T.C.
ALTINBAS UNIVERSITY
Institute of Graduate Studies
Electrical and Computer Engineering

CREATE AND ANALYZE NEW AUDIOLOGY DATA SET AND USING DATA MINING TECHNIQUES TO PREDICT HEARING AID FACTORS FOR AUDIOLOGY PATIENTS
(FIELD OF BIOINFORMATICS AND HEALTHCARE SYSTEM)

Maalim A. ALJABERY

Ph.D. DISSERTATION

Supervisor

Asst. Prof. Sefer KURNAZ

Istanbul, 2020

CREATE AND ANALYZE NEW AUDIOLOGY DATA SET AND USING DATA MINING TECHNIQUES TO PREDICT HEARING AID FACTORS FOR AUDIOLOGY PATIENTS

(FIELD OF BIOINFORMATICS AND HEALTHCARE SYSTEM)

by

Maalim A. ALJABERY

Electrical and Computer Engineering

Submitted to the Institute of Graduate Studies

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

ALTINBAŞ UNIVERSITY

2020

The thesis titled “CREATE AND ANALYZE NEW AUDIOLOGY DATA SET AND USING DATA MINING TECHNIQUES TO PREDICT HEARING AID FACTORS FOR AUDIOLOGY PATIENTS” prepared and presented by “Maalim A. ALJABERY” was accepted as a Doctor of Philosophy Thesis in Electrical and Computer Engineering.

____

Asst. Prof. Sefer KURNAZ

Supervisor

Thesis Defense Jury Members:

Asst. Prof. Dr. Sefer KURNAZ	School of Engineering and Natural Sciences, Altinbas University	___________________
Prof. Dr. Oguz BAYAT	School of Engineering and Natural Sciences, Altinbas University	___________________
Prof. Dr. Osman Nuri UCAN	School of Engineering and Natural Sciences, Altinbas University	___________________
Asst. Prof. Muhammad ILYAS	School of Engineering and Natural Sciences, Altinbas University	___________________
Asst. Prof. Oguz KARAN	School of Engineering and Natural Sciences, Altinbas University	___________________
Prof. Dr. Adem KARAHOCA	Faculty of Engineering and Architecture, University	___________________
Prof. Dr. Ahmet Mesut RAZBONYALI	Faculty of Engineering and Architecture, Maltepe University	___________________

I certify that this thesis satisfies all the requirements as a thesis for the degree of Doctor of Philosophy.

Approval Date of Institute of Graduate Studies:

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Maalim A. ALJABERY

____/____/____

DEDICATION

The most successful and worthy man who deserves all the great things

To my role model in whole my life

“To my father” …

“As we look back over time, we find ourselves wondering…

Did we remember to thank you enough for all you have done for us?

For all the times you were by our sides

To help and support us…

To celebrate our successes

To understand our problems

And accept our defeats?

Or for teaching us by your example,

The value of hard work, good judgment,

Courage and integrity?

We wonder if we ever thanked you

For the sacrifices you made.

To let us have the very best?

And for the simple things as well

Like laughter, smiles and times we shared?

If we have forgotten to show our gratitude enough for all the things you did,

We’re thanking you now.

And we are hoping you knew all along,

How much you meant to us.”

“"*Gone, but never forgotten*"”

ACKNOWLEDGEMENTS

The authors would like to thank the Department of Otolaryngology, Al-Basrah Hospital, especially Prof. Dr Husam Haider and Dr Hashim Hameed for their support and cooperation and for making the sample medical audiology records available to us. Besides, we wish to gratitude the staff of the Speech-Hearing Audiometry at Al-Nour health clinic for speech and hearing evaluation in Basrah, Iraq, while working together and making the large set of audiology records available to us.

ABSTRACT

CREATE AND ANALYZE NEW AUDIOLOGY DATASET AND USING DATA MINING TECHNIQUES TO PREDICT HEARING AID FACTORS FOR AUDIOLOGY PATIENTS

(FIELD OF BIOINFORMATICS AND HEALTHCARE SYSTEM)

ALJABERY, Maalim,

PhD, Electrical and Computer Engineering, Altınbaş University,

Supervisor: Sefer KURNAZ

Date: September/ 2020

Pages: 131

Clustering algorithm has been used in many studies to find patterns and relationships in audiology dataset. Employs clustering algorithm in grouping distinctive audiometric profiles of hearing-impaired patients. However, association analysis with Frequent Pattern Growth (FP-Growth) algorithm is more efficient in extracting related items in a dataset. Majority of these techniques are depending on components of conventional software that not mostly advanced for the field of audiology like SAS and SPSS. Despite the recent works upon audiology records and medical information in general, now a days actually no accessible of discovery appliance of hybrid knowledge enable to transact with aggregate of structured data, unstructured information, and audiograms as it found within our records of audiology data. In this research, we will present an approach of integrating supervised techniques and unsupervised clustering of audiology patient’s data with the identification of textual keywords associated with each cluster, in more precisely, those related to text comment, diagnosis and Hearing Aid (HA) type. This research deals with new specific data set which collects and analyze depends on audiology information and patient's diagnosis. It consists of 72 fields distributed on 71 fields for data details and one further for class. All fields are categorical with some of missing values. It was subjected to a very accurate analysis (before cleaning) based on the correct medical diagnosis and comprehensive information of the most important points that directly affect the selection of appropriate HA for Audiology Patient (AP), and via applying Data Mining (DM) techniques, we obtained a prediction of 100% for HA selection, and 98% to determine which power type of HA that those patients should use. To reach our goal, we examined DM techniques utilizing Python for coding and modeling.

Keywords: Clustering algorithm, audiology dataset, supervised techniques, unsupervised clustering, audiology patients, hearing aid, data mining.

TABLE OF CONTENTS

Pages

LIST OF TABLES .... **xi**

LIST OF FIGURES .... **xii**

LIST OF ABBREVIATIONS .... **xv**

LIST OF SYMBOLS .... **xvii**

1. INTRODUCTION **1**

1.1 DM AND HEALTHCARE SYSTEM 3

1.1.1 Treatment Effectivness. 3

1.1.2 Effective Management of Hospital Resorce. 3

1.1.3 Hospital Ranking. 3

1.1.4 Smarter Treatment Techniques. 3

1.1.5 Improved Patient Care 3

1.1.6 Health Policy Planning 4

1.1.7 DM Challenge in Healthcare. 4

1.1.8 For Active DM. 4

1.1.9 Advanced Analytics and Healthcare Services. 5

1.1.10 Healthcare Analytic and DM. 5

1.2 CHARACTERISTICS OF AUDIOLOGY DATA 6

1.2.1 Visual Examination and Patient’s Interview 6

1.2.2 Semi-automatic Collection of Audiology Clinic Data 6

1.3 DM TECHNIQUES 6

1.3.1 Data Preprocessing 8

1.3.2 Data Collection 8

1.3.3 Big Data Storage and Management 9

1.3.4 Training Set 9

1.3.5 Knowledge Discovery in Medical Dataset 10

1.3.6 Supervised Learning Techniques 11

1.3.7 Unsupervised Learning Techniques 25

1.4 LITERATURE SURVEY 29
2. METHODOLOGY 31
2.1 DATASET 31
2.1.1 Dataset Comparision. 34
2.1.2 Dataset Created. 35
2.1.3 Data Cleaning. 40
2.2 MODEL EVALUATION 51
2.2.1 Evaluation Methods 52
2.2.2 Evaluation Measures 55
2.2.3 Best Hearing Aid Style 55
2.3 MAIN MODEL 57
2.3.1 Orange Canvas Modeler 57
2.3.2 Workflows in Orange 60
2.3.3 Python Script 62
2.3.4 Supervised Model 64
2.3.5 Unsupervised Model 89
3. RESULTS AND COMPARISION 96
3.1 DATASET SAMPLES 96
3.1.1 Supervised Results and Comparision. 97
3.1.2 Unsupervised Results and Comparision. 107
4. OVERVIEW AND CONCLUSION 111
4.1 OVERVIEW 111
4.1.1 Dataset Record Types 112
4.1.2 Important of Healthcare. 112
4.2 CONCLUSION 113
5. FUTURE WORK 114
REFERENCES 116

LIST OF TABLES

Pages

Table 2.1: Audiology dataset fields definition 44

Table 2.1: The details of new Audiology dataset created 51

LIST OF FIGURES

Pages

Figure 1.1: The ERD of DM procedure cycle 5

Figure 1.2: The ERD of DM Learning Techniques 8

Figure 1.3: Stages of Knowledge Discovery Process 11

Figure 1.4: Maximum hyperplanes margin of SVM 14

Figure 1.5: The ERD of Naïve Bayes classification 17

Figure 1.6: The ERD of RF construction 21

Figure 1.7: Brief idea of the ANN functioning 23

Figure 1.8: Main purpose of ML and dataset reduction 27

Figure 2.1: The dataset distribution parts 32

Figure 2.2a: ERD of data cleaning process 41

Figure 2.2b: The Data-Cleaning Framework 42

Figure 2.3: The screenshot for last part of our dataset attributes 43

Figure 2.4: General flowchart for main model. Gray vectors refer to validation process and Black vectors refer to training and final test process 54

Figure 2.5: ERD illustrates the principal factors of choosing Hearing Aid 56

Figure 2.6: Screenshot of a simple workflow with two connected widgets 61

Figure 2.7: The Supervised Training Phase 65

Figure 2.8: The Supervised Testing Phase 66

Figure 2.9: The setting of SVM widget for our model 68

Figure 2.10: The ERD of our model with SVM 70

Figure 2.11: The setting of CN2 Rule Induction widget for our model 72

Figure 2.12: The workflow of decision-based analysis of proposed methodology 74

Figure 2.13: The setting of Naive Bayes widget for our model…………………………………78

Figure 2.14: The ERD of Naïve Bayes DTA…………………..………………………………...79

Figure 2.15: The setting of Logistic Regression widget for our model 80

Figure 2.16: The setting of Random Forest widget for our model 82

Figure 2.17: The setting of AdaBoost widget for our model 84

Figure 2.18: The setting of NN widget for our model 86

Figure 2.19: The setting of ML widget for our model 90

Figure 2.20: The setting of MDS widget for our model 93

Figure 3.1: The details of dataset sample used for test and train 96

Figure 3.2: The condition details of select rows for test and train dataset................................... 97

Figure 3.3: The main model design for all supervised techniques used within our model........... 98

Figure 3.4: The train data table choosing for all supervised techniques used within our model.. 99

Figure 3.5: The Prediction results table for all supervised techniques used within our model .. 100

Figure 3.6: The details of Test and Score table for all supervised techniques used within our model.. 101

Figure 3.7: The CN2 rule conditions applying in CN2 algorithm 103

Figure 3.8: The Distributions probability of CN2 rule inducer for (behind_the_ear) class 104

Figure 3.9: The Scatter Plot of CN2 rule inducer for the classes BTE Hearing Aid as X exes and ITE Hearing Aid as Y exes 105

Figure 3.10: The Scatter Plot of compare predictions between CN2 and NN for BTE Hearing Aid as X exes and ITE Hearing Aid as Y exes 106

Figure 3.11: The ML main model 107

Figure 3.12: The specifics of ML prediction results 108

Figure 3.13: The MDS main model 109

Figure 3.14: The details of MDS prediction results 110

Figure 4.1: The samples of both “BTE” and “ITE” HAs 111

Figure 5.1: The basic electrical circuit design for the next proposed of HA 114

LIST OF ABBREVIATIONS

FPG : Frequent Pattern Growth

HA : Hearing Aid

DM : Data Mining

AP : Audio Patient

ML : Machine Learning

NN : Neural Network

KD; Knowledge Discovery;

KDD : Knowledge Discovery Database

AI : Artificial Intelligence

DT; Decision Tree

DTA : Decision Tree Algorithm

DSS : Decision Support System

KM : Knowledge Management

EPR : Electronic Patient Record

DW : Data Warehouse

NHDW : National Health Data Warehouse

SL : Supervised Learning

HOS : Healthcare Organization System

SML : Supervised Machine Learning

SRM : Structural Risk Minimization

SVM : Support Vector Machine

SRM : Structural Risk Minimization

CN : Clark and Niblett

NB : Naïve Bayes

LR : Logistic Regression

OLS : Ordinary Least Squares

RF : Random Forest

ART : Adaptive Resonance Theory

ML : Manifold Learning

MDS : Multidimensional Scaling

ERD : Entity-Relationship Diagram

GUI : Graphical User Interface

API : Application Program Interface

LIBSVM: Library for Support Vector Machines

KM : Kaplan-Meier

LIST OF SYMBOLS

α : Constant determine the logistic curve shape

β : The standardized coefficients of logistic regression

ε : Epsilon-SVR parameter

1. INTRODUCTION

Medical DM has considerable potential for hidden patterns exploring in the medical domain's data sets. These hidden patterns can use with clinical diagnosis. However, the obtainable pure medical data are distributing widely, voluminous, and heterogeneous in quality [1]. These data require to be composed in a systematic form and then will integrated in an information system form of the hospital. DM technology supplies the technique of user oriented for hidden as well novel datasets patterns [2].

DM essentially the methods for discovering the executable and expressive patterns, descriptions, and orientations data via elicitation the data utilizing the techniques of recognition the patterns like Machine Learning (ML), algorithms of genetic, and Neural Networks (NN) [3].

The tools of DM capable to resolve complex problems; and research hidden patterns to find predictive information. The major phase in Knowledge Discovery Databases (KDDs) is DM, despite these terms (DM and KDD) are diverse. DM works for discovering of new patterns from the data within databases via update the algorithms in the purpose of elicit beneficial knowledge, while KDD points to comprehensive approach of discover from main databases a beneficial knowledge [4].

Medical diagnosis regarded as significant complicated task yet, that requires to executed efficiently and accurately. So, this system automation would be so useful. Regrettably, more of doctors do not have experience in all subspecialty, furthermore there is an insufficiency of resource persons in certain places. Thus, a medical diagnosis via an automatic system would probably be extremely beneficial by collecting all of them [5].

Convenient Decision Support Systems (DSS) and/or computer-based information can be help in achieving the clinical tests within a reduced cost. Accurate and efficient implementation of the automated system requires comparative studies of various available techniques [6].

DM has two essential goals; description and prediction. Description includes finding the patterns of human orientations as well comprehensible within data like learning of correlation rules, summarization, clustering, and other. Prediction includes utilizing some variables within data set for predict unbeknown values for another related variables such as regression, aberration detection, and classification [7]. Applications of DM in healthcare DM for Knowledge Management (NM) supplies most advantages to the healthcare organizations. It can be employed for diagnosis and detection with elevated performance for a lot of diseases like diabetes, cancer, heart diseases, kidney, etc. [8].

There are several considerations that may take to refine further data as well as continue the processes, and it will certainly add to the clinic dependability when the specialist persons taking the right choice of diagnosing the HA type for their APs. Therefore, this research goal is to find out and examine most crucial factors which influence on choosing the types of HAs. This modeling will make the audiologists to easily gained a second opinion from a source based on an accurate study that includes beside the classification, the clarification of the attributes that weighed as mightily difficulty in predictions. Thus, the audiologists make an educated decision about whether go on depend on primary diagnoses or must reconsidered.

The research purpose is to build a network through specialist doctors, developers, and the agents from HAs industrialists and tools of clinical diagnostic to recognize how to use the data of APs to construct an integrated visualize of aid user. This visualizes must consider the type of hearing impairment, the susceptibility of HA user, the conductance of HA user, and the conditions of which type of HA should be user chosen when utilize this device via considering some appropriate measures like:

1) Area and pattern of the hearing loss.

2) The function cognitive for HA user.

3) The HA user experience with wearing new HA device.

Applying DM instruments from computer science domain, the aim is construct new inferences via individual or combinations datasets that will produce the following:

a) Individualized fittings.

b) Real-world enhanced characterization instead of the laboratory HAs execution.

1.1 DM AND HEALTHCARE SYSTEM

1.1.1 Treatment Effectiveness

DM applications applied to develop the evaluate of medical treatments effectiveness. DM can produce an analysis of what action path that demonstrate effective via comparing and measure symptoms, causes, and treatments courses [9].

1.1.2 Effective Management of Hospital Resource

It is possible via using DM techniques to detect the priorities of complications of patient disease as well as get better treatments within correct and on time manner. Healthcare organization present an online gateway of the patients medical data, offer well treatment details by communicating with their doctors, and fill the form of prescription online [10].

1.1.3 Hospital Ranking

Different DM techniques applied to analyze diverse hospital details to decide their ranks. Hospitals ranking is determined via analyzing how hospitals transact with the low and high priorities patients risks [11].

1.1.4 Smarter Treatment Techniques

Patients and doctors can simply compare different treatment techniques. They capable to analyze effectiveness of treatments available and detect which treatment is better as well cost effective. The particular treatment hazard, smart methodologies development, and side effects are also achievable by DM techniques [9].

1.1.5 Improved Patient Care

To analyze the massive data of hospitals, the predictive model constructed using DM techniques which discover beneficial information from it then extract decision to enhance healthcare quality. DM assist healthcare providers to discover the patient’s requirements of present and future as well as their preferences for enhancing their levels of satisfaction [3].

1.1.6 Health Policy Planning

DM play a significant role for building an effective politics of healthcare to improve a quality of health in general and reducing the health services cost [12].

1.1.7 DM Challenges in Healthcare

NM using DM faces more challenges like need for algorithms with extremely high accuracy.

Below listed most noteworthy features of Healthcare.

- Recognize the anomalies rapidly by access to all records of patients data;
- Analyzing data by utilize the automated systems, since it extremely useful within main cases as well frequented aberrations;
- Increase the productivity as well nursing quality via smallest consultations, more repeated, and remote;
- Easily as well quickly interacting within structured procedure through shared various tools among the suppliers of elementary care and responsible nurses on daily patient's surveillance;
- Present assistance for any patient needs for motivational supports.
- Provide an effective contribution to the research of biomedical via tools of healthcare databases [8].

1.1.8 For Active DM

An automated DM technique is prerequisite on each updating of data and for implanting discovered knowledge with the system of healthcare information, so every time several DM techniques are utilized and compared with most appropriate [13].

Figure 1.1 shows the ERD of DM procedure cycle:

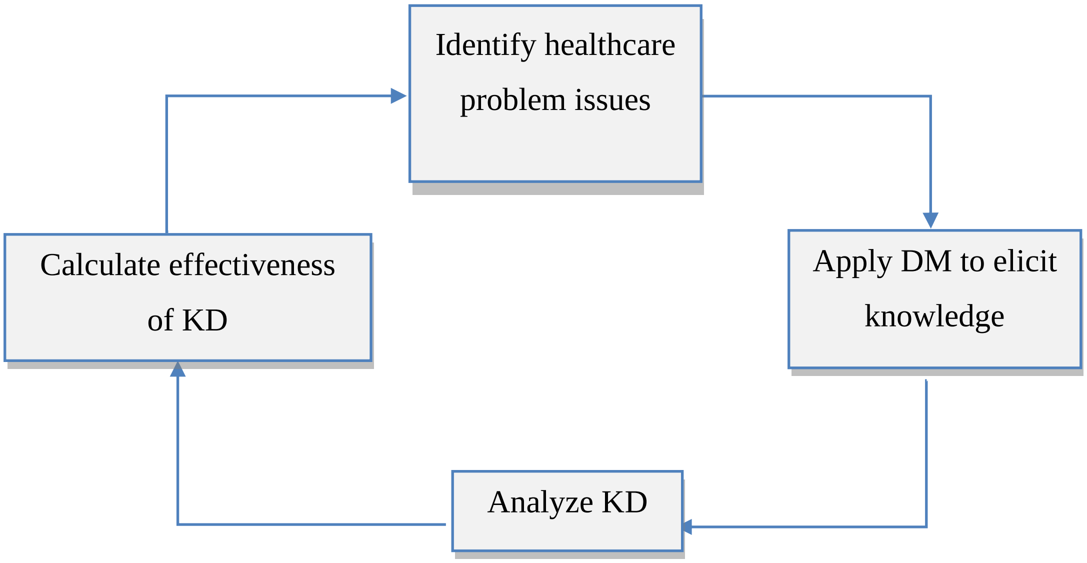


Figure 1.1: The ERD of DM procedure cycle

1.1.9 Advanced Analytics and Healthcare Services

The principal factor for gaining increased performance with supplying services of most industries is the capability for better understand the demand on these services. Using this information, the ability of decision makers may more specifically perform and best favorable resources that are very necessary to accept several quantities of requests [14]. Methods of predictive analytic can upgrade the accuracy of patient needs estimating for organizations which enhance the services of healthcare. Through easily analyzing the data of patient services in the available database, it was recognized that both of seasonal factors and general macroeconomic route had impressive effects on the needs of healthcare [12].

1.1.10 Healthcare Analytics and DM

DM defined as a process which gathered, analyzed, and stored the data in the purpose of produce high-quality knowledge and useful information. This term as well includes the technique of how this data is collected, filtered, and prepared for use, and finally DM includes the data processing to support predictive modelling and data analytics [15].

1.2 CHARACTERISTICS OF AUDIOLOGY DATA

1.2.1 Visual Examination and Patient's Interview

The visual checkup and the interview usually structured as a list of questions asked to the patient by the doctor who treads a well-clarify scheme. The answers of the patient are typically reported as a set of discrete values, usually of Boolean type [16]. In fact, clinicians are concerned to classify patient's data to unaffected (false values) or affected (true values) set. In addition to patient information like (name, surname, patient's identifier into the hospital, …., etc.), the rest data are Boolean and combine these Boolean data gives the comprehensive clinical picture for one audiology patient [17].

Data concerning to vestibular pathologies which characteristic in specialization of audiology are stored too, this data is mightily heterogeneous; because it includes time-based data like time to recurrence, categorical data like the category of patients with tinnitus, and lists like a list of diseases which not previously specified [18].

1.2.2 Semi-automatic Collection of Audiology Clinical Data

The specialized kind of data processed in the audiology unit makes hard to use standard tools of Electronic Patient Record (EPR) software for data storage as well data analysis. So, enhance the utilize of computerized tools is required to encourage the medical workers to utilize them without further workload [19].

1.3 DM TECHNIQUES

DM techniques can be classified as a predictive modeling based on namely the visualization with description, the clustering with association, and finally the estimation with classification.

Visualization with description can significantly contribute to comprehension the datasets, mainly the big sets of data, as well discovering the hidden patterns within data, particularly the complicated datasets which holding nonlinear and complex interactivities, which can performed generally before the modeling then tried to perform the data comprehension within DM methodology [20].

The main goal of clustering is grouping the objects or elements, like patients, in such a way which similar elements are belonging to the same cluster and dissimilar elements are within various clusters. Predictive modeling is probably the most significant as well commonly of DM applications [21]. The goal of association is detect the variables that work together like the market-basket analysis (the very popular form for analysis the association) which refers the produces probabilistic expressing technique like, “IF patient experience treatment X, then there is an expectation of 0.45 indicates this patient will appears symptom Y”. This type of Information may helpful for examine the associative relationships within healthcare system [22].

Estimation refers for the target metric variable prediction within nature (such as ratio or interval), like predicting resources amount used or the stay length. About the predictive modeling, DM algorithms ordinarily applied within conventional statistics, like logistic regression and multiple discriminant analysis. like logistic regression and multiple discriminant analysis. Furthermore, they contain nontraditional techniques which developed within areas of artificial intelligence (AI) and ML. Most significant techniques of these are Decision Trees and NNs [23]. The classification technique is predicting an objective variable that is categorical in nature, like predicting the fraudulence vs. integrity within healthcare system [24].

Can classified the DM techniques in six essential functions:

1. Classification: Detecting models that classify and analyze the data entries to various predefined classes;
2. Clustering: Determine the limited series of clusters or categories for data characterization;
3. Regression: Determining every entry of data to predict the variable of real-value [25];
4. Anomaly Detection (deviation detection): Discovers a most significant for the data subset;
5. Association Learning (dependency modeling): Discovering new pattern that describe the important dependency relationships among variables;
6. Summarization: Finding the compact attributes for the data subset [3].

Figure 1.2 shows the ERD of these DM learning techniques.

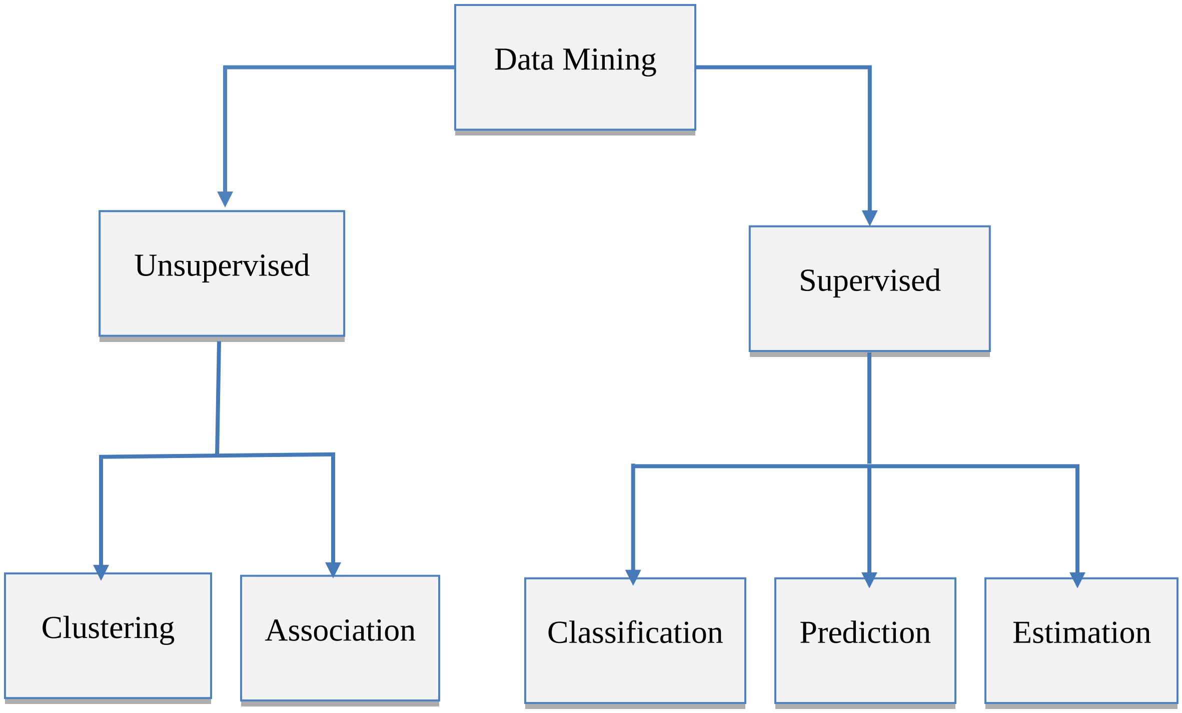


Figure 1.2: The ERD of DM Learning Techniques

1.3.1 Data Preprocessing

Data preprocessing classify as one of the main tasks for developing the Data Warehouse (DW) of heterogeneous sources, since it holds missing values, data cleaning, normalization, transformation, and more.

For National Health Data Warehouse (NHDW), data extracting from diverse sources like hospitals and diagnostic centers. On the other hand, diverse preprocessing procedures has been implemented on data, thus, NHDW can utilize within several methods to enhance the criterions of national health and for providing patients a prompt as well as better service besides encourage health concerned research through clinical investigators, specialist doctors, and scientific academics [26].

1.3.2 Data Collection

The first stage of DM is the conduct of gathering data. However, even before collecting data, plans and ideas should be supposed to decide what data should gained to gather specific data as needed and employ it efficiently. Moreover, several projects unsuccessful and surpass estimated costs since the low gathered data quality that may extract from bad data cleaning [13].

1.3.3 Big Data Storage and Management

One of the more principal elements in managing and dealing with data is to know how and where this data will save when gathered. The conventional methods of retrieving and storing such data are not effective anymore, because it is loading and extracting it from diverse outside sources, thereafter, it will be stored and structured in DWs as well as relational databases. Nevertheless, this data is classified and transformed before prepared to function and use [9].

Now a days, there are diverse data sources as well as most quantities of data has become obtainable. Thus, this data increasing will certainly require a powerful database which deal logically with the data and through data synchronization to adapt the rapid of data evolution [27].

1.3.4 Training Set

During the learning processes, the majority of learning algorithms employ all instances of an offered training set to evaluate the model parameters, but within training dataset, lot of instances often inutile and cannot be able to enhance the pattern predictive effectiveness or even affect deteriorate this effectiveness.

Some of considerable causes for ignoring such inutile instances are listed below:

- Noise reduction: Since learning algorithms mostly sensitized for noise, so should apply learning phase after these algorithms;
- Accelerate the response: Via reducing the computes since it significant to learners based on instances. These types of such learners are often named memory-based, case-based, or lazy learners [28].

Training datasets reduction may needful for large datasets. The structure as well training set size are required for accurately estimating the parameters of the model and discover the method of instance selection [29]. Training dataset is mainly specified used for providing classified data of recognized information which is utilized within Supervised Learning (SL) algorithms for constructing regression as well as classification patterns [30]. Every training example can consider as an attribute vector beside the convenient value of output (the identifier either class or label). SL algorithm extract the function of regression or classification from provided training dataset. Regression or classification concludes function would rather predict a proper value of output for every input vector. Training phase aim is to estimate model parameters to predict values of output with an efficient performance of predictive in a real implement of the model [31]. Within best case, the training dataset is an inhabitance representative set. Any approached techniques are unnecessary when having a population subset example, while we not ever have it in the practice. We commonly can gain random samples of these subsets as well as to make it as representative as possible, we should use different methods [32].

To determine a representative set, should determine a minimal symmetric subset of training set. Assuming denote to the training set as T, so when obtained a training set T, then should determine a subset $T* \subset T$ such that T∗ is a smallest instances set such that Acc(T∗) ∼= Acc(T), where Acc(C) refers to the accuracy of classification obtained by using C as training dataset. The subset applied to evaluate the model is validation dataset V, which utilized for choosing the model, besides testing dataset S that applied for model estimation.

1.3.5 Knowledge Discovery in Medical Databases

DM is a Knowledge Discovery (KD) essential step. The latest years has received a large saucepan of attention within Information manufacture. KD includes a reiterated series of data combination, knowledge presentation, DM pattern recognition, data cleaning, and data selecting as shown in Figure 1.3.

DM may perform association, time series analysis, class description, prediction, classification, and clustering. DM is discovery driven comparing with conventional data analysis [33]. DM is an interdisciplinary field which directly connected to inductive logic programming, Data Warehousing, Neural Networks, Machine Learning, and statistics. DM presents the pattern recognition as well tries to discover the patterns within datasets which hard to discover in conventional statistical techniques [34]. In the absence of DM, is so hard to perceive the data collected in complete possibility of Healthcare Organization System (HOS) since the information before analytics phases usually enormous, uncertain, high-dimensions, and distributing [35].

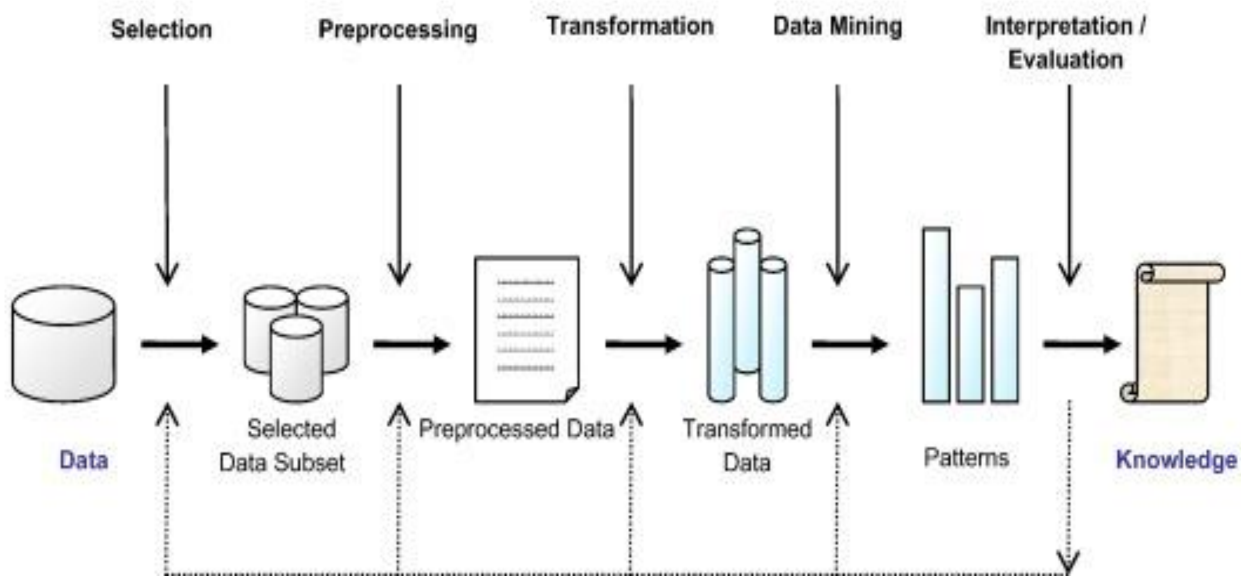


Figure 1.3: Stages of Knowledge Discovery Process [3]

In this research we used a comparison between some of the supervised and unsupervised techniques, so we can infer the best among them.

1.3.6 Supervised Learning Techniques

SL is a formalization of a learning idea from examples. The learner in SL (usually, the computer program) is provided together with two data sets; training and testing [36].

The idea is for make learners to “learn" from the set of classified examples within training set, so can recognize unclassified examples in test dataset with greatest obtainable precision. Thence, the learner’s aim is to improve the procedure, the program, or the rule that categorize other examples (within test phase) via analyze the given examples which already have class label [37].

Supervised Machine Learning (SML) works for searching algorithms that conclude from instances supplied externally to build general hypotheses, thereafter it makes predictions around future instances. Supervised classification defines as one of the most frequently executed tasks via intelligent systems [38].

SL is a ML mission of deducing functions via supervised training datasets. The training dataset usually contains training examples dataset. Within SL, every instance contains two of; input item (vector) as well as required output (indicative signal) [39].

The SL technique analyses training dataset then generates an estimated function, this function is regression function (when continuous outputs) or classifier function (when discrete outputs). This estimated function ought to predict an accurately output for every valid entry. Such that operation necessitates learning algorithm for popularize extracted from training dataset to the hidden positions within "logical" [40].

The main aim of SL technique is gain benefit from *n* samples of training to create a classifier capable to distinguishing between classes of m basis on length *d* of an input vector. In the linear classifier context, the *d* components of an input vector $x = [x1, ..., xd]\ T \epsilon\ Rd$ are totally the *d* detected features. In general context; *d* components of vector *x* are *d* values (probably nonlinear) of basic functions which computed by features observed [41].

There are some supervised techniques we applied for comparison the predictions for each of them, as mentioned below:

1- Support Vector Machine (SVM):

Support Vector Machine (SVM) is strong “State Of The Art” technique as well powerful theory basses, founded upon theory of Vapnik-Chervonenkis. Besids, SVM has powerful structured characteristics. Powerful structured indicates to the model popularization of novel data. Obviously, SVM designed as an instrument to resolve the problems of SL classification [24]. SVM is a powerful technique of training for SL. The first introduced of SVM was by Vapnik in 1992 (Vapnik, Boser, & Guyon 1992). It was applied for several applications for regression, classification, and feature selection [42].

