

https://dspace.org/

Knowledge extraction in population health datasets: an exploratory data mining approach

Khangamwa, Gift

2018

https://hdl.handle.net/10539/26890

Khangamwa, Gift (2018) Knowledge extraction in population health datasets: an exploratory data mining approach, University of the Witwatersrand, Johannesburg, https://hdl.handle.net/10539/26890

Downloaded from a DSpace repository. https://dspace.org/



FACULTY OF SCIENCE

Masters Thesis

Knowledge extraction in population health datasets: an exploratory data mining approach

---

Author:
Gift KHANGAMWA

Supervisor:
Prof. Clint VAN ALTEN
Dr. Pravesh RANCHOD

A thesis submitted in fulfillment of the requirements for the degree of Master of Computer Science

in the

School of Computer Science and Applied Mathematics

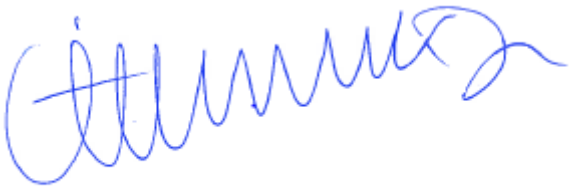
September 6, 2018

Declaration of Authorship

I, Gift KHANGAMWA, declare that this thesis titled, "Knowledge extraction in population health datasets: an exploratory data mining approach" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:



Date: **September 6, 2018**

Faculty of Science

Abstract

Faculty of Science

School of Computer Science and Applied Mathematics

Master of Computer Science

Knowledge extraction in population health datasets: an exploratory data mining approach

by Gift KHANGAMWA

There is a growing trend in the utilization of machine learning and data mining techniques for knowledge extraction in health datasets. In this study, we used machine learning methods for data exploration and model building and we built classifier models for anemia. Anemia is recognized as a crucial public health challenge that leads to poor health for mothers and infants and one of its main causes is malaria. We used a dataset from Malawi where the prevalence of these two health challenges of malaria and anemia remains high. We employed machine learning algorithms for the task of knowledge extraction on these demographic and health datasets for Malawi for the survey years 2004 and 2010. We followed the cross-industry standard process for data mining methodology to guide our study. The dataset was obtained, cleaned and prepared for experimentation. Unsupervised machine learning methods were used to understand the nature of the dataset and the natural groupings in it. On the other hand, supervised machine learning methods were used to build predictive models for anemia. Specifically, we used principal component analysis and clustering algorithms in our unsupervised machine learning experiments. Support vector machines and decision trees were used in the supervised machine learning experiments. Unsupervised ML methods revealed that there was no significant separation of clustering according to both malaria and anemia attributes. However, attributes such as age, economic status, health practices attributes and number of children a woman has, were clustered in significantly different ways, i.e., young and old women went to different clusters. Moreover, PCA results confirmed these findings. Supervised methods, on the other hand, revealed that anemia classifiers could be developed using SVM and DTs for the dataset. The best performing models attained accuracy of 86%, ROC area score of 86%, mean absolute error of 0.27, and kappa of 0.78, which was built using an SVM model having $C = 100$, $\gamma = 10^{-18}$. On the other hand, DTs produced the best model having accuracy 73%, ROC area score 74%, mean absolute error 0.36 and Kappa statistic of 0.449. In conclusion, we successfully built a good anemia classifier using SVM and also showed the relationship between important attributes in the classification of anemia.

Acknowledgements

First of all, I thank God almighty for his mercy and kindness to me as I struggle to find my way in this world. Secondly, I acknowledge all the help and moral support that I received from my colleagues in the masters and PhD computer lab. Thirdly, I wish to thank my supervisors for all the guidance and advice. I thank my main supervisor Prof van Alten in particular for his patience as I grappled with mastering machine learning theory and also for the financial assistance he rendered to me. I thank Dr Ranchod for the many insights that helped to shape this work. Finally, I thank members of my family for believing in me, praying for me, and for offering encouragement. In particular I thank my wife Blessings and our daughters Theodorah and Ruela, you guys are amazing!

Contents

Declaration of Authorship iii

Abstract v

Acknowledgements vii

Contents ix

List of Figures xiii

List of Tables xv

List of Abbreviations xvii

1 Introduction 1
- 1.1 Motivation . . . 1
- 1.2 Research Objectives . . . 3
  - 1.2.1 Specific objectives . . . 3
- 1.3 Description of work done . . . 3
- 1.4 Results obtained . . . 4
- 1.5 Thesis organization . . . 4

2 Background and Related work 5
- 2.1 Background . . . 5
  - 2.1.1 Data mining and machine learning . . . 5
  - 2.1.2 Statistical learning theory . . . 6
    - Empirical risk minimization (ERM) . . . 6
    - VC Dimension . . . 7
    - Structural risk minimisation (SRM) . . . 8
  - 2.1.3 Bayesian theory . . . 8
  - 2.1.4 Machine Learning algorithms of interest . . . 9
    - Unsupervised machine learning algorithms . . . 9
    - Supervised machine learning algorithms . . . 11
  - 2.1.5 Data mining methodology . . . 13
    - Business understanding . . . 14
    - Success criteria . . . 15
    - Measuring success of supervised ML . . . 15
    - Measuring success of unsupervised ML . . . 18
- 2.2 Related work . . . 19
  - 2.2.1 SVM . . . 20
    - Cancer prediction . . . 20
    - Diabetes prediction . . . 21
    - Malaria prediction . . . 22

Predicting flu outbreak or progression . . . 23
2.2.2 Decision trees and association rule mining . . . 24
Anemia prediction . . . 24
Injury narrative classification . . . 24
Salmonella surveillance . . . 24
Pharmaco-vigilance . . . 25
Predicting disease condition and life expectancy . . . 25
Predicting heart disease . . . 26
Predicting mortality or readmission . . . 26
2.2.3 Unsupervised learning . . . 27

3 Methods **29**
3.1 Study setting . . . 29
3.2 Data collection . . . 29
3.3 Experimental setup . . . 29
3.3.1 Hardware platforms . . . 30
3.3.2 Coding and software platforms . . . 30
3.3.3 Database design and development . . . 30
3.4 DM methodology . . . 31
3.5 Model building . . . 31
3.5.1 Grid Search . . . 31
3.5.2 Cross validation . . . 31

4 Dataset preparation **33**
4.1 Understanding the Dataset . . . 33
4.1.1 Data collection . . . 33
4.1.2 Data description . . . 34
Data dimensionality . . . 35
Themes . . . 35
4.1.3 Data exploration: preliminary ML experimental results . . . 35
4.2 Data preparation . . . 36
List of all features in full features dataset . . . 38
4.2.1 Controllable features dataset . . . 40
List of controllable features . . . 41
4.2.2 Dataset construction, integration and format . . . 42
4.3 Analysis of the data preparation process . . . 42
4.4 Conclusion . . . 43

5 Clustering **45**
5.1 Partition relocation clustering . . . 46
5.1.1 k-means algorithm . . . 46
Expectation maximization (EM) . . . 47
5.2 Principal component analysis (PCA) . . . 48
5.3 Self-organising maps (SOM) . . . 49
5.4 Methods . . . 49
5.4.1 Datasets used . . . 49
5.4.2 Cluster number identification . . . 50
5.4.3 Clustering patterns . . . 50
5.5 Results . . . 51
5.5.1 Number of clusters identified . . . 51
PCA . . . 51

SOM . . . 53
EM and k-means . . . 54
5.5.2 Clustering patterns . . . 54
SOM . . . 54
EM and k-means . . . 56
Clustering of anemia . . . 58
5.5.3 A comparison of the clustering algorithms . . . 59
5.6 Discussion . . . 60
5.6.1 Cluster size and patterns . . . 60
5.6.2 Principal components . . . 60

6 Support Vector Machines (SVM) 63
6.1 Support vector machines (SVM) . . . 64
6.1.1 Linearly separable SVM . . . 65
6.1.2 Non linearly separable SVM . . . 66
Kernel machines . . . 67
6.2 Methods . . . 69
6.2.1 Datasets used . . . 69
6.2.2 Approach . . . 69
6.2.3 Model development . . . 70
6.2.4 Model performance assessment . . . 70
6.2.5 Choosing most predictive features . . . 70
6.3 Results . . . 71
6.3.1 Model parameters . . . 71
6.3.2 Model building results . . . 71
Models for balanced full dataset . . . 72
Models on controllable features datasets . . . 73
Worst performing models . . . 74
6.3.3 Most predictive features based on Linear SVM weights . . . 74
6.4 Discussion . . . 77
6.4.1 Performance comparison with literature . . . 77
6.4.2 Anemia prediction . . . 77
6.4.3 Best predictors for anemia from Linear SVM . . . 78

7 Decision Trees 81
7.1 Defining decision trees . . . 81
7.1.1 Pros and cons of decision trees . . . 83
Pros . . . 83
Cons . . . 84
Entropy . . . 84
Information gain . . . 85
Gini index . . . 85
7.1.2 Decision tree algorithms . . . 86
ID3 algorithm . . . 86
C4.5 and variants . . . 86
CART . . . 87
7.2 Methods . . . 87
7.2.1 Datasets used . . . 87
7.2.2 Building decision tree classifiers . . . 87
7.2.3 Assessment of decision tree performance . . . 88
7.3 Results . . . 88

7.3.1 Summarized decision tree classifier results . . . 88
7.3.2 Analysis of good models . . . 89
Full features dataset decision trees . . . 89
Controllable features dataset decision trees . . . 90
Full features dataset - no mild anemia cases . . . 92
Controllable features - no mild anemia cases . . . 92
7.3.3 Detailed model performance assessment . . . 93
Full features dataset decision trees . . . 93
Controllable features dataset decision trees . . . 93
Full features dataset: no mild anemia cases . . . 94
Controllable features: no mild anemia cases . . . 95
Best decision tree models . . . 95
7.4 Discussion . . . 96
7.4.1 Decision tree performance . . . 96
7.4.2 Features identified . . . 96
7.4.3 Comparison of features identified by DTs with SVM results . . . 98
7.4.4 Conclusion . . . 99

8 Conclusion and future work **101**
8.1 Data preparation experience and proposed approach . . . 101
8.2 Anemia classifiers . . . 102
8.3 From key features to themes . . . 102
8.4 Some key findings . . . 103

Bibliography **105**

List of Figures

2.1 *This figure shows the CRISP-DM process model source (IBM, 2011). The rounded rectangles are the six phases of the model, while the arrows and their directions show how one can progress in the process model from one phase to another. The arrows indicate the direction of movement in the phases. Iterations between the phases are shown by reverse arrows in some phases based on their co-dependence i.e. business understanding and data understanding.* . . 13

4.1 *In this figure a summary of the information in the anemia attribute* $v457$ *is shown. The image was taken from the Weka data preprocessing tab. In the dataset information on anemia was found in instances where the survey participants got selected and consented to anemia testing. In this case* 31% *of the instances had data in this attribute. Therefore supervised experiments which required malaria related attributes for making predictions had* $10,660$ *instances for both training and testing.* 34

4.2 *Data pre-processing steps* 37

5.1 *Comparative plot of AIC and BIC for various cluster sizes. Smaller values for both measures are indicative of a good cluster size.* $x-$*axis are different cluster sizes in range* $[4, 20]$*. On the other hand AIC and BIC are on* $y-$*axis.* 54

5.2 *Clusters generated using SOM where the different colored blocks represent different clusters.* 54

5.3 *Current age of respondent.* 55

5.4 *Education levels are shown here,* 0 *is no education,* 1 *is primary,* 2 *is secondary and* 3 *is higher education. The various education levels did not cluster in any distinguishable pattern, they spread over all clusters.* 55

5.5 *Wealth index attribute ranging from* 1 *poorest to* 5 *richest. Likewise, this attribute did not have any unique clustering pattern.* 55

5.6 *Attribute v461 having information on whether respondent slept under bednet last night.* 0 *is no,* 1 *is yes and* 9 *is missing information.* 56

5.7 *Attribute v201 having information on the number of children the respondent has ever had.* 56

6.1 *This figure visually depicts the ideal linear separable SVM in 2D space. The separating hyperplane and decision boundary are clearly depicted by the middle line. Data points* x_1 *and* x_2 *are the support vectors for the two classes.* w *is the weight vector perpendicular to the decision boundary. Source* `http://www.precision-crop-protection.uni-bonn.de/gk_research/project_3_06/image_3.jpg` 65

List of Tables

2.1 *Confusion Matrix showing the relationship between known class values and predicted class values from a supervised ML algorithm during testing.* 16

4.1 *MDHS dataset used was for the* 2004 *and* 2010 *survey years. After completing the data cleaning exercise we had a dataset of* 10,039 *cases and* 133 *attributes, which we used in the experiments.* 37

5.1 *Clusters generated by the SOM algorithm on the full features dataset.* 53
5.2 *Analysis of cluster membership for an EM* 4 *cluster scenario.* 82% *of the data points are in cluster* 1 *with the remaining* 20% *spread over the other* 3 *clusters.* 57
5.3 *Analysis of cluster membership for an K-Means* 4 *cluster scenario.* 52% *of the data points are in cluster* 0 *while* 34% *and* 12% *are in clusters* 2 *and* 1 *respectively.* 57
5.4 *Analysis of cluster membership for an EM* 5 *cluster scenario.* 60% *of the data points are in cluster* 3*. The other large cluster is cluster number* 1 *with* 27% *of the data points.* 57
5.5 *Analysis of cluster membership for an K-Means* 5 *cluster scenario. This scenario has the largest cluster with* 30% *of the data points in cluster* 3 *followed by clusters* 0 *and* 1 *with* 29% *and* 27% *of the data points respectively.* 58
5.6 *Analysis of cluster membership for an EM* 6 *cluster scenario.* 58
5.7 *Analysis of cluster membership for an K-Means* 6 *cluster scenario. This scenario has smaller clusters compared with previous case for* 5 *clusters. Here we have* 28%*,* 26%*, and* 24% *of the data points in clusters* 1*,* 5 *and* 4 *respectively.* 58
5.8 *Distribution of anemia level cases in EM* 4 *cluster scenario. Cluster number* 1 *is the largest cluster and has most of the data points for all levels of anemic as well as non-anemic. Distribution of anemia level cases in K-Means* 4 *cluster scenario. In this scenario clusters,* 0 *and* 2 *are dominant having most of the cases clustered in them. However, there is no clear separation of the various anemia cases into specific clusters.* 59
5.9 *Distribution of anemia level cases in EM* 6 *cluster scenario. Distribution of anemia level cases in K-Means* 6 *cluster scenario.* 59

6.1 *Datasets and sub-datasets used in the study.* 69
6.2 *Linear kernel performance on the balanced full features dataset.* 72
6.3 *Polynomial kernel performance on the balanced full features dataset.* 72
6.4 *Sigmoid kernel performance on the balanced full features dataset.* 72
6.5 *Performance of the polynomial kernel on the balanced controllable features dataset.* 73
6.6 *Performance of the sigmoid kernel on the balanced controllable features dataset.* 73
6.7 *Performance of the RBF model on the controllable features without mild anemia cases.* 74

6.8 *Polynomial performance on the balanced full features dataset.* 74
6.9 *Top 20 best positive predictors for anemic.* 75
6.10 *Top 20 best predictors for non anemic.* 76

7.1 *Summary of results for the DTs on the full features dataset and controllable features dataset, that were conducted in both Weka and Python.* 88
7.2 *Summary of results for the DTs without mild anemic cases, that were conducted in both Weka and Python.* 89
7.3 *Results for Simple CART decision tree classifier on full features dataset.* . . . 94
7.4 *Results for Simple CART decision tree classifier on controllable features dataset.* 94
7.5 *Results for Simple CART decision tree classifier on full features dataset without mild anemia cases.* 94
7.6 *Results for Simple CART decision tree classifier on controllable features dataset.* 95
7.7 *Top 10 predictors from SVM and DTs results.* 98

List of Abbreviations

DM	**D**ata **M**ining
ML	**M**achine **L**earning
EM	**E**xpectation **M**aximisation
SVM	**S**upport **V**ector **M**achines
DT	**D**ecision **T**rees
SOM	**S**elf **O**rganising **M**aps
PCA	**P**rincipal **C**omponents **A**nalysis
SDG	**S**ustainable **Development G**oals
MDG	**M**illenium **D**evelopment **G**oals
SSA	**S**ub **S**aharan **A**frica
UN	**U**nited **N**ations
NSO	**N**ational **S**tatistical **O**ffice
DHS	**D**emographic **H**ealth **S**urvey
MDHS	**M**alawi **D**emographic **H**ealth **S**urvey

To Jim Daniel Mtengo and Dorothy my beloved parents!

1 Introduction

In this study, we used machine learning methods for data exploration and model building. We used principal component analysis, self-organizing maps, k-means and expectation maximization algorithms for unsupervised learning. After that, we built predictive models using support vector machines and decision trees for anemia, one of whose leading causes is malaria. Anemia is a recognized public health challenge that leads to poor health for mothers and infants. Hence, we used a demographic and health survey dataset from Malawi where the prevalence of these two health challenges of malaria and anemia remains high.

1.1 Motivation

We were motivated to research knowledge extraction from population health datasets using a data mining approach because according to the literature many available data sources can be analyzed more efficiently using this approach. In contemporary times it is a verifiable fact that the amounts of data currently being generated in various fields from business, marketing and advertising among other areas continue to grow. It is therefore not surprising that in the health sector as well data that is captured by health informatics systems, continues to grow exponentially (Herland, Khoshgoftaar, and Wald, 2014).

Once data collection is complete, if the methods employed in the process of extracting knowledge from these datasets are not well suited for the nature of the dataset, the task can be unnecessarily complicated and challenging. Sarlin (2012); Koua and Kraak (2004); Kaski and Kohonen (1996) showed that data mining techniques are very beneficial and appropriate for tracking international development goals and analyzing Demographic and Health Survey datasets.

Therefore, we hypothesized that data mining techniques could be used to extract additional information, and relationships between variables in the demographic and health survey (DHS) datasets, which can be useful for tracking and measuring sustainable development goals (SDG) progress, with a bias towards malaria.

SDG was born with the adoption of the 2030 agenda for sustainable development by the United Nations (UN) General Assembly, marking a new era of development targets for the world in the next fifteen years from 2016 to 2030, replacing the previous goals under the Millenium Development Goals. As was the case with the Millennium Development Goals, there are a variety of instruments that can be used to measure, and monitor progress towards the realization of these goals, one of which is public health surveillance (PHS), which generated the dataset that we used in this study.

Currently, Sub Saharan Africa lags behind the rest of the world, especially on health targets. In 2015 the UN produced the report that indicates that though under-five

mortality rate (U5MR) globally is at 43 per $1,000$ live births, it is at 86 per $1,000$ live births in SSA (United Nations, 2015a; United Nations, 2015b). Furthermore, even though the MDG achieved the targets on Malaria, there were still 214 million cases of malaria and $72,000$ malaria-related deaths in 2015 with 80% of the mortality happening in Africa (United Nations, 2015b). These statistics make this problem a relevant area to study in this region of the world, so that different ideas from an array of disciplines, can all make a positive contribution to tackle these challenges.

One of the challenges that face SSA is the problem of Malaria which is targeted by SDG goal number three. In Malawi for example, Malaria remains a tremendous public health concern, even though prevalence trends have been going down in recent years. Nevertheless, the country still experiences about 4 million episodes of Malaria per year which places a burden on families, communities and demands sizable government resources (Government of Malawi, 2014). Considering that the country has a population of about 15.8 million people, the Malaria burden can be readily appreciated. As a result, the state continues to spend resources to address the subsequent morbidity and mortality incidents from the disease.

The recognition of the health burden that this disease brings to the country has led to many interventions being implemented mainly targeting prevention and treatment. Some of the efforts include the development of a National Malaria Strategic plan and a National Malaria Control programme that aims to reduce the burden of this disease to a level of no public health significance through efforts such as proper diagnosis and treatment, vector control and monitoring and also health surveillance (Government of Malawi, 2014).

One of the tools for monitoring and surveillance of malaria is the collection of the Malawi Demographic and Health Survey dataset that we used in this study. The MDHS is a household survey which aside from collecting general health, demographic, socio-economic information, also captures information on the status of Malaria in the country using Malawi Malaria Indicator Survey instrument. The datasets contain information on women and children and use two main variables to measure the malaria prevalence rate for these groups. In the case of women, a measure of hemoglobin is used to ascertain anemia status, while for children information is collected on anemia status and whether a child had a fever in the previous two weeks before the survey (Rutstein and Rojas, 2013).

Related to malaria is the problem of anemia which is a condition where the human body has few red blood cells mainly due to deficiency of iron and folate minerals (Guyatt and Snow, 2001; Ayoya et al., 2006). This deficiency is brought about by two main factors, the first being having a diet that lacks these essential nutrients, as well as, exposure to infectious diseases such as Malaria, Hookworms and human immunodeficiency virus (HIV).

Within the malaria-endemic countries of southern Africa like Malawi, the significant cause of anemia condition is malaria. In a study on severe anemia among school kids Calis et al. (2008) found that the deficiencies in iron folate and sickle cell diseases are not the leading causes of severe anemia in Malawian children. Guyatt and Snow (2001) found that hemoglobin levels were low in areas where malaria is endemic in Sub Saharan Africa and higher elsewhere. The same authors further found the prevalence of severe anemia in women to be at 42 women per 1000. In a study on the prevalence of malaria and anemia in Malawi Rogerson et al. (2000) found a strong correlation between the two, whereby 48.3% of the women that were anemic,

were also infected with the malaria parasite Plasmodium Falciparum in a study that involved $4,764$ women at an antenatal clinic. Anemia mainly affects women and children; we, however, focus on women whose dataset is available for use in this study (Murphy and Breman, 2001).

DM and ML are capable of playing an important role in extracting knowledge from these datasets. For instance, a statement from one of the reports from CDC identified data analysis and knowledge extraction, as one of the significant challenges facing 21^{st} century public health surveillance. One of their proposals was that aside from using standard statistical approaches and tools, techniques that harness algorithms that require high computational power should be used to get as much information as possible from the various health datasets (Centers for Disease Control and Prevention, 2012).

This position strengthens the view that perhaps statistical approaches alone, may not be adequate to exploit the knowledge in DHS datasets fully. The nature of these datasets are very high dimensional having over $2,500$ variables on average per data file but also tend to suffer some deficiencies in the form of missing data, null values, and some data redundancies. This view implies, therefore, that perhaps automated machine learning algorithms can yield some valuable and useful knowledge which manual analysis can miss as earlier reiterated (Koua and Kraak, 2004).

1.2 Research Objectives

Our goal was to explore the use of data mining techniques to discover interesting patterns between variables in a demographic health survey dataset. In this regard, our focus was on questions that are related to sustainable development health goals on malaria.

1.2.1 Specific objectives

1. To clean and prepare the DHS dataset for DM experiments.
2. To investigate the potential of the data mining approach in isolating and identifying interesting, and useful patterns.
3. To apply unsupervised *ML* algorithms, such as *expectation maximization (EM)*, *KMeans*, *principal components analysis (PCA* and *self organising maps (SOM)*, for knowledge extraction (*KE*).
4. To develop predictive *DM* models for anemia using *decision trees (DT)* and *support vector machines (SVM)*.

1.3 Description of work done

The demographic and health survey datasets for Malawi for the survey years 2004 and 2010, were obtained, studied, cleaned, and prepared for ML experimentation. Through this process, we generated a final dataset that had fewer attributes and

fewer data instances from which we eliminated challenges of missing values and data redundancies.

We also generated a subset of the dataset that was made up of controllable features only. We defined the controllable features as features on which we can propose interventions. We selected such features from the final cleaned dataset. The selection involved a systematic process of reviewing each feature and asking ourselves the question of whether this feature represents an issue that can be resolved through intervention. If our response to this question was affirmative we selected the feature and added it to the subset of controllable features. Otherwise, the feature was not selected. Therefore, in the experimentation phase, we used both the full features dataset and the controllable features sub-dataset and compared model performance.

Our rationale was that, if we focus on controllable features they might provide guidance on what specific area malaria and anemia interventions need to target, e.g., maybe education or perhaps mosquito net usage. In this case, a predictive model may assist in highlighting an area that might be useful in combating the challenge.

Unsupervised machine learning methods were used to understand the nature of the dataset and the natural groupings in it. Specifically, we used principal components analysis and clustering algorithms in our unsupervised machine learning experiments. On the other hand, supervised machine learning methods were used to build predictive models for anemia. Anemia was used because of its correlation with malaria and also because it has been used before as a proxy variable to estimate malaria prevalence in line with the DHS survey protocols.

Finally, we attempted to make predictions at the personal level as opposed to the population level, which most researchers who tackled the related problem of malaria prediction did as reported in the literature see (Sharma et al., 2015).

1.4 Results obtained

Overall we obtained results that were acceptable and could be related to what is known in the literature on malaria and anemia.

Unsupervised ML results revealed that there was no significant separation of clustering according to both malaria and anemia attributes. However, attributes such as age, economic status, health practices attributes and number of children a woman has, were clustered in significantly different ways, e.g., young and old women went to different clusters. PCA results confirmed these findings through the principal components that we identified. The results obtained for supervised methods revealed that anemia classifiers could be developed using SVM and DTs for the dataset.

1.5 Thesis organization

The rest of the document has chapters as follows: chapter 2 provides background and related work, chapter 3 is on research methods, chapter 4 is on data preparation, chapter 5 is on clustering, chapter 6 is on support vector machines, chapter 7 is on decision trees, and after that, we conclude our discussion in the concluding chapter 8.

2 Background and Related work

2.1 Background

The fields of machine learning and data mining are vital in knowledge extraction. By definition, machine learning concerns the study and development of algorithms that can learn from data. While data mining on the other hand implements and utilizes these algorithms for automated learning from data stored in databases or other data storage areas.

Moreover, the exploratory nature of the data mining approach, is ideal for exploring health datasets because it combines techniques from computer science, statistics, mathematics, pattern recognition and artificial intelligence. This array of methods enhances the chances of success at knowledge discovery (Alpaydin, 2010).

2.1.1 Data mining and machine learning

In this section, we discuss some machine learning theories, techniques, and algorithms, that are relevant to our research. We also briefly touch on some of the areas that contribute to machine learning.

Machine learning encompasses an array of techniques and algorithms that allow for the development of software systems that are adaptable and capable of learning from data. On the other hand, as already reiterated, data mining is viewed as the implementation of various ML techniques and algorithms specifically for knowledge extraction in databases (Alpaydin, 2010).

Data mining has also been referred to as Knowledge Discovery in Databases (Fayyad, Piatetsky-Shapiro, and Smyth, 1996). It is therefore not surprising that many modern database management systems have built-in data mining capabilities.

However, there are also specific open source DM platforms, like the Waikato Environment for Knowledge Analysis in short WEKA, which implement many supervised and unsupervised machine learning algorithms (more details on `www.cs.waikato.ac.nz/ml/weka/`). Another platform is the SciKit-Learn a machine learning package, implemented in Python programming language for more information on the various algorithms (see `www.scikit-learn.org`), and (Pedregosa, Weiss, and Brucher, 2011). We used both of these platforms, in this study for data preparation and experimentation.

In the next subsections, we provide a discussion on some of the theoretical foundations for some of the machine learning algorithms of interest in this study. We briefly introduce the statistical learning theory as well as Bayesian theory in sub-sections 2.1.2 and 2.1.3 respectively, after that sub-section 2.1.4 introduces ML algorithms of interest and finally sub-section 2.1.5 discusses the DM methodology of choice.

2.1.2 Statistical learning theory

In this section we briefly discuss the fundamentals of the statistical learning theory on which SVM algorithms are based. Vapnik (1999) states that the goal of statistical learning theory is to minimize the expected loss or errors when training a learning machine using some observed data. The theory encompasses various learning tasks such as pattern recognition, classification or density estimation. The overall goal of learning is to ensure that a model or pattern is learned and that this model or pattern can be used to test new datasets and yield good results.

Vapnik (1999) describes the supervised learning problem using labeled training data, as being made up of three building blocks, which are a *Generator* of random vectors $(\mathbf{x})$ that are drawn from an unknown probability distribution $P(\mathbf{x})$, a *Supervisor* or trusted source that generates an output value (y) drawn from an unknown distribution $P(y|\mathbf{x})$ for every vector $(\mathbf{x})$ and finally a set of *learning machines* which are represented as functions $f(\mathbf{x}, \alpha)$, which learn from the labelled data and have to make predictions on some unlabelled test data.

In this case, the role of the learning machines is to provide an estimate of y, for $\alpha \in \Lambda$, where α are a set of parameters selected from Λ, whose choice leads to the identification of a trained learning machine (Vapnik, 1999; C, 1998). This formulation is the supervised learning problem which, as expected, relies on the availability of some labeled training data with n data points as shown below such that for each training data point $\mathbf{x_i}$ there is a corresponding label or class value y_i, i.e.;

$$(x_1, y_1), (x_2, y_2), \cdots (x_n, y_n) \tag{2.1}$$

Empirical risk minimization (ERM)

The training datasets (x_i, y_i) for $i \in n$ are drawn from an unknown joint probability distribution $P(x, y) = P(x)P(x|y)$. Therefore, the challenge in the learning problem is to select the best performing learning machine $f(\mathbf{x}, \alpha)$ based on how it minimizes the expected risk of loss or errors. The risk here is a scenario where the learning machine's output is different from the supervisor's $(y|\mathbf{x})$ output and is given by loss functions for various learning problems, i.e., regression, pattern recognition or density estimation as will be discussed later, for more details see (Vapnik, 1999; C, 1998). Hence, the expected value of the loss is given by the risk function formula below where the loss function is presented as L in the formulation;

$$R(\alpha) = \int L(y, f(\mathbf{x}, \alpha)) dP(\mathbf{x}, y) \tag{2.2}$$

The function R is the actual risk which is a probabilistic measure of how learning machines perform during testing with some realistic dataset. Therefore, the goal is to pick the best performing learning machine function $f(\mathbf{x}, \alpha)$ by selecting the machine with lowest error or risk of loss, specifically a learning machine that minimizes the risk $R(\alpha)$. This risk is as yet an unknown measure, since the probability distribution is unknown at this stage, and can only be estimated empirically through calculation of the training error for n training data points to compute $R_{emp}(\alpha)$, which we define in a subsequent section (Vapnik, 1999).

The loss function is different for various machine learning problems such as pattern recognition, regression or density estimation. However, Vapnik (1999) demonstrated how all the other risk and loss formulas are linked, and dependent on the risk function formula which is given above in equation 2.2. Therefore, in the case of classification, which is a type of pattern recognition for a scenario, where the supervisor y takes on two values only, i.e., $0, 1$ the loss function $L()$ for this classification scenario is given as;

$$L(y, f(\mathbf{x}, \alpha)) = \begin{cases} 0, & \text{if } y = f(\mathbf{x}, \alpha) \\ 1, & \text{if } y \neq f(\mathbf{x}, \alpha) \end{cases} \tag{2.3}$$

On the other hand for a regression case where the supervisor y takes on a real number value i.e. $y \in \mathbb{R}$, the loss function $L()$ is given as;

$$L(y, f(\mathbf{x}, \alpha)) = (y - f(\mathbf{x}, \alpha))^2 \tag{2.4}$$

In a generalization of this risk, therefore, given z as a pair (x, y) with a loss function $Q(z, \alpha)$, where z is independently identically distributed (iid) the general empirical risk R_{emp} is;

$$R_{emp}(\alpha) = \frac{1}{n} \sum_{i=1}^{n} Q(z, \alpha) \tag{2.5}$$

For $i = 1 \cdots n$ data points.

Where $Q(z, \alpha)$, can now be replaced by any of the loss functions for the various learning problems such as loss functions 2.3 and 2.4 given above.

