University of Sydney

COMP3308 Assignment 2 Report

Classification Study – Pima Indian Diabetes

Date Due: 14 May 2021

Contributors	**Student ID (SID)**
Student ID (SID)	470539664
Student ID (SID)	490155963

Table of Contents

Section 1 – Aim

Section 2 – Dataset Analysis

Section 3 – Classification Results and Evaluation

Section 4 – Conclusion

Section 5 – Reflection

Section 1 – Aim

Section 1.1 – Aim of Study

The aim of the Pima Indian Diabetes study is to implement, compare and evaluate the K-Nearest Neighbour and Naïve Bayes Algorithms by using the cross-validation method on a provided dataset. Specifically, the study implements the previously stated algorithms, uses cross-validation for model-prediction, analyses the impact of using CFS as a feature selection method, and compares the algorithms against different existing algorithms on Waikato Environment for Knowledge Analysis (Weka), a tried and tested open-source machine learning software (University of Waikato, 1993).

Section 1.2 – High-level Executive Summary

The known origins of the Pima India Diabetes study dates back to research conducted on 9 May 1990 by Vincent Sigillito at John Hopkins University which was later publicly released by the National Institute of Diabetes and Digestive and Kidney Diseases. The aim of this study was to identify factors which established the Pima Indians of Arizona, in the United States of America (USA), as the community with the highest prevalence of diabetes in the world. It should be noted that, since Type 1 Diabetes rates was on par with other global communities, Type 2 Diabetes remains the focus of the previous and current studies (Knowler, Bennett, Hamman, Miller, 1978). Following the conclusion of this study, recent known modifications are limited to the cleansing and validation of the dataset for further analytical use.

Given the cleaned dataset, a new study and the software provided by Weka, the dataset will be pre-processed with the goal of normalising values for the dataset to between the range of $[0, 1]$. Following the pre-processing, the resulting CSV file underwent a feature-selection procedure known as CFS that ranks attributes according to an evaluation function (based on correlation) such that only material attributes are identified and used for the predictive model (Wosiak and Zakrewska, 2018).

Using the re-defined dataset with material attributes, the model is trained using a training dataset and K-nearest Neighbour followed by Naïve Bayes algorithm. A separate test dataset is utilised in conjunction with ten-fold stratified cross-validation to determine the accuracy of the predictive model that is generated. After analysing the results of the two predictive models, pre-existing Weka classifiers are also run with the same training and test data, before being compared and analysed.

Section 1.3 – Research Significance

In addition to defining a clear aim for the Pima Indian Diabetes study, several core objectives were identified by the team as being personally important. Namely, these reasons are to confirm a foundational understanding of statistical algorithms in a practical environment, interweaving structurally sound theoretical techniques into practical applications, peer-review and synergy through teamwork, and efficient verbal and visual communication of research findings.

Section 2 – Dataset Analysis

Section 2.1 – Dataset Description

The re-defined version of the dataset used for the Pima Indians Diabetes study contains restrictions and alterations to ensure data consistency. These alterations and restrictions relate to the use of data relating only to females with an age of at least 21 and of Pima Indian heritage, alterations of classes to nominal values, and the replacement of missing values with averages.

With respect to meta-data associated with the provided dataset, there are 768 total instances each of which has a class and eight attributes. The attributes are used by the predictive model to generate a class denoted $C = \{yes, no\}$ where a 'yes'-instance indicates a high likelihood of Type 2 Diabetes and a 'no'-instance indicates low likelihood of Type 2 Diabetes.

A list of nominal attributes available in the dataset (prior to CFS) include:

Item Number	Attribute	Measurement	Description
1	Time_pregnant	-	Number of times pregnant.
2	Glucose_concentration	-	Plasma glucose concentration in a 2-hour time-period (oral glucose tolerance test).
3	Blood_pressure	mm Hg	Diastolic blood pressure.
4	Skin_thickness	mm	Thickness of skin fold in tricep area.
5	Insulin	mu $(\frac{U}{ml})$	2-hour serum insulin.
6	Body_mass	$\frac{kg}{m^2}$	Body mass index.
7	Pedigree_function		Diabetes pedigree function.
8	Age	years	Age in years.