In classification, SVM determines the best separating hyperplane utilizing the margin concept which is the SVM essence. The margin refers to the distance between hyperplane and the points which are closest to it on both sides, this distance should expand as maximum as possible for better generalization. Furthermore, there is an important tradeoff between minimizing the misclassified examples number and maximizing margin [43]. SVM founded on the principle of Structural Risk Minimization (SRM). It determines the input items vectors to the highly dimensions area where a maximal detaching hyperplane created. There are pair of parallel hyperplanes created in every hyperplane sides to detach the data. Detaching hyperplane works for maximize an area between the parallel hyperplanes [44].

SVMs are SL techniques concerned on using regression as well as classification which be held by the group of popularize linear classification. It reduces the error of experimental classification as well as raise geometric margin simultaneously. Thus, it denominated as "*Maximum Margin Classifiers*" [23]. An assumption generated that the larger distance or margin amid two parallel hyperplanes, so that bestead popularization error classifier shall be. The theorize of data items in the form:

$$\{(a_1, b_1), (a_2, b_2), (a_3, b_3), \ldots., (a_n, b_n)\}$$

Where: b_n = (1 , -1); constant value indicates class of a_n value belongs; n = total samples, and a is p-dimensional real vector [45].

The scaling is significant to protect against the larger variance attributes or variables. To show this training dataset via means of separating or dividing hyperplane, can be rely on the following formula:

$$w.x + b = 0 \tag{1.1}$$

Where, w is the vector of p dimensional as well as b is the scalar and w vector indicate to a vertical splitting hyperplane. Via inserting b as offset parameter, can maximize the margin, however when missing the b scalar parameter then hyperplane compelled to pass via origin which leads to limit the final solution [14].

Can describe the parallel hyperplanes via equations below:

$$w.x + b = 1 \tag{1.2}$$

$$w.x + b = -1 \tag{1.3}$$

We can choose these hyperplanes if linearly separable training data, so that it will be no elements amid them, thereafter, allows for attempting to expand the area amid them as maximum as possible. Via utilizing geometry, can discover the margin amid hyperplanes equal to 2 / $|w|$. Thus, there is a need to reduce the $|w|$ [46].

With activation of data points, should guarantee such activation for “i” either, so if “i” formed as: $w. x_i – b \geq 1$ or $w. x_i – b \leq -1$, or take the form of: $y_i (w. x_i – b) \geq 1, 1 \leq i \leq n$ [47].

Figure 1.4 shows the largest hyperplanes margin of SVM which trained from samples of pair of classes.

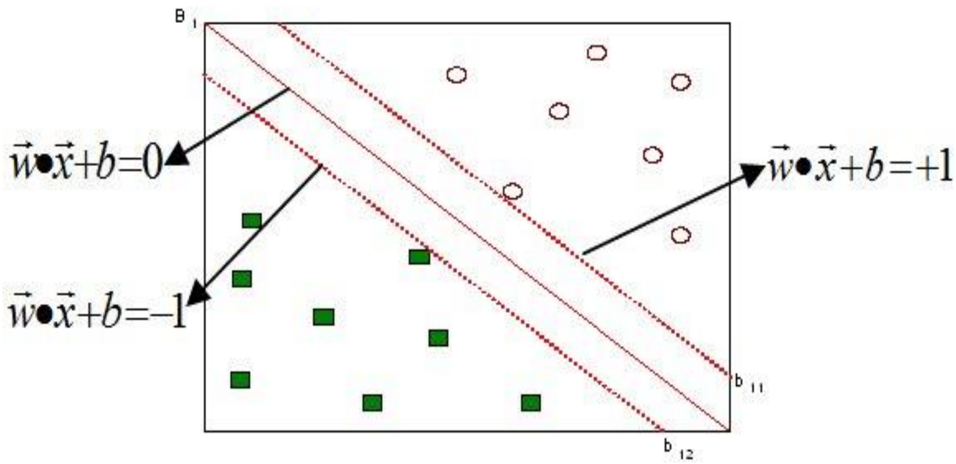


Figure 1.4: Maximum hyperplanes margin of SVM [47]

2- CN2 Rule Induction:

CN2 is a well-known and traditional DM method that is widely applied for rules detection from the databases as well searches for list of the rules in an increase fashion [48].

CN2 is rule induction learning technique. It intended to labor even when imperfect training dataset. Furthermore, CN2 algorithm depends on ideas deduced from algorithms like AQ and ID3 [49]. Generally, CN2 generates a set of rules as same as that created within AQ but it can treat noisy data such as ID3. This algorithm should be gives an examples and training datasets which formerly classified to create the categorization rules, the list of constraints, and the unpretentious constraints that may use alone or within group with examples dataset which defined previously to utilize for classification [50].

The induced rules include the form of "*if Condition then Class*", where Condition refers to the feature's conjunction (attributes pairs and their values) and Class refers to the value of class [51]. The first segment of the form (IF part) is renowned as earlier rule and it includes the conditions set, every condition usually mentions to a term. The second portion of the form (THEN part) is consequential rule in addition to determines the class predicted for cases that their attributes of predictor satisfy all terms which specified in the previous rule [48].

CN2 rule algorithm generates predicates, it does not generate rules, and the main example's class covered by a predicate will assign as a rule class [52]. The instances rule induction has found itself as the basic part for most systems of ML. Thus, the continuing expansion of inductive methods is beneficial for researches. This algorithm is described by [Clark and Niblett, 1987, 1989] [49].

3- Naive Bayes (NB):

Naïve Bayesian (NB) classifier consider as one of the most common DM techniques for classifying big dataset. NB has been successfully used for several classification task of problem domains like intrusion detection, weather forecasting, medical diagnosis, pattern and image recognition, bioinformatics, and loan approval etc. [53]. NB classifier as well applied efficiently in web classification and feature selecting. A classification task is for map the attributes set of sample data onto class labels set, furthermore, NB classifier specifically suitable as universal approximates proven [54].

NB classifier is statistical classifier founded on the hypothesis of largest posteriori and Bayes' Theorem. Some of reasons which make this classifier so simple and common that it is fast to implement and easy since the naïve hypothesis of class conditional independence decrease the cost of its computational [55]. NB classifier is a Bayesian statics term, it depends on implementing independence powerful (Naive) hypothesizes with Bayes' theorem [56]. NB classifier supposes that missing (or existence) for any class articular attribute is irrelevant to a missing (or existence) for other attributes [57].

Experiential studies which comparing classification algorithms discovered that the NB classifier is usually comparable in performance and efficiency with selected neural network and decision tree classifiers. NB had been utilized in ML algorithms for Sentiment Classification and Opinion Mining [58].

NB classifier is the classification of DM algorithms to find functions or models that differentiate or explain classes of data. Reverend Thomas developed the Bayes theorem within mid-18th century, based on the equation below:

$$P(H|X) = [P(X|H) * P(H)]/P(X) \tag{1.4}$$

When:

H: data hypothesis (specific grade);

X: Class Data;

P : Probability;

P (H |X): Hypothesis H Probability based on X conditions (the posterior probability) [59].

Depending on precise nature for a probability of the pattern, NB classifier technique enables to efficiently trained within SL settings [60].

The generics NB Classifier Algorithm steps are:

1) Read the attributes and dataset class;
2) Calculate posterior probability for all attributes to the existing class;
3) Calculate the prior probability for existing classes [61].
4) Compute the subsequent probability multiplication value for all classes as well prior value to every existing classes.
5) Find the greater value of probability within step four as a final classification [62].

Figure 1.5 shows the ERD of NB classification steps:

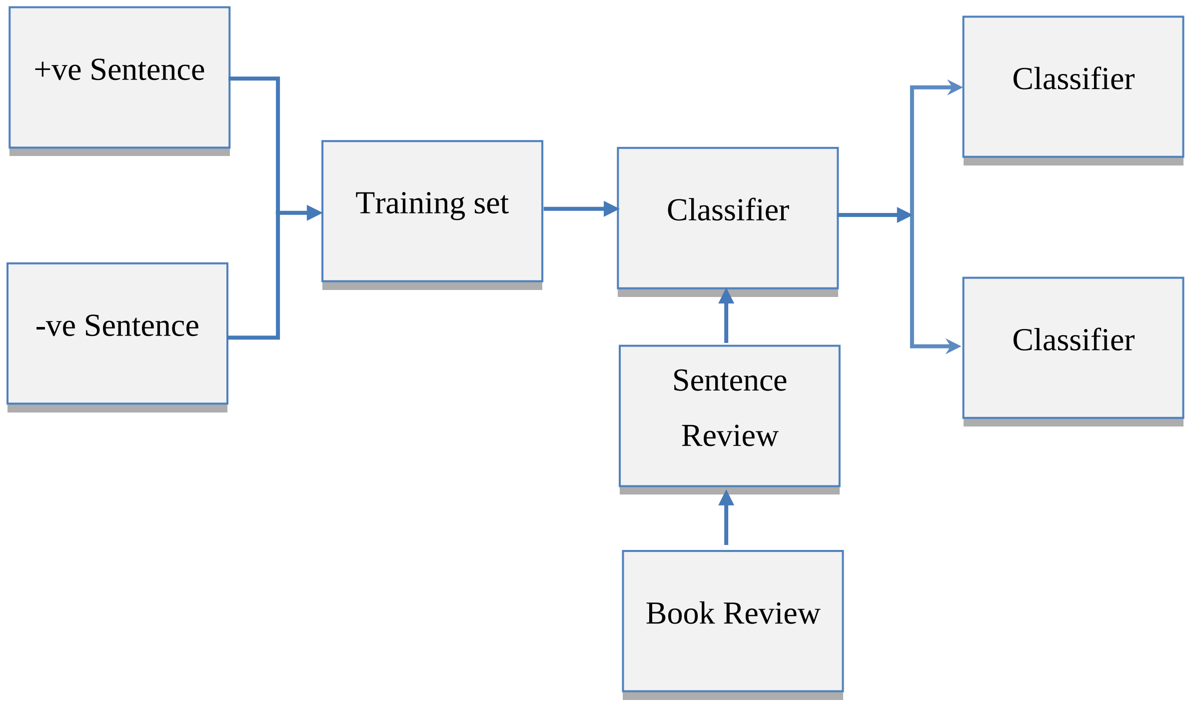


Figure 1.5: The ERD of Naïve Bayes classification

The feature of NB technique is it demands just little quantity of training dataset for estimating parameters (variables variances as well as means) which needful for the classification. Since autonomous variables will usually presumed, so just the contrasts variables of all classes require to specified [63].

4- Logistic Regression (LR):

LR is a prediction technique, such as Ordinary Least Squares (OLS). It's extremely significant for DM. LR enable to simply recognize the functions which helpful for give a demonstration of the correlation among various variables [64]. By the aid of training data, can easily build LR algorithm considering on two dissimilar types variables: (P, Q) which utilized in statics domain. One of these variables works as dependence variable and the other as independent variable. There is just one dependent variable number as maximum while can be exceeding one for independent [65].

Fox (1997) extracted that the regression applied to detect the functions which illustrate the correlation among various variables. The mathematical model structured utilizing training dataset [66].

LR is special case of conditional random fields and maximum entropy models, it is may expand the suggested approach for them. LR is widely utilized in a lot of studies because of its capability for achieving the probability prediction by its logistical formula [67].

For N samples set, N-1 fitted for LR model which defined by:

$$Log\ (p) = \alpha + \beta 1X1 + \beta 2X2 + \ldots + \beta kXk \tag{1.5}$$

when:

p: Malignancy probability;

α: Is a constant;

$\beta_1, \beta_2, , \beta_k$: The coefficients of LR,

$X_1, X_2, , X_k$: The LR parameters;

α, β: Define a logistic curve constitute [68].

LR is a common approach used within statistics to overcome its problems. Within a problem of two-classes, the logistic model which added has the formula of:

$$log \frac{p(y=1|x)}{p(y=-1|x)} = \sum_{m=1}^{M} fm(x) \tag{1.6}$$

Within left guarantees, the routine logit transformation for each value of $\boldsymbol{F(x) = \sum_{m=1}^{M} f_{m}(x)} \in \boldsymbol{R}$, The inverting estimates of probability lie in [0, 1] can be fetched:

$$p(x) = P(y = 1|x) = \frac{e^{F(x)}}{1 + e^{F(x)}} \tag{1.7}$$

This form is a general additive for F(x); an exist of exceptional cases which are well known within statistics. Linear LR and additive LR usually these models are fit by maximizing the binomial log-likelihood as well get benefit from all optimality features of associated asymptotic of maximum likelihood prediction [69].

Regression analysis constructs the relationship between outcome or dependent variable and predictors set. Regression, as a field of DM technique, is a SL which partitions the main database into validation and training data [70]. Can apply LR algorithm to pattern the relevance among autonomous and dependent variables. In DM the effective variables are needed for prediction while autonomous variables are the features which previously recognized [71].

The categories of LR techniques are:

- "Multivariate Linear Regression" (MLR);
- "Linear Regression" (LR);
- "Multivariate Nonlinear Regression" (MNR);
- "Nonlinear Regression" (NR) [70].

LR model usually used to model the experimental unit probability that belongs into specific class depending on the information measured onto experimental unit. Generally, LR model is usual statistical model using for classification and discriminant analysis. It is commonly applied in diverse fields including management, marketing, engineering, medical fields, and so on [66].

5- Random Forest (RF):

RF is an approach of general technique of the indiscriminate decision forests which aggregate learning dataset technique of regression as well as classification which control via constructing plenty decision trees during training dataset time then producing either a classification (classes style) or regression (average prediction) for every single tree [72]. indiscriminate tree usually is a supervised classifier works on aggregate learning dataset technique which creates most singular learners [73].

Adele Cutler and Leo Breiman (Breiman, 2001) introduced random trees. The algorithm can transact with both regression and classification problems. Every node within typical tree divide utilizing the better divide amongst all variables and among the predicators subset which chosen randomly at that node [74]. Bagging assists to obviate over-fitting concurrently and minimizes variance. Such that method concludes all cases from main training dataset randomly as well as use bootstrapping datasets to build every decision tree within RF. Every tree classifier will called component predictor [75].

Random trees are tree predictors ensemble which called Forest. The mechanisms of RF classification can be briefly illustrated as follows: A classifier of indiscriminate trees gets vector of input feature which classifies with all forest trees, then generates class name which received highest “votes”. With regression, the response of classifier is an average of all responses for each tree within forest [76]. Random trees are aggregation of two existing algorithms in ML: RF idea are merge with single model trees. Model tree define as decision tree as each individual leaf holds linear pattern enhanced for domestic subspace which expounded via such leaf [77].

RFs have been used for enhancing the performance of considerable individual decision trees. Variety trees usually generated via twain randomization methods. Firstly, like in Bagging, sampled the training dataset, and replaced for every individual tree. Second, when tree is growing, instead of every time calculate perfect potential split of every tree node, only considered the indiscriminate subset for all features within each node, then compute a better split of this subset [78]. These trees used for trees classification of random model for a first time gather the RFs and model trees. Indiscriminate trees utilize such produce to divide selection, so it induces the equiponderant trees reasonably as just single global setting works for ridge value over all leaves, and hence facilitate an optimization conducts [79].

The tree constructing process starts with dividing origin node to number of binary nodes via so easy interrogative formula like: $i \leq b$? Where i is any variable within dataset and b refers to a real number. At the beginning, each observation is in origin node. The executes of computer-intensive technique which investigates on better divide in each potential point for all variables. The technique that this algorithm utilizes for constructing trees is called Binary Recursive Partitioning [80].

Figure 1.6 shows the ERD of RF construction:

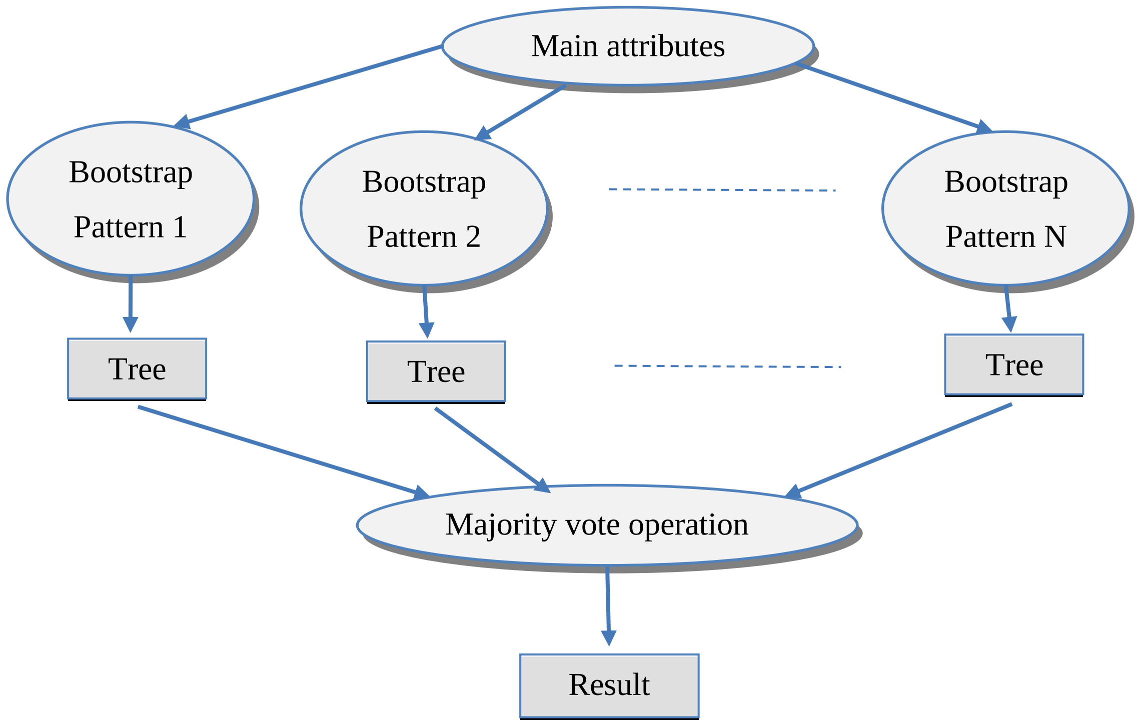


Figure 1.6: The ERD of RF construction [80]

6- AdaBoost:

Boosting is ML meta-algorithm for performing SL which combines most weak learners for creating strong learner. With "boosting" approach of ML, a powerful set of classifiers will be formed by refitting the weak classifier successively to different dataset weighted realizations [81].

The weak learner typically enables to classify the good rates samples somewhat higher than indiscriminate predicting sample's classes. Whereas, a powerful learner usually enables any dataset to be classified at higher rates, that denotes best attributions from that may done via weakly learners [82].

This axiomatic procedure has gotten a success tremendous amount. Shortly after its introducing, in the NIPS conference 1996, Leo Brieman ranked AdaBoost (Schapire and Freund, 1996) as a first boosting algorithm, the *"best off-the-shelf classifier in the world"* [83].

Boosting not restricted on algorithm constraints like dimension of each sample, training samples number, and training rounds number. After each learning round, the all samples weights (which are so important) are update depend on the weak learner classification error of that round. On the other hand, the samples that are wrongly classified will gain weight while the samples that are correctly classified will lose weight [84].

Early success of AdaBoost was at once followed via efforts to recast and explain it in more traditional statistical terms. The boosting statistical view holds that AdaBoost basically is stage-wise optimization for exponential loss function [85]. AdaBoost is binary boosting algorithm as well it may be the most important one stands for the fundamental milestone for most other algorithms of classification. This boosting algorithm works on divide the main dataset into two classes. Expanding AdaBoost directly to exceed two classes was avoided since it does not perform as perfect as binary companion [86]. Although, AdaBoost resistant for overfitting, the test of error increases after enlarging iterations number. For avoiding any overfitting, can validate each item of train data by data held-out from training set [87].

AdaBoost is classification algorithm that repeatedly calls the given weakly learner algorithm. Weakly learners can learn the algorithm which performs somewhat best than the indiscriminate predicting as well as finds separation boundary of two classes (negative and positive) [88]. AdaBoost combines the weakly learners' number for composing powerful learner in the purpose of achieving preferable class among whole classes. Strong learner is the vote of weighted majority of weak learners [89].

7- Neural Network (NN):

NN stands for a brain metaphor of information processing. These techniques are inspired biologically, not an exact replica about how actually the brain functions [90]. NNs have been proved to be so promising systems within many business classification applications and forecasting applications based on their nonparametric nature (such as no rigid assumptions), ability of generalize, and ability of "learn" from data [91].

NN is a machine designed for modeling the way of what the brain performs a function or task of the network interest is simulated within software of digital computer or is implemented usually by applying electronic components. A NN also referred as parallel distributed processors, connectionist networks, neuron computers etc. [92].

Neural computing points to the method of pattern recognition for ML. Often called a NN or an Artificial NN (ANN) to the final model produced from neural computing. NN computing is the key component for any of DM tool kit [93].

In general, the NN model can divided into three types as the following:

(a) Feed-forward network: It considers any function network and model of perception back-propagation as representatives, and basically used within areas like pattern recognition and prediction [90];
(b) Feedback network: It considers the models of continuous and Hopfield discrete as representatives, besides essentially utilized for optimizing computing as well as association memory;
(c) Self-Organization Network: It considers the models of Kohonen and Adaptive Resonance Theory (ART) as representatives, as well essentially used with cluster analysis [94].

Figure 1.7 illustrates an ANN and brief idea about how it works.

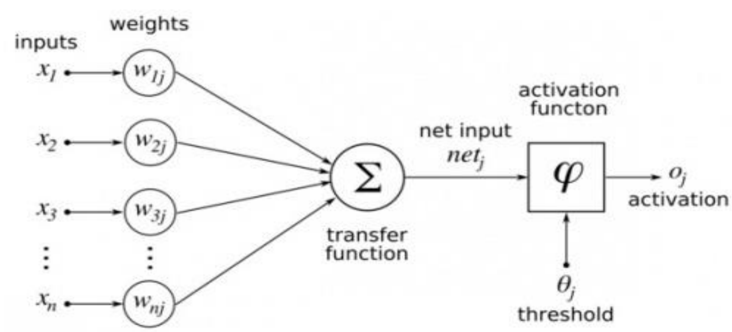


Figure 1.7: Brief idea of the ANN functioning [95]

There most important advantages of ANN are:

- Noise tolerance;
- High Accuracy;
- Parallel hardware Implementation;
- Independence for prior assumption;
- Maintenance ease [95].

The basic requirements and components of the ANNs are:

- Transfer Function;
- Artificial Neurons;
- Weight [96].

These three requirements or components are the ANN bases. Now we refer to the functionality of ANNs within brief steps:

- Input will give to an Artificial Neurons;
- Supervising an Input data, will creates indiscriminate weights on the connects;
- Within a node, the sum operation will executed on created weights multiplication as well as presented inputs;
- The biases upon some nodes as well impact on generated results from these nodes because of the surrounding environment [97].
- Thereafter, can apply the transfer functions for the nodes generated outputs and then supplies the outputs.
- The outputs then will be an inputs for more neurons linked this network.

The important factor is term "Weight", because the entire output is usually depending on this factor [98]. ANNs are perfect at observation because it generates the synaptic weight; so can expect high performance and accuracy [99].

The functioning of NNs can be divided into three phases, which we cite in the following:

1. ***Architecture:*** It refers to the ability of neurons to regulate their work and exchange interaction between them.
2. ***Learning Algorithm:*** The technique of trains ANNs for conduct a specified mission.
3. ***Activation Functions:*** The transfer functions, also can be defined as activation functions which conducted on outputs produced via the nodes [100].

Although the rule evaluation aim based on every particular implementation, however, can assess the rules according to the following basic goals:

- Find an Integrated concatenation for producing new rules, so that can gains more accurate results in the dataset given;
- Exam the extracted rules precision [101];
- Detect how amount of non-deduced knowledge within NN;
- Discover discrepancy among trained NN and deduced rules [102].

1.3.7 Unsupervised Learning Techniques

Unsupervised learning can assist to detect the patterns which unbeknown previously within dataset without any pre-existent tags, as well it called as a self-organizing. Furthermore, it lets the eventuality intensity modeling for the presented inputs [103]. Self-Organizing NNs learn by applying unsupervised learning techniques for recognize invisible samples within unlabeled inputs datasets. Such unsupervised indicates the capability for organizing and learning data without supplying any error signals to estimate the prospective outputs. [104].

Unsupervised learning is one of the three essential machine learning categories, besides reinforcement and supervised learning. The techniques of "Semi-supervised learning" basically mixed the unsupervised as well as supervised algorithms besides it has been described too [105].

Two of major methods applied with unsupervised learning are cluster analysis and principal component. Cluster analysis basically used within unsupervised learning to segment or group the datasets that shared attributes to extrapolate the relationships of algorithmic [40]. Cluster analysis defined as a field of ML which collects the data that normally has not classified, categorized, or labelled. Instead of feedback responding, cluster analysis usually determines commonalities within datasets then interacts depending on either absence or presence of these commonalities within every novel bit of such data. Such that technique helps for detecting the irregular data points which not appropriate with such sets [106].

A main usage of unsupervised learning being within domain of intensity prediction upon statics, further unsupervised learning includes numerous areas encompassing explaining as well as summarizing data attributes [107]. Can be compare such technique with SL since SL intends to deduce the distribution of conditioned probability *px(x|y)* upon *y* within inputs data. The aim of unsupervised learning is deduce the distribution of priori probability *px(x)* [105].

There are two unsupervised techniques we applied for comparison the predictions for each of them, as mentioned below:

1- Manifold Learning (ML):

In mathematics, the manifold is known as the points' set which are locally conduct such as Euclidean spaces. That means there is an attribution possibility to peculiarities for these points. thereafter, the manifold based on the local conduct is like that determined by Euclidean spaces [108].

ML technically faces about three challenges as mentioned below:

1) Training data should be populated densely in the actual space where the manifold resides; whether data are populated sparsely, or many data samples within local area cannot find the neighboring points, then no manifold will be learned;
2) The noise presence in a local area can prohibit the real structure from correctly learning;
3) When the data dimension is high (usually more than 30), the curse of dimensionality will aggravate the two above problems [109].

Large instances number required to describe the manifold in a high-dimensional space. The noise problems increase when data sparsely populated, which is unavoidable within high-dimensional space [110].

In more detail, if locally manifold such as R^m, so it dimensions defined *m*. Thus, manifold *m* dimensional, needs for its description a locally coordinate *m*. The most prevalent way for manifold describing is displaying a points' set in the space of aR^n Euclidean. This operation is called “*Embedding Manifold in R– space*” [111].

Figure 1.8 illustrate the main purpose of ML and dataset reduction.

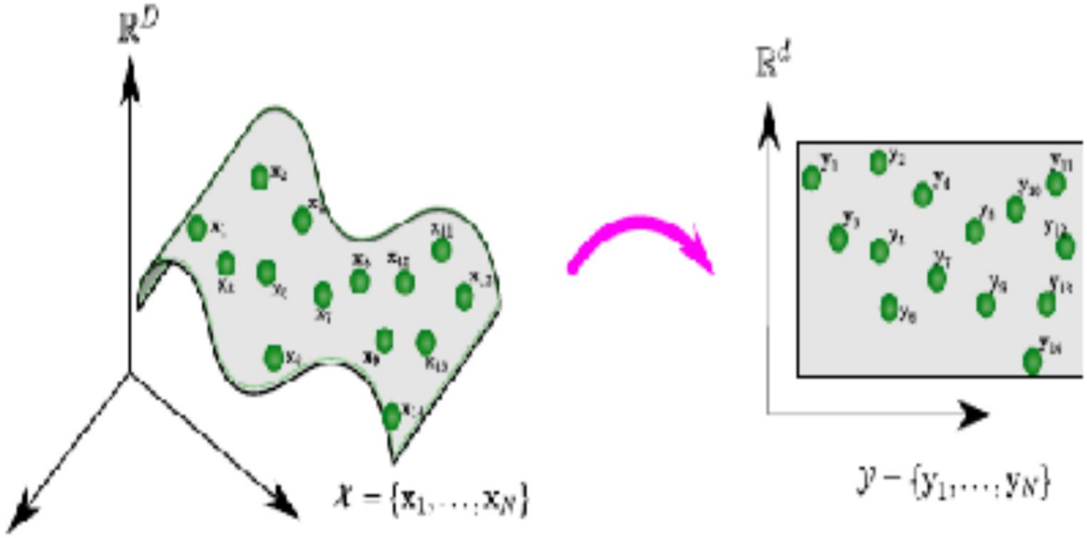


Figure 1.8: Main purpose of ML and dataset reduction [111]

ML methods mapping the dataset from high-dimensional to low-dimensional, thus it preserved their intrinsic geometry. These algorithms depend on a hypothesis that database desired is near or within the high-dimensional manifold structure which mapped in a low dimensional space via ML algorithms. These procedures used to identify and extract this manifold [112].

ML algorithms are consisting of local and global categories. Within local methods, the database usually mapped into low dimensional so that it would preserve the local properties. But In global methods, the database is mapped from high-dimensional to the low-dimensional thus it would preserve global properties of database [113].

2- Multidimensional Scaling (MDS):

MDS algorithm is a tool to evaluate similarity judgments. Practically, MDS refers to the statistical procedures set used for dimension reduction and analysis of exploratory data. It requires as input, similarity estimates within items group; these might be different ‘indirect’ measurements like perceptual confusions, or overt ratings, as well the stimuli might be conceptual or perceptual in nature [114].

In general view, MDS is a methods' set for detecting "hidden" structures within multidimensional data. Depend on the proximity matrix which extracted from variables evaluated on samples as input entity, all these spaces are mapped upon a spatial representation with lower dimensional (usually two or sometimes three dimensions) [115].

MDS aims to reconstructing the dissimilarities between objects pairs by distances within low-dimensional space. Although, this dissimilarity itself may unknown in some cases, but the dissimilarity range is given [116].

Like a fuzzy data, allows rise to the data matrix which every dissimilarity is a period of values. These period dissimilarities modelled through the distances ranges that defined as the maximum and minimum distance denotes the entities between two specific rectangles. For such data, previously have been suggested two approaches and then one of them already investigated [117].

MDS models are aim for representing the pairwise of similarity and dissimilarity data which stored as Euclidean distances in the similarity matrix within low-dimensional space to make these types of data are accessible for visual pattern and more explorations. This mapping is achieved from the similarities of a_{ij} to MDS configuration V via transformation function $\boldsymbol{f} : \mathbf{a_{ij}} \rightarrow \boldsymbol{V}$, where f choice is specifies the model of MDS [118]. MDS essentially is the visualization technique of proximity data, which means, data within dissimilarity matrices as N * N. It builds map ("embedding", "configuration") via interpreting all differences as spaces [119].

The dissimilarities recurrent sources are graphs and high-dimensions data. If the dissimilarities are spaces among high-dimensions targets, then MDS works as dimensions-reducing method which often nonlinear. If the dissimilarities are shortest-paths spaces within graphs, then MDS works as technique of graphs styling [120].

For conducting MDS algorithm, the 'proximity matrix' required; which means, the similarity estimates collect between every items pair in the set of stimuluses. So, for a set consisted of k items, the $(k * (k-1))/2$ proximities should achieved, so that every item will compare with others once at least. That's means the comparisons number grows rapidly like a stimulus set size function [119].

For instances, with 10 items set, 45 comparisons should made. And for collections of 20 items, 40 items, and 80 items, the grow of comparisons reaches respectively to 190, 780, and 3160, and so on [114].

With an enormous size of dataset, it may impractical to gather a completed matrix of each person, so this data can concatenate via people to compose an aggregate single matrix. For preference, a total matrix will collect from many participants, thus can be rated every pair of items several times (for protecting against noise of measurement) [121].

1.4 LITERATURE SURVEY

This study endeavors to use the effective of DM techniques.

Recently, DM techniques applied widely for most ML problems like pattern recognition, bioinformatics, estimation problems of nonlinear regression, and the analysis of linear regression. It is used in a lot of areas as well as proven effective particularly when resolving those problems [17]. DM has presented methods for both predictive (data induction for forecasting and learning) and descriptive (the properties characterization of the data) tasks. The classifier selection has a crucial effect on the efficiency and accuracy of monitoring [122].

In the medicine, a large amount of newly data generated constantly, furthermore, the sets of data created are higher than the sets of knowledge created. So, the large increases within data production require a rapidly convey to knowledge [123]. To achieve that, clustering tests data items groups which are dissimilar and similar. It differentiates the high samples density areas (data clusters) then displays the centers of all these clusters.

Clustering techniques include Artificial Neural Network (ANN) and statistical approaches. Within unsupervised clustering, data which unlabeled aggregated without any of supervised knowledge [18].

Medical information includes databases which stores the data of social insurance, such as patient's records. With the Information Technology advancement, such therapeutic data groups put away within electronic structures. In addition, these databases include information extensive volume [34].

"*Big Data*" is a paradigm introduced in a computer science field to abstract data size. It refers to the big data quantities which need precise processes depending on the requirements of business. To gain knowledge and obtained exact information, the professionals should apply methodologies, technologies, and intelligent tools in the major areas' fields [33].

Abstracting beneficial information from the huge data has usually been a challenging mission. DM is a most powerful technology includes a great potential for extracting knowledge-based information of such data [14].

In our study, we address most problems to understand further essential factors which influence on patients who need help fitted the HAs. Through a diversity of DM techniques, our aim is examining and discover the factors which affecting on successfully choosing HA.

2. METHODOLOGY

When the model built, first should evaluate it, and then compare it with parameters settings or other models to estimate the model predictive performance. In this chapter, we will discover as well as describe the measures and strategies of the model evaluation.

2.1 DATASET

For predicting the correct future errors, should evaluate the model on an identically distributed, independent, and diverse set which dissimilar with set which applied for constructing the main pattern. In the case while absence dataset that independent distributed, can divide the prime set for several subsets so that can imitate the impact of gaining further sets.

The generic framework for any ML study can epitomize as follows:

1) ML study begins with choosing dataset holds real data about any problem need to solve. Then this data split for three parts:

    - The part of training dataset;
    - The part of validation dataset;
    - The part of test dataset [124].

2) ML model usually constructed from training set. This operation for the part of "*learning*", where the ML algorithm analyses training set for founding the mathematical model which representing this set [125].

3) This model should evaluate upon a validation set for evaluating its performance. Within this phase, if the performance levels of resulting are unacceptable, the learning phase revisited for changing the parameters of algorithm to update the model. Once reaches the validation performance an acceptable specified level, the model then evaluated upon the test set. Thereafter, the results gained here represent final score of this model besides cannot be more tweaked [126].

All these parts of dataset extracted from the main dataset.

Training set, validation set, and test set usually selected depending on random sampling within prescribed ratios. Commonly, these ratios specified as 70;15;15, such that 70% from the data will be used for training data as well as 15% used for each of validation and testing data. Other ratios used depend on the data variety and volume. Ordinarily, it preferable for keeping data as much as possible into training set in the purpose of acquire a stronger model [127].