In empirical risk minimization, therefore, the actual risk function 2.2 is estimated using the available training dataset, and is estimated using empirical risk function 2.5. Therefore using training error the actual error which is unknown is estimated. It is important to point out that, in cases of large n the (ERM) is a reliable estimate of risk. However, when n is small, there is a problem, and there is a need for other means of ensuring the generalization ability of learning machines is achieved.

VC Dimension

The Vapnik-Chervonenkis dimension, otherwise referred to as the VC dimension is a measure of the capacity of a learning machine, to separate or shatter input vectors. This measure is useful in balancing bias and variance of a learning machine so that one can develop an ML algorithm that can learn general patterns in the data, without having that model being tied too much to the training dataset a situation called overfitting. Therefore, ML algorithms with low VC dimension are preferred because they are simpler and are capable of generalizing. Vapnik (1999) states that, the VC dimension of a set of hyperplanes with n dimensional input data space is $n + 1$.

However, for hyperplanes that have a wide separating margin, the VC dimension can be less than $n + 1$ (Vapnik, 1999), which is an important result and the goal for separation of data in SVM. The VC dimension is used together with the ERM in the formulation of the SRM which will be discussed next.

Structural risk minimisation (SRM)

The structural risk minimization (SRM), minimizes the expected risk for both ERM and VC dimension. In other words, the goal is to have ML models that yield low error rates in training but also have a lower VC dimension. Therefore the concept of structural risk minimization establishes a link between training and testing errors, i.e., expected risk and empirical risk. The structural risk minimization concept can be understood as a bound that is placed linking capacity of a learning machine and their performance (C, 1998; Vapnik, 1999). This bound is directly linked to training error, as well as testing errors as linked in the equation below;

$$R(\alpha) \leq R_{emp}(\alpha) + \sqrt{\frac{h(\log(\frac{2n}{h})) - \log(\frac{n}{4})}{n}} \tag{2.6}$$

Where $R(\alpha)$ is the expected risk as defined earlier, while R_{emp} is the empirical risk (training error), h is the Vapnik-Chervonenkis dimension, and n is the number of training data points. The goal in supervised learning, therefore, is to pick from a set of learning machines, one machine that minimizes the actual risk by minimizing the empirical risk and also choosing an appropriate value for the VC dimension.

2.1.3 Bayesian theory

Bayesian theory is the second topic that we discuss briefly. The theory is very significant in machine learning, where patterns are supposed to be discovered automatically in data. The theory relies on probability theory and forms an essential component in the ML discipline.

According to (Cox, 1946) probability theory has two main branches namely frequentist and reasonable expectation schools of thought. The frequentist theory measures probability as a computed proportional outcome of a large number of similar experiments, i.e., throwing of dice, picking a card for a deck, or coin toss outcome. On the other hand, Cox (1946) describes the reasonable expectation as an expected likelihood in a particular experiment. For example, given a fair two-sided coin, it is a reasonable expectation that in a coin toss event each side has $\frac{1}{2}$ chance of being face up, on the basis the coin has two sides only. This expectation in a coin toss in this scenario is not based on any experimental evidence, but rather the rational analysis based on the nature of a coin.

Bayesian theory, which falls under the expected likelihood school of thought, connects three probability factors. These are posterior probability which is unknown and what we are interested in finding, a prior probability which is learned from the data which is readily available, as well as, a class conditional probability, which likewise can be estimated from data. The theory is formulated as follows;

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{normalizing factor}}$$

In a general case, where there are k classes $C_1 \cdots C_k$, from the dataset X, Bayes rule is expressed as;

$$P(C_i|x) = \frac{P(x|C_i)P(C_i)}{P(x)}$$

Bayesian theory of statistics is dependent upon conditional probability, which states that the probability of event A happening, given conditional event B, is given by;

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

This theory has led to the emergence of several machine learning algorithms, where high computational power makes it possible to compute estimates of the prior, that are otherwise cumbersome if done manually.

2.1.4 Machine Learning algorithms of interest

In this section, our focus of discussion is to review the various machine learning algorithms used in studies that are similar to ours. Therefore, we will focus on the specific details about the implementation of the learning machines $f(\mathbf{x}, \alpha)$ mentioned in section 2.1.2, which these similar studies employed. We will further, discuss the machine learning algorithms according to their category of being either supervised ML or unsupervised ML. Our goal is neither to exhaustively review all machine learning algorithms nor to discuss the algorithms in-depth. However, we want to provide a brief introduction of the algorithms that are relevant for this study. We will present additional details on these ML algorithms in relevant chapters where we have utilized those particular algorithms for model building.

We will, therefore, discuss k-means, Expectation Maximization, Decision Trees and Support Vector Machines (SVM) algorithms. In the discussions below, we rely heavily on the work by Alpaydin (2010), who provided an excellent analysis of these algorithms.

Unsupervised machine learning algorithms

Unsupervised machine learning algorithms are used to learn from data, without having any prior information about the classes to which the data points belong during both training and testing, their task, however, is to estimate this information from the dataset itself. This process involves estimation of the sample distribution parameters α either for identifying clusters in cluster-based methods or for estimating cluster boundaries for discriminant based approaches. Other algorithms in the category just aim at simplifying the visualization of high dimensional data to reveal relationships in smaller dimensions which are easier to comprehend.

These methods are therefore very suitable for both data exploration and data understanding which are preliminary data mining tasks on most data mining process models. The results of these algorithms can also be used to inform supervised ML experiments and indeed to confirm findings from supervised ML experiments.

Unsupervised machine learning algorithms are also useful in a scenario where the researcher does not have much information about the dataset and wants to learn its natural groupings before performing supervised machine learning tasks. The results

from such methods can thus guiding a researcher on what patterns to concentrate on in the dataset. This capacity implies that these methods can be beneficial both in data preparation, e.g., for reducing dimensionality in high dimensional datasets, as well in the model building which can be subjected to verification tests using various supervised methods.

Popular among the unsupervised ML algorithms are two main clustering algorithms namely k-means and expectation maximization. Other algorithms include, Self Organising Maps and Principal Components Analysis also referred to as PCA.

k-means can automatically identify k natural groups or clusters in the dataset through a series of iterations. The value k is an integer that represents the number of clusters that the user expects in the dataset. This integer is specified a priori to the algorithm and leads to set up of k cluster centers, around which the clusters form. These methods generally will create a set of k data clusters from N data points for a dataset $D = \{X^i; i = 1, \cdots, m\}$ such that each data point belongs to at least one cluster C_i for some $i = [1, \cdots, k]$.

Consequently, in each cluster a mean μ_i is computed from the data points $d \in D$ allocated to that particular cluster, this mean vector represents the center of that particular cluster. At every iteration, this mean μ_i is recomputed, using the distance to all cluster members. After the computation, the cluster center moves to the central most position in the cluster relative to all the data points in that cluster. The iteration continues until convergence, which is a case where data points do not move significantly between clusters (Alpaydin, 2010).

Data points are allocated to a cluster if they happen to be closer to the cluster center of that particular cluster than to all other cluster centers. The closeness of a data point to a cluster center is based on a measure of distance from a given data point to all available cluster centers in the dataset.

k-means allocates data points to hard clusters, meaning that a data point or instance either belongs to a particular cluster or not. On the other hand, the *EM* algorithm will provide a probability for every data point's membership to all the k clusters such that one can assess the magnitude of the probability of a data point's membership to a particular cluster. There are also many variations of this general *k-means* algorithm that addresses various issues such as how best to select the k cluster centers during the initialization step to avoid converging to a local maximum (Jain, 2010).

Expectation maximization *(EM)* algorithm is used for maximum likelihood estimation and is another unsupervised method which performs clustering.

A data instance's membership to a particular cluster is made by assigning it to the cluster to which that particular data instance has the highest probability of membership. Hence, once the algorithm converges, one can query the algorithm through a predict function with any data point, and a Gaussian model of the cluster to which that data point belongs is given. Furthermore, the algorithm also provides the probabilities of that data instance belonging to all the other clusters. These probabilities are useful information and reveal relationships between clusters and their members.

Supervised machine learning algorithms

Supervised methods use training datasets, which specify classes, to which all the data points belong. These methods require training datasets that have the form $D = \{X_i, t_i\}$; for $i = [1, \cdots, m]$ where X_i are features, and t_i are targets or classes, such that for each data point i a corresponding class label t_i, to which those inputs belong is given. The classes can be viewed as emanating from a supervisor $P(y|\mathbf{x})$, which gives out the class to which every **x** vector belongs as per the formulation of the learning problem by Vapnik, which was mentioned earlier. On the other hand, the output of our supervised machine learning algorithm is $f(\mathbf{x}, \alpha)$. Therefore, errors occur when the two functions yield different values, and the goal is to make sure that these errors are minimized, refer subsection 2.1.2, where risk minimization was discussed.

The task for supervised ML algorithms is, therefore, to learn these patterns in the training dataset, and to use that learning for later classification or prediction tasks, on some unseen datasets for which classes are not provided. One of the key principles in these approaches is avoidance of over-training which leads to models overfitting the dataset. This is an important principle to ensure that models can generalize enough to be able to classify different data with acceptable error margins.

There are two main categories of supervised methods, namely classification and regression methods. Classification methods will test some hypothesis $h(x)$, to check whether a given data point belongs to a given class i or otherwise as illustrated below.

$$h(x) = \begin{cases} 1, & \text{if x is classified as belonging to class i} \\ 0, & \text{if x is classified otherwise} \end{cases}$$

On the other hand regression methods yield a real number value y as shown below.

$$h(x) = y \text{ , for } y \in \mathbb{R}$$

Therefore, this real number y will be used to make class allocations using some thresholding mechanism, i.e., if $(y >= threshold)$ classify as 1, otherwise classify as 0 for some scenario with binary classes 1 and 0. In a non-binary scenario, where there are more classes, one can use a one versus all approach, where each class will be compared against all the other classes in subsequent steps.

To avoid overtraining, the datasets usually are split into training, validation and testing datasets. The training datasets are usually about $\frac{3}{4}$ of the dataset D. The remaining dataset is then split between testing and validation datasets. The other approach is k-fold cross-validation, where the dataset D is split k times. The $k-1$ splits are consequently used for training, and the remaining k^{th} split is used for testing. These efforts are necessary to avoid over-training as stated earlier and to ensure that an ML model can generalize its learning and be employed for making predictions on other new datasets post model building.

One approach that we will not use in this research, but that is worthy of mention is reinforcement learning. Reinforcement learning makes use of an intelligent agent, which is placed in an environment at some start state with a target of arriving at a goal state. The agent must then figure out how to arrive at a goal state from the initial

start state, through an exploratory process, with feedback in the form of rewards or punishment being awarded in the process, for good and bad moves respectively. The agent receives a reward or punishment based on how their immediate action contributes to the target of getting to the goal state. Through the process of getting either rewards or punishments the agent eventually figures out a solution, that can take them to the goal state with maximum rewards (Alpaydin, 2010).

Decision trees are a "hierarchical data structure implementing a divide and conquer strategy" (Alpaydin, 2010). These decision trees are capable of performing regression, classification as well as association rule mining. Furthermore, these algorithms can be used in the data preparation stage for feature extraction, in scenarios where the data is too complex such that features are either few and not easily distinguishable or too many.

In order to build decision trees the idea is to, first of all, identify the best feature on which to split the dataset, following that, use this feature as a root from which branches will emerge based on the number of values the feature takes, i.e., two branches for a binary feature or otherwise. The decision to split the tree on a particular feature is based on its measure of information contribution, using a measure of information impurity or entropy. There are several measures used to assess a feature's information value. The methods are $i)$ entropy, $ii)$ Gini index and $iii)$ misclassification error.

In our case, we used the $J48$ decision tree algorithm. This algorithm is based on the $C4.5$ decision tree algorithm, which makes use of the entropy as a measure of information impurity indicating the importance of any feature in the dataset. The other decision tree algorithm that we used is the CART, which is a regression classification decision tree, which uses the Gini impurity measure.

Both of these algorithms can be used to generate a set of rules, by analysis of the algorithm's graphical output. Finally, it is necessary to state that these decision trees have also been selected because of their excellent performance for association rule mining, which is essential in the current study.

Support vector machines (SVM) are in the class of supervised machine learning algorithms and are based on linear discriminant method. SVM are used for classification and can also be used for regression. The SVM work as follows: given a dataset $X = \{\mathbf{x}^t, y^t\}$ where y are the classes, the algorithm identifies a maximum separating hyperplane for the classes.

$$y^t = \begin{cases} +1, & \text{if } \mathbf{x}^t \in C_1 \\ -1, & \text{if } \mathbf{x}^t \in C_2 \end{cases}$$

The major aim of this algorithm is to ensure that there is a good distance between members of the two classes C_1 and C_2. The algorithm aims to maximize the distance that separates the classes to reduce the risk of class C_1 instances being classified as C_2. The algorithm makes use of border values which are called support vectors because they support the decision boundaries margins for the two classes, details are in chapter 6.

2.1.5 Data mining methodology

Methodologies in computer science are common and necessary for providing a common language and a set of tools and techniques to be able to both do the work and communicate effectively. We guided this project using the cross-industry process model for data mining (CRISP-DM) methodology. In the CRISP-DM, below each phase is a set of generic tasks, beneath which are specific sub-tasks and at the bottom of the hierarchy, there are processes, which are the atomic tasks in the methodology where all the work is done(Wirth and Hipp, 2000; IBM, 2011). These tasks are articulated at four levels of abstraction from the general to the specific. The general high-level phases are business understanding, data understanding, data preparation, modeling, evaluation, and deployment (Chapman et al., 2000).

Each particular phase has a milestone that becomes input to the next phase, and there is the allowance for iteration of phases until the results are satisfactory to the researcher. The process model is suited to both predictive data mining where goals are clear, as well as to exploratory data mining projects, where one is trying to learn from the natural patterns and groupings of the data to determine what knowledge is hidden in the datasets (IBM, 2011).

This methodology has a well-articulated rationale for the phases as well as the required inputs and expected outputs of the phases. The nature of this methodology is such that the milestone of a phase becomes input to the subsequent phase while allowing for looping between the stages as depicted in the figure 2.1.

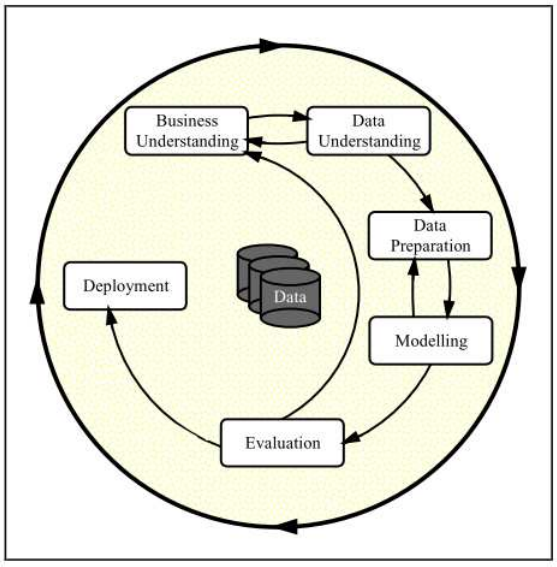


FIGURE 2.1: *This figure shows the CRISP-DM process model source (IBM, 2011). The rounded rectangles are the six phases of the model, while the arrows and their directions show how one can progress in the process model from one phase to another. The arrows indicate the direction of movement in the phases. Iterations between the phases are shown by reverse arrows in some phases based on their co-dependence i.e. business understanding and data understanding.*

The life cycle depicts the top level phases only, however, within each phase there are generic tasks that need to be done in that particular phase. Likewise, in each generic task, there are a set of specific low-level actions, i.e., data collection, review of model performance, that are carried out and expected outputs, i.e., data quality report, data description report e.t.c. (Wirth and Hipp, 2000; IBM, 2011) which are generated for use in subsequent phases.

Other related data mining methodologies which we could have used are Knowledge Discovery in Databases and Sample Explore Modify Model Assess. These methodologies were compared to CRISP-DM by Azevedo and Santos (2008), who concluded that the methods achieve the same results albeit in different phases in some cases.

KDD has the following phases: pre-KDD, selection, pre-processing, transformation, data mining, interpretation, and evaluation, finally post-KDD. On the other hand, SEMMA has phases as per the letters in its acronym (Azevedo and Santos, 2008). A detailed analysis of these methodologies confirms that indeed the methodologies are similar. The only difference that exists is that CRISP-DM is explicit on the final deployment phase which is a top-level phase, while the other methodologies either incorporate deployment in another phase, i.e., post-KDD in KDD or omit it all together in SEMMA. We, therefore, selected CRISP-DM because of its extensive use and acceptability in literature and also as a matter of preference.

Business understanding

The business understanding phase is the first phase in the CRISP-DM process model that seeks to put a data mining project in context, and also to assess how that project will impact on an organization and its business. Some of the major considerations in this stage include $i)$ what specific questions can DM answer that other approaches cannot, and $ii)$ what business value will be gained from answering those questions for an organization. These considerations are the necessary ingredients for the formulation of the overall data mining project goals.

As can be seen in the figure 2.1, business understanding goes hand in hand with data understanding, in an iterative fashion until the researcher is satisfied to move to the next phase. Moreover, after the evaluation phase which is phase number five, if one is not satisfied with the results that one has obtained, a researcher can return to the business understanding phase so that data mining goals can be re-articulated and changes to the scope of the project be made accordingly.

The first task of the business understanding phase is to justify the use of DM approach and identify DM goals. Our primary objective in this project as stated in our study aims was to undertake knowledge extraction on a demographic and health dataset for Malawi, with a focus on malaria and anemia.

Knowledge in machine learning refers to various machine learning models, which are learned from data but can be used to make predictions on other novel or unseen data such as testing datasets or real-life datasets encountered after model deployment. The various models that we planned on identifying are: clusters, patterns (Barakat, Bradley, and Barakat, 2010; Mathias et al., 2013), associations (Brossette et al., 1998), decision trees and rules (Brossette et al., 1998; Barakat, Bradley, and Barakat, 2010; Mathias et al., 2013), using a set of machine learning algorithms mentioned in chapter 2. These models, once identified, can be used to explain the data better, and to aid the understanding of malaria and anemia attributes and their relationship to other attributes in the dataset used.

The second goal was to select interesting and useful knowledge as it pertains to malaria and anemia. Interesting knowledge or models are those that have strong predictive power in line with the chosen measures of performance such as accuracy, sensitivity, and specificity for the specific algorithm. Secondly, useful models are those that reveal something new regarding either information about the variables or new information concerning the association between the variables.

Success criteria

Once the data mining models are built, the next question that arises is, how can we know if we are learning useful knowledge. Therefore, to know whether we are extracting useful information we used various measures for the performance of the algorithms. We know that we are learning useful knowledge, when our models can achieve acceptable levels of accuracy with low error measures, in prediction on the testing dataset. In addition to this, we compared our model's performance with benchmarks for the specific algorithms as reported in the literature. Moreover, we used a set of common performance metrics to measure the performance of our models to determine success.

Since the learning goals of supervised machine learning algorithms and unsupervised machine learning algorithms are different, measures of success used for these categories were also different. We, therefore, used measures such as accuracy, sensitivity, specificity, precision, receiver operating characteristic (ROC) among other measures or errors such as relative mean square error, for classification algorithms. On the other hand, Akaike information criterion (AIC) and Bayesian information criterion (BIC) were used in assessing the performance of clustering algorithms. We proceed, to briefly discuss these success metrics in the forthcoming subsections.

Measuring success of supervised ML

In order to assess success of supervised ML algorithms, measures such as true positives (tp), false positives (fp), true negatives (tn) and false negatives (fn) are computed (Barakat, Bradley, and Barakat, 2010; Palaniappan and Awang, 2008; Cohen et al., 2004). Table 2.1 provides a pictorial view of the relationships between these measures, and how they arise. As we might recall, the difference is based on what the learned models predict, what we have been referring to as a learning machine $f(\mathbf{x}, \alpha)$, and the actual class membership or supervisor output y, for data instances in labeled testing datasets. Specifically, these measures assess how predicted values from the ML models $f(\mathbf{x}, \alpha)$ represented here by $\hat{y}_i$, compare to, y_i values which are known class allocations for the data-points $i \in n$, where n is the number of data points in our dataset D.

For all supervised methods, there are two types of errors namely, training errors and testing errors, which we discussed in subsection 2.1.2 in chapter 2. Training errors are useful and assist in fine-tuning the model's parameters α, however testing errors are the ones which give a better picture of how effective a model will be once deployed. Therefore, our measures here focus on testing errors, where one uses a set of data that has not been used in the training stage to assess the trained model's predictive power and accuracy.

Specifically, all cases where $\hat{y}_i \neq y_i$ are errors, and our goal is to ensure that we pick the best parameters α, for our ML models to minimize these errors. Hence, for the supervised learning algorithms, we used measures of accuracy, sensitivity, specificity, and ROC just to mention a few. We now begin to discuss the relevant metrics of true positives (tp), true negatives (tn), false positives (fp) and false negatives (fn), which are used to compute the other performance measures. We assume a binary case, where we have positive and negative classes, represented by either 1 and 0 or 1 and -1 respectively.

TABLE 2.1: *Confusion Matrix showing the relationship between known class values and predicted class values from a supervised ML algorithm during testing.*

		Predicted Class	
		Positive	**Negative**
Actual Class	**Positive**	*tp*	*fn*
	Negative	*fp*	*tn*

tp is a count of how many predicted values match with true values for a positive class value i.e. $\sum_{i=1}^{n} 1(\hat{y}_i = y_i)$ for $y_i = 1$. On the other hand *tn* is an opposite measure that considers correctness of classifying negative class values i.e. $\sum_{i=1}^{n} 1(\hat{y}_i = y_i)$ for $y_i = 0$. *fp* and *fn* are measures of how the predicted value is an error by predicting the opposite value of a class value i.e. predicting a positive value when the true value is negative and vice versa.

These basic metrics are, hereafter, used to compute other performance measures such as classification accuracy, sensitivity, precision and specificity whose formulations are demonstrated below (Mathias et al., 2013).

Classification accuracy (CA)

Classification accuracy is a measure of model performance, which counts the number of data points that are correctly predicted by a model. Here, $\hat{y}_i$ are the predicted values, while y_i are the true values for $i \in n$, where n is the number of data points in the dataset.

$$accuracy = \sum_{i=1}^{n} 1(\hat{y}_i = y_i)$$

The classification accuracy is also expressed as a fraction of correctly classified instances and the total number of data instances.

$$accuracy = \frac{tp + tn}{tp + tn + fp + fn} \tag{2.7}$$

Classification accuracy has the challenge that, when datasets are skewed having, for example, more non-anemic cases than anemic as in our dataset, the accuracy of the dominant class may be misleading and inflate the accuracy score, just because the fewer class members are not as influential on the result.

Sensitivity

Sensitivity measures the performance of an ML model on the prediction of anemic class in our dataset. It is also referred to as recall or true positive rate.

$$sensitivity = \frac{tp}{tp + fn} \tag{2.8}$$

Specificity

Specificity is also referred to as true negative rate and is a measure of correctness in the prediction of the negative class values.

$$\text{Specificity} = \frac{tn}{tn + fp} \tag{2.9}$$

Precision

Precision is a ratio of true positives against all predicted positives in a dataset. This measure gives a weight of the predicted positives, against all values that the model considers as positive. If the model performs very well *fp* will be close to 0 and precision will be close to 1 implying that it is a good model.

$$\text{Precision} = \frac{tp}{tp + fp} \tag{2.10}$$

False positive rate (FPR)

The false positive rate is a measure of how a model predicts positive cases incorrectly, with all the negative cases in the dataset. We include this measure because it will be used for the computation and plotting of the ROC curves and the subsequent interpretation of the same.

$$\text{FPR} = \frac{fp}{fp + tn} \tag{2.11}$$

F

Finally F, also referred to as F1, as well as, F-score is given below. This measure is useful because it considers predictive ability not only regarding the positive case but also the performance on false positives.

$$\text{F1} = \frac{2 \times precision \times sensitivity}{precision + sensitivity} \tag{2.12}$$

Kappa

Kappa statistic, just like the ROC measure, is not affected by data skewness. The metric takes into account the performance of a random classifier that generates what is termed the expected accuracy.

$$\text{Kappa statistic} = \frac{\text{Observed accuracy} - \text{expected accuracy}}{1 - \text{expected accuracy}} \tag{2.13}$$

Where the observed accuracy is the output of an ML algorithm, and the expected accuracy is the accuracy one gets if they randomly classify all data point to one class. In a sense, the expected accuracy is based on the class proportions in the dataset.

Receiver operating characteristic (ROC)

Receiver operating characteristic (ROC) curve is a plot of sensitivity or *tp rate* against *fp rate* and is used to select acceptable levels of accuracy of a prediction algorithm. In addition to this, (Hamel, 2009) states that ROC curves are not susceptible to class skewness, which as stated before is a problem that manifests itself in classification accuracy.

The ROC is a measure of choice for applications that make predictions on health, where predicting positive cases correctly is as important just as well as predicting negative cases correctly (Cohen et al., 2004).

High sensitivity with low false positive rate is a preferred scenario which can be detected using a ROC curve.

Through a visual analysis of the ROC curve one can identify models that have a good performance, by analyzing the area under the curve in the plot, the larger the area, the better the model. The area under the ROC curve, called the AUC, is used as a measure of how models perform. By definition, the area is provided as a percentage of the rectangular area covered by the TP rate and FP rate plot.

Measures of error

Errors occur when there is a discrepancy between supervisor and predictor values i.e. $\hat{y}_i \neq y_i$. There are several error measures which we describe in the formulations below:

$$\text{Error rate} = \frac{fp + fn}{P + N}$$

Where P is all positive cases while N is all negative cases available in the dataset.

Root mean square error;

$$\text{RMSE} = \sqrt{\frac{(\hat{y}_1 - y_1)^2 + \cdots + (\hat{y}_n - y_n)^2}{n}}$$

Mean square error;

$$\text{MSE} = \frac{(\hat{y}_1 - y_1)^2 + \cdots + (\hat{y}_n - y_n)^2}{n}$$

Relative squared error;

$$\text{RSE} = \frac{(\hat{y}_1 - y_1)^2 + \cdots + (\hat{y}_n - y_n)^2}{(\overline{y}_1 - y_1)^2 + \cdots + (\overline{y}_n - y_n)^2}$$

Where, $\overline{y}_n = \frac{\sum_{i=1}^{n} \hat{y}_i - y_i)}{n}$, is the average prediction error of the model.

We, consequently, proceed to discuss model performance measures for unsupervised machine learning.

Measuring success of unsupervised ML

Unsupervised machine learning algorithms use datasets that do not have class labels. Therefore, to assess their performance this particular factor is taken into consideration, such that the learning goal in these methods is to estimate groups within the data that have similar parameters θ. In other words, the learning is to estimate the statistical distribution from which data clusters belong. Another goal is to identify relationships in the data through dimensionality reduction techniques, which project the data onto lower dimensional spaces allowing for visualization and reduced dimension.

Essentially, for any training dataset X, unsupervised ML methods will identify groups or clusters that have common parameters, which implies a similarity of those data

points. These parameters for the groups are expressed and assessed using a likelihood function which is a measure of belief that particular data points originate from a population having the said parameters θ. Therefore, the likelihood function is expressed as $\mathbf{L}(\theta|X)$ or its variant the log likelihood $Log\mathbf{L}(\theta|X)$.

We will, therefore, discuss the measures used for assessing the performance of our unsupervised machine learning algorithms such as log likelihood, Akaike information criterion (AIC) and Bayesian information criterion (BIC). To compute these measures, assume we are given n data points, and K parameters, which are to be estimated for the clusters or groups in the dataset.

Log likelihood is given by;

$$\text{ll} = Log\mathbf{L}(\theta|X) \tag{2.14}$$

Akaike information criterion (AIC) is given by;

$$\text{AIC} = -2ll + 2K \tag{2.15}$$

Bayesian information criterion (BIC) is given by;

$$\text{BIC} = -2ll + Klog(n) \tag{2.16}$$

Using these measures, one can decide which clustering approach yielded the best results and is thus suitable for this dataset. Of course, it is important also to be aware of what common benchmarks exist in the literature so that one can have confidence in the findings after making a comparison. For this reason, results from the literature were used to validate our results for all experiments, and to assess whether our knowledge extraction efforts, yielded good results in comparison.

2.2 Related work

In this section, we will provide an overview of some related work to contextualize our study and demonstrate its relevance. We discuss various research works related to our study that others have done. In reviewing these studies, we consider data mining approaches, machine learning algorithms as well as the specific population health datasets used. Consequently, we picked all studies for review herein on the basis that they made use of data mining or machine learning techniques. Other qualifying criteria were the usage of some form of public health dataset.

We adopted the new definition of public health surveillance as defined by CDC (Centers for Disease Control and Prevention, 2012). We, therefore, reviewed studies that used traditional methods such as; health and demographic surveillance systems (HDSS), demographic and health surveys, DHS (Ng et al., 2009; Sié et al., 2010; Wang, 2015), medical health records and injury surveillance datasets. In addition to the stated traditional Public Health Survey datasets, whose examples we gave above, we also included innovative web-based health surveillance datasets for consideration in our review of the literature.

Therefore, in this study the broad term "population health datasets" refers to the following datasets; i) occupation health and safety, ii) medical health records, iii)

drug safety surveillance, iv) cancer surveillance, v) cloud-based health surveillance also called digital health surveillance and vi) demographic and health surveillance in all its various forms.

One can quickly verify that machine learning techniques have been used extensively in both predictive medicine and other public health applications through a literature survey. We will show this fact more clearly through the works that we will be referencing in the chapters of this work. Perhaps this popularity of ML in health may be due to that fact that in the fields of predictive medicine as well as public health, there is a need to make a forecast into the future to offer better service or prevent diseases. In this case, possession of some prior knowledge gained from available datasets, about a potential future outcome can help health service providers to tailor or improve treatment of a disease, as well as, to prevent the incidence of new diseases. Machine learning techniques when applied to available datasets, therefore, can provide this advantage of prior knowledge to aid decision making in the health-care provision services and public health in general.

Given the preceding, the applications cited herein involved efforts in making predictions on disease prognosis, forecasting disease transmission as well as prediction of disease epidemics just to name a few. Furthermore, we also reviewed work where researchers have used ML algorithms for attribute selection in high dimensional datasets. In this regard weights of attributes from SVM linear kernel models were used to identify the attributes with good predictive power. The process is as follows, models of linear SVM were built to identify best features and remove those that had low weights in a recursive process, the method is called recursive feature elimination see (Guyon et al., 2002).

We will, therefore, classify the works reviewed herein by the dominant ML algorithms that was used in the study. Even though, our principal interest is on how ML algorithms have been used in the prediction of malaria. Nonetheless, we still highlight a few other relevant cases where other algorithms and techniques have been used targeting other diseases or health outcomes.

2.2.1 SVM

In this sub-section, we focus our discussion on usage of SVM, by looking at applications of this robust machine learning algorithm for cancer, diabetes and malaria prediction.

Cancer prediction

Cancer is probably one of the significant public health concerns worldwide and has therefore received extensive interest in the literature. In this section though we will only focus on those studies that have specifically used SVM or decision trees. Specifically, we briefly discuss three studies that considered dimensionality reduction, binary classification, and multi-class classification problems.

In the first work considered here, Guyon et al. (2002) addressed a dimensionality reduction problem of selecting sets of genes with highest predictive power, from many gene expression data in DNA microarrays. Gene expression is the natural process of making new genes, through a decoding process of the DNA's information,

to generate new gene material, and is a process that happens naturally in all living organisms at the cellular level.