The final attribute that is given is non-nominal which is the class attribute that can either be 'yes' or 'no' indicating whether the individual tested positive for diabetes. In total, there are 500 'no' instances and 268 'yes' instances summing to the total of 768 instances.

Section 2.2 – Dataset Pre-processing

Data pre-processing takes place at the initial phase where, given as input a 'pima-indians-diabetes.data' file, Weka's normalisation filter was used to normalise values for each nominal attribute (as listed in the table above). The resulting file was saved as a CSV file and used for future operations. This step setup the foundations for which CFS could be conducted and material attributes chosen by discarding irrelevant or unrelated attributes.

Section 2.3 – Correlation-based Feature Selection (CFS) Analysis

Feature selection is the manual (or automatic) process of filtering features such that only significant features are used for training and testing of the prediction model, and subsequent output. Applying CFS to the original features will reduce the dataset dimensionality, mitigate the chance of overfitting, and remove irrelevant or redundant features. Additional characteristics of CFS also extend to include a reduction in overall time complexity (Blessie and Karthikeyan, 2012).

The CFS was conducted using a three-step process in conjunction with Weka's built-in best-first algorithm. This process included pre-processing of the dataset and generation of a feature class of correlation matrices, using a Breadth-first Search (BFS) to define subset space and determine merit value for all feature classes, and finally traversing the dataset to retrieve the collection with the largest merit value.

At the conclusion of the CFS process, the five selected attributes were glucose, insulin, BMI, pedigree and age. Given the provided attributes/features, it made intuitive sense that these attributes were correlated and provided significant contribution towards the outcome and were hence selected. In order to analyse the impact of the CFS, the results were also recorded in the two tables and chart below (rounded to 2 decimal places).

Section 3.1 - Analysis of Feature Selection (Accuracy Improvement 2 d.p.)

	ZeroR	1R	1NN	5NN	NB	DT
No feature selection	64.96%	69.90%	65.45%	74.42%	74.80%	72.84%
CFS	64.96%	69.90%	66.54%	74.42%	75.02%	72.95%
Improvement	0	0	0.08%	0	0.22%	0.11%

	MLP	SVM	RF	My1NN	My5NN	MyNB
No feature selection	75.15%	75.64%	75.02%	66.42%	74.75%	74.90%
CFS	75.59%	76.00%	75.23%	66.83%	74.84%	75.25%
Improvement	0.44%	0.36%	0.22%	0.42%	0.09%	0.34%

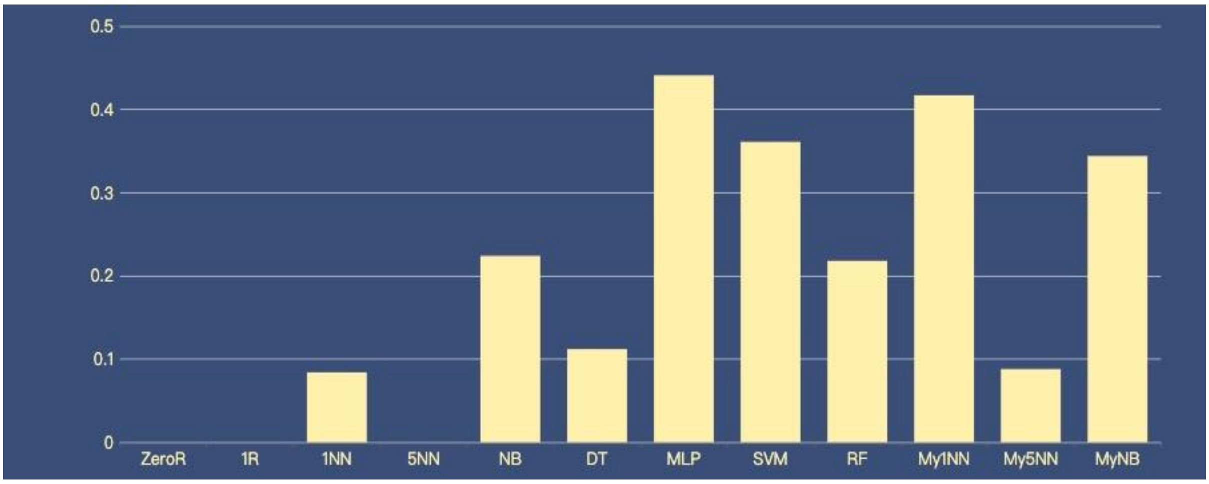


Figure 1: Improvement in Predictive Model Accuracy

The tables present an analytical foundation for understanding the impact of the CFS on ten different predictive models (ten different algorithms) and the diagram charts a visual representation of the recorded figures. Whilst feature selection improved a majority of the predictive models, the maximum increase in accuracy was capped by 0.50%.