Figure 2.1 below displays the classification and distribution datasets:

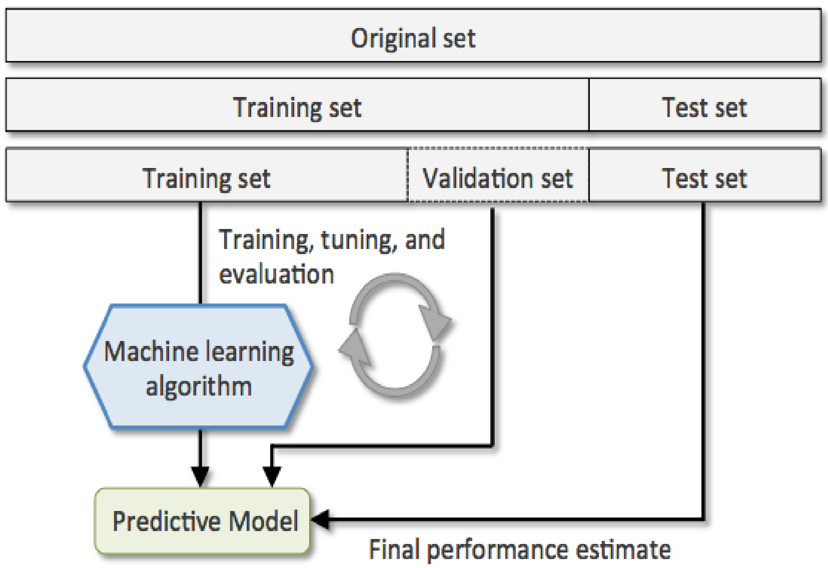


Figure 2.1: The dataset distribution parts [124]

Within validation phase, the data samples utilize to supply inequilateral evaluations for the model proper to the training dataset via hyperparameters of setting model. Such evaluation then will be more equilateral where incorporate skills upon validation data within model configuration. Thus, the validation dataset prevents from training this model which is utilized to give a model skill estimate while model's hyperparameters tuning [32].

It different substantially comparing with test dataset, which is also prevented from the model training, but it used instead for awarding an equilateral predicted for proficiency of comprehensive required pattern if either selecting or comparing amongst final patterns. Usually, the phrase "validation set" utilized mutually with phrase "test set" as well it mentions to a piece of data prevented from training the pattern [128].

Summary definitions for these three terms of datasets are listed below:

- **Training Dataset**: Is a data sample utilized to proper new pattern;
- **Validation Dataset**: The instance of dataset which utilize to present an equitable evaluation for a pattern fitting upon training dataset during setting pattern hyperparameters. Such evaluation will be inequitable in proficiency upon the validation data combined into pattern composition;
- **Test Dataset**: Dataset instance utilizes for evaluating the effectiveness of final pattern. Basically, it an examples' set which applied to assess a fully classifier performance [129].

Can create the specific of datasets types via a pseudocode, as it shown in Algorithm #1 below:

Algorithm #1:

```
# data division
data := ..
train; validation; test := division (data)
#setting pattern hyperparameter
parameters := ..
for param in parameters:
    pattern := fitting(train, params)
    proficiency := evaluate(pattern, validation)
    #evaluate final pattern for comparing with others
     pattern := fitting(train)
     proficiency := evaluate(pattern, test)
```

Algorithm #2 describes these processes in more details.

Let T_{tr} refers to training set, T_v refers to validation set, and T_t refers to test set.

Algorithm #2:

Algorithm 1 Split dataset to Train-Validation-Test

Input: *T (*as main dataset); function error performance; computational model "$L_1, \cdots, L_m$; $m >= 1$"

Divide *T* for three subsets T_{tr}(training); T_v(validation); and T_t(testing).

For $j := 1, \cdots, m$;

    Train the model L_j upon T_{tr} then utilize T_v for assessing the performance of the model;

    $E^j_v := \text{error}(L_j(T_v))$.

    End training; if the stop-criterion of E^j_v is acceptable and satisfied.

For j ;= $1, \cdots, m$;

    Evaluate the final patterns' performance upon T_t:

    E^j_t ;= error($L_j(T_t)$).

2.1.1 Dataset Comparison

Sometimes, we need for comparing several sets of data, if these sets have similar distributions, like when original dataset divided into testing and training sets, will expect that the representative samples for each of sets distributions and subset are the same (with a deviation specific tolerance). When assess the splitting algorithms, then one of the criterions of algorithm will have an ability for dividing the main set into two or more symmetry subsets [130].

For compared any sets of data dispensation, have to apply one of statistical tests within Null Hypothesis which the datasets dispensation is same. Usually, these types of tests are known as "*goodness-of-fit"* tests [131].

2.1.2 Dataset Created

Choosing datasets carefully is extremely significant for experiment since the data selected needs to be applicable and dependable. The applicable data means the data which necessarily ready for classification and within rectify format.

Moreover, not all data types are proper to specified for classification methods or decision tree learning. Therefore, the datasets ought to be selected accurately, besides it must be pre-processed if it is new or not previously processed [2].

Within this study, a new dataset created that consists of 72 columns (fields) distributed upon 71 columns for data specifics and one extra field for class. All fields of this dataset are categorical; as well as it holds a small percentage of missing values.

Concerning to the actual data we have, we collected about 17,300 patients' records during working with the "*Hearing and Speech Evaluation Unit of Al-Basrah General Hospital*" and also from "*Al Nour clinic for Hearing and Speech Audiometry*" in Iraq through continuous work along 8 years at these places [1].

For collecting data, we prepared a form of general information which filled up by the audiologists as well specialist doctors, then this information saved to use it later within computer application of data processing. Thereafter, analyzed this data carefully and accurately for determining the accurate and correct patient's condition diagnosis since this phase represents a most importance task which is detecting and creating the appropriate class to this dataset from information analysis.

Finally, imposed specific data restrictions to implement the algorithms of data cleaning for the aim of extracting final dataset which ready for any test of DM techniques. Below we mention to the standard patient information form, which we adopted for the data collection:

Audiology Patients History Survey

Name: ________________________ Date Of Birth: ______________ Today's Date: ___/___/___

General information:

What the major reason makes you coming today?

____

Since when have you feeling hearing impaired?

____

What do you think may be the reason for your hearing impaired?

____

Was the beginning of the injury mild, then worsened, or sudden?

____

Which ear can hear better than the other? □Right ear, □Left ear, or □Both are same

Do you hear sometimes better comparing with other times?

□YES, I do □NO, always same

Do you previously expose to vocational or entertainment noise? (E.g.: explosions, Turbines, big machines, guns, music....etc.). □YES, I do □NO, I do not

If yes, explain in brief please:

____

Does any of your family suffer from hearing problems?

□NO ……. □YES, who?…………………….

Did you make hearing evaluated before? □NO, I did not

□YES, I did. when? …………………….. and what the outcomes?

____

Did you meet specialist physician before? □NO, I did not

□YES, I did. when? …………………….. and where?

____

Medical

Did you suffer from ear pains or ear discharge through the last 3 months? □ YES □ NO

Did you ever make a medical or surgical therapy for one or both ears? □NO, I did not

□YES, I did. During what age? ________________________

Did you have vertigo, or balance problems before? □ Yes, I have □ NO I have not

Do you distinguish any kind of tinnitus such as roaring, ringing, or buzzing? □NO, I do not

□YES, I do. which ear? □ Right ear , □ Left ear, or □Both ears

How frequent? _______________

Is it annoying? □YES, it is □NO, it is not

Explain the type of tinnitus sound that you are hearing, please:

____

List the medicaments (even if non-receipts) you are using now or have used recently, Please:

1. ____

2. ____

3. ____

Do you suffer from ulcers, bleeding, or discharge presently? □YES, I do □NO, I do not

Hearing History

Do you feel hardness hearing/comprehension within any of activities below?

□ Watching screens (e.g. TV) □ Cafes □ Conventions and meetings

□ Telephone □ Movies □ Attending places of worship

Do you feel hearing disturbance from any of statuses below?

□ Telephone ringing □ Doorbell □ Alarm Clock

□ Alarms of smoke detector □ Baby crying □ Siren and Bells

List places where you feel the more hardness hearing or comprehension:

1. ____

2. ____

3. ____

Which ear you are utilized for answering the telephone? □ Right ear □ Left ear

Do you use left hand or right hand usually? □ Right hand □ Left hand

Any other details concerning to your hear situation you feel it may paramount for an Audiologist to inform?

____

Hearing Aid History

Have you utilized a HA before? □YES, I have □NO, I have not

Do you utilize a HA now? □NO, I do not

□YES, I do. What the time have you been started wearing a HA?

____

Which ear that you are using for the HA?	□ Right ear	□ Left ear
Do you utilize the HA regularly?	□YES, I do	□NO, I do not
Do you think you obtain a help from your HA?	□YES, I am	□NO, I am not

Mention the problem(s) you encounter with using a HA:

1. ____

2. ____

What desiderate you enhance with your present HA?

____

Whom we have to gratitude for advising you to consult us?

____

Have you suffered from one or more of the cases below?

□ Arthritis or rheumatism	□ Diabetes Class I	□ Mumps
□ Allergies reactions	□ Diabetes Class II	□ Pacemaker
□ Bell's Palsy	□ Hepatitis disease (a, b, c)	□ Parkinson's
□ Cancerous disease	□ Sclerosis (MS)	□ Scarlet Fever
□ (Type/Therapy: _____)	□ Frequent Fevers	□ Seizures
□ Concussion/Calvary Break	□ Human Immunodeficiency	□ Stroke/TIA
□ High Blood Pressure	Virus (HIV)	□ Tuberculosis
□ Mental illness/Alzheimer's	□ Meningitis disease	□ Vision Problem
□ Gloominess/Disquiet	□ Measles, Mumps, Rubella	

2.1.3 Data Cleaning

Basically, our data subjected to an accurate and thorough analysis before the cleaning depending on a carefully studied medical diagnosis as well as the information inclusive most important factors which affect directly to the appropriate HA selection for APs [132].

There are important conditions related with cleaning the data to gain final dataset which can be ready for applying DM techniques, as mentioned below:

- Remove duplicate entries or corresponding record(s).
- Remove all records which have missing data or empty field(s).
- Remove all records which have wrong symbol(s).
- Remove all records which have incorrect or invalid data.
- Remove all records which have duplicate occurrences.
- Remove all records which have inconsistent values of data.

After data cleaning, we obtained 210 records of patients' cases which ready to exploration via using DM techniques to predict certified factors related to HAs such as HA type and high-power HA which necessary for Aps.

Figure 2.2a shows the Entity-Relationship Diagram (ERD) of data cleaning process.

Figure 2.2b shows the Data-Cleaning Framework.

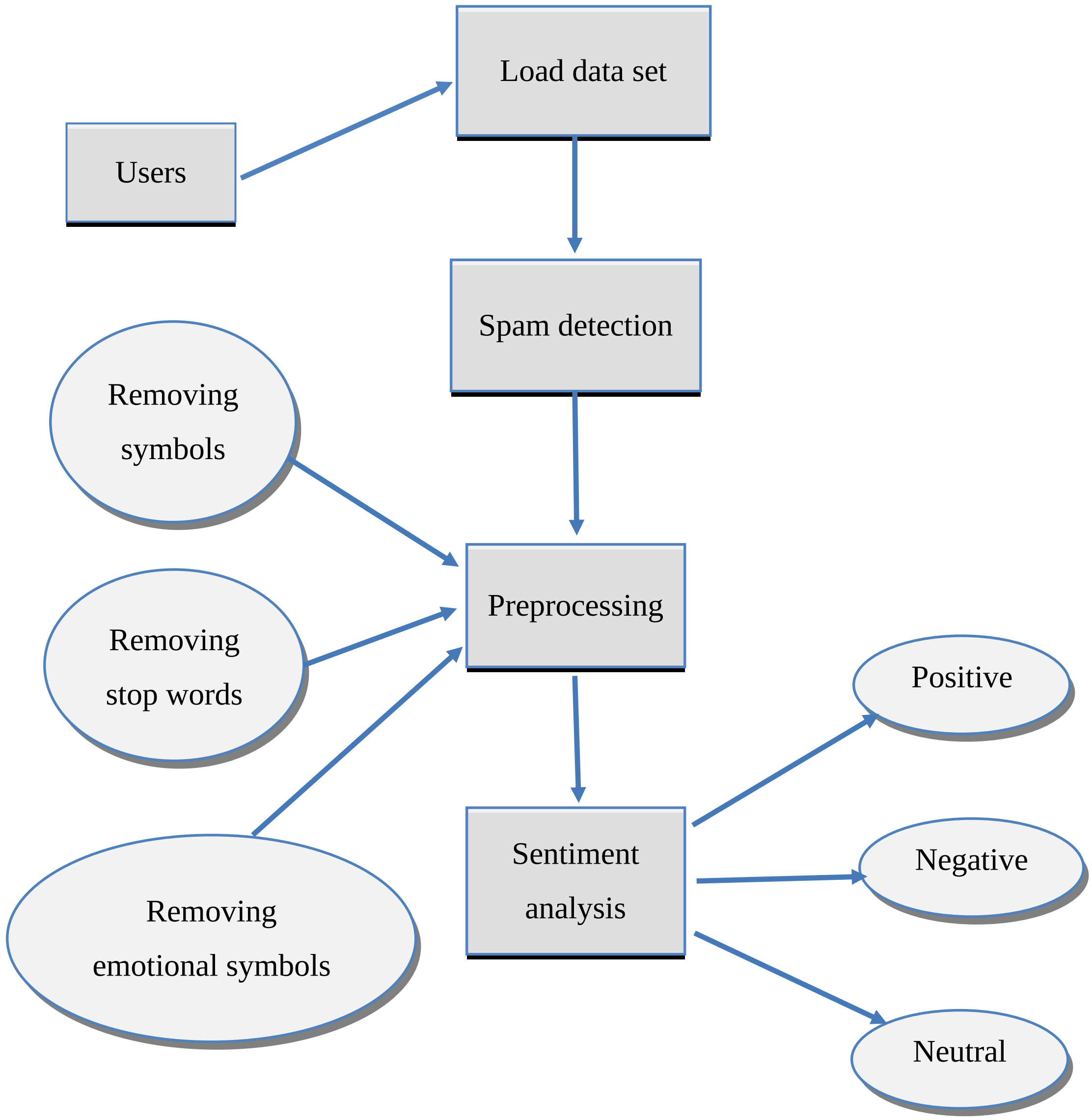


Figure 2.2a: ERD of data cleaning process

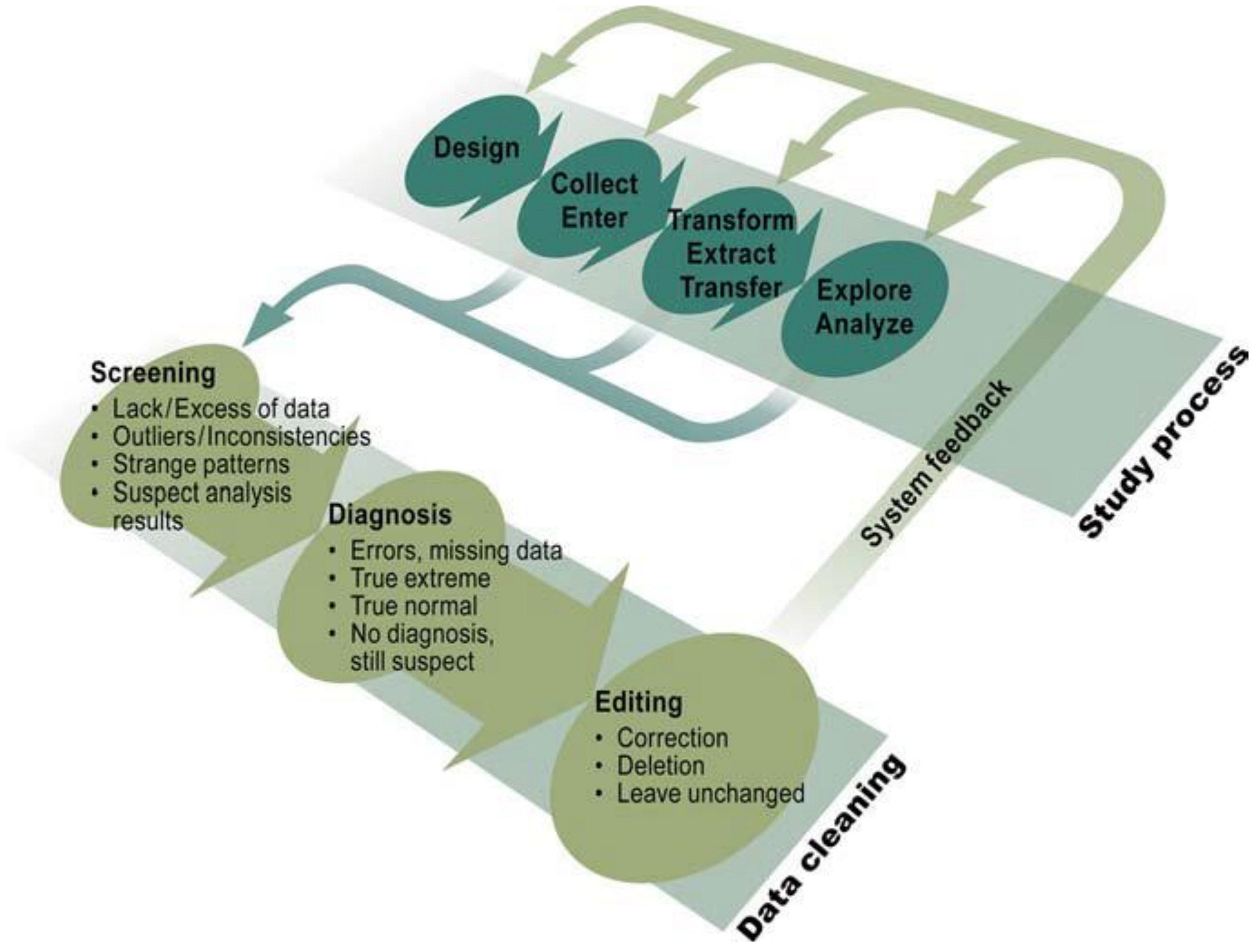


Figure 2.2b: The Data-Cleaning Framework [133]

Figure 2.3 shows the screenshot for last part of our dataset attributes.

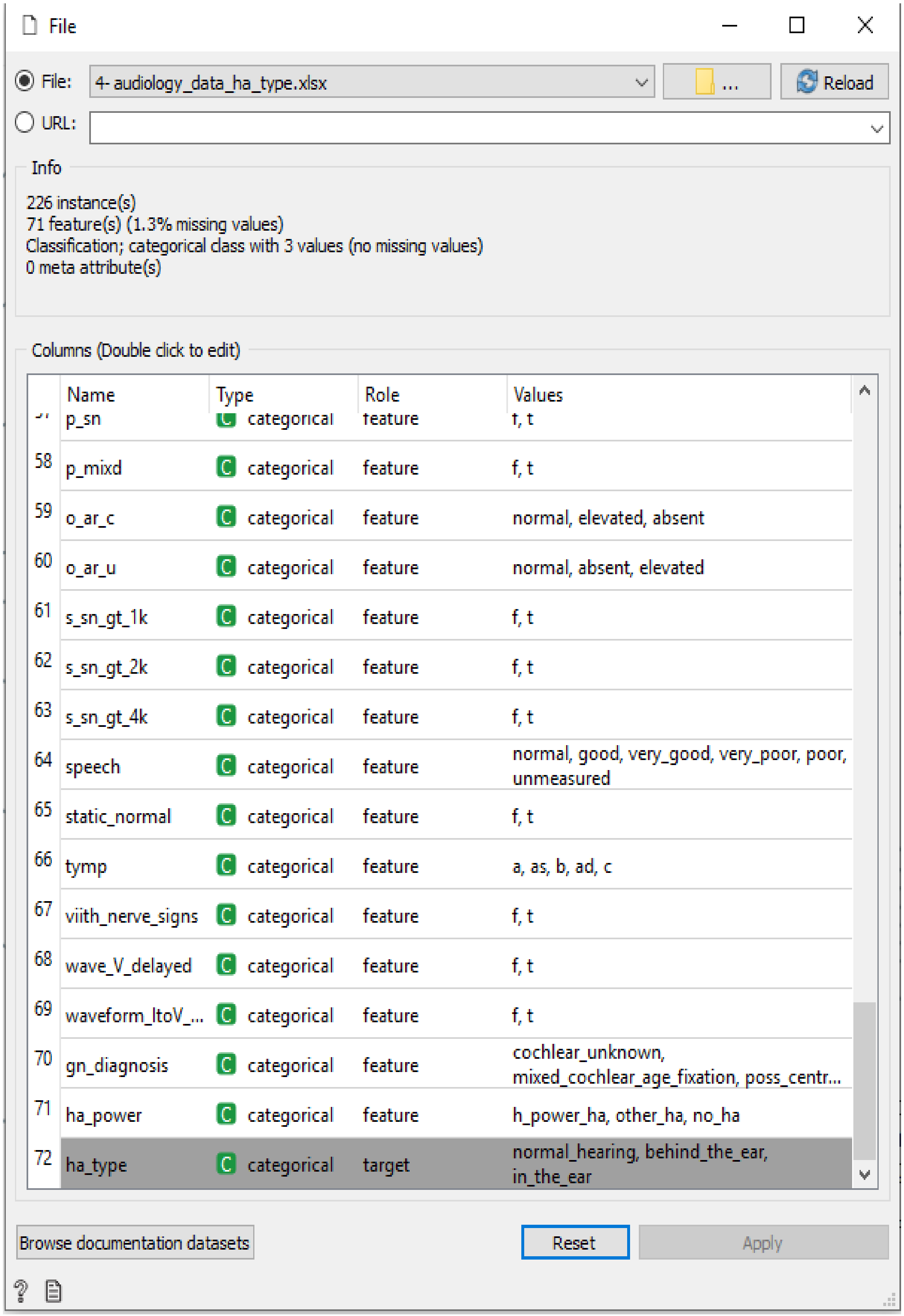


Figure 2.3: The screenshot for last part of our dataset attributes

Table 2.1 illustrates the Audiology Dataset attributes fields definitions.

Table 2.1: Audiology dataset fields definition

I.	Field	Attribute	Definition
1	age_lt_18:	f, t.	Patients' age less than 18
2	age_gt_60:	f, t.	Patients' age greater than 60
3	air():	mild, moderate, severe, normal, profound	Air conduction function
4	airBoneGap:	f, t.	The gap value between air and bone conductions of patient (If there is a gap or not)
5	ar_c():	normal, elevated, absent	Tunic fork air function
6	ar_u():	normal, elevated, absent	Tunic fork bone function
7	bone():	mild, moderate, normal, unmeasured	Bone conduction function
8	boneAbnormal:	f, t.	Test hearing of ear bone is abnormal
9	history_buzzing:	f, t.	The patient has history of buzzing
10	history_dizziness:	f, t.	The patient has history of dizziness
11	history_fluctuating:	f, t.	The patient has history of fluctuating
12	history_fullness:	f, t.	The patient has history of fullness
13	history_heredity:	f, t.	The patient has history of heredity
14	history_nausea:	f, t.	The patient has history of nausea

15	history_noise:	f, t.	The patient has history of noise
16	history_recruitment:	f, t.	The patient has history of recruitment
17	history_ringing:	f, t.	The patient has history of ringing
18	history_roaring:	f, t.	The patient has history of roaring
19	history_vomiting:	f, t.	The patient has history of vomiting
20	late_wave_poor:	f, t.	The late wave is poor
21	m_cond_at_2k:	f, t.	Mild conductive HL at 2k
22	m_cond_lt_1k:	f, t.	Mild conductive HL less than 1k
23	m_cond_gt_1k:	f, t.	Mild conductive HL greater than 1k
24	m_m_cond_gt_2k:	f, t.	Mild-Moderate conductive HL greater than 2k
25	m_sn:	f, t.	Mild sensory neural HL
26	m_m_sn:	f, t.	Mild-Moderate sensory neural HL
27	m_m_sn_gt_1k:	f, t.	Mild-Moderate sensory neural HL greater than 1k
28	m_m_sn_gt_2k:	f, t.	Mild-Moderate sensory neural HL greater than 2k
29	m_m_sn_gt_500:	f, t.	Mild-Moderate sensory neural HL greater than 500k

30	s_p_sn_gt_2k:	f, t.	Severe-Profound sensory neural HL greater than 2k
31	m_s_cond_gt_500:	f, t.	Moderate-Severe conductive HL greater than 500
32	m_s_sn:	f, t.	Moderate-Severe sensory neural HL
33	m_s_sn_gt_1k:	f, t.	Moderate-Severe sensory neural HL greater than 1k
34	m_s_sn_gt_2k:	f, t.	Moderate-Severe sensory neural HL greater than 2k
35	m_s_sn_gt_3k:	f, t.	Moderate-Severe sensory neural HL greater than 3k
36	m_s_sn_gt_4k:	f, t.	Moderate-Severe sensory neural HL greater than 4k
37	m_sn_2_3k:	f, t.	Mild sensory neural HL from 2-3k
38	m_sn_gt_1k:	f, t.	Mild sensory neural HL greater than 1k
39	m_sn_gt_2k:	f, t.	Mild sensory neural HL greater than 2k
40	m_sn_gt_3k:	f, t.	Mild sensory neural HL greater than 3k
41	m_sn_gt_4k:	f, t.	Mild sensory neural HL greater than 4k
42	m_sn_gt_500:	f, t.	Mild sensory neural HL greater than 500k
43	m_sn_gt_6k:	f, t.	Mild sensory neural HL greater than 6k

44	m_sn_lt_1k:	f, t.	Mild sensory neural HL less than 1k
45	m_sn_lt_2k:	f, t.	Mild sensory neural HL less than 2k
46	m_sn_lt_3k:	f, t.	Mild sensory neural HL less than 3k
47	middle_wave_poor:	f, t.	The middle wave is poor
48	mod_cond_gt_4k:	f, t.	Moderate conductive HL greater than 4k
49	mod_mixed:	f, t.	Moderate mixed HL
50	mod_s_mixed:	f, t.	Moderate-severe mixed HL
51	mod_s_sn_gt_500:	f, t.	Moderate-severe sensory neural HL greater than 500k
52	mod_sn:	f, t.	Moderate sensory neural HL
53	mod_sn_gt_1k:	f, t.	Moderate sensory neural HL greater than 1k
54	mod_sn_gt_2k:	f, t.	Moderate sensory neural HL greater than 2k
55	mod_sn_gt_3k:	f, t.	Moderate sensory neural HL greater than 3k
56	mod_sn_gt_4k:	f, t.	Moderate sensory neural HL greater than 4k
57	p_sn:	f, t.	Profound sensory neural HL
58	p_mixd:	f, t.	Profound mixed HL
59	o_ar_c():	normal, elevated, absent.	Vibration sound of tuning fork behind the ear

60	o_ar_u():	normal, elevated, absent.	Vibration sound of tuning fork in front of the ear
61	s_sn_gt_1k:	f, t.	Severe sensory neural HL greater than 1k
62	s_sn_gt_2k:	f, t.	Severe sensory neural HL greater than 2k
63	s_sn_gt_4k:	f, t.	Severe sensory neural HL greater than 4k
64	speech():	normal, good, very_good, very_poor, poor, unmeasured	Speech measure function
65	statc_norml:	f, t	Static frequencies
66	tmp():	a, as, b, ad, c	Tympanogram type test measure
67	viith_nerve_signs:	f, t	Signs from VIIth nerve
68	wave_V_delayed:	f, t	The wave V is delayed
69	wavform_ItV_prolonge:	f, t	Identifier unique for each instance
70	gn_diagnosis:	cochlea_unknown	The case of cochlear is unknown
		mixd_cochlea_age_fxation	Mixed case: Cochlear age and cochlear fixation
		poss_centrl	Indicating possible central hearing loss; since ABR testing
		mixd_cochlea_age_otitis_mdia	Mixed case: Cochlear age and otitis media
		mixd_poss_nois_om	Mixed case: Possible noise and otitis media

		cochlear_age	The case caused by cochlear age
		normal_ear	Normal ear and normal hearing
		cochlear_poss_noise	The case possibly caused by cochlear noise
		cochlear_age_and_noise	The case caused by cochlear old age and noise
		acoustc_nuroma	The case caused by acoustic neuroma
		mixd_cochlea_unk_ser_om	Mixed case: Cochlear unknown and severe otitis media
		conductive_discontinuity	The case caused by conductive discontinuity
		retrocochlear_unknown	The case: retro cochlear is unknown
		conductive_fixation	The case caused by conductive fixation
		bells_palsy	Case of bell's palsy
		cochlear_noise_and_heredity	The case caused by cochlear noise and heredity
		mixd_cochlea_unk_fxation	Mixed cases: Cochlear unknown and cochlear fixation
		otitis_media	The case caused by otitis media

		possible_menieres	The case possibly caused by Meniere's disease
		possible_brainstem_disorder	The case possibly caused by brainstem disturbance
		cochlea_age_plus_pos_meniere	The case caused by cochlear age plus possibly Meniere's disease
		mix_cochlea_age_s_om	Mixed case: Old age of cochlear and severe otitis media
		mix_cochlea_unk_discontinuity	Mixed case: Cochlear unknown and damage ossicular chain (may be caused within inner ear (cochlea))
		mixed_poss_central_om	Mixed case: Indicating possible central hearing loss; since ABR testing and otitis media
71	1) ha_power: 2) ha_type:	1) h_power_ha, other_ha, no_ha 2) in_the_ear, behind_the_ear, normal_hearing	1) High Power Hearing Aid type 2) Hearing Aid type
72	Class:	1) in_the_ear, behind_the_ear, normal_hearing 2) h_power_ha, other_ha, no_ha	

Table 2.2 illustrates the details of new Audiology dataset created for specific HA factors:

Table 2.2: The details of new Audiology dataset created

New Datasets summary	
Dataset Source	collected by a researcher
Dataset Name	Audiology for HA
Number of cases	216
Number of attributes	71
Number of classes attributes	1
Number of class classifications	3
Number of Categorical Attributes	72
Number of Numeric Attributes	0
Continuous data	0
Missing values	2.8 %
Uniform distribution	No

2.2 MODEL VALUATION

The valuation of any model is incredibly significant, however usually underrated in portion of assessment and constructing of the model. After preprocessed and preparation the main dataset, we need to build strong model which able for accurately predicting of future observations [134].

Usually, the purpose is not a pattern which match the training dataset perfectly, instead, we are looking for dependable pattern after deployment within real use. For this goal, there are two necessary phases of the model evaluation:

1) Evaluate the model to estimate model parameters during learning phase. This refers to one of the model choice part when selecting the better results pattern, as well this stage was known as a "validation phase".

It is unnecessary chooses a model which bestead fits a specific data set. The model which will learned, contains just an implicit phenomenon without noise. While the "*Over-Fitted"* term refers to the model which captures the noise [135].

2) Evaluate the model selected to consider the valuation of real performance for the model on novel hidden dataset.

As a conclusion, can summarize this process in the steps shown below:

1) Selecting phase (Model selecting):

   a) Training phase (Model learning);

   b) Validation phase (Model validation).

2) Testing phase (Model assessment).

2.2.1 Evaluation Methods

Over constructing new model, it is necessary to analyze the effectiveness valuation in the purpose of assess or confirm it as mentioned before.

Further techniques are used for validating the model, however, these techniques not always applicable or even adequate within all conditions. So, always be careful when choosing the reliable and more appropriate technique to reach the exact purpose [136].

There are number of common valuation methods such as:

a) Gathering appropriate data for evaluation

Utilizing the collected dataset within separate experiment is a most preferred and best method for pattern evaluation. Besides, it represents the best choice which gives the actual estimation of whole pattern performance upon collected data [135].

Only the novel datasets capable to discover the bias in all processes of earlier sampling. Thus, this way is much easier if there is an ability to iterate the experiment as well as the sampling process easily. Unfortunately, in some situations may not being able to gather independent novel dataset for such task, either for the reason of experiment high cost or another process unrepeatability [129].

b) Utilize same dataset that used for constructing model

c) Usually, utilize same dataset for building as well as evaluation the models, leads to sanguine actual effectiveness estimation because of the favorable bias. Method such this is not preferable as well as if there is another way it cannot adopted to any evaluation of the model at all [129].

d) Keep part of learning dataset to evaluate the model

The learning data saved part in training phase is most common method of how to transact with the separate dataset absence of the evaluation. Since choice of a saved part of the main dataset usually not an effortless way, so, several techniques have been creating. Dividing datasets required to obtain such impact of two separated datasets. However, that's not true, just newly data collected can refers to the bias within training dataset [137].

e) Re-sampling methods

Re-sampling allows the division of dataset into subsets if there is no independent set for the assessment of prediction performance or model validation. The methods of choosing sample decrease training dataset via eliminate the useless samples of the model estimating parameters, thus it can be accelerate the response time and learning phase as well [138].

Figure 2.4 shows the datasets types flowchart which build the main model.

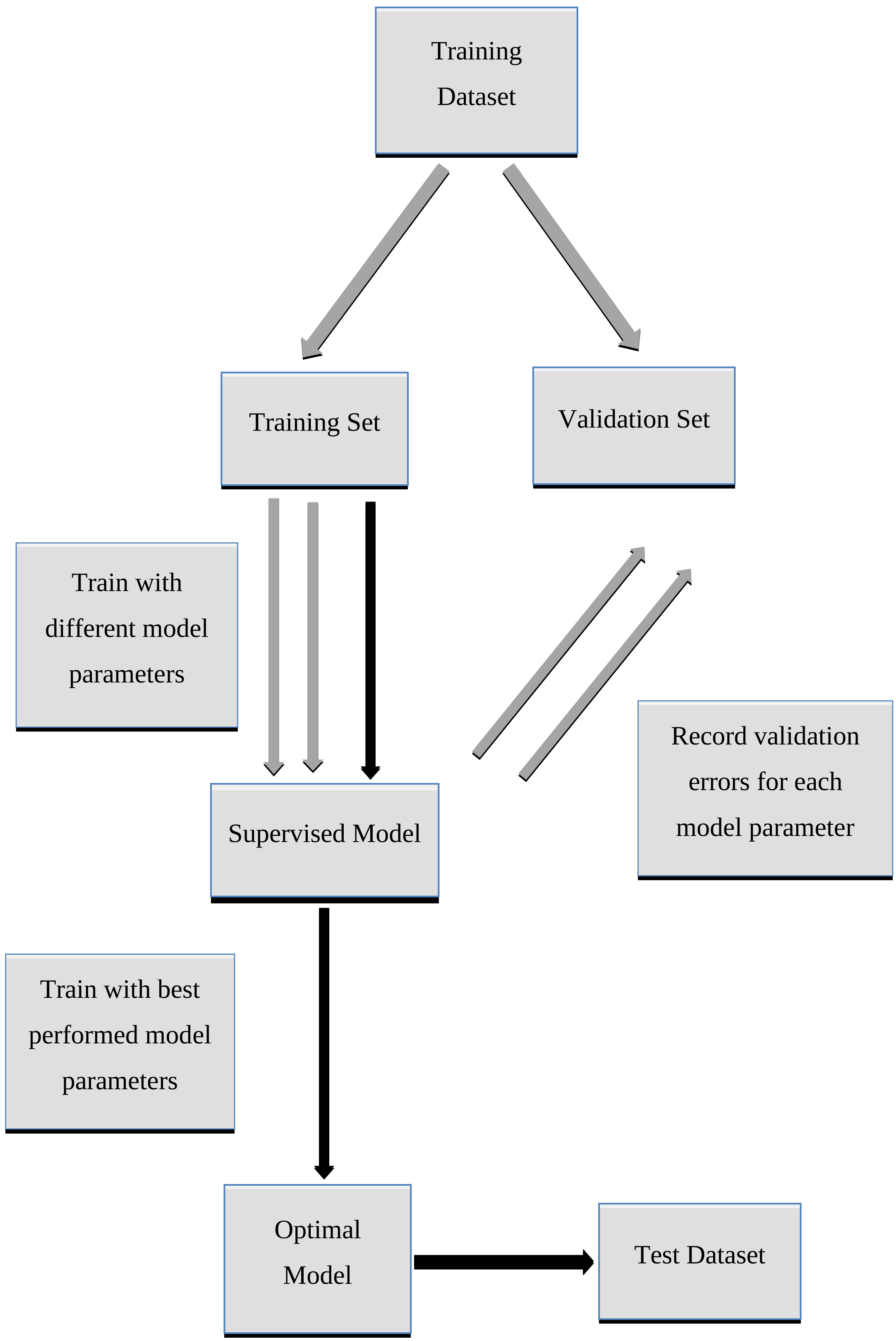


Figure 2.4: General flowchart for main model. Gray vectors refer to validation process and Black vectors refer to training and final test process [130]

2.2.2 Evaluation Measures

There is great divergence of performance measures for evaluating both of predictor and classifier. Though, a proper measurement may perfect to evaluate the model within certain area while inappropriate for another area and vice versa. The selection of evaluation measurement depends upon the given problem and domain utilized [139].

when we learned some of models and there is a need to choose the best of them, some of qualified measurements usually should use for estimating the model effectiveness and then select the pattern of elevated performance easily. This usually an enough method for pattern selection. But may a problem found when need for evaluating the enhancement of model efficiency, mostly when need for establish that one model is outperforms another upon specific learning mission, should implement the statistical significance tests as well verify an improved performance of such hypothesis.