DNA microarray gene expression data, therefore, captures data on the behavior of thousands of genes simultaneously in various stages in this expression process (Lee and Lee, 2003). The behavior of these genes is used to reveal useful information to research, for instance, infected cells and healthy cells can behave differently in this process.

In their research, Guyon et al. (2002) used both cancerous and non-cancerous tissue to represent positive as well as negative cases on which SVM models were trained. Their idea, therefore, was to identify a small set of genes that could effectively be used for cancer prediction, i.e., strong predictors for either the positive or negative classes. The authors used recursive feature elimination (RFE) with SVM in order to identify these genes. In the RFE, at every stage, all features were ranked using weight magnitude generated by a linear SVM classifier, and features with weights below some predetermined threshold were removed recursively.

In a related study, Lee and Lee (2003) used multi-class support vector machines for classification of several cancer types similarly using DNA microarray gene expression data. In this study, the question was whether the DNA microarray data could be used to classify different cancer types. The authors did not want to use a one-versus-rest implementation of SVM where each class becomes a subject in a classification against the rest of the classes. Instead, they trained SVMs to classify multiple cancer cases and leukemia types using the multi-class SVM.

A third study also using DNA microarray data, was done by Furey et al. (2000), who considered two questions the first one being can SVMs be used to classify unseen cancer DNA datasets correctly, the second question was on whether SVM could be used to review samples classified by a human expert and identify any possible misclassification. These authors were interested in classifications of ovarian cancer tissue. However, they also tried the method with other UCI repository datasets on diabetes and colon tumor datasets on which they got acceptable results.

Our study differs with these studies, mainly because we used population-level datasets and not cellular data. We also differ with the first study by Guyon et al. (2002) in that though we also make feature selection with SVM our principal goal is model building for classification of anemic and non-anemic cases. Our classification task is a two class scenario, unlike the work of Lee and Lee (2003) who was interested in a multi-class classification. Where essentially, one performs a one versus all classification for the various classes of interest.

Diabetes prediction

Diabetes is another tremendous public health concern, that is caused either by a failure of the human body to produce insulin (type I diabetes) or failure to utilize insulin (type II diabetes) in blood sugar regulation see (Barakat, Bradley, and Barakat, 2010). In this section, we discuss two applications that used SVM for diabetes prediction or diagnosis by authors (Yu et al., 2010) and (Barakat, Bradley, and Barakat, 2010).

Yu et al. (2010) addressed the question of using SVM to diagnose a person as diabetic, using a set of select variables. In this study, the authors had two schemes for classification. Scheme 1 had a class of the diabetic patients (either medically

diagnosed or not) versus another class of non-diabetic or pre-diabetic people. In Scheme 2 they had a class of diabetic and pre-diabetic versus non-diabetic people. The authors used a nationally representative health survey in the USA called the national health and nutrition examination survey for the time periods 1999 to 2004. The authors used variables such as family history, physical activity, hypertension, body mass index, smoking, alcohol use, education, household income, and sex. The variable used to indicate whether a person has diabetes or otherwise was a fasting plasma glucose measure of $\geq 126\ mgdl$. According to the findings, RBF and linear kernels gave good performance on the schemes 1 and 2 respectively.

The second application of SVM for diabetic prediction was made by Barakat, Bradley, and Barakat (2010), who addressed the question of developing an SVM driven predictive tool for type II diabetes. They also aimed to generate rules to explain the SVM classification process using a rule extraction method to make classification decisions understandable. The authors used a dataset from Oman having similar attributes as the one described earlier for (Yu et al., 2010). The authors used oral glucose tolerance test (OGTT) as follows, $ogtt \geq 200mgdl$ indicated diabetic case. Due to a negatively skewed dataset which they had, where positive diabetic cases were only 9% in the data, the authors reported that they had to employ sub-sampling in-order to overcome this problem.

Our work is similar to these works first of all on the basis that they both used nationally representative datasets. Secondly, these authors also used population-level dataset. However, the significant difference lies in the target class for prediction we focus on anemia which is a proxy variable for malaria, while they focused on diabetes.

Malaria prediction

Unlike the other diseases considered so far, malaria is an infectious disease caused by Plasmodium and spread by the Anopheles mosquito (Sharma et al., 2015). We have not come across a lot of work that has utilized SVM for making malaria or anemia predictions. This section, therefore, only considers three studies as outlined below.

In the first two studies, authors Sudheer et al. (2014) and Sharma et al. (2015) used a combination of environmental factors like temperature, humidity, rainfall and some recorded information on the presence of a malaria epidemic in different malaria endemic regions in India, to make predictions on the occurrence of a malaria epidemic. We also briefly mention two malaria prediction models that do not fall under ML. However, they are well established and accepted in the literature namely, the Liverpool malaria model (LMM) and the vector-borne disease community model (VECTRI) (Ermert et al., 2012).

Sudheer et al. (2014) used a combination of support vector machines as well as a Fire-fly algorithm (FFA), for malaria prediction. The authors used the FFA, to find the best hyperparameters for the SVM models. Subsequently, the models were built using the ideal parameters identified and were trained and tested. Apparently, FFA is inspired by the behavior of the tropical firefly insect in seeking prey, where they follow the luminosity generated by fellow fireflies (Sudheer et al., 2014). The authors used climatic and weather data to build their model using attributes such

as mean monthly values for temperature, humidity, precipitation and malaria incidences. The authors made comparisons with other contemporary machine learning algorithms namely artificial neural networks and found the SVM results to be better.

The LMM and VECTRI are biological and mathematical models, that use climatic, environmental, weather and biological information to predict malaria epidemic (Ermert et al., 2011; Hoshen and Morse, 2004). Specifically, the models make use of daily temperature and precipitation in their computation, and the models also take into account mosquito breeding trends and measures of average bite rates per population.

Finally, our work differs from works cited in this subsection in the sense that, all these models discussed make predictions at the population level mainly using weather, climatic and biological factors. In other words, all these models are concerned with predicting where an epidemic will occur in a particular population. We, however, are interested in making predictions at a personal level, using personal, demographic, health, social economic and anemia status information.

Predicting flu outbreak or progression

Achrekar et al. (2011); Achrekar et al. (2012) investigated the use of Twitter data, to build a predictive data mining (DM) model foretelling the emergence of a flu epidemic in the US. The authors built a model that could predict the number of people that will visit clinics displaying influenza-like illness (ILI) in the US. They validated their results using data from the CDC, which reports on the real prevalence of ILI, based on physicians reports of actual patient visits to hospitals. Data were collected at various time intervals from Twitter users on their public posts or tweets, which contained the following keywords 'flu', 'H1N1' or 'swine flu', and information on the geolocation of the users.

Their method involved feeding their data into three text classifier types namely J48, naive Bayes and support vector machines. The flu epidemic prevalence as predicted by the model from the Twitter data was then matched with the actual prevalence of flu by geographic zones as reported by CDC in its ILI weekly reports. The DM model could then be used to make future predictions, which can be subjected to accuracy verification using CDC reports.

The first research work to consider here was done by (Signorini, Segre, and Polgreen, 2011), who considered the use of Twitter to track $i)$ public concern related to H1N1 or swine flu epidemic and $ii)$ actual progress of the disease using a time-stamped geo-referenced twitter stream. The authors used a dataset generated from the Twitter stream, isolating tweets with keywords like 'flu', 'swine', 'Tami flu', 'influenza' and 'flu illness' among others. All tweets were time stamped and geo-located to ensure that when and where tweets originate is known. Tweets were included, if and only if, they originated in the USA for the period beginning October 2009 until December 2009. The authors used support vector regression a variant of SVM classification, with tweets being treated as attributes and regression variables in their non-linear model. Furthermore, for the support vector machine implementation, they used the open source platform libSVM for carrying out their experimentation. The authors verified their findings using CDC weekly influenza-like illnesses reports, which is from a nationwide influenza surveillance programme in the USA.

2.2.2 Decision trees and association rule mining

In this subsection, we discuss studies that have used decision trees and association rule mining. Though it is possible to learn association rules from decision trees, there are many tailor-made association rule mining algorithms which we will briefly discuss here. Nevertheless, it seems logical to discuss studies that used both these algorithms in this subsection, because of their capacity for generation of association rules.

Anemia prediction

In his master's thesis Setotaw (2013) worked on a similar problem of anemia prediction using decision trees and rule induction ML algorithms, this author used the Ethiopian Demographic Health Survey in his study. The author identified a set of features from the dataset which he used for the prediction task.

Injury narrative classification

In their work, Nenonen (2013) addressed a different problem of trying to understand: causal, circumstantial and consequential factors that pertain to work injuries. The authors employed decision trees and association rule mining for their work; they built a model tree, based on the data which they used to identify the association rules. The authors used a dataset from the Finnish occupational accidents and diseases database collected by a federation of insurance institutions in Finland. Generally the association rules take the form of $(A\&B \implies C)$ i.e. If $(A = 1)$ AND $(B = 1)$ THEN $(C = 1)$ with some probability p. They further used the knowledge measures such as support, lift, and confidence to assess the quality of their knowledge.

These measures were used for deciding which association rules are interesting and useful enough for further consideration. They, therefore, attempted to relate cause, circumstances, and consequence of every incident using this approach.

Salmonella surveillance

Another piece of work was done by Brossette et al. (1998), who used association rule mining on *Salmonella* surveillance data. The method that they developed involved analyzing partitions or small time slices of the dataset, where each partition has a few records. The time slice data would then be passed through the association rule mining algorithms to identify high support rules of the form $A \implies B$.

The rules identified in each time slice were then saved for comparison with other subsequent rules from other time slices data, using measures of support and confidence of the rules. Their method was applied to public health and hospital infection surveillance datasets. To ensure that they identify interesting, i.e., nontrivial rules, the authors paid attention to rules that had low confidence, with a justification that such rare rules may reveal new information, which can otherwise be undetected using other conventional methods.

The limitation of their work was that what they developed was shown to work best with categorical variables only and did not involve continuous variables.

Pharmaco-vigilance

Ibrahim et al. (2016) addressed the problem of drug safety surveillance, also referred to as pharmacovigilance. The authors used association rule mining (ARM) on pharmaco-vigilance (PhV) datasets to detect novel, frequent and significant drug adverse effects interactions in the USA. Their work was motivated by the fact that, the introduction of any new drug has its side effects on patients, but also other adverse effects may be caused, due to the drug's interaction with other medicines that it is taken concurrently with by the patients. PhV is a special form of PHS, which is concerned with drug safety surveillance, to monitor any harmful impact of a drug both prior as well as after being released to the market for public consumption.

The authors adopted an *ARM* algorithm called *A-Priori*, which they used on a consolidated dataset that had information on demographics, drugs and drug interactions, $I = [I_1, I_2, I_3 \dots I_n]$, in the database and transactions $T \subseteq I$. The aim was to find rules $[X_1, X_2, X_3 \dots X_n] \implies Y$, such that $X \subseteq I$ and also $Y \subseteq I$ where X are drugs and Y are a set of interacting drugs. The authors used a three-step method in the A-Priori *ARM* variant that they used. First of all, they defined frequent itemset such as Y for drugs that have potentially dangerous interactive adverse effects from the dataset. Secondly, they computed Chi-Square (χ^2) and proportional reporting ratio (*PRR*), which are the metrics they chose to measure the strength of the associations. Finally, they filtered very strong associations based on the metrics stated earlier using a predetermined threshold. They, therefore, reported those associations after some additional qualitative review.

Predicting disease condition and life expectancy

A real-time clinical decision support system was developed by Zhang et al. (2012) to aid medical practitioners in diagnosis and prediction of medical conditions based on a stream of medical data about patients health status. The EHR dataset used was of the form $T = [T_1, T_2, T_3, \dots T_n]$ where T_n represented a data stream at time n such that $T_n = [X_1, X_2, \dots X_k]$ where every X_k is a vital sign value i.e. body temperature, blood pressure etc. In order to classify the data therefore, three data streams from T were classified into classes where $C = [C_1, C_2, C_3, \dots C_m]$ such that C_m is an illness class. The classifications made are later used to make predictions by comparison with new cases. The authors implemented their data mining system by altering the very fast decision tree (VFDT) algorithm, to allow for storage of classification knowledge in pointers for prediction of a disease class. They compared their system to a commercial product, that does a similar task from IBM that uses a distance measure of the form; $d_m(x_i, x_j) = \sqrt{(x_i - x_j)^T P (x_i - x_j)}$ they found their system to be better because it did not require an off-line analysis of the inputs before making predictions.

Mathias et al. (2013) developed a life expectancy predictive data mining model. The model used electronic medical records for senior citizens ($\geq 50 years$) and aimed to predict mortality in 5 years. The motivation for the authors was to utilize accurate life expectancy to facilitate better clinical care decisions for patients. The authors

used feature selection to identify best predictors from some patient attributes in the EHR for instance: social, demographic, vital signs and laboratory results. They used *Rotation Forest Ensemble* with an alternating decision tree as a classifier which gave comparable results to other existing approaches.

Predicting heart disease

Palaniappan and Awang (2008) used three ML algorithms namely artificial neural networks, naive Bayes and decision trees for predicting heart disease using EHR.

Marucci-wellman, Lehto, and Corns (2015) addressed the problem of generating semi-automated injury narrative classification from large workers compensation database in the USA. They aimed to improve the time taken to classify injuries, which is an important step in the processing of workers compensation for injuries sustained at work. The authors developed a semi-automated classification system. The system had two components an automated ML classification system and a human supervisor to verify the classifications made.They employed Naive Bayes classification algorithms and their model for finding the probability of an accident event given a set of words in the narrative as follows;

$$P(C_i|n) = P(C_i) \prod \frac{P(n_j|C_i)}{P(n_j)}$$

Where $P(n_j|C_i)$ is the probability of word n_j corresponding to event i. $P(n_j)$ is marginal probability for observing word n_j and C_i is marginal probability of event code i in training dataset.

Predicting mortality or readmission

The authors Fialho et al. (2012) worked on creating a fuzzy logic model for predicting readmission between 24 to 72 hours for patients, who were discharged from the intensive care unit (*ICU*). The authors used a retrospective dataset from several ICU's at a hospital in Boston, USA, which contained physiological variables for patients. The authors identified a set of physiological variables that had the most predictive power for readmission. The model they developed was a classifier that was used to classify into readmission or no readmission classes. The model was of the form: if X_1 is $A_{j1} \cdots X_{jN}$ Then $Y_j = (a_j)^T X + b_j$ where X are the features with a_j and b_j being the slope and intercept respectively.

Campbell et al. (2008) considered the question of predicting patient readmission or death after intensive care unit *ICU* discharge. The motivation for their research was to allow for a prospective risk analysis of patients to avoid preventable mortality. The authors used a prospective dataset of routine clinical audit data collected for all patients at a mixed medical and surgical ward in Aberdeen, Scotland from 1995 to 2005. They focused on patients that were admitted to the ICU and got discharged alive. They built logistic regression models that have variables like age, sex, duration in the hospital before ICU admission, the severity of illness among others to predict death and readmission of patients, that were discharged from the ICU. The authors used logistic regression models for these predictions and were able to identify major risk factors for both readmission and death.

Tourassi, Yoon, and Xu (2016) worked on a project that aimed to use innovative web-based cancer mortality surveillance and compare performance to the traditional cancer surveillance approach. The authors employed a web crawler that automatically collected online obituaries, US newspapers and social media posts for deceased cancer patients. Their target was to extract: lung and breast cancer obituaries and gauge; temporal, geospatial, age and gender information from the posts. The authors used web crawling technology and text mining, from the Stanford Natural Language Processing Library, in building their dataset from the web. Data were filtered based on some factors to obtain clean data having only the relevant posts. Finally, statistical measures were taken to assess trends, against the traditional approaches of cancer surveillance, and were found to be comparable in performance.

In conclusion in this section, we have discussed research works that have made use of the various supervised ML algorithms on PHS datasets as per our earlier definition.

2.2.3 Unsupervised learning

Two relevant pieces of work were done by Kaski and Kohonen (1996) and Sarlin (2012), who used self-organizing maps for exploratory data analysis. These authors tackled the problems of comparing economies of various countries and tracking of the Millennium Development Goals. These algorithms were used efficiently for spatio-temporal analysis of economic and development datasets that the authors used.

Kaski and Kohonen (1996) tackled the problem of comparing various economic systems using macro and microeconomic factors, to identify poverty and welfare states. They used a dataset from the World Economic Report having 39 attributes including information on health, education, consumption and social services. Using SOM, the authors were able to visualize the clustering of countries based on the economic indicators.

Sarlin (2012) considered the problem of tracking the progress on attainment of the Millenium Development Goals, for Sub Saharan Africa. The dataset contained information on various targets of the 8 Millennium Development Goals, such as poverty, education, women, diseases, and development, obtained from the United Nations Statistics Division. In both studies, the authors superimposed their self-organizing maps on the global geographic map, so that various clusters could easily be identified by having a similar color shade.

In summary, in this chapter, we have motivated the role of data mining in extracting knowledge for the measurement of global development goals. Furthermore, we highlighted our research objectives which are, to explore the use of data mining to extract useful and interesting knowledge from a demographic and health dataset from Malawi. We also stated that we used the anemia attribute as a proxy for malaria in the supervised ML experiments. We also discussed, some related works that have been done, while highlighting the ML algorithms used and specific data mining problems that were being addressed.

3 Methods

Overall, our research objective was to demonstrate to what extent knowledge extracted using DM techniques can contribute to the understanding of progress towards and or attainment of the SDG health-related goals in particular. We now proceed to discuss details of the methods used in the study to accomplish those goals in the subsequent sections.

3.1 Study setting

Geographically, Malawi is located in the southern part of Africa and borders Mozambique, Zambia, and Tanzania. Furthermore, Malawi is a reasonably small country and is located along the great rift valley with land that covers $118,484$ square kilometers, and water comprises 20% of the country. In addition to this, the country lies at an altitude that is between $1,700$ to $3,000$ meters above sea level (Government of Malawi, 2014).

Consequently, being located in a tropical region south of the equator, vector-borne diseases like malaria are endemic and constitute a significant cause of morbidity and mortality. The usual higher temperatures, humidity, and rainfall are known to lead to an increase in Malaria transmission, through the provision of conducive environmental conditions for the breeding of the Malaria vector the Anopheles mosquito (Government of Malawi, 2014; Sharma et al., 2015).

3.2 Data collection

This study used a demographic and health survey dataset from Malawi (MDHS). We used secondary data for the Malawi demographic health surveillance, obtained freely from `www.dhsprogram.com` upon registration. Our focus was on the dataset files that contained information on women, who are the primary sources of information in these survey data. We discuss the dataset in more detail in chapter 4; however, here we describe our research methods and data mining approach.

3.3 Experimental setup

We will now discuss the hardware and software tools that we used for all the experiments in this research study. Due to the high computational requirements of some machine learning algorithms, there is a need to have access to hardware and software platforms that have the required physical resources to make the experimentation easier.

3.3.1 Hardware platforms

The experimental hardware platform setup was as follows; we used two personal computers for the development of Python model building scripts. We used a MacBook Pro laptop with 4 gigabytes of RAM, 500 gigabytes of hard disk drive storage space with a 2.5 gigahertz processor. The second machine was a Dell desktop with 16 gigabytes of RAM, 1 terabyte of hard disk drive storage space with a 3.6 gigahertz processor.

3.3.2 Coding and software platforms

Experiments were conducted both in Weka data mining platform and in Python 3 using the Spyder integrated development environment. Weka comes with a GUI and is well suited for data exploration and visualization, we have used several screen-shots from Weka in this document in various chapters. Weka can also be used programmatically by importing its classes in a Java program. On the other hand, Python was used more for data cleaning, removal of null values, experimentation and for doing deeper analysis of the algorithms code for parameter tuning. The Python scripts were written using the Spyder integrated development environment package in the Anaconda software package running Python version 3.6. We used the following Python Data Science packages namely: Pandas, NumPy and SciKit-Learn. MatPlotLib was used for graphical presentation of the dataset exploration tasks and the results of model building.

The dataset was preprocessed in Spyder using Python language and Weka data mining platform to generate comma-separated values (CSV) and attribute relational file format (ARFF) files. The ARFF is a special file format for Weka and has a concise way of recording the nature and type of attributes if an attribute is categorical, all the values that it can take are indicated in the file immediately after the attribute name otherwise if an attribute is numeric that information is also stated. After that, we created a database and stored the resultant datasets, together with a data dictionary having explanatory notes on the attributes.

3.3.3 Database design and development

Furthermore, experimental results were also saved in the same MySQL database for later retrieval and analysis. We designed specific tables to hold the final datasets and other tables to hold experimental results for all the ML algorithms that were used in this study. Direct database interaction was done using MySQL Workbench and PHPMyAdmin database management platforms. Otherwise, drivers for Python and Java were used for accessing the databases from the Spyder Python environment and Weka respectively.

We designed database tables for the dataset and tables to hold results for the various experiments that we used in this study. In the table design for the dataset, we entered the entire clean dataset with all the cleaned attributes. Datasets used in the various supervised and unsupervised scheme were afterward queried from this table by specifying the attributes required either for the full dataset or controllable features dataset which was based on a systematic attribute selection process.

3.4 DM methodology

In this study, we made use of the six-step CRoss Industry Standard Process for Data Mining (CRISP-DM) methodology, which has a hierarchical process model (IBM, 2011). Details were discussed in chapter 2, sub-section 2.1.5.

3.5 Model building

In this section, we discuss two techniques that were useful for us in the model building phase, namely grid search and cross-validation.

3.5.1 Grid Search

Grid search involves searching for optimum hyper parameter values in some chosen parameter space.The grid search is an exhaustive search within the chosen parameter space, where a set of values for each hyper-parameter are given (Chih-Wei Hsu, Chih-Chung Chang and Lin, 2008) and models are built for each given set of parameters, after that one has to assess model performance, to identify the best set of parameters. Ideally the grid search process is performed on a subset of the training dataset prior to the model building process.

3.5.2 Cross validation

K-fold cross-validation is a technique that splits a dataset into k folds or partitions for example $k \in \{3, 5, 10\}$ folds. Using these folds after that one trains the model using $k - 1$ folds, and holds out the other 1 fold for testing. The rationale is to adequately train a model with a large dataset so that once it has appropriately learned model parameters or patterns so that it can be used for prediction on the single hold-out dataset during testing.

Cross Validation is helpful in ensuring that each data point is seen once by the model, this ensures that the average performance of the model after training, is closer to the real performance score for the model after deployment.

Overall, in this section we have provided an overview of the methods that were used in the study. We revisit these methods in the specific chapters where we report on specific experiments using our ML algorithms of interest. In those sections therefore any methods that are specific to the ML experiments are mentioned.

4 Dataset preparation

4.1 Understanding the Dataset

We explored the structure of the data and the meaning of the attributes names to understand the information content. We further analysed the relationships between the attributes to identify any redundancies and ambiguities in the process of establishing an understanding of our dataset. Moreover, since we used a secondary dataset in this study, we revisited the questionnaires that were used for data collection in the original surveys, to aid our understanding. Revisiting the questionnaires was quite helpful, mainly because the coding of some attribute names made them somewhat ambiguous. Therefore, we needed to understand the intent of collecting these data for those attributes, so that our interpretation of the information in these attributes is consistent with reality and the original purpose.

Furthermore, some preliminary data mining experiments were performed to assess the feasibility of employing particular ML algorithms on the dataset, in the model building phase. We also provide some descriptive statistics on select attributes to give a picture of the character of the dataset.

4.1.1 Data collection

The Malawi Health and Demographic Survey is a household survey conducted periodically as a monitoring and evaluation tool, for national and international development goals. The survey captures information on general health and well being, contraceptive usage among women, household sanitation status, malaria prevalence, social economic status, fertility, malaria prevention and treatment, infant mortality, maternal mortality and domestic violence (Malawi National Statistical Office and ICF, 2017). The Malawi Health and Demographic Survey is implemented by the National Statistical Office of Malawi (Malawi National Statistical Office and ICF, 2017).

The survey uses two main questionnaires; the first one is a household questionnaire used for gathering household related information such as the number of household occupants, possessions, health status, number of children. The second questionnaire is a woman's questionnaire used to get information from women in participating households along the themes mentioned earlier. It is necessary to point out here that women are the primary source of information for these surveys, though men are also interviewed in selected households.

The dataset was obtained freely upon registration, from the DHS programme website on `http://www.dhsprogram.com`. The DHS is implemented by NSO's in a number of countries in Europe, Africa, Asia and South America (Rutstein, Winter, and International, 2014). Though themes covered in these surveys are standard, there is an increasing addition of topics in each survey year in different countries,

to capture additional variables of interest as determined by specific country NSO's. Therefore, the number of variables in these survey years are not the same. In our study, however, we focus on the dataset that had information on women in the various households, that provided data through the surveys. We obtained all survey datasets for Malawi since 1992, but we only used the 2004 and 2010 datasets for our study.

Through a review of the survey questionnaires, we noted that the dataset had two implicit categories, the first being a category for all participants and the second being a category for participants, chosen to answer questions on specific issues such as iodine, malaria, and domestic violence. As stated before, we were interested in all malaria and anemia related questions; therefore we identified features that had information on malaria and anemia related issues for use in our supervised ML experiments.

4.1.2 Data description

To provide a picture of how the information on malaria was distributed in the original merged dataset, we have chosen one attribute for illustration. We did this as part of the process of data understanding, which involves going through the dataset and analyzing the attributes individually. In this case information about anemia status, and coded as $v457$ is shown in figure 4.1. This attribute is categorical and takes five different values namely severely anemic, moderately anemic, mildly anemic, not anemic and missing values coded as 1, 2, 3, 4 and 9 respectively. In the image, we visually depict the class distributions, levels of information completeness and missing values status for the attribute. We selected this attribute because, it represents the distribution of attributes having anemia and malaria information, which were chosen for our supervised ML experiments.

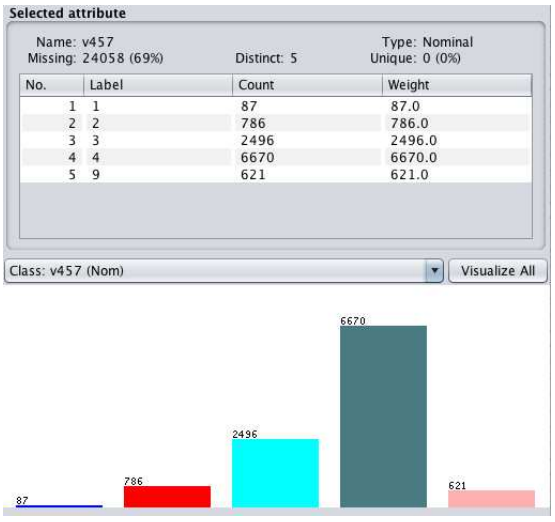


FIGURE 4.1: *In this figure a summary of the information in the anemia attribute $v457$ is shown. The image was taken from the Weka data preprocessing tab. In the dataset information on anemia was found in instances where the survey participants got selected and consented to anemia testing. In this case 31% of the instances had data in this attribute. Therefore supervised experiments which required malaria related attributes for making predictions had $10,660$ instances for both training and testing.*

Some of the specific information contained in these attributes of interest were, anemia status, use of anti-malaria drugs and its prevalence, prevention, and treatment.

Data dimensionality

Table 4.1, shows a summary of our datasets including dimensionality status. This dataset was deemed adequate for running both supervised and unsupervised experiments, mainly because in comparison to the sizes of the datasets that other researchers used in the literature, our dataset was of a good size.

However, the dataset was unbalanced with more negative cases approximately two thirds more, than positive anemia cases, which we were interested in predicting. Cohen et al. (2004) tackled a similar problem of data skewness and stated that this problem is addressed in two main ways.

The methods of tackling data skewness are oversampling the under-represented class and undersampling the over-represented class. In either case, the goal is to reduce bias by allowing an algorithm to see equal numbers of both positive and negative cases.

We employed undersampling in this study for decision tree experiments because these algorithms are more susceptible to over-fitting. The general idea therefore was to ensure that the same number of positive and negative cases were selected for the supervised methods.

Themes

The Malawi Demographic and Health Survey collects information on many themes, some of which are presented in this sub-section. We present a sample of themes here and provide a more detailed list of variables later in the chapter. For each theme, there is a data section or set of variables, where that information is found in the dataset.

The list of themes that are found in the dataset are presented in no specific order as follows: $1)$ Maternal and Child health, $2)$ Family planning and $3)$ Fertility, $4)$ Malaria, $5)$ Childhood mortality, $6)$ HIV and Aids knowledge and behavior and finally $7)$ Nutrition (Malawi National Statistical Office and ICF Macro, 2010). We were mainly interested in malaria and its prevalence, prevention, and treatment among women. In the dataset malaria prevalence was estimated by the anemia attribute.

4.1.3 Data exploration: preliminary ML experimental results

We performed a series of data mining experiments on the dataset to both understand the structure of the data as well as to determine which ML algorithms would yield good results. The goal of this data exploration exercise was to gain insights into the dataset and to assess the results that will be generated from the various supervised and unsupervised machine learning algorithms before the modeling phase. These experiments were conducted on the full clean dataset refer table 4.1 and generated various ML models.

We explored our dataset using two supervised machine learning algorithms namely naive Bayes and $J48$ decision tree to assess classification performance. We used the anemia attribute $v457$, as a target class to make predictions on. As stated in 4.1.2 the attribute took on five different values for various levels of being anemic, not anemic

and for missing values. According to the results for these supervised learning algorithms, it was difficult to separate the various anemic cases, e.g., separating mild and moderate anemic cases. Moreover, the missing value category's accuracy result did not give any useful information because we cannot tell to what class that result belongs.

We, therefore, decided to remove the class of the missing value 9 in this attribute and generate a new binary attribute called anemic which takes the values 0 for not anemic and 1 for anemic. Therefore, this led to merging of classes 1, 2, 3 as anemic and 4 maintained as non-anemic.

As was shown in figure 4.1, the anemic cases were: 87 severe , 786 moderate and $2,496$ mild. The anemic class, therefore, had about $3,269$ cases while the non-anemic class had $6,670$ cases, the remaining were missing cases coded as 9. This new attribute called anemia, is the predictive class that was used in the supervised experiments later on, as will be seen in chapter 6 and chapter 7.

Furthermore, the preliminary clustering results showed that the five different values that the anemia level takes were randomly clustered into the various clusters, without any distinct patterns of separation.

4.2 Data preparation

This is the third phase in the CRISP-DM and is done prior to model building. This phase involves data preprocessing, data cleaning and data formatting, each of which is discussed in the sub-sections below.

We conducted data preprocessing which involved the processes of removing: null values, redundant attributes, compilation of final dataset and selection of a reduced dataset with controllable features, as outlined in the figure 4.2. We started off with the DHS dataset files for survey years 2004 and 2010 The merged data file was subjected to data cleaning as illustrated in the same figure 4.2.

The two main survey files for years 2010 and 2004 were named IR2010 and IR2004 respectively for those survey years, and contained information for women aged 15 years and beyond. These data files were consequently merged, which led to the compilation of a merged data file having $34,718$ data points and $5,725$ attributes.

After that we proceeded to remove all attributes with missing data, we further selected only data points with information on anemia and malaria, this led to a reduction of data points to $10,660$ from the initial $34,718$.

Finally, at the end of this phase, we generated a file with all clean attributes. The dataset which we generated in this process was $10,660$ data points by 132 attributes. However, this file still had coded missing values for anemia, once those were removed the final dataset had $10,039$ cases.

