The improvement of the Learning Environment in the context of Multi-label data

Khalida Douibi, Nesma Settouti, Chikh Mohammed Amine, Jesse Read

Ministry of Higher Education and Scientific Research
Abou Bekr Belkaid University, Tlemcen
Faculty of Technology
Biomedical Engineering Laboratory (GBM)

–

A Thesis submitted for the degree of Doctor of Philosophy
In Biomedical Engineering

Speciality: Biomedical Informatics

DOUIBI Khalida

02 Mai 2019

The improvement of the Learning Environment in the context of Multi-label data

Abderrahim Mohammed Amine	MCA	Univ. Tlemcen.	President
Chikh Mohammed Amine	Professor	Univ. Tlemcen.	Supervisor
Read Jesse	MCA	École Polytechnique, Paris, France.	Co-supervisor
Benyettou Abdelkader	Professor	Univ. Mohamed Boudiaf, USTO.	Examiner 1
Souier Mehdi	MCA	École Supérieure de Management, Tlemcen.	Examiner 2
Settouti Nesma	MCB	Univ. Tlemcen.	Invited member
Benabid Malik	Doctor	CHU Sétif.	Invited member

Dedication

I dedicate my dissertation work to ...

My beloved grandparents and all my family.
My great parents, my wonderful brother and sister: Mohcene and Selma.
My friends: Amine, Mostafa, Amina, Souad, Mounir, Riheb, Siwar, Olivier and Ahmed.
My dear Nesma Settouti who deserve my deepest appreciation for her continual support.
My colleagues from CREDOM and DaSciM team.

Khalida, March 2019

Acknowledgements

My PhD experience was three years full of Scientific Research and Personal Development that taught me how to keep always positive attitude, be patient, perseverant and work hard to achieve my goals.

Firstly, I wish to deeply thank the members of my dissertation committee for accepting to review my work and providing me with the directions and comments to improve it: Prof. Abderrahim Mohammed Amine, Prof. Souier Mehdi from Tlemcen University and Prof. Benyettou Abdelkader from Mohamed Boudiaf University, Oran.

In addition, I would like to express my sincerest thanks and appreciations to my advisor Prof. Chikh Mohamed Amine and Dr. Settouti Nesma from Tlemcen University, for their continuous guidance, patience and valuable contributions during all my PhD study.

My gratitude goes also to my passionate Co-advisor Prof. Read Jesse from Ecole Polytechnique, Palaiseau, who provided me with the opportunity to join DaSciM team at LIX laboratory. His support, contributions and expertise has greatly assisted this research.

Furthermore, I acknowledge my gratitude to Prof. Boussouf and Dr. Benabid from CHU Sétif cardiology department, Algeria- for their participation and shared expertise during my internship in their service.

I would like to thank my colleagues from Biomedical Engineering laboratory in Tlemcen University, especially Dr. Bechar Amine, Dr. El Habib Dahou Mostafa from CREDOM team for their presence and help when needed during all my studies. Likewise, my colleagues from Informatics laboratory of Ecole Polytechnique (LIX) at Palaiseau who made my experience amazing and very successful, great memories with you stay forever.

My special gratitude goes to my beloved parents, my brother and sister who supported me throughout my studies and encouraged me to keep going and stay strong and positive all the time.

Khalida, March 2019

Résumé

Au cours des dernières années, l'apprentissage Multi-labels a attiré l' attention d'une large communauté de chercheurs de plusieurs domaines. La catégorisation du texte était parmi les premières applications de ce type d'apprentissage dans laquelle un document peut être annoté par plusieurs labels à la fois. Par la suite, ce domaine de recherche a été étendu vers d'autres applications du monde réel.

Dans notre thèse, nous nous sommes intéressés à l' application de la Classification Multi-labels pour l'aide au diagnostic médical. Notre première piste de recherche a été consacrée à l' étude des avantages de l'utilisation d' un comité de modèles d'apprentissages à la place d' un seul apprenant. L' approche qui a été étudiée adapte l'algorithme du *k-plus-proches-voisins* au Multi-labels [1]. Deux stratégies de méthodes d' *Ensembles Homogènes* ont été étudiées y compris le *Bagging* [2] et le *Boosting* [3].

La seconde contribution de notre travail concerne une collecte d'une nouvelle base de données médicale de la *Mesure Ambulatoire de la Pression Artérielle* (*MAPA*) [4], qui constitue un outil très puissant et largement sollicité par les cardiologues pour une meilleure prise en charge des patients hypertendus. Dans le même travail, nous avons proposé l'utilisation des méthodes Multi-labels pour une analyse automatique des données *MAPA* [5]. Une première étude de corrélation entre les six labels de cette base de données a été également réalisée et ce qui nous a permis de déduire l'importance d'étendre notre étude de dépendance de labels en utilisant des techniques plus spécialisées.

La dernière partie de notre thèse a été consacrée à l'étude de ce concept en détails. Nous avons présenté une revue de la littérature des algorithmes étudiant également cette problématique, et nous avons appliqué six algorithmes Multi-labels issus des deux grandes familles de méthodes de Transformation et d' Adaptation, basées sur les arbres de décision pour une meilleure interprétabilité des résultats. A la fin, les résultats retrouvés ont été discutés et plusieurs pistes de recherches pour le futur ont été proposées.

Mots clés: Classification Multi-labels, Corrélation des labels, méthodes de Transformation, algorithmes d' Adaptation, Données Médicales, MAPA.

Abstract

Over the last few years, *Multi-label Learning* (*MLL*) has attracted the attention of a large community of researchers in many fields. Initially, it was applied for text categorization in which the annotation of a document that belongs to multiple categories require specific approaches. Thereafter, *MLL* is being increasingly required in other many real-world applications.

In our work, we considered *MLL* for the medical aid diagnosis, our first research goal was the investigation of the advantages of using committee of learners to improve a Multi-label algorithm that adapts *K-Nearest-Neighbors* (*KNN*) to Multi-label problem called *MLKNN* using *Bagging* and *Boosting*.
- Secondly, we gathered a medical Multi-label dataset that concerns *Ambulatory Blood Pressure Monitoring* (*ABPM*) which currently occupies a central place in the diagnosis and follow-up of hypertensive patients. We also proposed, an intelligent analysis of *ABPM* records using Multi-label Classification algorithms allowing the expert to analyze them more quickly and efficiently. In addition, it could help to investigate label dependencies and provide interesting insights.

The satisfactory findings and interpretations of this work, conducted us to investigate more about the advantages of using *Decision Trees* (*DT*) to extract new and implicit correlations between different labels and features in a given dataset. For that, we reviewed recent works addressing Label dependencies based on several Multi-label algorithms based on *DT*. We presented also the main differences between the two defined types of Label correlation named Conditional and Unconditional (Marginal) Label dependence. Finally, we conducted a comparative study of six well-known algorithms in the literature, and we discussed the benefits of considering Label dependence using *DT* algorithm as a base classifier for both Transformation and Adaptation algorithms.

Finally, potential further works and future directions of our thesis were highlighted.

Keywords Multi-label Classification, Label Correlation, Transformation methods, Adaptation algorithms, Medical Dataset, ABPM.

Contents

List of Figures

List of Tables

Glossary

ABPM: Ambulatory Blood Pressure Monitoring.
AdaBoost: Adaptive Boosting.
AdaBoost MLKNN: Adaptive Boosting Multi-label K-Nearest-Neighbors.
AI: Artificial Intelligence.
API: A Programming Interface. *Bagged MLKNN*: Bagging Multi-label K-Nearest-Neighbors.
Bagging: Bootstrap and Aggregating.
BCC: Bayesian Chain Classifiers.
BN: Bayesian Network.
Boostexter: Boosting for text categorization.
BP: Blood Pressure.
BPL: Blood Pressure load.
BP-MLL: Multi-label Back Propagation.
BPV: Blood Pressure Variability.
BR: Binary Relevance.
BRKNN: Binary Relevance k-Nearest-Neighbors.
CC: Classifier Chain.
CCo: Conditionally Correlated.
CHU : University Hospital Center.
CI: Conditionally Independent.
CLR : Calibrated Label Ranking.
DAG : Directed Acyclic Graph.
DBP : Diastolic Blood Pressure.
DMl-KNN: Dependent Multi-label K-Nearest-Neighbors.
DT: Decision Trees.
EBR : Ensemble of Binary Relevance classifiers.
ECC : Ensemble Classifier Chains.
ELPPJD: Ensemble Label Powerset Pruned datasets Joint Decomposition.
EnML: Multi-label Ensemble Learning.
EPS: Ensemble of Pruned Sets.
ERT: Extremely Randomized Trees.
Fn: False negatives.
Fp: False positives.
HBP : High Blood Pressure.
HOMER: Hierarchy of Multi-label classifiers.
IBLR: Instance-Based Learning by Logistic Regression.
ICD: International Classification of Diseases.
IR: Information Retrieval.
KNN : K-Nearest-Neighbors.
LaCova: A Tree-based Multi-label Classifier using Label Covariance as splitting criterion.
LP: Label Powerset.
LR: Label Ranking.

MAP: Maximum A Posteriori.
MAPA: Mesure Ambulatoire de la Pression Artérielle.
ML: Multi-label.
ML SVMDT: Multi-label Support Vector Machine Decision Tree.
MLC : Multi-label Classification.
MLKNN: Multi-label K-Nearest-Neighbors.
MLL: Multi-label Learning.
MLNB: Multi-label Naive Bayes.
ML-RBF: RBF Neural Networks for Multi-label Learning.
ML-SVDD: Support Vector Data Description.
ML-Tree: Multi-label Tree.
MMP: Multi-label Multi-Class Perceptron.
MS: Morning Surge.
NC: Not Correlated.
NCBI: National Center for Biotechnology Information. *NN*: Neural Network.
OOB: Out-Of-Bag.
PC: Probably Correlated.
PCA : Principal Components Analysis.
PCC : Probabilistic Classifier Chain.
PCT: Predictive Clustering Tree.
PNN : Probabilistic Neural Network.
PP: Pulse Pressure.
RAkEL RAndom k-labELsets.
RBF : Radial Basis Function.
RFMLC4.5: Random Forest of Multi-label-C4.5.
RFPCT: Random Forest Predictive Clustering Tree.
RPC : Ranking by Pairwise Comparison.
SBP: Systolic Blood Pressure.
SMOTE: Synthetic Minority Oversampling.
SVM: Support Vector Machine.
TCM: Traditional Chinese Medicine. *Tn*: True negatives.
Tp: True positives.

Introduction

1 The Scope of the Thesis

One of the main challenges of Information Technology is the extraction of knowledge from data, the availability of data online and the advances of data storage technologies helped researchers to mine huge datasets to drive important decision making in many fields, the concerned task is known as data mining.
Data mining was widely used in many domains to find human-interpretable patterns hidden in huge datasets. In Financial for example, it allows to predict the customer's behaviour and propose relevant products based on his past actions. Similarly in the education field, it helps to determine the most effective tools and way to teach students.

Another crucial domain that interest a large community of researchers is the medical field, mining healthcare data can help to discover new information about patients, their correlations and also to inform the expert about the effectiveness of received treatments. Another advantage of using data mining tools is the gain of time by using several advanced techniques. For example, recently many Magnetic Resonance Imaging (MRI) of the brain provides automated segmented images that play an important role in neurology and neurocognitive research [8]. Such a task can be accomplished using data mining techniques that learn from collected datasets, where images should be annotated preliminary by an expert. The set of images are the examples of learning of the model, known as classifier if we aim to classify images according to their contents and lesions, this task is called Classification in machine learning.

Classifying medical images can aim to only determine the presence or the absence of a targeted lesion, the Classification is called then Binary Classification. However, in many advanced applications, the goal is more complex and the image is labeled by many labels at once, in this case, the Classification is known as Multi-label Classification (MLC).

Over last few years, MLC has been applied in many other real-world applications, such as text categorization, tagging several multi-media resources including images, audio and videos and Classification of genes according to their genomic functions etc.

Plenty of research works addressed the issue of learning from Multi-labeled data in several ways. On the one hand, a lot of them interested in the adaptation of well-known algorithms directly by modifying them to consider Multi-label (ML) problem such as the adaptation of K-Nearest-Neighbors rule in [1] and the adaptation of Naive Bayes rule in [9], similarly Decision Trees algorithm was adapted by modifying the entropy formula of MLC4.5 in [10]. On the other hand, a large community of researchers focused on transforming the Multi-label problem to other types of popular learning tasks like Binary Classification, Multi-class Classification etc.

The two main transformations are Binary Relevance (BR) and Label Powerset (LP) [11]. In the first transformation, each label is predicted separately without taking any consideration about the presence of other labels. However, the second address this issue partially, by considering each combination of labels in the dataset as a new class to predict, i.e, it transforms the Multi-label problem to Multi-class Classification.

Another strategy of learning that demonstrated their efficiency over the past is the use of committee of learners instead of only one, known as Ensemble Methods. Each model is learned on a diverse learning set, this strategy was adapted to MLC in many works to overcome many challenges encountered by researchers in this research field such as taking Label dependence during the Classification process, also the improvement of predictive performance by using many classifiers each one is specialized in a label subset as in RAkEL [11]. The Evaluation Measures used to determine whether the model predictions are correct or not differs from those used traditionally in machine learning, the main reason is that the prediction could be fully correct, the metric used for that is called Exact Match or Subset Accuracy [12]. It could be fully wrong or partially correct, many Evaluation Metrics were proposed for Classification task as Accuracy (Jaccard index) [13], Hamming Loss, F1-Score [13]. Other Evaluation Measures were proposed for Ranking labels according to their relevance such as One Error, Ranking Loss, Average Precision [1].

Throughout this thesis, all brief notions presented above about MLC will be detailed with reviewing major works in the literature. We present in the next section our research goals and the scope of each chapter of our manuscript.

2 Summary of Research Goals and Contributions

In the medical field, the use of machine learning algorithms for decision aid diagnosis was greatly discussed in many works over the past few years. Many of them addressed the Multi-label issue for healthcare data by proposing very powerful methods based Transformation or Adaptation algorithms or an hybrid approach from both families. However, in general, the validation of such approaches is conducted using biological datasets proposed in the literature. Unfortunately, the publicly available Multi-label medical datasets are very rare.

In our work, we focused on considering Multi-label (ML) for medical applications. Our first research goal was the investigation of the benefits of using homogeneous Ensemble Methods (*Bagging* and *Boosting*) for *MLKNN* algorithm [1] that adapts *KNN* to Multi-label data. The results are very competitive and show that the use of several learners simultaneously for the prediction of labels improve the performance of the individual classifier. Further details about the first contribution can be found in Chapter 2.

Our first work motivates us to focus on gathering a medical Multi-label dataset to investigate the impact of using ML approaches to solve real medical problems. For that, we studied the Ambulatory Blood Pressure Monitoring (*ABPM*) that currently occupies a central place in the diagnosis and follow-up of hypertensive patients.

ABPM involves measuring blood pressure by means of a tensiometer carried by the patient for a duration of 24 hours, it provides crucial information which allows to make a specific diagnosis and adapt therapeutic attitude accordingly. In this work, we attempt to improve the analysis of *ABPM* data using Multi-label Classification methods, where a

record is associated with more than one label (class) at the same time. Our contribution aims to solve many problems by saving time and manual calculations by the expert, who is generally very busy.
For that two major contributions were proposed, the first one was the publication of a new Multi-label dataset that concerns ABPM data, the records are characterized by 40 features and are categorized into one or more out of 6 labels. The dataset is released to the public [4] to allow comparative experiments by other researchers.

Our second contribution in this work was the intelligent analysis of *ABPM* records using Multi-label Classification algorithms. The medical diagnostic process supported by such techniques constitutes a modern and useful tool for medical aid decision, allowing the expert to analyze *ABPM* record more quickly and efficiently. Results show that the Multi-label modelling of *ABPM* data helps to investigate label dependencies and provide interesting insights, which can be integrated into the *ABPM* devices to dispense automatically detailed reports with possible future complications. More details and explanations about this second work can be found in Chapter 3.

Satisfactory results and interpretations found in the work above, conducted us to investigate more about the advantages of using *DT* to extract new and implicit correlations between different labels and features in a given dataset. For that our last research goal focus on studying in depth this point.

For that, we reviewed recent works addressing Label dependence based on several Multi-label algorithms, including Transformation methods and Adaptation algorithms. We present also the main differences between the two defined types of Label correlation named Conditional and Marginal. Finally, we presented a comparative study of six well-known algorithms in the literature based DT, and we discussed the benefits of considering Label dependence using *DT* algorithm as a base classifier for both Transformation and Adaptation algorithms.

Chapter 4 presents in-depth this study. Nevertheless, further works are currently underway including the publication of many algorithms of literature that was implemented in Python.

3 Thesis Organization

The idea behind the organization of the present thesis was to guide a new reader on machine learning approaches by giving simplified example from real-world applications, the main reason for that is that the targeted audience of this thesis is both researchers on machine learning but also doctors who are interested in such task to include it in their future research works. Hence, the manuscript gives a detailed overview on Multi-label Classification framework and it is organized as follows.

- Chapter 1 aims to provide the reader with the proper insight to take advantage of machine learning task to deal with real-world applications. It presents a comprehensive review of Multi-label Classification approaches and also some popular ML toolboxes and data repository widely used by researchers in this field.

- Chapter 2 studies the use of Ensemble Methods in the Multi-label framework, it briefly presents the basic concepts of Ensemble Methods in Single-label Classification, then, introduce a work that we conducted by applying Ensemble Methods for a Multi-label algorithm called MLKNN, that adapt KNN to ML. We aims to improve its performance using Ensemble Methods (*Bagging* and *Boosting*).

- Chapter 3 presents in depth the application of ML approaches to solve a real medical problem, by providing an automatic analysis of the Ambulatory Blood Pressure Monitoring (ABPM). We aim to help the expert to exploit and to analyze them easily since the traditional analysis is time-consuming which constitutes a real gene for expert.

- Chapter 4 reviews one of the recurrent studied topic in Multi-label Classification (MLC) which is Label dependence and its advantages for this research field. We presented recent works addressing Label dependence based on several Multi-label algorithms, including Transformation methods and Adaptation algorithms. The main differences between the two defined types of Label correlation namely Conditional and Marginal were highlighted by using Decision Trees as a base classifier. The goal was to investigate the benefits of using Decision Trees to model Label dependence in an interpretable way. A comparative study were conducted based on six well-known algorithms from literature using five datasets commonly used, in addition of our collected ABPM dataset (Refer to Chapter 3).

- Finally, the last part of our manuscript (Conclusions and Future Directions) concludes the work and outlines Further Research directions.

Chapter 1

Introduction to Multi-label Learning Framework

1 Introduction

Extracting knowledge from data has been a huge topic of discussion in recent years, and has attracted the whole information technology community in the world. The dramatic growth of the data available on-line and the advances data storage technologies has made data mining a required task, to identify the knowledge hidden in the data and for gaining insight to drive decision making.

Data mining is widely used in several domains, we will discuss briefly in this chapter its main applications and its new trend, and *why can such task interest the medical community ?* Moreover, *how can it help in the creation of medical aid diagnosis systems ?*

The first application that can make you realize the interest of such task in our modern life is, the use of data mining systems in supermarkets, the idea is to manage the customers data and predict future action based on past actions, in other words, if someone buys a certain group of products it's easy to predict what else they will buy. Such information is crucial for supermarkets because it helps them to change their layouts accordingly and it makes sense to keep the targeted products close together.

Another interesting application is education, where advanced data mining tools can discover the most effective way to teach students. It can help also to adapt the content of courses based on their skills.

In the financial field, data analysis plays an important role in allowing banks to predict customers behavior and propose relevant services and products accordingly.

Finally, the analysis of healthcare data can help greatly discovering the relationships between diseases, for example, it can inform about the effectiveness of treatments and identify new drugs, or ensure that patients receive appropriate, timely care. It can also predict the number of people falling victim to every pathology and inform the appropriate institutions how can they reduce health costs too. More applications and details about data mining and knowledge discovery in databases can be found in Fayyad et al. [14].

Data mining process needs several techniques to extract information from the data,

in practice, its main high-level goals tend to be Prediction and Description. While the first one involves predicting unknown patterns based on some variables, the second one focuses on finding human-interpretable patterns hidden in the data. Although the boundaries between Prediction and Description are not sharp (some of the predictive models can be descriptive, to the degree that they are understandable, and vice versa) [14]. The ideal case is to find a model that combines perfectly between Description and Prediction goals. Classification, Regression and Clustering are the widely used methods by data miners. Classification is a learning task that classifies the data into one of several predefined classes, Regression task has the same learning goal, although, it maps a data item to a real-valued prediction variable. Finally, clustering is a common descriptive task that aims to identify a finite set of categories (clusters) describing the data.

In the present thesis, we are interested in Classification task, and specifically, learning from Multi-label data, we study how can the analysis of medical data using machine learning algorithms helps the doctors in the diagnosis and treatment process. We present a healthcare application that concerns Ambulatory Blood Pressure Monitoring (*ABPM*) data (refer to Chapter 3). The next section describes in depth our motivation and the fundamental concepts of such task and provide the necessary background to understand the upcoming parts of this thesis.

2 Multi-label Classification Problem

In machine learning, the Classification is one of the main tasks that aim to learn a model on a set of training examples, characterized by a set of features and labeled by a class. This model should be able to assign the proper class to new examples based on only its features. There are many possible Classification learning tasks, the following example explains well the difference between the different forms of Classification.

Suppose that we have a dataset containing a set of patient's data, each patient has some features, such age, gender, heart rate, weight, height, blood pressure, glucose level etc. the aim is to predict the pathology (output) for each patient. Hence, the Classification task depends on the output vector, it is called *Single-label Classification* when we consider just one class, for example, the class is called Diabetes, and we have only two values for this class (Diabetic or normal), then the Classification process is called *Binary Classification*.

Suppose now that we have many classes in the output vector, which represent the types of Diabetes including Normal case: Normal (N), Type 1, Type 2, Type 3. Thereby, the learning task is called *Multi-class Classification*, since the output contains four classes which are mutually exclusive, i.e. we cannot find healthy patient (class=N) attained by Diabetes Type1 at the same time. Figure 1.1 highlight the main difference between Single-label and Multi-label Classification.

	Single Output		Multi Outputs					
	Binary	Multi-class	Multi-label			Multi-Dimensional		
	Diabete		Diabete	CVR	RF	Diabete	CVR	RF
Instance 1	P	Type 1	P	P	N	Type 1	VES	P
Instance 2	N	Type 2	N	N	N	N	N	N
.....	.....	.....	.....	.....	.....	.....	.....	.....
Instance i	P	N	P	P	P	Type 2	AES	P

N: Normal
P: Pathological
CVR: CardioVascular Risk
RF: Renal Failure
VES: Ventricular ExtraSystole
AES: Atrial ExtraSystole

Figure 1.1: Single-label vs. Multi-label dataset.

However, in the medical field, a patient can be attained by several pathologies simultaneously considered in the learning process as many outputs (labels), where each instance is associated with many labels simultaneously. In our example, a patient can be affected by many pathologies as Diabetes, Cardiovascular risks, Renal failure, and Hypertension at the same time. If we consider for each label just two possible values (normal or pathological), then the Classification is called *Multi-label*. Furthermore, if many classes are considered for each label, the Classification process is more general and it is known in the literature as Multi-dimensional Classification. Table 1.1 present the different Classification tasks.

Table 1.1: Classification problems according to the output to be predicted.

Number of outputs	Output type	Classification task
1 per instance	Binary	Single label
1 per instance	Multivalued	Multi-class
q per instance	Binary	Multi-label
q per instance	Multivalued	Multi-dimensional
1 per M instances	Binary/Multivalued	Multi-instance

In summary:

- In Single Classification, we have only one output vector, if it contains (True/False) values, then the Classification is called Binary Classification. Nevertheless, if it contains more than one class, which are mutually exclusive, the Classification task is called Multi-class Classification.
- In Multi-label Classification, each instance is associated with a set of labels simultaneously, and each label has only binary values. If in addition, each label has more than one possible class, then the Classification is called Multi-dimensional. Table 1.1 summarize Classification problems according to the predicted outputs.

2.1 Multi-label for real-world applications

During last years, Multi-label Learning (*MLL*) has become a very hot topic, due to the increasing number of fields where it can be applied, also to the emerging number of techniques that are being developed to deal with the learning task. Initially, *MLL* was used for text categorization [15]; [16] where a document can belong to multiple categories at once or have multiple tags. Thereafter, *MLL* is being increasingly required in many real-world applications, such as semantic annotation of images [17], [1], [18] and video [19], functional

genomics [20], [10], [19], music categorization into emotions [[21], [22]. Table 1.2 reviews some Multi-label Learning applications from literature [23].

Table 1.2: Some Multi-label Learning Applications from literature.

Data type	Application	Resource	References
Text	Categorization	News article	[24]
		Web page	[25]
		Patent	[[26], [27]]
		Email	[28]
		Legal document	[29]
		Medical Report	[30]
		Radiology Report	[31]
Image	Semantic annotation	Pictures	[[17], [1], [18]]
Video	Semantic annotation	News Clip	[19]
Audio	Noise detection	Sound Clip	[32]
	Emotion detection	Music Clip	[[21], [22]]
Structured	Functional genomics	Gene	[[20], [10], [19]]
	Proteomics	Protein	[27]

We present hereafter several application areas of Multi-label Learning with some case studies found in the literature. Table 1.3 presents dataset collected from each case studies with their original references and the download links.

Text mining is one of the main tasks where Multi-label Classification learning was greatly applied. the idea is to transform a set of text documents into Multi-label dataset, each row corresponds to a document and the columns represent the characteristics kept after the preprocessing step, where uninformative words are removed and only representative words with their frequencies are used as a discriminative features. We present below some datasets usually used in this field:

- *Bibtex dataset* was introduced in [33] as part of a tag recommendation task, it contains the meta-data for bibliographic entries. The input attributes are the words presented in the papers' title, authors names, journal name, and publication date and there is in total 1 836 features associated with a total of 159 different labels. The data was collected from Bibsonomy [1], a specialized social network where the users can share bookmarks and BibTeX entries assigning labels to them, The boolean Bag of Words model is used to represent the documents, so all features are binary indicating if a certain term is relevant to the document or not [34].

- *Medical* [35]: the dataset was created from anonymized clinical texts, where patient symptoms are described. Words describing each document represent features that are associated with a total of 45 labels, that represent codes from the International Classification of Diseases, precisely ICD-9-CM8 codes [34].

- *Enron*: The Enron corpus is a large set of email messages, with more than half a million entries, from which a dataset for automatic folder assignment research was

[1] http://www.bibsonomy.org.

generated [36]. The Enron Multi-label dataset has 1701 instances assigned to 53 labels, each label correspond to the folders in which each message was stored into by the users [34].

- *Bookmarks* [33]: comes from the same source as *Bibtex* dataset, the data are obtained from the bookmarks shared by the users. Specifically, the URL of the resource, its title, date, and description are included in the dataset. The input features consisted in 2150 different terms and outputs features are the 208 tags assigned to each instance.

- *Delicious* [37]: the nature of this dataset is the same as the previous one, but this time the links to web pages were taken from the del.icio.us4 portal. The page content for a set of popular tags was retrieved and parsed, and the resulting vocabulary was filtered to avoid nonfrequent words [34]. The produced dataset contains a so large number of labels.

- *Reuters* [38]: is a subset of RCV1V2 (Reuters Corpus Volume 1 Version 2) text corpus generated from the full text of English news published by Reuters along one year, from August 20, 1996, to August 19, 1997. The dataset contains a reduced set of attributes as the goal of the study was to work with more representative features to improve the speed of the learning process [34].

Multimedia Ressources Labeling Although text categorization application was the first to use Multi-label Classification, the huge amount of collected data in other fields demand automated Multi-label Classification mechanisms be labeled including images, sounds, music, and video. we present below some of those applications:

- *Emotions* [22]: The main goal is to automatically identify the emotions produced by different songs. A hundred songs from each one of seven music styles were taken as input. The songs were labeled by three experts, using the six main emotions of the Tellegen-Watson-Clark abstract emotional model.

- *Scene* [17] is related to image labeling, specifically to scene Classification. It is made up of 400 pictures for each main concept, beach, sunset, field, fall foliage, mountain, and urban. The images are transformed to the CIE Luv color space, known for being perceptually uniform, and latter segmented into 49 blocks, computing for each one of the values such as the mean and variance. The result is a vector of 294 real-value features in each instance [34].

- *Mediamill*: It was introduced in [39] as a challenge for video indexing. The goal was to discover what semantic concepts are associated with each entry, among a set of 101 different labels. Some of these concepts refer to environments, such as road, mountain, sky, or urban, others to physical objects, such as flag, tree, and aircraft.

- *Birds* [40]: this dataset aims to identify multiple birds species from acoustic recordings. After recording the audio, the researchers used 2D time-frequency segmentation approach to separate overlapping time since in each snippet, one to five different species appears. The generated dataset contains as features the statistical profile of each segment.

- *Flags* [41]: This dataset is considered as a toy dataset since it only has 194 instances with a set of 19 inputs features and 7 labels. The labels represent colors appear in the flag or the presence of a certain image or text and the input features describe land mass the country including its area, religion, population, etc. [34]

Genetics/Biology Multi-label Classification was greatly used in the literature for bioinformatic applications, we name below two datasets widely used in this field called Genbase and Yeast, the first concerns the Classification of genes in line with their functional expression, while, the second focused in predicting multiple functions of the protein.

- *Genbase* [42]: the dataset contains 662 proteins where 1185 motif patterns and profiles were used as input features, each feature indicates the presence or absence of each profile and motifs for each protein. Genbase contains 27 different protein classes, each protein associated with one or more label.
- *Yeast* [20]: concerns the prediction of the functional expression for a set of genes. The input features for each gene come from micro-array expression data, with a 103 real values vector per instance. Each gene can express more than one function at once, the dataset contains in total 14 functional classes.