2.2.3 Best Hearing Aid Style

Figure 2.5 illustrates the Entity Relationship Diagram (ERD) of the principal factors that detect which HA style and type is suitable for any AP who suffer from hearing impairment or hearing loss. Depends on these factors, we build our models and choosing the best DM techniques for predicting and estimating.

Below the illustrate of an abbreviation (which appeared in the Figure 13) of the Hearing Aid types:

ITE: In the ear

BTE: Behind the ear

ALDs: Assistive Listening Devices

CIC: Completely-In-the-Cannel

ITC: In-The-Canal Hearing Aid

RIC: Receiver-In-the-Canal

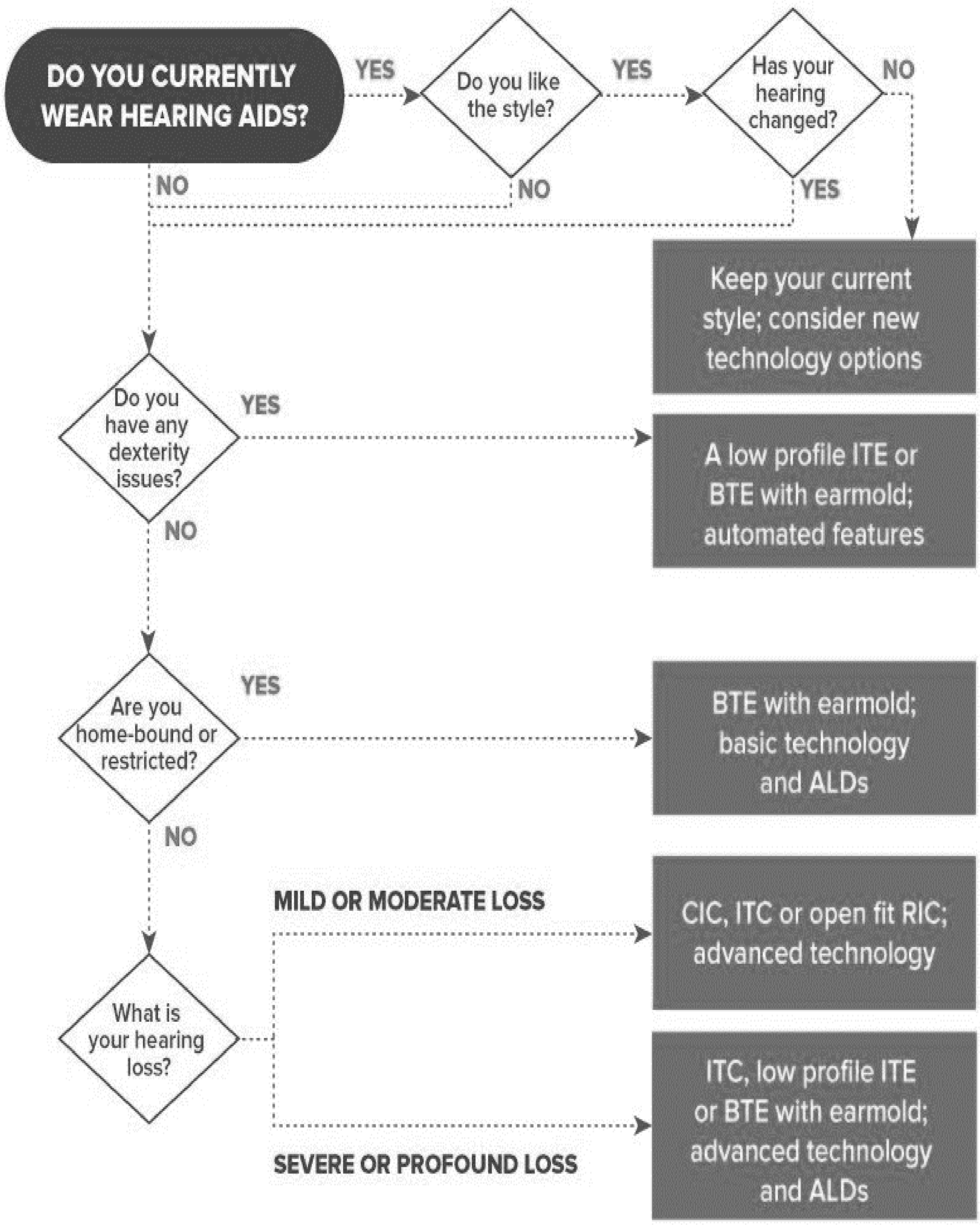


Figure 2.5: ERD illustrates the principal factors of choosing Hearing Aid [140]

2.3 MAIN MODEL

Within our research we described and compared both supervised and unsupervised techniques to classify, predict, estimate, cluster, and associate the perfect results for some factors related with choosing the right HA for APs. We used Orange Canvas to design and analyze our models and Python for coding via Anaconda application.

2.3.1 Orange Canvas Modeler

Orange is an application of Python Open Source Library works with both of ML and DM. It is the open source instrument for visualization and data analysis developed in the "Bioinformatics Lab./ Faculty of Computer and Information Science/ University of Ljubljana". It provides the important tools for performing DM tasks via graphical interface as well as by utilize Python coding script. Orange filled with a lot of features for components and analytics of ML, the main feature of it is basically the possible expansion high level via utilizing “add-ons” which afford extra possibilities of main package for missions like bioinformatics, text mining, and many of DM techniques [141].

The visual interface which called "Orange Canvas Modeler" simple utilize as it offers the functionalities comprehensible division via nine various groups–data, classification, visualization, evaluation, regression, association, unsupervised learning, prototypes, and bioinformatics. To increase this ease of use, introduced the widgets as the functionalities representation which can connected and placed between them within visualization area of the most axiomatic way [142].

Orange Canvas supports the following component:

I. Interactive Data Visualization

It performs simple analysis of data through intelligent visualization of it. Further, it explores scatter plots and box plots, statistical distributions, hierarchical clustering, dive deeper within decision trees, MDS, linear projections, and heatmaps. Even the multidimensional data it become sensible with 2dimensional, particularly within intelligent selections and ranking attribute [143].

II. Visual Programming

Exploration the interactive data for rapid analysis qualitative through clean visualizations. The graphical user interface lets to transact with analysis of exploratory data rather than of coding, whereas intelligent assumptions make creating rapid prototypes for the workflow of data analysis extremely easy. This phase of Orange summarizes in placing the widgets upon the canvas, link them, load the datasets, and then get a comprehensive view for the model [144].

III. Add-ons Extend Functionality

Several add-ons which available in Orange for mining the data from external sources, besides perform text mining and processing the natural language, conduct the analysis of network, do mining of association rules and deduce frequent itemset. Additionally, molecular biologists and bioinformaticians can utilize Orange to classify genes according to their differential expressions and perform the analysis of enrichment [145].

IV. Component-Based DM

With Orange modeling application, data analyzing via accumulate the ingredients into workflow space. These ingredients named widgets which includes pre-processing, data retrieval, modelling, evaluation task, or visualization. Combining various widgets within one workflow enables to construct the comprehensive schemas of data analysis [142].

V. Interactive Data Exploration

The widgets connect each other through get any data as inputs then transmit models, processed data or out filtered, or any option that widget does for output. For instance, begin from “File” widget which contains any data then link the output with other widgets such as one of DM algorithms or “Data Table” [145]. The modification within any widget, will make all modifications propagated Immediately during downstream workflow. Furthermore, modifying data within “File” widget usually caused responsive trigger within workflow widgets [143]. Such operation so interesting particularly when open new widgets for the aim of find their outcomes for each data changes immediately, methods parameters, or any selections within interactive visualizations.

VI. Intelligent Workflow Interface Design

Orange is restful to use, it starts from *"File"* widget and then propose following widgets which will be linked with it. To more clarify, Orange realizes that may need for Hierarchical Clustering after setting up the "*Distance*" widget. The other standards within widgets are tune with the way which allows to conduct modest analysis even if not knowing much about statistics, ML, or exploratory DM in general [141].

The strong point of Orange is that it offers diverse visualization outputs like maps or scattered matrices, 3D graphs, multiple interfaces like Graphical User Interface (GUI) or the unit of batch processing, pre-processing methods accuracy, and a complete toolbox for more than1500 of processes available [143].

RapidMiner is an important tool for text mining, ML, and DM, as well the business analyses and predictive analyses designed not for data researchers only but for business administrators as well, it offers hundreds of recently methods for modeling, visualization, and data transformation. Further, it awards the users an extensible and powerful API that used for improving the instrument to comprise their techniques. RapidMiner is modifiable within provision of sizes of data sets and configurations where all techniques can be implement within-clusters, within-memory, or within-database [145]. It divides the DM tasks for 7 groups:

1) "Feature Weighting";

2) "Regression, Classification";

3) "Segmentation, Clustering";

4) "Item Set Mining, Association";

5) "Dependency Computation, Correlation";

6) "Model Implementation";

7) “lastly Similarity Computation”.

Within these groups can find diverse algorithms. RapidMiner developed by Java, and it runs in each operating system and major platform [144].

The version (Open source usually) contains limit on a memory size only which can allocate with (1GB), as well the types of data sources accepted are (Excel and .csv only). In short, the main advantages of RapidMiner are:

- Boost the whole computer domains.
- Presents visual interface which abstracts user from the execution details.
- Application Program Interface which supplies configuration versatility as well as extension capabilities.
- Support for in-database, cluster processing, and in-memory.
- Diversity set of the visualization outputs [146].

Other Orange advantage is the qualities above-mentioned make RapidMiner so appealing for the people who unable to spend a lot of time through learning curve [146].

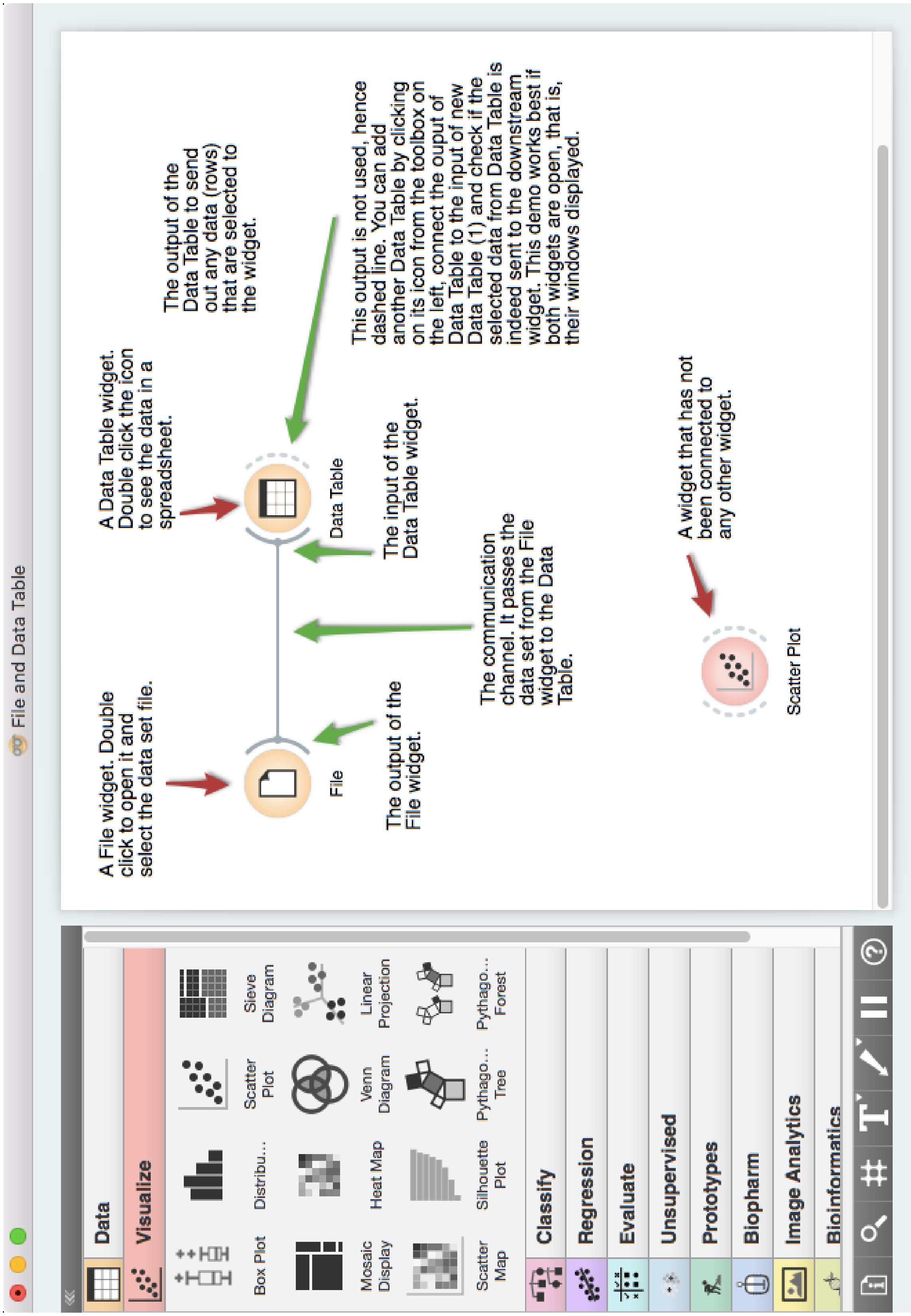
2.3.2 Workflows in Orange

Orange workflows primarily consists of a set of components which visualize, read, and process data. These components known as *"widgets"* The user puts the widgets upon designing panel ("canvas"). The widgets connect via transmission the data through connection channels. The final output of any widget utilized as initial input for other [146].

The workflows construct via dragging any of widgets into the space of canvas panel then joining them via dragging line(s) which start from transmit widget toward receive widget. The outputs of widget place into right side while the inputs into left.

Figure 2.6 shows a screenshot of simple workflow which consists of two widgets connected and one without connections. The widget outputs appear on a right side of canvas, whereas the inputs on a left. The "File" widget fetches required dataset file from the storage location, thereafter forwards it as an output towards "Data Table" which can displays data within spreadsheet, and then for "Scatter Plot" widget which can visualize it. The data dots selected from "Scatter Plot" widget can transfer for other widgets.

Figure 2.6: Screenshot of a simple workflow with two connected widgets using Orange application



2.3.3 Python Script

Python is modern commonly texting language, clear attributes syntax, powerful, non-meddlesome objects-oriented, data structures built-in high-level, introspection full run-time, and functional programming items. Python presents inclusive procedures library to system utilities, string processing, data compression, Internet-linked Protocols, etc., such that library can complement with libraries available for free [142].

Widget of Python Script is prepared to extend the progressive users' functionalities. Generally, extend of functionalities via Python scripting can illustrated in steps below:

Inputs

Data;

Learner;

Classifier;

Object input bound of python object to the in_object variable.

Outputs

Data(Orange.data.Table) dataset restored from the variable out_data;

Learner;

Classifier;

Object python object restored from the variable out_object.

The widget of Python script may utilize to run the Python script in input, if not implemented any suitable functionality within current widget. Main script contains input_distance, input_data, input_classifier, input_object, and input_learner variables within local namespace. When the indication did not receive data yet or may not connected, so those variables have indicated "None".

Next, the executed variables extracted from namespace and then used as widget outputs. The widget on the other hand, can connects with other widgets to visualize the output [142].

For example, the script below would pass upon all receives signals:

Note: It is so important to not modify any of input points within same position.

1. “Info box”: Usually includes on basic operators’ denotations for the script of Orange Python.
2. "Library": Can utilize to control multiple scripts. For instance, to add as well as open new entry, press “+” within editor of Python script. After modified the script, the entry within script editor of "Library" would replace to "unsaved changes". Pressing "Update" for saving the script. The script will remove when selecting it then pressing “-” button.
3. "Execute" within "Run" box: Pressing this key will execute the script of (exec) extension. Each output can capture as well as displayed via "Console" key. Besides, when choose "Auto execute" key, the implement will start any time that inputs changed.
4. Editor of Python script: Which take position on left side of the editor script can utilized for script editing.
5. "Console": Displays an output of a script.

The following code explains how to import Orange library through Python for SVM algorithm as supervised example:

```
>>> import Orange Modeler;

>>> data := Orange.data.Table("audiology_ha ");

>>> data := data.filtered(audiology_ha =["xx", "xx"]);

>>> data := data.translated(Orange.data.Domain(data.domain.feature, \

>>> Orange.data.preprocessing.RemoveUnusedValue(data.domain.class_var, data)));

>>> folds := Orange.data.samples.SubsetIndices;
```

```
>>> train := data.selected(fold, 0);

>>> test := data.selected(fold, 1);

>>> svm := Orange.classification.svm.SVMLearner(train);

>>> corr := 0.0;

>>> for instant in test:

>>>       if svm(instant) := instant.get_class():

>>>             corr := 1;

>>>              print "Accuracy: ", corr / len(test)
```

2.3.4 Supervised Model

Supervised DM, also named as a direct DM, aims to explaining the relationships when they found. It is a predictive approach so that classified the variables involved as dependent and explanatory, in addition the essential aim is an achievement of a liaison among them as with the Regression Analysis [147].

Let us clarify the benefits of Orange through the data analysis using Python shell with supervised algorithms used in our main model:

```
>>> import Orange Modeler;

>>> data := Orange.data.Table("audiology_ha ");

>>> len(data);

210

>>> svm := Orange.classification.svm.SVMLearner();

>>> cn := Orange.classification.cn.CNLearner();

>>> nbc := Orange.classification.bayes.NaiveLearner();
```

```
>>> lr := Orange.classification.lr.LRLearner();

>>> rf := Orange.classification.rf.RFLearner();

>>> ab := Orange.classification.ab.ABLearner();

>>> nn := Orange.classification.nn.NNLearner();

>>> stack := Orange.ensemble.stacking. StackedClassificationLearner
        ([svm,cn,nbc,lr,rf,ab,nn]);

>>> result := Orange.evaluation.testing. cross_validation([svm, cn, nbc, lr, rf, ab ,nn, stack],
        data);

>>> Orange.evaluations.scoring.AUC(result)
```

Figure 2.7 shows the Supervised Training Phase of our main model:

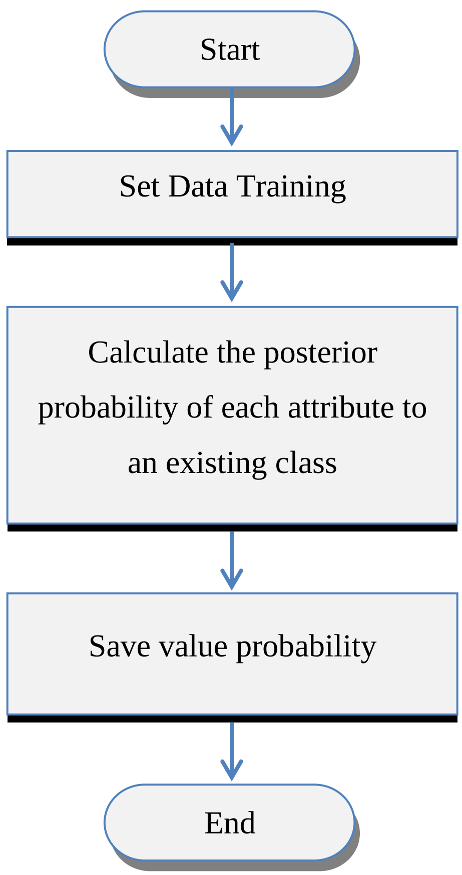


Figure 2.7: The Supervised Training Phase

Figure 2.8 shows the Supervised Testing Phase of our main model:

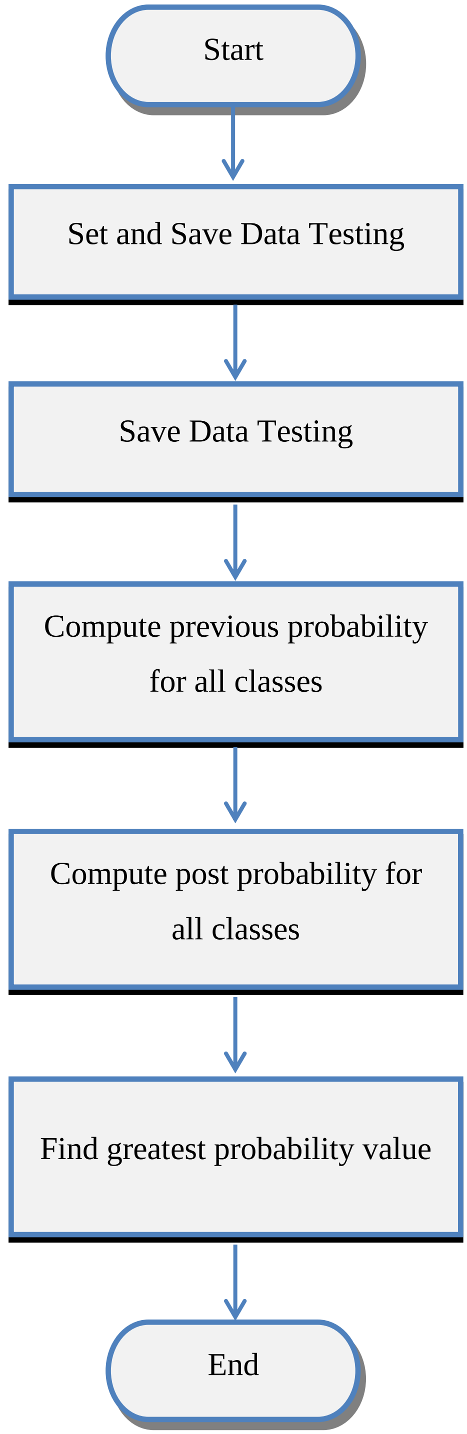


Figure 2.8: The Supervised Testing Phase

Now we illustrate the supervised techniques used for our model with details and before extracting results:

1) **Support Vector Machine (SVM):**

SVMs map inputs to the feature spaces with higher-dimensional.

Input:

- Dataset;
- Preprocessing techniques(s).

Output:

- Learner;
- Model;
- Support Vectors Model.

SVM algorithm a ML method using hyperplane for dividing an attribute space, so that expand the margin among class values or instances. This technique usually yields high results for predictive effectiveness. The application of Orange includes SVM functionality within package of Library for SVMs (LIBSVM) that is the user graphical interface widget.

About tasks of regression, SVM performs a linear regression within feature space of high dimensions utilizing ε-insensitive loss. The estimation precision for SVM algorithm based on perfect settings for kernel parameters, "ε", and "C". Then based on the SVM regression, widget outputs the class predictions.

This widget implements for both regression and classification tasks.

Figure 2.9 shows the SVM widget setting for our model.

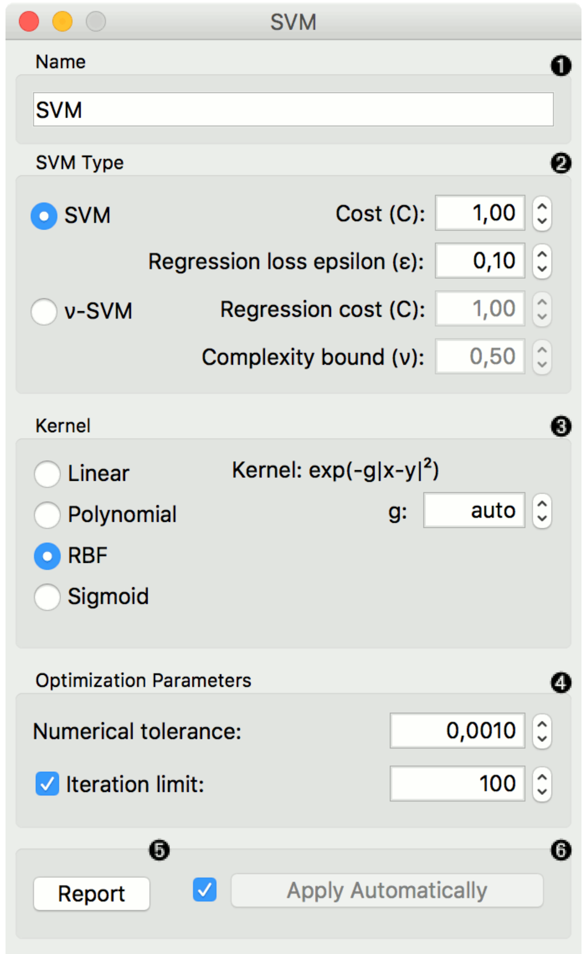


Figure 2.9: The setting of SVM widget for our model

While:

1. The "Name": Can assigned any name for the learner which will appear under it within other widgets. We used default name: "SVM".
2. The "SVM Type": Type of SVM with "test error" settings. We can determine the bounds of test error upon right side:

- "*SVM*":

  Cost: Sanction code for loss so that it applies for regression and classification tasks.

  ε: Epsilon-SVR parameter, used with regression. It refers to the distance of values when no sanction related to predicted values.

- “*v-SVM*”:

  v: *v*-SVR model variable applies for regression and classification tasks. Upper bound for the errors fraction which extract from training phase and the lower bound fraction of SVs.

  Cost: Sanction code of waste so that it utilizes for regression missions only.

3. The “Kernel”: Basically the function which converts the area of attribute into novel attribute area till appropriate a biggest-margin hyperplane, so that this algorithm can produce new pattern using Polynomial, Sigmoid kernels, RBF, and Linear methods. Functions which define kernel usually given on choosing them [148]. The details of constants which involved are listed below:

   - *g*: Gamma constant within kernel function (advised is "1/k"; k: an attribute number, however default 0 if no training dataset widget as well as it should defined manually by user);

   - *c*: Constant c_0 within kernel function (default 0);

   - *d*: Kernel degree (default 3).

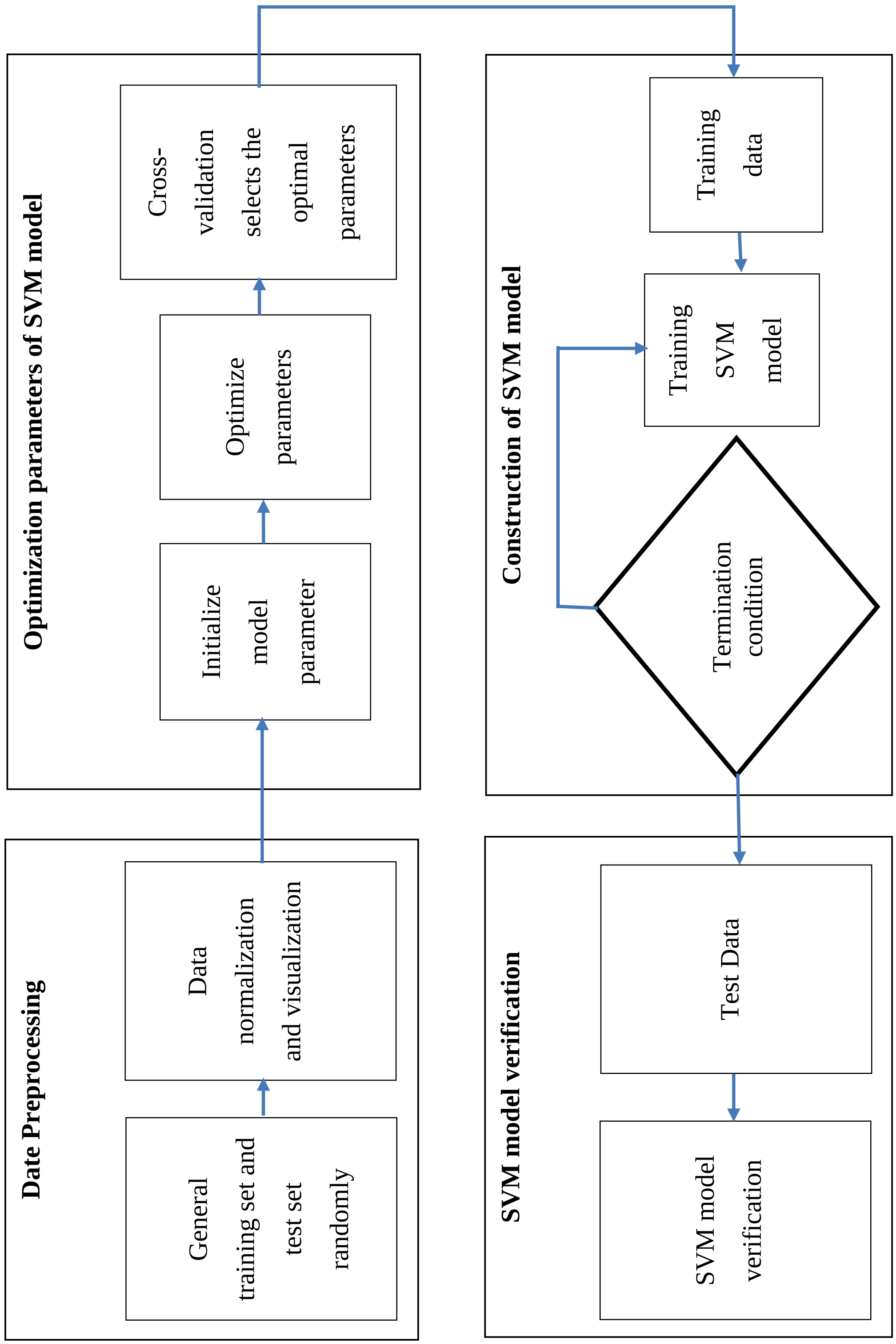
4. The “Optimization Parameters”: Set deviation permitted from the value expected within "Numerical tolerance". Mark the box in the right of "Iteration limit" so can set the iterations maximum number permitted.

5. The “Report” producing.

6. To implement changes, click “Apply”. If choose "Apply Automatically", then the changes will confirm automatically.

Figure 2.10 shows the flowchart of our model with SVM:

Figure 2.10: The ERD of our model with SVM



2) CN2 Rule Induction

The algorithm begins with removal the rules from input rules set that are not significant statistically. when these specific rules coverage is less than that specified by the user, then returned the empty rules set. Otherwise, the set of output rules usually initialized within best rules (according to quality measure of chosen rule) from every decision class (it is one rule of each class). Thereafter, rules that maximize the "2" value (i.e. the rules that cover the biggest number of instances which uncovered yet via the created rules set so far) are added to the set of output rules successively [149]. The rules will be added till they do not gain the user specified coverage. Noting, that within rules set which obtained from this way might be found some redundant rules, which is mean their removal could not cause the unsatisfied of criterion C2. And such rules will be removed within last phase of the filtration algorithm presented. The last step is "Kaplan-Meier" (KM) curves which usually calculated for every rule of the new obtained Model [150].

deduce the rules from dataset by CN2 algorithm.

Input:

- Data;
- Preprocessing technique(s).

Output:

- Learner: CN2 learning;
- CN2 Classifier Rule: the trained model.

"CN2 Rule Induction" algorithm is the classification techniques which constructed for an effective inducement of easy and understandable rules within formula of: "*if condition(s) then prediction class(s)*", and not matter if noisy areas.

CN2 algorithm works for classification only.

Figure 2.11 shows the setting of CN2 Rule Induction widget for our model.

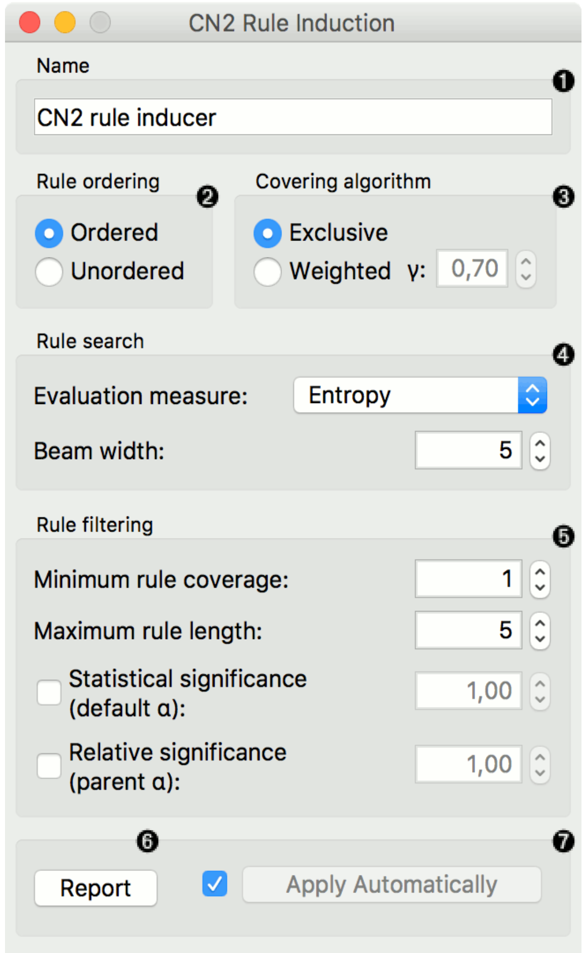


Figure 2.11: The setting of CN2 Rule Induction widget for our model

While:

1. The "Name" that the learner will appears under in the other widgets. We used default name: "CN2 Rule Induction".

2. The "Rule ordering":

   - The "*Ordered*" rules: Deduce rules as beneficial as possible. Find the rule constraints then allocate main class within rule principle.

   - The "*Unordered*" rules: deduce unordered rules (the rule set). Regarding the main learning dataset, learn the rules of every class individually.

3. The "Covering algorithm":

- The "*Exclusive*": after learning instance covering, remove this instance from any further consideration.

- The "*Weighted*": after learning instance covering, reduce its weight (via multiplication with gamma) and reduce its effect on other algorithm iterations.

