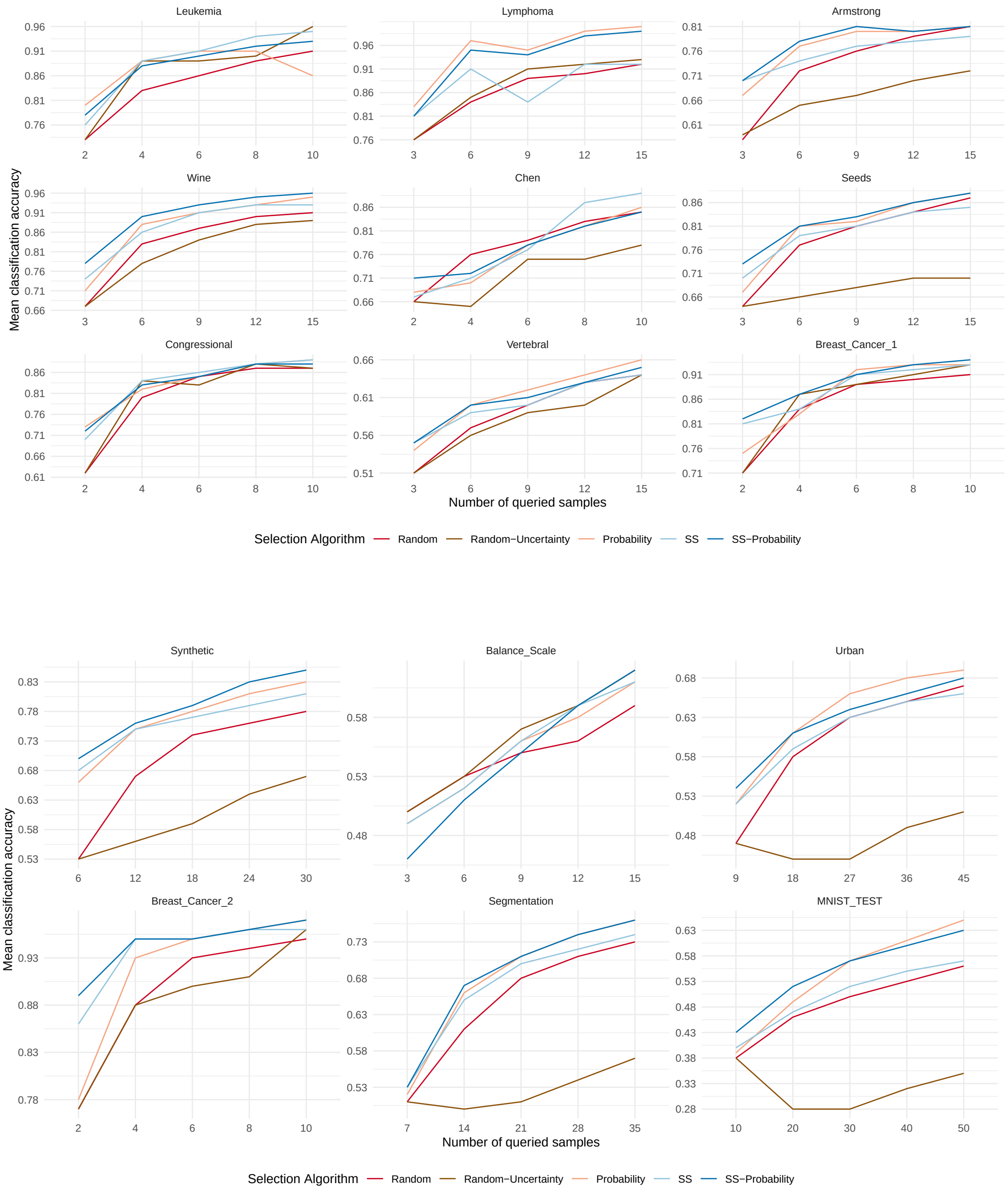


Figure 6.18: Classification mean accuracy curves for the Random, Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with uncertainty and CM update in 15 datasets.

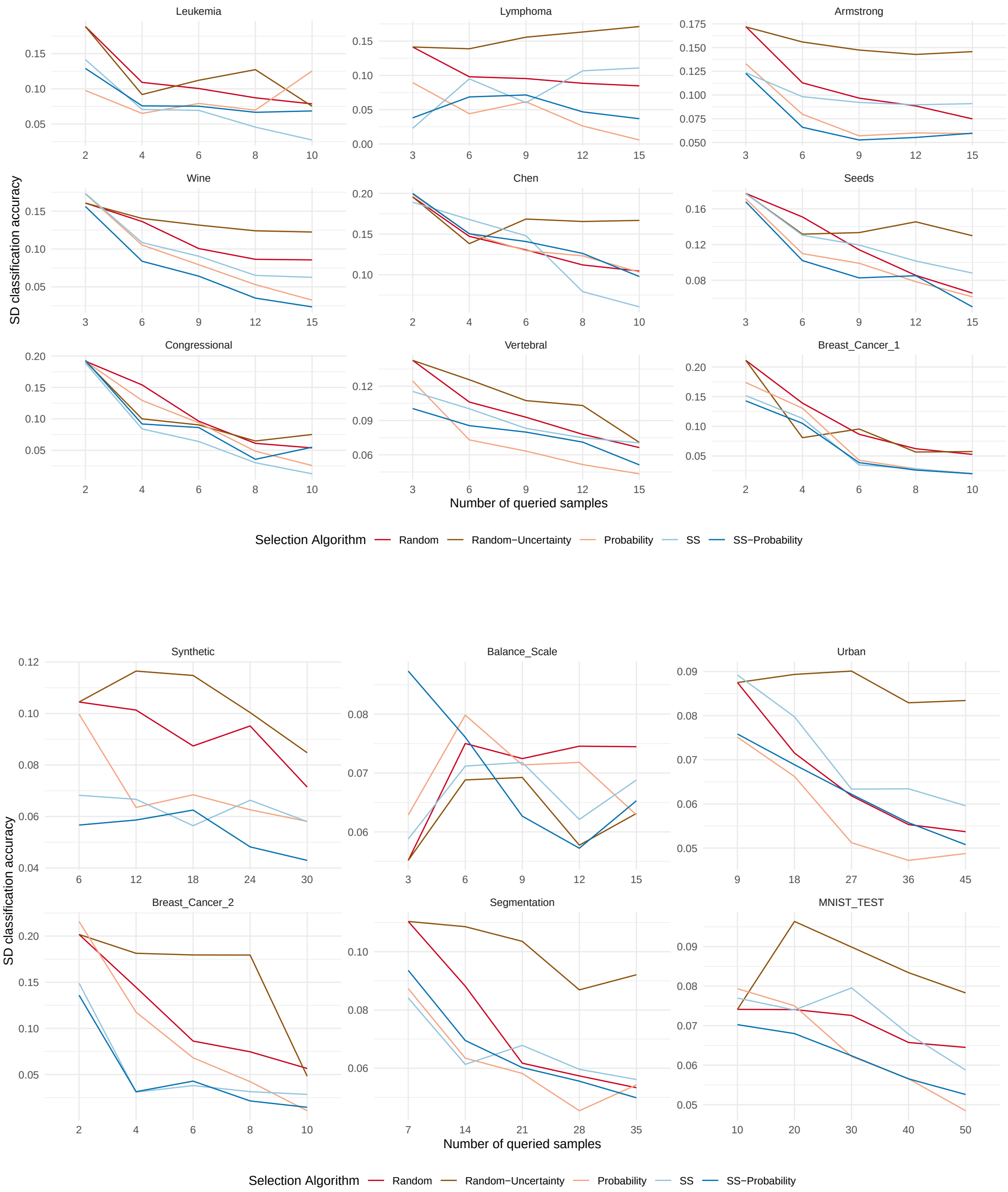


Figure 6.19: Classification standard deviation accuracy curves for the Random, Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with uncertainty and CM update in 15 datasets.