Noticeably, the ZeroR and 1R algorithms contributed towards two of the three predictive models that failed to improve after CFS. Findings, analysis and conclusion determined that, since the ZeroR and 1R algorithms began with low dimensionality, the Curse of Dimensionality is minimal or zero as few or no features could be removed. On the other hand, the multi-layer perceptron (MLP) proved to be most impacted by the CFS which is unsurprising given its high dimensionality.

Although the algorithms performed differently, it is important to note the different roles played by each predictive model. Whilst the ZeroR and 1R are simple classifiers, they provide a strong benchmark for which to compare the accuracy of other predictive models. MLP on the other hand, is aimed specifically at using high dimensionality to convert non-linear separable pattern problems into linearly separable problems.

A Confusion Matrix, as depicted below, is also used to determine the level of accuracy improvement achieved by the CFS. Two sections covered specific factors that contribute towards the determination of this result are false positive (FP) and false negatives (FN).

Classifier	FP	FP(CFS)	Improvement	FN	FN(CFS)	Improvement
ZeroR	0	0	0	230	230	0
1R	94	94	0	160	160	0
1NN	128	126	1.56%	123	117	-4.88%
5NN	98	95	3.06%	121	126	4.13%
NB	102	84	17.65%	118	122	3.39%
DT	126	114	9.52%	104	106	1.92%
MLP	101	95	5.94%	94	91	3.19%
SVM	52	50	3.85%	117	115	1.71%
RF	98	84	14.29%	105	104	0.95%
My1NN	122	119	2.46%	118	113	-4.24%
My5NN	101	93	7.92%	117	124	5.98%
MyNB	108	101	6.48%	115	125	8.70%

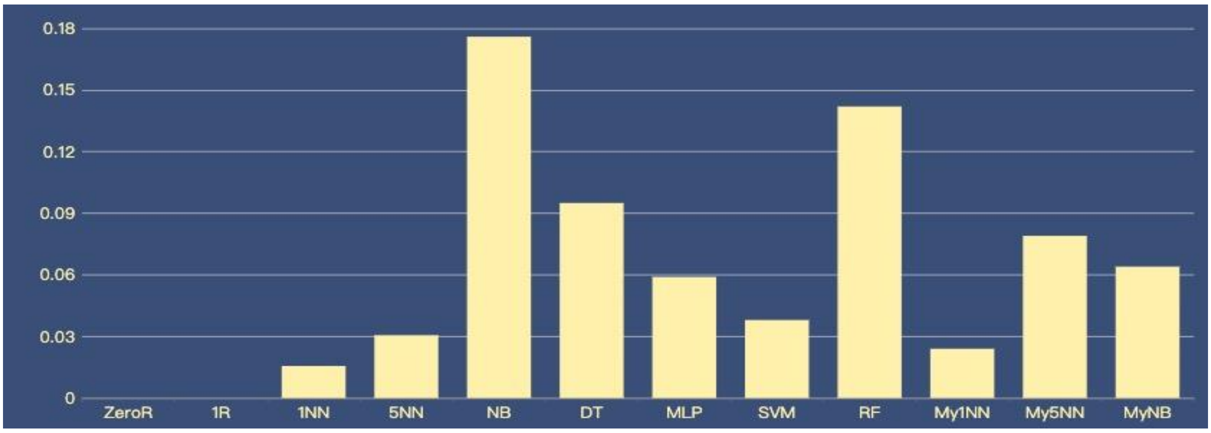


Figure 2: Improvement in FP

Section 3.2 – Comparison Analysis of Weka Classifiers

Given the charted output of the predictive models, it is important to compare the Weka Classifiers and understand the effectiveness of different algorithms. Each classifier algorithm must be analysed individually in order to identify specific differences and discrepancies between theoretical outcomes and the practical result.