4. The "Rule search":

- The "*Evaluation measure*": determined the heuristic for evaluating the exist hypotheses:

  - The "*Entropy*" (the measure of content unpredictability),

  - The "*Laplace Accuracy",*

  - The "*Relative Weighted Accuracy*".

- The "*Beam width*"; remember a best rule established thus far as well as consider a fixed of alternatives number.

5. The "Rule filtering":

- "*Minimum rule coverage*": Constructed rules should include least number required of comprehensive instances. Besides, random rules should include most instances of the main class.

- "*Maximum rule length*": Constructed rules might collect mostly a biggest number tolerated conditions.

- "*Statistical significance*" (*a*): The importance examines for reduce or remove most specialized (frequently less applicable) rules regarding to the classis's initial distribution.

- "*Relative significance*" (Parent alpha): the significance testing for reduce or remove most specialized (frequently less applicable) rules regarding to the distribution of parent class.

6. Mark “Apply Automatically” used for the purpose of automatically transferring updates into any widget as well for promptly training the classifier if connected with learning data. Alternatively, after configuration can press “Apply”.

Figure 2.12 shows the workflow of proposed methodology:

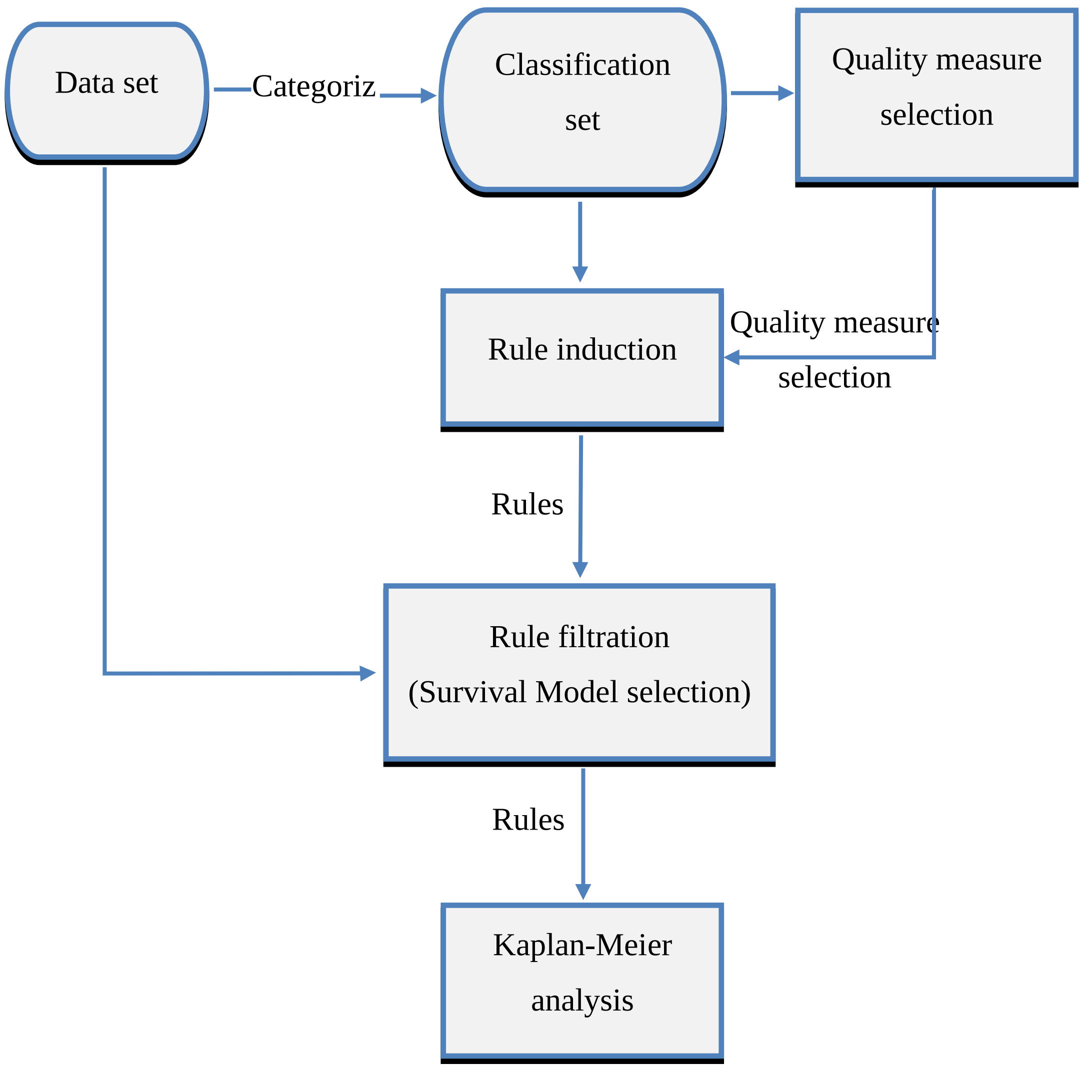


Figure 2.12: The workflow of decision-based analysis of proposed methodology

We have expanded the algorithm of rule induction via possibility of considering the preferences of user, which might consider one of the forms bellow:

I. The user awards the expert rules form of:

(a) the algorithm proceeds their evaluation upon the entire training dataset or in the mode of cross-validation; there are not added any initial conditions for the rules, or generated any novel rules,

(b) the algorithm will redefines the given rules through the user (that means it attempts to add initial conditions for each one of them, this is to maximize a value of the specified quality scale as much as possible) thereafter, it proceeds their evaluation upon the entire training dataset or in the mode of cross-validation,

(c) the algorithm usually specifies all the rules which present by the users, then generates extra rules to the uncovered examples.

II. The user then offers the form of initial conditions that should appear within one rule at least of every decision class.

III. The user offers the attributes that should appear within one rule at least of every decision class.

The procedure of rule induction algorithm illustrates in steps below:

Rule Forming Procedure

***STEP* 1**

Initialize the *RIALST* (forms the primary rules without the terms of the *HAE* classification)

Keep this rule within *RIALST*

Copy it to *best_rules*

***STEP* 2**

WHILE *best_rules* do not proportionate ***THEN*** **DO**

Initialize the *T_ RIALST* (empty)

FOR every rule within *RIALST* **DO**

rules_to_specialize := rule deduced from *RIALST*

***STEP* 3**

CE := an instance misclassified from *rules_to_specialize*, it closest to the *HAE*

IF all values attribute of *HAE* and *CE* are corresponding **THEN**

Select new *CE* and ignore current *CE*

FOR *j* := 1 to *j* := *n* **DO**

***STEP* 4**

IF $V^i_{CE} \neq V^i_{HAE}$ **HEN**

new_rules := rules_to_specialize

IF attribute continual **THEN**

IF $V^i_{CE} > V^i_{HAE}$ **THEN**

Append new condition [$V^i_{CE} > A^i$] for the *new_rules*

ELSE

Append new condition [$V^i_{CE} < A^i$] for the *new_rules*

ELSE

Append new condition [$V^i_{CE} = A^i$] for the *new_rules*

***STEP* 5**

IF the *new_rules* are proportionate **AND** *new_rules* are covering further non covered instances than *best_rules* **THEN**

Replace the *best_rules* with the *new_rules*

IF the *new_rules* do not proportionate **THEN**

IF rules number in *RIALST_size > T_ RIALST* **THEN**

Save *new_rules* in *T_ RIALST*

ELSE

IF *new_rules* of *H* measurement upper than *H* measurement for other rules within *T_ RIALST* ***THEN***

Replace lowest *H* rule measurement within *T_ RIALST* with the *new_rules*

END FOR

END FOR

<u>STEP</u> 6

Copy the *T_ RIALST* on the *RIALST*

END WHILE

IF the *best_rules* includes the constraints of continuous attribute ***THEN***

Constrain such covering for training instances

Add the *best_rules* into collection of rules

When:

- HAE: Hearing Aid Example;
- CE: The instance which near the HAE and does not belong to the target class;
- *RIALST*: The rules list to be specialized within Rule Induction Algorithm;
- *T_ RIALST:* The temporary rules list;
- *rialst _size: The* Maximum rules number within *RIALST;*
- *T_ RIALST* (Determined by user).

3) Naïve Bayes (NB)

Simple as well fast expectation classifier which depend on the hypothesis of Bayes beside the feature independency presumption [151].

Input:

- Data;
- Preprocessing technique(s).

Output:

- Learner of Naïve Bayes;
- Trained model.

Naive Bayes based on learns a model of Naive Bayesian using main dataset. NB works with classification missions only [152].

Figure 2.13 shows the setting of Naive Bayes widget for our model.

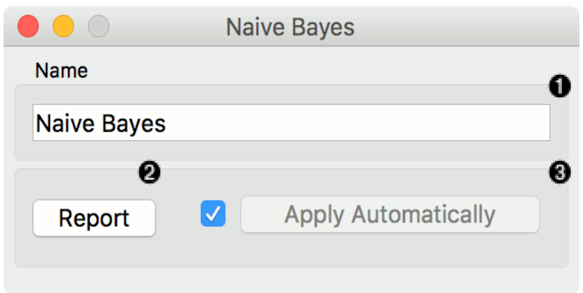


Figure 2.13: The setting of Naïve Bayes widget for our model

While:

This widget contains two options:

1. The "Name": can be assigned any name for the learner which will appear under it within other widgets. We used default name: "Naive Bayes".
2. The "Report" producing.
3. To implement changes, should click “Apply”. If choose "Apply Automatically", then the changes will confirm automatically.

Figure 2.14 shows the ERD of Naïve Bayes Decision Tree Algorithm (DTA):

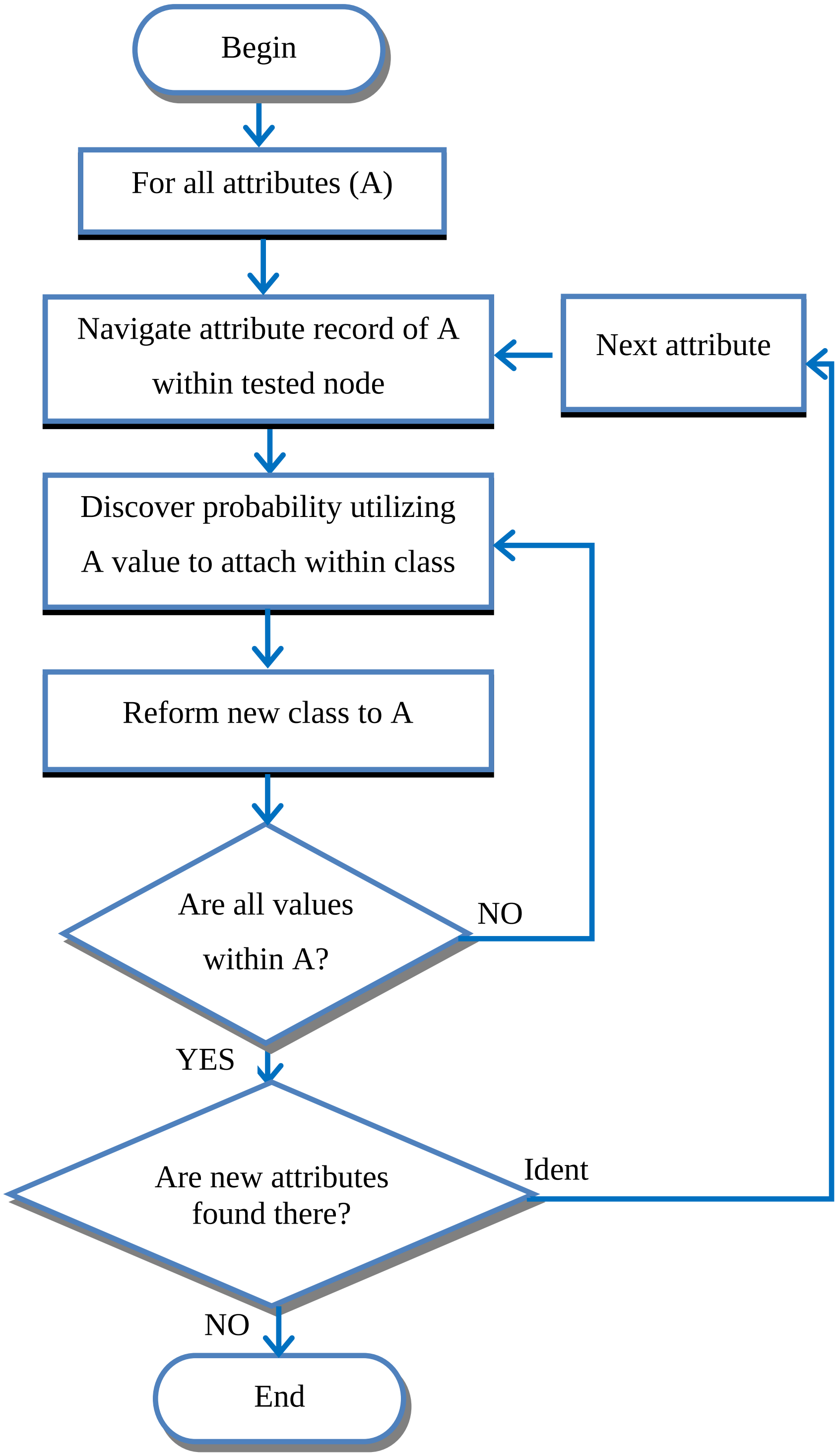


Figure 2.14: The ERD of Naïve Bayes DTA [55]

4) Logistic Regression (LR)

A classification algorithm of LR with regularization of LASSO (L1) otherwise ridge (L2) [153].

Input:

- Data;
- Preprocessing technique(s).

Output:

- Learner;.
- Trained model;
- Coefficients: Coefficients of LR.

LR learns a LR model from any data. It works only with classification tasks [153].

Figure 2.15 shows the setting of LR widget for our model.

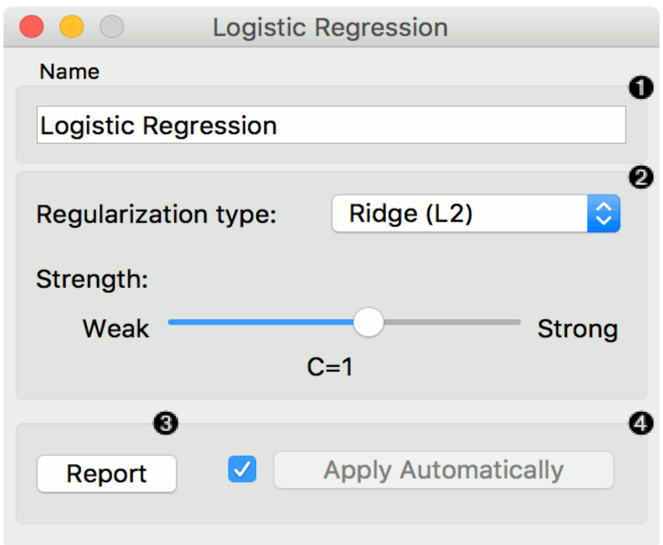


Figure 2.15: The setting of Logistic Regression widget for our model

While:

1. The "Name": can be assigned any name for the learner which will appear under it within other widgets. We used default name: “Logistic Regression”.
2. The "Regularization type": (L2, L1);

- The "Strength": It ranges from "Weak" to "Strong" of cost (C=1 is default).

3. The “Report” producing.
4. Mark "Apply" for confirm modifications. When choose "Apply Automatically", then the modifications confirmed automatically.

5) Random Forest (RF)

Predict using the decision trees ensemble.

Input:

- Data;
- Preprocessing technique(s).

Output:

- Learner of RF.
- Trained model.

RF is group learning technique utilized with regression as well classification missions. RF constructs the decision trees set. Each tree can be developed from one preliminary sample every time from training dataset. When individual trees developed, A random subset is extracted from attributes (that is why the term called “Random”), from which to determine the preferable attribute of split. The conclusive model basically based on a majority vote of developed trees individually within forest [154].

RF works with both regression and classification tasks.

Figure 2.16 shows the setting of RF widget for our model.

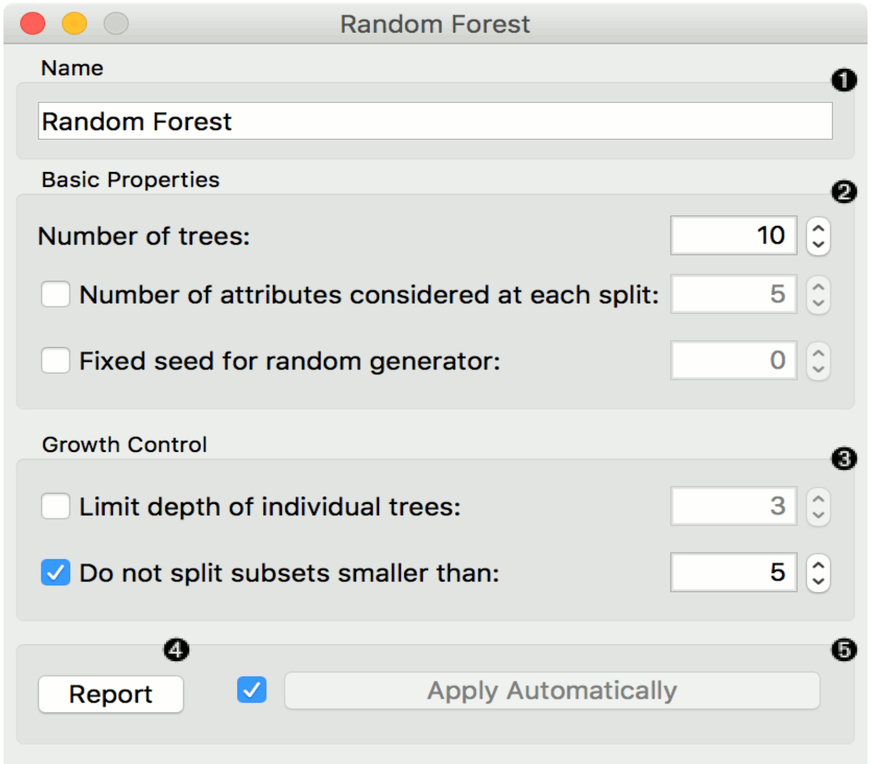


Figure 2.16: The setting of Random Forest widget for our model

While:

1. The “Name”: can be assigned any name for the learner which will appear under it within other widgets. We used default name: “Random Forest”.

2. The "Basic Properties": Assign the number of DTs that will contained within "Forest", as well the number of attributes exist that will randomly construct to considered in every node. When last one does not determine (the choice "*Number of attributes*" preserved without checking). The user capable to select "*Fixed seed for random generator*" to repair the seeds of tree generation, which enables results replicability [154].

3. Original proposal usually is for growing trees but without previously pruning, however since previously pruning often doings faster and quite well, then the user capable to determine the baths of trees that want to constructed (mark “Limit depth of individual trees” for this choice). The last choice of previously pruning is to determine as small subset as possible (mark “Do not split subsets smaller than” for this choice).

4. The “Report” producing.

5. Select "Apply" to transfer the changes for other widgets. Alternatively, can choose "Apply Automatically", so the changes will confirm automatically.

6) AdaBoost

A meta-algorithm or ensemble algorithm which gathers weak learners as well as adapts with ‘hardness’ for each sample of training.

Input:

- Data;
- Preprocessing technique(s).

Output:

- Learner of AdaBoost.
- Model trained.

AdaBoost (is term of “Adaptive Boosting”) is an algorithm of ML. The formula of AdaBoost was discovered by Robert Schapire and Yoav Freund. It may utilized for more supporting to another learning algorithms in the purpose of boosting their effectiveness via adjusting most of weak learners [155].

AdaBoost applies with regression as well as classification.

Figure 2.17 shows the setting of AdaBoost widget for our model.

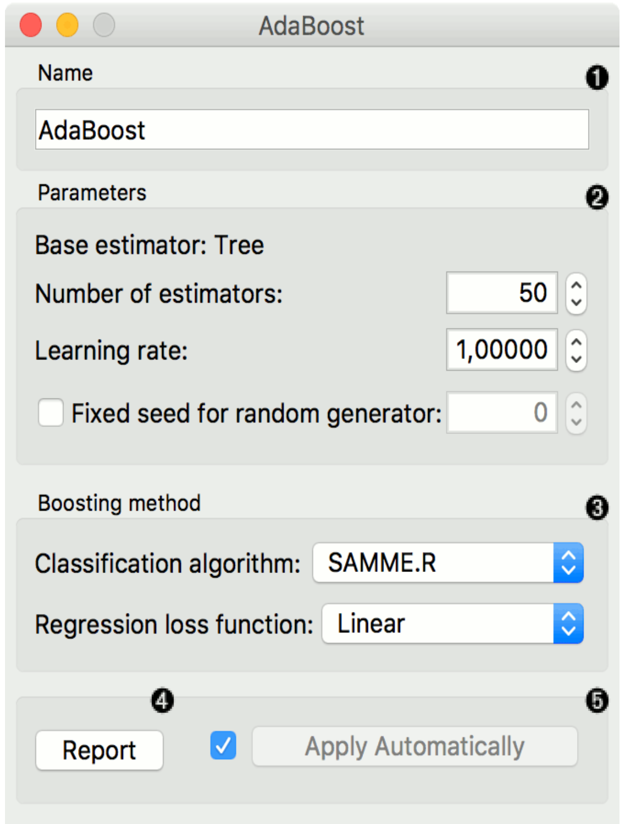


Figure 2.17: The setting of AdaBoost widget for our model

While:

1. The "Name": can be assigned any name for the learner which will appear under it within other widgets. We used default name: "AdaBoost".

2. The "Parameters" option setting: "*Base estimator*" usually "Tree" as well the user capable to update the options below:

   - "*Number of estimators*".

   - "*Learning rate*": Locates the limits of how recently information gained will exceed an ancient information (for clarify, operator= 0 or 1: 0 = nothing will learn and 1 = considers the recent information only).

   - "*Fixed seed for random generator*": mark this field to enable the results reproducing.

3. The "Boosting method":

- The *"Classification algorithm"* (classification on the input): there are two options of updates the weights of base estimator either "SAMME" (with results of classification) or "SAMME.R" (with valuations of probability).

- The *"Regression loss function"* (regression upon inputs): Like Square (), Exponential (), Linear ().

4. The "Report" producing.

5. Choose "Apply" when complete settings. This will place a new learner within output, further if given a new training example, then a new model will be constructed as well output it. If choose "Apply Automatically", then the changes will confirm automatically.

7) Neural Network

Multilayer perceptual algorithm with back propagation.

Input:

- Data;
- Preprocessing technique(s).

Output:

- Learner: learning algorithm of multilayer perceptual;
- Trained model.

NN widget utilizes the algorithm of "sklearn's Multilayer Perceptual" (MLP) which can be learned linear and non-linear patterns [156].

Figure 2.18 shows the setting of NN widget for our model.

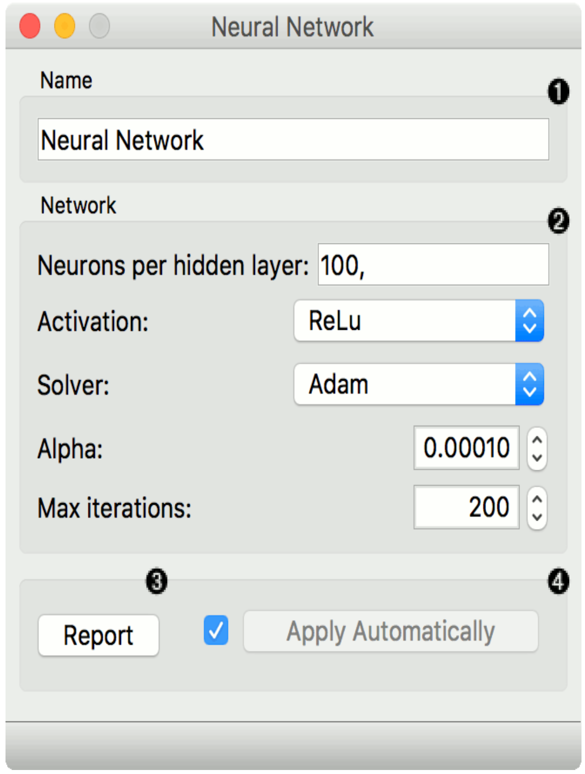


Figure 2.18: The setting of NN widget for our model

While:

1. The "Name": can be assigned any name for the learner which will appear under it within other widgets. We used default name: "Neural Network".
2. The setting of "Network" parameters listed below:

- "*Neurons per hidden layer*": An element i^{th} which refers to neurons within hidden i^{th} layer, for example the layer of weights [4x4x3] within input, have total of 4*4*3 = 48 weights (besides one extra for bias parameter).

- The "*Activation*" function of a hidden layers:

    - "*Identity*": Function "no-op" activation, beneficial for executing linear bottleneck,
    - "*Logistic*": Function of logistic sigmoid,
    - "*tanh*": Function of hyperbolical tan,
    - "*ReLu*": Rectified function of linear unit.

- The “*Solver*” of weight optimization:
  - “*L-BFGS-B*”: the optimizer within group of quasi-Newton techniques,
  - “*SGD*”: the principle of random probability gradient,
  - “*Adam*”: the optimizer of random probability gradient based.
- “*Alpha*”: L2 sanction (regularization terms) parameters.
- “*Max iterations*”: the iterations maximum number.
- All further parameters keep setting for sklearn’s default.

3. The “Report” producing.
4. Select "Apply" to transfer the changes for other widgets. Alternatively, can choose "Apply Automatically", so the changes will confirm automatically.

The simplified procedure for training NN is within steps below:

1. Input data propagated via network till it attains the output layer. Such forward operation generates outputs prediction.
2. Output which prophesied will abstracted from an existing output then calculate the incorrect value of the network.
3. The NN then utilizes SL The NN then utilizes SL with back propagation to train such network within most cases.

   “Back propagation is learning technique to adjust weights. It initiates from weights among output layers as well as last hidden layers then resume working stages backwards via network” [156].
4. When back propagation accomplished, the forward operation will iterate one more time, such conduct will be repeated till a mistake among actual outputs and predicted are decreased as minimum as possible.

The actual algorithm of NN that our model be based on is listed below:

1. Initialize the network weights (usually randomly),

2. iterate,

* for every instance i within training dataset Do

NO := neural-netw-outputs(network, i);

forward passing;

L_i := learner outputs for i;

Calculate the error generated(E := L_i - NO) within output unit;

Calculate the delta_w_i of weights, start with the hidden layers till output layers;

backward passing;

Calculate the delta_w_i of weights, start with the input layers till hidden layers;

continued backward passing;

Redefine network weights

* stop iteration

3. Till each instance correctly categorized or satisfied the discontinuation condition,

4. turn_back(network_step 2).

- **Advantages of NN:**

1. "High Accuracy": NNs capable for estimate mappings of complicated nonlinear [97].
2. "Noise Tolerance": NNs are extremely flexible regard to the noisy, incomplete, and, missing data [90].
3. "Independency": NNs do not create any of priori assumptions regarding to the data distribution, or any form of interactivities between factors [92].

4. “Maintenance ease”: NNs capable to update using fresh data, which making them beneficial to dynamic environments [99].

5. “NNs can be executed within parallel hardware 6”: If any of NN elements fail, then it capable resume work normally because of its parallel structure [95].

➢ **Design problems:**

1. No ascertain method for determining necessary neurons best favorable number to solve any problem [91].

2. Hard to choose training set that completely illustrates issue that should resolved [157].

2.3.5 Unsupervised Model

Unsupervised DM is an approach of bottom-up that do not offer any prior assumptions as well aims at finding out the relationships within the data. That is mean, the data can speak by themselves; and there is no difference between targets and attributes. From this principle, unsupervised DM is an approach of descriptive technique [147].

Let clarify the Orange benefits through data analysis by Python shell and unsupervised techniques that we have adopted in our model:

```
>>> import Orange modeler

>>> data := Orange.data.Table("Audiology HA")

>>> len(data)

210 {number of dataset instances}

>>> ml := Orange.classification.ml.MLLearner()

>>> mds := Orange.classification.mds.MDSLearner()

>>> stack := Orange.ensemble.stacking.StackedClassificationLearner([ml,mds])

>>> res := Orange.evaluating.testing.cross_validation([ml, mds, stack], data)
```

Now we illustrate these techniques in detail:

1) Manifold Learning (ML)

Reduction of nonlinear dimensionality.

Input:

- Data: input the dataset.

Output:

- Transformed Data: dataset with minimum coordinates.

ML technique works to find the non-linear divergent in space of the higher dimensions. Then widget will produce novel coordinates match the multidimensional area. Later, this dataset may displayed with any of visualization widgets like Scatter Plot [158].

Figure 2.19 shows the setting of ML widget for our model.

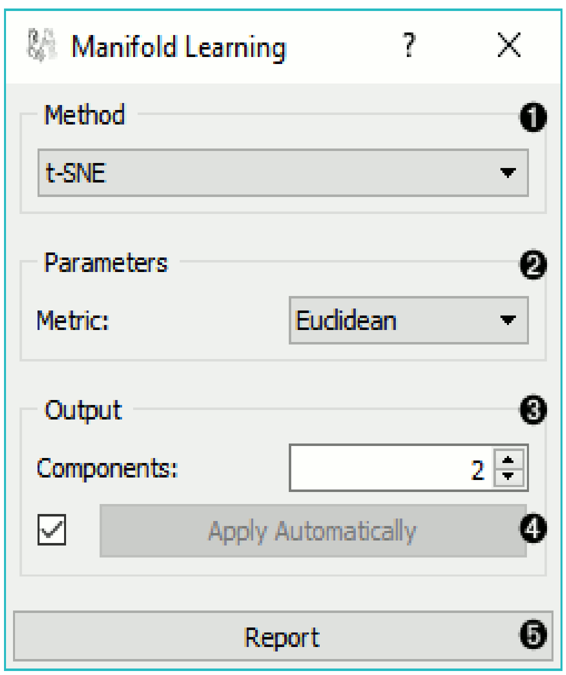


Figure 2.19: The setting of ML widget for our model

While:

1. The “Method” of ML:

- “t-SNE”
- “Isomap”
- “MDS”
- “Spectral Embedding”
- “Linear Embedding Locally”

2. Set method “Parameters”:

- “t-SNE” (measures of distance):
  - “Euclidean distance”
  - “Chebyshev”
  - “Cosine”
  - “Manhattan”
  - “Mahalanobis”
  - “Jaccard”
- “Isomap”:
  - the neighbors number
- “MDS” (initialization and iterations):
  - initialization: the method for an algorithm initialization (random or PCA)
  - max iterations: the maximum number for optimization interactions
- “Spectral Embedding”:
  - “affinity”:

        - “RFB kernel”
        - “nearest neighbors”
    - Linear Embedding Locally:
        - method:
            - “standard”
            - “local”
            - “hessian eigenmap”
            - “modified”
            - “max iterations”
            - “neighbors’ number”

3. “Output”: the reduced features number (the components).

4. Select "Apply" to transfer the changes for other widgets. Alternatively, can choose "Apply Automatically", so the changes will confirm automatically.

5. The “Report” producing.

ML widget produces diverse embeddings of high-dimensional data, like: t-SNE, Isomap, MDS, Spectral Embedding, and Linear Embedding Locally [158].

2) Multidimensional Scaling (MDS)

MDS works on drop the items onto a fitted area for giving suitable spaces among points [159].

Input:

- Data;
- Distance matrix;
- Data instances subset.

Output:

- Chosen Dataset: select samples from plots
- Dataset: coordinates of MDS

MDS method is to discovers low dimensional (two dimensions in our study) dropping of instances, so it will attempt to convenient the spaces among all instances to the closest fit distance. The ideal fitting normally unattainable to gain because either there are not Euclidean distances, or the data may high-dimensional [159].

Within input, all widgets need either a distances matrix or a dataset. Further, there are an ability of adjust the points color, output them on select, mark them, and alter their shape [116].

The algorithm moves all points iteratively around in the type of physical pattern simulation: when there are two instances nearer (or farther away) each other, then will be a force that pushes them away (or closer) [120].

Figure 2.20 shows the setting of MDS widget for our model.

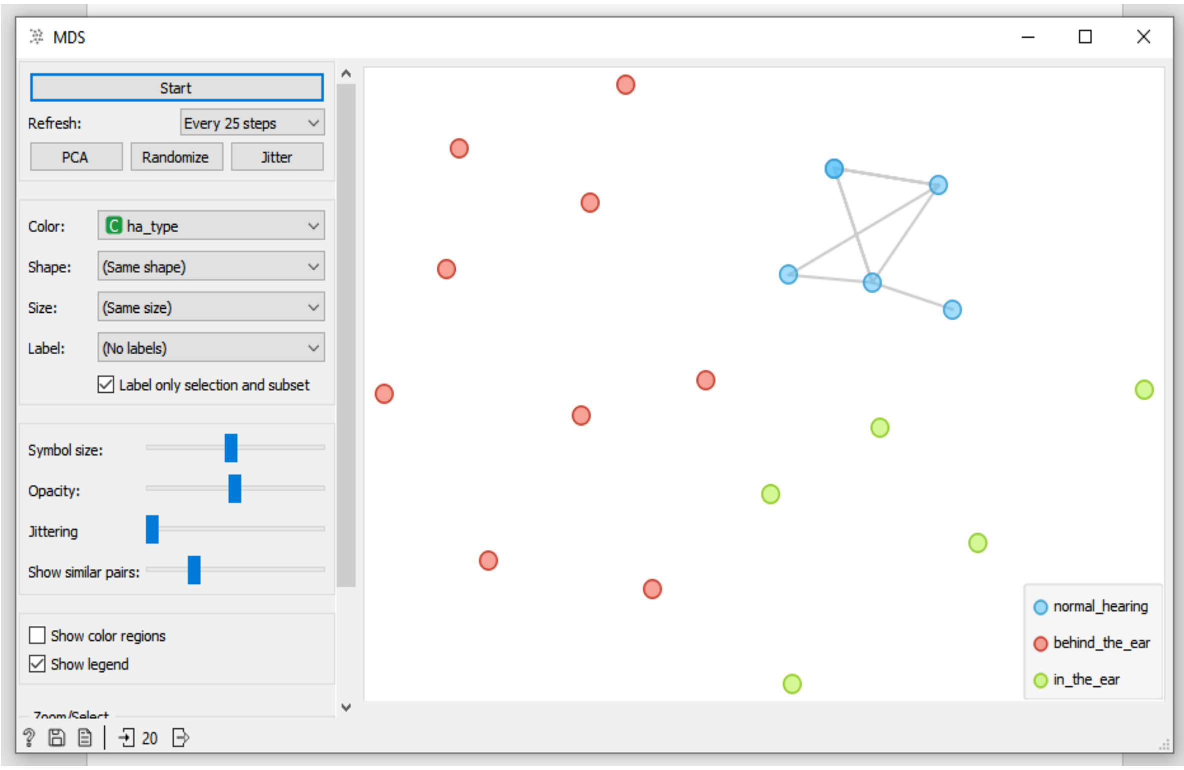


Figure 2.20: The setting of MDS widget for our model

While:

1. MDS widget reconfigures the delineation throughout optimizing. The automatic optimizing will be implemented in the initial or later via pressing “Start” option.

