Analysis of Classification Techniques for Prediction of Tuberculosis Defaulters

Brian Mc George
University of Cape Town
Cape Town, South Africa
mcgbri004@myuct.ac.za

ABSTRACT

1. INTRODUCTION

In 2013 over 210 000 patients defaulted from Tuberculosis (TB) treatment worldwide [35]. The rate of default in the Americas is the highest at 8% with Africa at 5% [35]. The consequences of defaulting TB treatment include: increased drug resistance, increased health system costs [22, 27], higher risk of mortality, continued risk of transmitting the disease to others [22] and increased rate of recurrent disease [18]. The spread of TB can be reduced if the individuals who have a high risk of defaulting can be predicted. This will also reduce health system costs.

In this paper we examine a wide range of classification techniques at their ability to predict patients as defaulters. We explore a number of techniques and strategies to improve the overall classification result. Most notably we look at how well these classifiers work out-of-the-box compared to when they are hand optimised or tuned by searching a grid of parameters. We examine different data balancing techniques which either over-sample the minority class or under-sample the majority class. Furthermore, to see how each classifier responds to having fewer features available, we apply a number of feature selection strategies on each classification technique.

To facilitate the aforementioned experiments, we developed a testing system which allows new classification techniques and data sets to be supported quickly and easily. The testing system is designed to allow near-exact reproducibility of results.

The field of credit scoring in the financial space aims to determine if a financial institution should provide credit to an individual. This area has been well researched. In this paper we aim to determine if our results generalise across the different fields. There are notable similarities in these problems which could make them comparable. Both problems typically have a labelled dataset consisting of both nominal and numerical data as well as the occurrence of missing data [citation needed]. However, TB datasets are more prone to

missing data as well as inaccuracies due to the nature of the data collection [citation needed]. The selected classification techniques are evaluated against real-world treatment default datasets and financial datasets. The paper will evaluate how the techniques differ across the datasets. If the relative results are similar then future credit scoring research could be applicable to treatment default prediction too.

2. RELATED WORK

2.1 Definition of a defaulter

The definition of a defaulter depends on its context. TB literature typically uses the World Health Organisation (WHO) definition that a defaulter is a person whose treatment has been disrupted for two or more consecutive months [6, 9, 18, 19, 27, 35].

2.2 Determining predictors of TB default

There have been many studies which focus on determining the factors associated with TB default but few have used machine learning techniques to predict treatment defaulters. Table 1 contains an overview of a selection of publications on determining the factors associated with TB default. The majority of techniques use a form of logistic regression ti determine the association.

The datasets used by the publications contain different features. Age and gender are common throughout the datasets. History of past default is available for all datasets except for Shargie et al. [31]. Lackey et al. [22] only picked individuals who did not have a history of past default. Jittimanee [19] [28] was the only publication with the feature that did not find it to be significant to the 95% confidence level. However, it did have an odds ratio of 2.19 and a p-value of 0.12. It can therefore be deduced that a history of past default has a strong correlation to default. Two out of three publications with the alcohol abuse feature available, found it to be significant. Three of the four publications with side effects, as a feature found it was significant. Shargie et al. [31] and Jittimanee et al. [19] measured distance and time to treatment site respectively. It can be reasoned that the aforementioned feature's significance will generalise to other datasets since they were found to be significant in the majority of the publications. Other significant features such as illegal drug use, use of herbal medication, daily jobs, history of lung cancer and history of liver disease only appeared once in the datasets. It cannot be discerned if the significance is generalisable or specific to the dataset. The identification of the same features as significant is fairly consistent for the

publications that have those features in their dataset.

2.3 Predicting defaulters in financial institutions

3. BACKGROUND

This section aims to provide an overview of all the techniques used in this paper.

3.1 Classifiers

This section examines the different classification techniques and outlines how each technique produces its output.

3.1.1 Support Vector Machines

Support vector machine (SVM) is a machine learning technique that can be used to produce regression or classification functions from a set of training data [25]. SVM works by mapping the input vectors into a high-dimensional feature space with the use of a kernel function [10]. The kernel function selected determines the if the mapping is linear or non-linear [25]. Linear, polynomial and radial basis function (RBF) are common kernel functions [15]. The polynomial and RBF kernel performs a non-linear mapping into the high-dimensional space [15]. This feature space is then searched to acquire an optimal hyperplane that separates the space with the maximum distance between the two classes [10]. Hsu *et al.* [15] recommends the RBF kernel as a reasonable first choice but notes that it is not suitable when there are a large number of features. The linear kernel is recommend when there are a large number of features.