How much the dataset is Multi-label? The selection of the best algorithm for learning needs a prior understanding of the inner traits of Multi-label data. The Multi-labelness of the dataset can be assessed using a specific characterization metrics proposed in the literature [13], we review in this section some of them:

- **Label Cardinality** (LCard [23]) : used to quantify the average number of the active labels that characterize each example of the database. High LCard denotes that data are truly Multi-label, while lower values state that most of samples have only one relevant label.

$$LC = \frac{1}{M}\sum_{i=1}^{M} \mid Y_i \mid \tag{1.1}$$

- **Label Density** (LDens [23]): takes into account the number of labels in the dataset, it is the average number of labels that characterize the examples when learning divided by the number of labels q. A high value indicates that the labels are present for each instance, and a low value indicates the existence of a small number of labels present for the majority of the instances.

$$LDens = \frac{1}{q}\frac{1}{M}\sum_{i=1}^{M} \mid Y_i \mid \tag{1.2}$$

- **Diversity** [23]: represents the total number of the labels in the dataset.
- **Distinct labelsets**(Distinct) [23] : calculates the number of the possible combinations of labels in the dataset, which is very important for Multi-label Transformation algorithms.

$$Distinct = \mid Y_i \subseteq L \mid \exists(x_i, Y_i) \in D \tag{1.3}$$

- **The Pmin**: is a measure that shows the percentage of instances in the dataset with only one active label [34].

$$P_{min} = \sum_{y' \in Y/|y'|=1} \frac{\mid y^{'} \mid}{M} \tag{1.4}$$

Table 1.3 presents the most common datasets used in the literature [43] with associated statistics. More datasets can be found in [34]

Table 1.3: Characteristics of some Multi-label datasets.

Dataset	Domain	Instances	Attributes	Labels	Cardinality	Density	Distinct	Reference
Medical	Text	978	1449	45	1.245	0.028	94	[35]
Bibtex	Text	7395	1836	159	2.402	0.015	2856	[33]
Enron	Text	1702	1001	53	3.378	0.064	753	[36]
Mediamill	Media	43907	120	101	4.376	0.043	6555	[39]
Emotions	Music	593	72	6	1.869	0.311	27	[22]
Scene	Media	2407	294	6	1.074	0.179	15	[17]
Genbase	Biology	662	1185	27	1.252	0.046	32	[42]
Yeast	Biology	2417	103	14	4.237	0.303	198	[20]
Flags	Image	194	19	7	3.392	0.485	54	[41]
Birds	Sound	645	260	19	1.014	0.053	133	[40]
Bookmarks	Text	87856	2150	208	2.028	0.010	18716	[33]
Delicious	Text	16105	500	983	19.017	0.019	15806	[37]
Reuters	Text	6000	500	103	1.462	0.014	811	[38]

3 Notations & Terminology

3.1 Terminology

This subsection present some terms used frequently in this manuscript:

- **ML**: Multi-label.
- **MLC**: Multilabel Classification.
- **MLL**: Multi-label Learning.
- **Label**: The output attribute associated with an instance.
- **Labelset**: A set of labels associated with an instance.
- **Instance**/**Sample**/**Example**: correspond to a row in a ML dataset, including its input attributes and associated labelset.
- **Attributes**/**Features**: Refer to the set of input attributes in the dataset, without including the output labelset.
- **Dataset**: A collection of instances, it is often presented as a matrix where the rows are instances and the columns present attributes.
- **Input space**: The space represented by the attributes used as predictors in a dataset.
- **Output space**: The space that represent labels (output attributes) in a Multi-label dataset.
- **Repository**: Usually it is a web site that provides free for resources researchers such as datasets and software. Refer to Table 1.5 for Multi-label resources widely used by researchers.
- **Preprocessing**: An important task of data mining that aims to clean data and select relevant input and output attributes for the learning task.

- **Bipartition**: Is the output generated by ML classifiers, that represent labels which are relevant or not to a given instance of learning.
- **Cardinality**: Average number of active labels per instance in a ML dataset.
- **Density**: Another metric derived from cardinality to measure the Multi-labelness of a ML dataset.
- **Diversity**: Label diversity refers to the total number of the labels in the dataset.
- **Resampling**: Technique used to create new subsets of learning from the original dataset.
- **Oversampling**: Resampling technique that produces additional data samples to the original dataset.
- **Bag**: Collection of data instances resulted from a Resampling strategy.
- **k-fold cross validation**: A strategy of Resampling used to estimate the skill of machine learning models on a new dataset. The parameter called K refers to the number of subsets of learning that a given dataset is to be split into.
- **Imbalance**: The dataset is said Imbalanced when there is a prominent inequality of its labels frequency.
- **Supervised**: Refer to learning from datasets fully annotated by the expert, that means supervised methods are guided by the labels associated with data samples.
- **Unsupervised** A learning task where the learner use only input features, without being guided by output space.
- **Clustering**: A strategy of learning that aims to discover the similarity between data points and to assemble them into groups.
- **Segmentation**: A process that aims to extract features from signal information such as images and audio.
- **Lazy method**: This expression is used for learning methods that do not generate a model apriori and defers the work until a new instance arrives.
- **Ensemble**: Set of learners that combines their predictions to outputs the final labelset.
- **Feature selection**: Technique to choose the most relevant attributes from a dataset.
- **PubMed**: A free resource developed and maintained by the National Center for Biotechnology Information (NCBI) at the National Library of Medicine.

3.2 Mathematical notations

Before introducing Multi-label approaches and Evaluation Metrics, let introduce in this section some formal definition and notations defined by Schapire and Singer [16], Zhang and Zhou [44] (Table 1.4).

According to G. Tsoumakas et al. [23], Multi-label Learning includes two main tasks: Multi-label Classification (MLC) and Label Ranking (LR). In this thesis, we focus basically

on MLC. The main task of *MLC* is to define function h that return a set of relevant labels Y, given an input space χ. While the LR task returns the ordering of all possible labels according to their relevance to a given instance x.

Symbol	*Definition*
χ	d-dimensional input space of numerical or categorical features f.
$L = \{\lambda_1, \lambda_2, \dots \lambda_q\}$	L :an output space of q labels, $q > 1$. Each subset of L is called labelset.
(x,Y)	$x = (x_1 \dots x_d) \in \chi$ is a d-dimensional instance which has a set of labels associated $Y \subseteq L$.
$Y = (y_1, y_2, \dots y_q) = \{0,1\}^q$	label associations represented as a q dimensional binary vector. Each element is 1 if the label is relevant and 0 otherwise.
$S = \{(x_i, Y_i)\} \,\vert 1 \leq i \leq M$	Multi-label training set with M instances.
Y_i	The sets of true labels for an instance.
$\hat{Y}_i$	The sets of predicted labels for an instance.
q, M, D	Number of *labels* and *instances* of the *Dataset* respectively.

Table 1.4: Mathematical Notations.

4 Learning From Multi-label Datasets

In the literature, many approaches were proposed to deal with Multi-label Classification, we review briefly in this section the most common approaches.

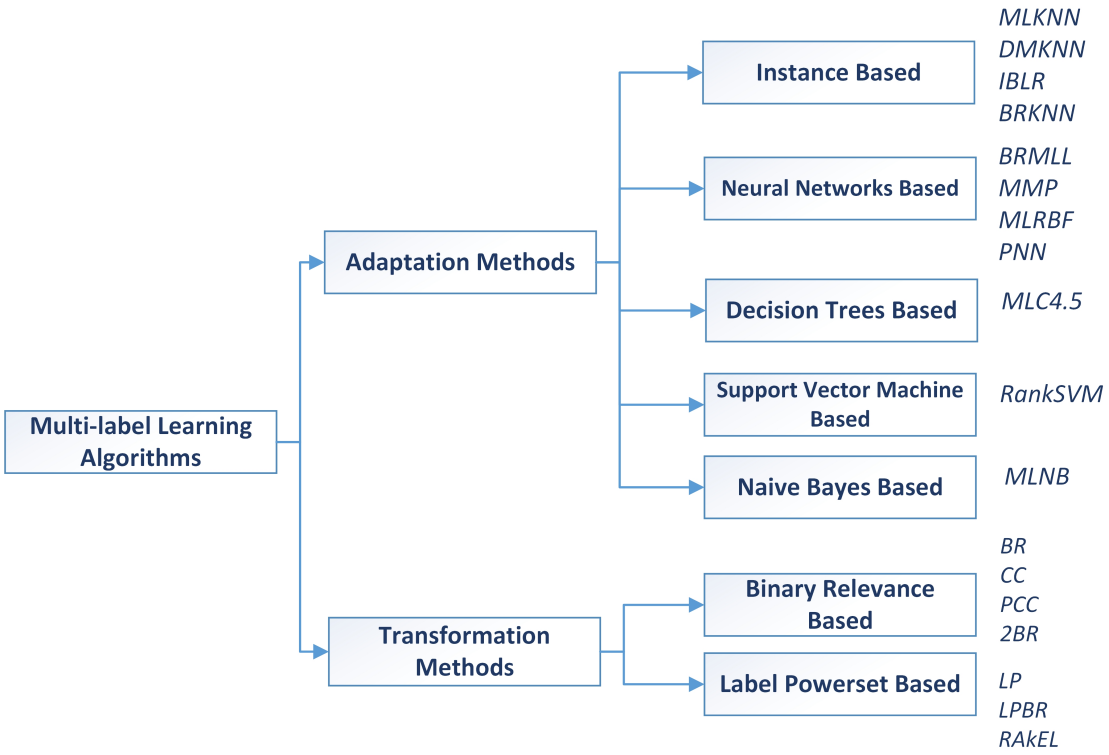

Figure 1.2: Categorization of Multi-label Algorithms.

Multi-label Classification methods can be categorized into two main groups [12]: Problem Transformation (PT) and Adaptation Algorithms (AA). The later tackle the learning problem by adapting popular learning approaches to deal with the Multi-label data directly. While the former, solve the problem by transforming it into other well-established learning scenarios such as Binary Classification (Binary Relevance [11]), Multi-class Classification (Label Powerset [11]). Figure 1.2 outline the main algorithms presented in this section.

4.1 Adaptation methods

By the apparition of the Multi-label problem and its need in many real-world applications, researchers tried firstly to adapt directly the existing algorithms to deal with the Multi-label Learning task, we present hereafter some of them categorized according to the strategy used during the learning process.

Instance Based

Multi-label K-Nearest-Neighbors (*MLKNN*): is an adaptation of the K-Nearest-Neighbors lazy learning algorithm (KNN) to Multi-label data [1]. *MLKNN* use the same basic principle of *KNN* for searching the K-Nearest-Neighbors, the difference is in the fact of using a Bayesian approach to specify relevant labelsets for a new instance based on its prior probability and posterior probability.

Dependent Multi-label K-Nearest-Neighbors (*DMl-KNN*) Younes et al. propose in [45] an algorithm derived from the *MLKNN* called: Dependent Multi-label k-Nearest-Neighbors (*DMl-KNN*) which takes into account the dependencies between the

different labels of the dataset. For each instance, the *DMl-KNN* identifies the k-Nearest-Neighbors and assigns a set of labels to that instance using the global maximum a posteriori (*MAP*).

Instance-Based Learning by Logistic Regression (*IBLR*) In [46], the authors combine Instance-Based learning and Logistic Regression (*IBLR*). (*IBLR*) consider the labels of Neighbors instances as an extra attributes in a Logistic Regression scheme.

Binary Relevance k-Nearest-Neighbors (*BRKNN*) *BRKNN* [47] is an adaptation of the *KNN* algorithm for Multi-label Classification that is conceptually equivalent to using the *KNN* as base classifier for the most popular Transformation method called Binary Relevance.

Multi-label Neural Networks Based

Multi-label Back Propagation (*BP-MLL*) is an adaptation of back propagation algorithm for Multi-label Learning [48]. The main modification is the introduction of a new error function that takes multiple labels into account.

Multi-label Multi-Class Perceptron (*MMP*) Crammer and Singer proposed a Multi-label Multi-Class Perceptron [49] for online topic Ranking from text documents. *MMP* use for each label a seperate perceptron as *BR* do, but the performance of the whole ensemble is considered to update each perceptron.

RBF Neural Networks for Multi-label Learning (ML-RBF) *ML-RBF* [50] was derived from the traditional Radial Basis Function (*RBF*). First, the algorithm conducts a clustering analysis on instances of each possible class in order to form the first layer of a *ML-RBF* Neural Network. The center of each cluster is considered as a prototype vector of a basis function. Then, the weight of the second layer is learned by minimizing a sum-of-squares error function.

Probabilistic Neural Network for Multi-label setting (PNN) In [51], a new version of a Probabilistic Neural Network (*PNN*) was proposed to tackle the Multi-label Classification problem. The adapted version of *PNN* was mainly proposed for executing the automatic Classification of economic activities.

Decision Tree Based

MLC4.5 Clare & king [10] adapt the C4.5 algorithm to Multi-label problem by modifying the entropy formula as follows:

$$Entropy(D) = -\sum_{j=1}^{q} p_j log(p_j) + (1 - p_j) log(1 - p_j)$$

Where D is the Multi-label dataset. q is the number of labels in D and p_j is the relative frequency of label j.

Predictive Clustering Tree (*PCT*) [52] is a Decision Tree organized as an hierarchy of clusters. The data is partitioned in a top-down strategy by minimizing the variance, the leaves represent the clusters and are labeled with its cluster's. In Multi-label Learning, the variance function is computed as the sum of the Gini Index of the variables from the target tuple, and the prototype function returns a vector with probabilities for each label [12].

Support Vector Machine Based

Rank-SVM Elisseeff and Weston present a Ranking algorithm for Multi-label Learning [20] based on *SVM* strategy known as *Rank-SVM*. It uses a set of q linear classifiers that minimize a cost function called Ranking Loss, defined as the average fraction of pairs of labels that were miss-ordered by the algorithm.

Calibrated Label Ranking (CLR) extends the Ranking by Pairwise Comparison (*RPC*) [53] that introduces a virtual label to separate between relevant and irrelevant labels (Calibration label), by this way *CLR* manages to perform Multi-label Classification.

Naive Bayes Based

Multi-label Naive Bayes (*MLNB*) Zhang et al. present a new method [9] called: Multi-label Naive Bayes (*MLNB*) which is an adaptation of the Bayesian network for Multi-label Classification. Variable selection mechanisms are presented to improve this algorithm. The first step is to use the Principal Components Analysis (*PCA*) to eliminate insignificant and redundant attributes. In the second part, this method uses a genetic algorithm to select the most appropriate attributes for label prediction.

4.2 Transformation methods

Transformation methods include algorithms that tackle Multi-label Learning by transforming it to one or more Single label Classification, it can be categorized into two main groups: Binary Relevance and Label Powerset In thispart, we introduce the main differences between the two strategies.

Binary Relevance Based

Binary Relevance (*BR*) this method [11] transforms the Multi-label Learning problem to q Binary Classification problems (q denoting the number of possible labels in the dataset) and learns for each label a separate classifier. It is simple and inexpensive in terms of computational time, but it is criticized because it does not take into account Label correlation.

Classifier Chain (*CC*) Based To overcome the disadvantages of *BR*, many variations of *BR* were proposed in the literature. As Classifier Chain [54], where the authors propose to use a Chain of classifiers to deal with Label dependence, by extending the feature space of each classifier with the outputs of all previous classifiers. The main limit of *CC* is that different orders of the chain can affect results. For that many extensions were proposed to deal with the ordering issue by using a set of *CC* chains with diverse orders as Ensemble Classifier Chains (*ECC*) [55], Probabilistic Classifier Chain (*PCC*) [56], and other algorithms based on genetic algorithm [57].

Probabilistic Classifier Chain (*PCC*) was introduced by Dembczynski et al. [56], which is an extension of the Classifier Chain (*CC*) approach [54]. The algorithm randomly determines the order of the Classifier Chain in the training phase. According to this order, for each instance of the Classification phase, it estimates the entire joint distribution of all possible label combinations and sought to maximize the posterior probability of the predicted label combination [58].

2BR (Meta-BR) The *2BR* algorithm also known as *Meta-BR* and stacked-BR [26], its main idea is to apply *BR* twice and use the outputs predictions of the first *BR* model (base level) as extra features for the second model (meta-level).

Label Powerset Based

Label Powerset (*LP*) [11] have been proposed to deal with the aforementioned problem of *BR*, where each new combination of labels existing in the learning set of size M is considered as a new class. Multi-label Learning problem is then transformed into a Single-label Classification problem where the number of classes is at most min (N, 2^q).

LPBR combine between *LP* and *BR* approaches [59]. The algorithm first cluster labels into several independent subsets based on chi^2 test. Then, it applies a Multi-label classifier for learning, and for labels predictions, it uses LP for dependent labels and *BR* otherwise.

RAndom k-labELsets (*RAkEL*) The *RAkEL* method [11] is one of the improvements of the *LP*, it constructs an ensemble of *LP* classifiers, each one is trained using a different small random subset of labels. The classes are then determined by a voting strategy using a threshold.

5 Multi-label Evaluation Measures

The Evaluation Measures of Multi-label algorithms differ from those used in other Classification tasks such as Single label Classification and Multi-class Classification. In [23], the metrics to evaluate Multi-label models are categorized into two main categories: Label-based metrics and Example-based metrics. Figure 1.3 summarize the Evaluations Metrics presented in this section.

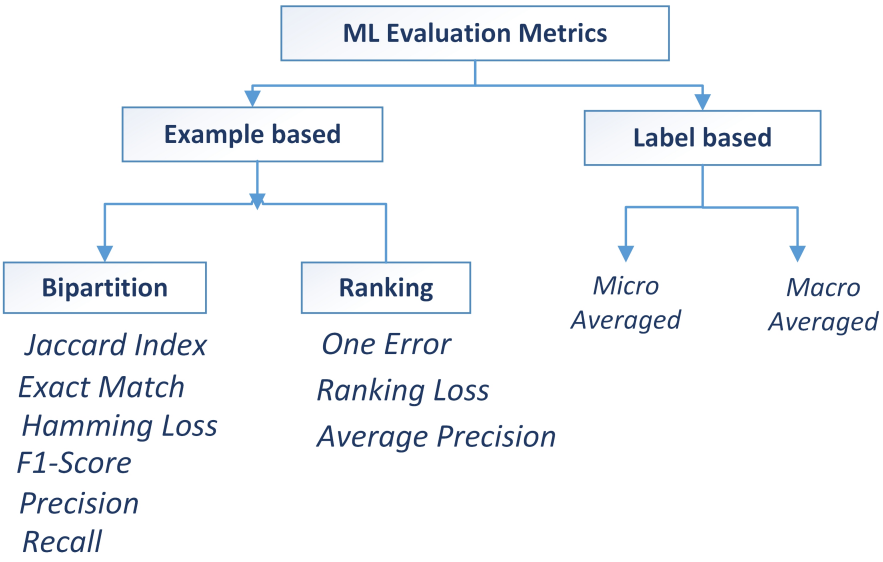

Figure 1.3: Categorization of Multi-label Evaluation Metrics.

5.1 Label-based strategy

Consists of computing a Single label metric for each label based on the number of True positives (Tp), True negatives (Tn), False positives (Fp) and False negatives (Fn) and then obtaining an average value. Note that there is two possible averaging strategy known as Micro-average and Macro-average approaches. The first one consider predictions of all instances together (aggregating the Tp, Tn, Fp, and Fn values of all classes) and then calculates the measure across all labels. While the second one computes one metric for each label and then the values are averaged over all the categories. The two strategies are defined as follows:

$$B_{Micro} = B(\sum_{i=1}^{q} Tp_i, \sum_{i=1}^{q} Fp_i, \sum_{i=1}^{q} Tn_i, \sum_{i=1}^{q} Fp_i) \tag{1.5}$$

$$B_{Macro} = \frac{1}{q}\sum_{i=1}^{q} B(Tp_i, Fp_i, Tn_i, Fn_i) \tag{1.6}$$

Example of Recall over the two approaches:

$$Recall_{Micro} = \frac{\sum_{i=1}^{q} Tp_i}{\sum_{i=1}^{q} Tp_i + \sum_{i=1}^{q} Fn_i} \tag{1.7}$$

$$Recall_{Macro} = \frac{1}{q}\sum_{i=1}^{q} \frac{Tp_i}{Tp_i + Fn_i} \tag{1.8}$$

5.2 Example-based strategy

Compute for each test example the Evaluation Metric and then averaged across the test set. We distinguish two main groups: Ranking and Bipartition Metrics. The most common Multi-label Evaluation Measures for each group are described in the following [60].

Let $T = (x_i, Y_i)|1 \leq i \leq t$ be a Multi-label test set with t instances. Given an instance, x, let $\hat{Y}_i = H(x_i)$ be the predicted labels by the hypothesis H and Y_i: the true labels. M and q are the number of instances and number of labels respectively. Let $z = (z_{\lambda_1}, z_{\lambda_2}, z_{\lambda_3} \dots z_{\lambda_q})$ be a vector with normalized output confidence scores in [0, 1]. For any predicate, π, $[[\pi]]$ returns 1 if the predicate is true and 0 otherwise.

Bipartitions Metrics

Accuracy (Jaccard index) this metric measures the degree of similarity between the set of predicted Y_i classes and the desired set of labels Y_i [13].

$$Accuracy = \frac{1}{t}\sum_{i=1}^{t}\frac{\mid Y_i \cap \hat{Y}_i \mid}{\mid Y_i \cup \hat{Y}_i \mid} \tag{1.9}$$

Exact Match (Subset Accuracy) is a very restrictive accuracy metric, considering a Classification as correct if all the labels predicted by a classifier for each example are corrects [12].

$$ExactMatch = \frac{1}{t}\sum_{i=1}^{t} I(\mid Y_i \mid = \mid \hat{Y}_i) \tag{1.10}$$

Hamming Loss takes into account prediction errors (incorrect label) and missing errors (label not predicted) [13]. Then, it evaluates the frequency that an example-label pair is misclassified, i.e. an example is associated to the wrong label or a label belonging to the instance is not predicted. The best performance is reached when Hamming Loss is equal to 0. The smaller the value of hamming loss is, the better is the performance.

$$HammingLoss = \frac{1}{t}\sum_{i=1}^{t} \mid Y_i \mid \triangle \mid \hat{Y}_i \mid) \tag{1.11}$$

where $\triangle$ is the symmetric difference between the real labels and the predicted labels, it corresponds to the XOR operation in Boolean logic. i.e: a class label belongs to the set of labels defined by $Y_i \triangle \hat{Y}_i$ if and only if that label occurs in either Y_i or $\hat{Y}_i$, but not in both sets.

F1-Score is the harmonic mean between Precision and Recall and is commonly used in Information Retrieval (IR).

$$F1 - Score = \frac{1}{t}\sum_{i=1}^{t}\frac{2 \mid Y_i \cap \hat{Y}_i \mid}{\mid Y_i \mid + \mid \hat{Y}_i \mid} \tag{1.12}$$

Precision is the proportion of labels correctly classified of the predicted positive labels, averaged over all instances.

$$Precision = \frac{1}{t}\sum_{i=1}^{t}\frac{\mid Y_i \cap \hat{Y}_i \mid}{\mid \hat{Y}_i \mid} \tag{1.13}$$

Recall is the fraction of predicted correct labels of the actual labels

$$Recall = \frac{1}{t}\sum_{i=1}^{t}\frac{\mid Y_i \cap \hat{Y}_i \mid}{\mid Y_i \mid} \tag{1.14}$$

Ranking Metrics

One Error evaluates how many times the top-ranked label is not in the set of proper labels of the instance. The performance is perfect when One Error is equal to 0; the smaller the value of One Error, the better the performance [1]. Note that, for Single-label Classification problems, the One Error is identical to ordinary Classification error.

$$OneError = \frac{1}{t}\sum_{i=1}^{t}[(argmin_{y\in Y} f(x_i, y) \notin Y_i] \tag{1.15}$$

Ranking Loss (Rloss) evaluates the average fraction of label pairs that are reversely ordered for the instance. The performance is perfect when Rloss is equal to 0; the smaller the value of Rloss, the better the performance [1].

$$Rloss = \frac{1}{t}\sum_{i=1}^{t}|(y^{'}, y^{''})|f(x_i, y^{'} \leq f(x_i, y^{''}), (y^{'}, y^{''}) \in Y_i \times \overline{Y_i})| \tag{1.16}$$

Average precision (Avgprec) determines for each label in an instance, the proportion of relevant labels that are ranked above it in the predicted Ranking. The goal is to know how many positions have to be checked, on average, before a non-relevant label is found [34]. It is originally used in IR systems to evaluate the document Ranking performance for query retrieval. The performance is perfect when it is equal to 1; the bigger the value of Average Precision, the better the performance.

$$Avgprec = \frac{1}{t}\sum_{i=1}^{t}\frac{1}{|Y_i|}\sum_{y\in Y_i}\frac{|y^{'}|rank_f(x_i, y^{'}) \leq rank_f(x_i, y), y^{'} \in Y_i|}{rank_f(x_i, y)} \tag{1.17}$$

6 Multi-label Toolboxes & Datasets Repository

Many Multi-label Toolboxes were proposed in the literature for learning from Multi-label dataset, Table 1.5 below reviews briefly some of them, note that each toolbox provides an associated data repository for ML datasets:

Table 1.5: Some Multi-label Toolboxes.

Toolbox	Language	Description	GUI	Reference
MULAN	Java	A Java library for MLL that provides a programming interface.	No	[43]
Meka	Java	A Multi-label/Multi-target extension to WEKA	YES	[61]
Scikit-Multilearn	Python	A Python library for performing Multi-label Classification	NO	[62]
Scikit-learn	Python	Simple and efficient tools for data mining and data analysis.	NO	[63]
RUMDR	R	R Ultimate Multi-label Datasets Repository	NO	[64]
LibSVM	Java, C++	software library for SVMs, it includes some Multi-label Classification algorithms	NO	[65]
KEEL	Java	Genral tool for data mining applications, it includes data repository for Multi-label.	YES	[66]
MLC	Matlab/Octave	a MATLAB/OCTAVE library for Multi-label Classification	NO	[67]

7 Conclusion

In our thesis, we are interested in *MLL* for medical aid diagnosis, the use of automated process can help greatly the doctors to decide quickly in many situations, for example, predicting the medical condition of a patient in the emergency based on a set of symptoms. A simplified example was introduced in [68], suppose that a patient arrived at the emergency with some symptoms and there are three possible diagnoses: Stroke, Drug overdose and

Epileptic seizure. The use of a Classification process that considers the possible diagnoses as outputs Y may greatly help the practitioner, we encode for example the pathologies as follows:

$$Y = \left\{ \begin{array}{c} \text{1 if Stroke;} \\ \text{2 if Drug overdose;} \\ \text{3 if Epileptic seizure.} \end{array} \right\}$$

As we presented previously, if all outputs are activated $Y = \{1, 2, 3\}$, the Classification task is called Multi-label. We focused on such task since it can deal very well with the problem of learning from medical data since the poly-pathology problem is often encountered by the doctors, where a patient may have multiple diseases at once.

The present chapter introduced the basic concepts of Classification task, especially learning from Multi-label data, the main goal was to provide the reader with the proper insight to take advantage of these machine learning task to deal with real-world data. It briefly reviews some Multi-label applications, with the main differences between several Classification tasks including Single label, Multi-class and Multi-label/ Multi-dimensional Classification. The main approaches from the literature were also reviewed with some Multi-label Toolboxes and data repository widely used. The next chapter gives an overview of the use of Ensemble Methods in MLL, the studied algorithm is called Multi-label K-Nearest-Neighbors.

Chapter 2

Ensemble Methods for Multi-label Classification

1 Abstract

The present chapter reviews the use of Ensemble Methods in the Multi-label framework. First, we briefly introduce the basic concepts of Ensemble Methods in Single-label Classification, then, we present a work that applies Ensemble Methods for a Multi-label algorithm. we aims to improve the performance of *Multi-label K-Nearest-Neighbors (MLKNN)* using Ensemble Methods (*Bagging* and *Boosting*), which adapts the *K-Nearest-Neighbors*(*KNN*) algorithm to Multi-label data. *Ensemble methods* use an ensemble of classifiers from either Transformation or Adaptation algorithms, it is divided into two main categories: *heterogeneous Ensemble Methods*, where the final decision is obtained by combining different algorithms responses on the same training set. While, *homogeneous Ensemble Methods* combine the prediction of the same algorithm using: adaptive (*Boosting*) or randomly (*Bagging* (Bootstrap and Aggregating)) strategies. In this work, we focus on the latter category since we aim to enhance the performance of *MLKNN* [1] that adapts the classical algorithm *KNN* to Multi-label data.

This work was published in Proceeding ICCDA '17 Proceedings of the International Conference on Compute and Data Analysis. Cite as: K. DOUIBI, N. SETTOUTI and MA. CHIKH. *The homogeneous Ensemble Methods* for MLKNN algorithm. Pages 197-201, Lakeland, FL, USA — May 19 - 23, 2017, ACM New York, NY, USA ©2017, ISBN: 978-1-4503-5241-3 doi:10.1145/3093241.3093262.

2 Introduction

Ensemble Methods were originally introduced to enhance the generalization ability of a Single classifier by building a set of base-models, also known as committee-based models and combine their predictions using a strategy of a vote. In the rest of this chapter, we denotes learning examples as $\{(x_1, y_1), \ldots, (x_M, y_M)\}$ for some unknown function y=f(x), x_i values are vectors of the form $(x_{i,1}, \ldots, x_{i,n})$, which are the features of x_i. Given a set S of training example, the algorithm outputs a classifier which is an hypothesis about the true function f. For a new instance x, the classifier predicts the corresponding output $\hat{Y}$, we denote classifiers by $h_1, h_2, \ldots, h_q$

The main idea of Ensemble Methods is that improved performance can be achieved based on the prediction of multiple models, instead of an isolated single prediction, it uses an ensemble of models (See Figure 2.2) called base classifiers, that can be one of the commonly used algorithms such as *Decision Tree (DT)*, *Neural Network (NN)*, *K-Nearest-Neighbors (KNN)* etc. The main idea of Ensemble Methods can be summarized into twofold: first training separate models, then, combine their decisions to give more accurate prediction. Ensemble Methods is known as homogeneous when the base classifier used is the same learner for all committee, otherwise, it is called heterogeneous Ensemble Methods.
The success of Ensemble Methods to improve the Accuracy does not relies only on the performance of base classifiers. Nevertheless, it depends also on the diversity of the learning set for each one.

Roughly speaking, two necessary and sufficient conditions for an Ensemble Methods to be more accurate than any of its individual members is if base classifiers are accurate and diverse [69]. A classifier is accurate if it has an error rate better than random guessing on a new item, and two base classifiers are diverse if they make different errors on new data points. A simple explanation of the importance of this condition was presented by Dieterich in [6], suppose that we have a committee of three classifiers: $\{h_1, h_2, h_3\}$ and consider new point x. If the three classifiers are not diverse (identical), then if the first classifier is wrong the two others are wrong. However, if the three errors made by h_1, h_2, h_3 are uncorrelated, then when the first is wrong, the two other classifiers may be correct so that a majority vote will correctly classify x.

Dietterich in [6] clarify the advantages of Ensemble Methods to improve the prediction of an isolated model in three views, let H be the hypothesis space, h the best hypothesis and h^* denotes the optimal hypothesis:

- Statistical: The role of a learner is to search the best hypothesis h from H. So, if the dataset used for learning is too small compared to the size of the hypothesis space, we talk about the statistical problem. Without sufficient data, the risk that a learner selects the wrong hypothesis with a poor generalization ability is higher. However, by constructing an ensemble committee based on accurate classifiers and then average their votes reduce this risk. See Figure 2.1 (top-left).