We finally provide a list of the features of the final datasets as described in this subsection. This information is provided in attribute relational file format, the native format for Weka data mining platform. The file begins with the keyword $@relation$ followed by the name of the ARFF file name. An attribute begins with $@attribue$ keyword followed by the name of the attribute and data type or list of values acceptable for categorical variables. After the attributes have been defined stating both names

#	Year	File name	Dimension	
			rows	cols
1	2004	data file 1 (2004)	11,698	4,864
2	2010	data file 2 (2010)	23,020	4,271
3	All	merged data file	34,718	5,725
4	All	final data file	10,039	133

TABLE 4.1: *MDHS dataset used was for the* 2004 *and* 2010 *survey years. After completing the data cleaning exercise we had a dataset of* 10,039 *cases and* 133 *attributes, which we used in the experiments.*

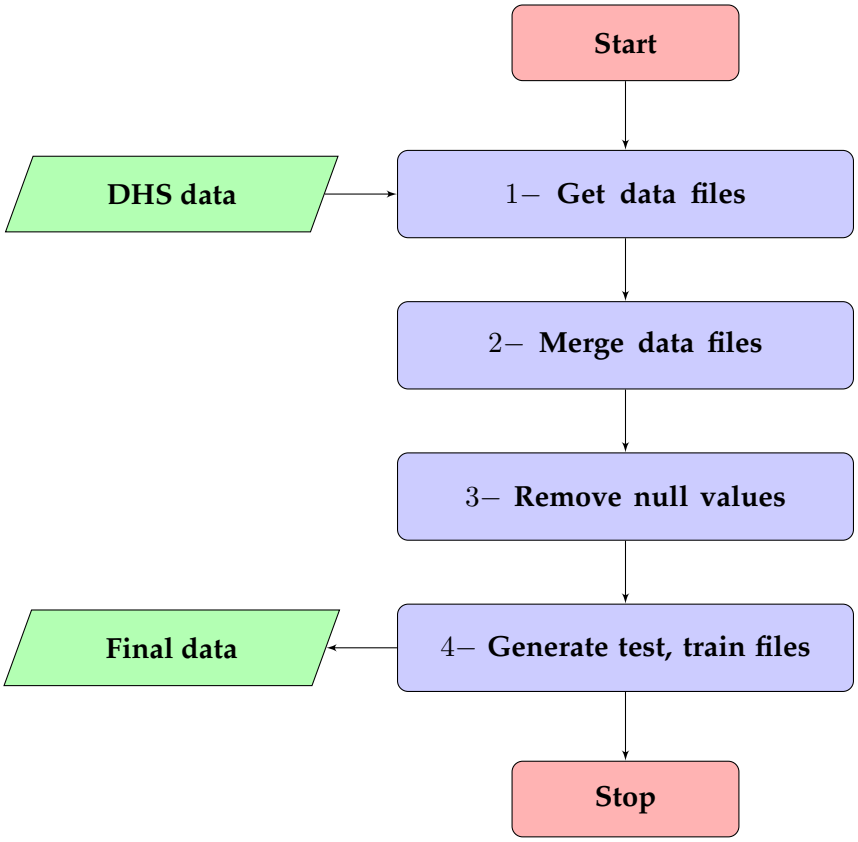


FIGURE 4.2: *Data pre-processing steps*

and data types or data values, the data instances follow preceded by the keyword $@data$. Each data instance is stored on a separate line, with commas separating individual attribute values for that data instance.

List of all features in full features dataset

@attribute 'Toilet facilities shared, v160' {0,1,7,9}
@attribute 'Results of salt iodine test, v166' {0,7,15,30,994,995,997,999}
@attribute 'Ever participated in a literacy program outside, v156' {0,1,9}
@attribute 'Children under 5 slept under bednet last night, v460' {0,1,2,3,9}
@attribute 'During pregnancy - took Fansidar for Malaria, m49a1' {0,1,8,9}
@attribute 'Highest year of education, v107' numeric
@attribute 'Type of bednet(s) slept under last night, ml101' {0,1,2,3,9}
@attribute 'Respondents month of birth, v009' numeric
@attribute 'Respondents year of birth, v010' numeric
@attribute 'Date of birth (CMC), v011' numeric
@attribute 'Current age - respondent, v012' numeric
@attribute 'Age 5-year groups, v013' {1,2,3,4,5,6,7}
@attribute 'Region, v024' {1,2,3}
@attribute 'Type of place of residence, v025' {1,2}
@attribute 'De facto place of residence,v026' {0,1,2,3,9}
@attribute 'Cluster altitude in meters, v040' numeric
@attribute 'Household selection for hemoglobin, v042' {0,1}
@attribute 'Selection for domestic violence module, v044' {0,1,2,3}
@attribute 'Region, v101' {1,2,3}
@attribute 'Type of place of residence, v102' {1,2}
@attribute 'Years lived in place of res, v104' numeric
@attribute 'Highest educational level, v106' {0,1,2,3,9}
@attribute 'Source of drinking water, v113' numeric
@attribute 'Type of toilet facility, v116' {10,11,20,21,22,23,30,31,41,42,43,96,97,99}
@attribute 'Has electricity, v119' {0,1,7,9}
@attribute 'Has radio, v120 ' {0,1,7,9}
@attribute 'Has television, v121' {0,1,7,9}
@attribute 'Has refrigerator, v122' {0,1,7,9}
@attribute 'Has bicycle, v123' {0,1,7,9}
@attribute 'Has motorcycle, v124' {0,1,7,9}
@attribute 'Has car, v125' {0,1,7,9}
@attribute 'Main floor material, v127' {10,11,12,20,21,22,23,30,31,32,33,34,35,96,97,99}
@attribute 'Religion, v130' numeric
@attribute 'Ethnicity, v131' {1,2,3,4,5,6,7,8,9,10,11,12,96,99}
@attribute 'Education in single years, v133' numeric
@attribute 'Usual resident or visitor, v134' numeric
@attribute 'Usual resident or visitor, v135' {1,2,9}
@attribute 'Number of household members, v136' numeric
@attribute 'Number of children 5 and under, v137' numeric
@attribute 'Number of eligible women in HH, v138' numeric
@attribute 'De jure region of residence, v139' {0,1,2,3,97,99}
@attribute 'De jure type of place of res, v140' numeric
@attribute 'De jure place of residence,v141' numeric
@attribute 'Educational attainment, v149' {0,1,2,3,4,5,9}
@attribute 'Relationship to household head, v150' numeric
@attribute 'Sex of household head, v151' {1,2}
@attribute 'Age of household head ,v152' numeric

@attribute 'Has a landline telephone, v153' numeric
@attribute 'Literacy v155' {0,1,2,3,4,9}
@attribute 'Frequency of reading newspaper or magazine, v157' {0,1,2,3,9}
@attribute 'Frequency of listening to radio, v158' {0,1,2,3,9}
@attribute 'Frequency of watching television, v159' {0,1,2,3,9}
@attribute 'Type of cooking fuel, v161' {1,2,3,4,5,6,7,8,9,10,11,95,96,97,99}
@attribute 'Wealth index, v190' {1,2,3,4,5}
@attribute 'Wealth index factor score,v191' numeric
@attribute 'Total children ever born, v201' {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20}
@attribute 'Sons at home, v202' {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20}
@attribute 'Daughters at home, v203' {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20}
@attribute 'Sons elsewhere, v204' {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20}
@attribute 'Daughters elsewhere, v205' {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20}
@attribute 'Sons who have died, v206' {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20}
@attribute 'Daughters who have died, v207' {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20}
@attribute 'Births in last five years, v208' {0,1,2,3,4,5,6}
@attribute 'Births in past year, v209' {0,1,2,3,4}
@attribute 'Births in month of interview, v210' {0,1,2,3}
@attribute 'Currently pregnant, v213' {0,1}
@attribute 'Time since last menstrual perd, v215' numeric
@attribute 'Menstruated in last six weeks, v216' {0,1}
@attribute 'Knowledge of ovulatory cycle,v217' {1,2,3,4,5,6,8,9}
@attribute 'Number of living children, v218' {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20}
@attribute 'Living children n current preg,v219' {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20}
@attribute 'Living children + curr preg 6+, v220' numeric
@attribute 'Entries in birth history, v224' numeric
@attribute 'Ever had a terminated preg, v228' numeric
@attribute 'Index last child prior to cal, v235' numeric
@attribute 'Births in last three years, v238' numeric
@attribute 'Knowledge of any method, v301' {0,1,2,3}
@attribute 'Ever use of any method,v302' {0,1,2,3}
@attribute 'Current contraceptive method, v312' {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,99}
@attribute 'Current use by method type, v313' {0,1,2,3,9}
@attribute 'Pattern of use, v361' {1,2,3,4}
@attribute 'Contraceptive use n intention, v364' {1,2,3,4,5}
@attribute 'Heard FP on radio last months, v384a' {0,1,9}
@attribute 'Heard FP on TV last months, v384b' {0,1,9}
@attribute 'Heard FP newspaper last months, v384c' {0,1,9}
@attribute 'Visited by FP worker last 12m, v393' {0,1,9}
@attribute 'Visited health facil. last 12m, v394' {0,1,9}
@attribute 'Currently breastfeeding, v404' {0,1}
@attribute 'Currently amenorrheic, v405' {0,1}
@attribute 'Currently abstaining, v406' {0,1}
@attribute 'Heard of oral rehydration, v416' {0,1,2,9}
@attribute 'Entries in maternity table, v417' {0,1,2,3,4,5,6}
@attribute 'Entries in health table,v418' {0,1,2,3,4,5,6}
@attribute 'Entries in height/weight table,v419' {0,1,2,3,4,5,6}
@attribute 'Result of measurement of resp, v447' {0,1,2,3,4,5,6,7,9}
@attribute 'Have bednet for sleeping (household report), v459' {0,1,9}
@attribute 'Respondent slept under bednet, v461' {0,1,9}
@attribute 'Smokes cigarettes, v463a' {0,1,9}

@attribute 'Smokes pipe, v463b' {0,1,9}
@attribute 'Chewing tobacco, v463c' {0,1,9}
@attribute 'Uses snuff,v463d' {0,1,9}
@attribute 'Does not use tobacco, v463z' {0,1,9}
@attribute 'Getting medical help for self: getting permission, v467b' {0,1,2,9}
@attribute 'Getting medical help for self: getting money needed, v467c' {0,1,2,9}
@attribute 'Getting medical help for self: distance to health, v467d' {0,1,2,9}
@attribute 'Getting medical help for self: having to take trans, v467e' {0,1,2,9}
@attribute 'Getting medical help for self: not wanting to go, v467f' {0,1,2,9}
@attribute 'Getting medical help for self: concern no female hw,v467g' {0,1,2,9}
@attribute 'Current marital status, v501' {0,1,2,3,4,5,9}
@attribute 'Currently/formerly/never married, v502' {0,1,2,9}
@attribute 'Recent sexual activity, v536' {0,1,2,3,9}
@attribute 'Fertility preference, v602' {0,1,2,3,4,5,6,9}
@attribute 'Desire for more children, v605' {0,1,2,3,4,5,6,7,8,9}
@attribute 'Ideal number of children, v613' numeric
@attribute 'Ideal number of children (grp), v614' numeric
@attribute 'Respondent currently working, v714' {0,1,9}
@attribute 'Respondent occupation, v716' numeric
@attribute 'Respondent occupation,v717' {0,1,2,3,4,5,6,7,8,9,99}
@attribute 'Worked in last 12 months, v731' {0,1,2,3,9}
@attribute 'Heard sexual transmit. disease, v750' {0,1,2,8,9}
@attribute 'Ever heard of AIDS, v751' {0,1,9}
@attribute 'Source for condoms: Friends, relatives, v762au' {0,1,9}
@attribute 'Had any STD in last 12 months, v763a' {0,1,8,9}
@attribute 'Heard about other STDs, v785' {0,1,9}
@attribute 'Currently pregnant (household report), v454' {0,1,9}
@attribute 'WtHt Percent ref median Fog, v443' numeric
@attribute 'WtHt Percent ref median WHO, v444' numeric
@attribute 'Wt/Ht Std deviations(resp) DHS ,v444a' numeric
@attribute 'Rohrers index for respondent, v446' numeric
@attribute 'Wt/Ht Percent ref. median (DHS),v442' numeric
@attribute 'Ht/A Percent ref. median (resp),v441' numeric
@attribute 'Result of measuring (Hemoglobin), v455' {0,3,4,6,7,9}
@attribute 'Hemoglobin level adjusted by altitude, v456' numeric
@attribute 'Anemia level, v457' numeric
@attribute anemia {-1,1}

4.2.1 Controllable features dataset

Using the final clean dataset discussed in the previous section, we systematically selected a set of controllable features. We asked ourselves in analysis of each attribute a question of whether it is possible to intervene and bring change to the issue represented by information kept in the attribute. If our view was in the affirmative, we kept that particular attribute, otherwise that feature was discarded. The list of these 50 controllable features is provided in the list below.

List of controllable features

@attribute 'Type of place of residence, v102' {1,2}
@attribute 'Highest year of education, v107' numeric
@attribute 'Highest educational level, v106' {0,1,2,3,9}
@attribute 'Source of drinking water, v113' numeric
@attribute 'Type of toilet facility, v116' {10,11,20,21,22,23,30,31,41,42,43,96,97,99}
@attribute 'Has radio, v120 ' {0,1,7,9}
@attribute 'Has television, v121' {0,1,7,9}
@attribute 'Has refrigerator, v122' {0,1,7,9}
@attribute 'Has bicycle, v123' {0,1,7,9}
@attribute 'Has motorcycle, v124' {0,1,7,9}
@attribute 'Has car, v125' {0,1,7,9}
@attribute 'Main floor material, v127' {10,11,12,20,21,22,23,30,31,32,33,34,35,96,97,99}
@attribute 'Main wall material, v128' numeric
@attribute 'Education in single years, v133' numeric
@attribute 'Number of household members, v136' numeric
@attribute 'Number of children 5 and under, v137' numeric
@attribute 'Number of eligible women in HH, v138' numeric
@attribute 'De jure region of residence, v139' {0,1,2,3,97,99}
@attribute 'Educational attainment, v149' {0,1,2,3,4,5,9}
@attribute 'Literacy v155' {0,1,2,3,4,9}
@attribute 'Ever participated in a literacy program outside, v156' {0,1,9}
@attribute 'Frequency of reading newspaper or magazine, v157' {0,1,2,3,9}
@attribute 'Frequency of listening to radio, v158' {0,1,2,3,9}
@attribute 'Frequency of watching television, v159' {0,1,2,3,9}
@attribute 'Toilet facilities shared, v160' {0,1,7,9}
@attribute 'Type of cooking fuel, v161' {1,2,3,4,5,6,7,8,9,10,11,95,96,97,99}
@attribute 'Results of salt iodine test, v166' {0,7,15,30,994,995,997,999}
@attribute 'Wealth index, v190' {1,2,3,4,5}
@attribute 'Children under 5 slept under bednet last night, v460' {0,1,2,3,9}
@attribute 'Respondent slept under bednet, v461' {0,1,9}
@attribute 'Result of measuring Hemoglobin, v455' {0,3,4,6,7,9}
@attribute 'During pregnancy took Fansidar for Malaria, m49a1' {0,1,8,9}
@attribute 'WtHt Percent ref median Fog, v443' numeric
@attribute 'WtHt Percent ref median WHO, v444' numeric
@attribute 'Result of measurement of resp, v447' numeric
@attribute 'Currently pregnant, v454' {0,1,9}
@attribute 'Heard about other STDs, v785' {0,1,9}
@attribute 'Ideal number of kids, v629' numeric
@attribute 'Pattern of use, v361' {1,2,3,4}
@attribute 'Region,v101' {1, 2, 3}
@attribute 'De facto place of residence, v134' {0, 1, 2, 3, 9}
@attribute 'Usual resident or visitor, v135' {1, 2, 9}
@attribute 'Sex of household head, v151' {1, 2}
@attribute 'Getting medical help for self: getting permission, v467b' {0, 1, 2, 9}
@attribute 'Getting medical help for self: getting money needed, v467c' {0, 1, 2, 9}
@attribute 'Getting medical help for self: distance to health, v467d' {0, 1, 2, 9}
@attribute 'Getting medical help for self: having to take trans., v467e' {0, 1, 2, 9}

@attribute 'Getting medical help for self: not wanting to go, v467f' {0, 1, 2, 9}
@attribute 'Getting medical help for self: concern no female hw, v467g' {0, 1, 2, 9}
@attribute anemia {-1,1}

4.2.2 Dataset construction, integration and format

We first obtained the dataset in a format of STATA the statistical data processing software. These files were then read into Python programming environment and subsequently converted to the comma-separated values (CSV) file format. The same file was also read in WEKA and converted into attribute relational file format (ARFF). Therefore our final dataset was in two formats comma separated variables CSV, for experiments in Python and ARFF for experiments in Weka.

We saved the final CSV dataset in a MySQL database, from where we were able to query and generate data files which we used in our python code for experimentation with both supervised and unsupervised machine learning algorithms. All experiments in Weka used the ARFF files whose attributes and values we have presented in this chapter.

4.3 Analysis of the data preparation process

The data preparation phase was the most involving in this study and required multiple iterations before we felt comfortable with the dataset that we had for use in the experimentation phase.

For instance, the process of generating the ARFF files that we have used to present the list of features for the full features dataset as well as the controllable features dataset was manual. We generated the ARFF file in Weka but we needed to edit each attribute so that we can specify the type of attribute that it is and values it accepts for categorical attributes. This meant that we had to revisit the data dictionary for the dataset and use it as reference in specifying all the various values that are acceptable in the categorical attributes as shown in the sub-sections 4.2 and 4.2.1. In this process if a single value gets omitted from the list of acceptable values for a categorical attribute Weka would reject that file considering it to be an ill formatted ARFF file until the error is identified and corrected.

On the other hand, for numeric attributes what was mostly required was the specification that the attribute accepts data of the numeric type. In this case there was no need to specify all acceptable attribute values a priori.

In Python on the other hand, a similar task of splitting categorical attributes based on values it takes could be achieved using the function called 'get dummies' in Pandas. The 'get dummies' function dummifies the categorical attributes and generates a set of new attributes for each value that an attribute takes. So if an attribute takes on four values, four different attributes will be made from this attribute.

The benefit of undertaking the data preparation process carefully lies in the fact that having completed the process of cleaning this dataset, it is available for use in testing

other methods without much effort in getting it ready for use in machine learning experiments.

In terms of the nature of the dataset, we learned that most of the features are categorical in nature, perhaps owing to the fact that questionnaires are used to collect such data. This realisation is important in the consideration of what methods are appropriate for such a dataset. Perhaps one can even select the categorical attributes only and build very meaningful patterns perhaps using decision trees or other transparent machine learning algorithms.

4.4 Conclusion

In conclusion, this chapter described the dataset that we used in this study for both supervised and unsupervised ML experiments. We discussed the source of the dataset, the themes in the data, data dimensionality and data cleaning process.

We described the rationale behind the selection of the attributes for inclusion in the final datasets. The first factor being the perceived usefulness of the attribute: by having data, not being redundant and having data that is not just serial numbers, date of survey or similar information. The other factor was availability of information on anemia and malaria attributes, which were our attributes of interest for the supervised ML experiments. Moreover, we made sure that all malaria and anemia attributes, though they have missing values, were included in the final dataset, which was then cleaned by rows to generate a final dataset.

Secondly, we also discussed, how we selected attributes to be included in a dataset of controllable attributes from the final dataset, based on our systematic analysis on whether that feature is controllable or not through policy intervention.

The subsequent chapters will discuss rationale, methods, and results for specific machine learning algorithms as indicated in the introductory chapter 2. In the latter chapters, using the CRISP-DM methodology we will be tackling two phases in the methodology namely modeling and evaluation to build and assess the performance of specific machine learning algorithm models.

Finally, we can state the quality and quantity of the final datasets, that we generated for both supervised and unsupervised machine learning experiments, were adequate and satisfactory for the intended experiments.

5 Clustering

The broad categories of unsupervised ML algorithms are clustering algorithms that aim at finding homogeneous groups and algorithms that aim at reducing data dimensionality. In this regard, the distinct characteristic of unsupervised machine learning is that, given a set of training data, there is no prior information on specific classes to which the instances belong. Instead, the training phase uses the entire dataset and whatever patterns emerge are analyzed afterward.

Berkhin (2006) defined clustering as the partitioning of data into groups of similar objects, which may broadly suffice as a definition for unsupervised machine learning. In this learning scenario, the goal is to identify natural clusters or groups to which all the data points in a training dataset belong. Once these clusters are known, one can use them to make predictions of cluster membership on a different testing dataset.

Consequently, the goal unlike in the supervised machine learning case, is no longer to see if we can correctly predict a class to which a data point belongs. Instead, the goal is to learn the characteristics of the data source population or sub-population to which that data point belongs. The rationale is that similar data is expected to naturally be found together in one group or within the proximity of each other. Similarly, data that is not similar clusters into separate groups or clusters.

This motivation of clustering begs the question of how do we say that data is similar to some other data and not similar to some other data, in the absence of class information. Hence, the need for this capacity to measure the degree of similarity or otherwise between data points. This measure of similarity is fundamental in the formulation of clustering methods and is thoroughly tackled in the paper by Berkhin (2006); Leskovec, Rajaraman, and Ullman (2014).

There are several ways of measuring the similarity or dissimilarity of data points using distance metrics. These various distance metrics generally view the dataset as a composition of data vectors or data points that exist in some space, the number of features in the dataset determining the data dimensionality. For instance, given a dataset X with p attributes and n data points, its dimension is p which is the number of features in the dataset (Berkhin, 2006; Leskovec, Rajaraman, and Ullman, 2014).

Distance metrics calculate the distance between data vectors, and uses that difference as a measure of similarity, i.e., vectors with a short distance between them are similar, and likewise, those with a more considerable distance between them are dissimilar. Some of the standard distance measures used in clustering algorithms are Euclidean distance and Manhattan distance, see (Berkhin, 2006).

In the other category of unsupervised ML algorithms, the goal is to reduce the data dimensionality to help in the understanding of that dataset. These methods achieve their goals by either projecting data onto a lower dimensional space to make the

data patterns understandable, or by making it possible to visualize the data in 2 dimensional plots to reveal the various relationships among the attributes. Two methods falling into this category are Principal Component Analysis and Self Organising Maps, which we will discuss later in the chapter.

Our goals in the chapter were to perform clustering on the datasets, to answer the following questions: $i)$ what is the number, composition or the nature of clusters in the datasets, $ii)$ do malaria and anemia attributes cluster together and $iii)$ how do the clustering results compare with the supervised methods experimental findings on anemia prediction.

5.1 Partition relocation clustering

The partition relocation clustering method, which we will delve into in this subsection works as follows, the beginning point of clustering are the individual data points which are assigned individually to the various clusters based on their closeness or similarity, to other data points that they get grouped with in the initial clusters (Berkhin, 2006; Leskovec, Rajaraman, and Ullman, 2014).

The process is iterative, and at each stage the algorithm checks to ensure that the data is clustered with data points similar to itself, using the various distance metrics. At every iteration data points can move to a different cluster and cluster centers are recomputed, this process continues until convergence. In this scheme, the number of initial clusters to which the data points get assigned to can either be determined a priori or otherwise. The unsupervised algorithms that we used in this study are members of this clustering method.

We employed two clustering algorithms namely k-means and expectation maximization (EM) to perform clustering on our dataset. These algorithms work as follows: $i)$ k-means algorithm allocates data points to a particular cluster, selected based on shortest Euclidean distance to a cluster center m_i, for some preselected k number of clusters. The method provides what is termed hard-labels, for cluster membership, i.e., 1 for a member, 0 for not a member of a particular cluster. On the other hand $ii)$ EM estimates the Gaussian population distribution parameters for the clusters, namely mean and standard deviation (μ_i, Σ_i) for $i = 1 \cdots k$ clusters. This particular algorithm, therefore, seeks to maximize the likelihood function $\mathbf{L}(\theta|X)$ for cluster membership. The method provides so-called soft-labels which are probabilistic estimates of class membership for each data point to all the clusters.

We will now discuss these unsupervised machine learning algorithms in more details in the following sub-sections.

5.1.1 k-means algorithm

The k-means algorithm allocates data points to a pre-determined number of clusters, based on each data point's proximity to k cluster centers. In the initialization step, k cluster centers i.e. m_j for $j = [1 \cdots, k]$, are selected at random from the data points X^i for $i = [1 \cdots, n]$ where n is the number of data points.

After that, the algorithm enters an iterative phase, where the Euclidean distance $||x_i - m_j||$ for each data point to all cluster centers is computed. Each data point

gets assigned to a cluster whose center it is closest to using the Euclidean distance measure. In other words, the data point is allocated to a particular cluster, if the Euclidean distance from that specific data point to a cluster center is the smallest. After this, the cluster centers are recomputed and moved to the central most position using the data points that are members in the clusters. The iteration stops when there are no more significant movements of data points between clusters, indicating a level of stability in the clustering.

Finally, the algorithm assigns a hard label b_i^t, to allocate the data point to a particular cluster as follows;

$$b_i^t = \begin{cases} 1, & \text{if } ||x^t - m_i|| = min_j ||x^t - m_j|| \\ 0, & \text{otherwise} \end{cases}$$

The k-means means algorithm suffers from the weakness that it can converge quickly to a local solution if the cluster centers are initialized very close together. Therefore, it is necessary to carry out several experimental runs when using the algorithm, to find the best solution.

However, the algorithm is relatively easy to understand and to implement. This simplicity has led to the development of several variations of the algorithm.

Expectation maximization (EM)

EM uses a probabilistic approach and aims to estimate a vector of parameters Φ that maximize the sample's likelihood given the population parameters. That is to say; given a dataset of observed variables X and hidden variables Z with unknown parameters Φ, EM will estimate the maximum likelihood $L_c(\Phi|X, Z)$, given the parameters Φ and variables X and Z. Essentially, the EM algorithm seeks to maximize the parameter vector Φ, that maximizes likelihood $\mathbf{L}(\theta|X)$, for the observed data X (Alpaydin, 2010).

The algorithm carries out the estimation taking into account the impact of the latent variables Z. The process therefore, has two major steps namely **estimation step** and **maximization step**. The first step in the algorithm is the estimation of the log likelihood L_c, using l because L_c cannot be estimated directly because it contains the hidden variables Z (Alpaydin, 2010). Thereafter, maximization of the likelihood that the data comes from the model represented by the groups with those distribution parameters is done. Below is a breakdown of these steps:

$$E - step : l(\Phi|\Phi^l) = E[L_c(\Phi|X, Z)|X, \Phi^l]$$

and subsequently iterate over;

$$M - step : \Phi^{l+1} = argmax_\Phi l(\Phi|\Phi^l)$$

The EM algorithm, is premised upon the idea that, data is made up of several Gaussian sub-populations, and each population or sub population has its own set of distribution parameters Φ i.e. (μ_i, Σ_i) for $i = 1 \cdots k$, where k is the number of clusters. The task of the algorithm is to learn these population or sub-population distribution

parameters, as well as, the number of clusters that naturally exist in the dataset. The Gaussian mixture model for the data, in this case, is represented by the formulation below;

$$p(x) = \sum_{i=1}^{k} p(x|G_i)p(G_i) \tag{5.1}$$

Here the G_i are individual clusters, for $i = 1 \cdots k$ different clusters. The goal of the algorithm is to estimate clusters or groups, to which data points naturally belong, by estimating population distribution parameters for these groups or clusters.

Just like the k-means algorithm, EM is supplied with the number of clusters a priori. However, the true number of clusters may be learned from the data, even though an initial number of clusters is provided to initialize the algorithm. This pattern may be seen through the creation of some clusters having very few members, indicating that perhaps those members might belong to some nearby clusters.

The algorithm then finally allocates a probabilistic label to each data point. Once the algorithm converges, as given in equation 5.2, where S_i and m_j are part of the parameters from the vector Φ i.e. (μ_i, Σ_i) based on the observed estimates of X;

$$h_i^t = \frac{exp[-(1/2S_i^2)||X^t - m_i||^2]}{\sum_j exp[-(1/2S_j^2)||X^t - m_j||^2]} \tag{5.2}$$

As can be noted here, h_i^t takes a value between $[0, 1]$ and is a soft or probabilistic label for cluster membership. This probabilistic membership allocation makes the algorithm flexible and useful for guiding a user on other potential clusters to which a given data point might belong.

5.2 Principal component analysis (PCA)

Principal Component Analysis is not necessarily a data clustering algorithm. However, it is an unsupervised learning method that can be used to complement clustering results on the basis that PCA identifies principal components. Principal components analysis, is a data dimensionality reduction technique, that assesses each feature's contribution to the model (Alpaydin, 2010). The components in PCA are new features that the method generates by combining the various features that are present in a given dataset, for example, if p features are input, PCA outputs d features for $d \leq p$.

At the heart of PCA is the generation of a covariance matrix that assesses each feature's impact on all the other features. The method calculates a covariance matrix and identifies eigenvalues, and eigenvectors which are used to distinguish the principal components in the data, e.g., eigenvectors with large eigenvalues are the principal components.

Each eigenvector is a linear combination of the input features from the dataset. The algorithm then projects the dataset to a lower dimensional space, using components

that account for most of the variation in the dataset. Through this process, the algorithm identifies a small set of feature components that can explain most of the variance in the dataset, and the rest of the features get discarded.

Furthermore, because it is difficult to visualize data beyond three dimensions, PCA helps in providing a way to simplify visualization of a high dimensional dataset, by making it possible to plot the most dominant or informative feature mixtures which are the principal components.

However, PCA results might be challenging to interpret for a large set of features, because their output is a new set of attributes, which are generated from the combination of the original attributes.

5.3 Self-organising maps (SOM)

The final unsupervised ML algorithm that we'll discuss is the self-organizing maps, which were developed by Kohonen (1998). He defines them as a nonlinear method based on neural networks that implement a mapping of higher dimensional data to a lower dimensional array.

The algorithms compress a lot of information and generate representations of the relationships in the data geometrically, which makes it easier to analyze. Furthermore, Kohonen (1998) specifies that SOM is both a similarity graph and a clustering tool where similar data are typically found close together.

Due to their clustering and similarity graph approach, these methods have been efficiently used to visualize and graphically present results using geographical maps, see (Koua and Kraak, 2004; Sarlin, 2012). For example, Koua and Kraak (2004) tackled the problem of geographically exploring DHS datasets to understand socio-economic, demographic and health patterns. On the other hand, Sarlin (2012) worked on a question of visualizing the progress towards attainment of MDGs for Sub-Saharan Africa. In their results, they demonstrated that it is possible to visually identify trends across countries having similar socio-economic profiles on the issues that were under investigation.

5.4 Methods

In this section, we outline the methods that we employed for the unsupervised learning phase of our study.

We detail these issues in the following subsections: in subsection 5.4.1 we highlight the datasets that we used, subsection 5.4.2 discusses the ideal cluster number identification process and in sub-section 5.4.3 we discuss clustering patterns with an emphasis on specific attributes of interest.

5.4.1 Datasets used

As indicated in chapter 4, on page 37, we generated a full features dataset, that we obtained through the data preparation phase of the CRISP-DM methodology. We

saved our dataset in a MySQL database and made an identical copy that was formatted and stored in ARFF files for Weka experiments. Hence to perform clustering in this phase, we queried the database table holding the dataset for the Python experiments, and used the ARFF file for the Java-based Weka data mining platform experiments.

The dataset had $10,039$ data points by 133 attributes. The dataset had anemia cases distributed as follows; 87 were severe anemia, 786 were moderate anemia, $2,496$ were mild anemia and $6,670$ were non-anemic cases.

5.4.2 Cluster number identification

The goal for cluster number identification experiments was to identify the appropriate number of clusters for our dataset.

Therefore, to begin the process we used SOM and PCA to explore the dataset and identify the potential ideal cluster number. After that, we used the EM and k-means algorithms for further clustering starting with ideal cluster number size from SOM experiments.