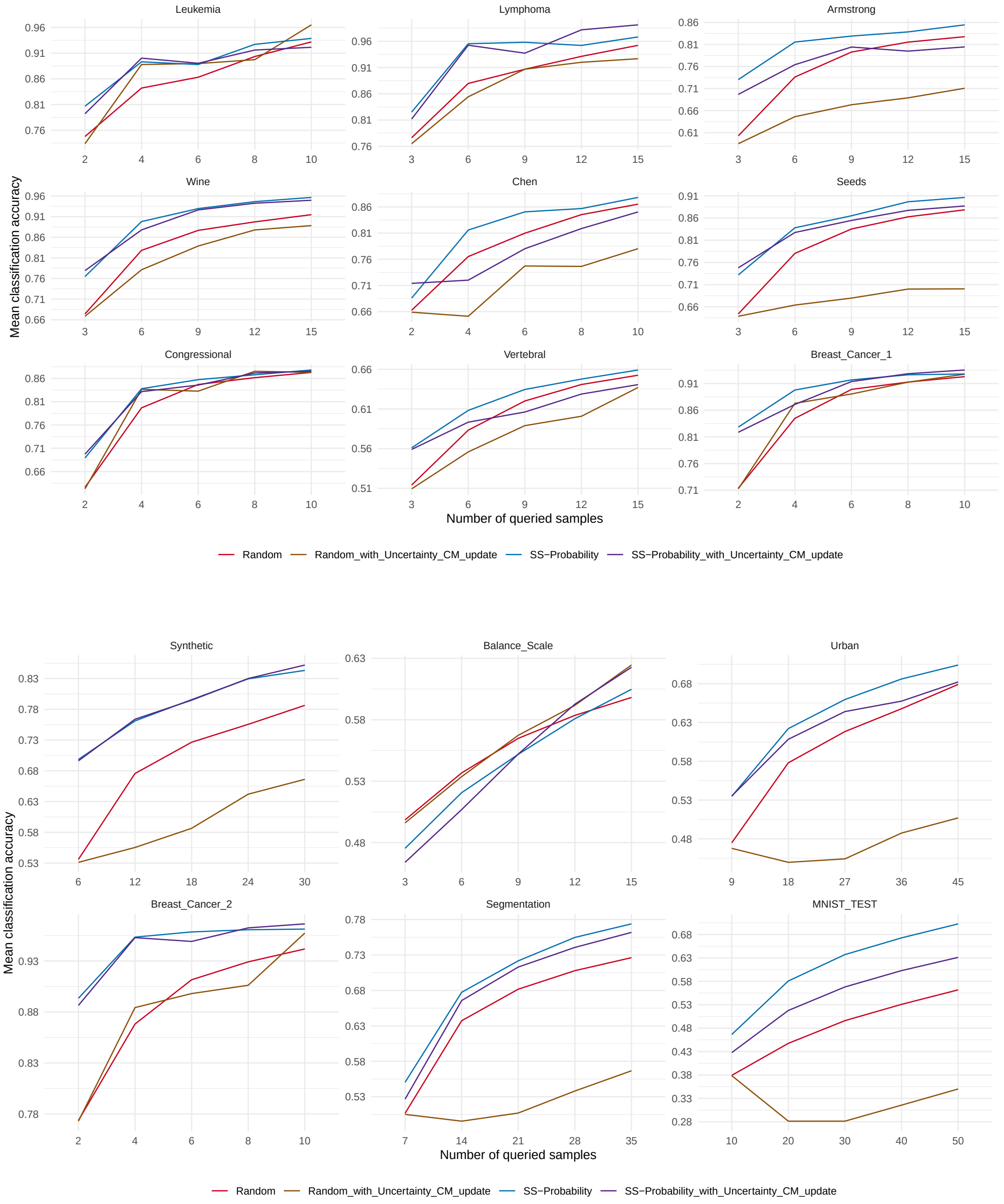


Figure 6.20: Classification mean accuracy curves for the Random, Random with Uncertainty and CM update, Stratified Sample Probability and Stratified Sample Probability with Uncertainty and CM update algorithms for the label propagation without replacement and active learning process with uncertainty and CM update in 15 datasets.

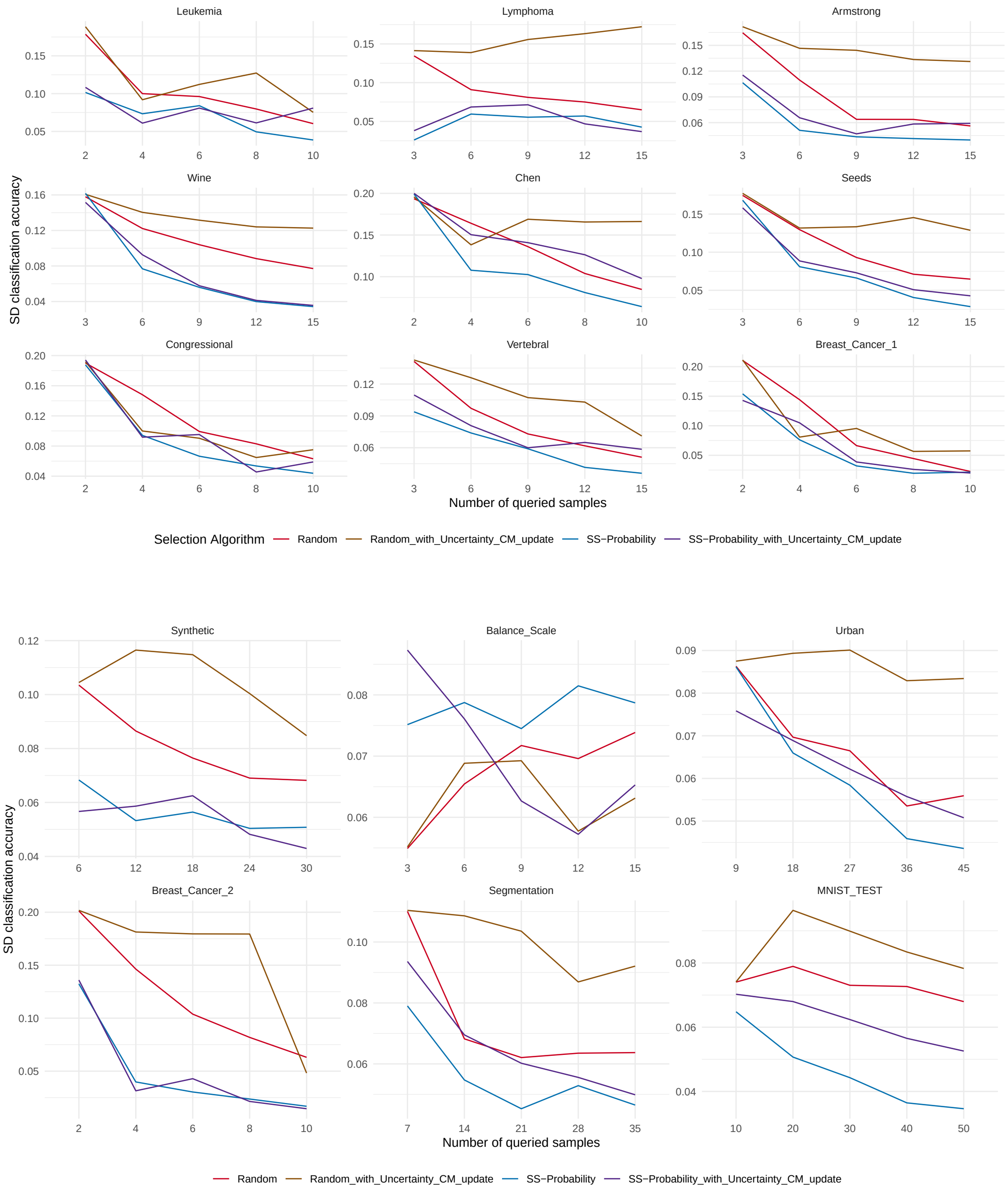


Figure 6.21: Classification standard deviation accuracy curves for the Random, Random with Uncertainty and CM update, Stratified Sample Probability and Stratified Sample Probability with Uncertainty and CM update algorithms for the label propagation without replacement and active learning process with uncertainty and CM update in 15 datasets.

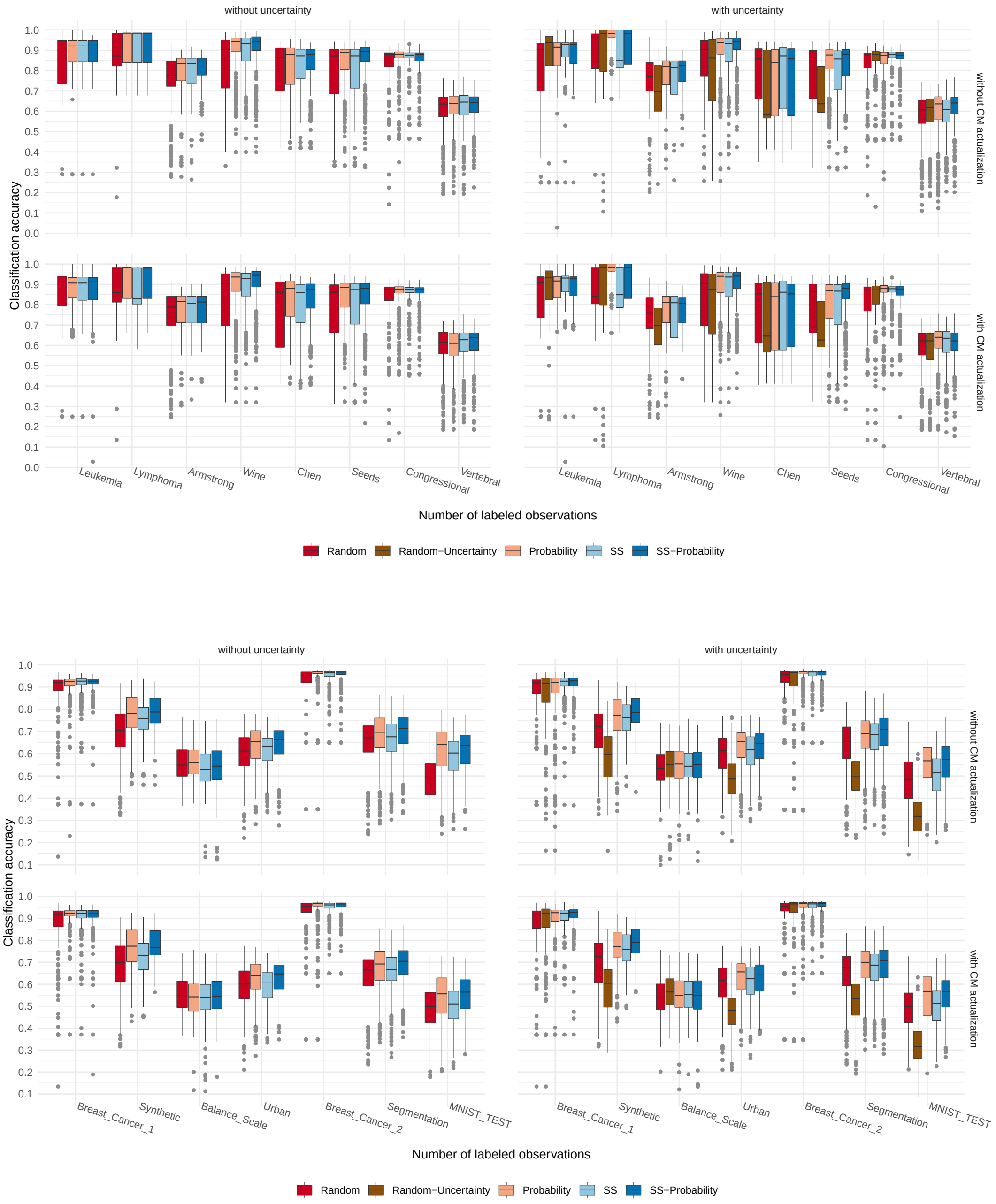


Figure 6.22: Classification accuracy curves of the 100 simulations for the Random, Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process by dataset.