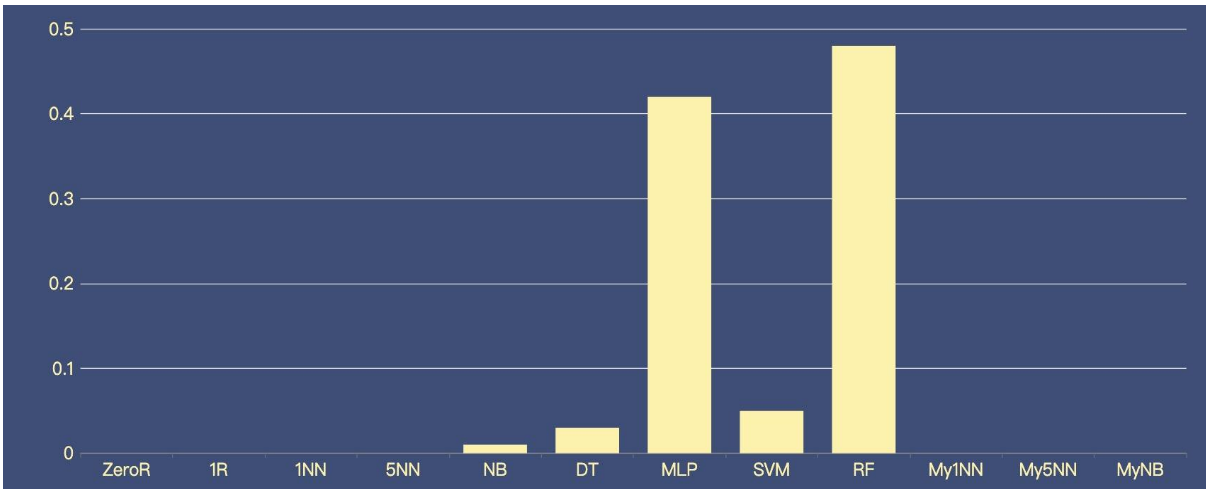
Zero R and 1R are known as simple algorithmic classifiers that use fewer features in order to formulate the predictive model and thereby determine output. ZeroR only uses the target feature to build the predictive model and therefore was not improved through the CFS method. 1R only uses a single feature to build a single-layer decision tree (feature selected based on outcome from training data). Although these algorithms are simple, they are used to provide benchmarks for performance and are therefore still widely used.

Two widely used algorithm classifiers, **K-Nearest Neighbour (K-NN) and Naïve Bayes**, are simple but effective and unlikely to overfit the data. In addition to providing, on average, strong results for accuracy, these two algorithms also have a good time complexity which makes them beneficial for quicker analysis. As noticeable in the diagrams above, they do not achieve the level of accuracy as RF and MLP.

A set of almost-equally performant classifiers are the **MLP, SVM, RF and DT**. Whilst decision trees are able to predict well, it can be difficult to construct and rely on a heuristic which must be calculated. This heuristic is based on information entropy which must be taken into account. Therefore, for this particular experiment the MLP, SVM and RF Weka built-in classifiers proved to be the most effective with a trade-off being their longer time complexity.

Tim Analysis (Rounded to 2 d.p.)

Classifier	Time(seconds)
ZeroR	**0**
1R	**0**
1NN	**0**
5NN	**0**
NB	**0.01**
DT	**0.03**
MLP	**0.42**
SVM	**0.05**
RF	**0.48**



Section 4 - Conclusion

Section 4.1 - Classifier Selection

In the perspective of accuracy, we concluded that both SVM and MLP were the best algorithmic classifiers in general, whilst the ZeroR performed the worst (as expected). However, as an aside, the SVM has the potential to inflate the false-negative rates. Implications of a higher false-negative rate has the impact of fasely claiming that a Pima Indian does not has Type 2 Diabetes and, as Type 2 Diabetes has physical implications, can lead to serious consequences. As a result, the MLP model was decided upon as a better choice.

Although the MLP uses an algorithm with a significantly longer time complexity, it was deemed acceptable based on the results demonstrating the classifier's ability to achieve strong results. An interesting fact uncovered by the project was the trade-off between the time complexity for training the predictive model, and the accuracy of the outcome.

