EECS 6690

Tenyu Zhou (tz2338), Yuxin Ding (yd2406)

Final Project Report

May 9, 2018

Data Mining on Pima Indian Diabetes Dataset

1 Introduction

Over the centuries, diabetes diagnosis has always been a major problem around the world. Doctors need years of training to be able to tell patients' diabetes status. The reason for a person getting diabetes could be a lack of insulin in blood. Insulin is the natural hormone which is harnessed by the body to decompose and absorb starch and sugar. Without enough insulin, not only will resulting a lack of energy for the body, but will also resulting in too much sugar in blood hence leading to diabetes. There are mainly two reasons for insulin deficiency. First is type I diabetes mellitus caused by pancreas not producing insulin at all. Second is type II diabetes. The reason for type II diabetes is that body cannot use the insulin produced by the pancreas, as a result, glucose cannot enter the cell. [2] In our paper, we focused on the Pima Indian Diabetes Dataset(PIDD) and tried several machine learning classification techniques. The Pima Indian Diabetes Dataset(PIDD) was originally from the National Institute of Diabetes and Digestive and Kidney Diseases and we downloaded the dataset from the UCI machine learning repository. [1]

We began our project by digging previous research based on the PIDD dataset and trying the approaches they used. Ilango et al. achieved an prediction accuracy of 98.94% by applying

the technique of F-score Feature Selection, K-Means Clustering and SVM. [3] Hussan used K-Means algorithm achieving 97% accuracy by replacing the zero value in the dataset with the mean of its' k nearest neighbors.[4] The algorithm is interesting mainly because it does not merely delete the samples with missing values, rather, it fills those values with what is expected. Jayaram et al. proposed a four stages algorithm on classifying the dataset. Firstly, samples with impossible zero value in several columns were claimed missing value samples and deleted. Secondly, K-Means clustering was used to determine wrongly classified samples and deleted. Thirdly, continuous data was converted to categorical data based on the suggestion of expired doctor. Fourthly, classification was done using C4.5 Decision Tree classifier. The four stages algorithm achieved an accuracy of 93.33%. [5] Breault et al. tried using rough sets for the classification and achieved an accuracy of 82.6%. [6] Han et al. used RapidMiner for the diabetes data analysis and prediction and achieved an accuracy of 72% by using decision tree. [5] When applying ID3 algorithm, an accuracy of 80% was achieved. Pujari et al. [8] tried to combine three different classifiers SVM, discriminant analysis and Bayesian network to get a better result. The model of the above combination achieved an accuracy of 76.03%. Pradhan et al. suggested using neural networks and fuzzy KNN algorithm to classify the data and they also delete samples with missing value like several other papers.

2 Original Dataset

This dataset we chose was originally from the National Institute of Diabetes and Digestive and Kidney Diseases and we downloaded the dataset from the UCI machine learning repository. [1]. The dataset contains several body health indicators and a label of whether certain patients had

diabetes. And one thing need to be considered is that all patients are females, also, they all have Pima Indian heritage for 21 years. The dataset contains 768 samples in total. For each sample, it contains 8 health attributes and 1 class variable. The class variable indicates whether the patient has diabetes: zero being the patient does not have diabetes and one being the patients have diabetes.

The health attributes include

- How many times the woman get pregnant (preg)
- Concentration of Plasma glucose during oral glucose tolerance test (plas)
- Pressure of diastolic blood in mm Hg (pres)
- The thickness of triceps skin fold in mm (skin)
- Serum insulin density in mu U/ml (insu)
- Body mass index to be measured by weight in kg/m^2 (mass)
- Diabetes pedigree function (pedi)
- The age of testers (age).

Many researchers have claimed this dataset as an incomplete dataset for missing values while the original contributor claimed the dataset is accurate and complete. But several columns like blood pressure and insulin are not likely to be zero. This fact introduces several interesting points for this dataset which deserves close study. Since whether to have certain preprocessing is highly debatable.