- Computational: At this level, the problem is not the small size of the dataset but it concerns the sensitivity of the algorithm to local optima while searching the best hypothesis in a large dataset. Constructing diverse classifiers provides many different starting points which may reduce this risk by combining all their predictions. As presented in Figure 2.1 (top-right).

- Representational: In some cases of machine learning applications, it is difficult to find an optimal classifier from the hypothesis space H, while the use of an ensemble of classifiers can approximate the best hypothesis by forming weighted sums of hypotheses drawn from H. See Figure 2.1 (bottom). The illustration of the third reason was explained by Dieterich as follows: if we consider for example Neural Networks and Decision Trees which are both flexible algorithms, that mean given a sufficient training data, they will explore the space of all possible classifiers. However, with a finite training sample, the hypotheses explored will be reduced, since they will stop searching when an hypothesis that fits the training data is found.

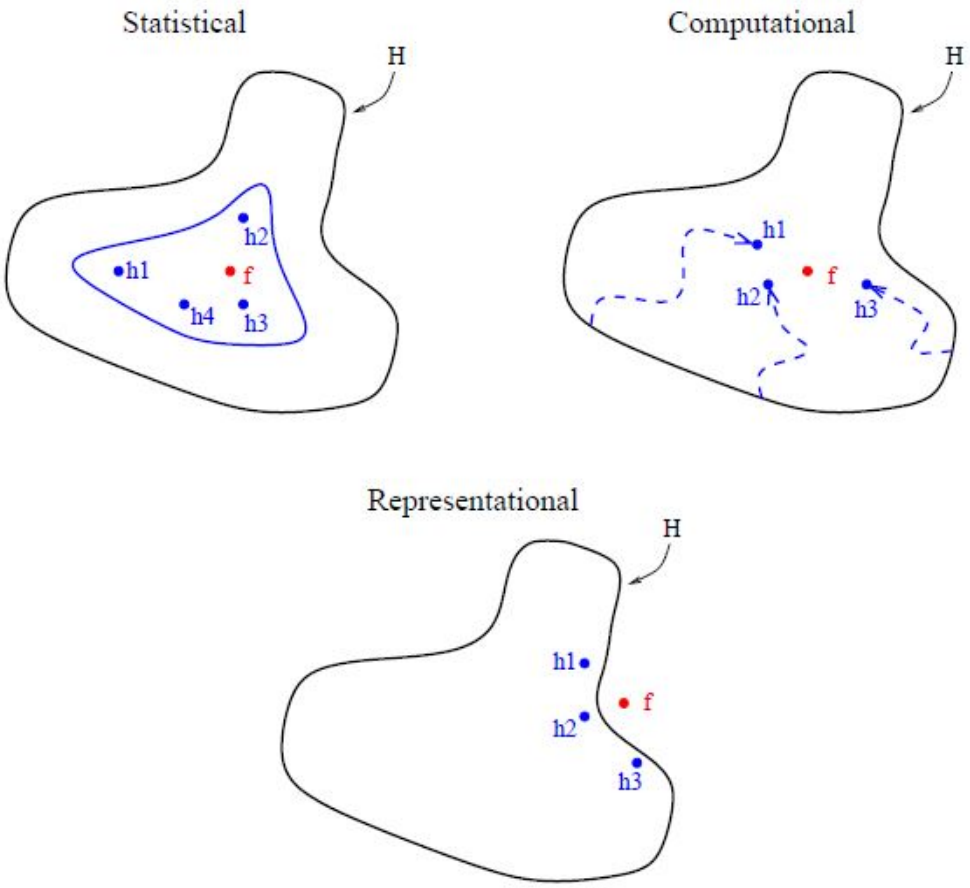

Figure 2.1: The Three Fundamental Views explaining the Advantages of Ensemble Methods to improve the prediction of an isolated classifier. The outer curve denotes the hypothesis space H, the inner curve denotes the set of hypotheses that all give good Accuracy on the training data. The point labeled f is the true hypothesis. We can see that by averaging the accurate hypotheses we can find a good approximation to f [6].

The strategy of learning of an Ensemble Methods can be divided into two architectures: *Parallel* and *Serial*, in the rest of the chapter we will detail two approaches called: *Bagging* [2] and *Boosting* [3], the first one (Refer to Section 2.1) illustrates the Parallel architecture while the second can be a good example for the Serial (Section 2.2). The main differences are summarized hereafter:

- Parallel: several base classifiers are trained in parallel, it is the most common strategy of Ensemble Methods used in the literature.
- Serial: known also as *Boosting* model where a series of base classifiers is used, and in each step an error function is used to improve the next base classifier according to the previous one.

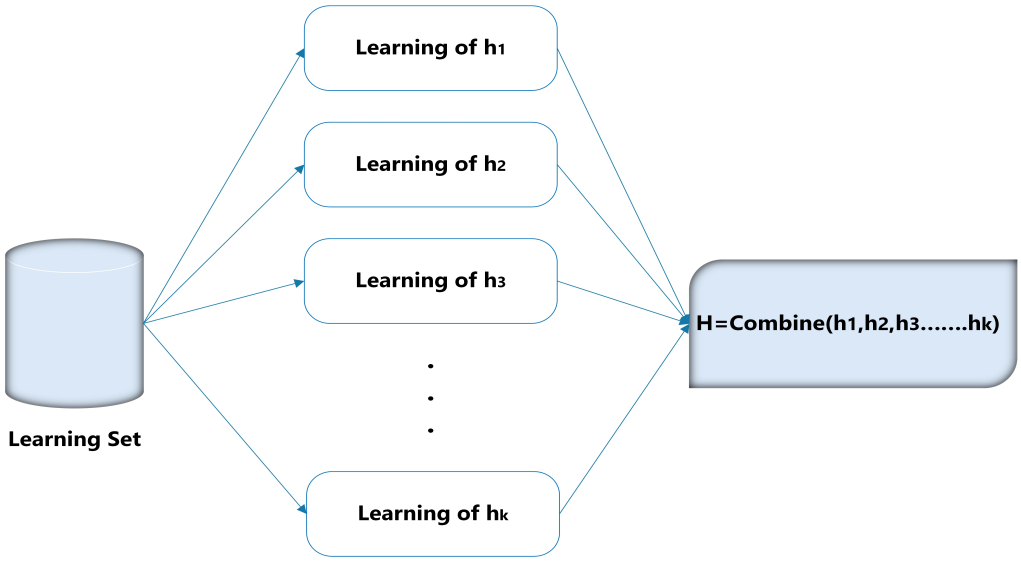

Figure 2.2: An example of Ensemble Methods Architecture. If each base classifier learns separately, the approach is called *Bagging*. However, If all base classifiers learn in a series, then the approach is called *Boosting*

At this level, we will discuss the idea of *how Ensemble Methods improves performance by consolidating multiple models instead of just using one isolated classifier? What are the different strategies of learning and sampling in this case?* And *how the final decision is computed from all predictions?*

In the literature, many approaches have been proposed to construct *Ensemble Methods*, one of the most useful technique is to manipulate the training examples to create diverse base learners. Constructing an ensemble method by manipulating the training examples means that the algorithm creates several subsets of training instances to learn many base classifiers, this technique works well for unstable learning algorithms whose output classifier undergoes major changes in response to small changes in the training data as *Decision Trees*, Neural Network etc. [6].

The two straightforward ways of manipulating the training set are *Bagging* [2] and *Boosting* [3], presented in details in the next section. Then, the rest of the chapter discusses the use of Ensemble Methods in the *Multi-label* framework, we present briefly some Related Work and the adaptation of *Bagging* and *Boosting* strategy to improve the performance of an adaptation of *K-Nearest-Neighbors (KNN)* algorithm to Multi-label called: *MLKNN* [1].

2.1 Bootstrap and aggregating (*Bagging*)

The *Bagging* is a strategy of learning introduced by Breiman (1996) [2], it includes two main steps, the first one called: *Bootstrapping*, that create new subsets of learning for each base classifier (bootstrap) in order to improve the individual prediction performance. Each bootstrap is generated by uniformly sampling with replacement M instances from the training dataset which are then used to train a separate base classifiers. Each bootstrap replicate contains on the average 63.2% of the original training set with several training examples appearing many times [2]. The second step is called: Aggregation that aims to output the final decision of the committee, that can be seen as a simple averaging process overall the base-classifiers.

The pseudo code 1 and the figure 2.3 below summarize the main steps of *Bagging* presented by Leo Breiman in [2]:

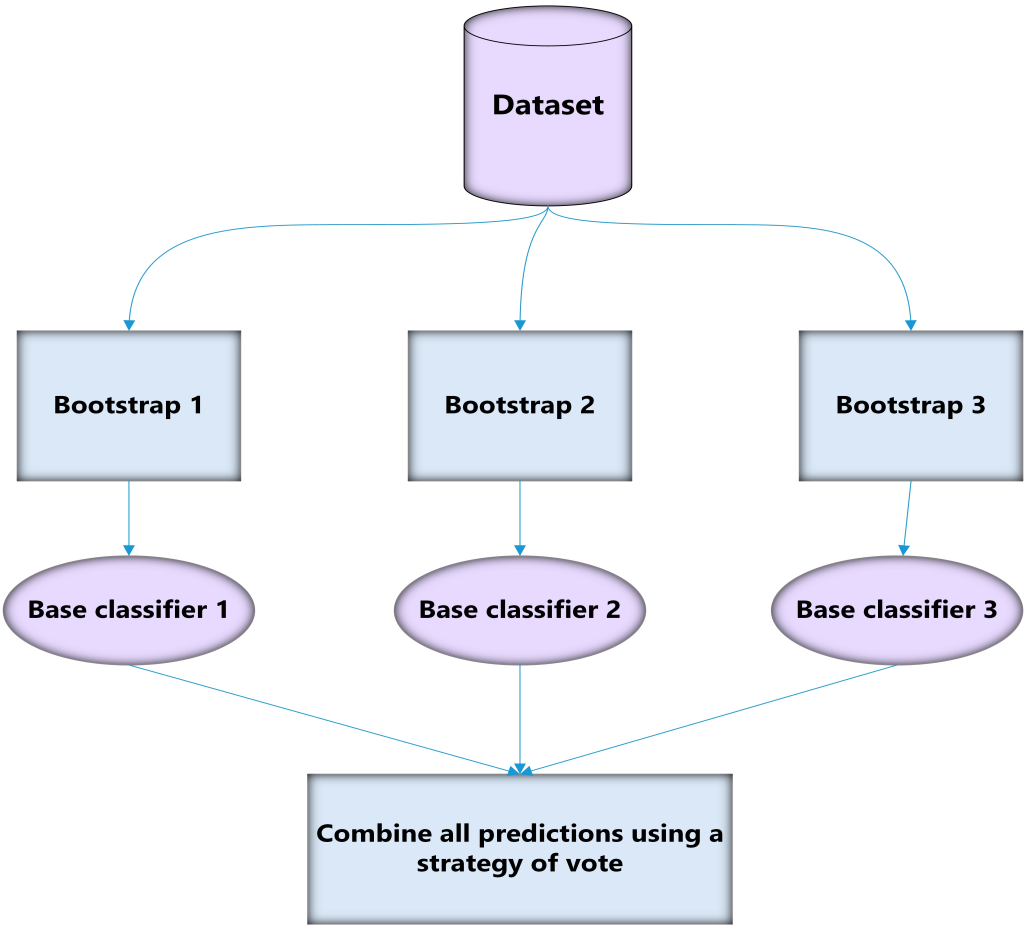

Figure 2.3: An illustration of *Bagging* Scheme using three classifiers in the Ensemble Methods.

Algorithm 1 *Bagging*

Input: Training set $'S'$

Output: Final prediction Generate k bootstraps by uniformly sampling with replacement $'M'$ instances from $'S'$

for Each bootstrap **do**

Learn a base classifier h

end for

The *Bagging* prediction is computed as:

$$\mathrm{H(x)}=\mathrm{sign}(\sum_{i=1}^{k} h_i(x))$$

Bootstrapping

Given a training set of M instances, according to Efron & Tibshiran [70], bootstrap is generated by sampling with replacement M times from the original training data. It exploits the independence by adding perturbation to enhance diversity within the committee,

each bootstrap sample contains only about 63% of unique instances, meanwhile, 37% [71] of instances will not appear in the bootstrap, called *Out-Of-Bag (OOB)* samples. They provide an effective way to estimate the generalization error of the base learner known as *OOB* estimation. Figure 2.5 and 2.4 presents an example of bootstrap and *Bagging* based Decision Trees respectively.

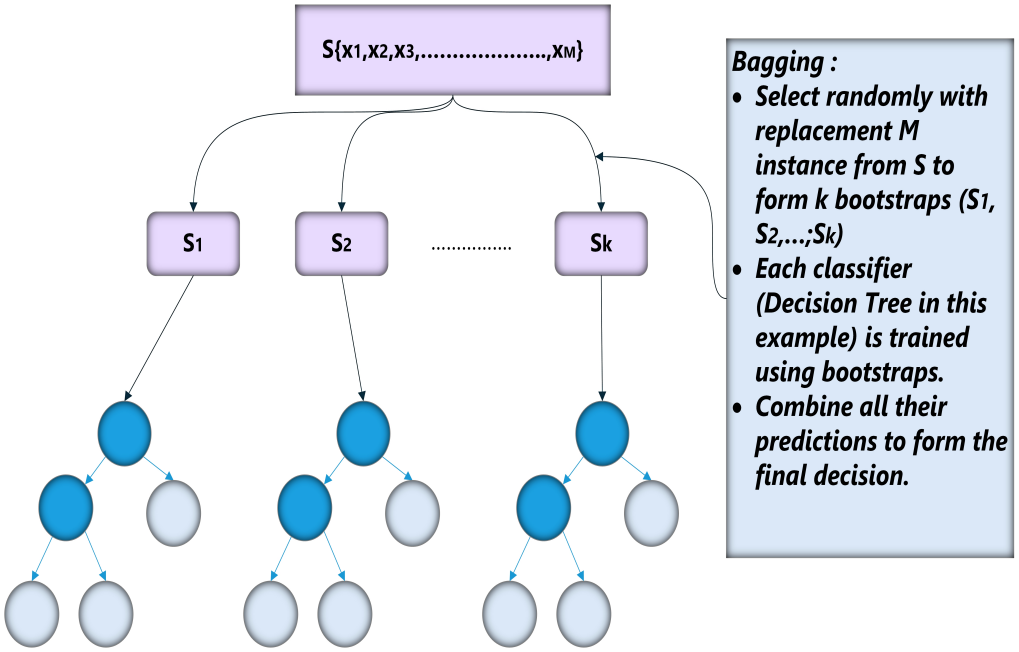

Figure 2.4: An example of *Bagging* based Decision Trees (CART)

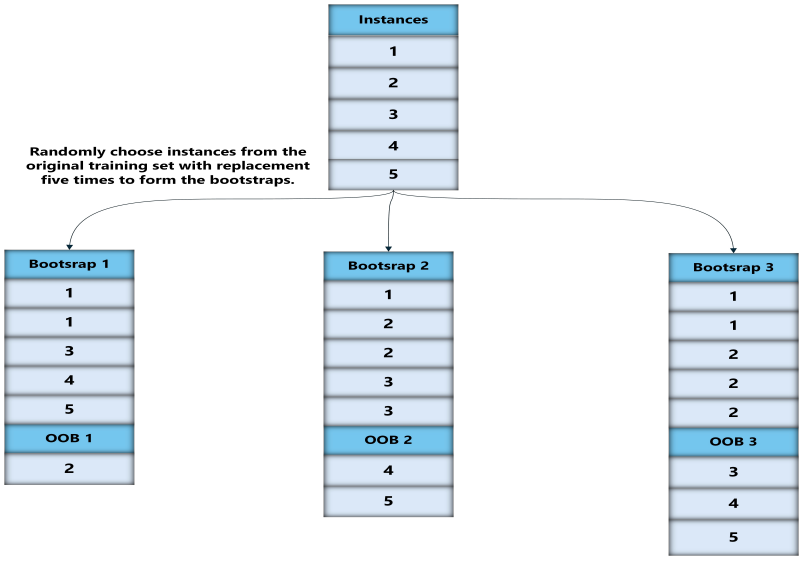

Figure 2.5: An example of Bootsrapping Sampling.

2.2 Boosting

Boosting is an ensemble method that allows, under certain conditions, to improve the performance of an algorithm by combining several **Weak learners** in order to form a

Strong learner. The idea of *Boosting* was proposed for the first time by Freund et al. in 1990 [3] to answer a question: *is it possible to make a weak algorithm as good as we want? It means a little better than hazard?* Shapire shows that a weak algorithm can always improve its performance by being trained in three training samples (*Boosting* by subsets see Figure 2.6)

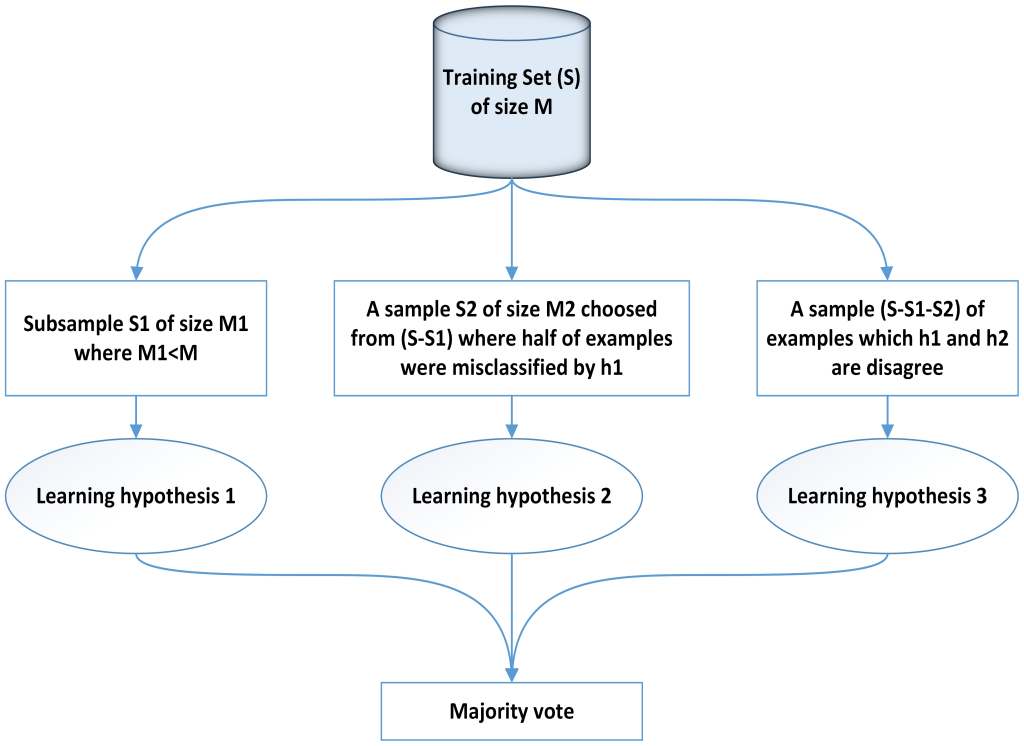

Figure 2.6: *Boosting by Subsets*

In 1996, Shapire and Freund proposed *AdaBoost* (Adaptive *Boosting*) [72], similarly to *Bagging*, *AdaBoost* manipulates the training examples to generate several hypotheses. It maintains a set of weights over the training examples, in each iteration l, the base classifier minimize the weighted error on the training set, and it returns an hypothesis h_l. The weighted error of h_l is used to update the weights in the training examples in order to place more weight on misclassified learning examples by h_l and less weights on examples that were correctly classified. The final classifier is constructed by a weighted vote of base classifiers as follows: $h_f(x) = \sum_l w_l h_l(x)$ where w_l is computed according to its Accuracy on the weighted training set that it was trained on [6].

In summary, the three basic idea [3] of *AdaBoost* are:

1. Using a specialized expert committee to make a decision.
2. Adaptive weighting of votes by using a multiplicative update technique.
3. Changing the distribution of samples available to train each expert, by overweighting misclassified examples in the preceding steps in order to force the learner to focus on the difficult examples of the sample learning.

3 Related Work of Multi-label Ensemble Methods

As we previously highlighted in the previous chapter, the algorithms proposed in the literature to deal with *Multi-label (ML)* problem are categorized into two main categories [12]: Transformation algorithms and Adaptation algorithms. The former transform the data to fit the algorithm however the later modify the algorithm to fit the data. The application of Ensemble Methods has attracted the attention of Multi-label community researchers for two major reasons: firstly to deal with problem of high complexity of *Label Powerset (LP)* approach [11], secondly to solve the problem of not considering Label dependence by the learner in the case of *Binary Relevance (BR)* models [54]. Note that the idea of Ensemble Methods is the same as proposed at the beginning of this chapter, thereby, the base classifier may be an *Adaptation* or *Transformation* method. We present briefly some of them thereafter:

Ensemble of Binary Relevance classifiers (EBR) *Binary relevance (BR)* [11] was first proposed to tackle *Multi-label Classification* problem by transforming it into a Binary Classification, then, EBR [54] was proposed where each base classifier is carried out on a random sub-sampling of the training dataset [73], each base classifier provide its Binary predictions, then the final prediction is computed by averaging all the predictions for each label using a threshold of 0.5.

Ensemble Classifier Chain (ECC) *Classifier Chain (CC)* was proposed in [54] to solve the problem of Label dependence of BR using a Chain of Classifiers, by extending the feature space of each classifier with the outputs of all previous classifiers. However, the order of classifiers in the chain was an issue, therefore the authors proposed the use of a committee of *CC* models to deal with that called *ECC* [55]. Each base classifier learns on random Chain orderings, using a random subset of training instances. The final prediction of *ECC* is computed using a majority vote strategy.

Ensemble of Pruned Sets (EPS) Another strategy of *Transformation* to deal with *ML* problem is *Label Powerset (LP)*, that transform it to a *Multi-class* problem. The main drawback of this approach is the high complexity computation especially for datasets with a large number of labels. Read et al. [74] proposed *EPS* that reduce the problem of complexity of *LP* by pruning samples with rare labelsets to allow the model focusing on the important label combinations. After that, *EPS* reintroduce the pruned examples along with subsets of their labelsets to compensate the information loss in the previous step. As *LP*, *PS* is not able to output the labelsets that are not in the training set, for that, the strategy of *Bagging* [2] was applied to learn a committee of *PS* learners. A strategy of a vote is then used to predict the final labels of an unseen instance.

Random Forest Predictive Clustering Tree (RFPCT) Random Forest [75] was adapted to ML framework by Kocev et al. in [76] called: *RFPCT*, it is an ensemble method based on *Predictive Clustering Tree (PCT)* algorithm [52]. *PCT* is top-down generated algorithm, i.e: at each node, data are partitioned into clusters in such a way that the intra-cluster variation is minimized. The result of the induction process is a Decision Tree in which each leaf contains the prototype of the instances belonging to that leaf [60]. The diversity among the base classifiers is obtained by using *Bagging* and selecting, at each node, the best feature from a random subset of the input attributes. The outputs are combined using a voting scheme.

Random Forest of Multi-label-C4.5 (RFMLC4.5) In [12], Random Forest was also adapted to Multi-label Learning using *MLC4.5* [10] as base model, called: *RFMLC4.5*. *MLC4.5* is a Decision Tree algorithm adapted to Multi-label framework by modifying the entropy to consider each label at leaves. While *RFMLC4.5* use the *Bagging* strategy to create a committee of *MLC4.5*, then their predictions are combined using a voting strategy over each label.

Multi-label Ensemble Learning (EnML) Chuan et al. in [77] develop Multi-label Ensemble Learning (*EnML*), that use a set of Multi-label classifiers to increase Classification accuracy. It is based on the evolutionary algorithms to optimize each classifier and find the optimal labels for each instance, a collection of predictors is used and the final decision is obtained by aggregating all their predictions.

Ensemble Label Powerset Pruned datasets Joint Decomposition (ELPPJD) Recently, Li et al. in [78] propose an ensemble Multi-label classifier based on Label Powerset strategy to resolve the multi-disease risk prediction based on physical examination records. They formulate the disease risk prediction into a Multi-label Classification problem, the proposed method is called *Ensemble Label Powerset Pruned datasets Joint Decomposition (ELPPJD)* and the dataset contains 110,300 records of anonymous examination records which include 62 examination items consisting of the basic physical examination items, Blood routine examination, Liver function test, as well as the diagnosis results marked by the physicians. In their experiment, they focus on 6 normal chronic diseases: Hypertension, Diabetes, Fatty liver, Cholecystitis, Heart disease, and Obesity.

Boosting for text categorization (Boostexter) Ensemble methods were widely used in text categorization field, which is one of the most important application domain of Multi-label Classification, where each textual document should be categorized according to its content. Schapire and Singer in [24] presented Boostexter: an algorithm of *Boosting* for text categorization for the *Multi-label* framework, and compare the results with other algorithms in the literature. The two versions of *AdaBoost* for *Multi-label* data are called *AdaBoost.MH* and *AdaBoost.MR* for Classification and Ranking respectively. In [79], the categorization of Chinese text was treated using the *Boosting* algorithm and the experimental evaluation was made on *WX95-96* dataset which contains chinese documents. The obtained results show the efficiency of the proposed algorithm compared to other algorithms of the literature as Bayesian network and *TFIDF/Rocchio algorithm*.

Finally, another work [73] study in depth the Ensemble *Multi-label Learning* in *Supervised* and *Semi-supervised* settings. The authors formulate the *Multi-label Learning* as an Ensemble learning problem to improve both Classification and feature selection tasks, they proposed a new semi-supervised Multi-label feature selection approach based on the ensemble paradigm.

One of the most studied *ML* algorithm in the literature is the adaptation of *K-Nearest-Neighbors* rule to *ML* framework, known as *MLKNN* [1]. In the present case study, we investigate the impact of applying two well known Ensemble Methods strategies called: *Bagging* [2] and *Boosting* [3] (presented previously) to improve its performance. First, we present in the next section more details about *Bagged MLKNN* and *AdaBoost MLKNN*. Then, we discuss the results found on the most common datasets using four *Multi-label* Evaluation Measures (previously defined in Chapter 1 1) such as Accuracy [13], Fmeasure [13], Subset Accuracy [12] and Hamming Loss [13].

4 Proposed Approach

4.1 *Bagged MLKNN*

Initially, the *Bagging* was introduced as a basic rule, a Decision Tree. However, the pattern is very general and can be applied to other basic rules such as Nearest Neighbour rule. The Bagged K-Nearest-Neighbors (*Bagged KNN*) was studied by Biau and Devroye (2010) in [80], in a Regression framework for *Single-label* Classification.

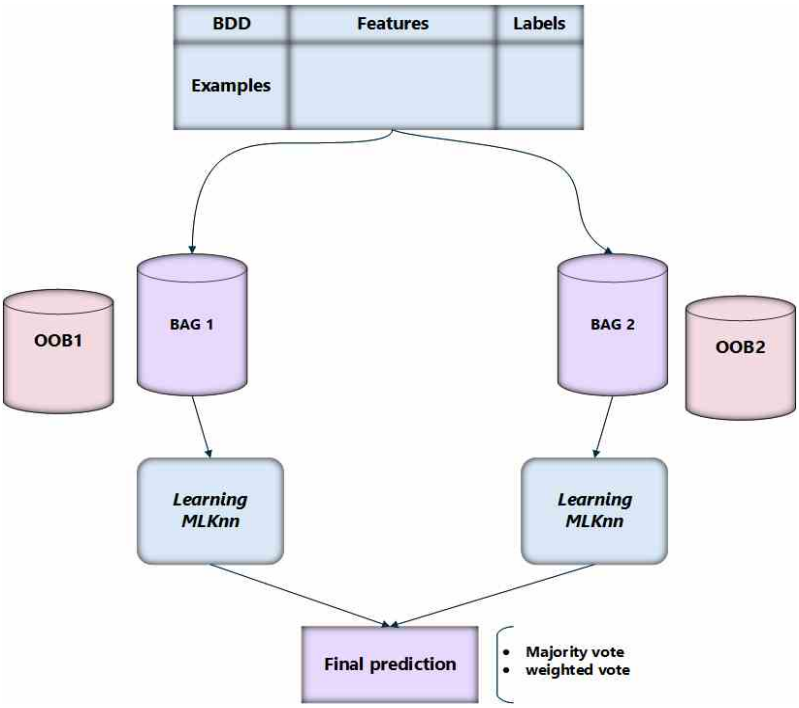

Figure 2.7: Classification process with *Bagged MLKNN*

This work establishes the consistency of the *Bagged KNN* method and illustrates perfectly the benefits of *Ensemble Methods*: starting from a basic rule relatively poor (Rule of *KNN* that is not consistent), the *Bagging* turn it into a very good rule with asymptotic properties (consistency and optimal convergence rate). In addition, *Bagging* has many advantages [2] because it is simple to implement, easily adapts to any learning methods and reduces the impact of the choice of the training set on Classification results. For that, we have implemented *Bagged MLKNN*.

MLKNN algorithm

The *Multi-label K-Nearest-Neighbors (MLKNN)* [1] adapts the *KNN* to deal with *Multi-label* data. *KNN* is one of the most useful approaches of Classification that classifies a new instance x based on its neighborhood, known also as instance-based learning [81]. It computes a distance between its features and all instances in the dataset. The class of x is predicted based on the classes of the closer instances. *KNN* is known as a lazy method since it does not create any model a priori, only when a new sample arrives the classifier does some work [34]. The key idea of the adaptation of *KNN* to the *Multi-label* scenario is the use of a posteriori principle to determine the labelset of unseen instance [1] (Refer to Figure 2.8), hence the *MLKNN* is not lazy because it starts building an initial model a priori based on two pieces of information [34]:

- Compute the a priori probabilities for each label, that consist of the number of times each label appears in the dataset divided by the total number of instances.
- Compute the Conditional probabilities for each label as the proportion of instances with the considered label whose *K-Nearest-Neighbors*, also have the same label.