   - “Max iterations”: An optimizing will stop if maximum iterations number reached or when projection changes minimally only at a last iteration.
   - “Initialization”: PCA places first points on the line with major axes coordinates.
   - “Refresh”: Setting the number of times that needs for conception update. Such refresh setting may either at ("Every iteration"), ("Every #step(s)"), or ("None"). Tuning lower interval update guides a vitality more attractive while may go slower when the points are too many.

2. Determines in which way can visualized the points. its available when distances visualized between rows only (within Distances widget):

   - “Color”: The points Color by attribute (colored for the discrete, gray for the continuous).
   - “Shape”: Points are Shaped depending on feature (for separated points only).
   - “Size”: Setting the points volume (select attribute, continuous attribute "Stress", or Same size).
   - “Label”: Can use the "Discrete attributes" as label.
   - “Symbol size”: Adjust the dots size.
   - “Symbol opacity”: Setting the dots directness range.
   - “Show similar pairs”: Setting the network lines intensity.
   - “Jitter”: Setting the tension for prohibit the points interlacing.

3. Modify the graphs by "Zoom and Select" option and can be do the zooming via magnifying glass as well as via in-out scrolling. Besides, the graph can move around by hand. Finally, can select the data instances using arrow.
4. Choose the output required:
   - Original attributes (the inputs datasets).
   - Regulates (the MDS coordinates).
   - Regulates as the regular attributes (the input dataset with MDS coordinates).
   - Regulates as the meta features (the input dataset with MDS coordinates).
5. Select "Send" to confirm and transfer the changes on the instances. Alternatively, can choose "Send Automatically", so the changes will confirm automatically.
6. “Save Image”: This choice authorizes for saving the generated image in the form of .png, .svg, or gnbg.
7. The “Report” producing.

MDS graph implements numerous of visualizations widget functions. From many perspectives, it is identical to the Scatter Plot formula: "<../visualize/scatterplot> widget" [159].

3. RESULTS AND COMPARISON

3.1 DATASET SAMPLES

For all algorithms we used the same sample of dataset for test and train, as shown in Figure 3.1:

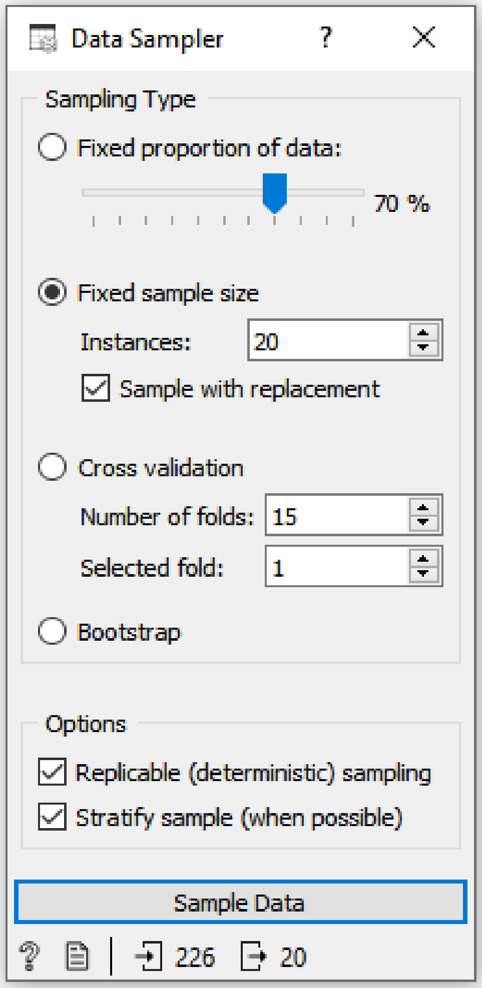


Figure 3.1: The details of dataset sample used for test and train

For all algorithms we used the same condition for select rows of test and train dataset, as shown in Figure 3.2:

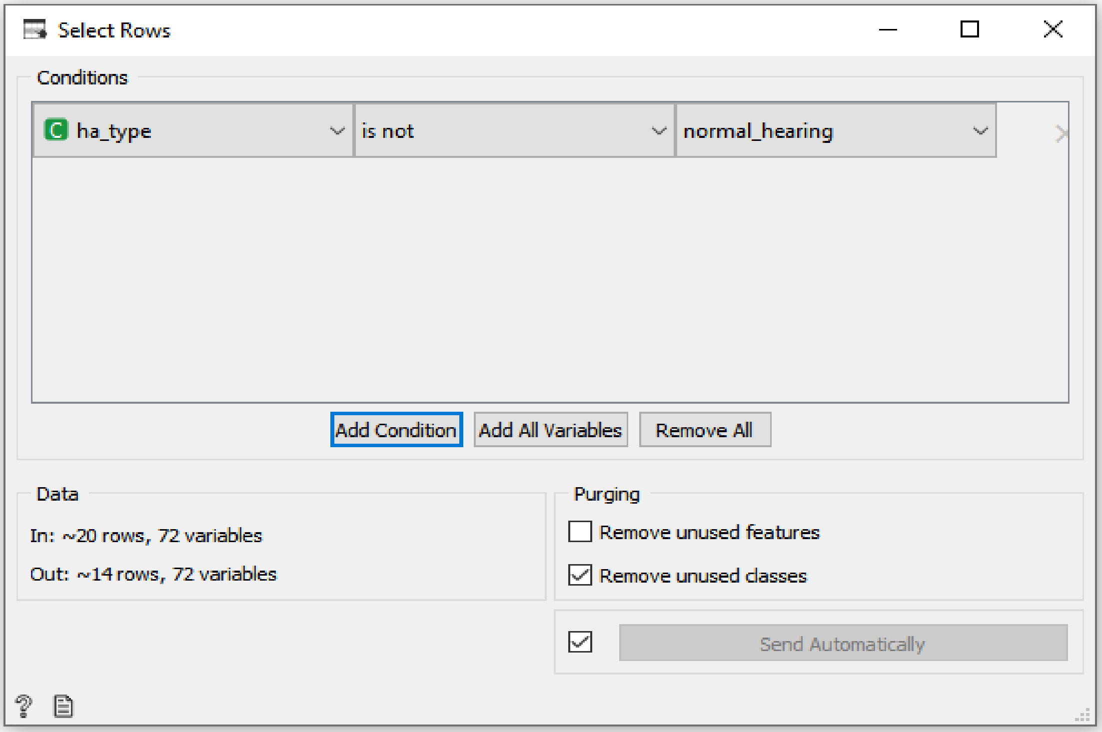


Figure 3.2: The condition details of select rows for test and train dataset

3.1.1 Supervised Results and Comparison

Figure 3.3 displays the structure of our main model with supervised DM Learning Techniques which applied within our model using Orange Canvas application for building the model and Python for coding.

Figure 3.3: The main model design for all supervised techniques used within our model

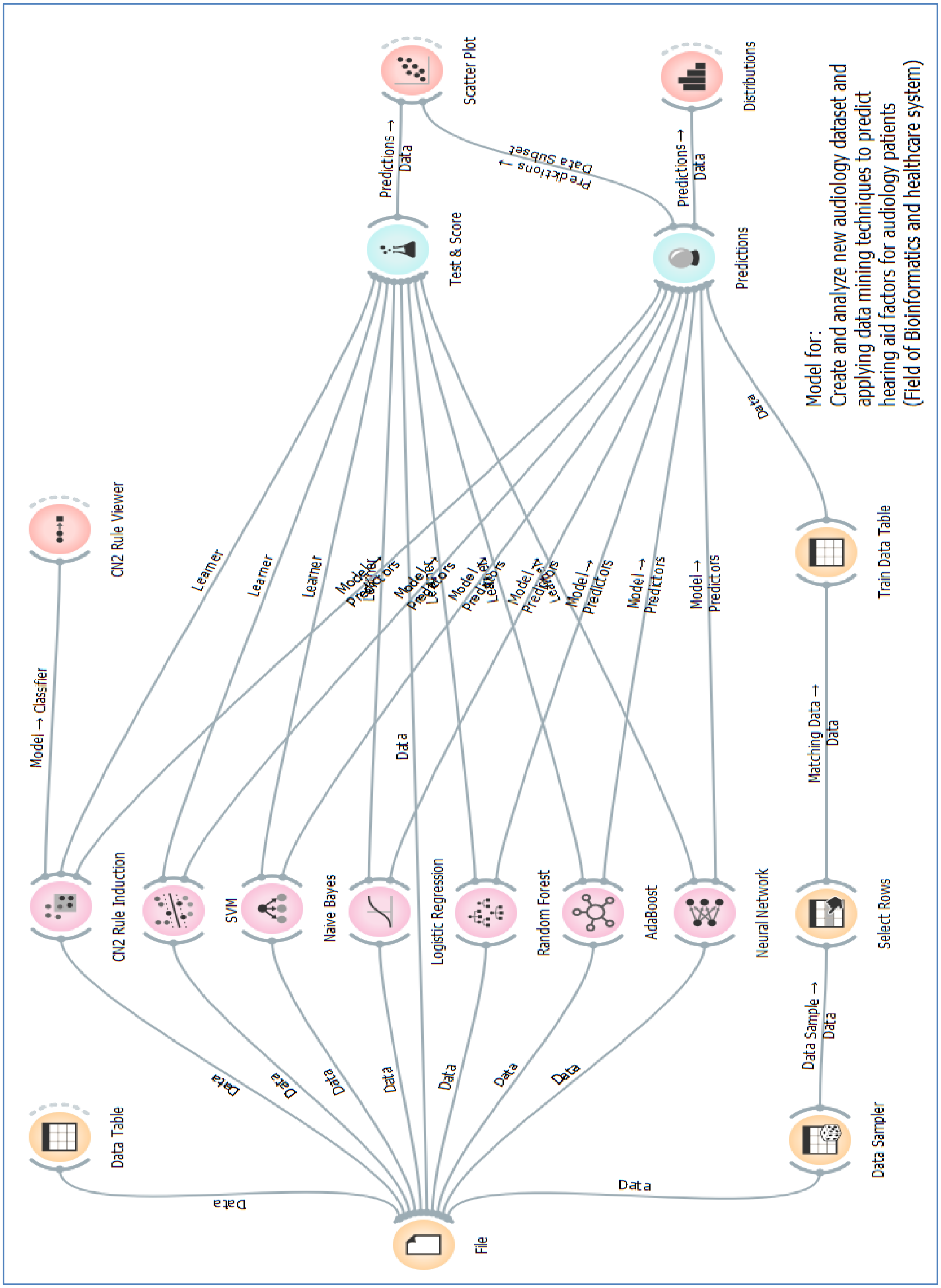
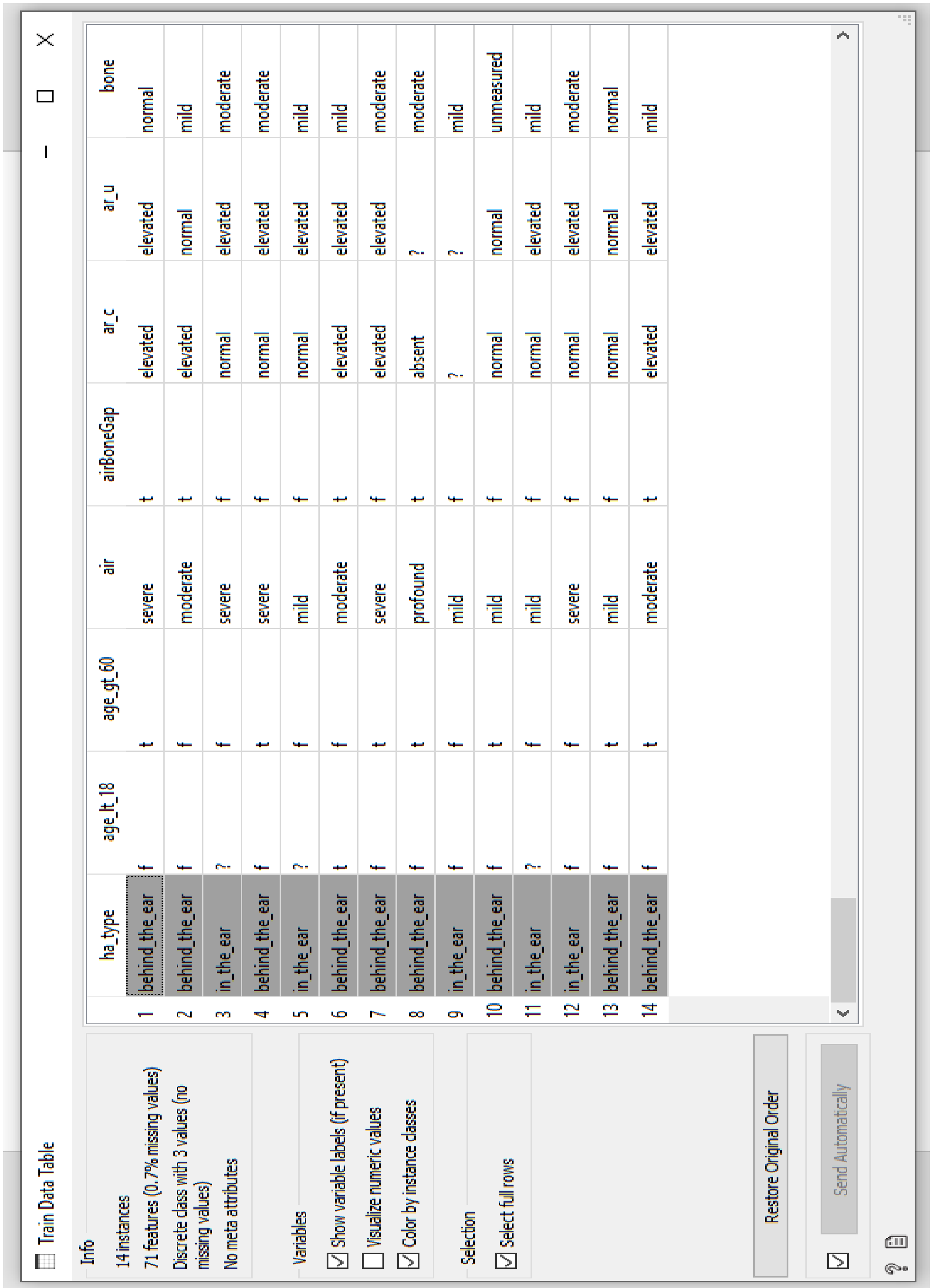


Figure 3.4 shows the train data table choosing for all supervised techniques that used within our model:

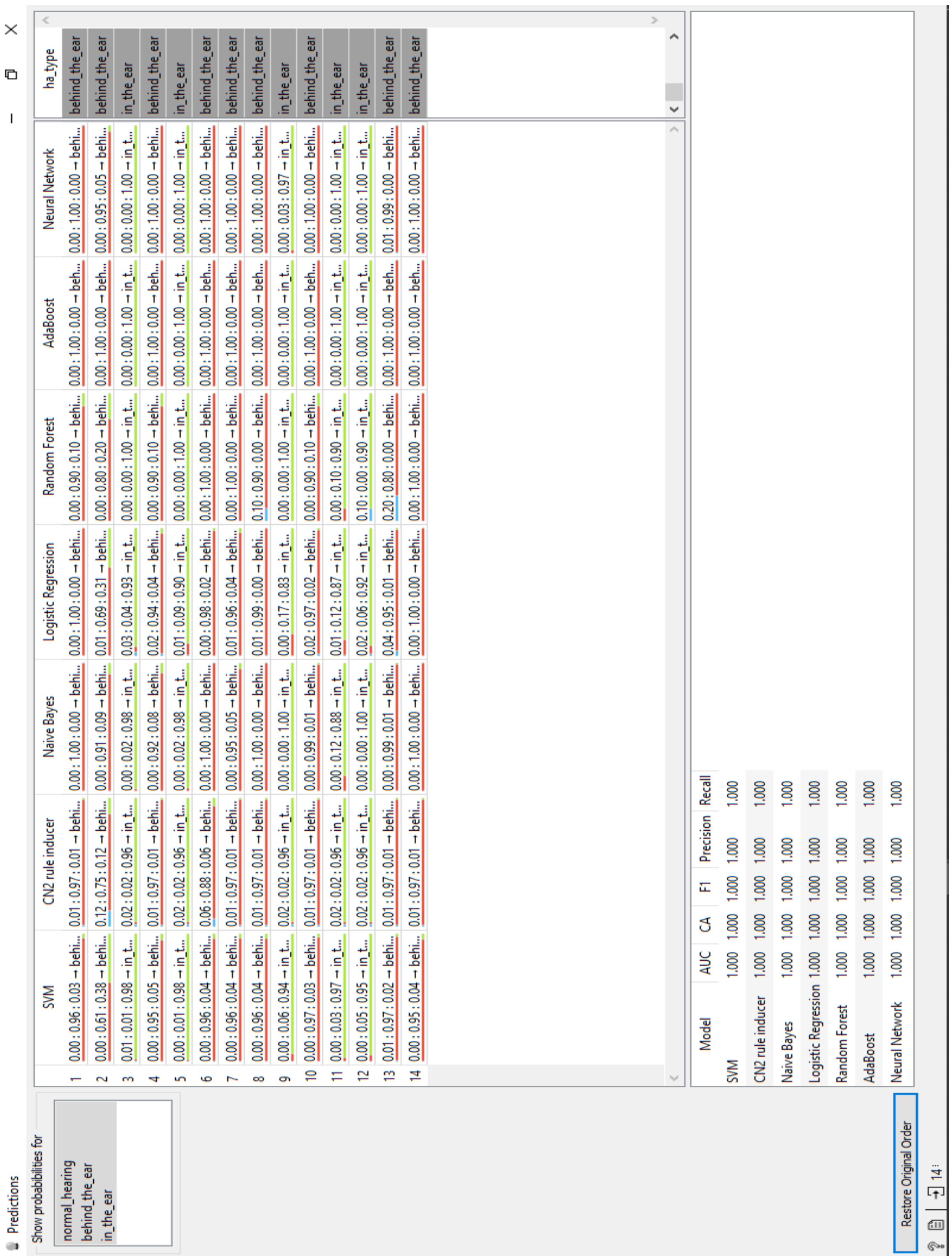
Figure 3.4: The train data table choosing for all supervised techniques used within our model



	ha_type	age_lt_18	age_gt_60	air	airBoneGap	ar_c	ar_u	bone
1	behind_the_ear	f	t	severe	t	elevated	elevated	normal
2	behind_the_ear	f	f	moderate	t	elevated	normal	mild
3	in_the_ear	?	f	severe	f	normal	elevated	moderate
4	behind_the_ear	f	t	severe	f	normal	elevated	moderate
5	in_the_ear	?	f	mild	f	normal	elevated	mild
6	behind_the_ear	t	f	moderate	t	elevated	elevated	mild
7	behind_the_ear	f	t	severe	f	elevated	elevated	moderate
8	behind_the_ear	f	t	profound	t	absent	?	moderate
9	in_the_ear	f	f	mild	f	?	?	mild
10	behind_the_ear	f	t	mild	f	normal	normal	unmeasured
11	in_the_ear	?	f	mild	f	normal	elevated	mild
12	in_the_ear	f	f	severe	f	normal	elevated	moderate
13	behind_the_ear	f	t	mild	f	normal	normal	normal
14	behind_the_ear	f	t	moderate	t	elevated	elevated	mild

Figure 3.5 shows the Prediction results table for all supervised techniques that used within our model:

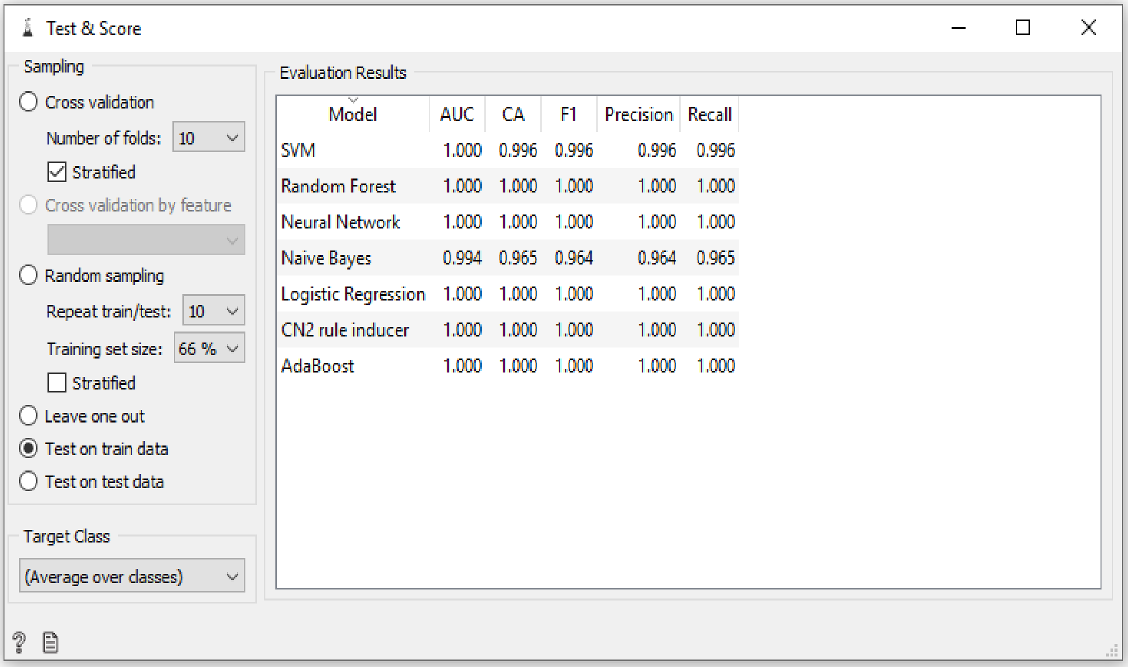
Figure 3.5: The Prediction results table for all supervised techniques used within our model



	SVM	CN2 rule inducer	Naive Bayes	Logistic Regression	Random Forest	AdaBoost	Neural Network	ha_type
1	0.00 : 0.96 : 0.03 → behi...	0.01 : 0.97 : 0.01 → behi...	0.00 : 1.00 : 0.00 → behi...	0.00 : 1.00 : 0.00 → behi...	0.00 : 0.90 : 0.10 → behi...	0.00 : 1.00 : 0.00 → beh...	0.00 : 1.00 : 0.00 → behi...	behind_the_ear
2	0.00 : 0.61 : 0.38 → behi...	0.12 : 0.75 : 0.12 → behi...	0.00 : 0.91 : 0.09 → behi...	0.01 : 0.69 : 0.31 → behi...	0.00 : 0.80 : 0.20 → behi...	0.00 : 1.00 : 0.00 → beh...	0.00 : 0.95 : 0.05 → behi...	behind_the_ear
3	0.01 : 0.01 : 0.98 → in_t...	0.02 : 0.02 : 0.96 → in_t...	0.00 : 0.02 : 0.98 → in_t...	0.03 : 0.04 : 0.93 → in_t...	0.00 : 0.00 : 1.00 → in_t...	0.00 : 0.00 : 1.00 → in_t...	0.00 : 0.00 : 1.00 → in_t...	in_the_ear
4	0.00 : 0.95 : 0.05 → behi...	0.01 : 0.97 : 0.01 → behi...	0.00 : 0.92 : 0.08 → behi...	0.02 : 0.94 : 0.04 → behi...	0.00 : 0.90 : 0.10 → behi...	0.00 : 1.00 : 0.00 → beh...	0.00 : 1.00 : 0.00 → behi...	behind_the_ear
5	0.00 : 0.01 : 0.98 → in_t...	0.02 : 0.02 : 0.96 → in_t...	0.00 : 0.02 : 0.98 → in_t...	0.01 : 0.09 : 0.90 → in_t...	0.00 : 0.00 : 1.00 → in_t...	0.00 : 0.00 : 1.00 → in_t...	0.00 : 0.00 : 1.00 → in_t...	in_the_ear
6	0.00 : 0.96 : 0.04 → behi...	0.06 : 0.88 : 0.06 → behi...	0.00 : 1.00 : 0.00 → behi...	0.00 : 0.98 : 0.02 → behi...	0.00 : 1.00 : 0.00 → behi...	0.00 : 1.00 : 0.00 → beh...	0.00 : 1.00 : 0.00 → behi...	behind_the_ear
7	0.00 : 0.96 : 0.04 → behi...	0.01 : 0.97 : 0.01 → behi...	0.00 : 0.95 : 0.05 → behi...	0.01 : 0.96 : 0.04 → behi...	0.00 : 1.00 : 0.00 → behi...	0.00 : 1.00 : 0.00 → beh...	0.00 : 1.00 : 0.00 → behi...	behind_the_ear
8	0.00 : 0.96 : 0.04 → behi...	0.01 : 0.97 : 0.01 → behi...	0.00 : 1.00 : 0.00 → behi...	0.01 : 0.99 : 0.00 → behi...	0.10 : 0.90 : 0.00 → behi...	0.00 : 1.00 : 0.00 → beh...	0.00 : 1.00 : 0.00 → behi...	behind_the_ear
9	0.00 : 0.06 : 0.94 → in_t...	0.02 : 0.02 : 0.96 → in_t...	0.00 : 0.00 : 1.00 → in_t...	0.00 : 0.17 : 0.83 → in_t...	0.00 : 0.00 : 1.00 → in_t...	0.00 : 0.00 : 1.00 → in_t...	0.00 : 0.03 : 0.97 → in_t...	in_the_ear
10	0.00 : 0.97 : 0.03 → behi...	0.01 : 0.97 : 0.01 → behi...	0.00 : 0.99 : 0.01 → behi...	0.02 : 0.97 : 0.02 → behi...	0.00 : 0.90 : 0.10 → behi...	0.00 : 1.00 : 0.00 → beh...	0.00 : 1.00 : 0.00 → behi...	behind_the_ear
11	0.00 : 0.03 : 0.97 → in_t...	0.02 : 0.02 : 0.96 → in_t...	0.00 : 0.12 : 0.88 → in_t...	0.01 : 0.12 : 0.87 → in_t...	0.00 : 0.10 : 0.90 → in_t...	0.00 : 0.00 : 1.00 → in_t...	0.00 : 0.00 : 1.00 → in_t...	in_the_ear
12	0.00 : 0.05 : 0.95 → in_t...	0.02 : 0.02 : 0.96 → in_t...	0.00 : 0.00 : 1.00 → in_t...	0.02 : 0.06 : 0.92 → in_t...	0.10 : 0.00 : 0.90 → in_t...	0.00 : 0.00 : 1.00 → in_t...	0.00 : 0.00 : 1.00 → in_t...	in_the_ear
13	0.01 : 0.97 : 0.02 → behi...	0.01 : 0.97 : 0.01 → behi...	0.00 : 0.99 : 0.01 → behi...	0.04 : 0.95 : 0.01 → behi...	0.20 : 0.80 : 0.00 → behi...	0.00 : 1.00 : 0.00 → beh...	0.01 : 0.99 : 0.00 → behi...	behind_the_ear
14	0.00 : 0.95 : 0.04 → behi...	0.01 : 0.97 : 0.01 → behi...	0.00 : 1.00 : 0.00 → behi...	0.00 : 1.00 : 0.00 → behi...	0.00 : 1.00 : 0.00 → behi...	0.00 : 1.00 : 0.00 → beh...	0.00 : 1.00 : 0.00 → behi...	behind_the_ear

Model	AUC	CA	F1	Precision	Recall
SVM	1.000	1.000	1.000	1.000	1.000
CN2 rule inducer	1.000	1.000	1.000	1.000	1.000
Naive Bayes	1.000	1.000	1.000	1.000	1.000
Logistic Regression	1.000	1.000	1.000	1.000	1.000
Random Forest	1.000	1.000	1.000	1.000	1.000
AdaBoost	1.000	1.000	1.000	1.000	1.000
Neural Network	1.000	1.000	1.000	1.000	1.000

Figure 3.6 shows the details of Test and Score table for all supervised techniques that used within our model:



Model	AUC	CA	F1	Precision	Recall
SVM	1.000	0.996	0.996	0.996	0.996
Random Forest	1.000	1.000	1.000	1.000	1.000
Neural Network	1.000	1.000	1.000	1.000	1.000
Naive Bayes	0.994	0.965	0.964	0.964	0.965
Logistic Regression	1.000	1.000	1.000	1.000	1.000
CN2 rule inducer	1.000	1.000	1.000	1.000	1.000
AdaBoost	1.000	1.000	1.000	1.000	1.000

Figure 3.6: The details of Test and Score table for all supervised techniques used within our model

While:

- One of the classes are consider as positive class (target) as well the other one as negative. This leads to four values:
  - True Positives (TPs), Target class instances that are predicted to be from same class;
  - True Negatives (TNs): Negative class instances that are predicted to be so;
  - False Positives (FPs): Negative class instances, but they are predicted to be from the target class;
  - and False Negatives (FNs): Target classes instances, but they are predicted to be from the negative classes.

Such calculated values (TPs, TNs, FPs, FNs) represent a rule for performance metrics evaluations that we should analyzing. The “Precision” is a dividing result of TPs by sum of FPs and TPs [143]. So:

$$Precision = \frac{TP}{FP + TP} \tag{3.1}$$

The “Recall” is a dividing result of TPs by sum of FNs and TPs:

$$Recall = \frac{TP}{FN + TP} \tag{3.2}$$

“F-Measure” is the "Recall" and "Precision" average weighted. It calculated via dividing the "Recall" and "Precision" multiplication, by the "Recall" and Precision" summation. The values can sort from 1 (the best value) to 0 (the worst value) and vice versa [2]:

$$F - Measure = \frac{(Recall * Precision)}{(Recall + Precision)} \tag{3.3}$$

Within our results table, the numbers are shown rather than graphs. These values are the AUC ("Area Under Curve") which refer to the areas inferred under ROC graph curve. AUC recently doubted as an algorithm comparison poor metric since it adopts the measure of noisy for classification [143].

AC refers to the "Accuracy" and represents to the total correctly values of prophesied classifications as well as it an outcome of divide the sum of both TNs and TPs on total instances analyzed. The higher an accuracy gained, the better an algorithm is:

$$AC = \frac{(TN + TP)}{T} \tag{3.4}$$

The “Specificity” is a division result of TNs by the total sum of FPs and TNs:

$$Specificity = \frac{TN}{(FP + TN)} \tag{3.5}$$

The “Sensitivity” is a division outcome of TPs by total aggregate of FNs and TPs. Both specificity as well as sensitivity calculate predicted instances correctly (specificity for all negatives and sensitivity for all positives) [2].

Figure 3.7 shows the CN2 rule conditions (IF Condition THEN Class) applying in CN2 algorithm for prediction the type of HA:

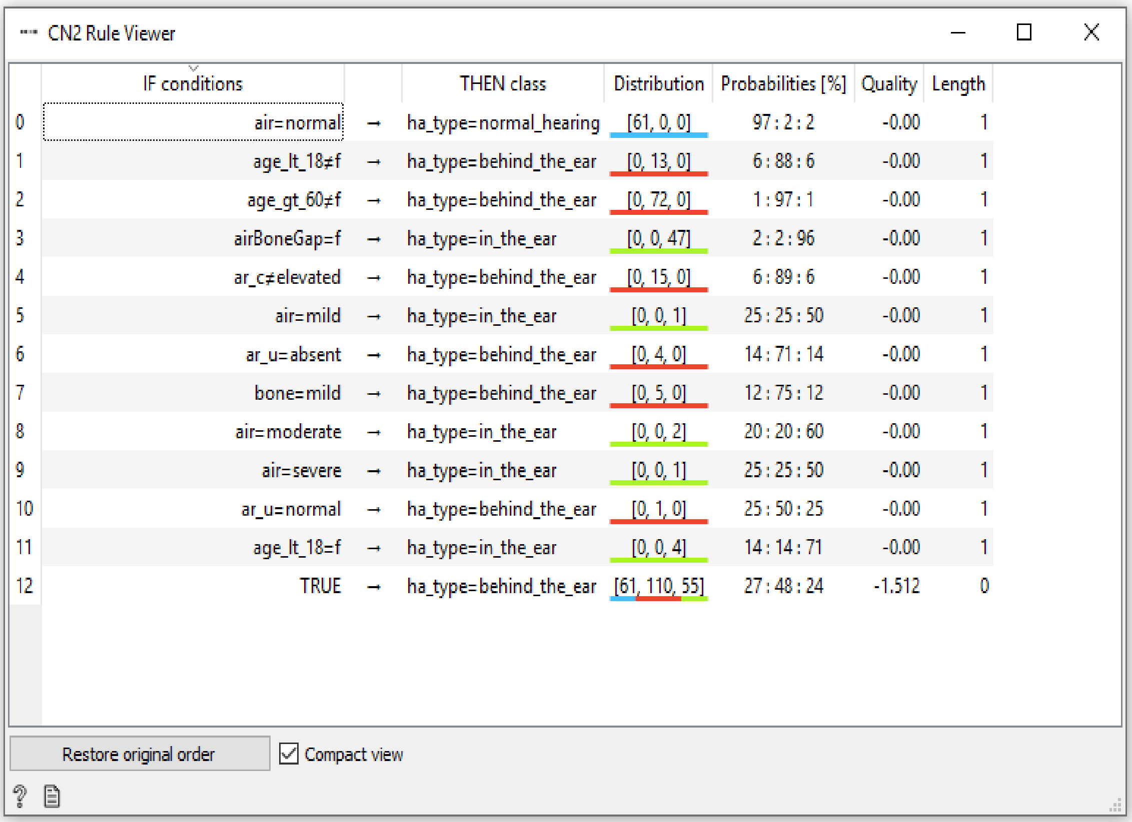


Figure 3.7: The CN2 rule conditions applying in CN2 algorithm

Figure 3.8 shows the Distributions probability of CN2 rule inducer for one class of dataset (behind_the_ear) which refers to one type of HA:

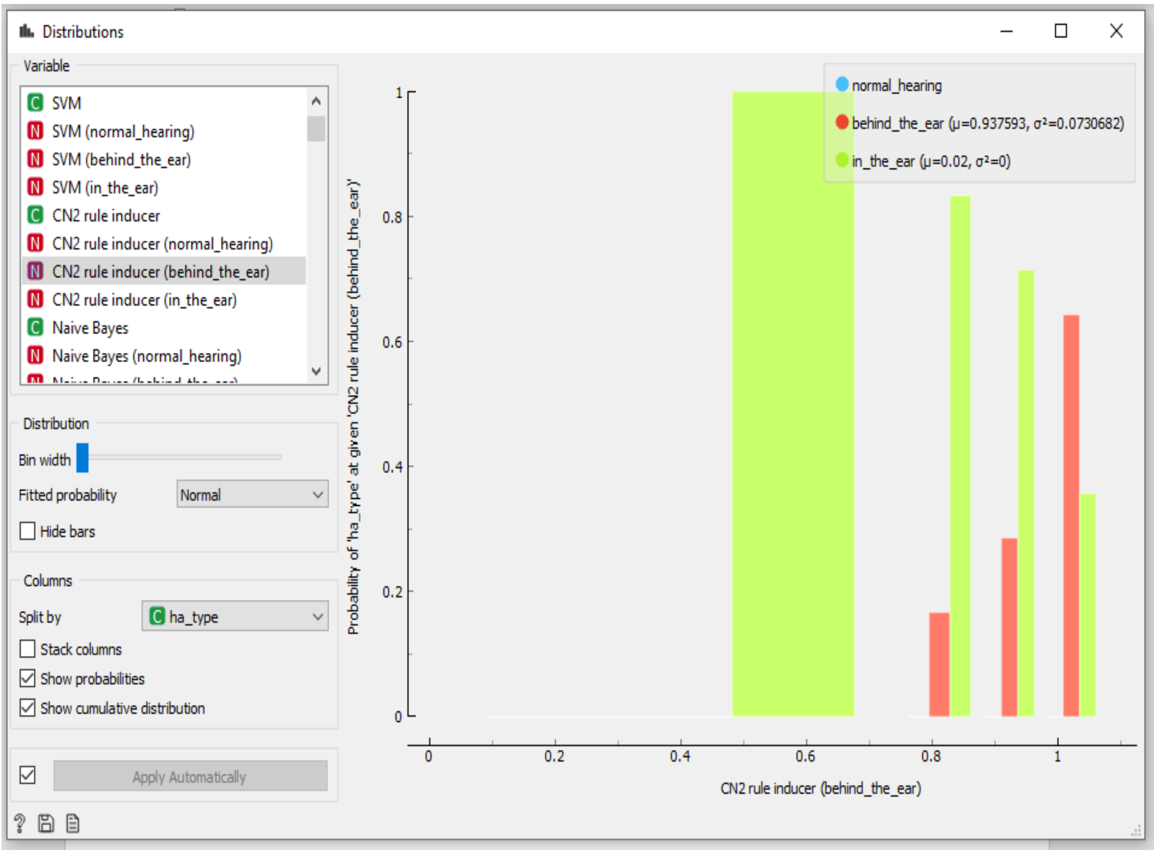


Figure 3.8: The Distributions probability of CN2 rule inducer for (behind_the_ear) class

Figure 3.9 shows the Scatter Plot of CN2 rule inducer for one class of dataset (behind_the_ear) HA as X exes and the second class (in_the_ear) HA as Y exes to predict the type of HA.