6.5.4 Summary of Active Learning experiment

Figure 6.22 presents a summary of the third experiment performed and Label Propagation without replacement. The combination without uncertainty and without CM actualization is the LP without replacement. This experiment has three stages. The first stage is the combination with uncertainty and without

CM update. The second stage is the combination without uncertainty and with CM update, and the third stage is the combination with uncertainty and CM update.

For each dataset, all the simulations carried out are shown without distinguishing between the number of representatives. In general, the SS-Probability strategy obtained a mean accuracy greater than or equal to that obtained by Random and Random Uncertainty but with less dispersion.

6.6 Discussion

The experiments carried out showed that the use of the selection strategies applied in the context of a real problem is promising. In our experimental condition, especially in the AL, we cannot use a predictive method because we do not have enough observations. Since we work with a few observations, we use LP methods with the minimum number of labeled data points.

The most uncertain data point in the propagation is not the best choice. This makes sense because those data points with the most significant uncertainty are in different groups' border regions. A solution would be not to propagate with these data points because they cause noise and only add them to the set of labeled data points. Alternatively, perhaps, have a parameter to regulate the data point label's propagation speed for its neighbors. In this case, it would be a parameter for each data point.

The proposed selection strategy SS-Probability improves in both cases (AL uncertainty and AL with CM update).

The proposed heuristics are used to select the initial set of data points for the LP algorithms. If we use these AL strategies, they provide a better result than just using the Uncertainty-based Query System. At least when we use LP algorithms to perform the classification. As we observe in the results of experiment three, propagating the label of uncertain data points brings more significant confusion, which is reflected in low average accuracy. In summary, we can think that the proposed strategies can be applied in other contexts.

7
Considerations about selection strategies

During the research, questions emerged about the use of the co-association matrix (Section 7.1) due to its computational cost. Furthermore, we made a comparison between the data points selected by SSP and the data points of an optimal solution to discover if SSP achieves an accuracy comparable to the optimal solution (Section 7.2).

7.1
Similarity Matrix vs Co-association Matrix

A key question in our research is what kind of matrix of relationships between data points we can use? Our proposed selection strategies are based on obtaining the CM. We know that it is a computationally expensive, and time-consuming process. So it is entirely feasible to ask why not use a cheaper matrix such as a Similarity Matrix (SM)?

One way to build the SM would be to calculate the distance between every pair of data points ($dist_{i,j}$). We normalize the values between 0 and 1 (we divide each value by the maximum value of the distance matrix). Depending on the type of variables, we use Euclidean distance for continuous variables and Hamming distance for categorical variables. Then, the SM is defined as in the equation 7-1.

$$SM_{i,j} = 1 - dist_{i,j} \tag{7-1}$$

where (i, j) are two data points in a dataset.

The SM showed in Figure 1.1 was built using the above procedure. Thinking about how our selection strategies work, the importance vector will not be discriminating when we use the SM. Therefore, we will not have well-defined regions of importance, which will affect the selection. Figure 7.1 shows the behavior when applied to an LP process. Once again, 100 simulations of the LP were carried out without replacement for each number of representatives. We use the SM both for selection strategies and for propagation. It is not easy to select the best strategy based on the average accuracy value because the curves are very mixed and with peaks. This is not the case when we use the CM (Figures 6.7 and 6.8).

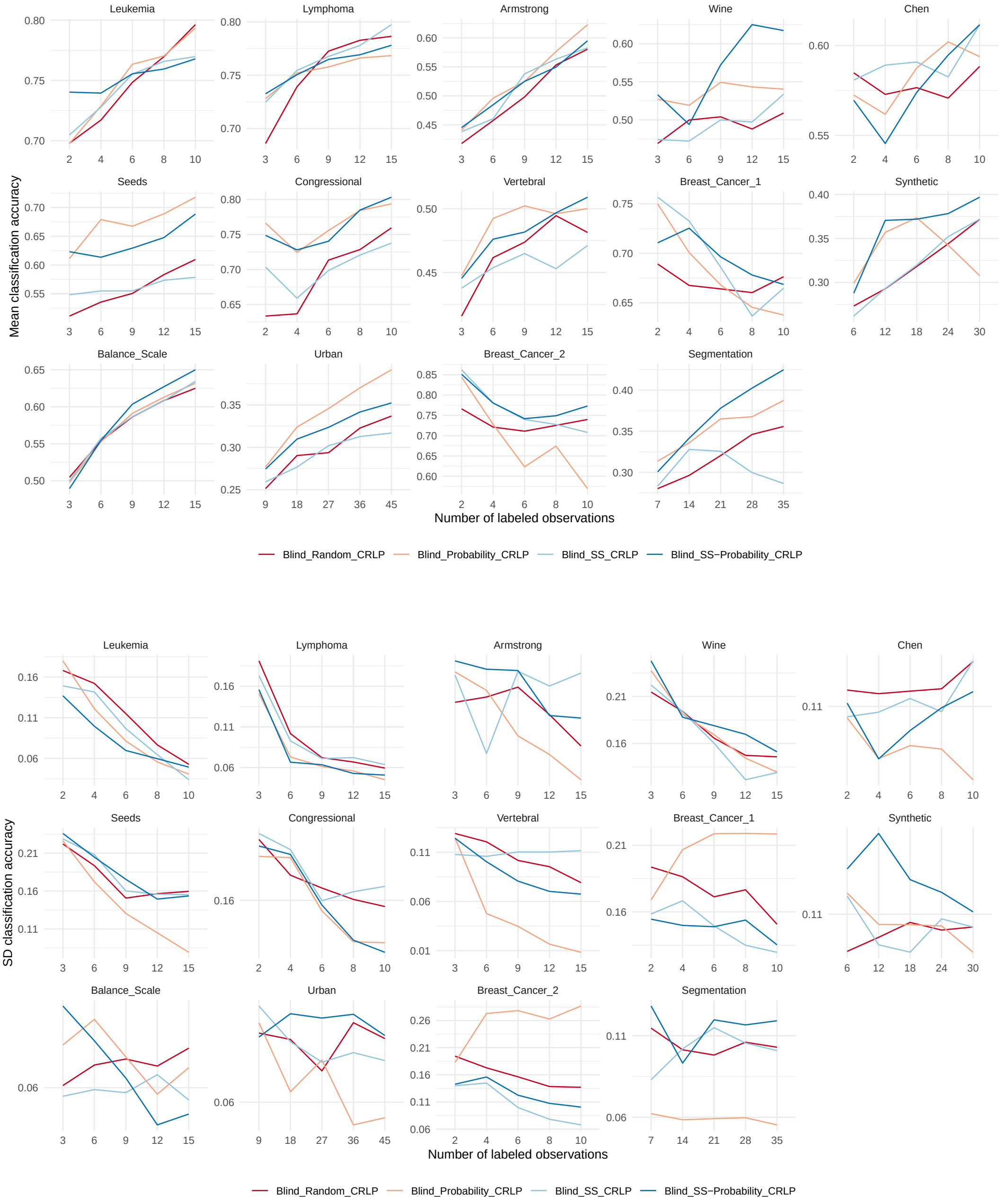


Figure 7.1: Classification mean and standard deviation accuracy curves for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms without replacement using the SM in 14 datasets.

For this reason, building the CM indeed requires computational effort and memory space, but the performance is much better, which reinforces Yu and Kim speech on the use of CM in (Yu and Kim, 2018).

7.2 Optimal solution analysis

The main idea of this analysis is to compare the solution proposed by the SS-Probability strategy and Uniform Random strategy with one optimal solution for the same dataset. In this context, we understand that the data points selection set that once labeled, will be used by the CRLP algorithm. By the nature of LP algorithms, the propagation will have better results if there is a balance between the initial data point set classes. Based on this premise, the optimal solutions are made up of sets of data points with balanced classes.

The algorithm implemented to find the optimal solution chooses combinations of data points from a list containing all the data points. For each combination, we propagate the label using step two of the CRLP algorithm. For example, suppose we have a dataset with three classes. We want to select nb.class, 2 * nb.class, 3 * nb.class, 4 * nb.class and 5 * nb.class data points without replacement.

In the first iteration of the algorithm, we obtain all the combinations of three data points, being a data point of each class to maintain the premise described above. Then, for each combination, we carry out a label propagation process. Of all the combinations of three data points, we save the one with the highest accuracy value. In the second iteration, we will select six data points but keeping the best solution with three data points obtained in the first iteration. We build combinations with only three data points without considering the data points of the best solution so far. Then we form the combinations of six data points by joining the combinations of three data points plus the data points of the previous best solution. Next, for each combination of six data points, we carry out a label propagation process, and the optimal six data point solution is the one with the highest accuracy value. In the third iteration, we obtain all the possible combinations of three data points again without counting the six data points that were already selected. Then we add to these combinations the optimal solution of six data points and find the best solution. Successively carry out the same procedure for twelve and fifteen data points.

Finally, when this search process is over, we have one optimal solution for 3,6,9,12, and 15 data points. Note that the optimal solution for 15 data points contains the optimal solution for 12 data points, which contains the optimal solution for nine data points, and so on until we reach the optimal solution of three data points.

With the algorithm described above, we generate a possible optimal solution. We select the first subset of data points that generates maximum accuracy. However, note that there can be multiple subsets of data points that generate the same accuracy value.

Generating all combinations of nb.class data points depending on the number of data points is computationally expensive. For this reason, we selected the Leukemia, Lymphoma, and Armstrong datasets as they have few data points.