We conducted SOM and PCA experiments in Weka, using our ARFF version of the dataset. On the other hand, we did k-means and EM experiments on both platforms. We generated result files for experiments on both algorithms, and these result files were analyzed and visualized, to gain insights into the clustering results.

For EM and k-means experiments, we began by writing Python code to perform clustering of our datasets, starting from the initial cluster number that was identified in SOM and increasing the cluster numbers until clustering performance metrics showed no benefit in performing further clustering. For each clustering iteration, we obtained measures of AIC and BIC, which were used to assess how good the clustering result is. AIC and BIC rely upon the likelihood function for a measure of the goodness of fit of a particular clustering scenario.

Using the AIC and BIC results, we were able to identify the ideal number of clusters for us to use in the subsequent phase for comparison of EM and k-means clustering.

After that, we rerun the clustering in Weka using the ideal cluster sizes that we discovered, to see if there was a difference in the outcome.

5.4.3 Clustering patterns

During the clustering process we used the two clustering algorithms to predict cluster membership for each data point and saved that information. Using the cluster results obtained, we compared the performance of the two algorithms to assess if their clustering patterns were different. Later on, we compared the two algorithms by assessing the sizes of the clusters that each method produced and also the cluster composition of each cluster.

Furthermore, for each iteration of clustering for the two clustering algorithms, we created a table in the MySQL database of the results which had a predicted cluster membership for each data point. We stored cluster membership in a column which we added as a new column in the results table which we named cluster.

The predicted class membership column was afterward used in the analysis, where SQL queries were run on the MySQL database clustering results tables to assess cluster membership and patterns for select attributes of interest from our dataset.

We proceed to present the findings in the next section.

5.5 Results

We provide the results of the unsupervised ML experiments in this chapter.

Our first analysis was to assess the sizes of clusters that we were able to generate using the ideal number of clusters that we identified. We therefore detail the number of clusters identified in sub-section 5.5.1.

Secondly, we were interested in assessing whether there was a significant difference in the clustering patterns for the two algorithms of interest. We, therefore, discuss cluster membership and compare the performance of the algorithms with a focus on the clustering of specific attributes of interest in subsection 5.5.2.

Our analysis also involved assessing how various attributes of interest clustered in the different clusters. Specifically, we checked how anemia, malaria and other health-related attributes got clustered. We were interested in finding out for instance whether anemic cases tended to cluster together or otherwise.

5.5.1 Number of clusters identified

We begin by presenting the performance of the Principal Components Analysis, which identifies the attribute combinations that explain the most considerable variance in the dataset.

PCA

Below are the top 10 principal components identified in the dataset. These principal components indicate which combination of features is significant in the dataset using a measure of the EigenValue for the principal component.

Furthermore, a clear relationship between the components is observable, and overall each component has one major recurring theme in its combination of attributes.

In the generation of the PCA model, the categorical attributes get split into the individual values which they assume. Therefore, an attribute with five values will be split into five different values for each of the five values, for example attribute 'Region' takes 3 values $[1, 2, 3]$ it therefore breaks into '$Region = 1$', '$Region = 2$' and '$Region = 3$'. Numeric attributes, for example Age are treated as a single attribute. Each principal component is an Eigenvector whose direction in the feature space is determined by the sign and magnitude of the coefficients.

We present the principal components below, whereby we have provided a label, the eigenvalue and a summary of the dominant theme in the principal component. The components are identified by PC short for principal component and a number.

PC 1, Eigen Value = 0.9514 main theme is **Demographic**
0.151 (Number of living children, v218 = 0) + 0.151 (Total children ever born, v201 = 0) + 0.147 (Living children n current preg,v219 = 0) - 0.145 (Living children + curr preg 6+, v220) + 0.144 (Currently/formerly/never married, v502= 0)

PC 2, Eigen Value = 0.9136 main theme is **Economic**
0.197 (Usual resident or visitor, v135) = 1 - 0.197 (Has bicycle, v123 =7) - 0.197 (Type of toilet facility, v116=97) - 0.197 (Has refrigerator, v122=7) - 0.197 (Has motorcycle, v124=7)

PC 3, Eigen Value = 0.8835 main theme is **Economic**
0.186 (Wealth index factor score,v191) + 0.159 (Wealth index, v190=5) + 0.15 (Has electricity, v119 = 1) + 0.15 (Main floor material, v127 = 34) - 0.145 (Type of place of residence, v102=2)

PC 4, Eigen Value = 0.8591 main theme is **Demographic**
0.201 (Current age - respondent, v012) - 0.198 (Date of birth (CMC), v011) - 0.198 (Respondents year of birth, v010) + 0.158 (Entries in birth history, v224) - 0.153 (Fertility preference, v602=1)

PC 5, Eigen Value =0.8382 main theme is **Health practice**
-0.262 (Getting medical help for self: getting permission, v467b = 0) - 0.256 (Getting medical help for self: concern no female hw,v467g = 0) + 0.246 (Type of cooking fuel, v161 = 8) - 0.243 (Getting medical help for self: not wanting to go, v467f = 0) - 0.236 (Type of cooking fuel, v161=7)

PC 6, Eigen Value =0.8219 main theme is **Behaviour and Knowledge**
0.24 (Chewing tobacco, v463c = 9) + 0.24 (Uses snuff,v463d = 9) + 0.24 (Smokes pipe, v463b = 9) - 0.236 (Smokes pipe, v463b = 0) + 0.224 (Heard sexual transmit. disease, v750=9)

PC 7, Eigen Value =0.8068 main theme is **Demographic**
-0.23 (Entries in health table,v418 = 1) - 0.23 (Births in last five years, v208 = 1) - 0.23 (Entries in maternity table, v417 = 1) - 0.23 (Entries in height/weight table,v419 = 1) + 0.217 (Births in last five years, v208 = 2)

PC 8, Eigen Value =0.7927 main theme is **Geographic and contraception**
0.234 (Region, v024 = 3) +0.234 (Region, v101 = 3) + 0.231 (De jure region of residence, v139 = 3) +0.185 (Current use by method type, v313 = 0) - 0.185 (Pattern of use, v361 = 1)

PC 9, Eigen Value =0.7792 main theme is **Anthropometric**
-0.358 (Wt/Ht Std deviations(resp) DHS ,v444a) - 0.358 (Wt/Ht Percent ref. median (DHS),v442) - 0.357 (WtHt Percent ref median Fog, v443) - 0.357 (WtHt Percent ref median WHO, v444) - 0.345 (Rohrers index for respondent, v446)

PC 10, Eigen Value =0.767 main theme is **Geographic**
-0.234 Region, v101 = 2) - 0.234 (Region, v024 = 2) - 0.232 (De jure region of residence, v139 = 2) + 0.209 (Region, v101 = 3) + 0.209 (Region, v024 = 3)

Upon surveying these components, it is clear that each of these principal components has a consistent message that it communicates. The second, observation is that there are recurring themes in the components and explicit relationships between them.

For example, if we analyze principal component PC 3, we note that this component isolates respondents that fit a particular economic class. Within this component, we have the following attribute values: *'Wealth index, v190=5'* is richest on a metric scale where 1 is 'poorest', 2 is 'poorer', 3 is 'middle', 4 is 'richer' and 5 is 'richest'. Secondly, *'Has electricity, v119 = 1'* for has electricity is part of the component, followed by, has cemented floor in the home *'Main floor material, v127 = 34'* is selected also, and finally, lives in rural area *'Type of place of residence, v102=2'*. This component can be seen to isolate a specific profile of respondent.

Moreover, as already mentioned, there are specific relationships that exist between these principal components. Principal components 1, 4 and 7 fall in the same thematic area, though it is clear from the attribute values that they have different areas of focus. Specifically, component 1 isolates those that have never been married and are childless. Component 4 is about the age of respondent, while component 7 focuses on the number of children that were born to the respondents.

These PCA results, therefore, reveal the nature of the dataset and isolates some key feature combinations that help explain the variance in the dataset. We now proceed to review SOM clustering results.

SOM

We present the results for SOM, on the full features dataset that was obtained using the Weka DM platform. We begin by presenting a table showing clustering results for the SOM algorithm in table 5.1.

Cluster	Instances
0	2,847 (28%)
1	2,053 (20%)
2	2,729 (27%)
3	2,410 (24%)

TABLE 5.1: *Clusters generated by the SOM algorithm on the full features dataset.*

Cluster 0 is the most massive cluster followed by cluster 2, the least cluster being cluster 1. However, the cluster sizes do not seem to be too different from each other from a size perspective considering that each cluster has about 20% of the cases.

EM and k-means

Having experimented with building cluster models with clusters in the range $[4, 20]$ for the EM clustering algorithm, which we carried out to identify the ideal number of clusters for our dataset, we collected measures of AIC and BIC for each scenario. Below is a plot of the AIC and BIC measures results presented graphically, for ease of identifying suitable cluster sizes using both metrics.

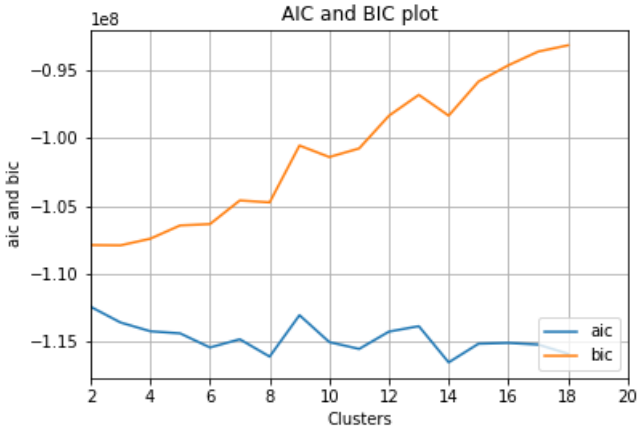


FIGURE 5.1: *Comparative plot of AIC and BIC for various cluster sizes. Smaller values for both measures are indicative of a good cluster size. $x-$axis are different cluster sizes in range $[4, 20]$. On the other hand AIC and BIC are on $y-$axis.*

We observe that clusters from size 4 to 8 have small BIC values indicative of the fact that they are ideal cluster sizes with acceptable likelihood. We, therefore, proceed to analyze these cluster sizes in the subsequent sub-section.

5.5.2 Clustering patterns

SOM

Below are the images showing the clustering patterns for some of the attributes of interest. The features that we showcase here have been selected on the basis that they were part of the top 10 principal components that were presented earlier. The different colors in the images are for the four clusters, which were identified using this SOM algorithm.

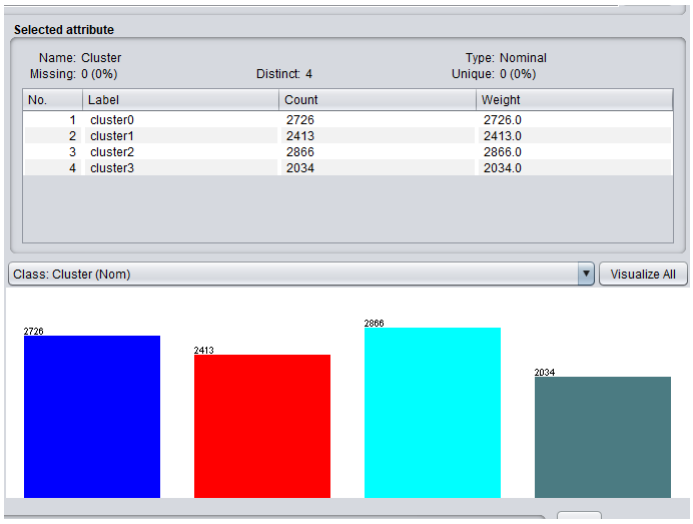


FIGURE 5.2: *Clusters generated using SOM where the different colored blocks represent different clusters.*

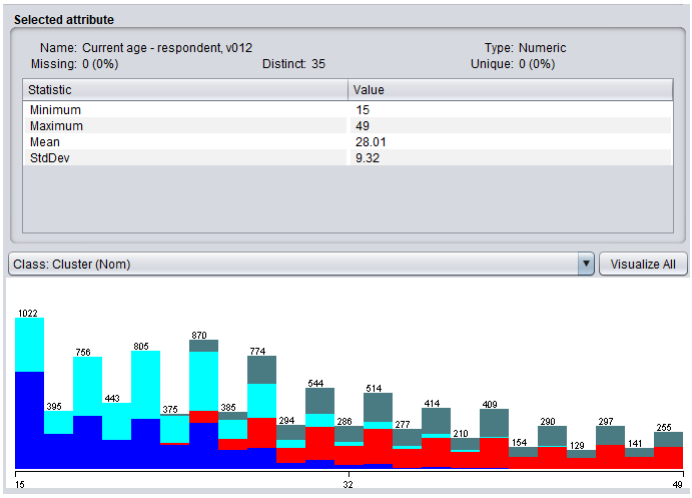


FIGURE 5.3: *Current age of respondent.*

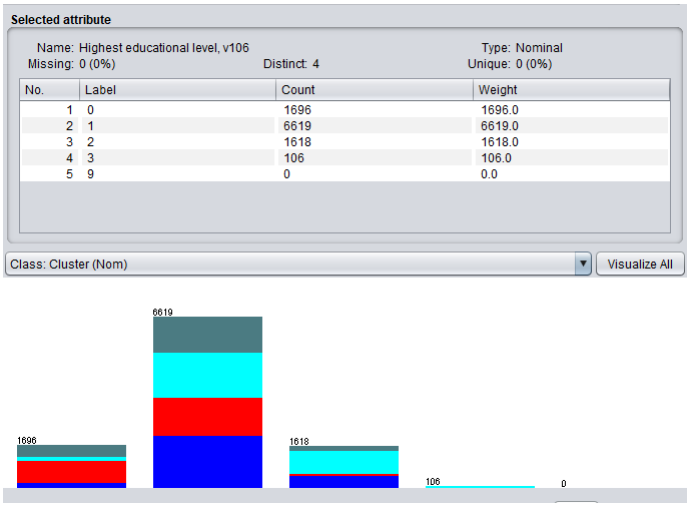


FIGURE 5.4: *Education levels are shown here, 0 is no education, 1 is primary, 2 is secondary and 3 is higher education. The various education levels did not cluster in any distinguishable pattern, they spread over all clusters.*

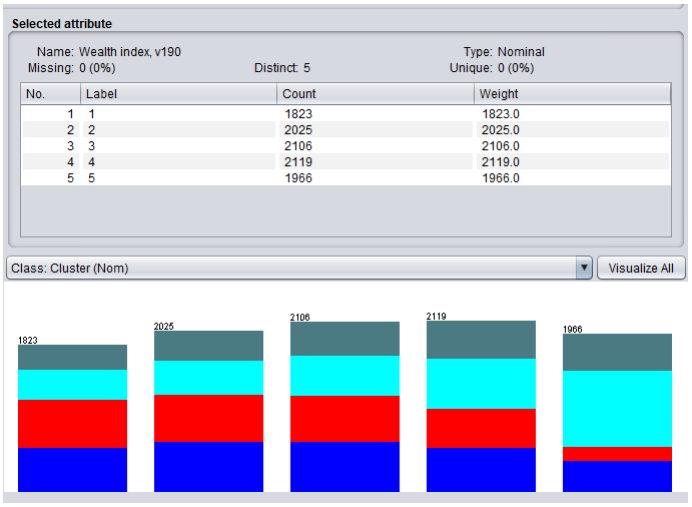


FIGURE 5.5: *Wealth index attribute ranging from 1 poorest to 5 richest. Likewise, this attribute did not have any unique clustering pattern.*

The clustering patterns of SOM seemed to produce well-distributed data points in all clusters. Furthermore, there does not appear to be a clear separation of the cases using the anemia and malaria-related attributes like *'anemia status'*, *'usage of bednets'* and other related attributes.

However, health practices captured in $v467$ seems to have some distinct difference in how the cases having different values for these attributes were distributed in the

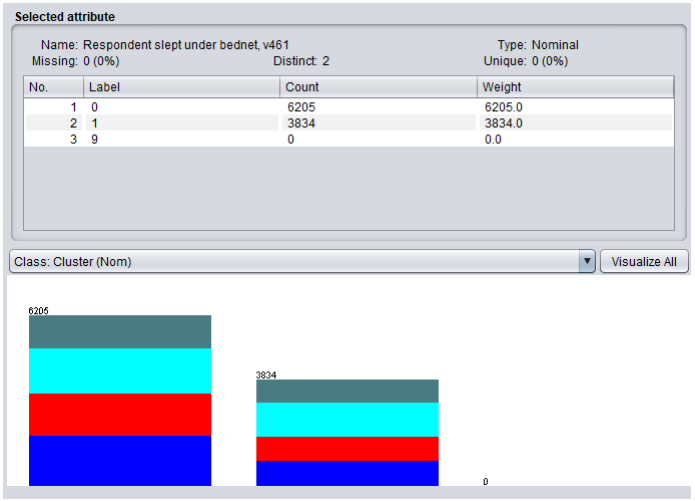


FIGURE 5.6: *Attribute v461 having information on whether respondent slept under bednet last night. 0 is no, 1 is yes and 9 is missing information.*

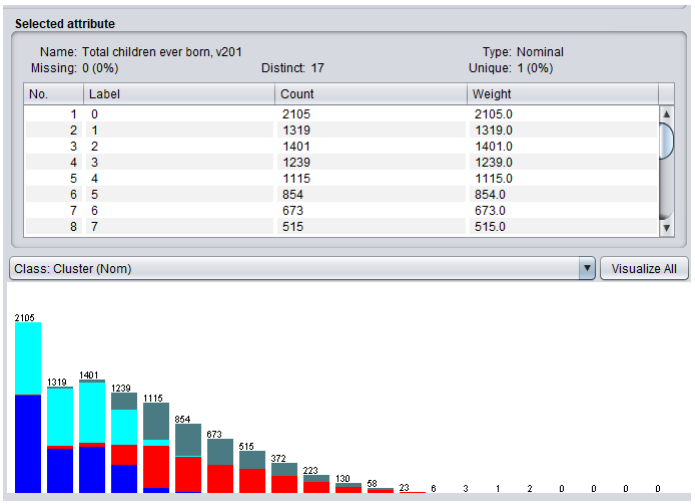


FIGURE 5.7: *Attribute v201 having information on the number of children the respondent has ever had.*

clusters. We use the cluster number 4 identified as a basis for clustering in the subsequent section where k-means and EM were used.

EM and k-means

We provide results of clustering that we performed using the EM and K-Means algorithms using the identified ideal cluster sizes. The results provide clustering details for cluster sizes ranging from $[4, 8]$, which were identified based on their AIC and BIC plots presented earlier.

In the subsequent tables, we provide results on a select set of attributes, to show their distribution in the clusters. We state the sizes of the clusters that were formed for each scenario.

The first attribute to consider is the *'anemia level'* attribute which has two categories. One category is for any anemia, i.e., severe, moderate and mild anemia while the second category is for not anemic.

The second attribute to analyse is *'the type of bednets used'* having responses 0 for *'no net'*, 1 is for *'insecticide treated mosquito net'*, 2 is for *' treated and non treated bednet'* and 3 is for *'non treated net'*.

The third attribute assessed is whether respondent used the antimalarial drug Fansidar. Finally, the last attribute is *'altitude'* which is a measure of meters above sea level where the respondent lives.

		Anemia		**Bednet used**				**Fansidar**		**Altitude**	
Clust	**Size**	0	1	0	1	2	3	0	1	< 500	≥ 500
0	839	623	216	434	341	–	64	281	53	126	713
1	8,269	5,507	2,762	5,145	2,571	2	531	3,970	4,299	1,377	6,892
2	245	156	89	148	81	–	16	101	144	29	216
3	686	404	282	478	178	–	30	331	336	116	570

TABLE 5.2: *Analysis of cluster membership for an EM* 4 *cluster scenario.* 82% *of the data points are in cluster* 1 *with the remaining* 20% *spread over the other* 3 *clusters.*

		Anemia		**Bednet used**				**Fansidar**		**Altitude**	
Clust	**Size**	0	1	0	1	2	3	0	1	< 500	≥ 500
0	5,184	3,396	1,788	3,120	1,697	2	365	2,384	2,787	1,037	4,147
1	1,252	861	391	797	398	–	61	611	639	111	1,141
2	3,421	2,295	1,126	2,193	1,008	–	220	1,616	1,796	471	2,950
3	182	118	64	99	68	–	15	72	110	29	153

TABLE 5.3: *Analysis of cluster membership for an K-Means* 4 *cluster scenario.* 52% *of the data points are in cluster* 0 *while* 34% *and* 12% *are in clusters* 2 *and* 1 *respectively.*

		Anemia		**Bednet used**				**Fansidar**		**Altitude**	
Clust	**Size**	0	1	0	1	2	3	0	1	< 500	≥ 500
0	649	376	273	452	156	–	32	313	317	101	548
1	2,727	1,598	1,129	1,975	572	–	160	1,321	406	438	2,289
2	178	116	62	119	50	–	9	82	96	19	159
3	6,052	4,265	1,787	3,437	2,195	2	418	2,864	3,188	1,016	5,036
4	433	315	118	204	189	–	42	103	325	74	359

TABLE 5.4: *Analysis of cluster membership for an EM* 5 *cluster scenario.* 60% *of the data points are in cluster* 3*. The other large cluster is cluster number* 1 *with* 27% *of the data points.*

		Anemia		Bednet used				Fansidar		Altitude	
Clust	**Size**	0	1	0	1	2	3	0	1	< 500	≥ 500
0	2,940	1,957	983	1,940	822	–	178	1,385	1,548	352	2,588
1	2,703	1,831	872	1,560	935	–	208	1,237	1,460	566	2,137
2	182	118	64	99	68	–	15	72	110	29	153
3	2,996	1,928	1,068	1,839	955	2	200	1,392	1,597	590	2,406
4	1,218	837	381	767	391	–	60	597	619	111	1,107

TABLE 5.5: *Analysis of cluster membership for an K-Means* 5 *cluster scenario. This scenario has the largest cluster with* 30% *of the data points in cluster* 3 *followed by clusters* 0 *and* 1 *with* 29% *and* 27% *of the data points respectively.*

		Anemia		Bednet used				Fansidar		Altitude	
Clust	**Size**	0	1	0	1	2	3	0	1	< 500	≥ 500
0	535	299	236	386	121	–	28	242	274	83	452
1	2,315	1,358	957	1,680	488	–	147	1,075	1,240	391	1,924
2	347	242	105	176	139	–	29	104	237	66	281
3	807	567	240	416	331	–	60	292	515	119	688
4	5,744	4,000	1,744	3,374	1,996	2	372	2,872	2,902	951	4,793
5	291	205	86	170	96	–	25	128	163	38	253

TABLE 5.6: *Analysis of cluster membership for an EM* 6 *cluster scenario.*

		Anemia		Bednet used				Fansidar		Altitude	
Clust	**Size**	0	1	0	1	2	3	0	1	< 500	≥ 500
0	2,791	1,771	1,020	1,727	879	2	183	1,297	1,487	564	2,227
1	947	664	310	621	307	–	46	479	493	101	873
2	1,106	769	337	641	381	–	84	527	578	169	937
3	176	115	61	95	67	–	14	68	108	23	153
4	2,410	1608	802	1,636	640	–	134	1,133	1,269	236	2,174
5	2,582	1,743	839	1,485	897	–	200	1,179	1,397	555	2,027

TABLE 5.7: *Analysis of cluster membership for an K-Means* 6 *cluster scenario. This scenario has smaller clusters compared with previous case for* 5 *clusters. Here we have* 28%, 26%, *and* 24% *of the data points in clusters* 1, 5 *and* 4 *respectively.*

Clustering of anemia

We analyze how the specific anemic and non-anemic cases were clustered in the 4 and 6 cluster size scenarios. The results of this clustering are presented in tables

where the various anemic levels are given in columns for severe, moderate, mild and non-anemic. We were interested in identifying the most dominant clusters for each of the 87 severe anemic, 784 moderate anemic, $2,498$ mild anemic and $6,670$ non-anemic cases.

	Anemia level (EM)				**Anemia level (k-means)**			
Cluster	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**
0	4	43	169	623	42	416	1,330	3,396
1	68	652	2,062	5,487	10	97	284	861
2	2	20	67	156	34	257	835	2,295
3	13	71	198	404	1	16	47	118

TABLE 5.8: *Distribution of anemia level cases in EM* 4 *cluster scenario. Cluster number* 1 *is the largest cluster and has most of the data points for all levels of anemic as well as non-anemic. Distribution of anemia level cases in K-Means* 4 *cluster scenario. In this scenario clusters,* 0 *and* 2 *are dominant having most of the cases clustered in them. However, there is no clear separation of the various anemia cases into specific clusters.*

	Anemia level (EM)				**Anemia level (k-means)**			
Cluster	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**
0	6	56	175	298	25	249	746	1,771
1	34	229	694	1,358	8	77	225	664
2	3	19	83	242	4	85	248	769
3	6	55	179	567	1	15	45	115
4	35	408	1,301	4,000	29	174	599	1,608
5	3	19	64	205	20	186	633	1,743

TABLE 5.9: *Distribution of anemia level cases in EM* 6 *cluster scenario. Distribution of anemia level cases in K-Means* 6 *cluster scenario.*

5.5.3 A comparison of the clustering algorithms

We assess the clustering patterns for the three algorithms SOM, EM, and k-means on the 4 cluster scenarios. The sizes of the clusters given in ascending order of clustering algorithms were $2,053$, $2,410$, $2,729$ and $2,847$ and for the SOM 245, 686, 839 and $8,269$, and for the EM 182, $1,252$, $3,421$, $5,184$ and for the k-means.

The SOM algorithm generated well spread out clusters while the EM and k-means yield mainly two massive clusters and other very small clusters.

This pattern continues for EM and k-means across the other cluster sizes that we considered, giving a consistent clustering pattern for these algorithms. On the other hand, the largest clusters in EM though tended to be much larger than the other

clusters by a larger margin. Moreover, the largest cluster in EM was always smaller than the largest cluster in k-means.

5.6 Discussion

We now discuss the result of clustering experiments that we presented in this chapter.

5.6.1 Cluster size and patterns

Going by the definition that states that a cluster is a natural grouping of similar objects by Berkhin (2006), the identified number of clusters, therefore, is an indication of what natural groups exist in a dataset. The SOM identified four clusters while EM and k-means performed reasonably well from 4 clusters up to 8.

In our case, we expected that we would see the consistent separation of the groups according to specific features of interest. For example, we expected to see different levels of being anemic in different clusters or a difference of clustering between those that took antimalarials in pregnancy or not in different groups.

On the contrary, our findings did not confirm this opinion. However, we observed that some attributes falling under demographic and health practices features clustered in significantly different ways. If one revisits the graphics of clustering given earlier, this fact will be made abundantly clear.

For example in figure 5.3 showing the age of respondents per cluster, we observe that young women between ages of 15 and 30 clustered in cluster 0 and cluster 2.

Likewise for the wealth index in figure 5.5, those ranked as richest clustered mostly in cluster 2. Further in graphic 5.7 those that had 0 or 1 child clustered in clusters 0 and 2 as well.

However graphic 5.6 on whether respondents slept under a bednet last night, did not have any distinct separating ability as mentioned earlier that specific anemia and malaria did not have a unique clustering pattern, unlike the other features.

Moreover, we also assessed the breakdown of clustering results for EM and k-means to assess how the clustering patterns for *anemia level*, cluster altitude, *use of bednets* and *use of fansidar anti-malarial during pregnancy* was. The results for different cluster sizes showed no significant difference in clustering patterns at all. This pattern meant that clusters that had more data points overall tended to have more of every specific case. For example, a large cluster which has more mild anemia cases, it would also have more moderate anemia cases, and more non-anemic cases refer to table 5.9 and 5.8, thereby making it impossible to make a statement as to whether specific anemic levels clustered together or otherwise.

5.6.2 Principal components

The PCA results revealed a set of attributes that are most influential in the features projection and cluster modeling process. Economic status, education level, the region of residence and demographics were some of the themes which stood out from

the PCA modeling phase. These results are consistent with the clustering results presented earlier. Mainly specific essential attributes as manifested in the PCA components were also dominant in the clustering results as reflected in the isolated themes. For example, principal component number 3 which isolated women of a particular economic class are related to clusters 0 and 2 under SOM results as discussed above which also picked a similar profile of women.

This chapter answers the questions on what is the correct cluster size, what are the clustering patterns for the features of interest in the study primarily those on anemia and malaria.

We learned that clustering patterns were not definitive for anemia and malaria-related attributes.

However, we also discovered that other sets of features had a clear clustering pattern which emerged in the analysis.

We identified and presented, a set of themes under which some dominant features fall, this information was valuable to possess as we moved to the classification experiments whose results we present next.

6 Support Vector Machines (SVM)

Support vector machines (SVM) are arguably among one of the most popular machine learning algorithms. These algorithms are used for solving various problems, such as classification, regression, density estimation and data dimensionality reduction (Alpaydin, 2010).

It is therefore not surprising that SVM was named among the top 10 machine learning algorithms at an IEEE data mining conference in 2006, and was compared to the other popular algorithms in a discussion by Ramakrishnan et al. (2009).

Furthermore, the method seems to still be a subject of active research in recent times as will be seen from the literature that we briefly mentioned in this chapter and our discussion on related work provided in chapter 2. However, when one searches for SVM applications in relevant scholarly sources on the Internet for anemia or malaria prediction, very few results are obtained, indicating that perhaps this area has not been explored extensively.

In line with the guidance from the CRISP-DM methodology in this chapter, the goal was to tackle the model building and model evaluation phases of the methodology as was outlined in chapter 2. We, therefore, completed the model building and model evaluation in the Python programming platform, and in Weka data mining platform, to ensure that the models learned are realistic, useful and can be reproduced.

In this part of our study, the aim was to use support vector machines for two tasks: firstly, we used SVM for building predictive anemia models. We generated two types of SVM models, the first being those that used the entire cleaned dataset and the second type of model was built for a sub-dataset of controllable features. These models were meant to answer two research questions.

The first question was, can an SVM model be built using the DHS dataset to predict anemia. In addition to this, we were interested in finding the best parameters for this SVM model, i.e., what are the best values for γ and C, and for which specific kernel types. This question was answered using the clean full features dataset as described in the data cleaning section in chapter 4, which we obtained following the first three phases of the CRISP-DM methodology.

The second question was, can a predictive SVM model be built using a subset of controllable features only from the dataset. The answer to this second question is necessary for the study to comply with requirements for two phases of the CRISP-DM methodology's outputs, namely identification of business value and model deployment.

The second question was answered using the set of controllable attributes subset dataset which was described in the data cleaning section in chapter 4 subsection 4.2. The model generated on the controllable features dataset is what we wanted

to deploy in the final phase of the CRISP-DM methodology called the deployment phase, where we have to specify the set of best features from the dataset and best model hyper parameters for the winning SVM models that can be deployed.

Furthermore, we were also interested in measuring the accuracy of the models, identifying the best parameters to use, and which specific controllable features from the subset dataset are more influential and useful in the predictive ability of the model.

The rest of the chapter has sections organized as follows; section 6.1 provides a brief discussion of support vector machines. section 6.2 provide a detailed discussion of the methods used, section 6.3 contains results, section 6.4 discusses the results and concludes the chapter.

6.1 Support vector machines (SVM)

In chapter 2 section 2.1.2, we discussed the fundamentals of the statistical learning theory on which SVM algorithms are based. We also discussed the principles of empirical and risk minimization, as well as the VC dimension, which are all relevant in the formulation of SVM.