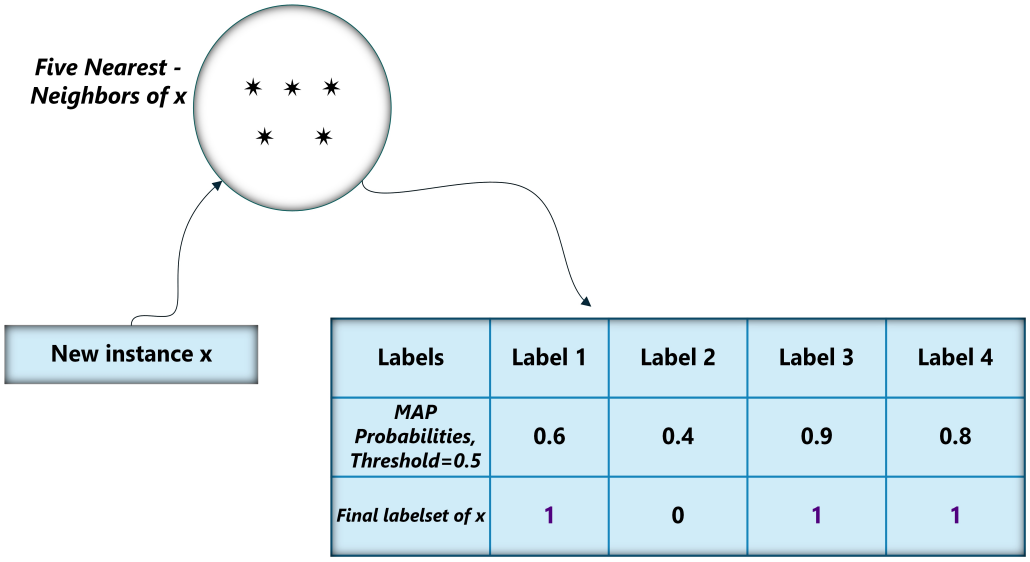

Figure 2.8: A Simple illustration of *MLKNN* principle to classify a new instance.

Note that *MLKNN* does not take dependencies among labels into account since both probabilities evoked previously are computed individually for each label. The advantage of considering potential Label dependencies during the *Classification* process will be discussed in depth in Chapter 4.

The steps described below summarize the Classification process for a new instance using *MLKNN* [1], the *MATLAB* implementation of the algorithm is freely available in [1]

- The algorithm use the euclidean distance (by default) to measure the similarity between the new instance and all the examples of the dataset to find its *K-Nearest-Neighbors*.
- Then, the presence of each label in the neighborhood is used as evidence to compute *Maximum A Posteriori (MAP)* probabilities from the Conditional ones obtained before [34].
- The labelset of the new sample is generated from the *MAP* probabilities. The probability itself is provided as a confidence level for each label, thus making possible to also generate a label Ranking [34].

According to [1], the results obtained by applying *MLKNN* show their simplicity and ease. Therefore, the use of several *MLKNN* simultaneously for the prediction of labels necessarily improve the performance of an individual classifier. The replacement of the conventional ordinary voting 'majority vote' by the 'weighted vote' is also proposed. This is justified by the fact that the traditional vote depends on the choice of a majority of

[1]http://cse.seu.edu.cn/people/zhangml/Resources.htm#codes.

classifiers that gives the same class for a given instance but without considering the performance of each classifier. However, the weighted vote is used to validate and to weight these classifiers, by taking into account the capacity of each classifier individually which gives more efficient results. Figure 2.7 show the process of *Bagged MLKNN*.

4.2 *AdaBoost MLKNN*

Adaptation of *AdaBoost* to *MLKNN*

The *AdaBoost MLKNN* use a set of *MLKNN* classifiers sequentially in order to optimize the values of Prior, PriorN, Cond and CondN. Those parameters are used by the Bayesian inference to adapt the *KNN* to the Multi-label data (See [1]) and are used as probabilities to determine the label of an example considering its neighbourhood. The *AdaBoost MLKNN*, in each iteration overweight examples that were misclassified by the hypothesis of the previous step, in order to force the learner of the next step focuses on them, and try to find the optimal values of variables that will be used in the Classification process (See the pseudo code 2).

Algorithm 2 Pseudo code of *AdaBoost MLKNN*

Input: Train data, Train target, Number of *K-Nearest-Neighbors* (Num), Smoothing parameter (Smooth)

Output: Output, Prelabels

1: Training *MLKNN* [1]
[Prior, PriorN, Cond, CondN]= MLKNN_train(Train data, Train target,Num, Smooth)

2: Find the optimal values of: Prior_optim, PriorN_optim, Cond_optim, CondN_optim using the *AdaBoost* Principle [72]

3: Testing MLKNN [1]
[Output, Prelabels]= MLKNN_test(Test data, Test target, Num, Prior_optim, PriorN_optim, Cond_optim, CondN_optim)

5 Experimental Setup

The effectiveness of the proposed algorithms (*Bagged MLKNN*, *AdaBoost MLKNN*) is evaluated through five datasets commonly used, obtained from Mulan library [43]: *Genbase*, *Yeast*, *Scene*, *Medical* and *Emotions*, Table 2.1 describes each dataset (Refer to Chapter one for more details). The Evaluation Metrics used are *Accuracy* [13], *Fmeasure* [13], *Subset Accuracy* [12] and *Hamming Loss* [13]. The algorithms were evaluated using 5-fold cross-validation by varying the number of *Nearest-Neighbors* K= 4, 8, 12, 16, 20 and 24 while keeping the same default value of the smoothing parameter=1.

Table 2.1: Characteristics of the used datasets.

Dataset	Domain	Instances	Attributes	Labels	Cardinality	Density	Distinct
Genbase	Biology	662	1185	27	1.252	0.046	32
Scene	Media	2407	294	6	1.074	0.179	15
Yeast	Biology	2417	103	14	4.237	0.303	198
Medical	Text	978	1449	45	1.245	0.028	94
Emotions	Music	593	72	6	1.869	0.311	27

5.1 Results

By varying the number of Nearest-Neighbors K, we notice that this parameter does not influence significantly on the performance of the algorithm *MLKNN*. This finding reaffirms the results of Zhang and Zhou in their Article [1], for that reason we present in Table 2.2 for each dataset, the results obtained with k=4 (The value following ±gives the standard deviation). In addition, in our experiments, we tested the algorithms with a different number of the hypothesis (*MLKNN*) varying from 5 to 120 hypothesis. We present in the same Table 2.2 the results obtained using 100 hypotheses because we noticed a stabilization and improvement of the results from this value, the final decision of all hypothesis was calculated by the weighted vote. The winning results are marked with bold font.

Table 2.2: Experimental Results

Dataset	Evaluation Metrics	Algorithm		
		MLKNN	Bagged MLKNN	AdaBoost MLKNN
Genbase	Accuracy	99,66 ±0	**99,72** ±0,0003	97,36 ±0,1989
	Subset Accuracy	0,93 ±0	**0,94** ±0	0,49 ±0,0098
	Fmeasure	96,3 ±0	**96,94** ±0,0385	60,08 ±0,0811
	Hamming Loss	**0** ±0	**0** ±0	0,03 ±0
Yeast	Accuracy	80,4 ±0	**80,61** ±0,0104	75,89 ±0,0843
	Subset Accuracy	0,17 ±0	**0,19** ±0	0,11 ±0,0007
	Fmeasure	63,17 ±0	**64,26** ±0,0639	57,09 ±0,0189
	Hamming Loss	0,2 ±0	**0,19** ±0	0,24 ±0
Scene	Accuracy	90,82 ±0	**91,17** ±0,0172	84,14 ±0,0107
	Subset Accuracy	**0,62** ±0	0,61 ±0	0,29 ±0,0057
	Fmeasure	71,64 ±0	**72,36** ±0	43,76 ±0,0686
	Hamming Loss	**0,09** ±0	**0,09** ±0	0,16±0,0001
Medical	Accuracy	98,27 ±0	**98,38** ±0.0014	97,38 ±0,0598
	Subset Accuracy	0,45 ±0	**0,47** ±0,0001	0,18 ±0,0018
	Fmeasure	62,7 ±0	**64,70** ±0,0080	32,74 ±0,0047
	Hamming Loss	**0,02** ±0	**0,02** ±0	0,03 ±0
Emotions	Accuracy	73,19 ±0	**73,93** ±0,0017	65,9 ±0,0963
	Subset Accuracy	0,12 ±0	**0,15**±0,0001	0,08 ±0,0009
	Fmeasure	46,72 ±0	**48,96** ±0,0083	39,2 ±0,0033
	Hamming Loss	0,27 ±0	**0,26** ±0	0,34 ±0

5.2 Discussion

The experimental results of all Evaluation Metrics (Table 2.2) demonstrate that *Bagged MLKNN* outperforms the original algorithm (*MLKNN*) for all datasets. However, the use of *AdaBoost* gives poor results because its performance depends on data and a weak learner, therefore with a weak classifier, *AdaBoost* converges too slowly and need a great number of hypothesis to attain the performance obtained by the *Bagged MLKNN* with just 100 hypotheses as represented in Table 2.2.

We discuss in the following, some results of the application of the *Bagged MLKNN*:

The results obtained by this algorithm are very encouraging for all datasets, for *Genbase*, we achieved 99% of *Accuracy*, with 94% of *Subset Accuracy* indicating that, if we have 100 genes we can identify correctly for 94 genes all their genomics functions simultaneously. The 96% of the *Fmeasure* indicates a good compromise between the two measures precision and recall.

In this context, the precision means to identify correctly the relevant and irrelevant labels, while a high value of recall means that the classifier considers all the labels as relevant which is not necessarily true. The *Fmeasure* combines perfectly between those two measures. The *Hamming Loss* metric is equal to zero, which is very interesting since the low values of this metric indicate a good Classification and the number of misclassified labels pairs is reduced.

On the medical text, we note that the *Bagged MLKNN* gives the best results and it improves the Classification performance efficiently with 99,72% of Accuracy and 47% of Subset Accuracy. Similarly, for the other metrics we note an improvement of the results comparing with *MLKNN*, the Fmeasure increases from 62% using *MLKNN* to 64% by the application of multiple *MLKNN* (*Bagged MLKNN*), 2% of improvement is interesting in this field because it means that the proposed method decreases the Recall, so we have more significant labels (Precision increases). For the Hamming loss, the low value indicates that the classifier is able to assign the correct labels to the learning examples with a minimum of error.

6 Conclusion and Further Work

The present case study investigate the impact of using *homogeneous Ensemble Methods* (*Bagging* and *Boosting*) for *MLKNN* algorithm [1] that adapts *KNN* to *Multi-label* data. The experiments were carried out on five small to large datasets from a different domain (biology, image, medical text, music). The results are very competitive compared with those obtained by the original algorithm and show that the use of several *MLKNN* simultaneously for the prediction of labels improve the performance of the individual classifier. Also, the replacement of the conventional ordinary voting 'majority vote' by the 'weighted vote' for the aggregation of the results gives more efficient results because it takes into consideration the performance of each classifier individually in the weighting vote process.

Many points are likely to be considered as part of future work, such as the performance improvement of *Bagged MLKNN* by using variable selection methods, in order to identify the relevant variables for each label of each dataset. Such a task is really important for the medical applications since the physician prefer to have a good compromise between the performance and interpretability of the medical aid diagnosis systems. Unfortunately, few medical datasets are publicly available online to conduct experimental studies, for that we propose in the next chapter a new medical dataset of Ambulatory Blood Pressure Monitoring (*ABPM*) collected recently [4], we intend to evaluate the proposed approach of the present chapter on this new dataset in the near future.

Chapter 3

Multi-label Classification for Ambulatory Blood Pressure Monitoring.

1 Abstract

Ambulatory Blood Pressure Monitoring (*ABPM*) involves measuring Blood Pressure by means of a tensiometer carried by the patient for a duration of 24 hours, it currently occupies a central place in the diagnosis and follow-up of hypertensive patients, it provides crucial information which allows to make a specific diagnosis and adapt therapeutic attitude accordingly. The traditional analysis process suffers from different problems: it requires a lot of time and expertise, and several calculations should be performed manually by the expert, who is generally very busy. In this work, we attempt to improve the analysis of *ABPM* data using Multi-label Classification methods, where a record is associated with more than one label (class) at the same time. Seven algorithms are experimentally compared on a new Multi-label *ABPM-dataset*. Experiments are conducted on 270 hypertensive patient records characterized by 40 attributes and associated with six labels. Results show that the Multi-label modeling of *ABPM* data helps to investigate label dependencies and provide interesting insights, which can be integrated into the *ABPM* devices to dispense automatically detailed reports with possible future complications. This chapter presents in-depth this work with a detailed explanation about the medical context and reviews some related work from literature, it describes also the experimental results and discussions with study limits and Further Research.

This work was published in Australasian Physical & Engineering Sciences in Medecine. Cite as: Douibi, K., Settouti, N., Chikh, M. A., Read, J., & Benabid, M. M. (2018). An analysis of Ambulatory Blood Pressure Monitoring using Multi-label Classification. Australasian physical & engineering sciences in medicine, 1-17. DOI 10.1007/s13246-018-0713-0.

The dataset used in this work is availible online in `http://dx.doi.org/10.17632/y4dh3b3tfx.1`. Cite as: Douibi, K., Benabid, M. M., Settouti, N., Chikh, M. A.(2017), "Data for: An analysis of Ambulatory Blood Pressure Monitoring (ABPM).", Mendeley Data, v1.

2 Introduction

Nowadays, machine learning is used daily by researchers in all companies to track customers behaviors for example to sell more products or to check if a received email is spam or not, not just that, but also to recommend you some products based on your navigation on the net, although, this huge trend of technology is not used only in such business applications but it is also very useful in medicine and health-care industry.

Recently, machine learning solutions are moving medical applications to a whole new level, in medical imaging, it helps to extract more accurate data from images and provides better and efficient interpretations aiming to detect tumors. Another healthcare application is called predictive medicine, where the main goal is to improve both the quality of patient care and working conditions of the practitioners by providing better information, the collected patient' s data is used to make predictions based on symptoms associations and their correlations, historical patient diseases and familiar antecedents.

An interesting example was provided by health catalyst in [82] show the importance for a doctor to have more useful information about his patient in real time to aid in clinical decision making: having easy access to the Blood Pressure and other vital signs when I see my patient is routine and expected. Imagine how much more useful it would be if I was also shown my patient's risk for stroke, coronary artery disease, and kidney failure based on the last 50 Blood Pressure readings, lab test results, race, gender, family history, socioeconomic status, and latest clinical trial data. The value of machine learning in healthcare is its ability to process huge datasets beyond the scope of human capability, and then reliably convert analysis of that data into clinical insights that aid physicians in planning and providing care, ultimately leading to better outcomes, lower costs of care, and increased patient satisfaction.

Another recent work [83] discuss the importance of Artificial Intelligence (AI) and Big data in public health, the authors highlight the significance and potential impacts of using such approaches in healthcare applications, in addition, they discussed the challenges that doctors will face in future since the role of the human specialist may move to case management and intelligent machines will screen, detect and make diagnosis!

Similarly, another recent work [84] report the unintended consequences of machine learning in medicine, such as: reducing the skills of physicians, especially in medical imaging applications where many of modern algorithms outperform or perform as well as human observers, which could cause a subtle loss of self-confidence and affect the willingness of a physician to provide a definitive interpretation or diagnosis. The authors discuss also the disadvantage of focusing on data and demising of context, because many of clinical elements are not included in the recorded data that may lead to partial or misleading interpretations of medical decision support system outputs. They discussed also the need to produce interpretable systems or in other words to open the machine learning black box and offer to the physician richer interactive visualization tools to explore the implications of potential variables in the decision provided.

Finally, we show in numbers the use of machine learning in medicine for the last ten years. Figure 3.1 presents the number of publications about machine learning in medicine in the PubMed database using the following query: Search (machine learning) AND Medicine).

One of the main fields that was studied in the literature by applying machine learning

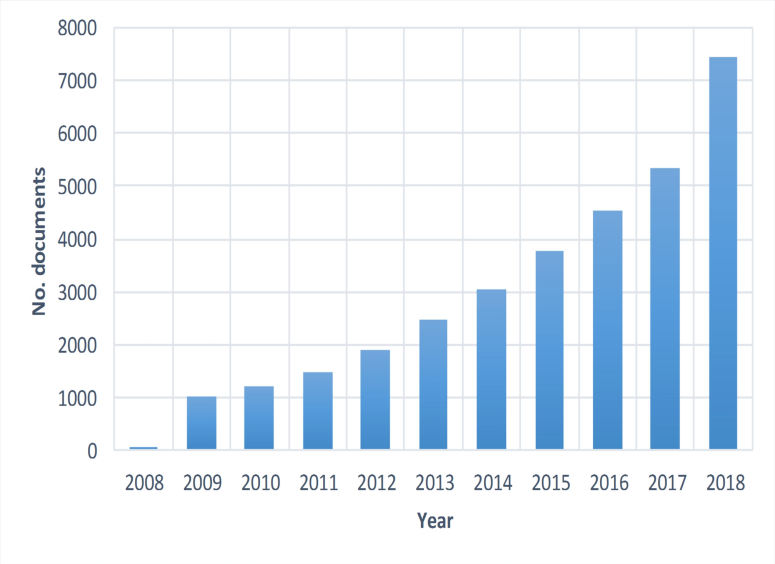

Figure 3.1: Number of PubMed publications for the query: (machine learning) and Medicine) from 2008 to 2018.

algorithms is the Cardiovascular field, we present in the following a case study that concerns the analysis of Ambulatory Blood Pressure Monitoring (*ABPM*) using *Multi-label* Classification.

High Blood Pressure (*HBP*) is one of the main risk factors for the Cardiovascular disease. it is essentially a silent disease, the suffering of target organs is the cause of clinical manifestations of this pathology (in particular: brain, eyes, heart, and kidney). *HBP* is directly responsible for about half of coronary artery disease and about two-thirds of cerebrovascular problems [85]. There are over 972 million hypertension patients worldwide [86]. The diagnosis of the *HBP* consists of the identification of pathological values and the exploration of a possible secondary origin [87].

The information given by the Blood Pressure (*BP*) clinical measurement remains limited. The values are unique and can be influenced by stress, and it does not reflect the conditions of life in which the person lives. In other words, this measurement is not enough to make an accurate diagnosis [88].

Different techniques are used to measure the *BP* throughout the day:

1. Self-measurement at home with approved electronic devices.
2. Clinical measurement.
3. *A*mbulatory *B*lood *P*ressure *M*onitoring (*ABPM*) is carried out using a small automated digital Blood Pressure monitor for 24 hours, attached to a belt around the waist of the patient and connected to a cuff around the upper arm.

Many clinical studies have shown that the risk of the Cardiovascular complications and renal prognosis are better correlated with *ABPM* than with clinical measurement [89]. For that reason, we are interested in such data, which provides a complete overview either of

the patient's condition and the major future complications. It gives an idea about the general health state of the patient by determining if there is an association between High Blood Pressure, Diabetes, Cardiovascular risk and other pathologies. This case of poly-pathologies is often encountered by practitioners, known in machine learning as *Multi-label* Classification.

Multi-label Classification is a hot topic in machine learning (See Chapter 1 for more details), it is required in many real-world applications such as text categorization, bioinformatics, medical, image, and videos etc. where data instances are usually associated with multiple labels simultaneously. For example, in medical the field, a patient can be affected by multiple illnesses (labels). According to [38] a *Multi-label* problem has the following settings:

- The set of labels is predefined, meaningful and human-interpretable.
- The number of labels is limited in scope and not greater than the number of attributes.
- Each training example is associated with several labels of the labelset.
- Labels may be correlated.
- Data may be unbalanced.

The study we carried out shows that *ABPM* data [4] are *Multi-label* and fulfill the above criteria, where each record is associated with six labels (statistics on those data are presented in Section 5.1). The problem we addressed here, is on one hand to find a model H that maps input x (attribute vector of *ABPM*) to a vector output $y = [y_1, y_2 \dots y_q]$, q=6. On the other hand to study possible correlation between labels and features using the Decision Trees algorithm. For that, *Multi-label* Learning approaches are required (Sections 5.2 and 5.3 goes into details about this topic).

The Figure 3.2 shows the general process of *ABPM*, divided into two main steps:

- **Step 1**: the device should be placed on the patient for 24 hours. The portable monitor is worn on a belt connected to a standard cuff on the upper arm (Figure 3.2, Step1). When complete, it should be connected to a computer to recover the *ABPM* record.
- **Step 2**: the expert should perform a manual analysis to detect the six outputs: *Validity*, *Circadian rhythm*, *Pulse Pressure*, *Blood Pressure Variability*, *Blood Pressure Load* and the *Morning Surge*. The six outputs will be presented in details in Section 4.

ABPM data provide crucial information for the diagnosis and follow-up of the hypertensive patients. However, a lot of doctors and nurses are unfamiliar with the technique and find some difficulties to analyze the generated data. In addition, the traditional analysis process (Figure 3.2) suffers from different problems: it requires a lot of time and expertise, and several calculations should be performed manually by the expert, who is generally very busy. In this work, we propose the use of machine learning algorithms to improve the *ABPM* device and make it intelligent, in order to:

- Facilitate the interpretation of such data.

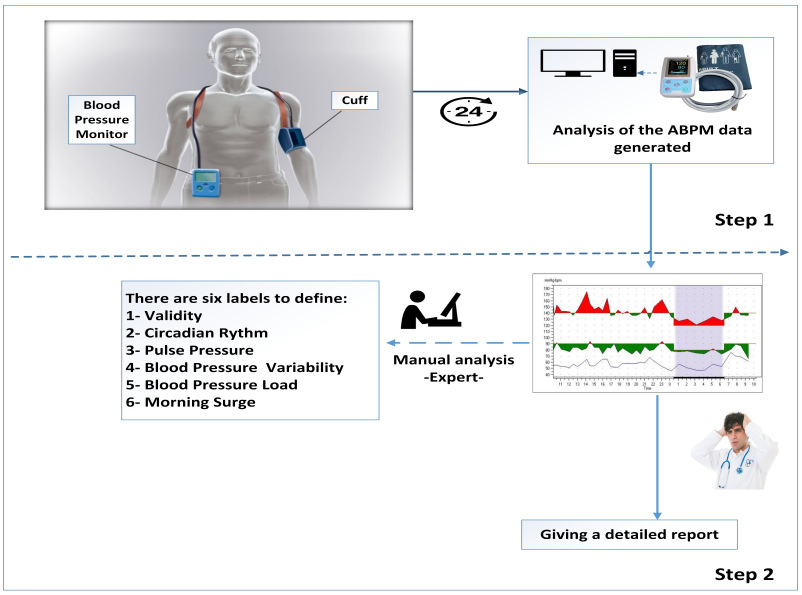

Figure 3.2: General process of the *ABPM*

- Help doctors in the diagnosis process.
- Avoid manual analysis of complex values.
- Define the implicit correlations between the different attributes and labels, this need was expressed by the cardiologist since it helps greatly to prescribe the appropriate treatment and to have an idea about possible complications.

The major steps of the proposed approach are summarized in Figure 3.3.

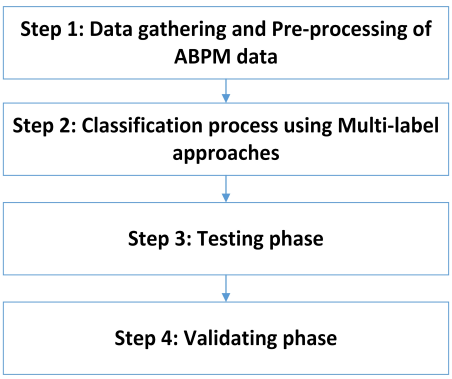

Figure 3.3: The main steps of the Intelligent Analysis of *ABPM* data

The major aim of this article is twofold:

- The first contribution is a new *Multi-label* dataset with 40 *ABPM* features for 270 numeric patient records categorized into one or more out of 6 labels (Figure 3.3, Step 1). The dataset is released to the public [4], in order to allow comparative experiments by other researchers and especially medical researchers while the publicly available Multi-label medical datasets are very rare. Section 5.1 gives more information about the dataset.

- The second contribution is the intelligent analysis of *ABPM* records using Multi-label Classification algorithms (Figure 3.3, Step 2). The medical diagnostic process supported by such techniques constitutes a modern and a useful tool for medical aid decision, allowing the expert to analyze *ABPM* record more quickly and efficiently, and to have an idea about label dependencies. We will give further details in the proposed approach Section 5.

3 Related Work

Ambulatory Blood Pressure Monitoring is the record of the Blood Pressure for a duration of 24 hours, the interpretation of such data gives a relevant information about the general health status of the patient, for that the indication of this examination is increasing nowadays. Unfortunately, in the literature there are just a few theoretical works interested to analyze the generated data; this is due to their complexity or the difficulties of understanding their medical context. Also, Single-label Classification approaches may not be effective for improving the interpretation process, as *ABPM* records are associated with several labels simultaneously. We propose for the first time an automated analysis of the *ABPM* data, considering this constraint. The present study is based on Multi-label Classification algorithms.

We start by presenting some works on those data, the first category makes statistical studies for the control of the hypertensive patients as well as the definition of different associations with other pathologies, and the second category regroups some studies about the use of intelligent systems for the control of hypertensive patients conditions. Last part of this section presents some recent works for the same topic.

3.1 Medical statistical studies

In [90], a study was performed on 206 patients (83 men and 123 women), known and treated to be hypertensive Diabetics, in order to evaluate the Blood Pressure (*BP*) profile using the *ABPM*. With the same objective, another work [91] concluded that High Blood Pressure (*HBP*) is frequently associated with diabetes, leading to an increase in the Cardiovascular risks. 80% of people with diabetes and *HBP* die from Cardiovascular disease. They showed that *ABPM* makes possible a better evaluation of the true level of *BP* under the usual conditions of life. It thus appears to be an effective means of assessing the *BP* balance in diabetics, due to the large number of measures, which is not clinically feasible [92].

3.2 Intelligent system for *ABPM*

Few works in the literature studied the use of intelligent systems for the control of hypertensive patients. Guo-Zheng et al. [93], present a collection of a Multi-label clinical data, and aims to study the impact of the traditional Chinese medicine, in the treatment based on Syndrome differentiation [1] of the patients.

[1]Syndrome differentiation in Traditional Chinese Medicine (*TCM*) [94] is the comprehensive analysis of clinical information gained by the four main diagnostic *TCM* procedures: observation, listening, questioning, and pulse analysis, and it is used to guide the choice of treatment either by acupuncture and/or *TCM* herbal formulae

Traditionally, the differentiation of the syndrome is performed by a practitioner who should have a strong theoretical foundation and experience. In this work, the authors automate this task by applying a *Multi-label* Classification of clinical hypertension data, from 908 patients with 129 symptoms considered as attributes, and 12 syndromes that represent the labels of the dataset. The proposed approach is called: *BrSmoteSVM*, first, it transforms the problem of Multi-label Classification into one or several Binary Classification problems (*Binary Relevance*) and then, applies the *SMOTE* (synthetic minority over-sampling) algorithm to reduce the influence of the problem of imbalanced data, and at the end, the *SVM* classifier is used to get the predicted label set. The obtained results were compared to other algorithms of the literature, such as: *MLKNN* (Multi-label K-Nearest-Neighbors), *BRKNN* (A lazy *Multi-label* Classification method based on the *KNN*), *RAkEL* (RAndom k-labELsets) and *IBLR* (Instance-Based learning and Logistic Regression).

In [95], Copetti et al. propose a theoretical study of an intelligent surveillance system, for hypertensive patients at home. The considered variables are: the *physiological variables*, collected from sensors linked to the body of the patient (using the *ABPM* for the *BP* and other sensors for the heart rate), the *environment variables* are collected by sensors in the house to detect the ambient temperature, humidity, light, the cigarette smoke, etc. and finally the *behavior variables*, to detect the patient's activities at home as asleep, sport, swimming, etc. The data collection is done in real time and so a filtering and extraction module is used to eliminate noise. Also, the fuzzy logic was introduced in the knowledge representation phase in the form of decision rules, in order to explain the relationships between the patient's activities and the variation of systolic and diastolic *BP* figures.

In the field of data mining, the analysis of *ABPM* data can be regarded as a Multi-label Classification problem, which can be solved with a specific data mining and machine learning techniques. In traditional Classification problems, one record would be only classified to one category (i.e. label) which is called Single-label Classification. While in the medical context, each *ABPM* record may have more than one labels. In this work, we study the use of Multi-label Classification algorithms for the automatic analysis of *ABPM* data, which provides better control of *HBP* than clinical data (presented in the first study [93]).

3.3 Other recents work on ABPM

Recently, the authors in [96] investigated the association between ambulatory BP parameters and Cardiovascular risk in elderly treated hypertensive patients with normal achieved ambulatory BP. They concluded that in elderly treated hypertensive patients with normal achieved ambulatory BP, dippers with high Morning Surge (MS) and non-dippers are at increased Cardiovascular risks.

Another recent work [97] studied the benefits of using ABPM after one Cardiovascular (CV) event in the prediction of a second one. The goal was to compare ABPM values after a first CV event between patients with (2EV) and without (1EV) a second CV event and to evaluate if ABPM has a role in secondary prediction. The conducted study was on 187 hypertensive patients with ABPM after a first CV event. ABPM data in 2EV vs 1EV were compared and they concluded that in patients with previous Cardiovascular events, higher values of 24H, daytime and night-time Systolic Blood Pressure (SBP) are more predictive of Cardiovascular events. In their 2EV population, a 24H SBP higher than 124 mmHg is more predictive of secondary events.

In [98], authors update the scientific statement on ABPM (2008) in children and adolescents, by highlighting the use of ABPM in the pediatric population with additional data and also presents a revised interpretation schema. They discuss in depth the importance of ABPM in determining the CV risk and also the risk for target-organ damage. In addition, they explain the usefulness of ABPM to classify BP including white coat hypertension, masked hypertension, pre-hypertension and progression to sustained (ambulatory) hypertension and finally they discuss some methods for performance of ABPM.

Other researchers worked on the application of machine learning methods to support medical decisions in the prognosis of fatal Cardiovascular diseases [99] based on ABPM data. They evaluated the performance of their method by determining new prognostic thresholds for well-known and potential Cardiovascular risk factors that are used to support medical decisions in the prognosis of fatal Cardiovascular diseases. The dataset used in their study was composed of 551 observations with seven attributes.

4 Ambulatory Blood Pressure Monitoring (*ABPM*)

ABPM is the measure of the *BP* by means of a tensiometer carried by the patient for a duration of 24 hours [87]. It is programmed to automatically measure *BP* every fifteen to twenty minutes a day and every thirty minutes during sleep. This is carried out using an automated small digital Blood Pressure monitor attached to a belt around the waist of the patient, and connected to a cuff around the upper arm. It takes into account the activity of the person and shows how the pressure can evolve according to the circumstances.

The *logbook* should be used by the patient to note sleeping hours and special events (unusual exertions, bedtime, meals, time of anti-hypertensive drugs, etc.) during the *ABPM*. It is a very useful document for analyzing allowing the practitioner to ignore some peaks of the non-pathological *BP*, caused by a physical activity of the patient requiring an effort. Figure 3.4 shows an example of a *logbook*.