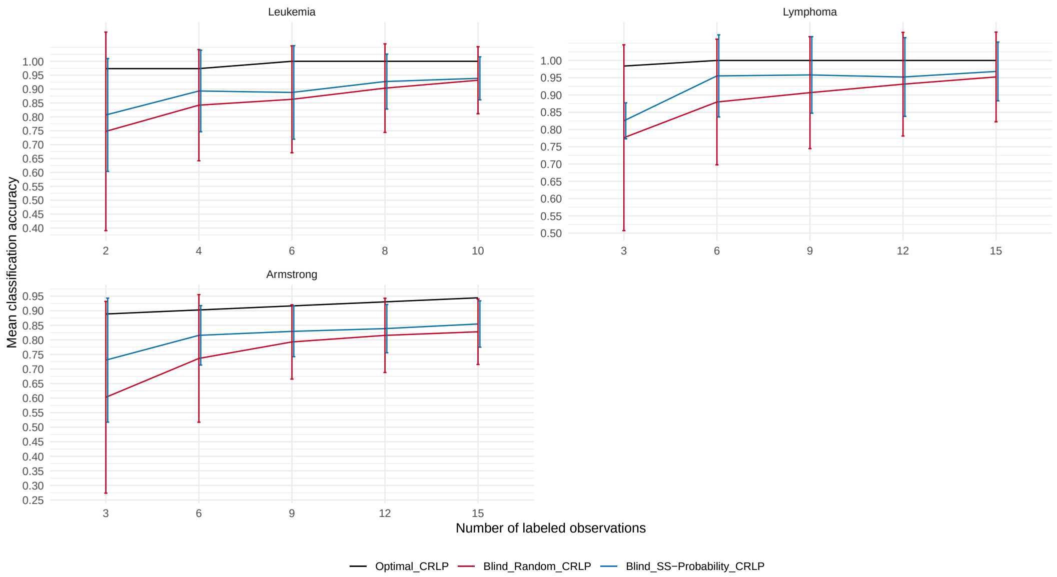


Figure 7.2: Classification mean accuracy curves for the Optimal, Random and Stratified Sample Probability algorithms without replacement when applied to Leukemia, Lymphoma and Armstrong datasets.

Figure 7.2 shows the Uniform Random and SS-Probability strategies' mean accuracy. In addition to the optimal solution represented in black. In general, we observe that SSP obtains a higher mean accuracy than that obtained by Uniform Random. In this sense, the mean accuracy of the solutions obtained by the SSP is closer to the optimal solution. Table 7.1 shows the mean accuracy and standard deviation by the number of labeled data points for the best solution found, SSP and Uniform Random strategies in the Leukemia, Lymphoma and Armstrong datasets. The columns %I_SSP and %I_Random represent the average frequency of the intersection between the solutions obtained (SSP and Random Uniform strategies) and the optimal solution.

For example, in the 100 simulations, we have on average 7.5% intersection between the data points selected through the Uniform Random selection strategy and a possible best solution obtained by selecting 2 data points in the Leukemia dataset. In Leukemia and Armstrong datasets, the SSP obtained the highest percentage of the intersection with the values 32.1% and 21.7%, respectively. In the case of Lymphoma, the Uniform Random obtained the highest percentage of the intersection with a value of 24.6%. However, despite having a higher intersection percentage, the SSP obtained a higher accuracy value (96.8%). One hypothesis is that the SSP selects better-distributed data points in the domain. Another possibility is that there are other optimal solutions.

By construction, the optimal solution guarantees to select data points while maintaining the balance of the dataset classes, while our strategies do not guarantee to find a representative of each class. For example, for the Leukemia dataset, in none of the simulations performed (SSP and the Uniform Random strategies) we obtained a set of data points with balanced classes. In the case of Lymphoma with three representatives and Uniform Random strategy, in 9% of the simulations the selected data points were balanced. For the rest of the representatives, it was unbalanced. In the case of the Armstrong dataset, we observe a greater balance between the classes. For example, with three representatives, the Uniform Random strategy obtained a balance of 18% while SSP obtained 47%.

Table 7.1: Mean accuracy and Standard Deviation by number of labeled data points for best solution found, SSP and Uniform Random strategies.

Dataset	nb_repr	Optimal	SSP	Random	%I_SSP	%I_Random
Leukemia	2	0.974 (±0)	0.807 (±0.102)	0.748 (±0.179)	8.0	7.5
	4	0.974 (±0)	0.893 (±0.073)	0.842 (±0.1)	13.5	12.2
	6	1.000 (±0)	0.888 (±0.084)	0.863 (±0.096)	16.5	17.0
	8	1.000 (±0)	0.927 (±0.049)	0.903 (±0.08)	24.6	22.0
	10	1.000 (±0)	0.939 (±0.039)	0.932 (±0.06)	32.1	26.0
Lymphoma	3	0.984 (±0)	0.825 (±0.026)	0.776 (±0.135)	4.3	6.3
	6	1.000 (±0)	0.955 (±0.059)	0.880 (±0.091)	5.2	10.2
	9	1.000 (±0)	0.958 (±0.055)	0.907 (±0.081)	9.0	14.2
	12	1.000 (±0)	0.952 (±0.057)	0.931 (±0.075)	14.9	19.7
	15	1.000 (±0)	0.968 (±0.043)	0.952 (±0.065)	21.1	24.6
Armstrong	3	0.889 (±0)	0.73 (±0.106)	0.603 (±0.165)	3.7	5.3
	6	0.903 (±0)	0.816 (±0.051)	0.736 (±0.11)	8.3	9.3
	9	0.917 (±0)	0.829 (±0.043)	0.793 (±0.064)	9.8	13.2
	12	0.931 (±0)	0.839 (±0.041)	0.815 (±0.064)	17.3	16.6
	15	0.944 (±0)	0.855 (±0.04)	0.828 (±0.056)	21.7	21.2

8
Conclusion

This chapter presents the main contributions of this research and outlines directions for future work.

8.1
Contributions

This research introduced three strategies for data point selection based on Stratified and Non-uniform Sampling from a Probability mass function extracted from the Co-association Matrix. Our experiments in 15 datasets shows the effectiveness of the proposed selection methods in a semi-supervised context. The proposed selection strategies opens the door to be also used in Active Learning (AL) algorithms due to the data points selection step in the AL loop. Our three research questions were answered. Unlike Yu and Kim (Yu and Kim, 2018) who obtains the average results through simulations of 100 times each set of data points with label knowledge, we proposed blind strategies.

However, we still have issues to be addressed in the future. For example, taking into account the limitation of the SS strategy, we can still improve the calculation of the importance indicator. In addition, another future line could be to improve the control of the initial class balance, regardless of the distribution of the real classes in the dataset. As explained in the LP articles, classes must be balanced in order to obtain best results. This is not a simple task because our process is totally blind, we only trust on the quality of the co-association matrix.

The main limitation of our work is the previous construction of the CM. We know that it is a computationally expensive, time consuming process and depending on the size of the dataset it consumes a considerable portion of memory. However, in certain scenarios they can be addressed by methods such as the one presented in (Huang et al., 2019). We have observed that the information that is hidden in CM is important, and reinforces the Yu and Kim speech on the use of CM in (Yu and Kim, 2018).

8.2 Future works

For future research, we list some suggestions that may improve our work:

- Explore the use of selection strategies in large scale situations.
- Explore the use of CM in different contexts and not only AL and LP. The CM contains valuable information about data.
- Compare the proposed selection strategies within AL framework with other query system techniques. In this work we limited to uncertainty query system.

Bibliography

Aggarwal, C. C. (2015). *Data mining: the textbook*, chapter 11. Data Classification: Advanced concepts. Springer.

Armstrong, S. A., Staunton, J. E., Silverman, L. B., Pieters, R., den Boer, M. L., Minden, M. D., Sallan, S. E., Lander, E. S., Golub, T. R., and Korsmeyer, S. J. (2002). Mll translocations specify a distinct gene expression profile that distinguishes a unique leukemia. *Nat Genet*, 30(1):41–47.

Baram, Y., Yaniv, R. E., and Luz, K. (2004). Online choice of active learning algorithms. *Journal of Machine Learning Research*, 5(Mar):255–291.

Berikov, V., Karaev, N., and Tewari, A. (2017). Semi-supervised classification with cluster ensemble. In *2017 International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON)*, pages 245–250. IEEE.

Berikov, V. and Litvinenko, A. (2019). Semi-supervised regression using cluster ensemble and low-rank co-association matrix decomposition under uncertainties. *arXiv preprint arXiv:1901.03919*.

Boulesteix, A.-L., Durif, G., Lambert-Lacroix, S., Peyre, J., and Strimmer., K. (2018). *plsgenomics: PLS Analyses for Genomics*. R package version 1.5-2.

Chen, X., Cheung, S. T., So, S., Fan, S. T., Barry, C., Higgins, J., Lai, K.-M., Ji, J., Dudoit, S., Ng, I. O., van de Rijn, M., Botstein, D., and Brown, P. O. (2002). Gene expression patterns in human liver cancers. *Mol. Biol. Cell*, 13(6):1929–1939.

Chollet, F. et al. (2015). Keras. `https://keras.io`.

Chung, D., Chun, H., and Keles, S. (2019). *spls: Sparse Partial Least Squares (SPLS) Regression and Classification*. R package version 2.2-3.

Das, A., Nair, M. S., and Peter, D. S. (2020). Batch mode active learning on the riemannian manifold for automated scoring of nuclear pleomorphism in breast cancer. *Artificial Intelligence in Medicine*, 103:101805.

De Sousa, C. A. (2015). An overview on the gaussian fields and harmonic functions method for semi-supervised learning. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Fern, X. Z. and Brodley, C. E. (2003). Random projection for high dimensional data clustering: A cluster ensemble approach. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, pages 186–193.

Fiol-Gonzalez, S., Almeida, C., Barbosa, S., and Lopes, H. (2018). A novel committee–based clustering method. In *International Conference on Big Data Analytics and Knowledge Discovery*, pages 126–136. Springer.

Fiol-González, S., Almeida, C. F., Rodrigues, A. M., Barbosa, S. D., and Lopes, H. (2019). Visual exploration tools for ensemble clustering analysis. In *VISIGRAPP (3: IVAPP)*, pages 259–266.

Forestier, G. and Wemmert, C. (2016). Semi-supervised learning using multiple clusterings with limited labeled data. *Information Sciences*, 361:48–65.

Fred, A. L. and Jain, A. K. (2005). Combining multiple clusterings using evidence accumulation. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (6):835–850.

He, N. and Huang, D. (2019). Meta-cluster based consensus clustering with local weighting and random walking. In *International Conference on Intelligent Science and Big Data Engineering*, pages 266–277. Springer.