Figure 3.9: The Scatter Plot of CN2 rule inducer for the classes BTE Hearing Aid as X exes and ITE Hearing Aid as Y exes

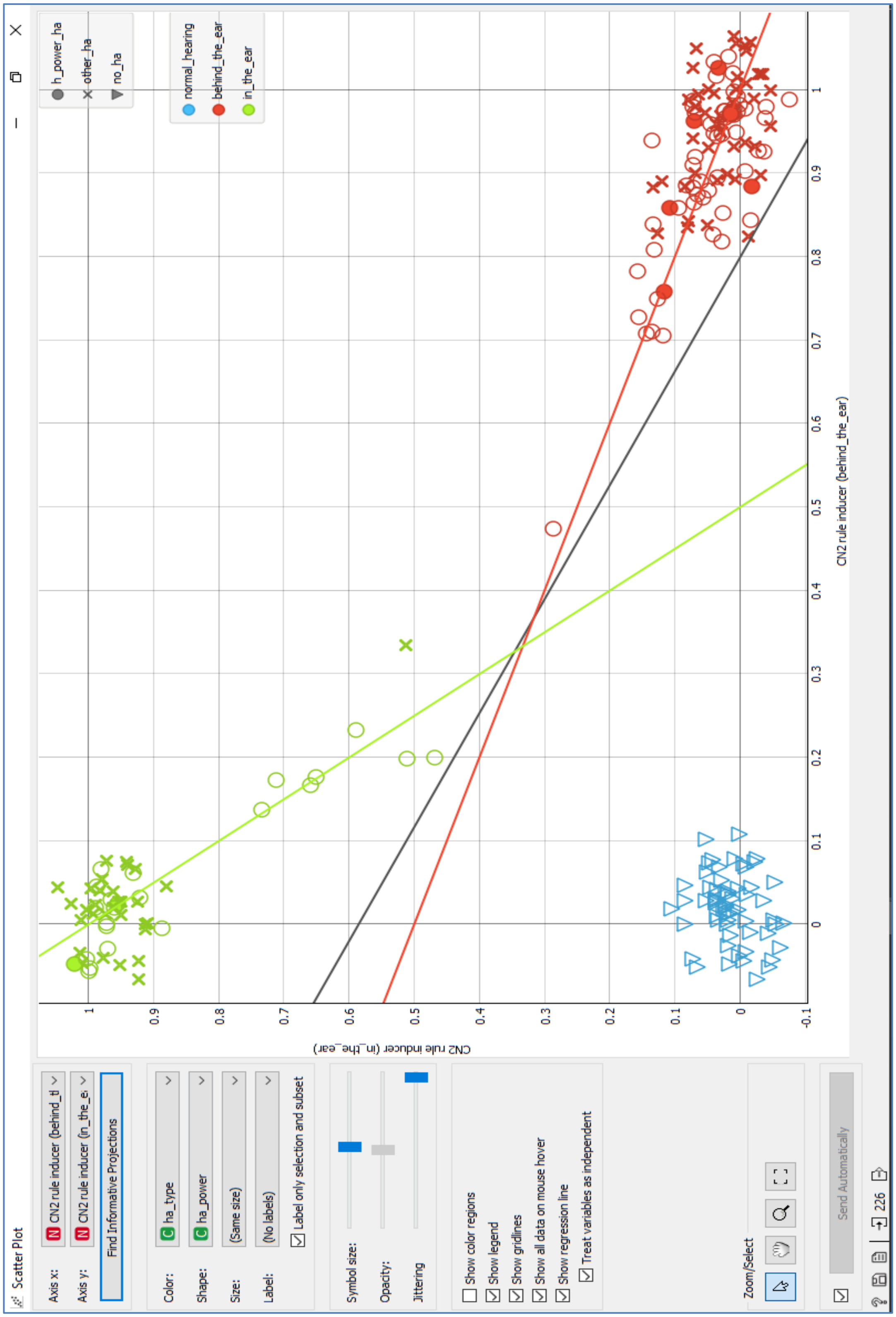
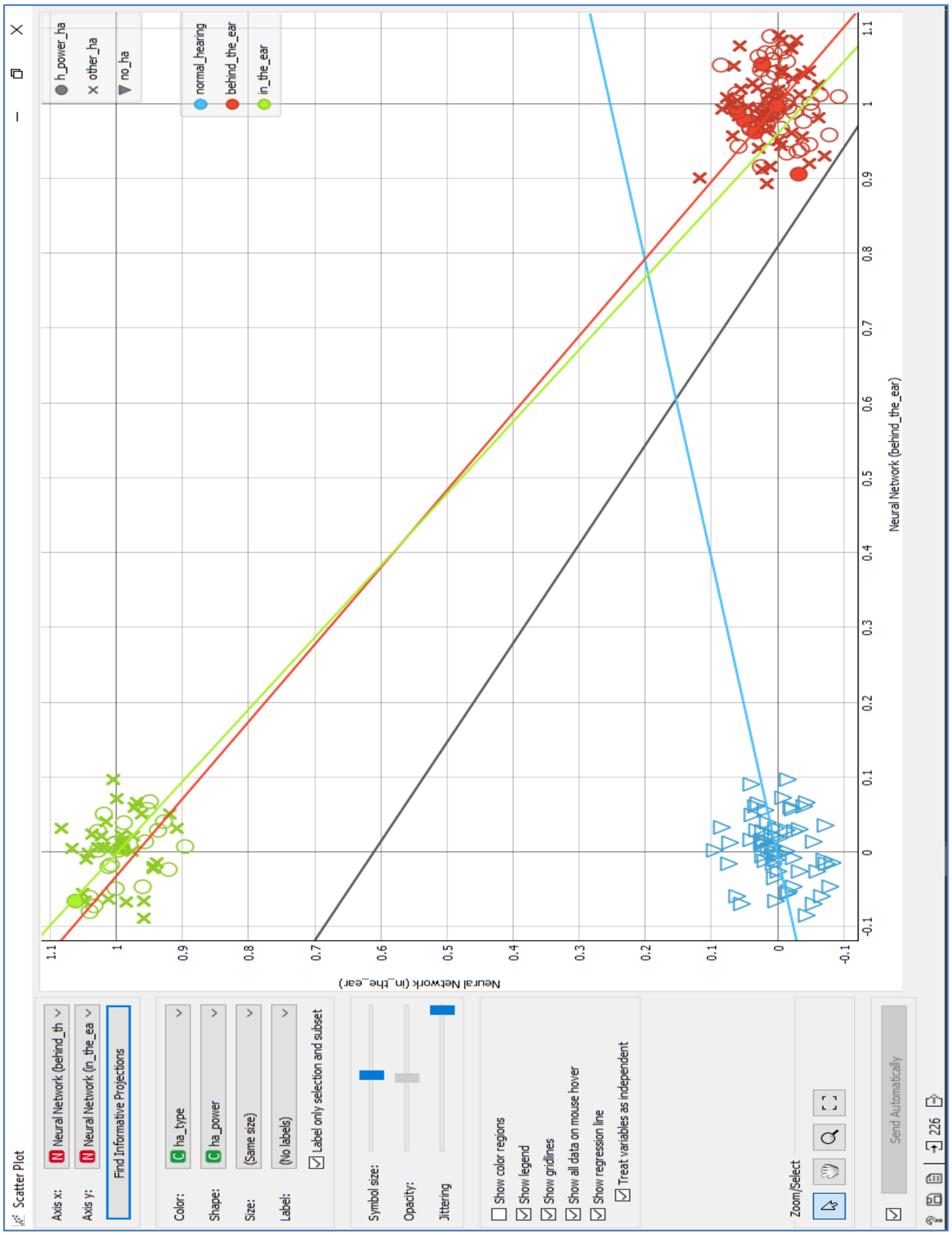


Figure 3.10 shows the Scatter Plot comparison predictions between CN2 rule inducer for one class of dataset (behind_the_ear) HA as X exes and NN for other class (in_the_ear) HA as Y exes:

Figure 3.10: The Scatter Plot of compare predictions between CN2 and NN for BTE Hearing Aid as X exes and ITE Hearing Aid as Y exes



3.1.2 Unsupervised Results and Comparison

Figure 3.11 shows the Mainfold Learning (ML) main model as unsupervised technique applying Orange Canvas Modeler:

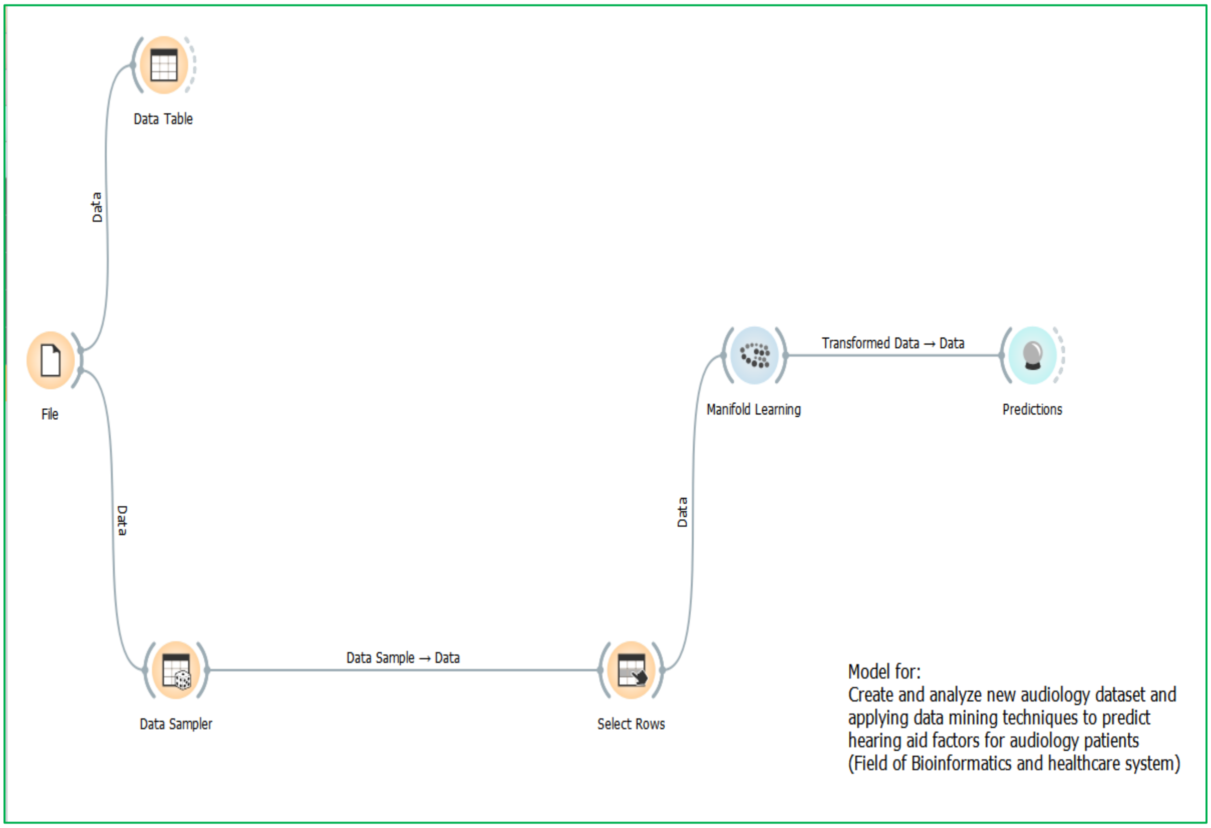
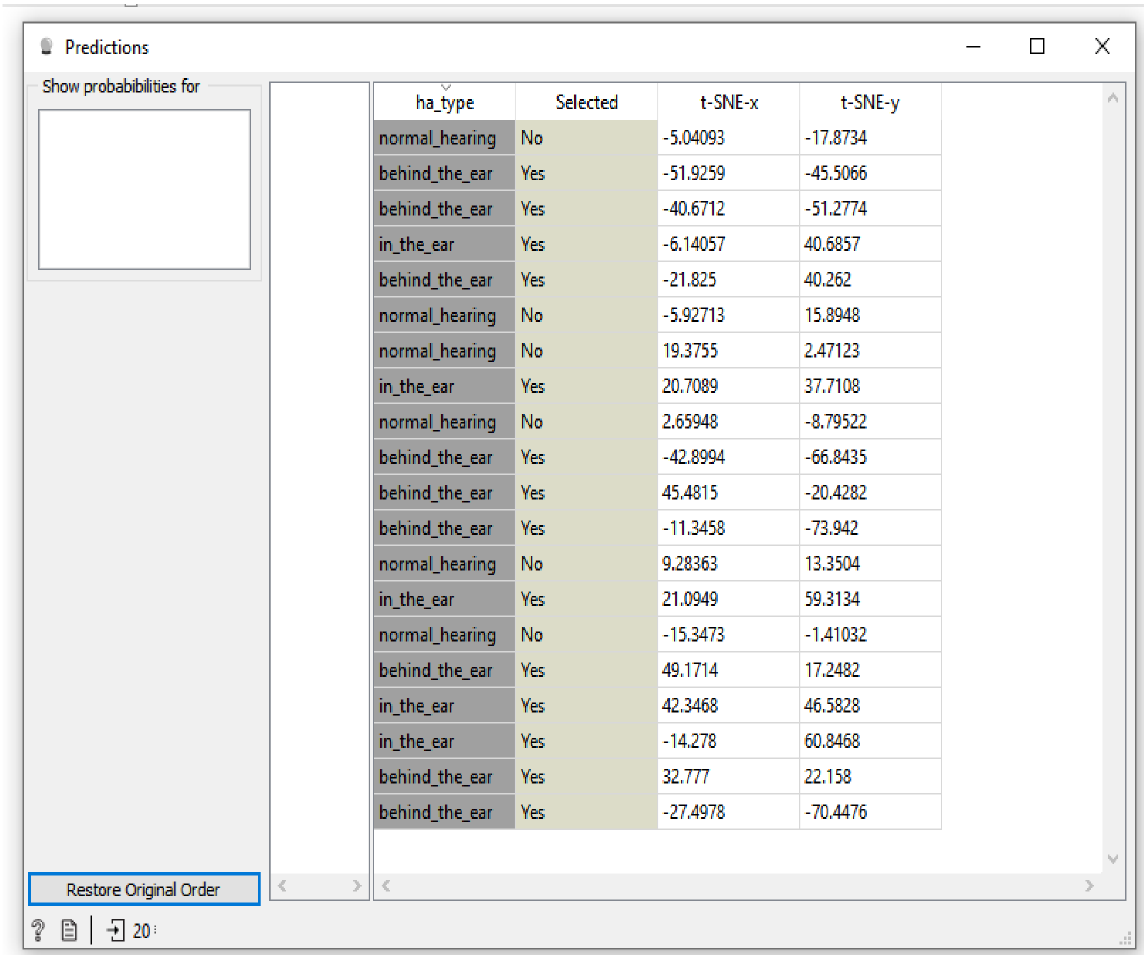


Figure 3.11: The ML main model

Figure 3.12 illustrates the specifics of ML prediction results:



ha_type	Selected	t-SNE-x	t-SNE-y
normal_hearing	No	-5.04093	-17.8734
behind_the_ear	Yes	-51.9259	-45.5066
behind_the_ear	Yes	-40.6712	-51.2774
in_the_ear	Yes	-6.14057	40.6857
behind_the_ear	Yes	-21.825	40.262
normal_hearing	No	-5.92713	15.8948
normal_hearing	No	19.3755	2.47123
in_the_ear	Yes	20.7089	37.7108
normal_hearing	No	2.65948	-8.79522
behind_the_ear	Yes	-42.8994	-66.8435
behind_the_ear	Yes	45.4815	-20.4282
behind_the_ear	Yes	-11.3458	-73.942
normal_hearing	No	9.28363	13.3504
in_the_ear	Yes	21.0949	59.3134
normal_hearing	No	-15.3473	-1.41032
behind_the_ear	Yes	49.1714	17.2482
in_the_ear	Yes	42.3468	46.5828
in_the_ear	Yes	-14.278	60.8468
behind_the_ear	Yes	32.777	22.158
behind_the_ear	Yes	-27.4978	-70.4476

Figure 3.12: The specifics of ML prediction results

Figure 3.13 shows the MDS main model as unsupervised technique applying Orange Canvas Modeler:

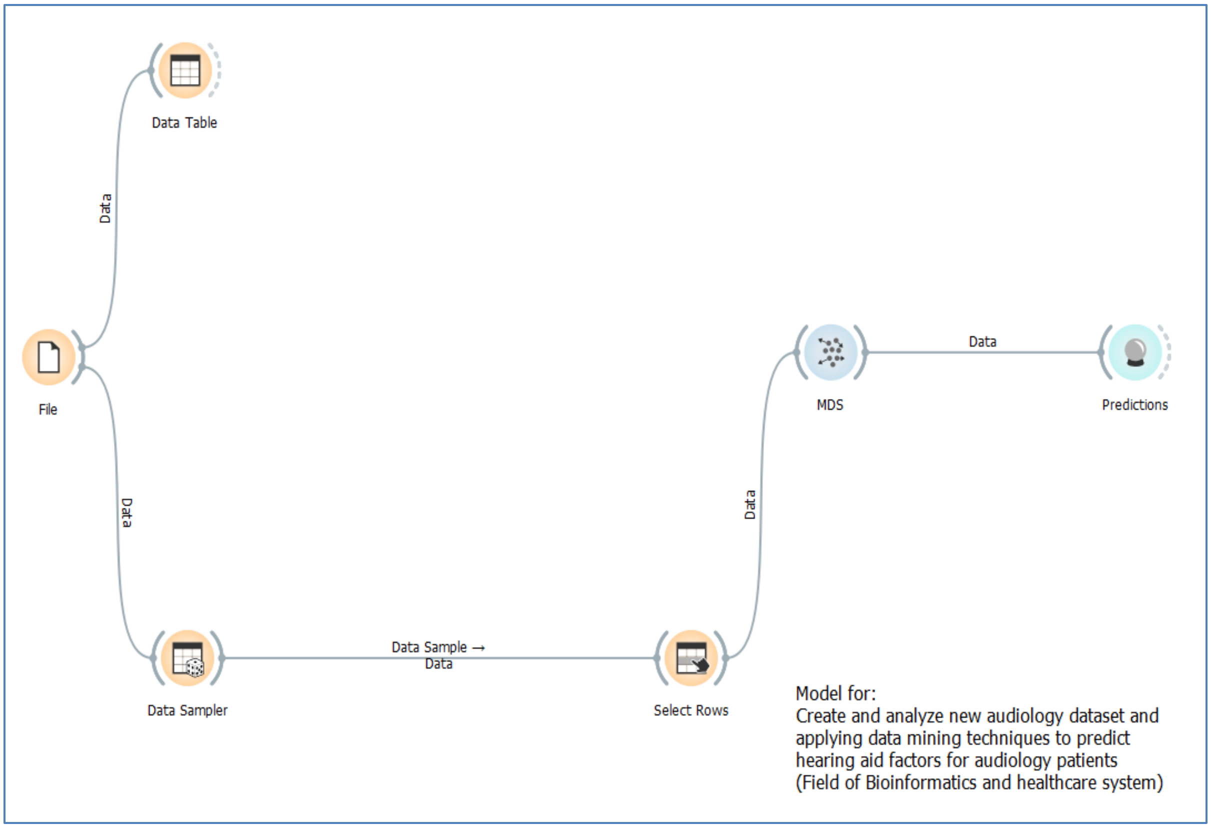
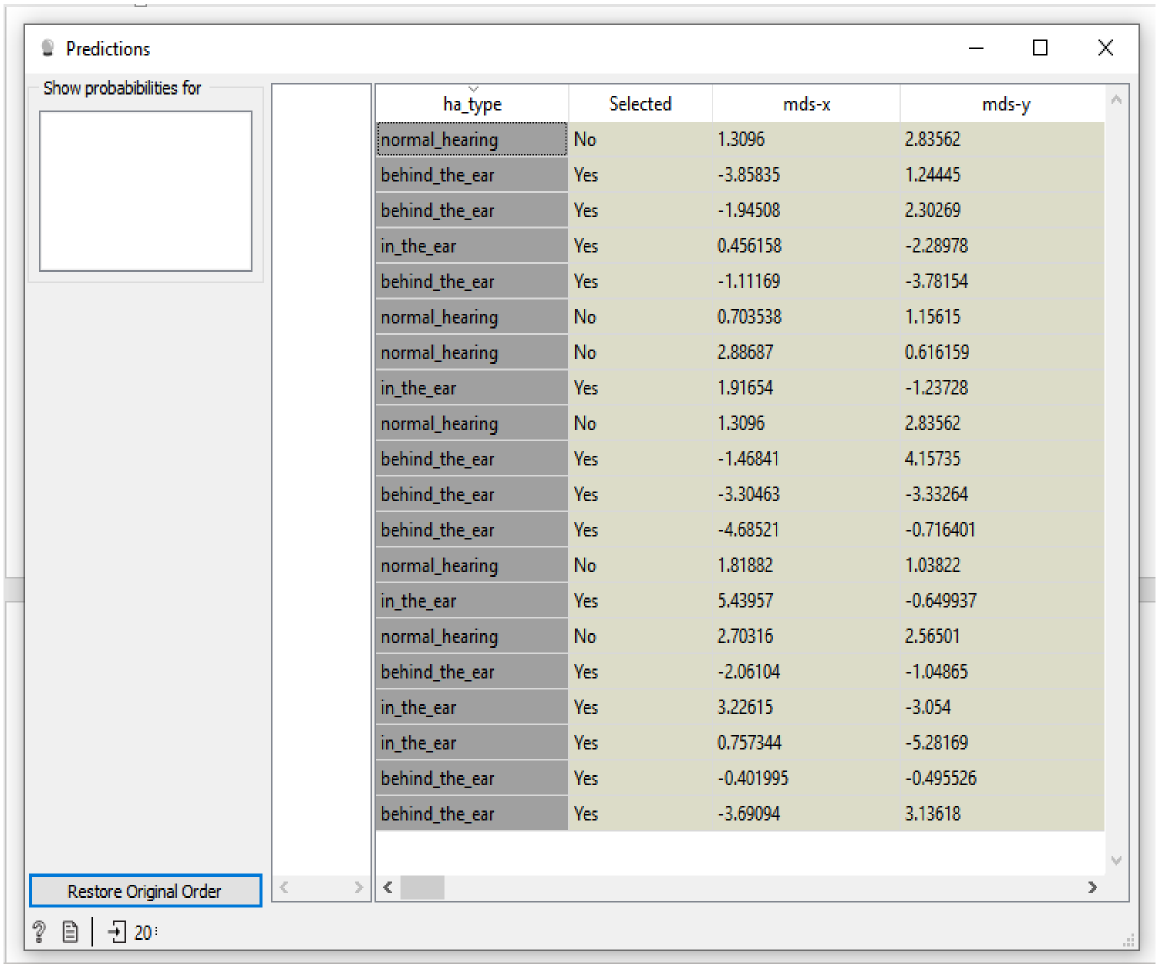


Figure 3.13: The MDS main model

Figure 3.14 illustrates the specifics of MDS prediction results:



ha_type	Selected	mds-x	mds-y
normal_hearing	No	1.3096	2.83562
behind_the_ear	Yes	-3.85835	1.24445
behind_the_ear	Yes	-1.94508	2.30269
in_the_ear	Yes	0.456158	-2.28978
behind_the_ear	Yes	-1.11169	-3.78154
normal_hearing	No	0.703538	1.15615
normal_hearing	No	2.88687	0.616159
in_the_ear	Yes	1.91654	-1.23728
normal_hearing	No	1.3096	2.83562
behind_the_ear	Yes	-1.46841	4.15735
behind_the_ear	Yes	-3.30463	-3.33264
behind_the_ear	Yes	-4.68521	-0.716401
normal_hearing	No	1.81882	1.03822
in_the_ear	Yes	5.43957	-0.649937
normal_hearing	No	2.70316	2.56501
behind_the_ear	Yes	-2.06104	-1.04865
in_the_ear	Yes	3.22615	-3.054
in_the_ear	Yes	0.757344	-5.28169
behind_the_ear	Yes	-0.401995	-0.495526
behind_the_ear	Yes	-3.69094	3.13618

Figure 3.14: The specifics of MDS prediction results

4. OVERVIEW AND CONCLUSION

4.1 OVERVIEW

Audiology is the medicine and science branch concerned of the hearing sense. Audiologist is the particularism person concerned of the hearing evaluation as well diagnosis of balance problems. The essential roles of audiologist are understanding the problems of balance testing, hearing assessment, hearing diagnosis and treatment, balance problems, forbidding hearing loss or impairment, and speech problems [151].

“Behind The Ear” (BTE) HAs are most widely utilized because this type appropriate for all patients, especially the patients over 60 years and under 18 years who unable to use “In The Ear” (ITE) HA for some reasons [160].

This HA type can easily clean, as well it is separate and simple to preserve with easy to maintenance. That is why this HA type is proper for all APs ages [161].

Figure 4.1 illustrates the types of both “BTE” as well as “ITE” HAs.

Figure 4.1: The samples of both “BTE” and “ITE” HAs

“In The Ear” (ITE) HAs type are fitted within the concha (the outer ear bowl). They made individually to proper each ear. But in general, they cannot be preferred for children and young people, since the size of them cannot be adjusted such as the BTE HAs earmold, thus while the children grew up, the HAs for them should be repeatedly changed [162]. That is why this HAs type are not convenient for APs over 60 and under 18 years, since they are unable to sustain, care enough, and keep safe these tiny pieces of HAs type [161].

4.1.1 Dataset Records Types

The types of dataset records consist of three different data types:

- Audiograms: The graphs that illustrate the exact threshold of human hear (the lowest sound can be heard) in both ears. There are two colored graphical lines needed for both ears, first line to air conduction (the sound within the ear via headphone, it measures the ability overall hearing), and the second line to bone conduction (the sound given behind the ear through mastoid bone).
- Structured data: The tabular data (the conventional fields of database like gender, age, HA type, and other structured data).
- Unstructured text: The information like reports, phrases, or short sentences [17].

4.1.2 Importance of Healthcare

We will highlight the most considerable features of Healthcare as mentioned below:

- Detect the anomalies rapidly by the ability of access into all records of patients;
- Analyzing data using an automated system, that will helpful within repeated anomalies as well main case;
- The care quality as well as boost productivity by more frequent, shorter consultations, and remote;
- Interact quickly as well as easily with the structured method by shared several tools among elementary care supplier as well as the responsible fosters on continuous observation of APs;
- Supply support of motivational to the patients who wish to do so;
- Participating in biomedical studies via making use of healthcare databases tools [162].

4.2 CONCLUSION

With applying DM techniques, we acquired around 100% prediction to select the HA for patients who suffer from hearing problems in general, and around 98% prediction to determine which type of HA as well the power type of this HA that those patients have to use.

We studied most problems of summarizing and constraining assorted algorithms of DM. So, we concentrated on using these algorithms for predicting sets of diverse target attributes. In this research, we have supplied an effective and intelligent HAs prediction methods applying DM techniques. First of all, we have presented a dynamic approach of the significant patterns extraction from the data warehouses of hearing disease for the HA efficient prediction based on a calculated weightage, then chose the most frequent patterns which have value larger than the predefined threshold to assess the prediction for hearing impairment. Based on data exploration and business intelligence, most goals of mining are determined. Besides, the goals evaluated versus the models which trained. These models can answer complex questions in HA predicting.

Some considerations may take to keep on the process and refine further data, thus it will be surely supplement to the clinic dependability when they taking a right diagnosing of choose HA type which their patients need.

Moreover, this research addresses information mining uniqueness like restorative information. Thence, the research goals are to examine and discover factors influence the HA type accommodation. This modeling result makes an audiologist obtains second opinion from unbiased source includes beside classification an explanation of attributes which considered as most complexity in prediction, thus the audiologist usually makes a careful thoughtful decision about whether to should reconsidered or go on depend on the primary diagnoses.

For future work, this can be more expanded and enhanced. To predict the most hearing problems, the importantly attributes should list. In addition to the attributes which listed within medical literature, we can as well incorporate other DM techniques.

5. FUTURE WORK

This topic is the main basis for the next our work. After we discussed the importance of **HAs** for **APs** and how we worked to help them to choose the **HA** that suits their conditions, so this research is considered the base stage of whole work. Next, we will discuss how to design a certain electronic **HA** which make patients control the frequency that suits their hearing status.

Figure 5.1 review the basic circle of the approach **HA** which we will work during the next research to develop it as much as possible to be easier to use for all ages of **APs** and highly effective at the same time.

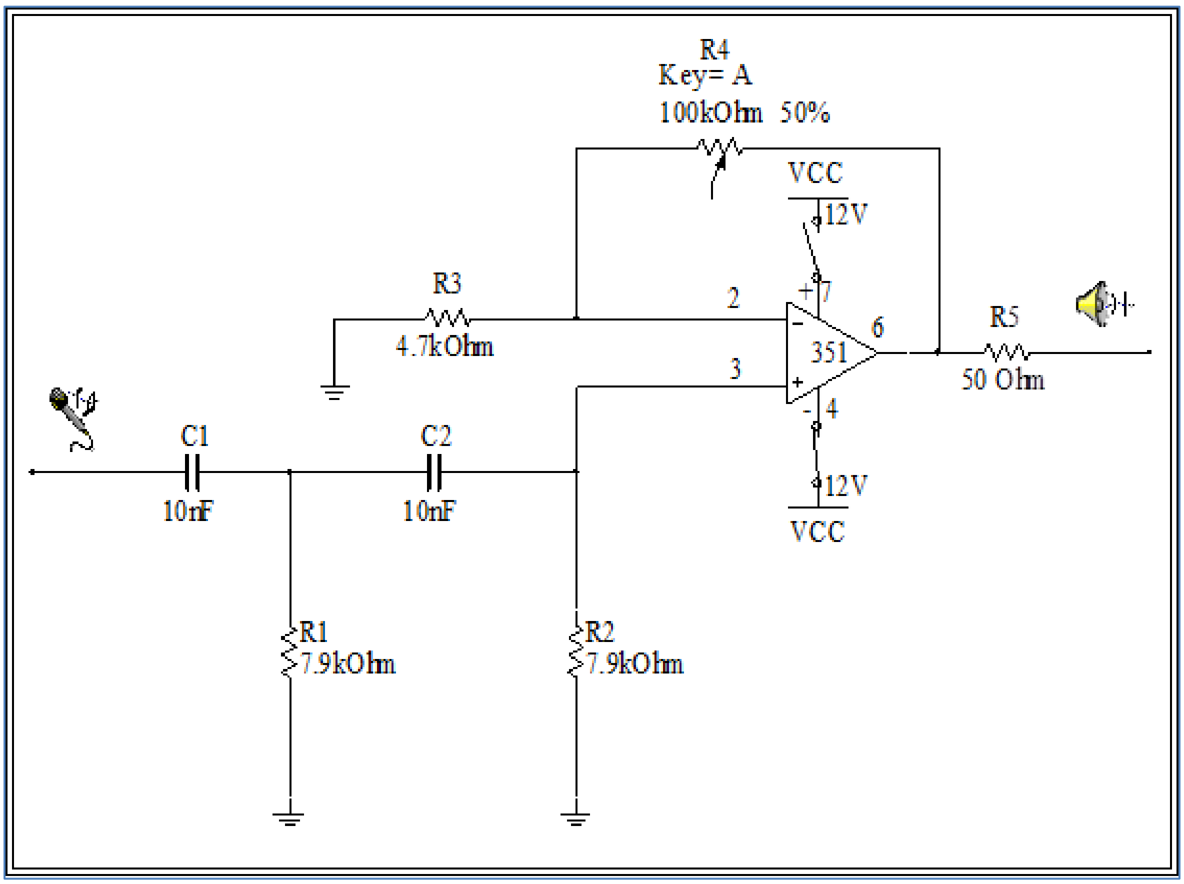


Figure 5.1: The basic electrical circuit design for the next proposed of HA

While:

$$f_0 = \frac{1}{2\pi\sqrt{R_1C_1R_2C_2}} \tag{5.1}$$

Where the f_0 is the cut off frequency.

Let $R_1 = R_2 = R$ and $C_1 = C_2 = C$

$$f_0 = \frac{1}{2\pi\sqrt{R^2C^2}} \tag{5.2}$$

Taking $C = 10nf$ and $f_0 = 2KHz$

$$R = \frac{1}{2\pi f_0 C} \tag{5.3}$$

$$= \frac{1}{2\pi \times 2 \times 10^3 \times 10 \times 10^{-9}}$$

$$= 7.9K\Omega$$

$$f_0 = \frac{1}{2\pi \times 10 \times 10^{-9} \times 7.9 \times 10^3}$$

$$= \frac{10^5}{15.8 \times 3.14}$$

$$= 2.014KHz$$

REFERENCES

[1] Maalim Ajabery and Sefer Kurnaz, "Applying datamining techniques to predict Hearing Aid type for audiology patients," *J. Inf. Sci. Eng.*, vol. 36, no. 2, pp. 205–215, 2020.

[2] Maalim Aljabery and Sefer Kurnaz, "Predict the type of hearing aid of audiology patients using data mining techniques," in *ACM International Conference Proceeding Series*, 2018, pp. 2–6.

[3] M. Mahamune, S. Ingle, P. Deo, and S. Chowhan, "Healthcare knowledge management using data mining techniques," *Adv. Comput. Res.*, vol. 7, no. 1, pp. 274–278, 2015.