In a learning problem, given a set of labeled training data, the learning problem was shown as being made up of three building blocks (Vapnik, 1999). The first is a *generator* of random vectors $\mathbf{x}$ that are drawn from an unknown probability distribution $P(\mathbf{x})$. The second is a *supervisor* or trusted source that generates an output value y drawn from an unknown distribution $P(y|\mathbf{x})$ for every vector $\mathbf{x}$. Finally, there is a set of *learning machines* which are represented as functions $f(\mathbf{x}, \alpha)$ (Vapnik, 1999). In this scenario the *supervisor* knows the class for each data instance, while the *learning machine* seeks to estimate or predict the class using its learned parameters α for the distribution.

Support vector machines, are one such supervised machine learning algorithm $f(\mathbf{x}, \alpha)$ and therefore require a training dataset as was defined earlier in equation 2.1. Given such a labeled training dataset, with p attributes, the algorithm views each data instance as a point in some p dimensional space. In such high dimensional spaces, the separation between the data instances is given by a hyperplane which can classify the data, on its either side as being positive $+1$ or negative -1.

The decision boundary of SVM is based on the linear function $y = \mathbf{x}.\mathbf{w} + b$, where y is the class to which the data point belongs i.e. $\{-1, +1\}$, $\mathbf{x}$ is the data vector, $\mathbf{w}$ is the weight vector and b is the bias input. Moreover, SVM will identify a separating hyperplane between two classes $\{-1, +1\}$ for data that is linearly separable. On the other hand, in the case of non linearly separable data, the algorithms project the data into higher dimensional space, where they become separable, and a separating hyperplane is similarly identified in that new high dimensional feature space.

In both cases, however, the algorithms maximize the margin between the hyperplane and data from the two classes using quadratic optimization techniques. Furthermore, SVM also identifies the data points that lie on the margin boundary which

are called support vectors, because these particular data points are taken to be supporting the decision boundary, which is the separating hyperplane. Therefore after training once identified the support vectors, which are just some of the data instances in the dataset, can be used for classifying new unseen datasets, because they represent the decision boundary separating the classes.

The strength of the algorithm lies in the fact that SVM can handle both linearly separable datasets, as well as linearly non-separable datasets. Furthermore, the algorithms have an added feature of robustness, which allows them to handle errors through the usage of a regularization parameter, which favors generalization ability, while at the same time penalizing mistakes. We will now proceed, to discuss two versions of the SVM, one for a linearly separable case and another for the non linearly separable case (Ramakrishnan et al., 2009).

6.1.1 Linearly separable SVM

Linearly separable support vector machines, aim to find an optimal separating margin that maximizes the distance between the classes. The general idea is as follows: given a training dataset such as in equation 2.1 on page 6, the support vector machine algorithm finds an optimal hyperplane $[(w \cdot \mathbf{x}_i) - b = 0]$ that separates the data points $\mathbf{x}_i$ into classes $y_i \in \{-1, +1\}$. Where, $\mathbf{w}$ is a weight vector and is normal to the separating hyperplane, $(w \cdot \mathbf{x}_i)$ is a dot product of w and $\mathbf{x}_i$, and $(b/\|w\|)$ is the perpendicular distance from the hyperplane to the origin, and $\|w\|$ is the euclidean norm of w. The formulations here follow that of Vapnik (1999).

As in the preceding discussion our parameter set α, is given by the weights $\mathbf{w}$ and bias b respectively. The SVM algorithm is depicted in the figure below:

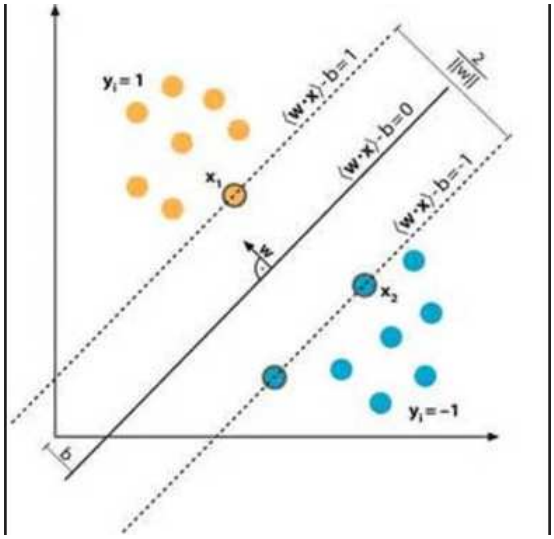


FIGURE 6.1: *This figure visually depicts the ideal linear separable SVM in 2D space. The separating hyperplane and decision boundary are clearly depicted by the middle line. Data points x_1 and x_2 are the support vectors for the two classes. w is the weight vector perpendicular to the decision boundary. Source* `http://www.precision-crop-protection.uni-bonn.de/gk_research/project_3_06/image_3.jpg`

In order to have maximum separation all training data must be in either of the classes $\{-1, +1\}$ and hence satisfy either of the following in-equalities:

$$(\mathbf{w} \cdot \mathbf{x}) - b \geq +1 \qquad \textit{for } y_i = +1 \tag{6.1}$$

$$(\mathbf{w} \cdot \mathbf{x}) - b \leq -1 \qquad \textit{for } y_i = -1 \tag{6.2}$$

These inequalities can also be compactly rewritten as:

$$y_i[(\mathbf{x} \cdot \mathbf{w}) - b] \geq 1, \qquad y_i \in \{-1, 1\} \tag{6.3}$$

The optimal hyperplane maximizes the function 6.3 with respect to w and b to get a large margin;

$$\Phi(w) = \frac{1}{2}\|w\|^2 = \frac{1}{2}(w, w) \tag{6.4}$$

Consequently, this optimization problem is solved as a Lagrangian function, which is obtained by multiplying constraint equations by Lagrangian multipliers λ and subtracting from the objective function above, yielding;

$$L(w, b, \lambda) = \frac{1}{2}(w, w) - \sum_{i=1}^{n} \lambda_i \{[(x_i \cdot w) - b]y_i - 1\} \tag{6.5}$$

Furthermore, the Lagrangian formulation allows for the solution to be generalizable to non-linear scenarios. The approach involves introduction of Lagrangian multipliers λ_i, for $i = 1, 2, \cdots n$ for each of the constraints. The Lagrangian function is therefore maximized according to $\lambda_i \geq 0$ and minimized according to w and b. This solution yields the following result for the weight vector;

$$w_0 = \sum_{i=1}^{n} \lambda_i y_i \mathbf{x}_i \tag{6.6}$$

and the following coefficients of the optimal hyperplane

$$\sum_{i=1}^{n} \lambda_i y_i = 0 \tag{6.7}$$

Replacing solutions 6.6 and 6.7 into the Lagrangian formulation yields;

$$L_D = \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i,j} \lambda_i \lambda_j y_i y_j (x_i \cdot x_j) \tag{6.8}$$

Support vectors are therefore all the data points for which $\lambda_i > 0$, as given below:

$$w_0 = \sum_{support vectors} y_i \lambda_i x_i, \qquad \lambda_i \geq 0, \qquad i = 1 \cdots n \tag{6.9}$$

6.1.2 Non linearly separable SVM

In the case of the non linearly separable data, the SVM introduces a slack variable (ξ) and a regularization parameter (C) that allows some data points to be misclassified with the objective of improving the model's final generalization ability as given in equation 6.10. A small value of C implies a small penalty for error, meaning that

the model is accommodative of some misclassification. On the other hand, a higher C forces the model to seek the best linear discriminant or hyperplane for higher dimensions that minimizes misclassification error. Tuning this parameter is important because it might be the difference between having a model that is generalizable, or having an over-fitting model.

$$\Phi(\xi) = (w, w) + C \sum_{i=0}^{n} \xi_i \tag{6.10}$$

The objective function 6.10 must be minimized subject to equation 6.11;

$$y_i((w \cdot w) - b) \geq 1 - \xi_i, \qquad i = 1 \cdots n \tag{6.11}$$

The optimal separating hyperplane is found by maximizing the same Lagrangian formulas 6.5 and 6.8 under the following conditions in equation 6.12;

$$\sum_{i=1}^{n} \lambda_i y_i = 0, \qquad 0 \leq \lambda_i \leq C, \qquad \text{for } i = 1 \cdots n \tag{6.12}$$

In order to get the support vectors from the non linearly separable SVM, an additional step of projecting the data into a higher dimensional space Z is done using a non-linear mapping function or kernel. This is done such that for vectors x_1 and x_2, these are represented by their mapping functions $z(x_1)$ and $z(x_2)$. Therefore, the dot product of the transformation of this data in the higher space can be represented by kernel functions K of the form.

$$(z_i \cdot z) = K(x, x_i) \tag{6.13}$$

The decision functions in the high dimensional space now becomes;

$$I(x) = sign\left(\sum_{supportvectors} \lambda_i K(x_i \cdot x) + b_0 \right) \tag{6.14}$$

This equation 6.14 is still subject to the constraints in equation 6.12. Furthermore, there are some kernel functions $K(x_i \cdot x)$ that can lead to the construction of different learning machines and lead to different performance. In this study, however, we will only use the kernel machines described in the subsequent subsection to determine which particular kernel works best with our dataset.

Kernel machines

As alluded to earlier, kernels are functions Φ that are used to transform non linearly separable data, into higher dimension space called feature space, where the data becomes separable. This transformation enables the construction of non-linear decision boundary that is equivalent to linear decisions in feature space (Vapnik, 1999; C, 1998; Guyon et al., 2002). The transformation is therefore done on the dot product of the training data as in formulation 6.8 such that $x_i \cdot x_j$ becomes $\Phi(x_i) \cdot \Phi(x_j)$

We will, therefore, consider, cases where our kernel function represented by Φ, is either a linear function, a radial basis function, a polynomial function or a sigmoid function. The kernel formulations presented here are based on the work by Chih-Wei Hsu, Chih-Chung Chang and Lin (2008), who wrote a piece on guidance to classification using SVM, where they provided a step by step process to using SVM.

Polynomial kernel

The polynomial kernel introduces two parameters d and r, whose functions are to control the impact of individual data points in the classification and raising the transformation to a power of the polynomial to allow learning of a non-linear relationship in the data and specification of the intercept, respectively. The polynomial kernel can identify the higher order polynomial through which the data are related.

$$K(x_i, x_j) = (x_i^T \cdot x_j + r)^d, \qquad d > 1 \tag{6.15}$$

Radial basis function kernel

Radial basis function kernel uses the exponential function as given in the formulation below:

$$K(x_i, x_j) = exp(-\gamma||x_i - x_j||^2), \qquad \gamma > 0 \tag{6.16}$$

Sigmoid kernel

The sigmoid kernel uses a trigonometric function as given below:

$$K(x_i, x_j) = \tanh(\gamma x_i^T \cdot x_j + r), \qquad \gamma > 0 \tag{6.17}$$

6.2 Methods

6.2.1 Datasets used

We used the full features dataset and the controllable features datasets in this study. For details on these datasets, please refer back to chapter 4, to the section on data preparation.

Dataset	Scenario	Balanced	Features	Cases
Full features	All cases	yes	133	6,738
		no	133	10,039
	No mild anemia	yes	133	1,746
		no	133	7,543
Controllable	All cases	yes	50	6,738
		no	50	10,039
	No mild anemia	yes	50	1,746
		no	50	7,543

TABLE 6.1: *Datasets and sub-datasets used in the study.*

Several sub-datasets were obtained from these two primary datasets as follows: 1) sub-datasets of equal numbers of anemic and non-anemic cases, and 2) sub-datasets where all mild anemic cases were removed from the datasets. We summarise all these dataset scenarios for ease of reference in the table 6.1.

The target class for predictions in our experiments was the binary anemia attribute, which takes the values -1 and 1, for non anemic and anemic cases respectively.

6.2.2 Approach

Chih-Wei Hsu, Chih-Chung Chang and Lin (2008) proposed the following approach to classification using SVM, which we adopted:

Step 1, the transformation of the dataset to a format acceptable to SVM package. In line with this step in this study, our dataset was transformed from Stata format (.dta) then the comma separated variables (.csv) and finally saved in a MySQL database for use with Python. Moreover, an identical copy was kept in attribute relation file format ARFF for use in Weka. The dataset was split into three parts, two parts were for training and validation while the third part was reserved for testing.

Step 2, scale the dataset by normalizing it to a normal distribution $N(0, 1)$ such that mean $(\mu = 0)$ and standard deviation $(\sigma = 1)$. In our case, we used the built-in preprocessing functionalities in the data mining platforms that we used to scale the data.

Step 3, use the RBF kernel as the first kernel of choice on a subset of the training dataset for hyper-parameter grid search. In our case, we explored all the four kernels that we chose for use in the study. We implemented a parameter grid search process

similar to the Scikit-Learn function by the same name, which allowed us to do an exhaustive parameter search in our chosen space for hyper-parameters. The approach is that various values were used for cost C, gamma γ and kernel which placed in an array or parameter grid. Using this parameter grid we built models for the various combinations of these hyper-parameters of kernel, γ and C. This approach allowed us to determine which kernels performed best, and for which particular values of C and γ. Specifically, we tested the following values for the hyperparameters: $1)$ for kernels we focused on linear, RBF, sigmoid and polynomial. $2)$ γ values were searched in the interval $\{10^{-8}, 10^{4}\}$. Finally, $3)$ values for C were searched in the interval $\{1, 1000\}$. Three fold cross validation was used in this case to determine the best performing model parameters.

Step 4, use the best SVM parameters to train models using the entire training dataset and test using the reserved test dataset from step 1. We provide the results of this step in our results section.

6.2.3 Model development

We developed SVM models for all the supervised ML dataset scenarios that were presented earlier. Performance of the models was assessed using three-fold cross validation. Weka platform provides a GUI interface where all model options can be set, including cross validation, in Python however though the same provision exists we did our own implementation of the three fold cross validation.

Model development was done in Python using the LibSVM implementation found in SciKit-learn machine learning library. While in Weka, the same task was done using a Java implementation of the SVM algorithm called sequential minimal optimization SMO. All the models were subjected to evaluation for their correctness and performance. We will, therefore, provide results of the best SVM predictive models from both of these two platforms.

6.2.4 Model performance assessment

We used measures of tp, tn, fp and fn, which were also used for computing other performance metrics such as accuracy, precision, specificity, sensitivity, f measure and ROC, to measure the success of our supervised machine learning models.

Furthermore, to assess the performance of the models, we carried out many experiments for the various combination of parameter values to obtain mean performance.

6.2.5 Choosing most predictive features

Another important goal of this study was to identify features that have the most predictive power for anemia. To this end therefore, we used the linear kernel SVM to extract feature weights. These feature weights provide an indication of which attributes are considered most important by the SVM model in the prediction task. Therefore, having identified the best features, we listed these and ranked them by the absolute feature weight value.

The results of using these methods are provided in the subsequent section.

6.3 Results

We provide results in three subsections as follows: sub-section 6.3.1 provides a summary of best model parameters which were identified in the parameter grid search, sub-section 6.3.2 provides results for the model building results and sub-section 6.3.3 provides a summary of the best predictors in the full features dataset.

6.3.1 Model parameters

At the end of the parameter grid search process, the best model performance was obtained on polynomial, linear, sigmoid, and RBF kernels. We accomplished the objective of the grid search which was to narrow down the parameter space, by isolating those parameters that were the best performing parameters.

We identified the following hyper-parameters, in the case of C, values in the range $[10, 100]$ produced good results, while concerning values of gamma γ a wide range of values were acceptable from $10^{-}2$ up to $1e^{-18}$. This range of values is the parameter space that we identified through the parameter grid search, and is the set of hyper parameters that were subsequently used to build models.

Hence, having identified the best model parameters for the svm models, we used them to build SVM models using the training dataset and tested with the reserved test dataset as detailed in the subsection 6.2.2.

6.3.2 Model building results

The best SVM model parameters were used to build models for the various dataset scenarios as outlined earlier in table 6.1. As will be noted from these results subsections, we successfully developed SVM classifiers for anemia and identified a set of useful parameters for the polynomial, linear, RBF, and sigmoid kernels for the different dataset scenarios. Our balanced sub-datasets produced the best performing models from model 1 to model 3 on the full features dataset and model 4 on the controllable features dataset. These models had the Kappa statistics ranging from 0.46 to 0.72, showing that these are good models, whose performance is different from any random classifier. The other metrics such as sensitivity, precision and F measure which are indicators of predictive power, also yielded acceptable values for these models too.

On the other hand, for our dataset, scenarios that used the entire data points for model building, performance was poor. One such model is shown in model 5, the model had accuracy of 66%, and Kappa statistic of 0.0006. These metrics are indicative of the performance of a typical random classifier, that classified almost all data points as non-anemic. This erratic behavior, therefore, led to the 66% accuracy because that is the proportion of non-anemic cases in the dataset.

We now provide the details of the model building and evaluation in the subsequent sub-sections. We built models for all the 8 dataset scenarios which we outlined in table 6.1. In the process we identified some good performing models, on those dataset scenarios. Therefore, the selected good performing models are presented in the subsections below.

Models for balanced full dataset

We identified models 1, 2 & 3 which performed pretty well on the balanced full features dataset.

Model 1

This SVM model had a linear kernel and C value of 100.0. The computed performance metrics are as follows: 1) accuracy 83.13%, 2) Kappa statistic 0.66 and 3) mean absolute error of 0.34. Detailed class performance is given in table 6.2.

Algorithm	Sensitivity	Precision	F measure	ROC area	Class
SVM	0.87	0.81	0.84	0.83	−1
	0.79	0.86	0.82	0.83	1
Weighted Av.	0.83	0.83	0.83	0.83	

TABLE 6.2: *Linear kernel performance on the balanced full features dataset.*

Model 2

The second good model had a polynomial kernel with $C = 100.0$ and $\gamma = 1e^{-18}$.

The detailed performance was: 1) accuracy 75%, 2) Kappa statistic 0.51 and 3) mean absolute error 0.49. Detailed performance by class follows in table 6.3:

Algorithm	Sensitivity	Precision	F measure	ROC area	Class
SVM	0.74	0.76	0.75	0.75	−1
	0.77	0.75	0.76	0.75	1
Weighted Av.	0.75	0.75	0.75	0.75	

TABLE 6.3: *Polynomial kernel performance on the balanced full features dataset.*

Model 3

The third model had a sigmoid kernel with $C = 10.0$ and $\gamma = 0.0001$.

Over the performance was as follows: 1) accuracy 73%, 2) Kappa statistic 0.46 and 3) mean absolute error 0.26. Below is the detailed performance by class in table 6.4.

Algorithm	Sensitivity	Precision	F measure	ROC area	Class
SVM	0.70	0.75	0.72	0.73	−1
	0.76	0.71	0.74	0.73	1
Weighted Av.	0.73	0.73	0.73	0.73	

TABLE 6.4: *Sigmoid kernel performance on the balanced full features dataset.*

Models on controllable features datasets

Regarding performance of models on the controllable features sub-datasets, and the primary controllable features dataset, the models that performed well are model 2 and model 3 as detailed below in tables 6.5 and 6.6.

Model 2

The second good model had a polynomial kernel with $C = 100.0$ and $\gamma = 1e^{-18}$.

The computed performance metrics are as follows: 1) accuracy 86.19%, 2) Kappa statistic 0.72 and 3) mean absolute error of 0.27, details are in table 6.5.

Algorithm	Sensitivity	Precision	F measure	ROC area	Class
SVM	0.88	0.85	0.86	0.86	-1
	0.84	0.88	0.86	0.86	1
Weighted Av.	0.86	0.86	0.86	0.86	

TABLE 6.5: *Performance of the polynomial kernel on the balanced controllable features dataset.*

Model 3

The sigmoid kernel had the following parameters $C = 10.0$ and $\gamma = 0.0001$. The computed performance metrics are as follows: 1) accuracy 72.94%, 2) Kappa statistic 0.46 and 3) mean absolute error of 0.54, details are in table 6.6.

Algorithm	Sensitivity	Precision	F measure	ROC area	Class
SVM	0.70	0.75	0.72	0.73	-1
	0.76	0.71	0.74	0.73	1
Weighted Av.	0.73	0.73	0.73	0.73	

TABLE 6.6: *Performance of the sigmoid kernel on the balanced controllable features dataset.*

We were also able to attain good performance on the balanced controllable features dataset without mild anemia sub-dataset. This development led us to identify model 4 whose results are provided in the next sub-section.

Model 4

The SVM model with rbf kernel, γ value of 0.1 and C value of 10.0 , produced the following performance on a controllable features dataset without mild anemia cases.

Accuracy was 72.39%, while Kappa statistic was 0.45 and mean absolute error was 0.28. Detailed class performance is provided in the table 6.7.

Algorithm	Sensitivity	Precision	F measure	ROC area	Class
SVM	0.92	0.66	0.77	0.72	-1
	0.52	0.87	0.66	0.72	1
Weighted Av.	0.72	0.77	0.71	0.72	

TABLE 6.7: *Performance of the RBF model on the controllable features without mild anemia cases.*

Worst performing models

The worst performance was obtained on dataset scenarios that used the entire set of $10,039$ cases. This was the case for either the full features dataset or the controllable features datasets.

Model 5

We showcase one such model in this sub-section. This is an SVM model with a polynomial kernel with exponent $d = 3$ and cost $C = 100$. These results were obtained on an unbalanced full features dataset.

The detailed performance is as follows: $1)$ accuracy 66.4508%, $2)$ Kappa statistic 0.0006 and $3)$ mean absolute error 0.34. While the detailed performance per class is given in table 6.8.

Algorithm	Sensitivity	Precision	F measure	ROC area	Class
SVM	1.000	0.665	0.798	0.500	-1
	0.001	0.667	0.001	0.500	1
Weighted Av.	0.665	0.665	0.531	0.500	

TABLE 6.8: *Polynomial performance on the balanced full features dataset.*

In conclusion, in this model building process, we obtained the best performance on two dataset scenarios namely the balanced full features sub-dataset and the balanced controllable features sub-dataset without mild anemic cases. Regarding kernels, the kernel that provided the best performance was the polynomial kernel on controllable features dataset with the accuracy of 86% and ROC area score of 86% as well.

6.3.3 Most predictive features based on Linear SVM weights

A ranking of the most predictive features was generated using the support vector machines on the full features dataset. The ranking is based on feature weights for predictive power on either anemic and non-anemic classes.

The lists are presented separately for positive and negative predictors in tables 6.9 and 6.10 respectively.

Rank	Weight	Attribute
1	1.721	Type of toilet facility: $v116 = 41$
2	1.5404	Getting medical help for self: concern no female hw: $v467g = 0$
3	1.5383	Getting medical help for self: not wanting to go: $v467f = 0$
4	1.4291	De facto place of residence: $v134 = 0$
5	1.333	Currently pregnant: $v454 = 1$
6	0.9988	Getting medical help for self: getting permission: $v467b = 0$
7	0.6669	During pregnancy took Fansidar for malaria: $m49a1 = 0$
8	0.6665	During pregnancy took Fansidar for malaria: $m49a1 = 1$
9	0.2088	Getting medical help for self: having to take trans. $v467e = 1$
10	0.2085	Getting medical help for self: getting money needed: $v467c = 1$
11	0.2084	Getting medical help for self: getting money needed: $v467c = 2$
12	0.2081	Getting medical help for self: having to take trans. $v467e = 2$
13	0.2078	Getting medical help for self: getting money needed: $v467c = 0$
14	0.2078	Getting medical help for self: having to take trans. $v467e = 0$
15	0.1032	WtHt Percent ref median Fog: v443
16	0.0013	Usual resident or visitor, $v135 = 1$
17	0.0008	Has television, $v121 = 0$
18	0.0007	Has radio, $v120 = 1$
19	0.0007	Has refrigerator, $v122 = 1$
20	0.0005	Has motorcycle, $v124 = 0$

TABLE 6.9: *Top* 20 *best positive predictors for anemic.*

Rank	Weight	Attribute
1	-1.1833	De jure region of residence, $v139 = 2$
2	-0.9998	Getting medical help for self: concern no female hw, $v467g = 1$
3	-0.9998	Getting medical help for self: concern no female hw, $v467g = 2$
4	-0.7917	Getting medical help for self: distance to health, $v467d = 2$
5	-0.7916	Getting medical help for self: distance to health, $v467d = 1$
6	-0.5919	Region, $v101 = 1$
7	-0.5918	Region, $v101 = 3$
8	-0.4999	Getting medical help for self: getting permission, $v467b = 2$
9	-0.4998	Getting medical help for self: getting permission, $v467b = 1$
10	-0.4582	Getting medical help for self: not wanting to go, $v467f = 2$
11	-0.4581	Getting medical help for self: not wanting to go, $v467f = 1$
12	-0.3331	Results of salt iodine test, $v166 = 0$
13	-0.3331	Results of salt iodine test, $v166 = 7$
14	-0.333	Results of salt iodine test, $v166 = 15$
15	-0.2995	Has car, $v125 = 0$
16	-0.2995	Has car, $v125 = 1$
17	-0.25	Frequency of listening to radio, $v158 = 0$
18	-0.25	Frequency of listening to radio, $v158 = 1$
19	-0.25	Frequency of listening to radio, $v158 = 2$
20	-0.25	Frequency of listening to radio, $v158 = 3$

TABLE 6.10: *Top* 20 *best predictors for non anemic.*

6.4 Discussion

We will discuss the results in this section, in line with the way we carried out our modeling and evaluation, and how we presented the findings thereof, as reported in the preceding section. We discuss three main issues, first we compare our findings to a few similar studies in sub-section 6.4.1, then we consider anemia prediction in sub-section 6.4.2, we thereafter move on to discuss the best anemia predictors that were identified in sub-section 6.4.3 and then sum up the discussion.

6.4.1 Performance comparison with literature

The performance levels that we achieved in this study, where the best performance was 86% for accuracy, are not too far from the reported accuracy levels by others in the literature. For instance Kesorn et al. (2015) achieved 96% on training set and 88% on testing set, Cohen et al. (2004) attained 75% accuracy with sensitivity at 92% and a low specificity, while Barakat, Bradley, and Barakat (2010) attained 92% accuracy. Therefore, we can state that our models were a success, even though our results have not exceeded the performance of those whose work we have reviewed in this study.

Regarding our dataset size, having $10,039$ data points overall is a reasonable size, even considering the sub-datasets which we used whose sizes ranged from $6,738$ and $1,746$ data points for the balanced sub-datasets was equally good. Comparatively, in some of the works cited in this thesis such as, Barakat, Bradley, and Barakat (2010), who had access to a total dataset size of $3,014$ cases, but ended up selecting 174 diabetic and 315 non-diabetic cases to use in their study. The other author was Kesorn et al. (2015) had 648 data points which were selected and used for model building in their study. Considering that, these researchers had to generate both training and testing datasets from these datasets which seem to be fairly smaller, we believe our dataset was good enough from a data size perspective.

Therefore, even though we are aware that, SVMs are not easily affected by data sizes, which happens to be their advantage. Mainly because, the decision boundary hinges on the availability of support vectors, which usually are fewer than the dataset. Nevertheless, having a dataset with a sufficient number of unique data points for both training and testing does benefit supervised machine learning methods generally, and further enhances the process of learning patterns for better generalization in deployed models.

6.4.2 Anemia prediction

The author Setotaw (2013) worked on an anemia prediction task, and he achieved a prediction accuracy of 85% on a set of select attributes from the dataset. However, on the complete set of attributes, he was able to achieve 95% accuracy using J48 decision tree algorithm and PART association rule mining algorithms.

In our case, our prediction accuracy of 86% for our best SVM is a good and comparable result, demonstrating the effectiveness of SVM for anemia prediction.

Moreover, the fact that we achieved high accuracy on the controllable features dataset was confirmation of our hypothesis that anemia can be predicted using a set of controllable features. It is good to have a set of controllable issues that one can use as a start in advising on a possible policy guide for combating the anemia challenge.

However, this may not necessarily mean that we can fully disregard the non controllable features from the full features dataset. For instance if one considers the fact that the best predictive models on full features dataset and controllable features dataset have accuracy of 83% and 86% respectively, which is not a large difference. In addition to this, the list of the best predictors for our SVM model, which we obtained on the full features dataset revealed an interesting idea. In this particular list the attribute *'Region'*, was selected as the most dominant feature for predicting non-anemic cases, refer table 6.10. Specifically, all the three values for *'Region'* had different weights, central region 2 was most highly ranked followed by southern 1 and finally northern region 3. In this case a non controllable attribute had more impact than controllable attributes. Therefore, it may be worth considering that a good mixture of controllable and non controllable features may lead to more accurate predictors using SVM.

6.4.3 Best predictors for anemia from Linear SVM

We now turn to consider the best anemia predictors, which were reported in tables 6.9 and 6.10 as referred to before. These tables provide a list of the best predictors for anemia arranged in order of decreasing weight values. This listing of the attributes can be interpreted as a list of the most important issues that have the greatest impact on anemia or its causative agent malaria.

In the list of the strongest predictors for being anemic we have *Type of toilet facility: v116=41* this refers to a composting toilet and assesses sanitation level, *Getting medical help for self: concern no female hw: v467g=0* refers to not being worried about presence of female health worker as a hindrance to accessing medical help, *Getting medical help for self: not wanting to go: v467f=0*, *De facto place of residence: v134=0* and *Currently pregnant: v454=1* are the other top most important attributes.

On the other hand, for the non anemic case the following attributes are listed as the most crucial *De jure region of residence, v139=2*, *Getting medical help for self: concern no female hw, v467g=1*, *Getting medical help for self: concern no female hw, v467g=2*, *Getting medical help for self: distance to health, v467d=2* and *Getting medical help for self: distance to health, v467d=1*.

In terms of the themes to which all the important attributes listed in the two tables belong, we identified the following themes, *sanitation*, *health practices*, *geography*, *demographics*, *anthropometry* and *socio-economic*. However, amongst these themes, the most dominant was health practices, which had the largest number of features on the top 20 lists for both being anemic and non-anemic. Moreover, when compared to the themes identified by DTs in chapter 7 we can see that it is the same themes that are selected even though the specific attributes that are selected are different.

Finally, even though we used the full features dataset to obtain the feature ranking, the rank shows that the most dominant features are controllable. As indicated earlier, most of the controllable features fall under the umbrella of health practices

making it possible to make recommendations on what can be tackled in an effort to combat anemia.

That fact notwithstanding, our models still performed pretty well for both platforms, yielding a result which is more than 20% above the random classifier performance for both accuracy and ROC area measure.

In this chapter, we provided results where support vector machines were used for classification of anemia, using a full features dataset and a controllable features dataset and their sub-datasets.

We also identified a set of best attributes for the prediction task, which led us to identify the dominant themes, which are indicative of issues that matter in anemia prediction. We conclude that SVM models can be created on the DHS dataset using a set of the most useful features which are a combination of both controllable and uncontrollable features from the dataset.

7 Decision Trees

In this chapter, we discuss how we conducted our supervised ML algorithm experiments using decision trees, and provide the results that we obtained. Decision trees were used as our second supervised machine learning algorithm of choice, mainly because we had already successfully built an SVM classifier, as was discussed in the previous chapter.

Therefore, we wanted to use a different ML approach from SVM, which aside from allowing us to build a classifier, would also provide us with further insights into the classification process. In line with this, we used decision trees to both verify our earlier results, and also, to gain new insights on the classification process through identification of the features that were most informative in building the decision tree model. The decision tree model accomplishes this task, by showing the relationship between prominent features in the models as decision nodes at various levels of the decision trees.

This quest is in line with the CRISP-DM methodology because once knowledge has been extracted and verified, one needs to fully understand the models that represent that knowledge, to be in a position to deploy a useful product or solution, which is the ultimate goal of data mining (Wirth and Hipp, 2000).

The specific objectives of the chapter were as follows; 1) to build a decision tree based classifier using the anemia attribute as a target class, 2) to identify important attributes in the decision trees, and finally, 3) to infer sensible rules from the best decision trees.

We used the same datasets as was used in the previous chapter on SVM. Therefore in this chapter, we discuss decision trees which we built. These were created using the full features dataset and the controllable features dataset. Other decision trees were constructed also using sub-datasets of these two primary datasets.