Huang, D., Lai, J.-H., and Wang, C.-D. (2015). Combining multiple clusterings via crowd agreement estimation and multi-granularity link analysis. *Neurocomputing*, 170:240–250.

Huang, D., Wang, C.-D., and Lai, J.-H. (2018). Locally weighted ensemble clustering. *IEEE transactions on cybernetics*, 48(5):1460–1473.

Huang, D., Wang, C.-D., Wu, J., Lai, J.-H., and Kwoh, C. K. (2019). Ultra-scalable spectral clustering and ensemble clustering. *IEEE Transactions on Knowledge and Data Engineering*.

Iam-On, N., Boongoen, T., and Garrett, S. (2008). Refining pairwise similarity matrix for cluster ensemble problem with cluster relations. In *International Conference on Discovery Science*, pages 222–233. Springer.

Kim, A. and Cho, S.-B. (2019). An ensemble semi-supervised learning method for predicting defaults in social lending. *Engineering Applications of Artificial Intelligence*, 81:193–199.

Kumar, P. and Gupta, A. (2020). Active learning query strategies for classification, regression, and clustering: A survey. *Journal of Computer Science and Technology*, 35(4):913–945.

Lichman, M. et al. (2013). Uci machine learning repository.

Lin, J. (1991). Divergence measures based on the shannon entropy. *Information Theory, IEEE Transactions on*, 37(1):145–151.

Livieris, I. (2019). A new ensemble semi-supervised self-labeled algorithm. *Informatica*, 43(2):221–234.

Long, J., Yin, J., Zhao, W., and Zhu, E. (2008). Graph-based active learning based on label propagation. In *International Conference on Modeling Decisions for Artificial Intelligence*, pages 179–190. Springer.

Molnar, C. (2020). *Interpretable Machine Learning*. Lulu. com.

Ovelgönne, M. and Geyer-Schulz, A. (2012). An ensemble learning strategy for graph clustering. *Graph Partitioning and Graph Clustering*, 588:187.

Park, S. H. and Kim, S. B. (2019). Active semi-supervised learning with multiple complementary information. *Expert Systems with Applications*, 126:30–40.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Ramirez-Loaiza, M. E., Sharma, M., Kumar, G., and Bilgic, M. (2017). Active learning: an empirical study of common baselines. *Data mining and knowledge discovery*, 31(2):287–313.

Settles, B. (2009). Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences.

Shannon, C. E. and Weaver, W. (1963). *Mathematical theory of communication*. University Illinois Press.

Strehl, A. and Ghosh, J. (2002). Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of machine learning research*, 3(Dec):583–617.

Tao, Z., Liu, H., Li, S., and Fu, Y. (2016). Robust spectral ensemble clustering. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 367–376. ACM.

Tomanek, K. and Hahn, U. (2009). Semi-supervised active learning for sequence labeling. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1039–1047. Association for Computational Linguistics.

Topchy, A., Jain, A. K., and Punch, W. (2004). A mixture model for clustering ensembles. In *Proceedings of the 2004 SIAM international conference on data mining*, pages 379–390. SIAM.

Tur, G., Hakkani-Tür, D., and Schapire, R. E. (2005). Combining active and semi-supervised learning for spoken language understanding. *Speech Communication*, 45(2):171–186.

Van Rossum, G. and Drake, F. L. (2009). *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA.

Vega-Pons, S. and Ruiz-Shulcloper, J. (2011). A survey of clustering ensemble algorithms. *International Journal of Pattern Recognition and Artificial Intelligence*, 25(03):337–372.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Jarrod Millman, K., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C., Polat, İ., Feng, Y., Moore, E. W., Vand erPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and Contributors, S. . . (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*.

Wang, F. and Zhang, C. (2007). Label propagation through linear neighborhoods. *IEEE Transactions on Knowledge and Data Engineering*, 20(1):55–67.

Wang, X., Yang, C., and Zhou, J. (2009). Clustering aggregation by probability accumulation. *Pattern Recognition*, 42(5):668–675.

Xu, D. and Tian, Y. (2015). A comprehensive survey of clustering algorithms. *Annals of Data Science*, 2(2):165–193.

Yang, Y. and Loog, M. (2019). Single shot active learning using pseudo annotators. *Pattern Recognition*, 89:22–31.

Yi, J., Yang, T., Jin, R., Jain, A. K., and Mahdavi, M. (2012). Robust ensemble clustering by matrix completion. In *2012 IEEE 12th International Conference on Data Mining*, pages 1176–1181. IEEE.

Yin, L., Wang, H., Fan, W., Kou, L., Lin, T., and Xiao, Y. (2019). Incorporate active learning to semi-supervised industrial fault classification. *Journal of Process Control*, 78:88–97.

Yu, G., Feng, L., Yao, G., and Wang, J. (2016). Semi-supervised classification using multiple clusterings. *Pattern Recognition and Image Analysis*, 26(4):681–687.

Yu, J. and Kim, S. B. (2018). Consensus rate-based label propagation for semi-supervised classification. *Information Sciences*, 465:265–284.

Zhang, H., Zhang, Z., Zhao, M., Ye, Q., Zhang, M., and Wang, M. (2020). Robust triple-matrix-recovery-based auto-weighted label propagation for classification. *IEEE Transactions on Neural Networks and Learning Systems*.

Zhang, X.-Y., Yang, P., Zhang, Y.-M., Huang, K., and Liu, C.-L. (2014). Combination of classification and clustering results with label propagation. *IEEE Signal Processing Letters*, 21(5):610–614.

Zhong, C., Hu, L., Yue, X., Luo, T., Fu, Q., and Xu, H. (2019). Ensemble clustering based on evidence extracted from the co-association matrix. *Pattern Recognition*, 92:93–106.

Zhou, D., Bousquet, O., Lal, T. N., Weston, J., and Schölkopf, B. (2004a). Learning with local and global consistency. In *Advances in neural information processing systems*, pages 321–328.

Zhou, Z.-H., Chen, K.-J., and Jiang, Y. (2004b). Exploiting unlabeled data in content-based image retrieval. In *European Conference on Machine Learning*, pages 525–536. Springer.

Zhu, X., Ghahramani, Z., and Lafferty, J. D. (2003a). Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the 20th International conference on Machine learning (ICML-03)*, pages 912–919.

Zhu, X., Lafferty, J., and Ghahramani, Z. (2003b). Combining active learning and semi-supervised learning using gaussian fields and harmonic functions. In *ICML 2003 workshop on the continuum from labeled to unlabeled data in machine learning and data mining*, volume 3.

Zhu, X. and Ghahramani, Z. (2002). Learning from labeled and unlabeled data with label propagation.

A
Appendices

A.1
Appendix 1

Table A.1: Median accuracy with Median Absolute Deviation by k for Blind Uniform Random CRLP algorithm.

	α			
Dataset	0.2	0.4	0.6	0.8
Leukemia	0.92 (±0.04)	**0.92** (±0.04)	0.92 (±0.04)	0.92 (±0.04)
Lymphoma	0.85 (±0.08)	**0.85** (±0.07)	0.84 (±0.04)	0.84 (±0.02)
Armstrong	**0.8** (±0.06)	0.78 (±0.07)	0.76 (±0.09)	0.75 (±0.04)
Wine	**0.91** (±0.05)	0.9 (±0.06)	0.87 (±0.08)	0.72 (±0.11)
Chen	**0.84** (±0.07)	0.82 (±0.1)	0.72 (±0.14)	0.59 (±0.01)
Seeds	**0.86** (±0.05)	0.84 (±0.07)	0.78 (±0.11)	0.67 (±0.09)
Congressional	0.87 (±0.01)	**0.87** (±0.01)	0.87 (±0.01)	0.87 (±0.01)
Vertebral	0.61 (±0.07)	**0.61** (±0.06)	0.61 (±0.07)	0.57 (±0.07)
Breast_Cancer_1	0.91 (±0.01)	**0.92** (±0.01)	0.92 (±0.01)	0.92 (±0.01)
Synthetic	**0.71** (±0.09)	0.69 (±0.11)	0.67 (±0.07)	0.6 (±0.09)
Balance_Scale	**0.54** (±0.05)	0.53 (±0.03)	0.5 (±0.03)	0.47 (±0)
Urban	**0.61** (±0.08)	0.59 (±0.08)	0.55 (±0.06)	0.51 (±0.05)
Breast_Cancer_2	0.95 (±0.01)	**0.95** (±0.01)	0.95 (±0.01)	0.94 (±0.01)
Segmentation	**0.65** (±0.09)	0.64 (±0.08)	0.62 (±0.08)	0.56 (±0.07)
Mnist_Test	0.62 (±0.07)	**0.62** (±0.06)	0.58 (±0.06)	0.51 (±0.06)

A.2 Appendix 2

In this appendix, we show the accuracy dispersion over 100 simulations in the selection strategies applied to LP algorithm with replacement experiment.