3.1.2 Artificial Neural Network

An artificial neural network (ANN) is based on the functionality of the human brain [34]. Neurons in the brain are interconnected and process information in parallel [34]. A typical ANN is comprised of an input layer, k number of hidden layers and an output layer. The neurons in each layer are connected to the neurons in the next layer. A numeric weight is defined between each pair of connected neurons. An activation function defines if a neuron will fire [34]. The activation function bounds the value of a neuron to a specific range to limit the effect of divergent neurons [34]. By using an activation function, a non-linear combination of weights can be generated [34]. It has been proven that an ANN is able to approximate any continuous function if has at least one hidden node and an activation function that is both bounded to some range and non-constant [14].

Extreme learning machines (ELM) is an alternative approach to the conventional back-propagated ANN. Huang *el al.* [17] proved that the input and hidden layer weights can be randomly assigned if the activation function in the hidden layer is infinitely differentiable. By randomly assigning these weights, the weights for the output nodes can be determined analytically [17]. This allows ELMs to be trained orders of magnitude faster than a back-propagated ANN [17] [16]. Various studies have shown ELMs to provide better results than SVMs and ANNs on classification and regression tasks [17] [16].

3.1.3 Logistic Regression

Logistic regression is a technique that models the chance of an outcome based on the input features [30]. Since chance is a ratio, the logarithm of the chance is what is modelled instead [30]: $\log(\frac{\mathcal{P}}{1-\mathcal{P}}) = \beta_0 + \beta_1 x_1 + ... + \beta_m x_m$. $\mathcal{P}$ represents the probability of an event (likelihood to default for example). β_0 represents the value of the criterion when the predictor is equal to 0. $\beta_1, ..., \beta_m$ are the regression coefficients associated with the $x_1, ..., x_m$ input features. The probability of an event can then be calculated as $\mathcal{P} = \frac{1}{1+e^{-(\beta_0+\beta_1 x_1+...+\beta_m x_m)}}$. A detailed overview of logistic regression can be found in [26] and [30].

3.1.4 k-Nearest Neighbours

The k-Nearest Neighbours (kNN) algorithm determines the output classification by examining the k nearest training examples in the feature space [20]. An input is classified by the majority vote of its neighbours.

3.1.5 Ensembles

An ensemble classifier is one which typically makes use of multiple classifiers and aggregate the results. A decision tree can be used for classification by branching on conjunctions of features and having the leaves represent the output class. A decision tree allows for easy interpretation of the generated model, however, typically provides relatively poor classification accuracy [33].

Random forest is a technique that fits a number of decision trees on *random samples with replacement* of the dataset. For each tree, n features are randomly selected and the tree is grown [5]. Data that was not selected for training the tree is called out-of-bag data [5]. It is used to test the error rate of the tree [5, 33]. The output classification is decided by the majority vote of each tree [5].

Freund and Schapire's [11] AdaBoost uses a weighted sum of multiple classifiers in order to determine the output. A classifier is constructed in an iterative fashion. At each iteration a pool of classifiers are considered and one classifier is added to the committee of classifiers. Input which is still misclassified is assigned a higher weight at each iteration [4, 29]. The aim being that the process will select a classifier which helps the still misclassified inputs. The chosen classifier is assigned a weight which determines its power in the overall classification [4, 29]. The sign of the function: $C(x) = \sum_{i=1}^{m} \alpha_i k_i$ is used to determine the final classification with α_i denoting the weight of each classifier k_i [4].

3.1.6 Naive Bayes

Naive Bayes makes an assumption that each input feature is independent of each other [24, 28]. This allows multiple features be uncoupled from one another which simplifies the algorithm. Naive Bayes uses conditional probability to classify new inputs [24]. Using Bayes theorem, the following equation is derived: $p(C_i|\mathbf{x}) = \frac{p(C_i) \times p(x_1|C_i) \times ... \times p(x_n|C_i)}{p(\mathbf{x})}$ with input $\mathbf{x} = (x_1, ..., x_n)$ and class label C_i [24, 28]. Since $p(x)$ is identical for each class, it is typically ignored [28]. To classify an input, the probabilities for each class are calculated using the aforementioned equation and the output is the class with the highest probability [24].