7.1 Defining decision trees

Decision trees in short DT, are a supervised classification method that is appreciated both for its performance, as well as its understandability. DT are inverted tree structures, where the base of the tree called the root node is at the top, and branches with leaves are below this root node, the exact opposite of the natural tree (Quinlan, 1986). A DT is made up of nodes representing the attributes in the dataset, and edges which are lines showing a relationship between nodes in the decision tree. Edges link parent nodes at the top, to child nodes, which are below them.

Decision trees are built iteratively. Hence this parent-child relationship is duplicated across the tree in the various branches that emerge. The process continues until no

more features are available to split the decision tree further, consequently, leaf nodes are formed at the terminal nodes.

By definition a decision tree can have one or more nodes. The first node in the decision tree is called a root node and subsequent nodes are called child nodes, while nodes that are found at the same depth of the tree are called sibling nodes. Nodes can be of two types, decision nodes or leaf nodes.

Decision nodes are the venue for decision making, where test's on an attribute are done, boolean expressions such as $A_i > val$ are used, the response to which leads to different branches of the tree. The number of branches may be dependent on the number of values that the attribute takes.

The second node category is leaf nodes, also referred to as class nodes, these nodes are terminal nodes, which by definition is the lowest level node in a tree branch, and gives a classification decision.

The depth of the decision tree is the number of levels from the root of the trees to the termination nodes called leaves of the tree, where classification decisions occur. On the other hand, the size of the tree is given by the number of attributes that are selected in the decision nodes of the tree.

Decision trees partition a dataset using the best feature in the dataset, for example, given a dataset D which has a set of features, the algorithm picks an attribute A that take values A_1 and A_2, the dataset will be split into as many sub-datasets as there are values that A can take.

The best attribute is selected at level 1 to become the root node. Subsequently, other good attributes get selected for splitting the sub-datasets D_1 and D_2, and the process continues iteratively until leaves get generated at terminal points.

A terminal node that has all data points that belong to one class is labeled by that class name. Otherwise, impure leaf nodes are labeled by the majority class for the data points that it has (Quinlan, 1986).

This ML method has a complexity of $nlog(n)$, where n is the number of data points, which makes it suitable for classification tasks that require a quick generation of results. The algorithm works by recursively picking good features for building the decision tree, therefore, by tracing the path from the root to the leaves, it is possible to identify all attributes that lead to a particular classification decision and the corresponding values for those key attributes.

Through this process, one can build association rules that are sensible enough for one to use as new knowledge, which can be applied in new scenarios for decision-making purposes. Specifically, Quinlan (1986) indicates that decision trees are useful for knowledge generation in expert system application domains, where they can be used for automation of the rule extraction process, in building knowledge bases for any specific application domain. However, since decision trees have the intrinsic property of simplicity making them easy to interpret, they have been used successfully in many business, medical and marketing applications (Quinlan, 1986).

7.1.1 Pros and cons of decision trees

We will provide a very brief overview, of the merits and de-merits of DT algorithms, in this subsection. An understanding of the pros and cons is essential because, it allows a summary of fundamental principles of an algorithm, to be presented concisely. We will not provide an exhaustive discussion of DT, but rather we will highlight a select number of points to support our rationale for choosing this particular ML algorithm, over the many other ML algorithms that are available, as our second supervised ML algorithm in this study.

Pros

There are many advantages of the method that have been identified by several authors. However, we'll discuss three advantages. First of all, DTs are easy to understand and interpret because the algorithm follows a white box approach, allowing model analysis after the DT is built. This transparency of the classification approach makes it easy to review results of classification, by reviewing how the decision tree was built, and which particular attributes were most influential in the model.

This transparency allows a user to analyze how the algorithm makes its decisions in choosing which particular features to use for splitting a tree to build branches, and more importantly, how the algorithm eventually arrives at a final classification decision.

Unlike other machine learning algorithms, where the user can only see the inputs and outputs, DTs enable the user to visualize the entire classification process. Furthermore, tools such as the *.dot* language, makes it easier for decision trees to be visualized, allowing for visual inspection of the results. This ease in understandability makes them also very useful as a communication tool, whereby they can be used to communicate with those experts knowledgeable in some specific domain of interest, but who may otherwise not very familiar with ML techniques.

Secondly, DTs can handle both categorical as well as numerical data; this means that a decision tree, once generated, is viewed using the actual values that it has in its attributes, without a need for scaling or discretization of the values. However, for numerical variables it may still be useful to do some discretization, if the range of values taken by the attribute is too big, to ensure that simpler trees are created.

Consequently, DTs do not need much data preparation before usage, one can just load the data with its original values into the algorithm. This flexibility is unlike SVM and other ML algorithms that use various distance metrics, which require data normalization or scaling to avoid biased results, in cases where specific attributes take large values, as is the case in our dataset for some attributes. DT does not need to have normalized or scaled data before use, however, the original data values can be used with the discretization of some numeric attributes into categories for ease of classification as stated earlier.

This idea further strengthens the point already made on the understandability of results, because decision points in the tree are based on raw data values, e.g., if a tree splits on an age attribute, different age groups will lead to different branches in the decision tree leading to different classification decisions. This strength, therefore, also makes DTs pretty useful as a data exploration tool, as one is going through the

process of understanding the dataset and how the different attributes relate to one another (Quinlan, 1986; Alpaydin, 2010; Ramakrishnan et al., 2009).

Cons

On the other hand, decision trees also have some shortcomings, which we highlight below. We, however, provide mitigation options, which we used in this study to overcome these challenges.

The first challenge is that DTs tend to over-fit data, thus affecting their generalization ability. We resolved this weakness by using algorithms that implement pruning techniques. This solution ensures that branches that have precise rules for a small set of data points get removed because they are influenced by the particular training dataset used. A measure of the decision trees prediction accuracy was also used to assess whether a decision tree is a suitable classifier on test dataset or otherwise.

Secondly, DTs are influenced by any small changes in the dataset, for example, the removal of specific attributes can lead to very different trees being built.

Finally, decision trees also can lead to the production of very complicated and meaningless rules. This situation is partly as a result of the over-fitting problem that we mentioned earlier. Usage of balanced datasets with an equal number of members from the different classes can help ease the challenge. In our case, we selected an equal number of positive and negative anemia cases for our training dataset, as part of one of the scenarios that we experimented with for both the full features dataset, the controllable features dataset and the no mild anemia sub-datasets.

These shortcomings notwithstanding, DTs are a popular ML method and hence our choice for their use in this research study.

We will be using the Iterative Dichotomizer Version 3, otherwise referred to as ID3, an inductive decision tree algorithm developed by J. Ross Quinlan for our discussion (Quinlan, 1986). This algorithm was used in the development of many other decision tree algorithms, which we will also discuss. These other algorithms include C4.5, which is an advanced version of the Iterative Dichotomiser again developed by J. Ross Quinlan. J48 decision tree algorithm is a Java implementation of C4.5 in Weka, and we used this implementation in our model building and evaluation experiments. The CART is another inductive decision tree algorithm, which was developed independently and concurrently with the ID3 algorithm by other researchers (Ramakrishnan et al., 2009).

Before discussing these various decision tree algorithms in detail, we will provide a brief overview of the methods used for choosing the best feature or the most informative attributes, and these measures are; Entropy, Information Gain, and Gini Impurity.

Entropy

Information entropy is a measure of the usefulness of a feature or attribute, based on the information that it contains. The rationale of the measure is to quantify the amount of information that a feature or attribute contains. In this regard, entropy

provides a means of quantifying whether one attribute is better than another attribute, which becomes the basis for selecting such good features in building decision trees. These selection criteria present useful information in the construction of the decision tree because ultimately a decision needs to be made in choosing an attribute at various levels in the decision tree building process.

$$H(p) = -\sum_{i} p_i log_2 p_i \tag{7.1}$$

Entropy makes use of a measure of the probability p of each feature i using the number of values a feature can take on as well as a count of those values.

We will use our dataset variable $v457$ which contains anemia information for illustration of entropy. Attribute $v457$ takes on the following class values $1, 2, 3, 4$ & 9, which represent severely anemic, mildly anemic, anemic, not anemic and missing values respectively. The attribute has $10,660$ data instances, having the following distribution 87, 786, 2496, 6670 & 621, for the class values respectively.

The entropy for this feature can be calculated, by first computing the individual probabilities of the relevant class values, e.g., $p_1 = \frac{87}{10,660}$ for the first class and similarly for the other class values. These values will hereafter be used to compute the entropy for the class, and in like manner for all attributes in the dataset to get the total entropy.

Information gain

Information gain is a measure of the impact of an individual feature on the total entropy of a dataset. In choosing the root feature for a decision tree, we need to identify the feature which leads to the most significant gain in information.

Supposing that S is the set of all examples, F is a feature which can be selected from S, $|S_f|$ are number of members in S having value f for feature F. Information gain is therefore defined as;

$$Gain(S, F) = H(S) - \sum_{f \in values(F)} \frac{|S_f|}{|S|} H(S_f) \tag{7.2}$$

Because of an inherent weakness is this measure of picking categorical attributes with many values, another measure called Gain Ratio is used to mitigate this challenge formulated as follows;

$$\text{Gain Ratio}(F) = \frac{Gain(F)}{H(F)} \tag{7.3}$$

(Ramakrishnan et al., 2009)

Gini index

Gini index is used in CART decision tree algorithm and requires binary valued attributes. Gini impurity measures the extent to which, for any correct classification p_i,

an incorrect classification $1 - p_i$ is possible or can be made. Gini impurity measures the impurity of a dataset D as follows;

$$1 - \sum_{i}^{k} p_i^2 \tag{7.4}$$

Where p_i is the probability of a data point being classified as belong to class C_i for $i = [1, \cdots m]$.

7.1.2 Decision tree algorithms

In this subsection, we will discuss three main decision tree algorithms, as a matter of laying a background for the chapter.

ID3 algorithm

The Iterative Dichotomizer version 3 otherwise referred to as the ID3 algorithm, which was developed by Quinlan (1986), is the first decision tree that we will consider. The algorithm uses the information gain described in the previous section recursively, to select the best attributes for inclusion in the decision tree.

The algorithm works as follows: A window, which is a subset of the training dataset is randomly selected and used to build a decision tree. The generated decision tree must be able to classify all the data points in the window correctly. After that, the resultant tree is used to classify the entire training dataset, if the classification is accurate the iteration stops. Otherwise, the misclassified data points are added to the window, and the tree building process is repeated (Quinlan, 1986).

The algorithm stops when it reaches a leaf and gives an output based on whether there are instances of one class at the node or not. In the case of instances from different classes, the algorithm outputs the class with the most representation at the leaf node, otherwise, if it is only data points from one class, it outputs the class of data instances that are found at the leaf (Quinlan, 1986).

C4.5 and variants

The C4.5 algorithm builds on the ID3 algorithm and was developed by the same researcher; this algorithm is an improved version of ID3 (Apte et al., 1997; Ramakrishnan et al., 2009). Weka uses the J48 decision tree algorithm which is a Java implementation of C4.5. Primarily, the algorithm accommodates the use of continuous attributes, which is not the case in the ID3 algorithm. An improved version of C4.5 called C5.0 was developed by the same developer of ID3 J. Ross Quinlan, offering better classification and smaller trees. However, this version is commercially distributed, and not many details are known about it (Alpaydin, 2010).

CART

CART algorithm was developed concurrently and independently from the ID3 by Leo Breiman and others (Ramakrishnan et al., 2009). The algorithm uses the Gini index for selection of the best attributes in the tree building process. The algorithm is iterative and follows a top-down induction process just like the algorithms of Ross Quinlan discussed earlier.

7.2 Methods

We will discuss the methods used in this chapter in the following sub-sections: sub-section 7.2.1 summarizes the datasets that were used in the study, we discuss how we built DT models in sub-section 7.2.2, and finally, subsection 7.2.3 discusses how we assessed the performance of our DT models.

7.2.1 Datasets used

We used our two main datasets which were; $1)$ the full features dataset which had 133 features and $10,039$ cases. The other dataset that we used was the controllable features dataset having 50 features and $10,039$ instances.

For each of these datasets, we generated sub-datasets, which had the same number of features as it primary dataset. In one set of sub-datasets, we removed all mild anemic cases. Therefore, this led to a set of sub-datasets as follows; $1)$ full features sub-dataset without mild anemia cases having $7,543$ cases with 133 features, and, $2)$ controllable features dataset without mild anemia cases having $7,543$ cases with 50 features.

The other sub-datasets that we generated involved selecting equal numbers of anemia and nonanemic cases in the primary datasets and the sub-datasets mentioned above. This situation meant that the number of cases was either $10,039$ for all cases, or $6,738$ for balanced datasets having mild anemic cases or $1,746$ for balanced datasets without mild anemic cases. Please refer to table 6.1 for a summary of these dataset scenarios. For each experiment that we conducted we had two-thirds of the data set for training and validation and a held out third part which we used for testing the developed models.

The target class for prediction was the binary attribute anemia, which took the values 1 and -1 for anemic and non-anemic cases respectively.

7.2.2 Building decision tree classifiers

We used the four datasets mentioned earlier in subsection 7.2.1 to build the decision trees. The decision trees were categorized based on the dataset that was used for generating them. These DTs were further divided into the full dataset or balanced dataset based on whether the entire number of cases was used or perhaps a subset of equal anemic and non-anemic cases was used.

We did the experiments until we were able to identify the best performing decision trees, for each dataset using the performance metrics discussed in chapter 4. CART and J48 decision tree algorithms were used to build our DTs models.

7.2.3 Assessment of decision tree performance

To assess the performance of our decision trees, we employed 3 fold cross validation during training for model tuning, we thereafter used the reserved test dataset to assess model performance, as we did in the previous chapter on SVM. We further generated a wide range of performance metrics such as accuracy, ROC area score, sensitivity, precision and various error measures for model performance assessment and comparison. Accuracy and ROC area score were the most important metrics in our performance analysis.

7.3 Results

In this section we provide results as follows; a summary of results is given in sub-section 7.3.1, thereafter we discuss some good tree models in subsection 7.3.2 and finally we analyse model performance in details and compare CART and J48 performance in sub-section 7.3.3.

7.3.1 Summarized decision tree classifier results

We provide the results for the decision tree building experiments in tables 7.1 and 7.2. In the results in these tables, we report only on the results of the CART algorithm as conducted using Scikit-learn in Python and the same algorithm as implemented in Weka.

Dataset	Balanced	Platform	Accuracy %	Roc Area %
Full features	Yes	$Weka$	64	67
	Yes	$Python$	69	68
	No	$Weka$	62	51
	No	$Python$	71	50
Controllable features	Yes	$Weka$	64	67
	Yes	$Python$	67	69
	No	$Weka$	63	55
	No	$Python$	71	50

TABLE 7.1: *Summary of results for the DTs on the full features dataset and controllable features dataset, that were conducted in both Weka and Python.*

We observe that the balanced decision trees produced results that were consistent across the platforms. They also provided better results when one considers the ROC area score that was computed, which provides an unbiased measure of overall model performance, irrespective of data skewness. However, for either the balanced

Dataset	Balanced	Platform	Accuracy %	Roc Area %
Full features	Yes	$Weka$	69	71
	Yes	$Python$	73	73
	No	$Weka$	62	51
	No	$Python$	91	50
Controllable features	Yes	$Weka$	69	70
	Yes	$Python$	73	73
	No	$Weka$	63	55
	No	$Python$	91	50

TABLE 7.2: *Summary of results for the DTs without mild anemic cases, that were conducted in both Weka and Python.*

or skewed dataset scenarios, performance was always better at detecting the non-anemic cases than the anemic.

7.3.2 Analysis of good models

We will now present graphics and details of some of the balanced decision trees that produced the good results shown in the earlier subsection, to allow for an analysis of the decision trees.

Full features dataset decision trees

We begin with a decision tree generated using the Simple CART classification decision tree algorithm. The tree is presented in textual form in figure 7.3.2. Within our discussion about the decision trees, feature names are quoted and given in italic form for ease of identification.

First of all, this particular DT identified five features from the total of 133 features that were available. The selected features are: *'Getting medical help for self: getting permission, v467b'* selected as root node, *'Getting medical help for self: concern no female hw, v467g'*, *'Cluster altitude in meters, v040'*, *'Ethnicity, v131'* and *'Current contraceptive method, v312'*.

In this DT the root node attribute code *v467b* splits the dataset into two parts one is a pure leaf node and another part a branch.

The leaf node is given on the first line in the DT as *Getting medical help for self: getting permission, v467b=(0): 1(1118.0/0.0)*. The condition test, checks if 'getting permission to go to the hospital is not a problem' for a respondent, consequently $1,118$ cases are classified correctly as anemic with 0 cases classified as non-anemic cases.

While on the branch side, the test on the root attribute v467b, asks if 'a respondent has a problem getting permission to go to the hospital'. Then on the subsequent decision node, v467g asks if respondent 'has concerns of not finding a female health

care provider' when they visit the hospital. These questions have the following responses: 0 is *'no problem'*, 1 is *'big problem'*, 2 is *'not a big problem'* and 9 is *'missing information'*.

On the third level, the attribute that records the cluster altitude of the respondent's home, attribute code $v040$, is picked and it splits the cases between those that live below 578.5 meters above sea level and others. This attributes classified $2,654$ of those living above the 578.5 meters above sea level correctly as not anemic and $1,461$ incorrectly. - The number of cases classified at this decision node shows how significant this particular feature is, in the decision tree.

Hereafter, an attribute that records the contraception method that the respondent uses is selected. Finally, the ethnicity attribute that records tribes is picked, in this attribute each number is a code for a tribe found in Malawi. Overall, it classifies 667 cases correctly and 501 cases incorrectly.

CART Decision Tree

Getting medical help for self: getting permission, v467b=(0): 1(1118.0/0.0)
Getting medical help for self: getting permission, v467b!=(0)
| Getting medical help for self: concern no female hw,v467g=(0): 1(77.0/0.0)
| Getting medical help for self: concern no female hw,v467g!=(0)
| | **Cluster altitude in meters, v040** < 578.5
| | | Current contraceptive method, v312= (7) | (10) | (5) | (0) | (2) | (6) | (4) | (12) | (13) | (15) | (16) | (17) | (18) | (19) | (20) | (99)
| | | | **Ethnicity, v131**=(11) | (12) | (4) | (1) | (5) | (6) | (3) | (9) | (99): 1(490.0/332.0)
| | | | **Ethnicity, v131**!=(11) | (12) | (4) | (1) | (5) | (6) | (3) | (9) | (99): -1(177.0/133.0)
| | | **Current contraceptive method, v312!**= (7) | (10) | (5) | (0) | (2) | (6) | (4) | (12) | (13) | (15) | (16) | (17) | (18) | (19) | (20) | (99): -1(206.0/90.0)
| | **Cluster altitude in meters, v040** >= 578.5: -1(2654.0/1461.0)

Decision tree on the full features dataset with equal number of anemic and non anemic cases. (7.5)

Controllable features dataset decision trees

We now consider the controllable dataset balanced decision trees. We first analyze the Weka decision tree presented in figure 7.3.2.

The first observation is that at the root node and level one node of this decision tree, the same attributes have been picked, as was the case in the full features dataset scenario presented earlier.

Secondly, aside from the attribute code $v040$ on *'cluster altitude'*, which is absent in this sub-dataset. All the other key attributes from the full features dataset decision tree scenario have been picked as well.

The root node $v467b$ correctly classifies $1,118$ anemic cases, of those having 'no problem' to get permission to go and access health care. In the right subtree, leading to

the rest of the branches of the tree, we highlight all the dominant attributes, within the tree in bold font. One of these attributes is the same root node which is picked on other levels in the tree. The second attribute is $v461$, which is an attribute having information on contraceptive use pattern, takes the following values: 1 is *'currently using'*, 2 is *'used since last birth'*, 3 *'used before last birth'* and finally 4 is *'never used'*. Those that are currently using a contraception method were classified as non-anemic $1,299$ correctly, and 624 otherwise.

The other attribute with a large impact, though not as much as the other two is attribute code $v166$, which has information on results for measurement of salt for iodine. This attribute classified 203 cases correctly as non-anemic, for people having iodized salt and 138 incorrectly.

CART Decision Tree

Getting medical help for self: getting permission, v467b=(0): 1(1118.0/0.0)
Getting medical help for self: getting permission, v467b!=(0)
| Getting medical help for self: concern no female hw, v467g=(0): 1(77.0/0.0)
| Getting medical help for self: concern no female hw, v467g!=(0)
| | **Pattern of use, v361**=(3) | (4) | (2)
| | | Education in single years, v133 < 4.5
| | | | Getting medical help for self: not wanting to go, v467f=(0) | (9) | (1)
| | | | | Type of cooking fuel, v161=(7) | (9): 1(34.0/7.0)
| | | | | Type of cooking fuel, v161!=(7) | (9)
| | | | | | Source of drinking water, v113 < 26.0
| | | | | | | WtHt Percent ref median Fog, v443 < 11611.0
| | | | | | | | **Pattern of use, v361**=(4) | (3) | (1): 1(144.0/78.0)
| | | | | | | | **Pattern of use, v361**!=(4) | (3) | (1): -1(27.0/18.0)
| | | | | | | WtHt Percent ref median Fog, v443 >= 11611.0: -1(32.0/25.0)
| | | | | | Source of drinking water, v113 >= 26.0: -1(106.0/76.0)
| | | | Getting medical help for self: not wanting to go, v467f!=(0) | (9) | (1)
| | | | | Respondent slept under bednet, v461=(1)
| | | | | | Literacy v155=(0) | (1) | (3) | (4) | (9)
| | | | | | | Highest educational level, v106=(1) | (2) | (3) | (9): 1(95.0/56.0)
| | | | | | | Highest educational level, v106!=(1) | (2) | (3) | (9): -1(66.0/55.0)
| | | | | | Literacy v155!=(0) | (1) | (3) | (4) | (9): -1(38.0/19.0)
| | | | | Respondent slept under bednet, v461!=(1)
| | | | | | WtHt Percent ref median Fog, v443 < 11065.0
| | | | | | | Main wall material, v128=(21): 1(16.0/5.0)
| | | | | | | Main wall material, v128!=(21)
| | | | | | | | Results of salt iodine test, v166=(0) | (997) | (995)
| | | | | | | | | Frequency of listening to radio, v158=(3) | (0) | (9): 1(36.0/18.0)
| | | | | | | | | Frequency of listening to radio, v158!=(3) | (0) | (9): -1(17.0/9.0)
| | | | | | | | **Results of salt iodine test, v166**!=(0) | (997) | (995): -1(203.0/138.0)
| | | | | | WtHt Percent ref median Fog, v443 >= 11065.0: -1(106.0/39.0)
| | | Education in single years, v133 >= 4.5
| | | | **Getting medical help for self: getting permission, v467b**=(1)
| | | | Type of toilet facility, v116=(21) | (22): 1(39.0/13.0)

| | | | | Type of toilet facility, v116!=(21) | (22)
| | | | | | Type of place of residence, v102=(1): 1(11.0/3.0)
| | | | | | Type of place of residence, v102!=(1): -1(103.0/68.0)
| | | | **Getting medical help for self: getting permission, v467b**!=(1): -1(1192.0/728.0)
| | **Pattern of use, v361**!=(3) | (4) | (2): -1(1299.0/624.0)

Summary: Number of leaf nodes 20, size of the tree 39

(7.6)

Full features dataset - no mild anemia cases

We now turn to assess the DTs generated on the full features dataset without mild anemic cases. This decision tree unsurprisingly selected the same root node as was selected on the balanced full features dataset DT. After that, it selected the feature *Currently pregnant, v213*, which asks if the respondent is pregnant or not leading to responses 0 for 'No or Unsure' and 1 for 'Yes'. From the root of the decision tree 323, not pregnant cases are classified as anemic.

CART Decision Tree

Getting medical help for self: getting permission, v467b=(0): 1(323.0/0.0)
Getting medical help for self: getting permission, v467b!=(0)
| Currently pregnant, v213=(1): 1(134.0/65.0)
| Currently pregnant, v213!=(1)
| | Getting medical help for self: concern no female hw,v467g=(0): 1(17.0/0.0)
| | Getting medical help for self: concern no female hw,v467g!=(0): -1(808.0/399.0)

Summary: number of leaf nodes 4, size of the tree 7 (7.7)

Controllable features - no mild anemia cases

This DT is almost entirely identical to the previous decision tree, having the same features selected and classifying nearly similar numbers of cases at the various decision node. Interestingly, this fact informs us that the most critical issues in anemia prediction are controllable as per the definition that we employed in this study.

CART Decision Tree

Getting medical help for self: getting permission, v467b=(0): 1(323.0/0.0)
Getting medical help for self: getting permission, v467b!=(0)
| Currently pregnant, v454=(1): 1(129.0/67.0)
| Currently pregnant, v454!=(1)
| | Getting medical help for self: concern no female hw, v467g=(0): 1(17.0/0.0)
| | Getting medical help for self: concern no female hw, v467g!=(0): -1(806.0/404.0)

Summary: number of leaf nodes 4, size of the tree 7(7.8)

7.3.3 Detailed model performance assessment

In this case, we only concentrate on the balanced sub-dataset scenarios, because they produced decision trees that performed well. Furthermore, to allow for a thorough analysis and comparison of performance, we generated decision trees with both CART and J48 decision tree algorithm, a C4.5 variant found in Weka.

We present a broader array of performance metrics for both decision trees algorithms, however for the CART algorithm we also provide detailed class performance metrics. More details are provided for the CART algorithm, because it outperformed the J48 algorithm, on most performance metrics for the various DT scenarios that we considered.

Detailed class performance for all the DTs revealed that the models performed well on the prediction of the negative anemia case. This result happened for both the scenario that used the entire $10,039$ cases or scenarios that selected a balanced sub-dataset from these datasets, in either the full features dataset or the controllable features sub-dataset.

Full features dataset decision trees

The J48 decision tree algorithm produced the following performance metrics: $1)$ accuracy 64.42%, $2)$ Kappa statistic 0.29, $3)$ mean absolute error 0.37 and $4)$ ROC area score of 67%.

The performance of the CART algorithm for the same scenario is as follows: $1)$ accuracy 68.70%, $2)$ Kappa statistic 0.37 and $3)$ mean absolute error 0.62.

We see here that the CART is better on measures of accuracy, Kappa and ROC area score. J48 did well on the mean absolute error, however overall the CART algorithm performed better.

The detailed per class performance is provided in table 7.3. We present in these tables sensitivity, precision, F score and ROC area score.

Controllable features dataset decision trees

In this case, first and foremost the J48 decision tree algorithm produced the following results: The performance on the controllable features dataset for the J48 algorithm was as follows: $1)$ correctly classified instances 63.92%, $2)$ Kappa statistic 0.28, $3)$ mean absolute error 0.38 and $4)$ ROC area score was 67%.

Algorithm	Sensitivity	Precision	F measure	ROC area	Class
CART	0.51	0.78	0.61	0.68	-1
	0.86	0.65	0.71	0.68	1
Weighted Av.	0.69	0.71	0.68	0.68	

TABLE 7.3: *Results for Simple CART decision tree classifier on full features dataset.*

On the other hand, the CART decision tree algorithm produced the following performance: 1) correctly classified instances 69.28%, 2) Kappa statistic 0.38 and 3) mean absolute error 0.61. Details are provided in table 7.4.

Algorithm	Sensitivity	Precision	F measure	ROC area	Class
CART	0.38	1.00	0.55	0.69	-1
	1.00	0.62	0.77	0.69	1
Weighted Av.	0.69	0.81	0.66	0.69	

TABLE 7.4: *Results for Simple CART decision tree classifier on controllable features dataset.*

Full features dataset: no mild anemia cases

This sub-section contains results for decision trees obtained on the full features dataset, for which all the mild anemia cases were removed. For the scenario without mild anemia, the J48 decision tree algorithm produced the following performance metrics: 1) correctly classified instances 68.50%, 2) Kappa statistic 0.37, 3) mean absolute error 0.33, and 4) ROC area score was 71%.

On the other hand, the CART decision tree algorithm for the same scenario produced the following results: 1) correctly classified instances 73.05%, 2) Kappa statistic 0.47 and 3) mean absolute error 0.53. We provide a detailed performance by class below in table 7.5;

Algorithm	Sensitivity	Precision	F measure	ROC area	Class
CART	0.50	0.92	0.65	0.73	-1
	0.96	0.66	0.78	0.73	1
Weighted Av.	0.73	0.79	0.72	0.73	

TABLE 7.5: *Results for Simple CART decision tree classifier on full features dataset without mild anemia cases.*

We finalise by showcasing the performance of the decision tree on the controllable features dataset without mild anemia.

Controllable features: no mild anemia cases

In this scenario the performance for J48 decision tree algorithm was as follows: 1) correctly classified instances 69.02%, 2) Kappa statistic 0.38 , 3) mean absolute error 0.34 and 4) a ROC area score of 70%.

The CART decision tree algorithm in this case performed as follows: 1) correctly classified instances 72.92%, 2) Kappa statistic 0.46 and 3) mean absolute error 0.54. Detailed class performance details are in the table 7.6

Algorithm	Sensitivity	Precision	F measure	ROC area	Class
CART	0.53	0.88	0.66	0.73	−1
	0.93	0.67	0.78	0.73	1
Weighted Av.	0.73	0.77	0.72	0.73	

TABLE 7.6: *Results for Simple CART decision tree classifier on controllable features dataset.*

Best decision tree models

As can be seen from the results given, DT models that we generated with balanced datasets were the best performers overall. These DT models yielded good scores on error measures, accuracy, ROC and the other metrics that we used. The DT models that performed best were as follows: 1) full features - without mild cases, with accuracy 73%, ROC area score 74% and mean absolute error of 0.54, finally 2) controllable features - without mild cases with accuracy 73% , ROC area score 73% and mean absolute error of 0.54. 3) controllable features, with accuracy 69%, ROC area score 69% and mean absolute error 0.61. 4) full features DT, with accuracy 70%, ROC area score 73% and mean absolute error of 0.38

This level of performance is not excellent. However, it is good enough for a classifier and sheds some more light on other important features in the classification of anemia which was done on our dataset.

On the other hand, models that used the entire set of cases, which was 10, 039, either for the full features dataset or the controllable features dataset, yielded high accuracy score, courtesy of excellent performance on predicting the highly represented negative anemia class. However, using measures such as Kappa and ROC area computation, we were able to confirm the overall poor prediction power for such models on our binary prediction problem.

These two measures take into account prediction accuracy of a random predictor in computing their performance metrics. In the cases of Kappa, a value close to 0 shows that the classifier performs just like a random classifier, while a value closer to 1 shows better predictive power. On the other hand, the ROC performance of 50% shows that performance is pretty random, a higher percentage of this value is preferable.

The controllable features dataset generated massive trees both in Weka as well as in Python. On the other hand, the sub-datasets without mild anemic cases for both full features dataset and the controllable features dataset produced good performance.

Even though the removal of the mild anemic cases from the datasets lead to reduction in the size of the dataset, the prediction accuracy was better. This may point to the fact that the lumping together of the various levels of being anemic may cause difficulties for a classifier to make correct predictions on class membership.

7.4 Discussion

In this section we highlight our findings and discuss them in the subsections: DT performance in 7.4.1, good features identification in 7.4.2 and afterwards conclude.

7.4.1 Decision tree performance

We attained the highest performance of 73% accuracy, 73% ROC area score and mean absolute error of 0.53. These results were compared with a study done by (Setotaw, 2013) who tackled a similar problem. This researcher was able to attain 76.8% accuracy for selected attributes and 91.9% on all features using a J48 decision tree algorithm. The dataset that was used by the researcher had some attributes that are different from the dataset that we used in this study, specifically certain health behavior attributes on alcohol consumption and use of chewing tobacco.