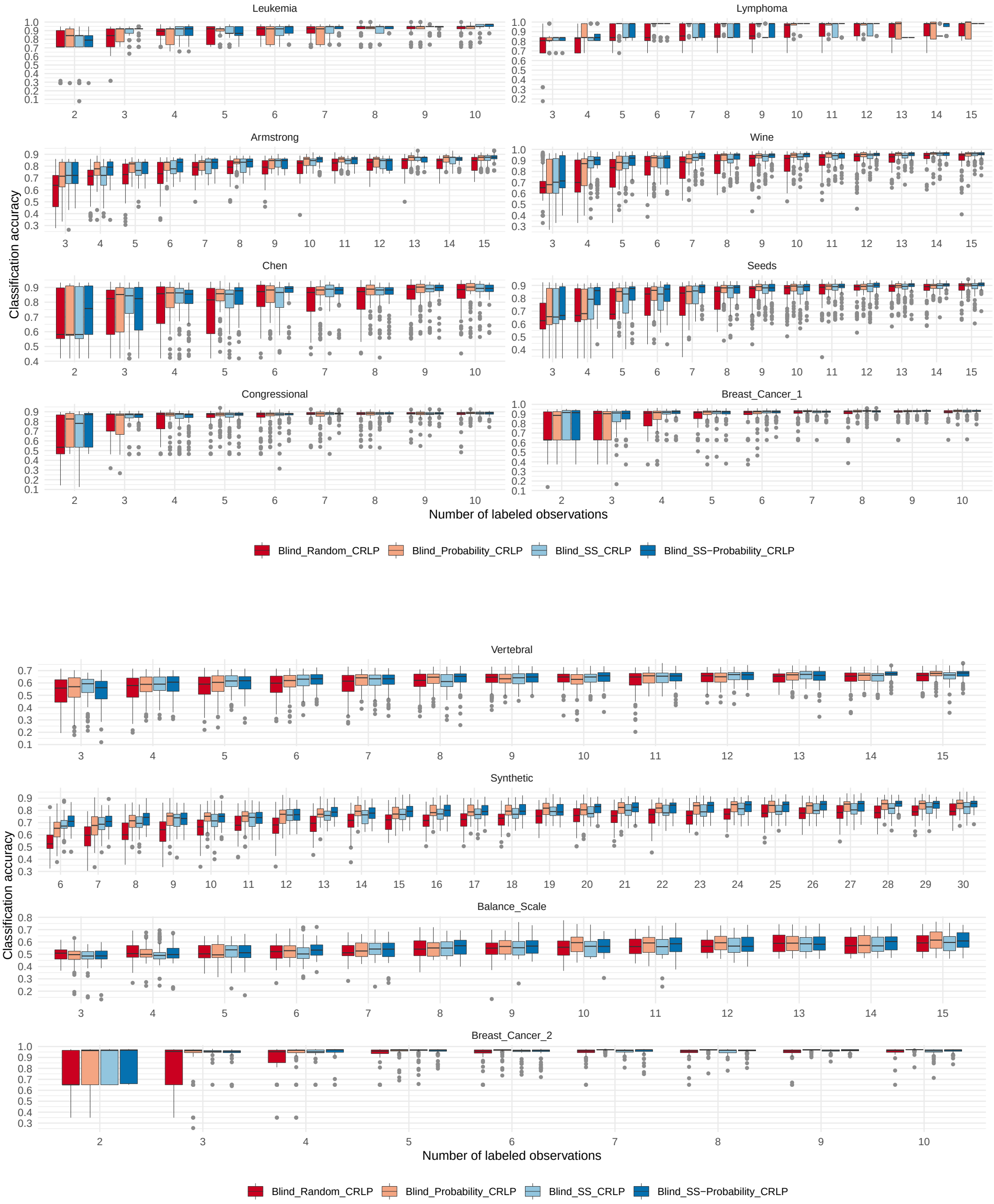


Figure A.1: Classification accuracy curves of the 100 simualtions for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms with replacement in 15 datasets.

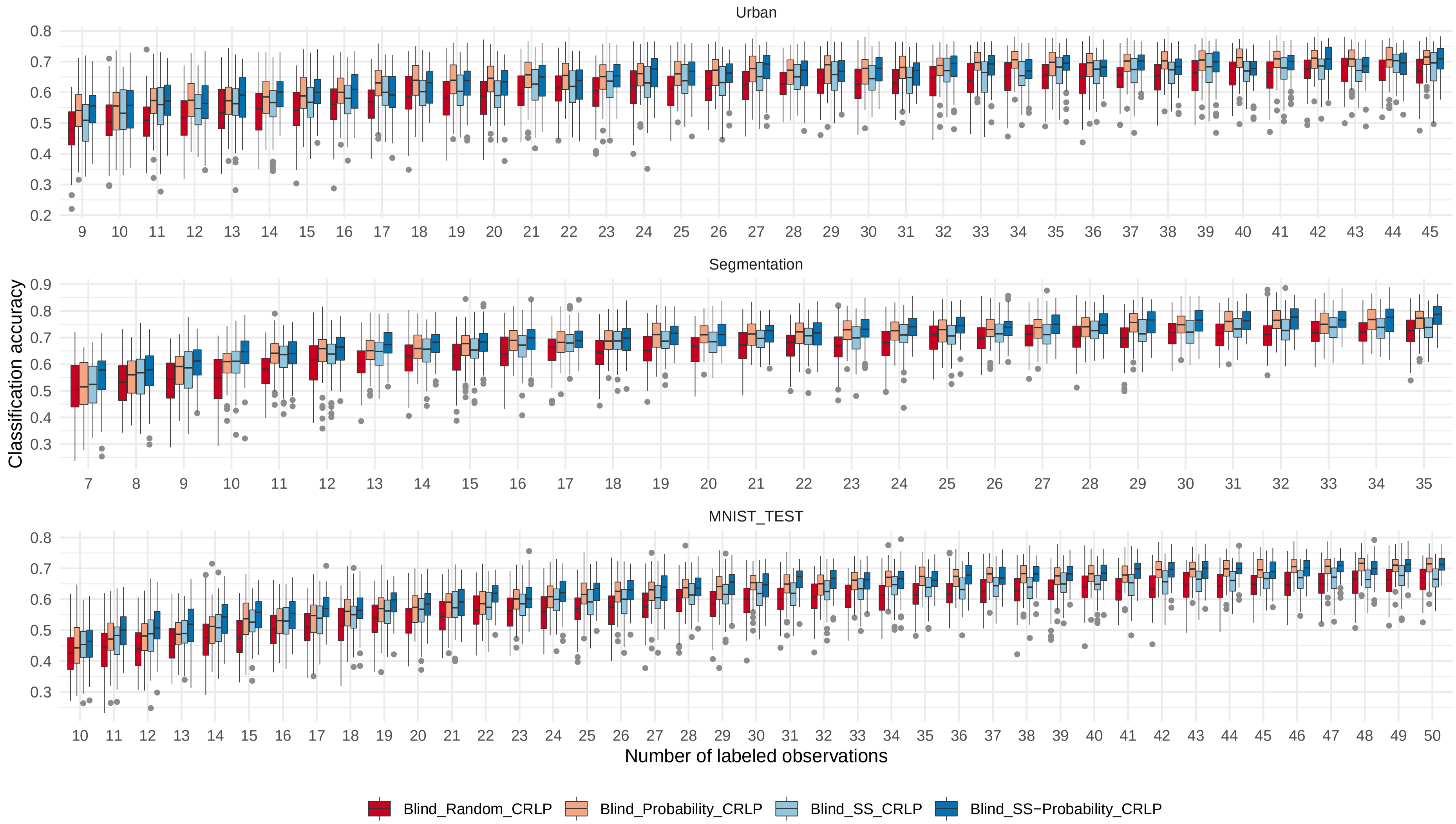


Figure A.2: Classification accuracy curves of the 100 simualtions for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms with replacement in 15 datasets (continued).

A.3
Appendix 3

In this appendix, we show the accuracy dispersion over 100 simulations in the selection strategies applied to LP algorithm without replacement experiment.

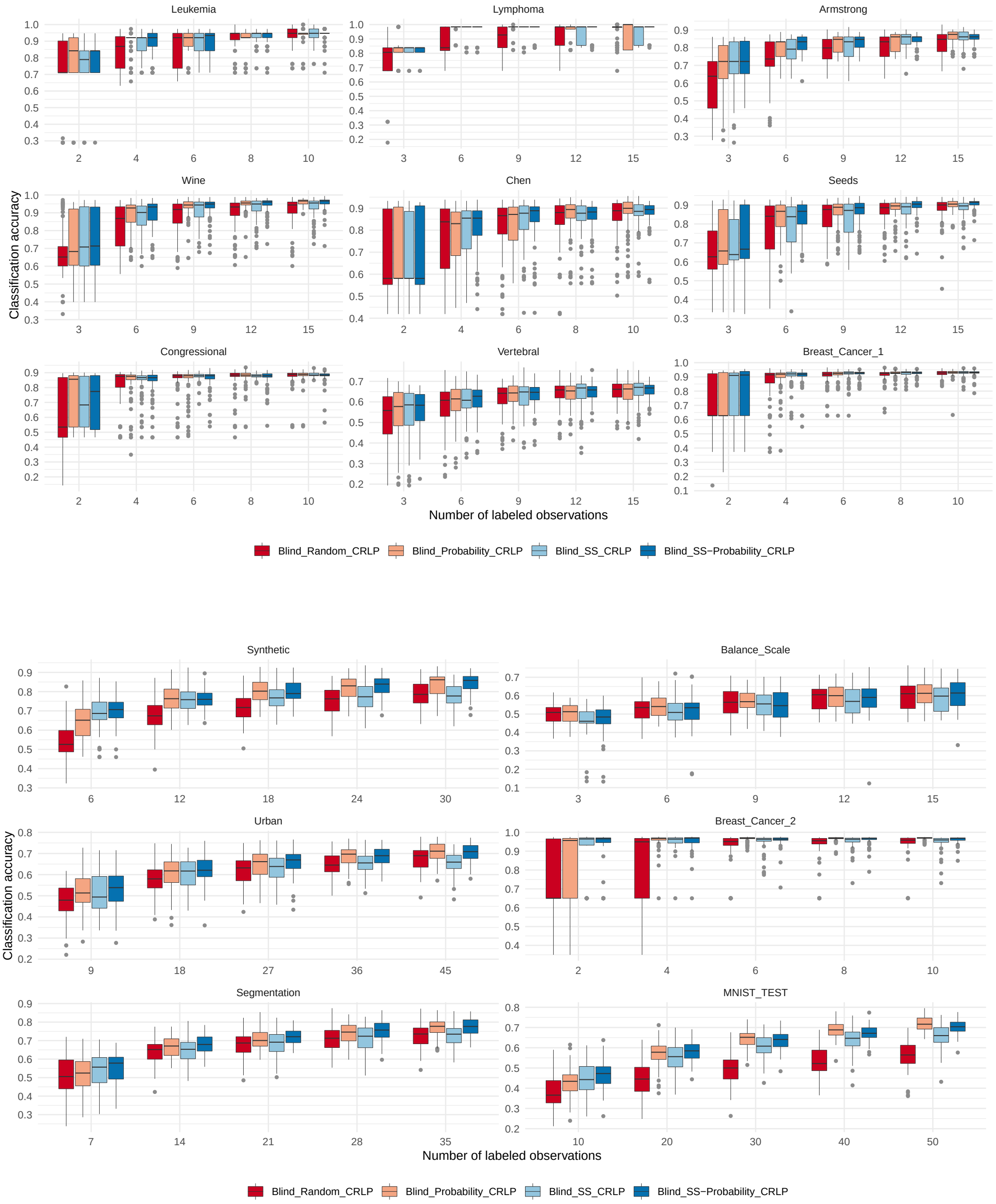


Figure A.3: Classification accuracy curves of the 100 simulations for the Blind Random CRLP, Blind Probability CRLP, Blind Stratified Sample CRLP and Blind Stratified Sample Probability algorithms without replacement in 15 datasets.