3.1.7 Clustering-launched classification

Clustering-launched classification (CLC) is a binary classifier. CLC first clusters the data into groups using diverse hierarchical k-means algorithm [25]. The clusters are divided into positive subsets and negative subsets [25]. Support vectors are then used to separate the positive and neg-

Table 1: Overview of publications on predictors of TB treatment defaulters

Publication	Sample Size	Key factors identified[*]	Evaluation
Chan-Yeung et al. [6]	1768 non-defaulters 442 defaulters.	History of default, history of lung cancer, liver disease and male patients	Multiple logistic regression is used to determine what factors are associated with default.
Jha et al. [18]	1189 non-defaulters 1141 defaulters.	Male patients, patients directly-observed treatment at public facilities, previous treatment outside India's Revised National Tuberculosis Control Programme and history of previous default	The chi-square test or Fisher's exact test (if there were less than 10 observations) was used to test the differences between defaulters and non-defaulters. Bivariate analysis was calculated on the features. Multivariate logistic regression using pre-selected features based on previous studies.
Jittimanee et al. [19]	106 non-defaulters 54 defaulters.	Jobs where one is only paid if one is at work that day, severe medication side-effects and time to travel to clinic.	Patients were interviewed and completed a questionnaire to obtain the information. Hierarchical logistic regression was carried out to assess the variable's relation to default.
Lackey et al. [22]	1106 non-defaulters 127 defaulters.	Has used illegal drugs, has multidrug-resistant TB, has not been tested for HIV, drinks alcohol at least once a week, underweight or has not completed secondary education.	Patients were interviewed to obtain the information. Bivariate analysis using Chi-square tests and odds ratios with 95% confidence intervals. Multivariate logistic regression is used with a backward fitting algorithm to determine if a variable is associated with default.
Muture et al. [27]	5659 non-defaulters 945 defaulters.	Inadequate knowledge on TB, herbal medication use, low income, alcohol abuse, previous default, suffering from HIV and male patients were determined through analysis to be associated with default.	Two-tailed χ^2 tests and Fisher exact tests (if a cell has less than 5 values) to assess categorical information. Odds ratio tests were used to measure association between features with a 95% confidence interval.
Shargie et al. [31]	310 non-defaulters 81 defaulters.	Distance from home to treatment site, age greater than 25 and the need to use public transport to get to treatment site.	Continuous features were analysed using sample t-test χ^2 tests and hazard ratios with 95% confidence intervals. The effect of each factor was assessed using Cox regression model with a backwards fitting algorithm.

[*]Ordered in descending order of significance based on odds ratio and hazard ratio at a confidence interval of at least 95%.

Table 2: German data set[¶]

Publication	Accuracy	Type I Error	Type II Error
Huang et al. [?]	**79.87%**[†]	n/a	n/a
Li et al. [?]	**84.83%**	10-20%[*]	10-20%[*]
Luo et al. [25]	77.06% (MySVM), 82.41% (SVM-GA)[†]	n/a	n/a
Huang et al. [?]	82.41% (SVM-GA)[†]	n/a	n/a
Danenas et al. [10]	**94.41%** (Linear SVM), **92.37%** (PSO-LinSVM)[†]	n/a[‡]	n/a[‡]

Table 3: Australian data set[¶]

Publication	Accuracy	Type I Error	Type II Error
Huang et al. [?]	**79.87%**[†]	n/a	n/a
Li et al. [?]	**84.83%**	10-20%[*]	10-20%[*]
Luo et al. [25]	77.06% (MySVM), 82.41% (SVM-GA)[†]	n/a	n/a
Huang et al. [?]	82.41% (SVM-GA)[†]	n/a	n/a
Danenas et al. [10]	**94.41%** (Linear SVM), **92.37%** (PSO-LinSVM)[†]	n/a[‡]	n/a[‡]

ative subsets for each cluster [25]. Redundant or repeated support vectors are removed thereafter [25].