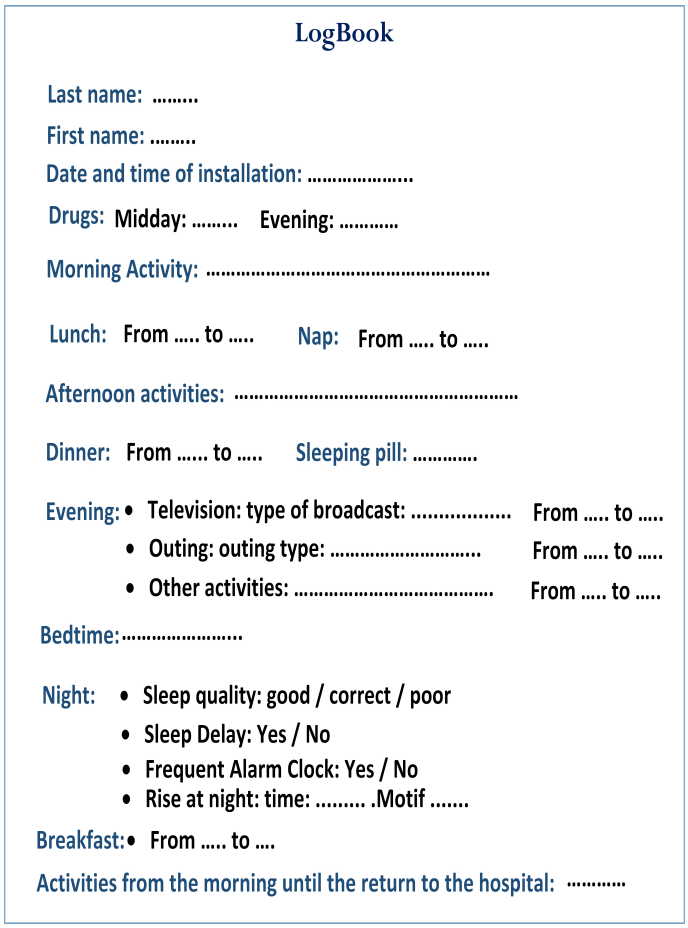

Figure 3.4: Example of the logbook

The *ABPM* used must be reliable and validated according to the international standardized procedures [100]. A list of different approved devices is available on the sites [101] [2] [3].

4.1 *ABPM* indications

ABPM is very useful in both diagnosing and managing the changes in the patient's *BP* during his usual activities. Based on the various clinical practices guidelines, and the expert opinions [87], the following major indications are acknowledged. *ABPM* is used to:

- Establish a diagnosis of High Blood Pressure..
- Identify patients who have higher *BP* readings in the clinic (known as *White Coat Effect*).
- Affirm the *Resistant Hypertension*: occurs when the *BP* is higher than 140/90mmHg despite a triple therapy (including a diuretic).
- Detect the *Masked Hypertension* [102]: defined as a clinical condition in which a patient's office *BP* level is <140/90 mm Hg but the ambulatory or home *BP* readings are in the hypertensive range.

[2] `www.swisshypertension.ch`
[3] `www.dablEducational.com`

- Search for a *HBP* in a pregnant woman.
- Well control the *BP* in the case of Parkinson's disease, diabetes, heart failure.
- Decide if the *BP* medication is required, especially in the case of an important Blood Pressure variability.
- Diagnosis the hypotension.
- Identify Nocturnal hypertension.

The *ABPM* is a non-invasive Blood Pressure measuring technique with several advantages: it provides more representative information than the conventional *BP* measurement, about both the Cardiovascular risks and the risk of the target organs damage. It also allows the measurement of *BP* during specific moments of the day and during sleep, and thus to identify patients whose Blood Pressure does not reduce at night-time (non-dippers), who are probably at high risk. And finally, the information provided constitutes a valuable diagnostic, therapeutic and prognostic aid.

4.2 The medical *ABPM* analysis

ABPM is being used increasingly in clinical practice. In recognition of this, the British Hypertension Society has published recommendations for the use and interpretation of *ABPM* [103], also the European Society of Hypertension has published recommendations on *BP* measuring devices, including devices for the *ABPM* [104]. We detail hereafter, the parameters considered in the interpretation of *ABPM* [7].

Validity The *ABPM* is considered reliable and interpretable when the following conditions are reunited: [7]

- Two-thirds of the *BP* measures are valid and, equally distributed over periods of awakening and sleep.
- *ABPM* recording must be spread over the 24 hours without interruption of the recording more than 2 hours consecutive.
- The quality of the sleep must be at least satisfactory, in order to be able to correctly interpret the nocturnal *BP*. In fact, if sleep is shortened, agitated or of poor quality, nocturnal *BP* values may be "abnormally" high.

Circadian Rhythm (*BP Profile*) The *ABPM* evaluates the Blood Pressure during sleep, the later decreasing physiologically from 10 % to 20 % at night (*dipper profile*), if it exceeds 20 % we speak of *extreme dipper*. However, in some patients (called *non-dippers*) this reduction is not sufficient, or even absent from 0 % to 10 % [87]. The *BP* may even be higher at night (inversion of the circadian rhythm called *reverse dipper*). The absence of nocturnal *BP* decrease is correlated with increased Cardiovascular risks, left ventricular hypertrophy, cerebral gaps detected by *MRI*, micro-albuminuria in Diabetic patients and decline in Renal function in chronic kidney disease.

Blood Pressure Variability (BPV) is usually considered as pathological if it exceeds 12-15 mm Hg [7]. It may be a reflection of a senescence of the baroreflex [4], usually involved in the adaptation of the heart rate and *BP* during activities and changes in position. The *BPV* may occur during the *ABPM* in the following situations:

[4] reflexes to control Blood Pressure

- Elderly patients.
- Diabetes.
- Primary neurological disorders (dysautonomia, Parkinson's disease).
- Drugs (antidepressants, anti-Parkinson's, etc.)

Pulse Pressure (PP) Defined as the difference between *Systolic Blood Pressure* (SBP) and *Diastolic Blood Pressure* (*DBP*), The *PP* is a good predictor of Cardiovascular events in the elderly, especially compared to the *SBP*, this value is considered suspicious when it exceeds 30 mm Hg and clearly pathological when it exceeds 50-55 mm Hg in hypertensive patients over 50 years [7].

Blood Pressure Load (BPL) The *BPL* is defined as the percentage of the *SBP* and the *DBP* values exceeding the upper limit of the standard, without taking into account the amplitude of this excess; It also appears to be a determinant of the Cardiovascular risks, especially when it exceeds 40%.

Morning Surge (MS) The *MS* is defined as the increase from the lowest *BP* during sleep to the average of the first two hours after waking, it is also a predictor of the occurrence of multiple pathologies, especially when it is greater than 55 mm Hg in the elderly. In [105], authors show that elderly subjects with Morning Surge had a higher baseline prevalence of multiple cerebral infarcts and a higher incidence of stroke compared with those whose *BP* did not show morning surge. Morning Surge has also been shown to be associated with left ventricular hypertrophy in people with untreated hypertensive [106]. The Table 3.1 summarizes the normal values of these five parameters (*Circadian Rhythm*, *BPV*, *PP*, *BPL*, *MS*). Note that in our study, we code all labels values as 0 if normal and as 1 if pathological. Except for MS and Validity, where 1 means presence of Morning Surge and the record is valid, respectively, and 0 otherwise.

Table 3.1: The Normal values of the five parameters: *Circadian Rhythm*, *BPV*, *PP*, *BPL*, *MS* [7]. The symbol ↓ means the decrease of this parameter between 10% and 20% indicates a normal *Circadian rhythm*. Zero values indicates normal case, otherwise pathological.

Parameters (labels)	Normal values (0)
Circadian rhythm: Dipper	↓ [10%-20%]
Blood Pressure Variability: BPV	<12-15 mm Hg
Pulse Pressure: PP	<55-60 mm Hg
Blood Pressure Load: BPL	<40 mm Hg
Morning Surge: MS	<55 mm Hg

5 The Proposed Approach

ABPM has become an indispensable technique for the management of hypertension. On the one hand, it gives more representatives information than conventional *BP* measurement. On the other hand, it gives an idea about the risk of developing pathologies that are associated (poly-pathologies), where several pathologies are present in the same patient. In machine learning, this Classification task is called *Multi-label* Classification, where

each instance is associated with several labels simultaneously, using powerful Classification algorithms in order to help the expert in the diagnostic process.

The study we carried out aims to facilitate the analysis of *ABPM* data using Multi-label algorithms. The summary of the proposed approach is shown in Figure 3.5, and detailed as follows:

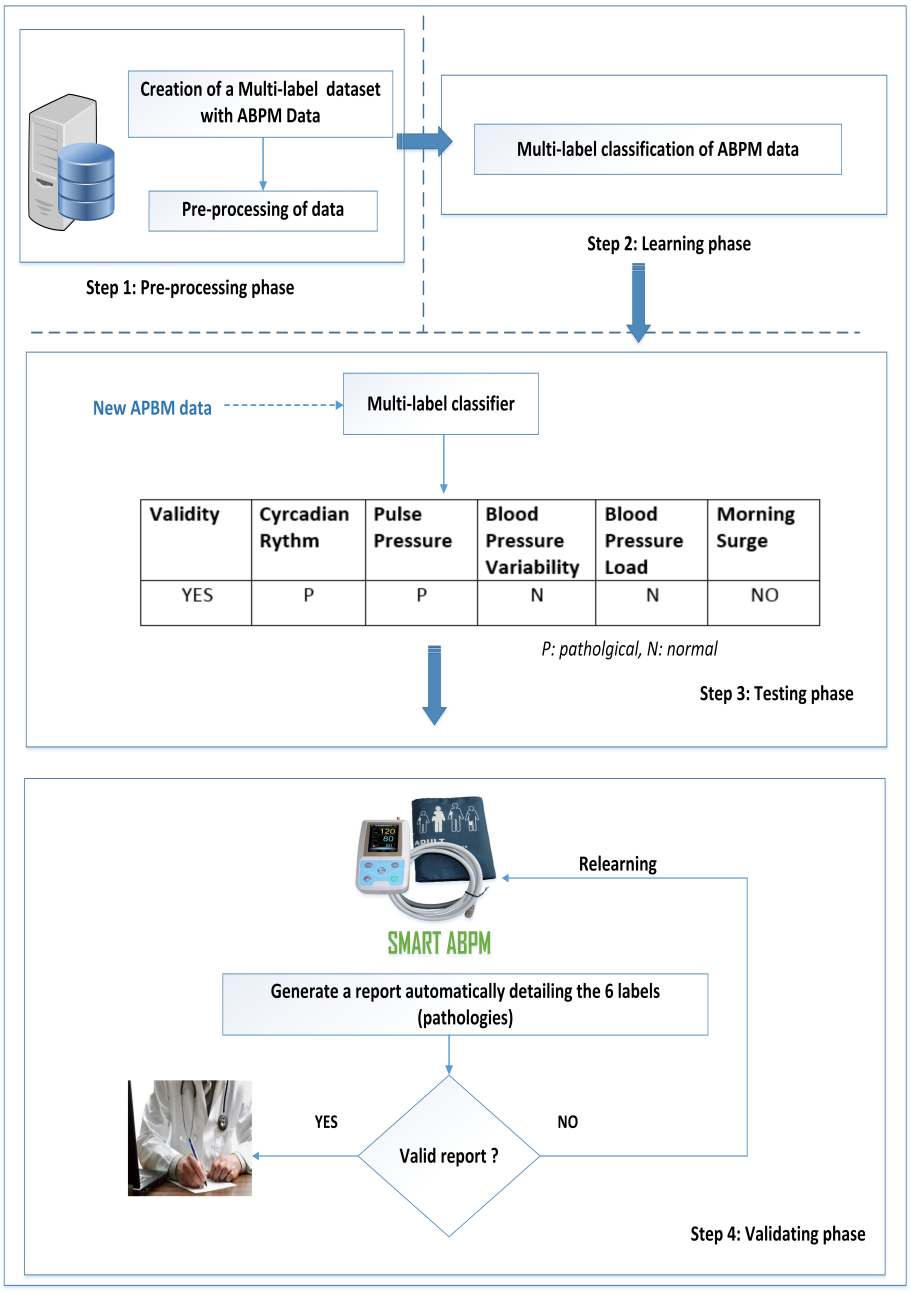

Figure 3.5: Intelligent Analysis of the *ABPM* data

- **Step 1**: **Pre-processing phase and data gathering**, aims to prepare the collected data [4] for training Multi-label classifiers (Section 5.1).
- **Step 2**: **Learning phase**, in this part we search for the optimal Hypothesis H, which associate to each instance of the dataset the correct labels. For that, we conduct a

comparative study of seven Multi-label algorithms using *MEKA* library [61] (Section 5.2).

- **Step 3**: **Testing phase**, Multi-label algorithms learned previously will be evaluated using the most common Multi-label Evaluation Measures (Section 5.3).
- **Step 4**: **Validating phase**, once the best Multi-label classifier is identified, the next goal is to study how can we integrate it with the Ambulatory Blood Pressure Monitor. Then, the proposed Smart *ABPM* will be tested using a new data in real time, the generated report will explain the outputs as the expert does in the manual analysis of the *ABPM* record. The expert should just verify if the report is valid or not, i.e. ensure that the predicted labels by the smart *ABPM* matches with the real labels. If the outputs matches, the report can be printed; if not the system relearn from this data and readjust its Classification, and that to improve the smart *ABPM* decisions for the future (Section 5.4).

5.1 Step 1: Pre-processing phase and data gathering

Our present study developed within the University Hospital Center (*CHU) SETIF cardiology* department, is focused on *ABPM* data which are Multi-label where each *ABPM* record belong to several labels simultaneously. A total of 270 records were collected with 40 attributes and 6 labels: *Validity*, *Circadian Rhythm*, *BPV*, *PP*, *BPL*, *MS* presented previously. This database is an advantageous tool for all studies based on Blood Pressure and *ABPM*'s contribution in this field, and it is released to the public [4], in order to allow comparative experiments by other researchers. The device used in this study is an approved CONTEC ABPM50 v3.2 .

Dataset Statistics The study was performed on 270 patients (159 women and 101 men), aged between 14 and 92 years old. *ABPM* records were labeled by a cardiologist from *CHU SETIF cardiology* department. The Table 3.2 below presents the labels used by the expert, with the number of examples per label.

Let y the label associated with each *ABPM* record, $y = 0$ if normal, otherwise pathological. Except for validity label where $y = 1$ means that the records is Valid, otherwise Not Valid.

Table 3.2: Description of *ABPM* labels and number of examples per label

Labels	# Examples	
	pathological (y=1)	normal (y=0)
Validity	185*	85*
Circadian Rhythm	97	173
Pulse Pressure	232	38
Blood Pressure Variability	270	0
Blood Pressure Load	173	97
Morning Surge	37	233

*Only for Validity label $y = 1$ means that the record is Valid, otherwise Not Valid

Table 3.3 presents the 40 attributes that characterize each *ABPM* data. In the present study, we have ignored 4 attributes which are not important for the Classification process

and most of the data are missing, such as ID patient, Medication, physician comment and pathologies.

Table 3.3: *Description of ABPM attributes, For the attribute Sexe, we present the number of Women and Men in the dataset. The negative values for Sys-Night-Des and Dia-Night-Des, indicates that the BP of some patients increased at night instead of decreasing, called reverse dipper (See Section 4.2).*

Code	Attributes	Type	Min	Max
ID**	*ID patient*	Numeric	1	338
HRecord	*Record over 24 hours*	Numeric	15.5	24
Perc	*Percentage of points ignored*	Numeric	0*	85.5
Interrupt	*Interruption of recording 2h consecutive*	Nominal	0	1
Age	*Age*	Numeric	14	92
Sexe	*Gender*	Nominal	#W:159	#M:101
Weight	*Weight*	Numeric	37	130
Height	*Height*	Numeric	106	190
Medication**	*Medication*	Text		
BPS-24	*All BP Systolic averages*	Numeric	101.8	219.9
BPD-24	*All BP Diastolic averages*	Numeric	56.3	142.9
BPS-Day24	*Day BP Systolic Averages*	Numeric	101.9	219.9
BPD-Day24	*Day BP Diastolic Averages*	Numeric	57.8	149
BPS-Night24	*Night BP Systolic Averages*	Numeric	0*	210.8
BPD-Night24	*Night BP Diastolic Averages*	Numeric	0*	167.8
BPS-load-Day	*Day BP Systolic load valuse*	Numeric	0*	100
BPD-load-Day	*Day BP Diastolic load valuse*	Numeric	1.5	100
BPS-loadNight	*Night BP Systolic load valuse*	Numeric	0*	100
BPD-loadNight	*Night BP Diastolic load valuse*	Numeric	0*	100
Max-Sys	*Maximum systolic*	Numeric	121	283
Min-Sys	*Minimum systolic*	Numeric	0*	157
Max-Dia	*Maximum Diastolic*	Numeric	88	210
Min-Dia	*Minimum Diastolic*	Numeric	0*	88
Sys-Night-Des	*Systolic Night Des.*	Numeric	-68.8	106
Dia-Night-Des	*Diastolic Night Des.*	Numeric	-71.5	100
BPS-CV-all	*BP CV all Sys*	Numeric	5.7	66.2
BPD-CV-all	*BP CV all Dia*	Numeric	6.2	55.3
BPS-CV-Day	*BP CV day Sys*	Numeric	5.9	34.4
BPD-CV-Day	*BP CV day Dia*	Numeric	6.3	42.8
BPS-CV-Night	*BP CV night Sys*	Numeric	0*	260.3
BPD-CV-Night	*BP CV night Dia*	Numeric	0*	137.5
Phys-comment**	*Physician comments*	Text		
Path**	*Pathologies*	Text		
BPS-wakeUp	*BP Systolic two hours after waking up*	Numeric	0*	251

BPD-wakeUp	*BP Diastolic two hours after waking up*	Numeric	0*	221
low-BPS-Night	*The lowest BP Systolic night*	Numeric	0*	191
low-BPD-Night	*The lowest BP Diastolic night*	Numeric	0*	131

**The ignored attributes marked with grey color. *Missing values indicated by 0.

The following Figure 3.6 shows the histogram for each variable of the ABPM dataset [4]

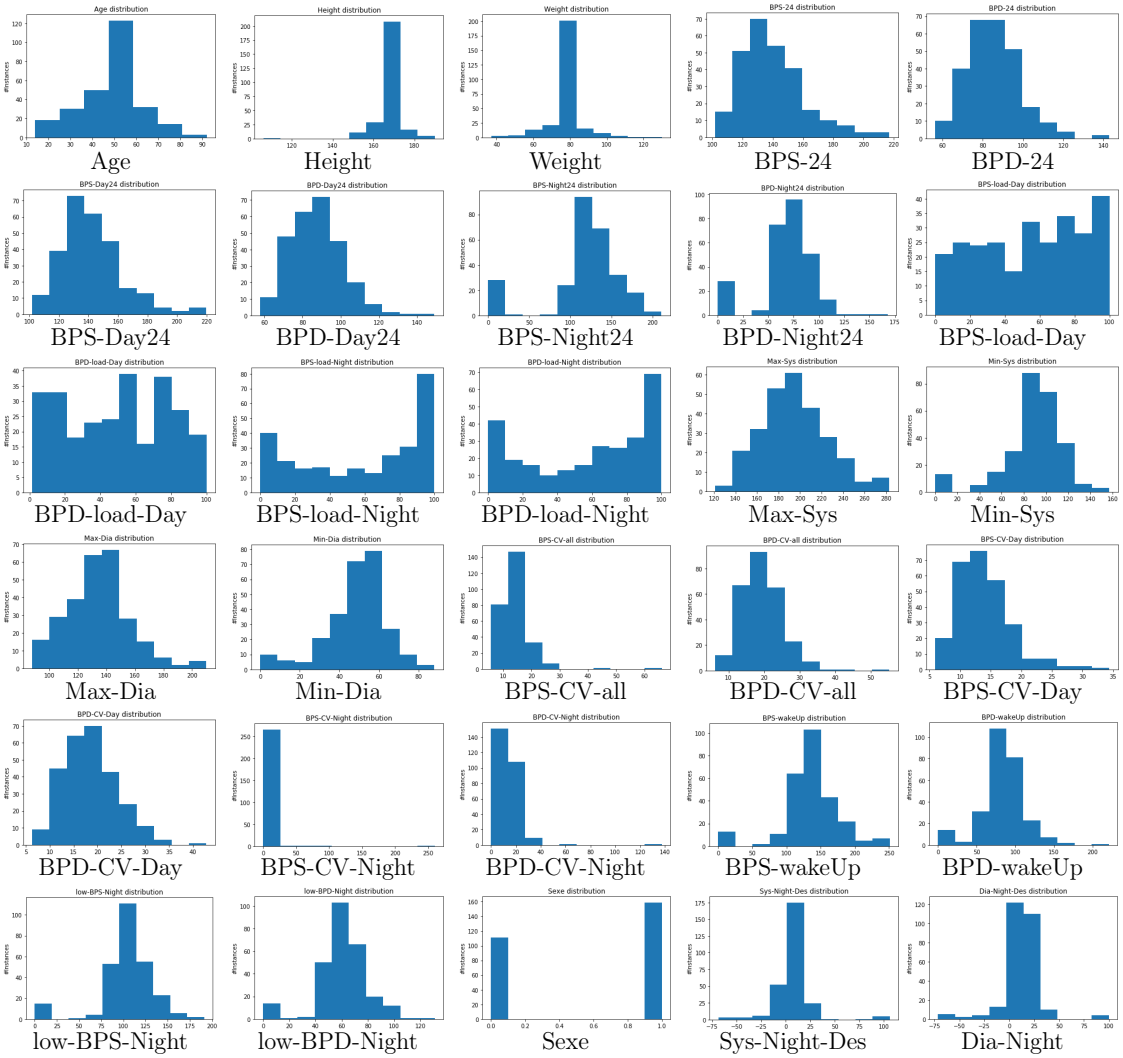

Figure 3.6: Number of Instances per variable for ABPM dataset.

The Multi-labelness of the dataset can influence the different Multi-label algorithms and, it can be assessed using a specific statistics proposed previously in Chapter 1.

Table 3.4 represents the main statistics of the *ABPM* Multi-label dataset. Figure 3.7 shows the distribution of labelsets occurrence.

Table 3.4: *ABPM* dataset statistics

Dataset	Domain	Instances	Attributes	Labels	Cardinality	Density	Distinct	Pmin
ABPM	Medical	270	40	6	4.4630	0.7438	26	0

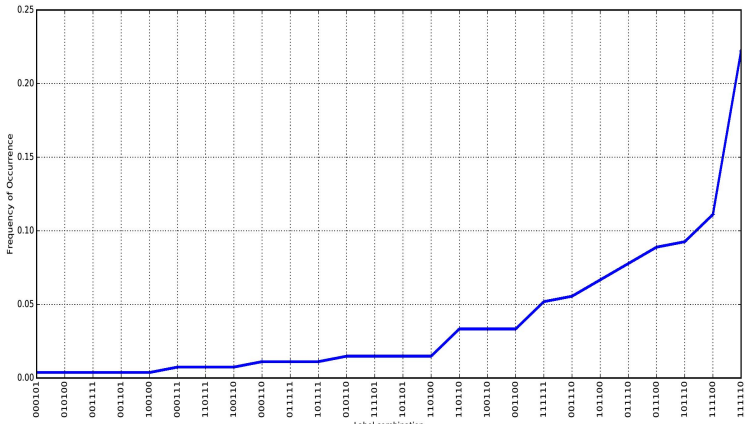

Figure 3.7: The distribution of labelset occurrence. Labels are ordered as per Table 3.2. e.g., 000101 indicates the labels *Blood Pressure Variability* and *Morning Surge*.

The statistics on the *ABPM* dataset [4] (Table 3.4) shows that the data are truly Multi-labeled, the cardinality level is higher indicating that there are in the average 4 to 5 active labels per instance. Similarly, a high density value indicates that the labels are well represented in each instance. Also, zero value for *Pmin* metric denote that there is no Single-labeled instance. It can be used to test the different Multi-label Classification algorithms proposed in the literature and can also be added to the list of available databases [34] (See Chapter 3, Section 3.3.4), which will be very interesting for medical studies.

The *ABPM* data contains crucial information, which can alert the physician to modify treatment accordingly, either by changing the type, dose, or timing of anti-hypertensives agents, or by intensifying treatment for high BP and coexisting Cardiovascular risk factors [107]. However, a lot of doctors and nurses are unfamiliar with the technique and have difficulty in analyzing the data generated. We propose in the following, the use of Multi-label Classification methods to analyze such data, which will greatly facilitate the interpretation and the definition of implicit correlations between *ABPM* attributes and labels.

5.2 Step 2: Learning phase

Classification is one of the most studied data mining topics, which aims to learn from labeled patterns a model able to predict the class for future, never seen before (Supervised learning task). Unlike the Single-label and Multi-class Classification models, the Multi-label Classification task [23,108,109] associated with the data instance a vector of outputs, instead of only one value. The length of this vector is fixed according to the number of different labels in the dataset. Each element of the vector will be a Binary value, indicating if the corresponding label is relevant to the sample or not. Several labels can be active at once and each distinct combination of labels is known as labelset [34]. For example, in text categorization, a page of a newspaper or a web page covers several subjects and should be labeled by several labels according to the subjects covered: sports, announcements, politics, art, news, health, etc.

The Multi-label Classification methods can be categorized into two main groups [12]: *Adaptation* Algorithms and Problem *Transformation*. The first, tackles Multi-label Learning problem by adapting popular learning algorithms to deal with Multi-label data directly, and the second group, solve this problem by transforming it into other well-established

learning scenarios.

In the experimental part (Section 6), we compare the most common Multi-label algorithms, named: *Binary Relevance* (BR), *Label Powerset* (LP), *RAndom k-labELsets* (RAkEL), *Calibrated Label Ranking (CLR)*, *Multi-label K-Nearest-Neighbors (MLKNN)*, *Multi-label Back Propagation* (BP-MLL) and *Probabilistic Classifier Chain (PCC)* , presented in details in Chapter 1.

5.3 Step 3: Testing phase

The Multi-label algorithms cited previously were used for learning from the *ABPM* dataset [4]. In the literature, many researchers and practitioners [110], [111], [112] [74] [113] have relied on a specific machine learning libraries to perform the Classification task. In this work, the *MEKA* library [61] is used, which provides a support for development, running and evaluation of the Multi-label and Multi-target classifiers.

Multi-label algorithms require different Evaluation Measures than traditional Single-label Classification, all algorithms were evaluated under *MEKA* using the most common Multi-label Evaluation Measures described in Chapter 1 named: *Accuracy (Jaccard index)*, *Exact Match (Subset Accuracy)*, *Hamming Loss*, *One Error*, *Ranking Loss* and *Average Precision* .

5.4 Step 4: Validating phase

In any medical decision support system, the validation step is paramount and aims to validate with the expert the results found using the intelligent methods. In our study, the expert should validate if the automatic analysis generated by Multi-label Classification methods matches with the classical analysis.

Our first study aims to find the best Multi-label classifier in order to integrate it in the future to the *ABPM* to make it intelligent. Then, the smart *ABPM* will be tested using data in real time, the generated report will explain the outputs as the expert does in the manual analysis of the *ABPM* record. The expert should just verify if the report is Valid or not, and that to improve the smart *ABPM* decisions for the future.

6 Results

This section provides details the experimental results of the seven Multi-label algorithms, used to evaluate the collected dataset [4] on both Classification and Ranking tasks. The evaluation is carried out using 10 times 10 cross-validation and the winning results are marked with bold font. We used basically two base classifiers: Decision Trees and random forest, which are very intuitive and provide interpretable models, also, they come to a conclusion about the most important attributes which can help experts to analyze the results easily. The results are presented in Table 3.5 and Table 3.6. Note that we kept default parameters defined in MEKA library [61] for both base classifiers.

Table 3.5: *Table of Results of the Application of 7 Algorithms on ABPM Dataset the predictive performance of seven competing algorithms. The base classifier used is Decision Trees. The symbol ↑ means that high values indicates good results and ↓ indicates that the low values are better.*

Algorithms / Evaluation Measure	BR	LP	RAkEL	MLKNN	BPMLL	CLR	PCC
Accuracy (Jaccard index)↑	**0.934**	0.889	0.920	0.815	0.83	0.878	0.922
Exact Match ↑	**0.756**	0.604	0.667	0.389	0.463	0.533	0.711
Hamming loss ↓	**0.051**	0.088	0.062	0.146	0.13	0.094	0.057
One Error ↓	0.022	0.067	**0**	0.007	0.007	**0**	0.015
Rank Loss ↓	0.061	0.131	0.048	0.058	0.051	**0.017**	0.077
Avg precision ↑	0.702	0.616	0.711	**0.831**	0.806	0.764	0.599
F1 (micro averaged) ↑	**0.962**	0.934	0.955	0.895	0.898	0.927	0.956

Table 3.6: *Table of Results of the Application of 7 Algorithms on ABPM Dataset the predictive performance of seven competing algorithms. The base classifier used is random forest.*

Algorithms / Evaluation Measure	BR	LP	RAkEL	MLKNN	BPMLL	CLR	PCC
Accuracy (Jaccard index)↑	**0.929**	0.408	0.915	0.815	0.833	0.871	0.927
Exact Match ↑	**0.719**	0.296	0.678	0.389	0.47	0.493	0.711
Hamming loss ↓	**0.053**	0.553	0.064	0.146	0.127	0.098	0.054
One Error ↓	**0**	0.067	0.011	**0**	0.007	**0**	0.041
Rank Loss ↓	**0.015**	0.53	0.049	0.058	0.053	0.019	0.054
Avg precision ↑	0.793	0.782	0.729	**0.831**	0.799	0.809	0.584
F1 (micro averaged) ↑	**0.96**	0.523	0.953	0.895	0.9	0.924	**0.96**

Table 3.7: *Accuracy (Jaccard index) per label using the studied seven Multi-label classifiers. Note that the results for BPV were ignored due to lack of non-pathological examples for the learning process.*

Labels	*BR*	*LP*	*RAkEL*	*MLKNN*	*BPMLL*	*CLR*	*PCC*	Average
Validity	0,985	0,944	0,967	0,752	0.859	0.941	0.985	0.92 (2)
Circadian Rhythm	0,993	0,933	0,985	0,741	0.719	0.978	0.993	0.91 (3)
Pulse Pressure	0,985	0,789	0,856	0,859	0.822	0.659	0.785	0.82 (5)
Blood Pressure Load	0,985	0,970	0,985	0,907	0.963	0.985	0.985	0.97 (1)
Morning Surge	1	0,896	0.837	0,867	0.859	0.874	0.907	0.89 (4)

Results from the analysis of labels-attributes dependencies for *ABPM* data In order to study relationship between attributes and labels of *ABPM* dataset [4], we plotted the generated Decision Trees by the *PCC* classifier for each label (Figures 3.10, 3.8 and 3.9), since it considers the labels correlations during the Classification process. The attributes marked in bold font are the same attributes used in practice by the expert for analyzing each *ABPM* label and have been well defined by the classifier.