[4] M. Islam, M. Hasan, X. Wang, H. Germack, and M. Noor-E-Alam, "A Systematic Review on Healthcare Analytics: Application and Theoretical Perspective of Data Mining," *Healthcare*, vol. 6, no. 2, p. 54, 2018.

[5] D. S. Kumar, G. Sathyadevi, and S. Sivanesh, "Decision Support System for Medical Diagnosis Using Data Mining," *Int. J. Comput. Sci. Issues*, vol. 8, no. 3, pp. 147–153, 2011.

[6] D. S. Jyoti Soni, Ujma Ansari, "Predictive Data Mining for Medical Diagnosis: An Overview of Heart Disease Prediction," *Int. J. Comput. Appl.*, vol. 17, no. 8, pp. 0975 – 8887.

[7] T. Silwattananusarn, "Data Mining and Its Applications for Knowledge Management : A Literature Review from 2007 to 2012," *Int. J. Data Min. Knowl. Manag. Process*, vol. 2, no. 5, pp. 13–24, 2012.

[8] V. R. M. Durairaj, "Data Mining Applications In Healthcare Sector: A Study," *Int. J. Sci. Technol. Res.*, vol. 2, no. 10.

[9] C. G. Soni, Nakul, "Application of Data Mining To Health Care," *Int. J. Comput. Sci. its Appl.*, no. June, 2014.

[10] E. M. Beulah, S. N. S. Rajini, and N. Rajkumar, "Application of Data mining in healthcare: A survey," *Asian J. Microbiol. Biotechnol. Environ. Sci.*, vol. 18, no. 4, pp.

999–1001, 2016.

[11] P. Ahmad, S. Qamar, and S. Qasim Afser Rizvi, “Techniques of Data Mining In Healthcare: A Review,” *Int. J. Comput. Appl.*, vol. 120, no. 15, pp. 38–50, 2015.

[12] R. Ray, “Advances in Data Mining: Healthcare Applications,” *Int. Res. J. Eng. Technol.*, pp. 3738–3742, 2018.

[13] S. Patel and H. Patel, “Survey of Data Mining Techniques used in Healthcare Domain,” *Int. J. Inf. Sci. Tech.*, vol. 6, no. 1/2, pp. 53–60, 2016.

[14] B. Lavanya and B. Divya, “International Journal of Computer Science and Mobile Computing BIG DATA ANALYSIS USING SVM AND K-NN DATA MINING TECHNIQUES,” *Int. J. Comput. Sci. Mob. Comput.*, vol. 6, no. 1, pp. 84–91, 2017.

[15] R. D. Canlas Jr, “DATA MINING IN HEALTHCARE : Data Mining in Healthcare : Current Applications and Issues,” *Unpubl. Master Thesis*, no. August, pp. 1–10, 2009.

[16] S. Scaramuzzino, P. H. Guzzi, C. Petrolo, G. Chiarella, P. Veltri, and M. Cannataro, “AudioEPR: A specialized electronic patient records for the semi-automatic management of clinical data in audiology,” *Proc. - IEEE Symp. Comput. Med. Syst.*, no. June, 2012.

[17] S. Cox, M. Oakes, S. Wermter, and M. Hawthorne, “Audiomine: Medical Data Mining in Heterogeneous Audiology Records,” *Int. J. Comput. Inf. Eng.*, vol. 1, no. 1, pp. 141–144, 2007.

[18] N. G. Noma, M. Khanapi, and A. Ghani, “Identifying Relationship between Hearing loss Symptoms and Pure-tone Audiometry Thresholds with FP-Growth Algorithm,” *Int. J. Comput. Appl.*, vol. 65, no. 21, pp. 975–8887, 2013.

[19] A. Pushpan and A. A. N, “Data Mining Applications in Healthcare National Conference On Discrete Mathematics & Computing (NCDMC-2017),” pp. 4–7, 2017.

[20] J. Iavindrasana, G. Cohen, A. Depeursinge, H. Müller, R. Meyer, and A. Geissbuhler, “Clinical data mining: a review.,” *Yearb. Med. Inform.*, no. June 2014, pp. 121–133, 2009.

[21] N. Jothi, N. A. Rashid, and W. Husain, “Data Mining in Healthcare - A Review,”

Procedia Comput. Sci., vol. 72, pp. 306–313, 2015.

[22] H. C. Koh and G. Tan, "Data mining applications in healthcare.," *J. Healthc. Inf. Manag.*, vol. 19, no. 2, pp. 64–72, 2005.

[23] S. Gupta, D. Kumar, and A. Sharma, "Data Mining Classification Techniques Applied for Breast Cancer Diagnosis and Prognosis," *J. Comput. Sci.*, vol. 2, no. 2, pp. 188–195, 2011.

[24] H. Bhavsar and M. H. Panchal, "A Review on Support Vector Machine for Data Classification," *Int. J. Adv. Res. Comput. Eng. Technol.*, vol. 1, no. 10, pp. 2278–1323, 2012.

[25] B. Milovic, "Prediction and decision making in Health Care using Data Mining," *Int. J. Public Heal. Sci.*, vol. 1, no. 2, pp. 69–76, 2012.

[26] A. S. M. L. H. Shahidul Islam Khan, "Development of National Health Data Warehouse for Data Mining," *Database Syst. J.*, vol. VI, no. 1, 2015.

[27] A. Khatib, M. Ahmad Alkhatib, A. Talaei-Khoei, and A. Hossein Ghapanchi, "Australasian Conference on Information Systems Analysis of Research in Healthcare Data Analytics," *Australas. Conf. Inf. Syst.*, pp. 1–16, 2015.

[28] J. A. Sáez, M. Galar, J. Luengo, and F. Herrera, "Tackling the problem of classification with noisy data using Multiple Classifier Systems: Analysis of the performance and robustness," *Inf. Sci. (Ny).*, vol. 247, pp. 1–20, 2013.

[29] M. Bataineh and T. Marler, "Neural network for regression problems with reduced training sets," *Neural Networks*, vol. 95, no. August, pp. 1–9, 2017.

[30] K. K. Dobbin and R. M. Simon, "Optimally splitting cases for training and testing high dimensional classifiers," *BMC Med. Genomics*, vol. 4, no. 1, p. 31, 2011.

[31] S. Theocharides, G. Makrides, G. E. Georghiou, and A. Kyprianou, "Machine learning algorithms for photovoltaic system power output prediction," *2018 IEEE Int. Energy Conf. ENERGYCON 2018*, no. June, pp. 1–6, 2018.

[32] J. A. Olvera-López, J. A. Carrasco-Ochoa, J. F. Martínez-Trinidad, and J. Kittler, "A

review of instance selection methods," *Artif. Intell. Rev.*, vol. 34, no. 2, pp. 133–143, 2010.

[33] D. A. Abdulameer, "Medical Data Mining : Health Care Knowledge Discovery Framework Based On Clinical Big Data Analysis," *Int. J. Sci. Res. Publ.*, vol. 5, no. 7, pp. 1–6, 2015.

[34] A. N, Rajashree, and S. B. K, "A Survey on Medical Data by using Data Mining Techniques," *Int. J. Sci. Eng. Technol. Res.*, vol. 07, no. 1, pp. 11–17, 2018.

[35] H. Kaur and S. K. Wasan, "Empirical Study on Applications of Data Mining Techniques in Healthcare," *J. Comput. Sci.*, vol. 2, no. 2, pp. 194–200, 2006.

[36] P. H. Bhavsar and A. Ganatra, "An Empirical Evaluation of Data Mining Classification Algorithms An Empirical Evaluation of Data Mining Classification Algorithms," vol. 14, no. May, pp. 142–150, 2016.

[37] E. R. S.Neelamegam, "Classification algorithm in Data mining: An Overview," *Int. J. P2P Netw. Trends Technol.*, vol. 3, no. 5, pp. 255–257, 2013.

[38] O. F.Y, A. J.E.T, A. O, H. J. O, O. O, and A. J, "Supervised Machine Learning Algorithms: Classification and Comparison," *Int. J. Comput. Trends Technol.*, vol. 48, no. 3, pp. 128–138, 2017.

[39] Z. (Z. R. Reitermanov´a and C. R. Charles University, Faculty of Mathematics and Physics, Prague, "Data Splitting," in *WDS'10 Proceedings of Contributed Papers*, 2010, pp. 31–36.

[40] Z. Ghahramani, "LNAI 3176 - Unsupervised Learning," *Mach. Learn.*, pp. 72–112, 2004.

[41] B. Krishnapuram, L. Carin, M. A. T. Figueiredo, and A. J. Hartemink, "Sparse multinomial logistic regression: Fast algorithms and generalization bounds," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 6, pp. 957–968, 2005.

[42] D. Delen, "Analysis of cancer data: A data mining approach," *Expert Syst.*, vol. 26, no. 1, pp. 100–112, 2009.

[43] A. E. Mohamed, “Comparative Study of Four Supervised Machine Learning Techniques for Classification,” *Int. J. Appl. Sci. Technol.*, vol. 7, no. 2, pp. 5–18, 2017.

[44] C. Z. Jiaqi Wang, Xindong Wu, “Support vector machines based on K-means clustering for real-time business intelligence systems,” *Int. J. Bus. Intell. Data Min.*, vol. 1, no. 1, pp. 54–64, 2005.

[45] S. S. Keerthi, O. Chapelle, and D. DeCoste, “Building support vector machines with reduced classifier complexity,” *J. Mach. Learn. Res.*, vol. 7, pp. 1493–1515, 2006.

[46] T. Evgeniou and M. Pontil, “Workshop on Support Vector Machines : Theory and Applications,” *Mach. Learn. Its Appl. Adv. Lect.*, no. January 2001, pp. 249–257, 2001.

[47] P. J. Mahendra H N, Shivakumar B R, “Pixel-based Classification of Multispectral Remotely Sensed Data Using Support Vector Machine Classifier,” in *National Conference on Advanced Innovation in Engineering and Technology (NCAIET-2015)*, 2015, vol. 3, no. 1.

[48] S. H. Zahiri, “Fuzzy gravitational search algorithm an approach for data mining,” *Iran. J. Fuzzy Syst.*, vol. 9, no. 1, pp. 21–37, 2012.

[49] D. T. Pham, S. Bigot, and S. S. Dimov, “RULES-5: a rule induction algorithm for classi_cation problems involving continuous attributes,” in *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 2003, vol. 217, no. 12, pp. 1273–1286.

[50] E. A. Kaur, E. K. Saurabh, and E. G. Singh, “A Combined Approach of Data Mining Algorithms Based on Association Rule Mining and Rule Induction,” *Int. J. Soft Comput. Eng.*, no. 5, pp. 61–62, 2013.

[51] N. Lavrač NADALAVRAC, B. Kavšek BRANKOKAVSEK, P. Flach PETERFLACH, and L. Todorovski LJUPCOTODOROVSKI, “Subgroup Discovery with CN2-SD,” *J. Mach. Learn. Res.*, vol. 5, pp. 153–188, 2004.

[52] D. G. L. D. Sujatha, M, S. Prabhakar, “A Survey of Classification Techniques in Data Mining,” *Int. J. Innov. Eng. Technol.*, vol. 2, no. 4, pp. 86–92, 2013.

[53] F. Kabir, C. Mofizur Rahman, A. Hossain, and K. Dahal, “Enhanced Classification Accuracy on Naive Bayes Data Mining Models,” *Int. J. Comput. Appl.*, vol. 28, no. 3, pp. 9–16, 2011.

[54] M. S. H. Shyara Taruna R, “Enhanced Naïve Bayes Algorithm for Intrusion Detection in Data Mining,” *J. Comput. Sci. Inf. Technol.*, vol. 4, no. 6, pp. 960–962, 2013.

[55] M. Karim and R. M. Rahman, “Decision Tree and Naïve Bayes Algorithm for Classification and Generation of Actionable Knowledge for Direct Marketing,” *J. Softw. Eng. Appl.*, vol. 06, no. 04, pp. 196–206, 2013.

[56] D. R. P. ADITI ANIL GHIVE, “IMPLEMENTATION OF SVM AND NB ALGORITHMS FOR CLASSIFICATION OF DISEASES AND THEIR TREATMENTS,” *Int. J. Adv. Sci. Eng. Technol.*, no. 4, pp. 141–145, 2015.

[57] P. d. P.Bhargavi, M.Sc., M.Tech., Dr.S.Jyothi, M.Sc., M.S, “Applying Naive Bayes Data Mining Technique for Classification of Agricultural Land Soils,” *Int. J. Comput. Sci. Netw. Secur.*, vol. 9, no. 8, pp. 117–122, 2009.

[58] S. Kumar, A. Jain, and P. Mahalakshmi, “Enhancement of Healthcare Using Naïve Bayes Algorithm and Intelligent Datamining of Social Media,” *Int. J. Appl. Eng. Res.*, vol. 13, no. 6, pp. 4109–4112, 2018.

[59] V. Cherian and M. S. Bindu, “Heart Disease Prediction Using Naïve Bayes Algorithm and Laplace Smoothing Technique,” *Int. J. Comput. Sci. Trends Technol.*, vol. 5, no. 2, pp. 68–73, 2017.

[60] Sfenrianto, I. Purnamasari, and R. B. Bahaweres, “Naive Bayes classifier algorithm and Particle Swarm Optimization for classification of cross selling (Case study: PT TELKOM Jakarta),” in *Proceedings of 2016 4th International Conference on Cyber and IT Service Management, CITSM 2016*, 2016, no. April.

[61] O. G. N, “Optimized Naïve Bayesian Algorithm for Efficient Performance Optimized Naïve Bayesian Algorithm for Efficient Performance,” vol. 9, no. July, pp. 8–13, 2018.

[62] S. Redjeki, M. Guntara, and P. Anggoro, “Naive Bayes Classifier Algorithm Approach for

Mapping Poor Families Potential," *Int. J. Adv. Res. Artif. Intell.*, vol. 4, no. 12, pp. 29–33, 2015.

[63] M. S. D. Pouria Kaviani, "Short Survey on Naive Bayes Algorithm," *Int. J. Adv. Eng. Res. Dev.*, vol. 4, no. 11, pp. 607–611, 2017.

[64] H. Leopord, W. Kipruto Cheruiyot, and S. Kimani, "A Survey and Analysis on Classification and Regression Data Mining Techniques for Diseases Outbreak Prediction in Datasets," *Int. J. Comput. Appl.*, vol. 17, no. 8, pp. 0975 – 8887, 2016.

[65] S. Zhang, C. Tjortjis, X. Zeng, H. Qiao, I. Buchan, and J. Keane, *Comparing data mining methods with logistic regression in childhood obesity prediction*, vol. 11, no. 4. 2009.

[66] Y. Jiang, "Using Logistic Regression Model to Predict the Success of Bank Telemarketing," *Int. J. Data Sci. Technol.*, vol. 4, no. 1, pp. 35–41, 2018.

[67] C. J. Lin, R. C. Weng, and S. Sathiya Keerthi, "Trust region Newton method for large-scale logistic regression," *J. Mach. Learn. Res.*, vol. 9, pp. 627–650, 2008.

[68] S. T. Chen *et al.*, "Comparative analysis of logistic regression, support vector machine and artificial neural network for the differential diagnosis of benign and malignant solid breast tumors by the use of three-dimensional power Doppler imaging," *Korean J. Radiol.*, vol. 10, no. 5, pp. 464–471, 2009.

[69] C. Y. J. Peng, K. L. Lee, and G. M. Ingersoll, "An introduction to logistic regression analysis and reporting," *J. Educ. Res.*, vol. 96, no. 1, pp. 3–14, 2002.

[70] F. Halili and A. Rustemi, "Predictive Modeling: Data Mining Regression Technique Applied in a Prototype," *Int. J. Comput. Sci. Mob. Comput.*, vol. 5, no. 8, pp. 207–215, 2016.

[71] D. P. Mishra and J. Min, "Analyzing the Relationship between Dependent and Independent Variables in Marketing: A Comparison of Multiple Regression with Path Analysis," *SSRN Electron. J.*, vol. 6, no. 3, pp. 113–120, 2010.

[72] A. K. Mishra and B. K. Ratha, "Study of Random Tree and Random Forest Data Mining

Algorithms for Microarray Data Analysis," *Int. J. Adv. Electr. Comput. Eng.*, vol. 3, no. 4, pp. 5–7, 2016.

[73] B. Lowe and A. Kulkarni, "Multispectral Image Analysis Using Random Forest," *Int. J. Soft Comput.*, vol. 6, no. 1, pp. 1–14, 2015.

[74] S. Bharathidason and C. Jothi Venkataeswaran, "Improving Classification Accuracy based on Random Forest Model with Uncorrelated High Performing Trees," *Int. J. Comput. Appl.*, vol. 101, no. 13, pp. 26–30, 2014.

[75] P. Probst and A. L. Boulesteix, "To tune or not to tune the number of trees in random forest," *J. Mach. Learn. Res.*, vol. 18, pp. 1–18, 2018.

[76] M. Denil, D. Matheson, and N. De Freitas, "Narrowing the Gap: Random Forests In TheDenil, M., Matheson, D., & De Freitas, N. (2014). Narrowing the Gap: Random Forests In Theory and In Practice. Proceedings of The 31st International Conference on Machine Learning, (1998), 665–673. Retrieved from ht," *Proc. 31st Int. Conf. Mach. Learn.*, vol. 32, no. 1998, pp. 665–673, 2014.

[77] A. Palczewska, J. Palczewski, and R. M. Robinson, "Interpreting random forest classification models using a feature contribution method," *Integr. Reusable Syst. Springer*, vol. 26, pp. 193–218, 2014.

[78] A. Liaw and M. Wiener, "Classification and Regression by randomForest," *R News*, vol. 2, no. 3, pp. 18–22, 2002.

[79] K. Tanaka, T. Kinkyo, and S. Hamori, "Financial Hazard Map: Financial vulnerability predicted by a random forests classification model," *Sustain.*, vol. 10, no. 5, pp. 1–18, 2018.

[80] B. S. Yang, X. Di, and T. Han, "Random forests classifier for machine fault diagnosis," *J. Mech. Sci. Technol.*, vol. 22, no. 9, pp. 1716–1725, 2008.

[81] P. L. Bartlett and M. Traskin, "AdaBoost is consistent," *Adv. Neural Inf. Process. Syst.*, vol. 8, pp. 105–112, 2007.

[82] H. Fleyeh and E. Davami, “Multiclass Adaboost Based on an Ensemble of Binary AdaBoosts,” *Am. J. Intell. Syst.*, vol. 3, no. 2, pp. 57–70, 2013.

[83] A. J. Wyner, M. Olson, J. Bleich, and D. Mease, “Explaining the success of adaboost and random forests as interpolating classifiers,” *J. Mach. Learn. Res.*, vol. 18, pp. 1–33, 2017.

[84] C. Niyomugabo, H. R. Choi, and T. Y. Kim, “A Modified Adaboost Algorithm to Reduce False Positives in Face Detection,” *Math. Probl. Eng.*, vol. 2016, p. 6, 2016.

[85] I. Y. H. G. Mohamed H. Changrampadi, Yixiao Yun, “Multi-Class Ada-Boost Classification of Object Poses through Visual and Infrared Image Information Fusion Mohamed,” in *21st International Conference on Pattern Recognition (ICPR 2012)*, 2012, pp. 2865–2868.

[86] T. Windeatt and R. Ghaderi, “AdaBoost and neural networks,” in *European Symposium on Artificial Neural Networks (ESANN)*, 1999, no. April, pp. 123–128.

[87] J. Bergstra, N. Casagrande, D. Erhan, D. Eck, and B. Kégl, “Aggregate features and ADABOOST for music classification,” *Mach. Learn.*, vol. 65, no. 2–3, pp. 473–484, 2006.

[88] V. K. P. German, B. D. Gerardo, and R. P. Medina, “Implementing Enhanced AdaBoost Algorithm for Sales Classification and Prediction,” *Int. J. Trade, Econ. Financ.*, vol. 8, no. 6, pp. 270–273, 2017.

[89] X. Li, L. Wang, and E. Sung, “AdaBoost with SVM-based component classifiers,” *Eng. Appl. Artif. Intell.*, vol. 21, no. 5, pp. 785–795, 2008.

[90] P. Gaur, “Neural Networks in Data Mining,” *Int. J. Electron. Comput. Sci. Eng.*, pp. 1449–1453, 1996.

[91] N. Upadhyay, “Educational Data Mining by Using Neural Network,” *Int. J. Comput. Appl. Technol. Res.*, vol. 5, no. 2, pp. 104–109, 2016.

[92] D. K. S. Er.Romil.V.Patel, “Introduction to Integrating Web Mining With Neural Network,” *Int. J. Comput. Sci. Inf. Technol. Secur.*, vol. 2, no. 6, pp. 1166–1169, 2012.

[93] N. Pushkar Shinde MCOERC, “Data Mining using Artificial Neural Network Rules,” *Int. J. Innov. Eng. Technol.*, vol. 3, no. 1, pp. 157–162, 2013.

[94] Ajay Gupta, “Neural Networks in Data Processing,” *Int. J. Adv. Res. Comput. Eng. Technol.*, vol. 5, no. 5, pp. 1638–1642, 2016.

[95] R. Singh Gill, “Neural Networks in Data Mining,” *IOSR J. Eng.*, vol. 4, no. 3, pp. 01–06, 2014.

[96] S. M. Kamruzzaman and A. M. Jehad Sarkar, “A new data mining scheme using artificial neural networks,” *Sensors*, vol. 11, no. 5, pp. 4622–4647, 2011.

[97] D. Das and M. S. Uddin, “DATA MINING AND NEURAL NETWORK TECHNIQUES IN STOCK MARKET PREDICTION: A METHODOLOGICAL REVIEW,” *Int. J. Artif. Intell. Appl.*, vol. 4, no. 1, pp. 69–71, 2013.

[98] S.Sathyamoorthy, “NEURAL NETWORKS IN DATA MINING,” *Int. Res. J. Eng. Technol.*, vol. 2, no. 9, pp. 431–434, 2015.

[99] R. S. Agrawal and P. M. H. Pandya, “Survey of Papers for Data Mining with Neural Networks to Predict the Student’s Academic Achievements,” *Int. J. Comput. Sci. Trends Technol.*, vol. 3, no. 5, pp. 246–249, 2015.

[100] M. C. Arockiaraj, “Applications of Neural Networks In Data Mining,” *Res. Inven. Int. J. Eng. Sci.*, vol. 3, no. 1, pp. 08–11, 2013.

[101] S. D. B.B. Misra, “Functional Link Artificial Neural Network for Classification Task in Data Mining,” *J. Comput. Sci.*, vol. 3, no. 12, pp. 948–955, 2007.

[102] S. S. Kharat and V. Krishna, “To Study Artificial Neural Networks in Data Mining and its methods,” *Int. J. Adv. Res. Comput. Sci. Manag. Stud.*, vol. 3, no. 7, pp. 75–80, 2015.

[103] Z. Solan, D. Horn, E. Ruppin, and S. Edelman, “Unsupervised learning of natural languages,” *Proc. Natl. Acad. Sci. U. S. A.*, vol. 102, no. 33, pp. 11629–11634, 2005.

[104] R. Sathya and A. Abraham, “Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification,” *Int. J. Adv. Res. Artif. Intell.*, vol. 2, no. 2, pp. 34–

38, 2013.

[105] M. Usama *et al.*, “Unsupervised Machine Learning for Networking: Techniques, Applications and Research Challenges,” *IEEE Access*, vol. 7, no. September 2017, pp. 65579–65615, 2019.

[106] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*, vol. 9781107057. 2014.

[107] E. Oja, “Unsupervised learningin neural computation,” *Theor. Comput. Sci.*, vol. 287, no. 1, pp. 187–207, 2002.

[108] P. Mordohai and G. Medioni, “Dimensionality Estimation , Manifold Learning and Function Approximation using Tensor Voting,” *J. Mach. Learn. Res.*, vol. 11, pp. 411–450, 2010.

[109] M. C. Yeh, I. H. Lee, G. Wu, Y. Wu, and E. Y. Chang, “Manifold learning, A promised land or work in progress?,” *IEEE Int. Conf. Multimed. Expo, ICME 2005*, vol. 2005, no. January 2005, pp. 1154–1157, 2005.

[110] Y. Goldberg, “Manifold Learning : The Price of Normalization,” *J. Mach. Learn. Res. 9*, vol. 9, pp. 1909–1939, 2008.

[111] E. Golchin and K. Maghooli, “Overview of Manifold Learning and Its Application in Medical Data set,” *Int. J. Biomed. Eng. Sci.*, vol. 1, no. 2, p. 23, 2014.

[112] B. Lin, X. He, and J. Ye, “Open Access A geometric viewpoint of manifold learning,” *Appl. Informatics (a springer open journal)*, vol. 2, no. 3, p. 12, 2015.

[113] N. Zheng and J. Xue, “Statistical Learning and Pattern Analysis for Image and Video Processing,” in *Advances in Pattern Recognition*, vol. 44, no. 10, London, 2009, pp. 319–341.

[114] M. C. Hout, M. H. Papesh, and S. D. Goldinger, “Multidimensional scaling,” *WIREs Cogn Sci 2013*, vol. 4, no. February, pp. 93–103, 2013.

[115] J. de Leeuw and P. Mair, “Multidimensional Scaling Using Majorization : SMACOF in

R," *JournalofStatisticalSoftware*, vol. 31, no. 3, p. 30, 2009.

[116] S. Bai, X. Bai, L. J. Latecki, and Q. Tian, "Multidimensional Scaling on Multiple Input Distance Matrices," in *Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17) Multidimensional*, 2017, no. Sun, pp. 1281–1287.

[117] P. J. F. Groenen, S. Winsberg, O. Rodríguez, and E. Diday, "I-Scal: Multidimensional scaling of interval dissimilarities," *Comput. Stat. Data Anal.*, vol. 51, pp. 360–378, 2006.

[118] Y. Sahillioılu, "A shape deformation algorithm for constrained multidimensional scaling," *Comput. Graph.*, vol. 53, pp. 156–165, 2015.

[119] A. Buja, D. F. Swayne, M. L. Littman, N. Dean, H. Hofmann, and L. Chen, "Data visualization with multidimensional scaling," *J. Comput. Graph. Stat.*, vol. 17, no. 2, pp. 444–472, 2008.

[120] L. F. Fitzgerald and L. J. Hubert, "Multidimensional Scaling: Some Possibilities for Counseling Psychology," *J. Couns. Psychol.*, vol. 34, no. 4, pp. 469–480, 1987.

[121] J. Tzeng, H. Lu, and W. H. Li, "Multidimensional scaling for large genomic data sets," *BMC Bioinformatics*, vol. 9, pp. 1–17, 2008.

[122] J. Du, Y. Liu, Y. Yu, and W. Yan, "A prediction of precipitation data based on Support Vector Machine and Particle Swarm Optimization (PSO-SVM) algorithms," *Algorithms*, vol. 10, no. 2, 2017.

[123] C. Anwar and M. P. Oakes, "Clustering Audiology Data," in *Proceedings of the 19th Machine Learning conference of Belgium and The Netherlands*, 2010, p. 6.

[124] A. I. Weinberg and M. Last, "Selecting a representative decision tree from an ensemble of decision-tree models for fast big data classification," *J. Big Data*, vol. 6, no. 1, 2019.

[125] H. S. Ghadikolaei, H. Ghauch, C. Fischione, and M. Skoglund, "Learning and data selection in big datasets," *36th Int. Conf. Mach. Learn. ICML 2019*, vol. 2019-June, pp. 3848–3857, 2019.

[126] A. K. Tanwani, J. Afridi, M. Z. Shafiq, and M. Farooq, "Guidelines to select machine

learning scheme for classification of biomedical datasets," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5483 LNCS, pp. 128–139, 2009.

[127] B. Genç and H. Tunç, "Optimal training and test sets design for machine learning," *Turkish J. Electr. Eng. Comput. Sci.*, vol. 27, no. 2, pp. 1534–1545, 2019.

[128] J. Qui˜nonero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, *Dataset Shift in Machine Learning*. London, England: Massachusetts Institute of Technology, 2009.

[129] J. Leskovec, A. Rajaraman, and J. D. Ullman, "Mining of Massive Datasets," *Min. Massive Datasets*, p. 513, 2014.

[130] Y. Xu and R. Goodacre, "On Splitting Training and Validation Set: A Comparative Study of Cross-Validation, Bootstrap and Systematic Sampling for Estimating the Generalization Performance of Supervised Learning," *J. Anal. Test.*, vol. 2, no. 3, pp. 249–262, 2018.

[131] A. Maydeu-Olivares and C. García-Forero, "Goodness-of-fit testing," *Int. Encycl. Educ.*, no. October 2017, pp. 190–196, 2010.

[132] B. K. Rani and A. Govrdhan, "Applications of Data Mining Techniques in Healthcare and Prediction of Heart Attacks," *Int. J. Comput. Sci. Eng.*, vol. 02, no. 02, pp. 250–255, 2010.

[133] J. Van Den Broeck, S. A. Cunningham, R. Eeckels, and K. Herbst, "Data cleaning: Detecting, diagnosing, and editing data abnormalities," *PLoS Med.*, vol. 2, no. 10, pp. 0966–0970, 2005.

[134] M. Halkidi, Y. Batistakis, and M. Vazirgiannis, "On clustering validation techniques," *J. Intell. Inf. Syst.*, vol. 17, no. 2–3, pp. 107–145, 2001.

[135] H. K. Jabbar and R. Z. Khan, "Methods to Avoid Over-Fitting and Under-Fitting in Supervised Machine Learning (Comparative Study)," *Comput. Commun.*, no. February, pp. 163–172, 2015.

[136] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion

detection dataset and intrusion traffic characterization," in *In Proceedings ofthe 4th International Conference on Information Systems Security and Privacy (ICISSP 2018)*, 2018, vol. 2018-Janua, no. Cic, pp. 108–116.

[137] S. Dutta, S. Bandopadhyay, R. Ganguli, and D. Misra, "Machine Learning Algorithms and Their Application to Ore Reserve Estimation of Sparse and Imprecise Data," *J. Intell. Learn. Syst. Appl.*, vol. 02, no. 02, pp. 86–96, 2010.

[138] T. Borovicka, M. Jirina, P. Kordik, and M. Jiri, "Selecting Representative Data Sets," in *Advances in Data Mining Knowledge Discovery and Applications*, 2012, pp. 43–70.

[139] R. Leite, P. Brazdil, and J. Vanschoren, "Selecting classification algorithms with active testing," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 7376 LNAI, pp. 117–131, 2012.

[140] "No Title." [Online]. Available: https://www.healthyhearing.com/help/hearing-aids/types.

[141] J. Demšar, B. Zupan, G. Leban, and T. Curk, "Orange: From experimental machine learning to interactive data mining," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2004, vol. 3202, no. November, p. 15.

[142] M. Štajdohar and J. Demšar, "Interactive network exploration with Orange," *J. Stat. Softw.*, vol. 53, no. 6, pp. 1–24, 2013.

[143] P. Almeida, L. Gruenwald, and J. Bernardino, "Evaluating open source data mining tools for business," in *DATA 2016 - Proceedings of the 5th International Conference on Data Management Technologies and Applications*, 2016, no. Data, pp. 87–94.

[144] M. S. Kukasvadiya, D. Nidhi, and H. Divecha, "Analysis of Data Using Data Mining tool Orange," *Int. J. Eng. Dev. Res.*, vol. 5, no. 2, pp. 1836–1840, 2017.

[145] M. North, *Data Mining for the Masses*. 2012.

[146] M. Peker, O. Özkaraca, and A. Şaşar, "Use of orange data mining toolbox for data analysis in clinical decision making: The diagnosis of diabetes disease," *Expert Syst. Tech.*

Biomed. Sci. Pract., pp. 143–167, 2018.

[147] C. El Moucary, “Data Mining for Engineering Schools,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 2, no. 10, pp. 1–9, 2011.

[148] Sphinx 3.0.2 & Alabaster 0.7.12, “SVM,” *Orange Data Mining*, 2015. [Online]. Available: https://docs.biolab.si//3/visual-programming/widgets/model/svm.html.

[149] J. Kishore, M. Goel, and P. Khanna, “Understanding survival analysis: Kaplan-Meier estimate,” *Int. J. Ayurveda Res.*, vol. 1, no. 4, pp. 212–216, 2010.

[150] Sphinx 3.0.2 & Alabaster 0.7.12, “CN2 Rule Induction,” *Orange Data Mining*, 2015. [Online]. Available: https://docs.biolab.si//3/visual-programming/widgets/model/cn2ruleinduction.html.

[151] S. D. Preety, “SENTIMENT ANALYSIS USING SVM AND NAÏVE BAYES ALGORITHM,” *Int. J. Comput. Sci. Mob. Comput.*, vol. 4, no. 9, pp. 212–219, 2015.

[152] Sphinx 3.0.2 & Alabaster 0.7.12, “Naive Bayes,” *Orange Data Mining*, 2015. [Online]. Available: https://docs.biolab.si//3/visual-programming/widgets/model/naivebayes.html.

[153] Sphinx 3.0.2 & Alabaster 0.7.12, “Logistic Regression,” *Orange Data Mining*, 2015. [Online]. Available: https://docs.biolab.si//3/visual-programming/widgets/model/logisticregression.html.

[154] Sphinx 3.0.2 & Alabaster 0.7.12, “Random Forest,” *Orange Data Mining*, 2015. [Online]. Available: https://docs.biolab.si//3/visual-programming/widgets/model/randomforest.html.

[155] Sphinx 3.0.2 & Alabaster 0.7.12, “AdaBoost,” *Orange Data Mining*, 2015. [Online]. Available: https://docs.biolab.si//3/visual-programming/widgets/model/adaboost.html.

[156] Sphinx 3.0.2 & Alabaster 0.7.12, “Neural Network,” *Orange Data Mining*, 2015. [Online]. Available: https://docs.biolab.si//3/visual-programming/widgets/model/neuralnetwork.html.

[157] A. S. C. YASHPAL SINGH, “NEURAL NETWORKS IN DATA MINING,” *J. Theor. Appl. Inf. Technol.*, no. June, pp. 37–42, 2009.

[158] Sphinx 3.0.2 & Alabaster 0.7.12, “Manifold Learning,” *Orange Data Mining*, 2015. [Online]. Available: https://docs.biolab.si//3/visual-programming/widgets/unsupervised/manifoldlearning.html.

[159] Sphinx 3.0.2 & Alabaster 0.7.12, “Multidimensional Scaling,” *Orange Data Mining*, 2015. [Online]. Available: https://docs.biolab.si//3/visual-programming/widgets/unsupervised/mds.html.

[160] J. Xu and W. Han, “Improvement of adult BTE hearing aid wearers’ front/back localization performance using digital pinna-cue preserving technologies: An evidence-based review,” *Korean J. Audiol.*, vol. 18, no. 3, pp. 97–104, 2014.

[161] K. Sickel, V. Daum, and J. Hornegger, “Shortest Path Search with Constraints on Surface Models of in-the-ear Hearing Aids,” in *52nd Internationales Wissenschaftliches Kolloquium Technische Universität Ilmenau*, 2007, no. September, p. 6.

[162] M. Dey and S. S. Rautaray, “Study and Analysis of Data mining Algorithms for Healthcare Decision Support System,” *Int. J. Comput. Sci. Inf. Technol.*, vol. 5, no. 1, pp. 470–477, 2014.