3.2 Data Balancers

3.2.1 Over-sampling

Random over-sampling (RUS) is a technique that randomly replicates examples in the minority class [3]. However, this technique can increase the likelihood of over-fitting since exact copies are made from the minority class [3].

The Synthetic minority over-sampling technique (SMOTE) [7] forms new minority samples by interpolating along the line segment on some or all of the k nearest minority class neighbours of a minority example. This approach attempts to alleviate the over-fitting that can occur from using random over-sampling.

Adaptive synthetic sampling (ADASYN) [13] is a variation of SMOTE which uses a weighed distribution for different minority class examples according to their difficulty in learning. More synthetic data is generated for minority class examples that are more difficult to learn compared to those that are easier to learn.

3.2.2 Under-sampling

Random over-sampling (ROS) is a technique that randomly eliminates examples in the majority class [3]. However, this technique can remove potentially useful information from the training set [3].

Condensed Nearest Neighbour (CNN) rule [12] finds a consistent subset of examples. A subset D of E is consistent if using 1-nearest neighbour using D correctly classifies E [3]. The process draws one random majority class example and all minority class examples and puts them in D [3]. Every misclassified example is then moved from E to D [3]. This process attempts to remove examples that are far away

from the decision border and therefore seen as less relevant for learning [3].

The Tomek link (TL) algorithm [1] examines two examples $\mathbf{x}_i$ and $\mathbf{x}_j$ belonging to different classes. A TL occurs if there is not an example $\mathbf{x}_l$ such that $d(\mathbf{x}_i, \mathbf{x}_l) < d(\mathbf{x}_i, \mathbf{x}_j)$ or $d(\mathbf{x}_j, \mathbf{x}_l) < d(\mathbf{x}_j, \mathbf{x}_i)$. If two examples form a TL then either one is noise or it is a borderline case [3]. This information can then be used to under-sample the majority class [3].

One-sided selection (OSS) [21] applies TL to remove borderline and noisy majority class examples then applies CNN to remove majority examples far from the decision border.

Neighbourhood cleaning rule (NCL) [23] uses the edited nearest neighbour rule (ENN) to remove majority class examples. ENN removes examples whose label differs from the class of at least two of its nearest 3 neighbours. NCL examines each example $\mathbf{x}_i$ and its 3 nearest neighbours. If $\mathbf{x}_i$ belongs to the majority class and the neighbours contradict this class then $\mathbf{x}_i$ is removed [3]. If $\mathbf{x}_i$ belongs to the minority class and the neighbours contradict this class then the neighbours from the majority class are removed [3].

Instance threshold hardening (ITH) [32] uses the probability estimates from a classifier when k-fold cross validation is applied to only select those examples from the majority class that have the highest probability in order to match the number of examples in the minority class.

NearMiss-1 (NM-1) [36] picks majority class examples which have the smallest average distance to the three nearest minority class examples.

Cluster centroids (CC) applies the k-means algorithm with N clusters to the majority class and uses the coordinates of the cluster centroids as the majority sample.

3.2.3 Combination

SMOTE + TL [2] first applies SMOTE then applies TL. Applying SMOTE can cause minority samples to extend too deep into the majority class space or the opposite where majority class samples extend too deep into the minority class space [2]. TL is used as a data cleaning method to remove examples from both classes to produce well-defined class clusters [2].

SMOTE + ENN works similarly to SMOTE + TL but ENN is more aggressive at removing samples than TL [3].

3.3 Metrics

The number of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) are used to define several metrics which are used to compare the results in this paper. The true positive rate (TPR) or sensitivity defines the proportion of actual positives which are predicted as positive. The TPR is calculated as $\frac{TP}{TP+FN}$. The true negative rate (TNR) or specificity defines the proportion of actual negatives which are predicted as negative. The TNR is calculated as $\frac{TN}{TN+FP}$.

Accuracy is typically a poor measure of quality for imbalanced datasets as classifiers tend to bias towards the majority class [3, 8]. Several balanced performance metrics are used instead. The balanced accuracy (BACC) provides an equal weighting in TPR and TNR and is calculated as $\frac{TPR+TNR}{2}$. Matthews correlation coefficient (MCC) also provides a balanced measure of classification quality and is scored between -1 and 1. The MCC is calculated as: $\frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$. Informedness is similar to balanced accuracy but ranges between -1 and 1 and is defined as $TPR + TNR - 1$.