7.4.2 Features identified

We now move to the other reason that we chose to use DT algorithms, which is their understandability beyond the model building process. Hence, as stated earlier, we were able to identify a set of attributes that are the best predictors of anemia in our models.

What might be more useful are the themes revealed by these features identified by different decision trees generated from the dataset scenarios. Hence, an analysis of the features identified reveals the themes under which the various selected features belong. Knowing what themes are common in the DT models can provide more guidance on what issues are important and need to be focused on to tackle the anemia problem under consideration.

In the full features model, the Weka DT had five features from the total of 133 features namely: *'Getting medical help for self: getting permission, v467b'*, *'Getting medical help for self: concern no female hw, v467g'*, *'Cluster altitude in meters, v040'*, *'Ethnicity, v131'* and *'Current contraceptive method, v312'*. We classified these features into the themes of *health practices*, *geography*, *demographics* and *contraceptive use*. These results agree with the findings by (Setotaw, 2013), who worked on anemia prediction using a DHS dataset for Ethiopia. This author identified a set of attributes that fall in the same themes as the attributes that we identified, as strong anemia predictors. Moreover, these results agree with our findings in the SVM chapter 6 as will be discussed later.

On the other hand, the decision tree for the same full features dataset in python identified the 13 features in the following themes: *education*, *demographics*, *socio-economic*, *anthropometry*, *family planning*, *health practices* and *contraceptive use*.

When we review these themes concurrently, we observe that we have six main themes of interest for these set of decision trees. Furthermore, all these themes have been known to contribute and influence anemia prevalence according to (Setotaw, 2013; Akhwale et al., 2004; Guyatt and Snow, 2001; Rogerson et al., 2000).

On the other hand, The controllable features dataset decision tree highlighted the following thematic areas from the attribute set: *health practices*, *education*, *demographics*, *socio-economic*, *nutrition*, *anthropometry*, *family planning* and *contraceptive use*. Which was an increase to the initial attribute list and set of themes. However, these decision trees did not provide good performance, as was the case in the other sub-dataset scenarios.

Furthermore, features that directly speak to malaria, such as questions on *'type of bednets used'*, whether *'respondent slept under bednet'* and *'whether respondent took antimalarials during pregnancy'*, do not immediately get selected as dominant attributes in these decision tree models.

Even though literature shows that there is an undeniable linkage between malaria and anemia, whereby malaria is a cause of anemia which may be severe, moderate or mild (Guyatt and Snow, 2001; Rogerson et al., 2000). Though, in our dataset, these attributes did not have any substantial predictive power on anemia.

However, features that speak to the economic status of respondents such as ownership of a television, listening to the radio, type of wall material, type of roofing material and other related attributes were identified as key in the models.

In the decision trees that we generated, where the same datasets were used without the mild anemic cases, we observed an improvement in prediction performance, which was much better than what we attained with the controllable features dataset.

Even the decision trees for these sub-datasets that were generated in Weka using CART were very small having size 7 with 4 clean leaves. Regarding attributes selected these decision trees chose three attributes namely: *'Getting medical help for self: getting permission, v467b'*, *'Getting medical help for self: concern no female hw, v467g'* and *'Currently pregnant, v213'*. The new attribute, in this case, being *'Currently pregnant, v213'*, which is in agreement with what we found in literature, that anemia is more prevalent in pregnant women making it a key predictor for anemia (Malawi National Statistical Office and ICF Macro, 2010; Guyatt and Snow, 2001; Rogerson et al., 2000; Setotaw, 2013)

Finally, we can state that our study was able to highlight specific attributes that are important predictors of anemia from the dataset used. Furthermore, our findings were in agreement with what is known about the phenomenon of anemia and its various determinants. For instance, the correlation between altitude with malaria and anemia was discussed in a study on malaria and anemia prevalence at different altitudes in Kenya (Akhwale et al., 2004). This result was confirmed through our DTs where cluster altitude was identified as a very important predictor of anemia.

The other cases have been articulated earlier in this subsection, showing that our ML approach is a viable option, that can be utilized to assess anemia risk, without drawing any blood for tests. This can be achieved by analyzing readily available information on health practices, education, demographics, socioeconomic status, nutrition, anthropometry, pregnancy status, family planning and contraceptive use.

7.4.3 Comparison of features identified by DTs with SVM results

In this subsection we provide a comparison of the best features that were identified using SVM and DTs in table 7.7. In this case we tabulate top ten features for the results of the two supervised ML algorithms.

The features were extracted from the experimental runs that used the full features dataset for both SVM and the decision trees algorithms CART and J48. We propose that the list must not be read as a pure ranking of features based on their importance in the algorithms only, but rather as also a list of the most important features out of the 133 features that these supervised learning algorithms identified.

#	Decision Trees	Support Vector Machines
1	*Getting medical help for self: getting permission, v467b*	*De jure region of residence, v139*
2	*Getting medical help for self: concern no female hw, v467g*	*Type of toilet facility: v116*
3	*'Cluster altitude in meters, v040'*	*Getting medical help for self: concern no female hw, v467g*
4	*'Ethnicity, v131'*	*Getting medical help for self: distance to health, v467d*
5	*'Current contraceptive method, v312'*	*De facto place of residence: v134*
6	*Education in single years, v133*	*Currently pregnant: v454*
7	*Respondent slept under bednet, v461*	*During pregnancy took Fansidar for malaria: m49a1*
8	*Type of cooking fuel, v161*	*Results of salt iodine test, v166*
9	*Pattern of use, v361*	*Weight Height Percent ref median Fog: v443*
10	*Source of drinking water, v113*	*Has car, v125*

TABLE 7.7: *Top 10 predictors from SVM and DTs results.*

We observe that the two methods do not yield the same list of important features. Moreover, we note that there is only one feature in this list, namely $v467g$, on positions 2 and 3 respectively that is picked by both algorithms. However, we also note that the other closely related attributes on the 467 set of questions are picked by both algorithms these are $v467b$ by DTs on position 1 and $v467d$ by SVM on position 4. All these attributes fall under the *health practices* theme.

When we begin to assess the information content of the other features selected by the two algorithms, we realise that most of them fall in the same thematic areas as

mentioned before in section 7.4.2. The features include were *education*, *demographics*, *socio-economic*, *anthropometry*, *family planning*, *health practices* and *contraceptive use*. In this analysis due to the fact that, the features are many and the algorithms use different feature selection techniques, it is not surprising that the specific list of important features are different, in that difference however the features picked tend to fall in the same thematic areas. This realisation may point to the fact that the important issues underlying in the features as identified through the themes are important, and it is those specific themes that are more informative in the classification and prediction of anemia task.

The final point to make on this analysis is that since we compare mainly two machine learning algorithms it may be difficult to conclusively make a statement on the predictive powers of the themes identified separately by these two supervised machine learning methods. Nevertheless, when we consider the findings from the unsupervised machine learning experiments as provided in chapter 5, we can see a consistent pattern in the findings. Specifically, some themes identified by the clustering algorithms are also found in this analysis of supervised methods such as *demographics* and *health practices* themes. However, a third supervised machine learning algorithm might perhaps have been able to shed more light on predictive power and confirmed the influence of some of these features in classification of anemia.

7.4.4 Conclusion

In conclusion, in this chapter, we developed DT models that performed acceptably well for balanced sub-datasets obtained from our main datasets used in the study which are full features dataset and controllable features dataset.

Furthermore, we were able to identify a set of six major themes from the features that are powerful predictors of anemia from these datasets. This information was confirmed, using results generated using an attribute importance ranking function in Weka that uses information gain ratio. These findings were further related to the work that other researchers have done, and was found to be consistent.

Moreover, we were able to identify *health practices* theme whose attributes capture respondents autonomy in access to healthcare, as a key factor in the prediction for anemia. In this theme, issues like getting permission, needing transport or money, concern about the availability of female health workers can hinder this access to healthcare and impact on anemic status.

In similarity to this, we observed that most of the features that are selected as important features in the DT models are in the category of controllable issues, as per our definition of what is a controllable attribute in this study.

Moreover, we also learned that the predictive accuracy of our models was enhanced, by removing mild anemic cases from the dataset. These cases are borderline anemic which might make their presence in the dataset hard for DT classifier models to isolate. However, because our dataset had the least number of cases of severe anemia with the rest being moderate anemia, these might not have been adequate to allow for a perfect separation using this particular classifier.

Finally, we also grappled with the problem of data skewness which is common in machine learning datasets and has been addressed by others, using either over-sampling or under-sampling depending on the case. However, even if one under-samples an over-represented class, this does not necessarily increase the under-represented class. On the other hand, regarding oversampling, the same cases are reused many times. This approach might also mean that the model may still not be able to get exposed to different kinds of being severely anemic, for example, for the model to adequately learn that particular pattern for its predictive accuracy to increase.

8 Conclusion and future work

In this chapter we provide concluding remarks for our work and comment on a few important issues that we have learned in this research.

First of all, the task of extracting knowledge from population health datasets is essential and requires concerted efforts from diverse disciplines. One of the factors that make the knowledge extraction task from these datasets more significant is the fact that these datasets among other things are used to track the progress towards the attainment of global development goals such as the sustainable development goals.

Specifically, in this study, we showcased the significant role of machine learning for knowledge extraction on the demographic health dataset for Malawi. The work we did involved using unsupervised and supervised machine learning algorithms. The supervised methods involved building classifiers for anemia using support vector machines and decision tree algorithms. On the other hand, unsupervised ML algorithms included PCA, SOM, KMeans and EM algorithms, which we used to learn naturally existing groups in the dataset.

Moreover, we used the CRISP-DM methodology to guide and organize the data mining effort in the study. We narrowed our focus to zero-in on anemia which is covered by sustainable development goal number three which articulates the target of attaining good health for all. Our interest in the problem of anemia came about because of its relationship to malaria and the availability of attributes that measure it in the dataset. A focus on resolving this problem or contribution to its understanding will assist in the realization of the stated sustainable development goal. We will now address some key issues from the study in the subsequent sections.

8.1 Data preparation experience and proposed approach

The data mining effort commenced with obtaining the dataset and preparing it for ML experiments in the data preparation phase of the CRISP-DM. Data preparation was the longest phase in this research study in line with the projections from literature, where it is stated that this phase normally takes up most of the data mining time and effort. Our experience was that once we had thoroughly completed our data preparation, during experimentation we still had to return to the phase to fine tune some details. Part of this process was to try out different combinations of the data such as balancing cases, removing mild anemic cases and so on in order to generate useful models.

The process also revealed that quite a large number of the features in the dataset are categorical in nature. This clearly implies that the dataset may not be suitable for other machine learning algorithms that are not accommodative of such types of features. This knowledge informed our choices on what algorithms to use and

how to read and interpret our findings. The knowledge of what type of data one is dealing with, which is part of the data understanding phase is quite important and may impact the success of extracting knowledge from such population health datasets.

The other important lesson from the data preparation phase is the issue of feature selection. The DHS datasets by their nature are meant to capture a wide range of issues on health, demography, socio-economic issues just to name a few. Therefore, due the nature of DHS datasets there are a lot of variables in the dataset, hence selecting the best features that are relevant for the classification task at hand is a challenge. A combination of ML techniques for feature selection coupled with consultation with literature, questionnaires and data dictionary guides can prove useful for one to succeed in feature selection.

On the basis of these experiences and having studied the various data mining process models we can propose the following approach for handling population health datasets: $1)$ obtain the data, $2)$ clean the data, $3)$ review the cleaned data, $4)$ understand the data, $5)$ select the set of features relevant for the study, $6)$ select algorithms in line with type of features available, $7)$assess performance. Iterate $4-7$ until satisfied with the results.

8.2 Anemia classifiers

In this study, we have successfully shown how a select set of features from the demographic health dataset can be used for the prediction of anemia. Specifically, we demonstrated how specific classifier models could be built using support vector machines and decision trees.

In chapter 6 we demonstrated that SVM is effective for building anemia classification models. We identified a set of parameters such as best kernel, C, and γ, that lead to the development of the best performing SVM classifier models. Moreover, in chapter 7 we also successfully built decision tree models which allowed us to gain insights into the anemia classification problem further.

Therefore, using the combination of controllable select features and best model parameters, we can propose anemia prediction models on these datasets. We were able to show, that these models with fewer features performed equally well using the set of best model parameters identified. Through these findings, we have demonstrated what the key attributes from the DHS dataset are for predicting anemia.

8.3 From key features to themes

The importance of building classifier models lies in the fact that once one has developed the models, they can be used to isolate key attributes from the dataset for the prediction task at hand. This fact means that the importance of the models lies more in the knowledge that they reveal about a dataset through attribute relationships and ranking of attribute importance in the model, than merely their predictive power.

We, therefore, grouped the identified features into themes which are easier to communicate. The SVM models identified the following themes: *sanitation*, *health practices*, *geography*, *demographics*, *anthropometry* and *socio-economic*. While on the other hand DTs identified the following themes: *education*, *demographics*, *socio-economic*, *anthropometry*, *family planning*, *health practices* and *contraceptive use*

Regarding clustering results, we observed some striking clustering patterns for principal components analysis methods as well as clustering methods. These patterns though not in line with our initial expectations on how anemia and malaria-related attributes would cluster, were still informative. For instance, we learned consistent clustering patterns by age, wealth index and region. All of these patterns are in agreement with the available body of knowledge on anemia and its determinants.

Furthermore, some of the issues identified in the themes relate directly to malaria for example geography, health practices, education and socioeconomic status. These findings confirm the known relationship between anemia and malaria and points to the fact that a reduction in malaria will be significant for eliminating anemia.

8.4 Some key findings

Among the key findings of the study was that, in all the supervised ML models that we developed, we discovered that all the models that we experimented with identified the health practices attributes as essential attributes. These attributes were *'Getting medical help for self: getting permission, v467b'*, *'Getting medical help for self: getting money needed, v467c'*, *'Getting medical help for self: distance to health, v467d'*, *'Getting medical help for self: having to take trans, v467e'*, *'Getting medical help for self: not wanting to go, v467f'*, *'Getting medical help for self: concern no female hw,v467g'*.

These attributes had information that concerned having access to healthcare and the various hindrances that people face. The finding was that those facing no hindrances such as permission, distance, transport, and money in having access to medical care were mostly not anemic.

The second key finding was that the models identified other key features that are a confirmation of what is known about the condition of anemia. For example, anemia is more prevalent among pregnant women, and this is evidenced from the selection of attributes such as *'Currently pregnant, v454'*, *'Current contraceptive method, v312* and *'Pattern of use, v361'*. Another example is the selection of the feature *'Cluster altitude in meters, v040'* which reaffirms the link between anemia and malaria. Malaria is more prevalent in the low altitude areas, and thus there are more anemia cases in such areas than in the high altitude areas.

The third finding that we will discuss is from the SVM model results which identified *'Type of toilet facility'* as a key attribute for the prediction of being anemic. This result was not clearly identified as as issue in the literature that we reviewed. Therefore, it is a finding that is worth being interrogated further and to be related to the literature that might be available linking the issue to anemia.

Overall, all the algorithms that we used in the study generated a consistent set of themes for the most important attributes as discussed in the previous sub-section. The consistency between our findings with those that used other methods reaffirms

the usefulness of the machine learning algorithms in a variety of fields of applications.

Finally, in the plans for future work, we intend to do more work within the same area of research and perhaps build more supervised ML models for other infectious and non-communicable diseases using the DHS datasets from more countries. We will widen the scope to consider more countries within the southern African region to and assess the stability of the generated ML models across the region.

Bibliography

Achrekar, Harshavardhan et al. (2011). "Predicting flu trends using twitter data". In: *2011 IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS 2011*, pp. 702–707. DOI: 10.1109/INFCOMW.2011.5928903.

Achrekar, Harshavardhan et al. (2012). "Twitter improves seasonal influenza prediction". In: *HEALTHINF 2012 - Proceedings of the International Conference on Health Informatics*, pp. 61–70.

Akhwale, Willis S et al. (2004). "Anemia and malaria at different altitudes in the western highlands of Kenya". In: *Acta Tropica, Elsevier* 91, pp. 167–175. DOI: 10.1016/j.actatropica.2004.02.010.

Alpaydin, Ethem (2010). *Introduction to Machine Learning Second Edition*. Ed. by Thomas G. Dietterich et al. 2nd. Cambridge, Massachusetts: The MIT Press. ISBN: 9780262012430.

Apte, C et al. (1997). "Data Mining with Decision Trees and Decision Rules". In: *Computer Systems* 13.November, pp. 197–210.

Ayoya, Mohamed Ag et al. (2006). "Determinants of anemia among pregnant women in Mali". In: *Food and Nutrition Bulletin* 27.1, pp. 3–11. DOI: 10.1177%2F156482650602700101.

Azevedo, Ana and Manuel Filipe F . Santos (2008). "KDD , SEMMA AND CRISP-DM : A PARALLEL OVERVIEW". In: *IADIS European Conference on Data Mining*. January, pp. 182–185. ISBN: 9789728924638. URL: http://recipp.ipp.pt/bitstream/10400.22/136/3/KDD-CRISP-SEMMA.pdf.

Barakat, Nahla H, Andrew P Bradley, and Mohamed Nabil H Barakat (2010). "Intelligible support vector machines for diagnosis of diabetes mellitus." In: *IEEE transactions on information technology in biomedicine : a publication of the IEEE Engineering in Medicine and Biology Society* 14.4, pp. 1114–1120. ISSN: 1558-0032. DOI: 10.1109/TITB.2009.2039485.

Berkhin, Parvel (2006). "A Survey of Clustering Data Mining Techniques". In: *Grouping Multidimensional Data* c, pp. 25–71. ISSN: 1557900X. DOI: 10.1007/3-540-28349-8_2.

Brossette, Se et al. (1998). "Association rules and data mining in hospital infection control and public health surveillance." In: *Journal of the American Medical Informatics Association : JAMIA* 5.4, pp. 373–81. ISSN: 1067-5027. DOI: 10.1136/jamia.1998.0050373.

C, Burges Christopher J (1998). "A Tutorial on Support Vector Machines for Pattern Recognition". In: *Data Mining and Knowledge Discovery*. NetGames '06 2.2. Ed. by Usama Fayyad, pp. 121–167. ISSN: 13845810. DOI: 10.1023/A:1009715923555. arXiv:1111.6189v1.

Calis, Job C.J. et al. (2008). "Severe Anemia in Malawian Children". In: *The new England journal of medicine* 358, pp. 888–899. DOI: 10.1056/NEJMoa072727.

Campbell, A. J. et al. (2008). "Predicting death and readmission after intensive care discharge". In: *British Journal of Anaesthesia* 100.5, pp. 656–662. ISSN: 14716771. DOI: 10.1093/bja/aen069.

Centers for Disease Control and Prevention (2012). *CDC ′ s Vision for Public Health Surveillance in the 21 st Century*. Tech. rep. Atlanta: Centers for Disease Control and Prevention (CDC), pp. 1 –44.
Chapman, Pete et al. (2000). *Crisp-Dm 1.0*. Tech. rep. IBM SPSS, p. 76.
Chih-Wei Hsu, Chih-Chung Chang and Chih-Jen Lin (2008). "A Practical Guide to Support Vector Classification". In: *BJU international* 101.1, pp. 1396–400. ISSN: 1464-410X. arXiv:0-387-31073-8. URL: https://mcs.une.edu.au/doc/libsvm/guide.pdf.
Cohen, Gilles et al. (2004). "An application of one-class support vector machine to nosocomial infection detection." In: *MedInfo:2004* 107.Pt 1, pp. 716–20. ISSN: 0926-9630. DOI: 10.3233/978-1-60750-949-3-716.
Cox, R.T. T. (1946). "Probability, Frequency Reasonable Expectation". In: *American Journal of Physics* 14.1, p. 1. ISSN: 00029505. DOI: 10.1119/1.1990764. arXiv:arXiv:1011.1669v3.
Ermert, Volker et al. (2011). "Development of a new version of the Liverpool Malaria Model. II. Calibration and validation for West Africa". In: *Malaria Journal* 10.1, p. 62. ISSN: 1475-2875. DOI: 10.1186/1475-2875-10-62. URL: http://malariajournal.biomedcentral.com/articles/10.1186/1475-2875-10-62.
Ermert, Volker et al. (2012). "Development of dynamical weather-disease models to project and forecast malaria in Africa".
Fayyad, Usama, Gregory Piatetsky-Shapiro, and Padhraic Smyth (1996). "Knowledge discovery and data mining: Towards a unifying framework". In: *Knowledge Discovery and Data Mining*, pp. 82–88. URL: http://www.aaai.org/Papers/KDD/1996/KDD96-014.pdf.
Fialho, A S et al. (2012). "Data mining using clinical physiology at discharge to predict ICU readmissions". In: *Expert Systems with Applications* 39, pp. 13158–13160. DOI: 10.1016/j.eswa.2012.05.086.
Furey, Terrence S et al. (2000). "Support Vector Machine Classification and Validation Of Cancer Tissue Samples Using Microarray Expression Data". In: *Bioinformatics* 16.10, pp. 906–914. DOI: 10.1093/bioinformatics/16.10.906.
Government of Malawi (2014). *Malawi Malaria Indicator Survey*. Tech. rep. Lilongwe: Ministry of Health, Government of Malawi.
Guyatt, Helen L and Robert W Snow (2001). "The Epidemiology and Burden of Plasmodium Falciparum -Related Anemia Among Pregnant Women in Sub-Saharan Africa". In: *American Journal of Tropical Medicine and Hygiene* 64, pp. 36–44. DOI: 10.4269/ajtmh.2001.64.36.
Guyon, Isabelle et al. (2002). "Gene Selection for Cancer Classification Using Support Vector Machines". In: *Machine Learning* 46, pp. 389–422. DOI: 10.1023/A:1012487302797.
Hamel, Lutz (2009). "Model Assessment with ROC Curves". In: *Encyclopedia of data warehousing and mining*. Hamel2009. DOI: doi:10.4018/978-1-60566-010-3.ch204.
Herland, Matthew, Taghi M Khoshgoftaar, and Randall Wald (2014). "A review of data mining using big data in health informatics". In: *Journal of Big Data* 1.2. DOI: https://doi.org/10.1186/2196-1115-1-2.
Hoshen, Moshe B and Andrew P Morse (2004). "A weather-driven model of malaria transmission". In: *Malaria journal* 14, pp. 1–14. DOI: 10.1186/1475-2875-3-32.
IBM (2011). *IBM SPSS modeler CRISP-DM guide*. Tech. rep. IBM.
Ibrahim, Heba et al. (2016). "Mining association patterns of drug-interactions using post marketing FDA's spontaneous reporting data". In: *Journal of Biomedical Informatics* 60, pp. 294–308. ISSN: 15320464. DOI: 10.1016/j.jbi.2016.02.009.

Jain, Anil K. (2010). "Data clustering: 50 years beyond K-means". In: *Pattern Recognition Letters* 31.8, pp. 651–666. ISSN: 01678655. DOI: 10.1016/j.patrec.2009.09.011. arXiv:0402594v3 [arXiv:cond-mat]. URL: http://dx.doi.org/10.1016/j.patrec.2009.09.011.

Kaski, S and T Kohonen (1996). "Exploratory Data Analysis By The Self-Organizing Map : Structures Of Welfare And Poverty In The World". In: *3rd International Conference on Neural Networks in the Capital Markets*, London, England, pp. 498–507. DOI: 10.1.1.53.3954.

Kesorn, Kraisak et al. (2015). "Morbidity rate prediction of dengue hemorrhagic fever (DHF) using the support vector machine and the Aedes aegypti infection rate in similar climates and geographical areas". In: *PLoS ONE* 10.5, pp. 1–16. ISSN: 19326203. DOI: 10.1371/journal.pone.0125049.

Kohonen, Teuvo (1998). "The self-organizing map". In: *Neurocomputing* 21.May, pp. 1–6. DOI: 10.1016/S0925-2312(98)00030-7.

Koua, Etien L and Menno-Jan Kraak (2004). "Geovisualization to support the exploration of large health and demographic survey data". In: *International journal of health geographics* 3, p. 12. ISSN: 1476-072X. DOI: 10.1186/1476-072X-3-12.

Lee, Yoonkyung and Cheol-koo Lee (2003). "Classification of multiple cancer types by multicategory support vector machines using gene expression data". In: *Bioinformatics* 19.9, pp. 1132–1139. DOI: 10.1093/bioinformatics/btg102.

Leskovec, Jure, Anand Rajaraman, and Jeffrey D. Ullman (2014). *Mining of Massive Datasets*. California, USA.

Malawi National Statistical Office and ICF (2017). *Malawi Demographic and health survey 2015-16*. Tech. rep. Zomba, Malawi: National Statistical Office(NSO) Zomba Malawi, ICF Rockville Maryland USA.

Malawi National Statistical Office and ICF Macro (2010). *2010 Malawi Demographic and Health survey key findings*. Tech. rep. Zomba.

Marucci-wellman, Helen R, Mark R Lehto, and Helen L Corns (2015). "A practical tool for public health surveillance : Semi-automated coding of short injury narratives from large administrative databases using Naïve Bayes algorithms". In: *Accident Analysis and Prevention* 84, pp. 165–176. ISSN: 0001-4575. DOI: 10.1016/j.aap.2015.06.014.

Mathias, Jason Scott et al. (2013). "Development of a 5 year life expectancy index in older adults using predictive mining of electronic health record data". In: *Journal of American Medical Informatics Association* 20, e118–e124. DOI: 10.1136/amiajnl-2012-001360.

Murphy, Sean C and Joel G Breman (2001). "Gaps in the childhood Malaria Burden in Africa : Cerebral Malaria , Neurological Sequelae , Anemia , Respiratory Distress, Hypoglycemia , and Complications of Pregnancy". In: *Journal of Tropical Medicine and Hygiene* 64.1, pp. 57–67. DOI: 10.4269/ajtmh.2001.64.57.

Nenonen, Noora (2013). "Analysing factors related to slipping, stumbling, and falling accidents at work: Application of data mining methods to Finnish occupational accidents and diseases statistics database". In: *Applied Ergonomics* 44.2, pp. 215–224. ISSN: 00036870. DOI: 10.1016/j.apergo.2012.07.001.

Ng, Nawi et al. (2009). "Using the INDEPTH HDSS to build capacity for chronic noncommunicable disease risk factor surveillance in low and middle-income countries." In: *Global health action* 2.May, pp. 7–18. ISSN: 1654-9716. DOI: 10.3402/gha.v2i0.1984.

Palaniappan, S. and R. Awang (2008). "Intelligent heart disease prediction system using data mining techniques". In: *2008 IEEE/ACS International Conference on Computer Systems and Applications* 8.8, pp. 343–350. DOI: 10.1109/AICCSA.2008.4493524.

Pedregosa, Fabian, Ron Weiss, and Matthieu Brucher (2011). "Scikit-learn : Machine Learning in Python". In: *Journal of Machine Learning Research* 12, pp. 2825–2830.

Quinlan, J R (1986). "Induction of Decision Trees". In: *Machine Learning* 1, pp. 81–106. DOI: 10.1007/BF00116251.

Ramakrishnan, Naren et al. (2009). *The top 10 algorithms in data mining*. Ed. by Xidong Wu and Vipin Kumar. First edit. New York, USA: Chapman and Hall. ISBN: 9781420089646.

Rogerson, S. J. et al. (2000). "Malaria and anemia in antenatal women in Blantyre, Malawi: A twelve-month survey". In: *American Journal of Tropical Medicine and Hygiene* 62.3, pp. 335–340. ISSN: 00029637. DOI: 10.4269/ajtmh.2000.62.335.

Rutstein, Shea, Rebecca Winter, and I C F International (2014). *Association among Economic Growth, Coverage of Maternal and Child Health Interventions, and Under-Five Mortality: A Repeated Cross-Sectional Analysis of 36 Sub-Saharan African Countries. DHS Analytical Studies No. 38*. Tech. rep. February 2014. Rockville, Maryland, USA: ICF International, pp. 1–58.

Rutstein, Shea Oscar and Guillermo Rojas (2013). *Guide to DHS statistics Demographic and Health Surveys Methodology*. Tech. rep. Calrverton, Maryland: Demographic and health surveys, ORC Macro.

Sarlin, Peter (2012). "Visual tracking of the millennium development goals with a fuzzified self-organizing neural network". In: *International Journal of Machine Learning and Cybernetics* 3.3, pp. 233–245. ISSN: 18688071. DOI: 10.1007/s13042-011-0057-5.

Setotaw, Getachew (2013). "Predicting the status of anemia in women aged 15-49 by applying data mining techniques using 2011 Ethiopian demographic and health survey datasets". PhD thesis. Addis Ababa University.

Sharma, Vijeta et al. (2015). "Malaria Outbreak Prediction Model Using Machine Learning". In: *Journal of Advanced Research in Computer Engineering and Technology* 4.12, pp. 4415–4419. URL: http://ijarcet.org/wp-content/uploads/IJARCET-VOL-4-ISSUE-12-4415-4419.pdf.

Sié, Ali et al. (2010). "The Health and Demographic Surveillance System (HDSS) in Nouna, Burkina Faso, 1993-2007." In: *Global health action* 3, pp. 1–10. ISSN: 1654-9716. DOI: 10.3402/gha.v3i0.5284.

Signorini, Alessio, Alberto Maria Segre, and Philip M Polgreen (2011). "The Use of Twitter to Track Levels of Disease Activity and Public Concern in the U . S . during the Influenza A H1N1 Pandemic". In: *PLoS ONE* 6.5. DOI: 10.1371/journal.pone.0019467. URL: www.plsone.org.

Sudheer, CH et al. (2014). "Neurocomputing A Support Vector Machine-Fire fl y Algorithm based forecasting model to determine malaria transmission". In: *Neurocomputing* 129, pp. 279–288. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2013.09.030. URL: http://dx.doi.org/10.1016/j.neucom.2013.09.030.

Tourassi, Georgia, Hong Jun Yoon, and Songhua Xu (2016). "A novel web informatics approach for automated surveillance of cancer mortality trends". In: *Journal of Biomedical Informatics* 61, pp. 110–118. ISSN: 15320464. DOI: 10.1016/j.jbi.2016.03.027.

United Nations (2015a). *Taking Stock of the Global Partnership for Development, MDG Task Force Report 2015*. Tech. rep. United Nations.

— (2015b). *The Millennium Development Goals Report 2015*. Tech. rep. New York: United Nations.
Vapnik, Vladimir N (1999). "An Overview of Statistical Learning Theory". In: *IEEE Transactions on neural networks* 10.5, pp. 988–999. DOI: 10.1109/72.788640.
Wang, Wenjuan et al. (2015). *DHS Analytical Studies No. 51: The Relationship Between the Health Service Environment and Service Utilization: Linking Population Data to Health Facilities Data in Haiti and Malawi*, tech. rep. September. Rockville, Maryland: USAID, pp. xv–21. URL: https://www.dhsprogram.com/pubs/pdf/AS51/AS51.pdf.
Wirth, Rüdiger and Jochen Hipp (2000). *CRISP-DM : Towards a Standard Process Model for Data Mining*. Tech. rep. 24959. Daimler Chrysler Research, pp. 29–39.
Yu, Wei et al. (2010). "Application of support vector machine modeling for prediction of common diseases: the case of diabetes and pre-diabetes." In: *BMC medical informatics and decision making* 10, p. 16. ISSN: 1472-6947. DOI: https://doi.org/10.1186/1472-6947-10-16.
Zhang, Yang et al. (2012). "Real-time clinical decision support system with data stream mining". In: *Journal of Biomedicine and Biotechnology* 2012, pp. 1–8. ISSN: 11107243. DOI: 10.1155/2012/580186.