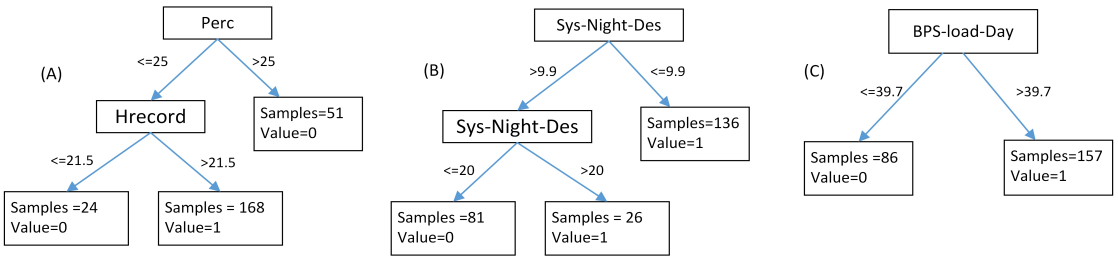

Figure 3.10: The Decision Trees for the Validity (Fig A), Circadian Rhythm (Fig B) and Blood Pressure Load (Fig C).

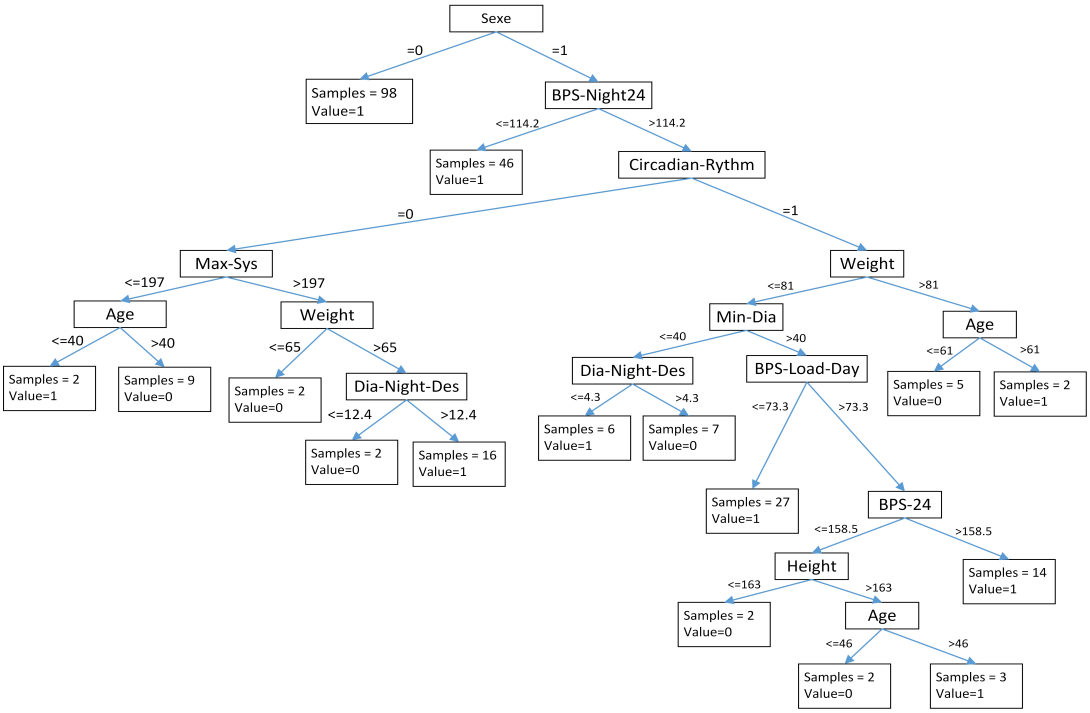

Figure 3.8: The Decision Tree for the Pulse Pressure label

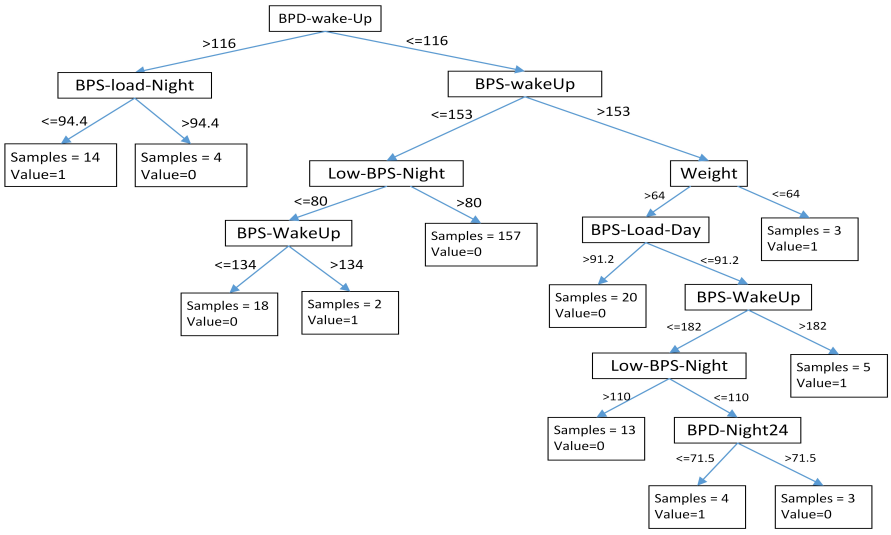

Figure 3.9: The Decision Tree for the Morning Surge label

Results from the analysis of label dependencies for *ABPM* data We analyze the dependencies between the different labels in order to draw conclusions on the different relationships between *ABPM* labels, which provides crucial information during the diagnostic process. For that, we propose the idea of dividing the dataset into 15 subsets, for each one we study the Conditional dependence of labels two by two. Then, we apply two classifiers on each subset: the *BR* which predicts each label separately and the LP which takes into account the correlations between labels. Table 3.8 shows the obtained results.

Table 3.8: *Analysis of Conditional dependence of ABPM labels two by two. The dataset was divided into 15 subsets, for each subset we apply two classifiers the BR which predicts each label separately and the LP which consider Label correlation.*

Dependencies	Evaluation Measure	BR	LP
Validity & Circadian Rhythm	Accuracy	**0,983**	0,978
	Exact Match	**0,978**	0,967
	Hamming Loss	**0,011**	0,017
	One Error	**0,126**	0,137
	Rank Loss	**0,004**	**0,004**
	average precision	**0,789**	0,789
	F1 Micro averaged	**0,992**	0,988
Validity & BPV	Accuracy	0,993	0,993
	Exact Match	0,985	0,985
	Hamming Loss	0,007	0,007
	One Error	0,004	0,004
	Rank Loss	0	0
	average precision	0,843	0,843
	F1 Micro averaged	0,996	0,996
Validity & MS	Accuracy	**0,926**	0,909
	Exact Match	**0,893**	0,874
	Hamming Loss	**0,059**	0,063
	One Error	0,315	**0,307**
	Rank Loss	**0,026**	0,041
	average precision	0,709	**0,717**
	F1 Micro averaged	**0,927**	0,923
Circadian Rhythm & BPV	Accuracy	0,996	0,996
	Exact Match	0,993	0,993
	Hamming Loss	0,004	0,004
	One Error	0,004	0,004
	Rank Loss	0	0
	average precision	0,826	0,826
	F1 Micro averaged	0,998	0,998
Circadian Rhythm & MS	Accuracy	**0,926**	0,896
	Exact Match	**0,893**	0,852
	Hamming Loss	**0,056**	0,074
	One Error	**0,319**	0,326
	Rank Loss	**0,033**	0,044
	average precision	0,707	**0,713**
	F1 Micro averaged	**0,929**	0,905
Pulse Pressure &BPL	Accuracy	**0,896**	0,865
	Exact Match	**0,822**	0,763
	Hamming Loss	**0,089**	0,119
	One Error	**0,078**	0,115
	Rank Loss	0,056	**0,03**
	average precision	0,804	**0,791**
	F1 Micro averaged	**0,942**	0,921
BPV &BPL	Accuracy	0,993	0,993
	Exact Match	0,985	0,985
	Hamming Loss	0,007	0,007
	One Error	0	0
	Rank Loss	0	0
	average precision	0,82	0,82
	F1 Micro averaged	0,996	0,996
BPL &MS	Accuracy	0,926	**0,928**
	Exact Match	**0,885**	**0,885**
	Hamming Loss	0,059	**0,057**
	One Error	0,359	**0,33**
	Rank Loss	**0,03**	0,044
	Average Precision	0,726	**0,733**
	F1 Micro averaged	0,924	**0,927**

Dependencies	Evaluation Measure	BR	LP
Validity & Pulse Pressure	Accuracy	**0,889**	0,865
	Exact Match	**0,826**	0,8
	Hamming Loss	**0,089**	0,1
	One Error	0,096	**0,085**
	Rank Loss	**0,044**	0,074
	Average Precision	0,835	**0,85**
	F1 Micro averaged	**0,943**	0,936
Validity & BPL	Accuracy	**0,983**	0,98
	Exact Match	**0,974**	0,967
	Hamming Loss	**0,015**	0,019
	One Error	**0,133**	0,141
	Rank Loss	**0,004**	0,015
	Average Precision	0,789	**0,804**
	F1 Micro averaged	**0,989**	0,986
Circadian Rhythm & Pulse Pressure	Accuracy	0,837	0,837
	Exact Match	0,774	0,774
	Hamming Loss	0,113	0,113
	One Error	0,133	0,133
	Rank Loss	0,059	0,059
	Average Precision	0,852	0,852
	F1 Micro averaged	0,925	0,925
Circadian Rhythm & BPL	Accuracy	**0,989**	0,981
	Exact Match	**0,978**	0,97
	Hamming Loss	**0,011**	0,015
	One Error	**0,13**	**0,13**
	Rank Loss	**0,004**	0,019
	Average Precision	0,774	**0,781**
	F1 Micro averaged	**0,991**	0,989
Pulse Pressure & BPV	Accuracy	0,919	0,919
	Exact Match	0,837	0,837
	Hamming Loss	0,081	0,081
	One Error	0,044	0,044
	Rank Loss	0	0
	Average Precision	0,93	0,93
	F1 Micro averaged	0,957	0,957
Pulse Pressure &MS	Accuracy	**0,806**	0,763
	Exact Match	**0,77**	0,715
	Hamming Loss	**0,133**	0,156
	One Error	0.174	**0,148**
	Rank Loss	**0,07**	0,137
	Average Precision	0,648	**0,681**
	F1 Micro averaged	**0,868**	0,848
BPV &MS	Accuracy	0,948	0,948
	Exact Match	0,896	0,896
	Hamming Loss	0,052	0,052
	One Error	0	0
	Rank Loss	0,037	0,037
	Average Precision	0,587	0,587
	F1 Micro averaged	0,954	0,954

7 Discussion

From Table 3.5 and Table 3.6, we noticed that the use of Decision Tree as a base classifier for the seven algorithms gives best results. *CLR* outperforms the other algorithms in terms of Ranking Evaluation Measures. Based on the pairwise comparisons of labels, the *CLR* ranks efficiency relevant labels higher than irrelevant labels.

With reference to the other Evaluation Measures, such as Accuracy (Jaccard index), Exact Match, Hamming Loss and Fmeasure, the *BR* and the *PCC* give the best results. The *BR* does not take into account Label correlation. It uses six classifiers separately which is equal to the number of labels in the *ABPM* dataset. However, the *PCC* algorithm estimates the entire joint distribution of all possible label combinations and seeks to maximize the posterior probability of the predicted label combination.

At the same time, *RAkEL* gives satisfactory results compared to the other algorithms. Additionally, it takes into account Label correlation in the dataset which is a very important criterion, since we would like not only to automate the analysis process of the *ABPM* data but also to give the cardiologist the different implicit relationship between the different labels. *RAkEL* is based on an ensemble of *LP* classifiers which justify the improvement of the *LP* results. Normally, the results given by *RAkEL* outperforms greatly the *LP* results, but one reason for the reconciliation of their results in this study is the relatively small number of labels in the *ABPM* dataset.

Table 3.7 shows the Classification Accuracy per label for the studied algorithms, using as base classifier the Decision Tree since it gives better results than Random Forest (Table 3.5 and Table 3.6). The last column indicates the average accuracy for each label and between brackets the descending rank of labels from the easier to predict up to the difficult.

The easiest labels to predict are: *Blood Pressure Variability*, *Blood Pressure Load* and the *Validity* with the 99 %, 97% and 92% mean accuracy, respectively. The results obtained from those labels are very encouraging, once the validity of the *ABPM* is determined, the expert could make other conclusions for the other labels. For example, the high *Blood Pressure Variability* indicates a high *Blood Pressure Load* which gives the cardiologist an idea about the *Morning Surge.*

However, the two hardest labels to predict are *Circadian Rhythm*, *Morning Surge* and *Pulse Pressure* with 91%, 89% and 82% respectively. This can be improved using feature selection methods because, even in the traditional interpretation of the *ABPM*, a good Classification requires to analyze specific attributes.

7.1 Discussion of the analysis of labels-attributes dependencies for *ABPM* data

We note from Figure 3.10 (A) that there is a strong correlation between the *Validity* label and the two attributes: **Perc** and **HRecord**. The two attributes are the most pertinent to judge the validity of *ABPM* record.

The same thing for the *Circadian Rhythm* (Figure 3.10 (B)) which is strongly correlated with **Sys-Night-Des**, the thresholds determined by the classifier are very interesting. In cardiology, if it is superior than 10 %, we speak about a pathological Blood Pressure Pro-

file (*BPP*), several intervals were defined in the literature for determining the *BPP* (See Section 4.2). We consider that in future works, by adding samples with several types of *BPP* and using the multi-dimensional algorithms to consider this issue.

The prediction of *Blood Pressure load* (Figure 3.10 (C)) depends mainly on the *BPS-load-Day* attribute. The threshold determined by the classifier is very close to the value used in practice (40 %), which is very interesting since Blood Pressure load provide useful information for diagnosing hypertension [114].

Figure 3.8 gives an intuitive and great visual analysis regarding pertinent attributes, we remark that *Pulse Pressure* label depends on the *Circadian Rhythm* represented as a node in the tree. The main attribute **BPS-24** for predicting pulse pressure was well defined with other interesting parameters such as age, gender, circadian rhythm and weight considered as risks factors for Cardiovascular morbidity [107].

PP has also been shown to be a powerful predictor of cardiovascular morbidity in elderly men [107], which explains the fact that we have practically pathological PP (value=1) in the Decision Tree for men (Sexe=0). Pruning the Decision Tree will be also interesting by removing nodes that provide less additional information such as height node.

Experts usually use the **BPS-wakeUp** and **lowBPS-Night** for predicting the *Morning Surge* label. However, the generated Decision Tree (Figure 3.9) shows that there are other possible alternatives, and other attributes to be considered such: BPD-wakeUP, BPS-load-Night, weight, BPS-load-Day and BPD-Night24 . For example: if ($BPD - wakeUP > 116$) and ($BPS - load - Night \leq 94.4$), then MS=1 (presence of morning surge). But, if we consider also the pertinent attributes used by the expert: **BPS-wakeUp** and **lowBPS-Night**, the prediction will be more relevant. for example, if ($BPD - wakeUP \leq 116$) and ($BPS - wakeUP \geq 153$) and ($lowBPS - Night > 80$), then MS=0 (absence of MS). More studies should be performed on other ABPM data with more instances to confirm the present findings.

Finally, it is not interesting to represent *Blood Pressure Variability* (BPV) since the collected dataset contains only patients with pathological BPV but we intend to enrich it with the non-pathological cases in the near future.

7.2 Discussion of the analysis of label dependencies for *ABPM* data

By analyzing the results (Table 3.8), we notice that:

- In the case of the following subsets: «Validity & Circadian Rhythm », «Validity & Pulse Pressure », «Validity & BPL »and «Validity & MS », BR give best results. We can conclude that the validity label is not correlated with the other labels of the dataset. This is confirmed in the medical context, as this attribute allows the expert only to judge whether the $ABPM$ record is valid or it is necessary to record another.

- From the «Circadian Rhythm & BPL »and «Circadian Rhythm & MS »subsets, we conclude that the label *Circadian Rhythm* is independent of the two labels: *BPL* and *MS*.

- According to the results obtained on the «Circadian Rhythm & BPV »subset, the *Circadian Rhythm* is probably correlated with the *BPV* label, this information must be affirmed after extending the *ABPM* dataset with cases having the non-pathological BPV.
- We also notice in the following cases: «Pulse Pressure & BPV », «BPV & BPL », «BPV & MS »,«Circadian Rhythm & Pulse Pressure , that the BR and LP classifiers gives the same results which indicate that there is a probable dependence on the one hand between the Pulse Pressure and Circadian Rhythm, and on the other hand the *BPV* with Pulse Pressure, BPL, and MS. This dependence must be confirmed using labels correlation approaches.
- The Pulse Pressure is Conditionally independent of the two labels *BPL* and *MS*, since the BR gives better results on the «Pulse Pressure & BPL »and «Pulse Pressure & MS »subsets.
- For the «BPL & MS », the LP gives good results, this allows us to conclude that the two labels *BPL* and *MS* are Conditionally correlated.
- For the following subsets: «Validity & BPV », «Circadian Rhythm & BPV », the two classifiers *BR* and *LP* gave the same results. In the case of the *BPV* label, we cannot come out with significant conclusions since the learning sample available contains only *ABPM* records with pathological Blood Pressure Variability.
- In many cases, the BR classifier gives good results for Classification Evaluation Measures such as Accuracy, Exact Match, Hamming Loss and Fmeasure. While the *LP* gives better results on Ranking measures like One Error, Rank Loss, and Average Precision. So using other correlation search approaches is necessary to check if there are any real dependencies between the labels.

The Table below (Table 3.9) summarizes the dependencies between the *ABPM* labels presented by the *PCC* classifier (Section 7, Table 3.8):

Table 3.9: *Summary Table of the dependencies between the ABPM labels. NC: Not Correlated, PC: Probably Correlated, CCo: Conditionally Correlated, CI: Conditionally Independent.*

	Validity	Circadian Rhythm	BPV	PP	BPL	MS
Validity		NC	NC	NC	NC	NC
Circadian Rhythm			PC	PC	NC	NC
BPV				PC	PC	PC
PP					CI	CI
BPL						CCo
MS						

7.3 Study limitations and further research

We are aware that our results may be interpreted with caution and there are some limitations that deserve mention. Firstly, the collected dataset contains only 270 records which limit the interpretation of results, we intend to expand our database with more examples of learning, by taking the case of multi-dimensional data into consideration for the two

labels: *Blood Pressure Load* and *Circadian Rhythm* which contains more than one class for each label. Secondly, the collected dataset contains only patients with pathological *BPV* since the study was developed in cardiology service. However, we intend to enrich it with the non-pathological cases in the near future in order to improve the learning process for this label.

8 Conclusion

The *ABPM* data occupies currently a central place in the diagnosis and follow-up of the hypertensive patients since it contains crucial information on the patient condition. It allows to make a specific diagnosis and adapt therapeutic attitude accordingly. However, this variety of the *ABPM* data is not exploited by all the doctors. Indeed, they are unfamiliar with its analysis and it is time-consuming, which constitutes a real gene for them. In this paper, we propose an automatic analysis of the *ABPM* data using intelligent methods, in order to help the expert to exploit and to analyze them easier.

The study consists of two major parts: the first is a data collection of 270 patients in the CHU SETIF cardiology department, each *ABPM* record is characterized by 40 attributes and associated with six labels simultaneously. The Multi-label dataset is published online [4], to allow researchers in the medical field to draw up statistical or even data mining studies more easily, as no *ABPM* dataset is published in the literature. The second part presented a comparative study of seven algorithms and also the analysis of dependencies between the labels and attributes of the *ABPM* dataset. The results show the efficiency of the *RAkEL* algorithm for Multi-label Classification task and the *CLR* for the Ranking Task. The initial results on Conditional dependence analysis of *ABPM* labels encourage us to expand our study in future using more specialized algorithms.

Further works are currently underway with the use of the feature selection and the Label correlation methods to improve on the one hand the accuracy per label since each label of the *ABPM* dataset is strongly associated with specific attributes. On the other hand, to extract new and implicit correlations between the different labels and features.

Chapter 4

Label Correlation for Multi-label Classification based Decision Trees

1 Abstract

One of the recurrent studied topic in *Multi-label* Classification (*MLC*) is Label dependence and its advantages for this research field. Many researchers addressed this issue locally before the Classification by searching possible Label dependence directly from the dataset. However, others interested to apply such task dynamically during the Classification process. Two major types of Label dependence were highlighted in many recent works, named: Conditional and Marginal Label dependence. The importance of investigating such correlation in medical datasets conducted us to study in depth this notion in this chapter. For that, we reviewed recent works addressing Label dependence based on several *Multi-label* algorithms in Section 3, including *Transformation* methods and *Adaptation* algorithms. We present also the main differences between the two defined types of Label correlation in Section 4 in which we focused on the use of Decision Trees as a base classifier and its main advantages. Finally, in Section 5, we present a comparative study of six well-known algorithms in the literature based Decision Trees, and we discuss the benefits of considering Label dependence using six datasets, five datasets from literature named: *Yeast* [20], *Scene* [17], *Emotions* [22], *Genbase* [42] and *Medical* [35], we added also our collected dataset named *ABPM* [4] for the experiments (See Chapter 3).

2 Introduction

Recently, *Multi-label* Classification (*MLC*) was widely applied by researchers in many fields such as text categorization, automatic tagging from multimedia resources including (images, audio and videos). The main reason for using *MLC* is the ability to classify each object into several labels at once. As presented in the previous chapters, the *MLC* problem can be addressed using many approaches including *Transformation* methods, *Adaptation* algorithms (refer to Chapter 1) and Ensemble Methods (refer to Chapter 2). The data miner can choose the relevant approach to use depending on the problem and the data used for that. However, one of the main challenges of dealing with *MLC* is the use of Label dependency information to improve the Classification task.

Many researchers highlighted the importance of this information [[115], [116]] in such a task, note that the basic *Transformation* method called *Binary Relevance (BR)* [11] ignore totally Label dependence since it applies a separate classifier for each label. The

other well-known Transformation approach is Label Powerset (LP) [11] that incorporate implicitly Label dependency during the Classification process since it considers all labels for an instance as a new class. Many other methods addressed this issue based on two types of Label dependence called: Conditional and Unconditional Label dependence [117].

Roughly speaking, the first consider Label dependence according to the feature space of instances, LP approach is a good example of this first category, while the second provides Label correlation only between labels i.e if we consider for example y_i and y_j two labels that are Unconditional dependent that means $P(y_i|y_j) \neq P(y_i)$.

Similarly, in [59], [117], the authors explain the benefit of exploiting dependencies among labels to improve the performance of Multi-label classifiers. In addition, they criticize the fact of addressing this problem by modelling explicitly existing Label dependence from the dataset. They present two types of Label dependence, namely: Conditional and Marginal dependence.

In this chapter, we present in depth the advantages of taking Label correlation information during the MLC, by highlighting the difference between the two approaches: Conditional and Unconditional (Section 4). Also, we review the well-known algorithms addressing this task in Related Work section 3, then, we focus on the methods based Decision Trees approach to compute Label correlation in ML datasets by presenting a case study with the results found (Section 5).

3 Related Work

Exploring Label dependencies during the Classification task is one of the key challenges in *MLC*, especially for datasets with a huge number of labels. However, ignoring this concept will lead to a high information Loss, for that many researchers highlighted the importance of taking Label correlation into account during the Classification process to reduce the complexity and help to discover hidden information in the labels space. They generally consider two types of Label dependence [56], [117]. Conditional Label dependence reflects how likely or unlikely labels are to occur together given the attribute values of a specific instance [60]. Nevertheless, Unconditional Label dependence (known as Marginal) focus only on the label space and how certain labels are likely or unlikely to occur together. The Conditional Label dependence was studied in depth in [56], [23], [11], similarly the Marginal dependence have been explored in multiple works as in [46], [118], [119].

Another categorization of ML algorithms based on Label correlation was proposed in [116] in which the authors define three order of correlation, the first-order ignore totally the Label dependence by decomposing the ML problem to a separate Binary problems [17], [10], [120]. While the second-order consider pairwise relations between labels such as the Ranking between the proper label and the improper label of an example [20], [53], [24] or the interaction between any pair of labels [[121], [19], [25]]. Finally, the high order approaches that consider deeply relatioships between labels such as the full order style of imposing all other labels' influences on each label.

In the following, methods considering Label dependence will be categorized as in the first Chapter 1, Transformation methods with the major Transformations: Binary Relevance (BR) and Label Powerset (LP), then, Adaptation algorithms and finally algorithms

based Decision Trees that consider Label dependence during the Classification process.

3.1 Transformation approaches

As we explained previously, two main Transformations of the Multi-label problem are known [11] as BR and LP, we present thereafter some recent algorithms from the literature considering Label dependence based on those two transformations.

Binary Relevance Approaches

In BR [11], the Multi-label problem is transformed to several Binary Classifications, for each label a Single classifier is built, and the final prediction is the union of all single predictions. Despite the simplicity of BR approach, it suffers from major limit is the fact that each classifier is learned separately with ignoring possible dependencies between labels.

To overcome this disadvantage, many variations of BR were proposed in the literature. As Classifier Chain (CC) [54], in which the use of a chain of Classifiers is proposed to deal with Label dependence, by extending the feature space of each classifier with the outputs of all previous classifiers. The main drawback of CC is that the order of the chain has an effect on predictive performances. For that many extensions were proposed to deal with the ordering issue in two ways: using heuristic for selecting a chain order or by using an Ensemble of Chain Classifier. However, most of the proposed approaches are much higher complexity than the CC classifier.
For the first view, we name Probabilistic Classifier Chain (PCC) [56] which is a probabilistic extension of the CC algorithm, for each label combination the Conditional probability is computed. Then, for estimating the joint distribution of labels, a model is learned for each label on a feature space augmented by previous labels as additional attributes. The Classification prediction is then derived from the calculated join distributions in an explicit way. In fact, the main disadvantage of the PCC method is its applicability only on datasets with a small number of labels, not more than about 15 [59].

Another similar work called Bayesian Chain Classifiers (BCC) was proposed in [122], first, a Bayesian Network (BN) based a Decision Trees structure is used to capture possible dependencies among labels, then, a CC model is build based on the dependence structure. The main advantages of using BN highlighted by the authors are: (i) represent the probabilistic dependency relationships between classes, (ii) constrain the number of class variables used in the Chain Classifier by considering Conditional independence conditions, and (iii) reduce the number of possible chain orders.

For the second point of view, i.e using Ensemble Methods to consider Label dependence, we name Ensemble Classifier Chains (ECC) [55], that use a committee of CC models and each one learn on random chain orderings, using a random subset of training instances. The final prediction of ECC is computed using a majority vote strategy (For more details about the use of Ensemble Methods in MLC, refer to Chapter 2).

As CC classifier, many other algorithms were proposed in the literature to deal with Label correlation issue for BR classifier, as in [123] where the authors use the idea of stacking [124] BR classifiers to model Label correlation based a meta-level classifier. In general, staking means to learn a second (meta) models that consider as input the output

of all first (base) level models. The authors show that the detected Label correlation are useful and meaningful.

Label Powerset Approaches

For the Label Powerset transformation, we name hereafter some algorithms considering Label dependence among labels.

LP [11] consider each combination of labels as a new class and learn a classifier to predict the outputs, which make it very simple and effective for datasets with few numbers of labels. However, LP is limited for large datasets, since it can be computationally expensive. Moreover, the learning process will be really limited with just few learning instances for some combinations at leaves. Another drawback of LP is the over-fitting problem since an LP classifier cannot predict a class that has not to be seen previously during the learning process. Many extensions were proposed to deal with this drawback such as Pruned Set (PS), Ensemble of Pruned Sets (EPS) [55] and RAndom k-labELsets (*RAkEL*) [11].

In [59], the authors propose an approach called LPBR, that explores two types of Label dependencies: Conditional and Unconditional, by clustering labels into several independent subsets based on chi^2 test. Finally, It applies a Multi-label classifier for learning, and for predicting the labels at leaves, it uses LP for dependent labels and BR otherwise.

Another work [125] combine LP and BR methods to well explore Label dependencies. First, it divides the set of labels into several mutually exclusive subsets of dependent labels. Then, a Classification algorithm incorporating dependencies among labels within each subset can be applied. They show that applying a combination of BR and LP methods to these subsets provides in many cases higher predictive performance than regular LP and BR approaches.

3.2 Adaptation approaches

The idea of Adaptation methods is to modify the existing algorithms to solve Multi-label Learning (MLL) problems, more details about this category of methods are presented in depth in Chapter 1. In the following, we present many learning rules as SVM, DT, Naive Bayes that was adapted to MLL by considering also dependencies among labels in various ways.

Zhang et al. in [9] (2009) proposed an extension of the popular Naive Bayes classifiers for dealing with Multi-label instances called MLNB. The authors highlighted the benefits of using feature selection techniques such as principal component analysis and genetic algorithms in addressing the inter-label relationships. The correlation between labels was explicitly addressed through the specific fitness function used by the genetic algorithm.

Another example of adaptation algorithms for Multi-label Classification considering inter-label relationships is ML-SVDD [126], the algorithm is based on Support Vector Data Description. First, a k-label problem is divided into k sub-problems each of which consists of instances from a specific class. Then, for each class, a sub-classifier is learned using support vector data description method. The combination of all predictions of all sub-classifier to form the Multi-label Classification is done as follows. The classes which

predicted pseudo posterior probability above some threshold are added to the set of predicted labels. To compensate for missing correlations between labels linear ridge Regression model is used when constructing a threshold function [59].

Another work called: LEAD [116], focus on Conditional Label dependence and propose the use of Bayesian Network structure to encode the identified Label dependencies. It decomposes the Multi-label problem into a series of Single-label Classification and for each one, its parental labels defined by the Bayesian structure are incorporated as additional features.

Acknowledging the advantages of Decisions Trees (DT) algorithm, we present in the following few works from the literature based on DT to exploit Label dependence. More explanations about our choice are presented in the motivation section in Section 4.1.

The first work [10] aims to adapt the Decision Trees C4.5 to the Multi-label problem by computing separately the Entropy for each label (MLC4.5), and then sums all Entropies to decide to split or not at each node of the tree. The main advantage is the identification of relevant features to all labels at once, nevertheless, the splitting criterion does not consider any Label correlation.