4. METHODS

4.1 Approach

We aimed to evaluate a set of statistical and machine learning classification techniques on treatment default data sets and compare it to that applied to other related classification datasets. A variety of classification techniques were chosen to test against: ANN, SVM using a linear, RBF and polynomial kernel, logistic regression, decision tree, AdaBoost, random forest, kNN, Gaussian naive Bayes, Bernoulli naive Bayes, ELM and CLC. These were chosen to include a selection of well known techniques, ensemble techniques and newer techniques that have shown promising results in other studies.

Many datasets have substantially fewer defaulters than those cured. However, most classification algorithms expect a 50:50 split between classes. We therefore include an analysis of several data balancing algorithms: CC, ENN, IHS, NM-1, NCR, OSS, RUS, TL, ADASYN, ROS, SMOTE, ensemble of SMOTE + EEN and ensemble of SMOTE + TL. These balancing techniques were chosen to facilitate an evaluation of both over-sampling and under-sampling techniques. Included in the evaluation is techniques that use simple approaches and as others that use sophisticated algorithms.

All the experiments use stratified 5-fold validation to divide a single dataset into multiple training and testing datasets. Stratified k-fold divides the dataset into k segments and ensures that each segment contains the same ratio of positive and negative examples as the dataset as a whole. k number of training and testing sets are created by using one segment as the testing dataset and every other segment as the training data. This is repeated such that every segment is used as testing data. Results for each fold are then averaged before being presented.

4.1.1 Parameter tuning

We want to determine how important parameter tuning is for our datasets and its effect on each classifier. In addition we want to see if the addition of a data balancing algorithm could improve the classification. We first test each classifier at its default parameters. In the case where a classifier does not have default parameters available, we provide our own reasonable defaults and label those classifiers accordingly. To see the effect of a data balancing algorithm on the classifiers with default parameters, we will use the data balancer that results in the highest BARR. Finally, for each classifier we search a grid of reasonable parameters and select the parameters that yield the highest BARR. The grid includes the parameters of the classifier as well as the different data balancing techniques.

Over-fitting the classifiers was a concern when applying the grid search. To prevent the classifiers from over-fitting the parameter grid was kept fairly coarse and three runs of 5-fold cross-validation was executed and the results averaged. To provide fair comparison between parameter sets, the data balancer, stratified k-fold algorithm and classifier itself use fixed initialisation values. These initialisation values differ across each of the three runs. This ensures that each parameter set utilises the same training and testing folds per run but that these folds differ across runs. Stratified 5-fold cross

validation was used to ensure that the ratio of defaulters to non-defaulters remained consistent in each fold. 5-fold was used instead of 10-fold to ensure that there was a reasonable sample of defaulters per fold.

To test each of the three scenarios, three runs of stratified 5-fold cross validation is executed. Each run uses a unique random initialisation value for the data balancer, stratified k-fold algorithm and classifier. This allows each run to use a different initialisation parameter and ensures that the same folds are used for each classifier on each scenario. The initialisation values are saved in the experiment results to facilitate reproducibility. We used BARR to compare the results between classifiers and scenarios.

4.1.2 Comparison of classification techniques

This experiment focuses on the comparing each optimised classifier in detail. The same testing procedure is used as outlined in section 4.1.1 except in this experiment we record several metrics to compare the classification techniques. In addition a receiver operating characteristic (ROC) curve is also plotted as an additional means of comparing the classification techniques and to indicate what true positive rate can be achieved for an acceptable false positive rate. To determine how the results generalise against the other similar classification datasets, a final scatter plot is presented which compares difference of true positive and false positive from the median of each for each classifier on each dataset.

4.1.3 Comparison of data balancing algorithms

The second experiment compares each data balancing technique on each classifier. The optimal parameters for each classifier are used for every data balancer. As in experiment 1, the highest BCR is used to determine the optimal parameters. Each classifier is run using every data balance technique and the metrics outlined in section **??** is recorded. To ensure repeatable results, 10 runs of 5-fold cross validation is executed and the results averaged. A set of unique randomly selected initialisation keys are selected for each run. These are used so that k-fold cross validation, data balancing and classification algorithms produce deterministic output. The initialisation values are saved in the results to ensure they can be reproduced.