A.4 Appendix 4

In this appendix, we show the accuracy dispersion over 100 simulations in the selection strategies applied to AL with uncertainty experiment.

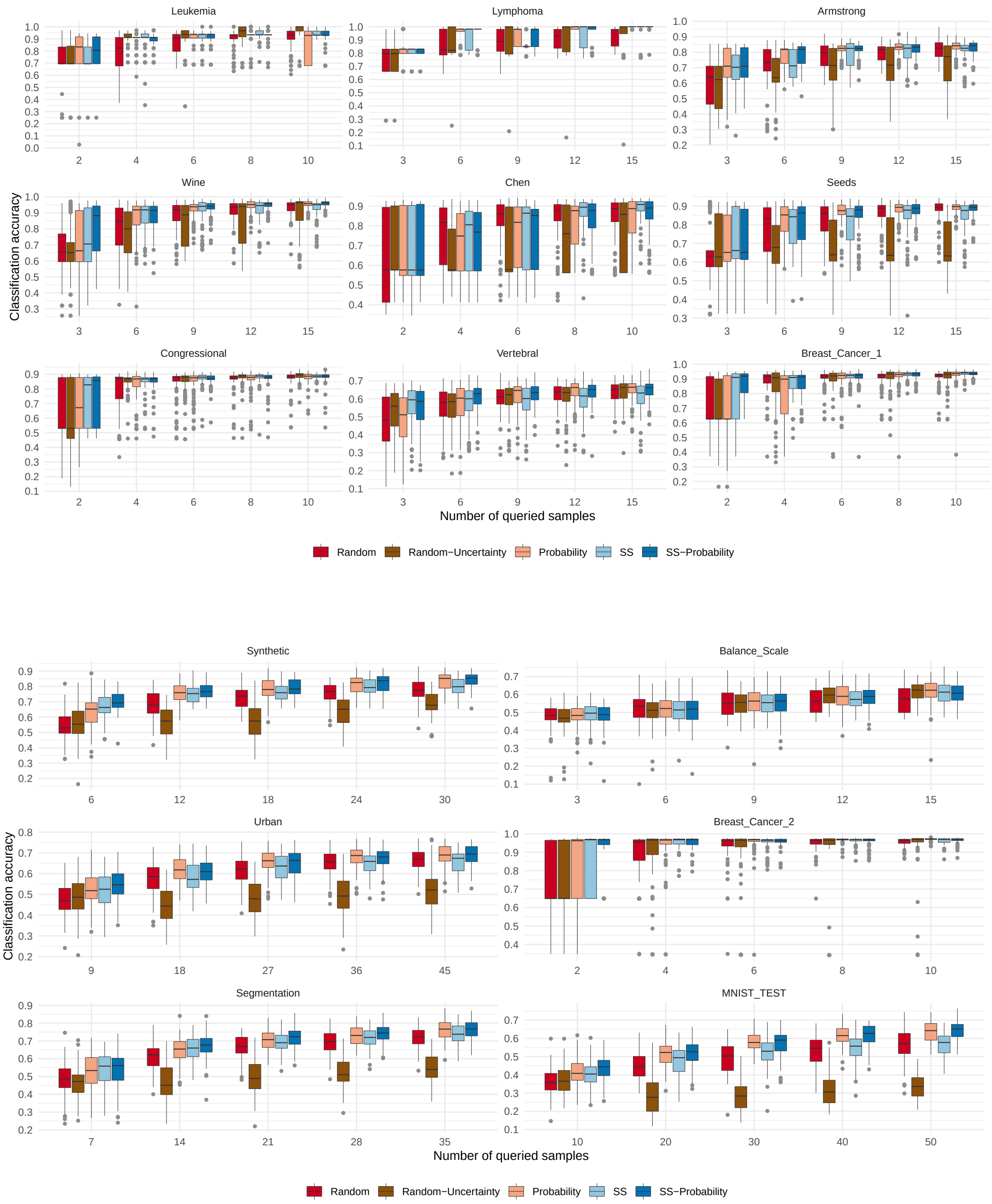


Figure A.4: Classification accuracy curves of the 100 simulations for the Random,Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with uncertainty in 15 datasets.

A.5 Appendix 5

In this appendix, we show the accuracy dispersion over 100 simulations in the selection strategies applied to AL with CM update experiment.

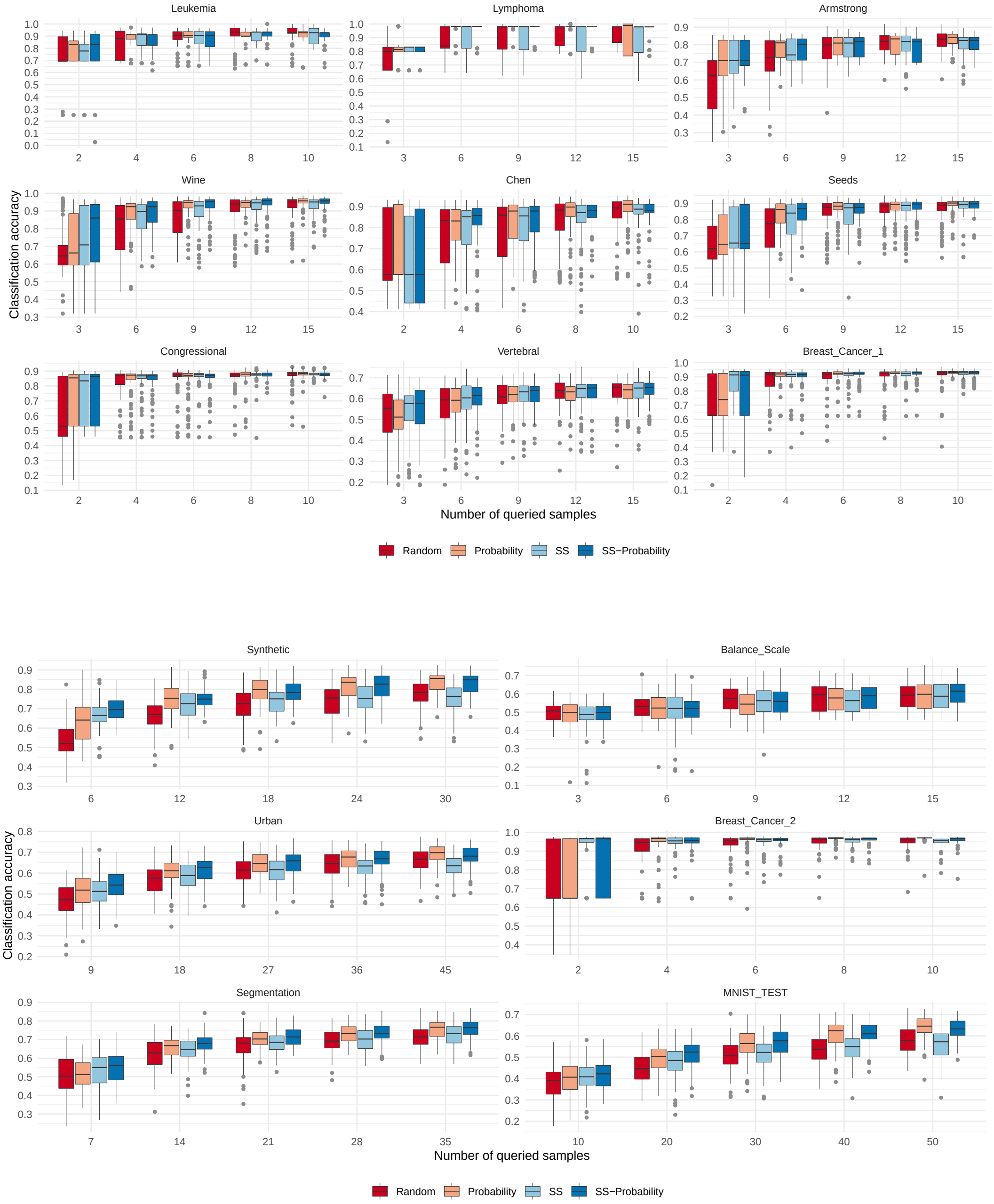


Figure A.5: Classification accuracy curves of the 100 simulations for the Random, Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with CM update in 15 datasets.

A.6
Appendix 6

In this appendix, we show the accuracy dispersion over 100 simulations in the selection strategies applied to AL with uncertainty and CM update experiment.