In the same context [127], the authors propose *ML SVMDT* that build DT as *MLC4.5* and use at leaves *BR* classifiers based on *SVM*. Similarly, *ML-Tree* [128] consider the Tree as a hierarchy of data and use *SVM* classifiers at each node for splitting. It considers the label relationship by estimating the co-occurrence at each node of the hierarchy and transferred it in a top-down manner.

Successful Multi-label Learning needs to consider Label correlation problem since in real-world applications labels are fully or partially correlated, and this dependence encodes very useful information for the Classification process.

In [129], the authors propose a DT based Multi-label classifier considering the covariance matrix as a splitting criterion at each node of the Tree, then, it applies for independent labels a *BR* at leaves and if dependent, the labels are kept together as in LP. The main advantage of LaCova is the fact that it takes into consideration correlation among labels locally during the splitting process which allows choosing between a horizontal split (branching on a feature) and a vertical split (separating the labels). However, its main disadvantage is that the considered Label dependencies are Marginal since the covariance matrix is computed on labels without taking into consideration the features space. To overcome this limit, the authors propose the clustering of labels dynamically during the construction of the Tree in [130] and [131] to model the Conditional Label dependence.

In the same way, other works focus on exploiting labels dependencies using clustering strategies over the entire dataset as Hierarchy of Multi-label classifiers (*HOMER*) [37]. The algorithm partitions labelset into small disjointed subsets using a clustering approach and organize each subset of labels at each node of the Tree-shaped hierarchy, then a Multi-label classifier is learned to predict the set of labels. One of the criticisms of HOMER is the direct and global identification of dependent labels from the dataset, so the splitting criterion is not guided by Label dependence at each node of the Tree. Moreover, the approach is computationally expensive even if it shows competitive results in terms of Classification accuracy.

On his side, Extremely Randomized Trees (*ERT*) [132] randomly choose both feature and cut point while splitting a Tree node. The authors consider that the main strength of ERT is the computational efficiency.

In the experimental section 5, we compared the performance of some algorithms described above including BR [11], CC [54], ECC [55], ERT [132], LaCova [129] and MLC4.5 [10] using the well known Multi-label datasets: *Yeast* [20], *Scene* [17], *Emotions* [22], *Medical* [35] and *Genbase* [42], we also added our collected *ABPM* dataset for the experiments. The idea is to study the impact of the use or not of Label correlation as a splitting criterion in the *Decision Tree*. Before that, in the next section we give more explanations about the main differences between the two types of Label dependence: Conditional and Unconditional, also we explain our main motivation about using Decision Trees for exploring Label dependence among labels.

4 Label dependence for Multi-label Classification

Exploiting Label dependencies among labels is one of the main challenges in Multi-label Classification and could greatly improve the classifier performance and facilitate the learning process. In addition, it is required in many real-world applications in which the analysis of such correlations leads discovering crucial hidden information inside datasets.

Label dependence, known also as Label correlation was widely addressed by many researchers in literature, the importance of taking this information during the Classification process may greatly improve the classifier performance also the interpretability of the model. A simple example of scene labeling may well explain its advantages, in such situation the probability of the label beach is higher if the label sea is also relevant. Hence, taking this information into account is important to produce efficient model.

Over the last few years, most of the proposed Multi-label Classification algorithms consider Label dependencies during the learning process, from the statistical perspective, two kinds of Label dependence were defined, namely Conditional and Unconditional. In the rest of this section, we present the formal differences and connections between both types, by means of simple examples. Our main reference is [117], where the authors give an interesting and recent overview of state-of-the-art algorithms for MLC and categorize them according to the type of Label dependence they seek to capture.

The authors in this work [117] confirm that optimal predictive performance can only be achieved by methods that explicitly account for possible dependencies between class labels. However, they argue that major studies of this kind do often fall short at deepening the understanding of the *MLC* problem for many reasons. We note in the following the main reasons discussed by the authors:

- The notion of Label dependence or Label correlation is often used in a purely intuitive manner, referring to a kind of non-independence, but without giving a precise formal definition. Likewise, *MLC* methods are often ad-hoc extensions of existing methods for Multi-class Classification.
- Many studies report improvements on average, but without carefully investigating the conditions under which Label correlation are useful.
- The reasons for improvements are often not carefully distinguished. As the performance of a method depends on many factors, which are hard to isolate, it is not

always clear that the improvements can be fully credited to the consideration of Label correlation.

We present in the following the main differences between the two types of dependency that has been highlighted in the literature. Conditional Label dependence captures Label dependencies Conditional to a specific instance, while Marginal Label dependence is global and independent from any observation.

Let Y a random vector: $Y = (Y_1, Y_2, Y_3, \cdots, Y_q)$. Y is Marginally independent if

$$P(Y) = \prod_{i=1}^{q} P(Y_i) \tag{4.1}$$

While Y is called Conditionally independent given an instance x, if

$$P(Y|x) = \prod_{i=1}^{q} P(Y_i|x). \tag{4.2}$$

In the same paper [117], the authors discuss the possible ways to model Label dependence for Multi-label Classification, they consider $Y_i = h_i(x) + \varepsilon_i(x)$ where: $i = 1, \ldots, q$ and functions $h_i : \chi \to \{0, 1\}$ represent the structural parts of the model and the random variables $\varepsilon_i(x)$ is the stochastic parts. Krzysztof et al. explain that the distribution of error terms can depend on x. Furthermore, two noise terms ε_i and ε_j may share some similarities between each other. As also, the structural parts of the model h_i and h_j can depend on each other. From this, they conclude that there are two possible sources of Label dependence:

- The structural part of the model h
- The stochastic part ε

Krzysztof et al. highlighted that the Marginal Label dependence is caused by the structural part of the model h. However, The stochastic part ε is responsible for the Conditional Label dependence. Labels are considered Conditionally dependent if the errors terms of the model are dependent. Moreover, the observation of Label dependence in the training data will not necessarily imply any error terms dependence, it only informs about the existence of Marginal dependence between labels.

In the same context, another work was presented in [116] studying Conditional Label dependence based on error terms. First, a Directed Acyclic Graph (DAG) is built to characterize the joint probability of all labels conditioned on the feature set, the correlations are represented by the DAG structure. Second, Multi-label Learning is decomposed into a series of Single-label Classification problems. Each Binary classifier is learned by considering its parental labels from the DAG structure as additional features. Finally, the labelsets of unseen examples are predicted recursively according to the label ordering given by the Bayesian Network.

4.1 Motivation

The main goal of data mining is the exploration of knowledge from huge real-world datasets, it refers to a complex machine learning algorithms for modeling and understanding data. We are interested in Multi-label approaches to extract implicit and crucial information

from medical data. In the literature, many statistical learning approaches were proposed to learn from Multi-labeled data and aims mainly to improve the performance of the classifier. Generally, the produced model is seen as a black box by the expert, especially in the medical domain where the use of such automatic process should be controlled by the doctor before giving any results to the patient. For that reason, selecting the best approach can be one of the most challenging parts of performing machine learning in practice. Hence, the approach used for the Classification process should be efficient but also interpretable, and produce a structure that can be easily understood by the expert (doctor).

In machine learning, Tree-based methods are simple and very useful for interpretation. In fact, they are more closely to human decision-making, and they are easily interpreted even by a non-expert; since Trees can be displayed graphically in intuitive structure. Additionally, the most discriminative features for the Classification can be caught easily with even the most critical values at Tree nodes. Moreover, dependence between labels and features can be easily analyzed from the DT structure. However, they typically are not competitive with the best-supervised learning approaches in terms of prediction accuracy [68].

In order to benefit from interpretability of DT and as well as to produce an efficient model, we make a comparative study between many algorithms from both Transformation and Adaptation strategies of *MLL*, based on Decision Trees to model Label correlation. We believe that modeling Label correlation during the Classification process can give an efficient and interpretable model.

In the following, we present briefly the algorithms that we used in this initial comparative study, including: BR [11], CC [54], ECC [55] from Transformation methods and ERT [132], MLC4.5 [10] from Adaptation methods and finally, LaCova [129] that consider Label correlation during the construction of the Tree. We give also some advantages and disadvantages of using each algorithm as well as an illustration of each method, in which we consider an example to classify x and the labels are $\{y1, y2, y3\}$.

Binary Relevance (BR)

In BR [11], the Multi-label problem is transformed to several binary classifications, for each label a single Decision Tree is built, and the final prediction is the union of all single predictions (See Figure 4.1). Despite the simplicity of the BR approach it suffers from many limits. First, the number of classifiers needed for Classification might be thousands for some domain as for example genetics, where several labels may label each gene. Second, the labels are considered independent during the Classification process, which is not suitable for real-world datasets where the prediction of one label gives crucial information about the other labels. Finally, relevant features identified separately for each label cannot be used as relevant features for the whole dataset, and the expert should analyze each Decision Tree separately to find some possibly useful information.

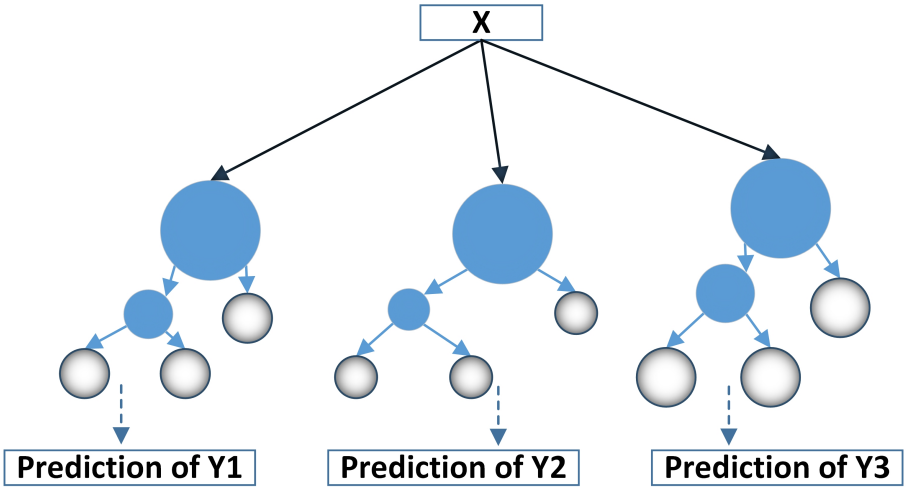

Figure 4.1: An illustration of Binary Relevance approach based DT.

Classifier Chain (CC)

To overcome the disadvantages of BR and take Label dependence into account, many variations of BR were proposed in the literature. As Classifier Chain (CC) [54] (See Figure 4.2), where the authors propose to use a chain of classifiers to deal with Label dependence, by extending the feature space of each classifier with the outputs of all previous classifiers. The main drawback of CC is that different ordering of labels in the chain can affect results. For that many extensions were proposed to deal with the ordering issue as Ensemble Classifier Chain (ECC) [55].

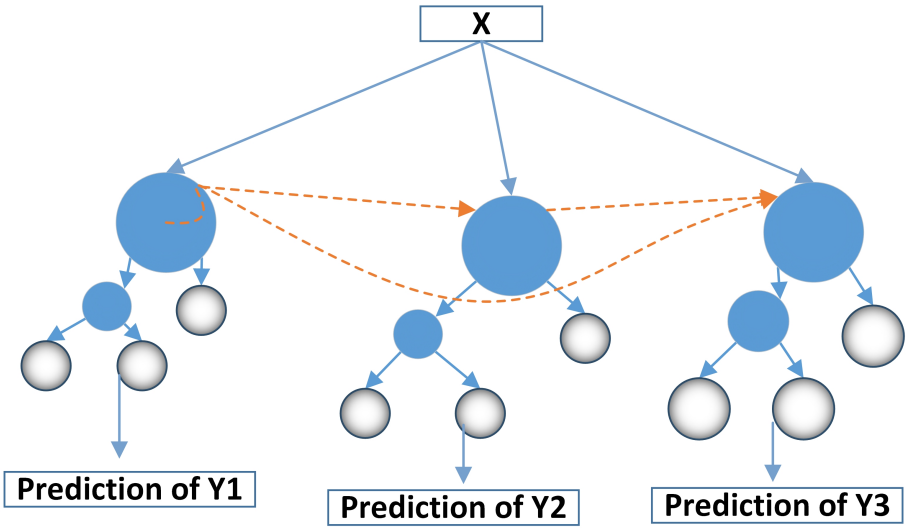

Figure 4.2: An illustration of Classifier Chain based DT.

Extremely Randomized Trees (ERT)

Extremely Randomized Trees (ERT) [132] randomly choose both features and cut points while splitting a Tree node. The authors consider that the main strength of ERT is computational efficiency. The pseudo code 3 explain the main steps of building the tree structure using ERT method.

Algorithm 3 Pseudo code of *ERT*

1: **Split_node(S)**
Input: The local learning subset S corresponding to the node we want to split.
Output: a split [f $< Cut_f$] or nothing
2: IF **Stop_split**(S) is TRUE THEN return nothing.
OTHERWISE select n features $\{f_1, f_2, \ldots, f_n\}$ among all non constant (in S) candidate attributes;
Draw K splits $\{s_1, s_2, \ldots, s_n\}$, where s_i=**Pick_random_split**(S,f_i), $\forall_i = 1, \ldots, n$;
Return a split s_* such that Score(s_*,S)=$max_{i=1,\ldots,K}$ Score(s_i,S).
3: **Pick_random_split**(S,f_i)
Input: a subset S and a feature f.
Output: a split Let f^S_{max} and f^S_{min} denote the maximal and minimal value of f in S;
Draw a random CutPoint Cut_f uniformly in $[f^S_{max}, f^S_{min}]$;
Return the split $[f < Cut_f]$.
4: **Stop_split**(S)
Input: a subset S, the minimum sample size for splitting a node m.
Output: a boolean IF $|S| < m$, THEN return TRUE;
IF all attributes are constant in S, THEN return TRUE;
IF the output is constant in S, then return TRUE;
OTHERWISE, return FALSE.

MLC4.5 algorithm

In [10] the authors adapted the Decision Trees C4.5 to the Multi-label problem by computing separately the Entropy for each label, and then sums all Entropies to decide to split or not at each node of the Tree. The main advantage is the identification of relevant features to all labels at once, nevertheless, the splitting criterion does not consider any Label correlation. The following pseudo-code 4 summarize the adaptation of C4.5 to ML framework.

Algorithm 4 Main steps of *MLC4.5*

Input: A and S are the attribute and the set of training examples being considered respectively.
S_v is the subset of S with value v for attribute A.

Output: Entropy (S)

1: C4.5 Tree is constructed top down.

2: For each node the best attribute is chosen using Information Gain formula as follows:

$$InformationGain(S, A) = Entropy(S) - \sum_{v \in A} \frac{|S_v|}{|S|} * Entropy(S_v)$$

3: Compute Entropy(S) as follows.

$$Entropy(S) = -\sum_{i=1}^{M} p(c_i) log(p(c_i) + (q(c_i) log(q(c_i))$$

.
$p(c_i)$=probability (relative frequence of class c_i) and $q(c_i) = 1 - p(c_i)$ is the probability of not being member of class c_i.

LaCova

In [129], the authors propose a DT based Multi-label classifier considering the covariance matrix as a splitting criterion at each node of the tree, then, it applies for independent labels a BR at leaves and if dependent, the labels are kept together as in LP. The main advantage of LaCova is the fact that it takes into consideration correlation among labels locally during the splitting process which allows choosing between a horizontal split (branching on a feature) and a vertical split (separating the labels). However, its main disadvantage is that the considered Label dependencies are Marginal since the covariance matrix is computed on labels without taking into consideration the features space.

To overcome this limit, the authors propose the clustering of labels dynamically during the construction of the Tree in [130] and [131] to model the Conditional Label dependence. However, LaCova still have some limitations, on the one hand, the high computational cost since it uses the covariance matrix at three points: to decide whether there is a split or not, in addition, it is used to find the best feature and the best cut for each split using a threshold computed dynamically. Moreover, adding the clustering of labels dynamically at each node cost more computational time. On the other hand, in order to predict labels for a new example, all generated Decision Trees in LaCovac should be tested, since each part of the Tree helps to predict some specific labels, hence, the final labels result from the aggregation of predicted labels in each branch. As consequence, the simplicity and the interpretability may be an issue as the generated Decision Trees may be complex for some real-world applications and especially the medical one.
The following pseudo codes 5, 6 and 7 summarize LaCova algorithm steps [129]:

Algorithm 5 LaCova(D): Learn a tree-Based ML classifier from training data.

Input: Dataset D, minimum number of instances m to split.
Output: Tree-Based ML classifier.

```
if SumOfVar(D)=0 or |D| < m then
  Leaf with relative frequencies of labels
else if SumOfAbsCov(D) ≤ λ(D) : then
  for each label j in D do
    T_j= Learn a Decision Tree for Single label j
  end for
  Return Node with Single label Decision Trees T_j
else
  f, {D_i}=FindBestSplit(D)
  for each child node D_i: do
    T_i = LaCova(D_i)
  end for
  Return Node splitting on f with subtrees T_i
end if
```

Algorithm 6 FindBestSplit(D): Find the best feature to split on.

Input: Dataset D
Output: Feature f, Data split into child nodes $\{D_i\}$

```
Initialize Q^best = ∞
for each feature f in D do
  if f is a numerical attribute then
    Cut_f = FindBestCut(D,f)
    Split D into child nodes {D_i} according to value of Cut_f
  else
    Split D into child nodes {D_i} according to values of f.
  end if
  for each child node {D_i} do
    Q_i=min(SumOfVar(D_i), SumOfAbsCov(D_i))
  end for
  Q = Σ_i (|D_i|/|D|) Q_i
  if Q < Q^best then
    Q^best, f^best, {D_i^best}=Q, f, {D_i}
  end if
end for
Return f^best, {D_i^best}
```

Algorithm 7 FindBestCut(D,f): Find the best cut-point to split the numerical attribute.

Input: Dataset D, Feature f.
Output: Best cut-point Cut_f^{best}.

```
Initialize Q^best = ∞
Sort D according to f in ascending manner
Find ll possible cut-points Cut
for each possible cut Cut_i in Cut do
  Split D into two child nodes {D_1}and{D_2} according to value of Cut_i.
  for each child node {D_i} do
    Q_i=min(SumOfVar(D_i), SumOfAbsCov(D_i))
    Q=∑_i (|D_i|/|D|) Q_i
    end for
    if Q < Q_f^best then
      Q_f^best=Q Cut_f^best=Cut_i
    end if
end for
Return Cut_f^best
```

The implementation of this algorithm was realized to the public by the main authors using JAVA. However, in this work, we implemented it using Python. For other algorithms, we used Scikit-learn framework [63]. More details about the experimental setup will be explained in the next section.

5 Experimental Setup

The present section is a comparative study between common algorithms from literature named: BR [11], CC [54], ECC [55] from Transformation methods and ERT [132], MLC4.5 [10] from Adaptation methods and finally LaCova algorithm [129] based covariance matrix as splitting criterion while constructing the Tree.

The main goal is to investigate the advantages of using Decision Trees algorithms as base classifier for both methods that takes into account Label correlation during the Classification or not, in addition for LaCova algorithm another point to study is the impact of using Label correlation as splitting criterion while constructing the Tree.
All the experiments was performed using 5 fold-cross validation using Six datasets: Yeast [20], Emotions [22], Scene [17], Medical [35], Genbase [42] and ABPM [4]. The Evaluations Measures used are: Accuracy [13], Exact Match [12] and Hamming Loss [13] (For more details about the datasets and Evaluations Measures, refer to Chapter 1).

Note that we used for the experiment Scikit-learn framework [63] for the following algorithms: BR, CC, ECC, ERT. The base classifier used is a Decision Tree which is an optimized version of CART provided by Scikit-learn. However, we implemented LaCova and MLC4.5 algorithms using Python by following the same format of Scikit-learn. The maximum depth of the Tree used as a base classifier is equal four for two main reasons, first, after many experiments we noticed a stabilization of predictive performances from depth=4, second, for good and easy interpretation of Trees especially for medical datasets

which is the optimal depth for that.

5.1 Results

The Tables 4.1, 4.2 and 4.3 shows the results obtained on six datasets using the following Evaluation Measures: Accuracy [13], Exact Match [12] and Hamming Loss [13]. Note that the numbers between brackets present rank of algorithms according to their predictive performances.

Table 4.1: *Comaprative results of six algorithms based Accuracy.*

	Datasets/Algorithms	BR	CC	ECC	ERT	MLC4.5	LaCova
Accuracy↑	Yeast	0.449(2)	0.433(3)	0.494(1)	0.388(6)	0.391 (5)	0.393(4)
	Scene	0.407(4)	0.466(1)	0.408(3)	0.166(6)	0.299(5)	0.460(2)
	Emotions	0.456(4)	0.493(1)	0.485(2)	0.364(6)	0.462(3)	0.367(5)
	Genbase	0.983(1)	0.982(2)	0.983(1)	0.039(5)	0.190(4)	0.793(3)
	Medical	0.757(3)	0.771(2)	0.773(1)	0.131(5)	0.116(6)	0.714(4)
	ABPM	0.977(3)	0.978(2)	0.983(1)	0.748(6)	0.758(5)	0.785(4)
	Average Rank	2.83(3)	1.83(2)	1.5(1)	5.66(6)	4.66(5)	3.66(4)

Table 4.2: *Comparative Results of six algorithms based Exact Match.*

	Datasets/Algorithms	BR	CC	ECC	ERT	MLC4.5	LaCova
Exact Match↑	Yeast	0.073(3)	0.141(2)	0.151(1)	0.062(4)	0.050 (5)	0.048(6)
	Scene	0.328(3)	0.396(1)	0.329(2)	0.161(6)	0.280(4)	0.260(5)
	Emotions	0.180(4)	0.222(2)	0.226(1)	0.156(5)	0.0254(6)	0.190(3)
	Genbase	0.965(1)	0.963(2)	0.965(1)	0.027(5)	0.176(4)	0.777(3)
	Medical	0.667(3)	0.696(1)	0.695(2)	0.105(5)	0.089(6)	0.634(4)
	ABPM	0.918(3)	0.922(2)	0.940(1)	0.244(6)	0.259(5)	0.303(4)
	Average Rank	2.83(3)	1.66(2)	1.33(1)	5.16(6)	5(5)	4.16(4)

Table 4.3: *Comparative Results of six algorithms based Hamming Loss.*

	Datasets/Algorithms	BR	CC	ECC	ERT	MLC4.5	LaCova
Hamming Loss ↓	Yeast	0.220(1)	0.221(2)	0.229(5)	0.226(3)	0.239(6)	0.228(4)
	Scene	0.154(1)	0.171(3)	0.180(4)	0.165(2)	0.184(5)	0.235(6)
	Emotions	0.245(2)	0.241(1)	0.250(3)	0.267(5)	0.245(2)	0.261(4)
	Genbase	0.0018(2)	0.0019(3)	0.0016(1)	0.043(6)	0.040(5)	0.017(4)
	Medical	0.0105(3)	0.0102(2)	0.0101(1)	0.0259(6)	0.0252(5)	0.014(4)
	ABPM	0.0179(3)	0.0172(2)	0.013(1)	0.201(6)	0.196(5)	0.163(4)
	Average Rank	2(1)	2.16(2)	2.5(3)	4.66(5)	4.66(5)	4.33(4)

5.2 Discussion

The idea behind the choice of the previous six algorithms from the literature is to investigate the impact of using several splitting criteria for the Decision Trees used as a base classifier. The first category concerns Transformation methods such as BR that ignores totally dependence between labels, on the other hand, we used CC and ECC that consider Label dependencies by using a chain of Decision Trees classifiers. The splitting criterion, in this case, is Gini index since we used the optimized version of CART provided by Scikit learn as a base classifier.

The second category of methods concerns the adaptation strategy that includes MLC4.5 and ERT that ignore Label dependencies. For the former, the splitting criterion is the Information Gain, while for the later random choice of the feature for a split is used.

The last studied algorithm is LaCova that consider Label dependence as splitting criterion while growing the Tree. We discussed in the following the results found in Tables 4.1, 4.2 and 4.3 based on the above points.

We are aware that those initial results may be interpreted with caution since another work is in progress to extend the comparison using more datasets. However, by comparing ranks of algorithms based on Accuracy, Exact Match and Hamming Loss using the actual results from the six datasets. We note that ECC and CC based Decision Trees outperform other algorithms. Nevertheless, by comparing them to literature we note that results are a bit lower in some cases, that can conduct us to conclude that the use of Decision Trees as a base classifier is good for interpretation. Although, if we are interested in higher predictive performances, other base classifiers as Logistic Regression, KNN, Naive Bayes etc. may be a good option for some applications, in which the interpretability of the model is not required.

Moreover, the idea of extracting decision rules from DT in CC and ECC in future using specialized approaches from Association Classification [133], a task that was widely studied to take advantages of Association and Classification rule mining. The first one can define all important associations between the different variables of the dataset, whereas the second defines the importance of using such variables to predict predetermined targets.

By comparing results from MLC4.5 and ERT, we note that the use of Information Gain as a splitting criterion is more interesting than the random choice of features for constructing the Tree. However, results from CC, ECC and BR that use a Gini index to grow the tree was better than those provided by MLC4.5. Note that, this choice between those two splitting criteria was widely discussed in the literature, many researchers concluded that testing both of them on the dataset can help to choose the best one since Gini Index is calculated by subtracting the sum of the squared probabilities of each class from one and favors larger partitions. However, Information Gain favors smaller partitions with many distinct values. We remarked also that growing the Tree with more depths for MLC4.5 could improve the results.

For LaCova, according to the results on the six datasets, we note that it outperforms significantly against MLC4.5 and ERT in terms of Accuracy, Hamming Loss and Exact Match. However, it is always ranked after ECC, CC and BR. More experiments are in progress to affirm these findings. Another point that we intend to address in future is the type of Label dependence studied. LaCova uses Marginal Label dependence since it computes covariance matrix on only the output space to determine whether the labels are dependent or not, and based on a threshold, it decides to split the node (More details about the algorithm were previously explained in Section 4). The use of Conditional Label dependence could improve greatly predictive performances by taking the features space into consideration because in many real-world applications possible correlations are not only between labels but also significant dependences may be present between features and labels of the dataset.

A simple example from our previous study on ABPM dataset [4] shows the importance

of Conditional dependence, as we noted in Chapter 3, section 7.1 that there is a strong correlation between the *Validity* label and the two attributes: *Perc* and *HRecord*. The two attributes are the most relevant to judge the validity of *ABPM* records.

5.3 Study limitations and future work

We are aware that this initial study is limited and many points are likely to be considered in the near future as follow:

- More experiments should be conducted using more datasets from literature to validate the results.
- Another work is in progress to improve the performance of LaCova algorithm using Conditional Label dependence instead of Marginal Label dependence while splitting the Tree. The use of other probabilistic functions to test as a splitting criterion is a part of our perspectives. Finally, the use of frequent labels at leaves to annotate the leave could be improved using other strategies that consider Label dependence at this level too.
- The interpretation of generated Decision Tree of winning algorithms especially on ABPM dataset is one of the main goals of our further work.

6 Conclusion

This chapter studied Label dependence issue in *Multi-label* Classification (*MLC*). Two major types were highlighted, named: Conditional and Marginal Label dependence with their major differences. Likewise, we reviewed recent works addressing Label correlation based on several *Multi-label* algorithms, including *Transformation* methods and *Adaptation* algorithms. We focused on the use of Decision Trees as a base classifier and its main advantages, for that, a comparative study were conducted using six well-known algorithms in the literature named: BR [11], CC [54], ECC [55], ERT [132], MLC4.5 [10], LaCova [129]. Six datasets were used for the experiments, named: *Yeast* [20], *Scene* [17], *Emotions* [22], *Genbase* [42] and *Medical* [35], we tested also our collected dataset named *ABPM* [4] for the experiments (refer to Chapter 3 for more details). Finally, study limitations and future work were presented.

Conclusions and Future Directions

The present thesis focused on Multi-label Learning and its main applications, especially to the medical field, in which many real problems often encountered by physicians needs Multi-label tools to be solved. The problem of learning is called Multi-label (ML) if the instances of learning are associated with multiple target labels at once.
In this manuscript, we presented first a detailed overview on Multi-label Learning framework including its main applications, the tools and popular strategies used for learning from ML data and finally the current challenges recently studied by researchers in this field.

Then, we investigated the benefits of using Ensemble Methods for the Multi-label framework based on two major strategies: *Bagging* [2] and *Boosting* [3]. The goal was the improvements of *MLKNN* algorithm [1] that adapts *KNN* to Multi-label data. The use of homogeneous ensemble methods (*Bagging* and *Boosting*) provide competitive results and affirm the hypothesis that using several ML learners simultaneously for the prediction of labels improve greatly the performance of the individual classifier.

Many points are likely to be considered as part of future work for this first contribution,

- The improvement of *Bagged MLKNN* by using variable selection methods, in order to identify the relevant variables for each label given a dataset. Such a task is really important for the medical applications since the physician prefer to have a good compromise between the performance and interpretability of the medical aid diagnosis system.

- We plan also to evaluate the present algorithm on a new medical dataset that we gathered recently as the second contribution of this work, it concerns Ambulatory Blood Pressure Monitoring (ABPM) [4].

Our second contribution includes two major parts:

- The first part is a new Multi-label dataset with 40 *ABPM* features for 270 numeric patient records categorized into one or more out of 6 labels named: *Validity*, *Circadian Rhythm*, *BPV*, *PP*, *BPL*, *MS*. The dataset is released to the public [4], in order to allow comparative experiments by other researchers and especially medical researchers while the publicly available Multi-label medical datasets are very rare.

- The second part is the intelligent analysis of *ABPM* records using Multi-label Classification algorithms. The medical diagnostic process supported by such techniques constitutes a modern and useful tool for medical aid decision, allowing the expert to

analyze *ABPM* records more quickly and efficiently. A comparative study of seven algorithms was conducted, also, an analysis of dependencies between the labels and attributes of the *ABPM* dataset using Decisions Trees was discussed.

Further works are likely to be considered for this work:

- The collected dataset contains only 270 records which limit the interpretation of results, we intend to expand our database with more examples of learning, by taking the case of Multi-dimensional data into consideration for the two labels: *Blood Pressure Load* and *Circadian Rhythm* which contains more than one class for each label.

- ABPM dataset contains only patients with pathological *BPV* since the study was developed in cardiology service. However, we intend to enrich it with the non-pathological cases in the near future in order to improve the learning process for this label.

- More data are available to expand the dataset with more examples. However, the manual annotation by the doctor takes huge time and efforts. For that reason, we intend to use semi-supervised approaches for the annotation and the current ABPM dataset could be very useful for the learning process to label new records.