A bar chart is plotted with each classifier against the MCC for the treatment default datasets. As in section **??**, a scatter plot is presented which compares difference of true positive and false positive from the median of each for each classifier on each data balancing algorithm on each dataset.

4.1.4 Comparison of feature selection algorithms

The third experiment investigates a number of different feature selection approaches in order to improve overall classification through the removal of features that each algorithm deems irrelevant to the classification. A number of feature selection strategies were selected for this experiment: Analysis of variance (ANOVA) F-value, logistic regression, linear SVM, Bernoulli naive Bayes, decision tree, random forest. For the ANOVA F-test, a p-value of greater 0.05 was selected to

4.1.5 An analysis of time to default in the Lima Peru dataset

4.2 Evaluation

4.2.1 Data sets

5. IMPLEMENTATION

6. RESULTS

1. Tables with true positive, true negative, false positive and false negative rate for each classifier on each dataset

2. ROC curves which can be used to determine true positive for an acceptable amount of false positives for each classification technique on each dataset

3. Graphs which outline difference in accuracy of each data balancing technique for each dataset

4. Discussion on best classification technique

5. Discussion on data balancing results

6. Outline similarities and differences in results between the two TB datasets and between TB and financial datasets and determine if they are similar enough that results for the German and Australian credit scoring datasets could be used for the two TB datasets.

7. CONCLUSIONS AND FUTURE WORK

1. Likely something along the lines of utilising more temporal based data so that classification is not just done at registration but also at each check-up for example. Future work may also be the testing of more classification techniques as well as datasets from other parts of the world.

8. REFERENCES

[1] Two modifications of cnn. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-6(11):769–772, Nov 1976.

[2] G. E. Batista, A. L. Bazzan, and M. C. Monard. Balancing training data for automated annotation of keywords: a case study. In *WOB*, pages 10–18, 2003.

[3] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard. A study of the behavior of several methods for balancing machine learning training data. *SIGKDD Explor. Newsl.*, 6(1):20–29, June 2004.

[4] J. Bergstra, N. Casagrande, D. Erhan, D. Eck, and B. Kégl. Aggregate features and adaboost for music classification. *Machine Learning*, 65(2):473–484, 2006.

[5] A.-L. Boulesteix, S. Janitza, J. Kruppa, and I. R. KÃűnig. Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(6):493–507, 2012.

[6] M. Chan-Yeung, K. Noertjojo, C. Leung, S. Chan, and C. Tam. Prevalence and predictors of default from tuberculosis treatment in hong kong. *Hong Kong Medical Journal*, 9(4):263–270, 2003.

Table 4: Data set summary

Data set	Entries	Number of numerical features	Number of categorical features	Data balance ratio (Cured:Defaul
Lima Peru TB [22]	1186	0	15	8.65:1
German Credit Scoring	1000	7	13	2.33:1
Australian Credit Scoring	690	6	8	1.25:1

[7] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: Synthetic minority over-sampling technique. *J. Artif. Int. Res.*, 16(1):321–357, June 2002.

[8] N. V. Chawla, N. Japkowicz, and A. Kotcz. Editorial: Special issue on learning from imbalanced data sets. *SIGKDD Explor. Newsl.*, 6(1):1–6, June 2004.

[9] I. Cherkaoui, R. Sabouni, I. Ghali, D. Kizub, A. C. Billioux, K. Bennani, J. E. Bourkadi, A. Benmamoun, O. Lahlou, R. E. Aouad, and K. E. Dooley. Treatment default amongst patients with tuberculosis in urban morocco: Predicting and explaining default and post-default sputum smear and drug susceptibility results. *PLoS ONE*, 9(4):1–9, April 2014.

[10] P. Danenas and G. Garsva. Selection of support vector machines based classifiers for credit risk domain. *Expert Systems with Applications*, 42(6):3194–3204, 2015.

[11] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119 – 139, 1997.

[12] K. Gowda and G. Krishna. The condensed nearest neighbor rule using the concept of mutual nearest neighborhood (corresp.). *IEEE Transactions on Information Theory*, 25(4):488–490, Jul 1979.

[13] H. He, Y. Bai, E. A. Garcia, and S. Li. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pages 1322–1328, June 2008.

[14] K. Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251–257, 1991.