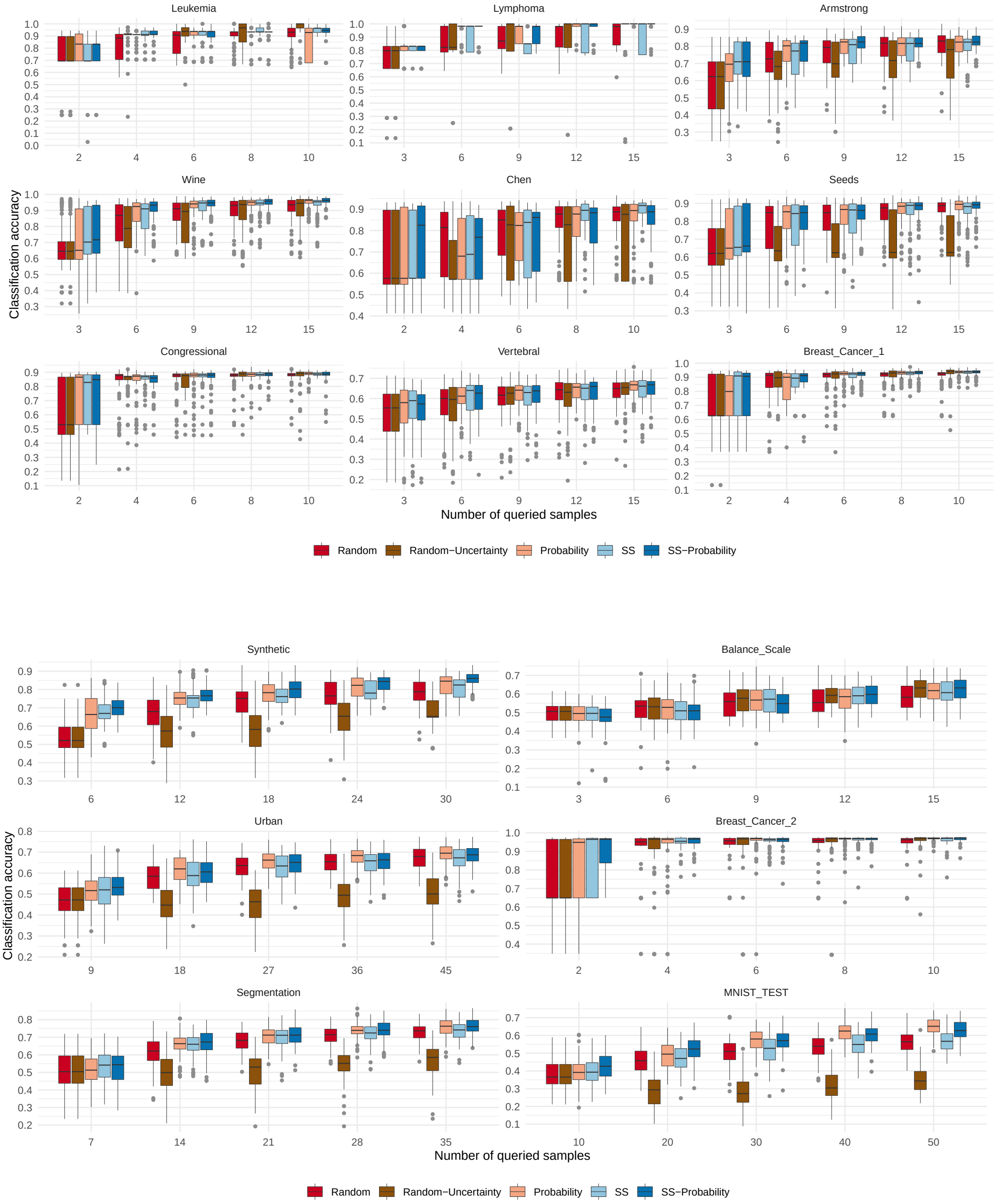


Figure A.6: Classification accuracy curves of the 100 simulations for the Random,Random Uncertainty, Probability, Stratified Sample and Stratified Sample Probability algorithms for the active learning process with uncertainty and CM update in 15 datasets.

A.7
Appendix 7

In this appendix, we show the CM for the 14 datasets.

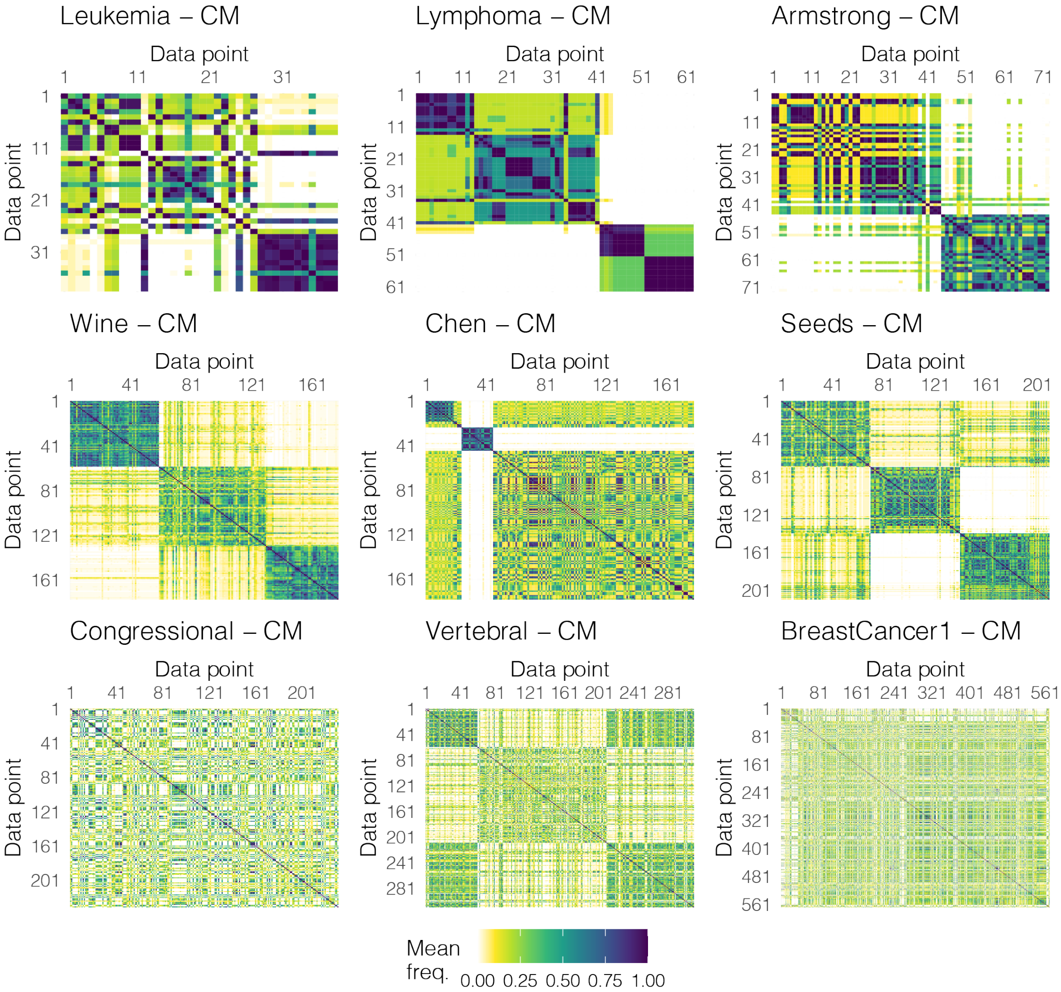
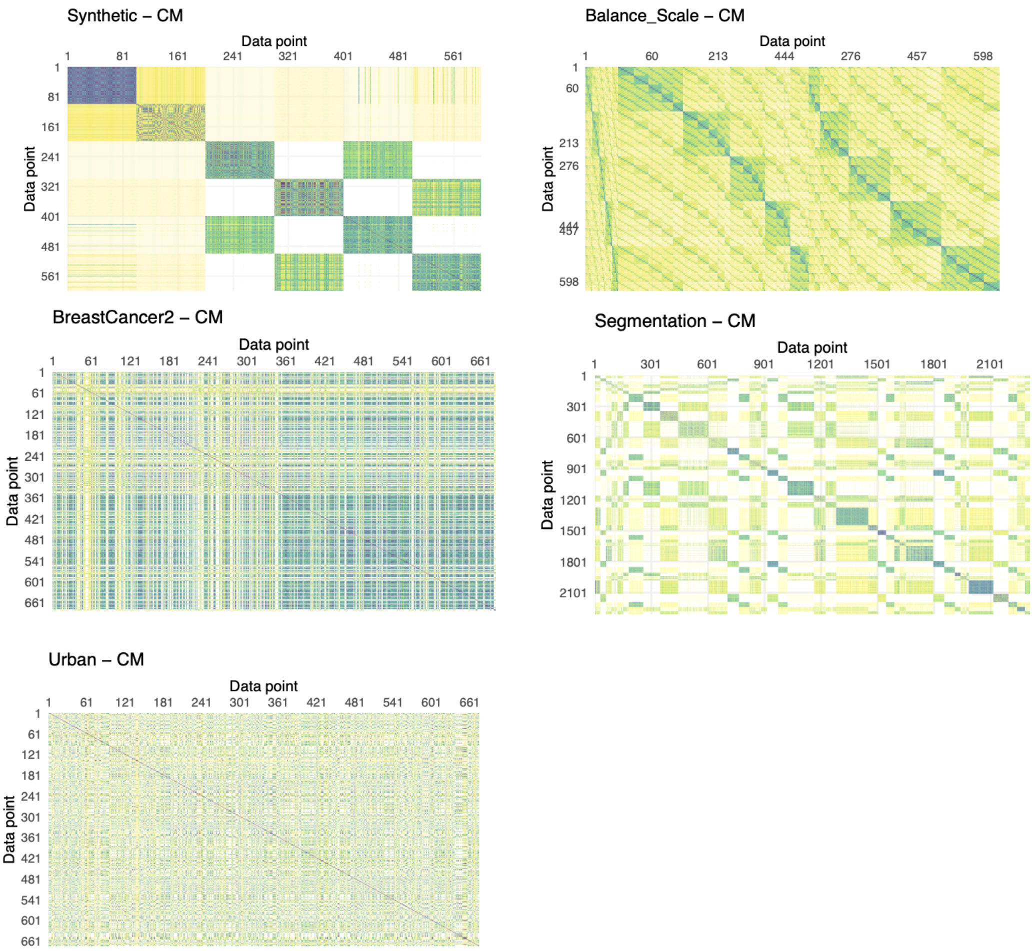


Figure A.7: Co-association matrix of Leukemia, Lymphoma, Armstrong, Wine, Chen, Seeds, Congressional, Vertebral and Breast Cancer 1 datasets.



Synthetic – CM

Balance_Scale – CM

BreastCancer2 – CM

Segmentation – CM

Urban – CM

Figure A.8: Co-association matrix of Synthetic, Balance Scale, Urban, Breast Cancer 2 and Segmentation datasets.