Given a different set of circumstances, different algorithmic classifiers may be used to train and test the predictive models. For example, for time-sensitive analysis, Naïve Bayes algorithm would be best suited due to its ability to achieve relatively strong performance with regards to accuracy, time complexity and least number of false-negatives. Thus, it is important to analyse the situation for which the classifier must be used.

Section 4.2 - Classifier Optimisation and Future Work

Finding 1

When comparing and contrasting Weka's built-in classifiers and our custom classifiers, there was a noticeable difference in performance with regards to accuracy. This was especially noticeable in the 1NN and 5NN algorithmic classifiers. Increasing the number of neighbours in the K-nearest neighbours' algorithm, given the same dataset, improved the classifiers performance. In future studies, identifying a suitable K value is believed to bring upon better classifier performance.

Finding 2

Another finding was that applying CFS can improve accuracy performance. By simply using Weka's in-built CFS, a BFS algorithm, and the summation of merit values for this project, the accuracy of certain classifiers could be improved. In anticipation of future projects, studies of the benefits of Lasso or Tree-based select-from-model methods may prove beneficial for further increase in accuracy of classifiers.

Finding 3

The third and final important finding was that, if all the attributes in the dataset were independent of each other, the performance of these machine learning based classifier could be improved. However, given the severity of correlation between the given attributes, this is not possible. In the future, there may be benefits in seeking and collecting a dataset consisting of purely independent attributes to determine the effect of that on the accuracy of the algorithmic classifiers.

Section 5 – Reflection

Section 5.1 – Personal Reflection

As a team, it was important to reflect upon the process from initiation to conclusion in order to emphasise and acknowledge development areas and key success areas.

Although the project aim was achieved, the process of teamwork and communication played a significant role in influencing the outcome. Since the release of the assignment, both parties agreed upon the standards for which the assignment would be executed and planned this in accordance with our combined schedules. One example of this was one of the many Zoom meetings held to discuss findings, learnings and upcoming time schedules to ensure that no person fell behind in studies.

Another interesting learning was the varying perspectives of each team member and how these transformed throughout the project. Initially, both parties had some similarities but a vast degree of disagreement in approaches that should be taken, and algorithms that would perform best. However, as the project unrolled, we worked together as a team to resolve disagreements, and allowed our contribution towards the assignment to provide quantitative proof of algorithms.

Section 5.2 – Future Implementations (Lessons Learnt)

These key learnings present potential opportunities for improvement in future research projects. When collaborating, we agreed that it was important to firstly determine collaboration tools and processes before attempting to define timeframes. This would ensure that proper procedures are conducted and fewer conflicts would need to be resolved. Moreover, it would reduce the churn rate required to produce the report as formatting changes would be saved appropriately.

Another potential opportunity presented itself – the opportunity to validate each team member's understanding in a topic by working together. Despite initial disagreements, by the completion of the project, both team members were confident in their analysis and aligned in vision. Bringing forward this perspective, it is important to work hard to solve the problem at hand and instil confidence in the other team member.

Bibliography

Blessie E. and Karthikeyan E. (2012). Sigmis: A Feature Selection Algorithm Using a Correlation Based Method. Algorithms & Computational Technology, 6(3), 385-386. Doi: 10.1260/1748-3018.6.3

Knowler WC., Bennett PH., Hamman RF., Miller M. (1978). Diabetes incidence and prevalence in Pima Indians: a 19-fold greater incidence than in Rochester, Minnesota. American Journal of Epidemiology, 108:497-505. doi: 10.1093/oxfordjournals.aje.a112648

Pavan Vadapalli (2020). Random Forest Vs Decision Tree: Difference Between Random Forest and Decision Tree. Retrieved from https://www.upgrad.com/blog/random-forest-vs-decision-tree/

Raheel Shaikh (2018). Feature Selection Techniques in Machine Learning with Python. Retrieved from https://towardsdatascience.com/feature-selection-techniques-in-machine-learning-with-python-f24e7da3f36e

University of Waikato. (1993). Data Mining with Open Source Machine Learning Software in Java. [online]. Retrieved from https://www.cs.waikato.ac.nz/ml/weka

Wosiak A. and Zakrewska D. (2018). Integrating Correlation-Based Feature Selection and Clustering for Improved Cardiovascular Disease Diagnosis. doi: 10.1155/2018/2520706