[15] C.-W. Hsu, C.-C. Chang, C.-J. Lin, et al. A practical guide to support vector classification. *Department of Computer Science, National Taiwan University*.

[16] G. B. Huang, H. Zhou, X. Ding, and R. Zhang. Extreme learning machine for regression and multiclass classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(2):513–529, April 2012.

[17] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew. Extreme learning machine: Theory and applications. *Neurocomputing*, 70(1âĂŞ3):489 – 501, 2006. Neural NetworksSelected Papers from the 7th Brazilian Symposium on Neural Networks (SBRN '04)7th Brazilian Symposium on Neural Networks.

[18] U. M. Jha, S. Satyanarayana, P. K. Dewan, S. Chadha, F. Wares, S. Sahu, D. Gupta, and L. S. Chauhan. Risk factors for treatment default among re-treatment tuberculosis patients in india, 2006. *PLoS ONE*, 5(1):1–7, January 2010.

[19] S. X. Jittimanee, E. A. Madigan, S. Jittimanee, and C. Nontasood. Treatment default among urban tuberculosis patients, thailand. *International Journal of Nursing Practice*, 13(6):354–362, 2007.

[20] J. M. Keller, M. R. Gray, and J. A. Givens. A fuzzy k-nearest neighbor algorithm. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-15(4):580–585, July 1985.

[21] M. Kubat and S. Matwin. Addressing the curse of imbalanced training sets: One-sided selection. In *In Proceedings of the Fourteenth International Conference on Machine Learning*, pages 179–186. Morgan Kaufmann, 1997.

[22] B. Lackey, C. Seas, P. Van der Stuyft, and L. Otero. Patient characteristics associated with tuberculosis treatment default: A cohort study in a high-incidence area of lima, peru. *PLoS ONE*, 10(6):1–11, 2015.

[23] J. Laurikkala. Improving identification of difficult small classes by balancing class distribution. In *Proceedings of the 8th Conference on AI in Medicine in Europe: Artificial Intelligence Medicine*, AIME '01, pages 63–66, London, UK, UK, 2001. Springer-Verlag.

[24] D. D. Lewis. *Naive (Bayes) at forty: The independence assumption in information retrieval*, pages 4–15. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998.

[25] S.-T. Luo, B.-W. Cheng, and C.-H. Hsieh. Prediction model building with clustering-launched classification and support vector machines in credit scoring. *Expert Systems with Applications*, 36(4):7562–7566, 2009.

[26] C. Mood. Logistic regression: Why we cannot do what we think we can do, and what we can do about it. *European Sociological Review*, 26(1):67–82, 2010.

[27] B. Muture, M. N. Keraka, P. K. Kimuu, E. W. Kabiru, V. O. Ombeka, and F. Oguya. Factors associated with default from treatment among tuberculosis patients in nairobi province, kenya: A case control study. *BMC Public Health*, 11(1):696–105, September 2011.

[28] I. Rish. An empirical study of the naive bayes classifier. In *IJCAI 2001 workshop on empirical methods in artificial intelligence*, volume 3, pages 41–46. IBM New York, 2001.

[29] R. Rojas. Adaboost and the super bowl of classifiers a tutorial introduction to adaptive boosting, Computer Science Department, Freie Universität, Berlin. 2009.

[30] S. Sandro. Understanding logistic regression analysis. *biochem*, 24(1):12–18, 2014.

[31] E. B. Shargie and B. Lindtjorn. Determinants of treatment adherence among smear-positive pulmonary tuberculosis patients in southern ethiopia. *PLoS Med*, 4(2):1–8, February 2007.

[32] M. R. Smith, T. Martinez, and C. Giraud-Carrier. An instance level analysis of data complexity. *Mach. Learn.*, 95(2):225–256, May 2014.

[33] V. Svetnik, A. Liaw, C. Tong, J. C. Culberson, R. P.

Sheridan, and B. P. Feuston.

[34] S.-C. Wang. *Interdisciplinary Computing in Java Programming*, chapter Artificial Neural Network, pages 81–100. Springer US, Boston, MA, 2003.

[35] World Health Organisation. Global tuberculosis report 2015.

[36] I. Zhang and I. Mani. knn approach to unbalanced data distributions: a case study involving information extraction. In *Proceedings of workshop on learning from imbalanced datasets*, 2003.