- The collected dataset contains only patients with pathological *BPV* since the study was developed in cardiology service and the pathological cases are more present than the normal one. We propose the use of Imbalance data solutions in Multi-label that was greatly addressed in the literature to balance label distributions [117], [134].

- The study of Label dependence approaches for ABPM dataset since the initial results on Conditional dependence analysis of *ABPM* labels was satisfactory and encourage us to expand our study in future to extract more relevant information and the correlation between labels and features of the dataset.

- Results show that the Multi-label modeling of *ABPM* data helps to investigate label dependencies and provide interesting insights, which can be integrated into the *ABPM* devices to dispense automatically detailed reports with possible future complications.

The last issue in Multi-label Learning that we studied is the use of Decision Trees (DT) to extract new and implicit correlations between different labels and features in a given dataset. The satisfactory results and interpretations that we got in the previous contribution show graphically a very interesting correlations between labels, and encouraged us to continue using this strategy of learning that fulfilled important criteria in the critical applications, especially in the medical field, where one of the main condition of the expert to use an automated system to improve the diagnosis process is that should be a white box and interpretable.

For that, we reviewed recent works addressing Label dependence based on several Multi-label algorithms, including Transformation methods and Adaptation algorithms based on DT. We presented also the main differences between the two defined types of Label correlation named Conditional and Unconditional (Marginal). Finally, we conducted a comparative study of six well-known algorithms in the literature based Decision Trees, and we discuss the benefits of considering Label dependence using Decision Trees algorithm as a base classifier for both Transformation and Adaptation algorithms. First results of this study were presented, however, we are aware that are limited and should be interpreted with caution since the complete comparative results is in progress and many points are likely to be considered in the near future.

- More experiments should be conducted using more datasets from literature to validate the results.
- May experiments are in progress to improve the performance of LaCova algorithm using Conditional Label dependence instead of Marginal Label dependence while splitting the tree. The use of other probabilistic functions to test as a splitting criterion is a part of our perspectives. Finally, the use of frequent labels to annotate the leaves could be improved using other strategies of learning to consider Label dependence at this level too.
- Highlighting label dependencies in ABPM dataset is one of our major research goals. However, the current version of ABPM dataset is not too large and with few examples of learning the Decision Trees may not give really valuable interpretations. For that, we intend to interpret the generated trees after expanding the dataset with more examples of learning.

The several published contributions during the PhD study are listed in the following:

- Thee first work [135] was published in Proceeding ICCDA '17 Proceedings of the International Conference on Compute and Data Analysis. K. DOUIBI, N. SETTOUTI and MA. CHIKH. *The homogeneous Ensemble Methods for MLKNN algorithm.* Pages 197-201, Lakeland, FL, USA - May 19 - 23, 2017, ACM New York, NY, USA ©2017, DOI : 10.1145/3093241.3093262.

- Second work [5] was published in Australasian Physical & Engineering Sciences in Medecine. Douibi, K., Settouti, N., Chikh, M. A., Read, J., & Benabid, M. M. (2018). An analysis of ambulatory blood pressure monitoring using Multi-label Classification. Australasian physical & engineering sciences in medicine, 1-17., DOI 10.1007/s13246-018-0713-0.

- The dataset of the above work [4] was published online Mendeley repository (Elsevier) at `http://dx.doi.org/10.17632/y4dh3b3tfx.1`. Douibi, k., Benabid, M. M., Settouti, N., Chikh, M. A.(2017), "Data for: An analysis of Ambulatory Blood Pressure Monitoring (ABPM).", Mendeley Data, v1.

- Our last contribution about using Decisions Trees for modeling Label dependence for medical application is in progress.

Bibliography

[1] M. L. Zhang and Z. H. Zhou, "Ml-knn: A lazy learning approach to multi-label learning," *PatternRecognition*, vol. 40(7), pp. 2038–2048, 2007.

[2] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24(2), pp. 123–140, 1996.

[3] A. Cornuéjols and L. Miclet, *Apprentissage artificiel: concepts et algorithmes*, 2010.

[4] K. Douibi, M. Benabid, N. Settouti, and M. Chikh, "Data for: An analysis of ambulatory blood pressure monitoring (abpm)," *Mendeley Data, v1, http://dx.doi.org/10.17632/y4dh3b3tfx.1*, 2017.

[5] K. Douibi, N. Settouti, M. A. Chikh, J. Read, and M. M. Benabid, "An analysis of ambulatory blood pressure monitoring using multi-label classification," *Australasian physical & engineering sciences in medicine*, pp. 1–17, 2018.

[6] T. Dietterich, "Ensemble methods in machine learning," *Multiple Classifier Systems*, 2000.

[7] N. Gobin, G. Wuerzener, B. Waeber, and M. Burnier, "Mesure ambulatoire de la pression artérielle sur 24 heures," *Forum Med Suisse*, vol. 12(3132), pp. 600–607, 2012.

[8] A. Elsayed, F. Coenen, M. García, and V. Sluming, "Segmentation for medical image mining: A technical report," *The University of Liverpool, Liverpool L69 3BX, UK*, 2008.

[9] M. L. Zhang, J. M. Pe na, and V. Robles, "Feature selection for multi-label naive bayes classification," *Information Sciences*, vol. 179, pp. 3218–3229, 2009.

[10] A. Clare and R. D. King, "Knowledge discovery in multi-label phenotype data," *Proceedings of the 5th European Conference on PKDD*, pp. 42–53, 2001.

[11] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Random k-labelsets for multi-label classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23(7), pp. 1079–1089, 2011.

[12] M. Gjorgji, K. Dragi, G. Dejan, and D. Saso, "An extensive experimental comparison of methods for multi-label learning," *Pattern Recognition*, vol. 45(9), pp. 3084–3104, 2012.

[13] G. Tsoumakas and I. Katakis, "Multi label classification: an overview," *International Journal of Data Warehouse and Mining*, vol. 3 (3), pp. 1–13, 2007.

[14] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," *AI magazine*, vol. 17, no. 3, pp. 37, 1996.

[15] A. McCallum, "Multi-label text classification with a mixture model trained by em," in *AAAI workshop on Text Learning*, 1999, pp. 1–7.

[16] R. E. Schapire and Y. Singer, "Improved boosting algorithms using confidence-rated predictions," *Machine Learning*, vol. 37(3), 1999.

[17] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown, "Learning multi-label scene classification," *Pattern recognition*, vol. 37, no. 9, pp. 1757–1771, 2004.

[18] S. Yang, S. Kim, and Y. Ro, "Semantic home photo categorization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 3, pp. 324–335, 2007.

[19] G. Qi, X. Hua, Y. Rui, J. Tang, T. Mei, and H. Zhang, "Correlative multi-label video annotation," in *Proceedings of the 15th ACM international conference on Multimedia*. ACM, 2007, pp. 17–26.

[20] A. Elisseef, J. Weston, T. G. Dietterich, S. Becker, and Z. Ghahramani, "A kernel method for multi-labelled classification," *Advances in Neural Information Processing Systems 14, MIT Press, Cambridge*, pp. 681–687, 2002.

[21] T. Li and M. Ogihara, "Detecting emotion in music," 2003.

[22] K. Trohidis, G. Tsoumakas, G. Kalliris, and I. Vlahavas, "Multilabel classification of music into emotions," 2008, pp. 325–330.

[23] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Mining multi-label data," *In Data Mining and Knowledge Discovery Handbook*, pp. 667–685, 2010.

[24] R. E. Schapire and Y. Singer, "Boostexter: A boosting-based system for text categorization," in *Machine learning*, 2000, pp. 135–168.

[25] N. Ueda and K. Saito, "Parametric mixture models for multi-labeled text," in *Advances in neural information processing systems*, 2003, pp. 737–744.

[26] S. Godbole and S. Sarawagi, "Discriminative methods for multi-labeled classification," in *Pacific-Asia conference on knowledge discovery and data mining*. Springer, 2004, pp. 22–30.

[27] J. Rousu, C. Saunders, S. Szedmak, and J. Shawe-Taylor, "Kernel-based learning of hierarchical multilabel classification models," *Journal of Machine Learning Research*, vol. 7, no. Jul, pp. 1601–1626, 2006.

[28] S. Zhu, X. Ji, W. Xu, and Y. Gong, "Multi-labelled classification using maximum entropy method," in *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2005, pp. 274–281.

[29] E. L. Mencia and J. Fürnkranz, "Efficient pairwise multilabel classification for large-scale problems in the legal domain," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2008, pp. 50–65.

[30] R. Moskovitch, S. Cohen-Kashi, U. Dror, I. Levy, A. Maimon, and Y. Shahar, "Multiple hierarchical classification of free-text clinical guidelines," *Artificial Intelligence in Medicine*, vol. 37, no. 3, pp. 177–190, 2006.

[31] J. P. Pestian, C. Brew, P. Matykiewicz, D. J. Hovermale, N. Johnson, K. B. Cohen, and W. Duch, "A shared task involving multi-label classification of clinical free text," in *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*. Association for Computational Linguistics, 2007, pp. 97–104.

[32] A. P. Streich and J. M. Buhmann, "Classification of multi-labeled data: a generative approach," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2008, pp. 390–405.

[33] I. Katakis and G. Tsoumakasand I. Vlahavas, "Multilabel text classification for automated tag suggestion," in *Proceedings of the ECML/PKDD*, 2008, vol. 18.

[34] F. Herrera, F. Charte, A. J. Rivera, and M. J. Del Jesus, *Multilabel Classification: Problem Analysis, Metrics and Techniques*, Springer, 2016.

[35] K. Crammer, M. Dredze, K. Ganchev, P. Talukdar, and S. Carroll, "Automatic code assignment to medical text," in *Proceedings of the workshop on bionlp 2007: Biological, translational, and clinical language processing*. Association for Computational Linguistics, 2007, pp. 129–136.

[36] B. Klimt and Y. Yang, "The enron corpus: a new dataset for email classification research," in *European Conference on Machine Learning*. Springer, 2004, pp. 217–226.

[37] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Effective and efficient multilabel classification in domains with large number of labels," *Proc. ECML/PKDD Workshop on Mining Multidimensional Data (MMD08)*, vol. 61, 2008.

[38] J. Read, *Scalable multi-label classification*, Ph.D. thesis, University of Waikato, 2010.

[39] C. Snoek, M. Worring, J. Van Gemert, J. Geusebroek, and A. Smeulders, "The challenge problem for automated detection of 101 semantic concepts in multimedia," in *Proceedings of the 14th ACM international conference on Multimedia*. ACM, 2006, pp. 421–430.

[40] F. Briggs, B. Lakshminarayanan, L. Neal, X. Fern, R. Raich, S. Hadley, A. Hadley, and M. Betts, "Acoustic classification of multiple simultaneous bird species: A multi-instance multi-label approach," *The Journal of the Acoustical Society of America*, vol. 131, no. 6, pp. 4640–4650, 2012.

[41] E. Goncalves, A. Plastino, and A. A. Freitas, "A genetic algorithm for optimizing the label ordering in multi-label classifier chains," in *Tools with Artificial Intelligence (ICTAI), 2013 IEEE 25th International Conference on*. IEEE, 2013, pp. 469–476.

[42] S. Diplaris, G. Tsoumakas, P. Mitkas, and I. Vlahavas, "Protein classification with multiple algorithms," *Proceedings of the 10th Panhellenic Conference on Informatics (PCI) Greece*, pp. 448–456, 2005.

[43] G. Tsoumakas, X. Spyromitros, J. Vilcek, and I. Vlahavas, "Mulan: A java library for multi-label learning," *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2411–2414, 2011.

[44] M. Zhang and Z. Zhou, "A k-nearest neighbor based algorithm for multi-label classification," in *Granular Computing, 2005 IEEE International Conference on*. IEEE, 2005, vol. 2, pp. 718–721.

[45] Z. Younes, F. Abdallah, T. Denoeux, and H. Snoussi, "A dependent multi-label classification method derived from the k-nearest neighbor rule," *EURASIP Journal on Advances in Signal Processing*, p. 14, 2011.

[46] W. Cheng and E. Hüllermeier, "Combining instance-based learning and logistic regression for multilabel classification," *Machine Learning*, vol. 76, no. 2-3, pp. 211–225, 2009.

[47] E. Spyromitros, G. Tsoumakas, and I. Vlahavas, "An empirical study of lazy multilabel classification algorithms," in *Hellenic conference on artificial intelligence*. Springer, 2008, pp. 401–406.

[48] M. L. Zhang and Z. H. Zhou, "Multi-label neural networks with applications to functional genomics and text categorization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18(10), pp. 1338–1351, 2006.

[49] K. Crammer and Y. Singer, "A family of additive online algorithms for category ranking," *Journal of Machine Learning Research*, vol. 3, no. Feb, pp. 1025–1058, 2003.

[50] M. L. Zhang, "M l-rbf: Rbf neural networks for multi-label learning," *Neural Processing Letters*, vol. 29, no. 2, pp. 61–74, 2009.

[51] P. M. Ciarelli, E. Oliveira, C. Badue, and A. F. De Souza, "Multi-label text categorization using a probabilistic neural network," *International Journal of Computer Information Systems and Industrial Management Applications*, vol. 1, no. 133-144, pp. 40, 2009.

[52] H. Blockeel, L. D. Raedt, and J. Ramon, "Top-down induction of clustering trees," *arXiv preprint cs/0011032*, 2000.

[53] J. Furnkranz, E. Hullermeier, E. L. Mencia, and K. Brinker, "Multilabel classification via calibrated label ranking," *Machine Learning*, vol. 73(2), pp. 133–153, 2008.

[54] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," *Machine Learning and Knowledge Discovery in Databases ,Lecture Notes in Computer Science*, vol. 5782, pp. pp 254–269, 2009.

[55] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," *Machine learning*, vol. 85, no. 3, pp. 333, 2011.

[56] K. Dembczynski, W. Cheng, and E. Hüllermeier, "Bayes optimal multilabel classification via probabilistic classifier chains.," in *ICML*, 2010, vol. 10, pp. 279–286.

[57] E. C. Gonçalves, A. Plastino, and A. A. Freitas, "Simpler is better: a novel genetic algorithm to induce compact multi-label chain classifiers," in *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation*. ACM, 2015, pp. 559–566.

[58] J. Lee, H. Kim, N. R. Kim, and J. H. Lee, "An approach for multi-label classification by directed acyclic graph with label correlation maximization," *Information Sciences, http://dx.doi.org/10.1016/j.ins.2016.02.037*, 2016.

[59] L. Chekina, D. Gutfreund, A. Kontorovich, L. Rokach, and B. Shapira, "Exploiting label dependencies for improved sample complexity," *Machine learning*, vol. 91, no. 1, pp. 1–42, 2013.

[60] E. Gibaja and S. Ventura, "Multi-label learning: a review of the state of the art and ongoing research," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 4, no. 6, pp. 411–444, 2014.

[61] J. Read, P. Reutemann, B. Pfahringer, and G. Holmes, "Meka: a multi-label/multi-target extension to weka," *Journal of Machine Learning Research*, vol. 17(21), pp. 1–5, 2016.

[62] P. Szymański and T. Kajdanowicz, "A scikit-based python environment for performing multi-label classification," *arXiv preprint arXiv:1702.01460*, 2017.

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

[64] F. Charte, D. Charte, A. Rivera, del M. Jesus, and F. Herrera, "R ultimate multilabel dataset repository," in *International Conference on Hybrid Artificial Intelligence Systems*. Springer, 2016, pp. 487–499.

[65] C. Chang and C. Lin, "Libsvm: a library for support vector machines," *ACM transactions on intelligent systems and technology (TIST)*, vol. 2, no. 3, pp. 27, 2011.

[66] J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, S. García, L. Sánchez, and F. Herrera, "Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework.," *Journal of Multiple-Valued Logic & Soft Computing*, vol. 17, 2011.

[67] L. Keigoandand K. Mineichi, "Mlc toolbox: A matlab/octave library for multi-label classification," *CoRR*, vol. abs/1704.02592, 2017.

[68] G. James, D. Wittenand T. Hastie, and R. Tibshirani, *An introduction to statistical learning*, vol. 112, Springer, 2013.

[69] L. K. Hansen and P. Salamon, "Neural network ensembles," *IEEE transactions on pattern analysis and machine intelligence*, vol. 12, no. 10, pp. 993–1001, 1990.

[70] B. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap*, Chapman and Hall, New York, NY, 1993.

[71] S. Caron, "Une introduction aux arbres de decision http://scaron.info," 2011.

[72] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *Proceedings of the Thirteenth International Conference on Machine Learning (ICML 1996)*, Lorenza Saitta, Ed. 1996, 1-55860-419-7, pp. 148–156, Morgan Kaufmann.

[73] O. Gharroudi, *Ensemble multi-label learning in supervised and semi-supervised settings*, Ph.D. thesis, Université de Lyon, 2017.

[74] J. Read, B. Pfahringer, and G. Holmes, "Multi-label classification using ensembles of pruned sets," *Proceedings of the 8th IEEE International Conference on Data Mining*, pp. 995–1000, 2008.

[75] L. Breiman, "Random forests," *Machine Learning*, vol. 45(1), pp. 5–32, 2001.

[76] D. Kocev, C. Vens, J. Struyf, and S. Džeroski, "Ensembles of multi-objective decision trees," in *Proceedings of the 18th European Conference on Machine Learning*, Berlin, Heidelberg, 2007, ECML '07, pp. 624–631, Springer-Verlag.

[77] S. Chuan, K. Xiangnan, S. Y. Philip, and W. Bai, "Multi-label ensemble learning„" *Machine Learning and Knowledge Discovery in Databases Lecture Notes in Computer Science*, vol. 6913, pp. 223–239, 2011.

[78] R. Li, W. Liu, Y. Lin, H. Zhao, and C. Zhang, "An ensemble multilabel classification for disease risk prediction," *Journal of healthcare engineering*, vol. 2017, 2017.

[79] J. Chen, X. Zhou, and Z.s Wu, "A multi-label chinese text categorization system based on boosting algorithm," 2004, pp. 1153–1158.

[80] G. Biau, F. Cerou, and A. Guyader, "On the rate of convergence of the bagged nearest neighbor estimate," *Journal of Machine Learning Research*, vol. 11, pp. 687–712, 2010.

[81] A. W. Aha, D. Kibler, and M. K. Albert, "Instance-based learning algorithms," *Machine learning*, vol. 6, no. 1, pp. 37–66, 1991.

[82] E. Corbett, "clinical-applications-of-machine-learning-in-healthcare," https://www.healthcatalyst.com/clinical-applications-of-machine-learning-in-healthcare, Nov. 2018, consulted, 11/18/2018.

[83] K. Benkeand and G. Benke, "Artificial intelligence and big data in public health," *International Journal of Environmental Research and Public Health*, vol. 15, no. 12, pp. 2796, 2018.

[84] F. Cabitza, R. Rasoini, and G. Gensini, "Unintended consequences of machine learning in medicine," *Jama*, vol. 318, no. 6, pp. 517–518, 2017.

[85] J. A. Whitworth and WHO. Organization, "International society of hypertension writing group: 2003 world health organization (who)/ international society of hypertension (ish) statement on management of hypertension," *Journal of hypertension*, vol. 21(11), pp. 1983–1992, 2003.

[86] J. F. Vilela-Martin, R. O. Vaz de Melo, C. H. Kuniyoshi, A. N. Abdo, and J. C. Yugar-Toledo, "Hypertensive crisis: clinical-epidemiological profile," *Hypertension Research*, vol. 34(3), pp. 367–371, 2011.

[87] S. Motamed and A. Pechére-Bertschi, "Hypertension artérielle," *Department of Primary Care, HUG Arterial Hypertension Unit, SMPR, HUG*, 2013.

[88] A. Zanchetti, "The role of ambulatory blood pressure monitoring in clinical practice," *American Journal of Hypertension, https://doi.org/10.1016/S0895-7061(97)00270-7*, vol. 10(9), pp. 1069–1080, 1997.

[89] D. L. Clement, De .M. L Buyzere, De. D. A. Bacquer, P. W. Leeuw, D. A. Duprez, R. H. Fagard, and P. Van Der Niepen, "Pronostic value of ambulatory blood-pressure recordings in patients with hypertension," *The new england journal of medecine*, vol. 348, pp. 2407–2415, 2003.

[90] F. Kanoun, N. Ben Alaya, S. driss, N. Sayem, M. Chihaoui, F. Harzallah, and H. Slimane, "Appréciation du profil tensionnel par mesure ambulatoire de la pression artérielle chez les diabétiques hypertendus traités," *La tunisie Médicale*, vol. 88(12), pp. 885–889, 2010.

[91] E. Ngendakumana and M. El hattaoui, *Evaluation du controle de l'Hypertension artérielle par la MAPA chez les patients diabètiques hypertendus*, Ph.D. thesis, Cardiology Department: Ibn Tofail Hospital. CHU Mohammed VI. Marrakech, 2014.

[92] S. D. Pierdomenico and F. Cuccurullo, "Ambulatory blood pressure monitoring in type 2 diabetes and metabolic syndrome: a review," *Blood Press Monit*, vol. 15(1), pp. 1–7, 2010.

[93] L. I. Guo-Zheng, H. E. Zehui, and S. Feng-Feng, "Patient classification of hypertension in traditional chinese medicine using multi-label learning techniques," *BMC Medical Genomics*, vol. 8(3), pp. 1, 2015.

[94] M. Jiang, C. Lu, C. Zhang, J. Yang, Y. Tan, A. Lu, and K. Chan, "Syndrome differentiation in modern research of traditional chinese medicine," *Journal of Ethnopharmacology*, vol. 140, no. 3, pp. 634–642, 2012.

[95] A. Copetti, O. Loques, J. C. Leite, T. P. Barbosa, and A. C. da Nobrega, "Intelligent context-aware monitoring of hypertensive patients," *In 2009 3rd International Conference on Pervasive Computing Technologies for Healthcare IEEE*, pp. 1–6, 2009.

[96] S. Pierdomenico, A. Pierdomenico, F. Coccina, D. Lapenna, and E. Porreca, "Prognostic value of nondipping and morning surge in elderly treated hypertensive patients with controlled ambulatory blood pressure," *American journal of hypertension*, vol. 30, no. 2, pp. 159–165, 2017.

[97] C. Neves, J. Bastos, J. Pires, and J. Polónia, "Ambulatory blood pressure monitoring after one cardiovascular event in prediction of a second cardiovascular event," *Journal of Hypertension*, vol. 36, pp. e105, 2018.

[98] J. Flynn, S. Daniels, L. Hayman, D. Maahs, B. McCrindle, M. Mitsnefes, J. Zachariah, and E. Urbina, "Update: ambulatory blood pressure monitoring in children and adolescents: a scientific statement from the american heart association," *Hypertension*, vol. 63, no. 5, pp. 1116–1135, 2014.

[99] L. Mena, E. Orozco, V. Felix, R. Ostos, J. Melgarejo, and G. Maestre, "Machine learning approach to extract diagnostic and prognostic thresholds: application in prognosis of cardiovascular mortality," *Computational and mathematical methods in medicine*, vol. 2012, 2012.

[100] E. OBrien, R. Asmar, and L. Beilin, "European society of hypertension recommendations for conventional, ambulatory and home blood pressure measurement," *J Hypertens*, vol. 21, pp. 821–48, 2003.

[101] A. Pechère-Bertschi, Y. Michel, H. Brandstatter, F. Muggli, and J. M Gaspoz, "Lecture de la mesure ambulatoire de la pression artérielle (mapa) par le médecin de premier recours.," *Rev Med Suisse*, vol. 5, pp. 1876–1880, 2009.

[102] D. P. Papadopoulos and T. K. Makris, "Masked hypertension definition, impact, outcomes: a critical review," *The Journal of Clinical Hypertension*, vol. 9, pp. 956–963, 2007.

[103] E. O'Brien, A. Coats, and P. Owens, "Use and interpretation of ambulatory blood pressure monitoring: recommendations of the british hypertension society," *BMJ*, vol. 320, pp. 1128–34, 2000.

[104] E. O'Brien, B. Waeber, and G. Parati, "European society of hypertension recommendations on blood pressure measuring devices," *BMJ*, vol. 322, pp. 532–6, 2001.

[105] K. Kario, T. G. Pickering, Y. Umeda, S. Hoshide, Y. Hoshide, M. Morinari, M. Murata, T. Kuroda, J. E. Schwartz, and K. Shimada, "Morning surge in blood pressure as a predictor of silent and clinical cerebrovascular disease in elderly hypertensives," *Circulation 20031071401–1405.*, vol. 107(10), pp. 1401–1406, 2003.

[106] P. Gosse, R. Lasserre, C. Minifie, P. Lemetayer, and J. Clementy, "Blood pressure surge on rising. journal of hypertension," vol. 22(6), pp. 1113–1118, 2004.

[107] K. Madin and P. Iqbal., "Twenty four hour ambulatory blood pressure monitoring: A new tool for determining cardiovascular prognosis," *Postgraduate Medical Journal 82.971. PMC. Web. 21 Nov. 2017*, vol. 548-551, 2006.

[108] E. Gibaja and S. Ventura, "A tutorial on multi-label learning," *ACM Comput. Surv*, vol. 47(3), 2015.

[109] M. L. Zhang and Z. Zhou, "A review on multi-label learning algorithms," *IEEE transactions on knowledge and data engineering*, vol. 26(8), pp. 1819–1837, 2014.

[110] K. Trohidis, G. Tsoumakas, G. Kalliris, and I. Vlahavas, "Multi-label classification of music by emotion," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 1, pp. 1, 2011.

[111] A. Aldrees and A. Chikh, "Comparative evaluation of four multi-label classification algorithms in classifying learning objects," *Computer Applications in Engineering Education*, 2016.

[112] V. Batista, F. Pintado, A. B. Gil, V. Rodriguez, and M. Moreno, "A system for multi-label classification of learning objects," *6th International Conference SOCO 2011 Advances in Intelligent and Soft Computing*, vol. 87, pp. 523–531, 2011.

[113] H. Modi and M. Panchal, "Experimental comparison of different problem transformation methods for multi-label classification using meka," *International Journal of Computer Applications*, vol. 59, pp. 15, 2012.

[114] P. K. Zachariah, S. G. Sheps, D. M. Ilstrup, C. R. Long, K. R. Bailey, C. M. Wiltgen, and C. A. Carlson, "Blood pressure load a better determinant of hypertension," *In Mayo Clinic Proceedings Elsevier*, vol. 63(11), pp. 1085–1091, 1988.

[115] E. Alvares-Cherman, J. Metz, and M. C. Monard, "Incorporating label dependency into the binary relevance framework for multi-label classification," *Expert Systems with Applications*, vol. 39, no. 2, pp. 1647–1655, 2012.

[116] M. L. Zhang and K. Zhang, "Multi-label learning by exploiting label dependency," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2010, pp. 999–1008.

[117] D. Krzysztof, W. Willem, C. Weiwei, and H. Eyke, "On label dependence and loss minimization in multi-label classification," *Machine Learning*, vol. 88, no. 1-2, pp. 5–45, 2012.

[118] D. J. Hsu, S. M. Kakade, J. Langford, and T. Zhang, "Multi-label prediction via compressed sensing," in *Advances in neural information processing systems*, 2009, pp. 772–780.

[119] J. Read, A. Bifet, G. Holmes, and B. Pfahringer, "Scalable and efficient multi-label classification for evolving data streams," *Machine Learning*, vol. 88, no. 1-2, pp. 243–272, 2012.

[120] F. De Comité, R. Gilleron, and M.Tommasi, "Learning multi-label alternating decision trees from texts and data," in *International Workshop on Machine Learning and Data Mining in Pattern Recognition*. Springer, 2003, pp. 35–49.

[121] N. Ghamrawi and A. McCallum, "Collective multi-label classification," in *Proceedings of the 14th ACM international conference on Information and knowledge management*. ACM, 2005, pp. 195–200.

[122] L. E. Sucar, C. Bielza, E. F. Morales, P. Hernandez-Leal, J. H. Zaragoza, and P. Larrañaga, "Multi-label classification with bayesian network-based chain classifiers," *Pattern Recognition Letters*, vol. 41, pp. 14–22, 2014.

[123] G. Tsoumakas, A. Dimou, E. Spyromitros, V. Mezaris, I. Kompatsiaris, and I. Vlahavas, "Correlation-based pruning of stacked binary relevance models for multi-label learning," in *Proceedings of the 1st International Workshop on Learning from Multi-label Data*, 2009, pp. 101–116.

[124] D. H. Wolpert, "Stacked generalization," *Neural networks*, vol. 5, no. 2, pp. 241–259, 1992.

[125] L. Tenenboim, L. Rokach, and B. Shapira, "Multi-label classification by analyzing labels dependencies," in *Proceedings of the 1st international workshop on learning from multi-label data, Bled, Slovenia*, 2009, pp. 117–132.

[126] J. Xu, "Constructing a fast algorithm for multi-label classification with support vector data description," in *Granular Computing (GrC), 2010 IEEE International Conference on*. IEEE, 2010, pp. 817–821.

[127] D. Gjorgjevikj, G. Madjarov, and S. DŽEROSKI, "Hybrid decision tree architecture utilizing local svms for efficient multi-label learning," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 27, no. 07, pp. 1351004, 2013.

[128] Q. Wu, Y. Ye, H. Zhang, T. WS. Chow, and S. S. Ho, "Ml-tree: A tree-structure-based approach to multilabel learning," *IEEE transactions on neural networks and learning systems*, vol. 26, no. 3, pp. 430–443, 2015.

[129] R. Al-Otaibi, M. Kull, and P. Flach, "Lacova: a tree-based multi-label classifier using label covariance as splitting criterion," in *13th International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2014, pp. 74–79.

[130] R. Al-Otaibi, M. Kull, and P. Flach, "Declaratively capturing local label correlations with multi-label trees.," in *ECAI*, 2016, pp. 1467–1475.

[131] R. Al-Otaibi, M. Kull, and P. Flach, "clustering based on covariance. in proceedings of the ecmlpkdd 2015 doctoral consortium (pp. 43-52).(aalto university publication series). aalto university.," .

[132] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Machine learning*, vol. 63, no. 1, pp. 3–42, 2006.

[133] E. Baralis, S. Chiusano, and P. Garza, "A lazy approach to associative classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 2, pp. 156–171, 2008.

[134] F. Charte, A. J. Rivera, M. J. del Jesus, and F. Herrera, "Mlsmote: approaching imbalanced multilabel learning through synthetic instance generation," *Knowledge-Based Systems*, vol. 89, pp. 385–397, 2015.

[135] K. Douibi, N. Settouti, and M. A. Chikh, "The homogeneous ensemble methods for mlknn algorithm," in *Proceedings of the International Conference on Compute and Data Analysis*, New York, NY, USA, 2017, ICCDA '17, pp. 197–201, ACM.