3 Original Papers

This section will introduce the two papers we tried to reproduce in detail, including the methods, algorithms and results produced by the authors.

3.1 Diabetes data analysis and prediction model discovery using RapidMiner

The first paper we experimented with is Diabetes data analysis and prediction model discovery using RapidMiner by Jianchao Han, et al. [7] The paper started by introducing the dataset and a software for data mining called RapidMiner. The experiments in the paper were all conducted on RapidMiner. Then, the paper discussed the three data preprocessing techniques applied. Third, the paper explored hidden relationships between variables using BasicRuleLearner, a tool in RapidMiner. Finally, the paper experimented the the tree based classification algorithm ID3 and compared its results with decision tree's results. The paper achieved a 80% accuracy.

The paper's approach is using preprocessing techniques and ID3 algorithm. The preprocessing techniques includes three steps in total.

1. Outlier removal and feature selection was applied on the dataset. Rows with missing values, meaning zero values variables, in plasma glucose, diastolic blood pressure or body mass index are removed.

2. The authors applied numerical data discretization on pregnant times, plasma glucose, diastolic blood pressure, body mass index, diabetes pedigree function and age. Pregnant

times, plasma glucose, diastolic blood pressure and body mass index were discretized manually by using medical diagnostic information. Diabetes pedigree function and age were automatically discretized by functions in Rapid Miner.

3. Data normalization was applied on plasma glucose and body mass index.

After data preprocessing, a decision tree and an ID3 tree was grown using the preprocessed data. The results were compared. The decision tree achieved a 72% accuracy and the ID3 algorithm achieved a 80% accuracy.

3.2 Classification for Diabetic Patients using Cascaded K-Means and Decision Tree C4.5

The second paper we chose to reproduce is Rule based Classification for Diabetic Patients using Cascaded K-Means and Decision Tree C4.5 written by Asha Gowda Karegowda et al. [7] The paper begins with a brief introduction of the diabetes mellitus itself and related work on diabetic data set classification. Later, Decision Tree C4.5 and K-means clustering algorithm are also briefly introduced. Decision tree is a simple tree classification structure that the non-terminal nodes represents the classification rule and C4.5 is an enhanced decision tree induction algorithm of ID3. K-means clustering is a classic classification algorithm which will iteratively update center and cluster assignment.

Then the paper proposed its own classification model. The model consists of two stages, data preprocessing and classification. For the preprocessing stage, the dataset contains 5 patients had a glucose of 0, 11 patients' body mass index are 0, 28 patients had a diastolic blood pressure of 0, 192 patients had 0 skin fold thickness readings, and 140 patients had serum insulin levels of

0. The data with those zero values were deleted from the dataset and the dataset size shrink from 768 to 392 with 130 tested positive and 262 tested negative.

For the classification stage, simple K-means clustering is firstly applied to the dataset and the sample which has a different label than the clustering result is claimed to be wrongly classified sample and those samples are deleted. Secondly, the continuous data is converted to categorical data following the suggestion of professional doctors in diabetes area. Thirdly, the categorical data is fed to Decision Tree C4.5 using 60-40 partition ratio. The classification accuracy is 93.33%.

4 Result Reproduction

In this section, the reproduced results are shown and the explicit steps we conducted to reproduce the results are explained.

4.1 Diabetes data analysis and prediction model discovery using RapidMiner

For the first paper, we followed steps stated in the paper and produced in a 82.6% accuracy, which is similar to the 80% result in the original paper.

The exact preprocessing steps in the paper were followed. After deleting rows with missing values in plasma glucose, diastolic blood pressure or body mass, 724 data were remained. Then, data discretization and normalization was conducted strictly as the paper states.

For the ID3 algorithm, the paper did not explicitly explain how the was ID3 algorithm was trained. So, we choose the first 500 data as the training data and the remaining 224 as testing data. The training testing split ratio is approximately $70:30$. The accuracy is 82.6%.

The confusion matrix of our reproduction is shown in table 1.

Table 1: Paper 1 Result Reproduction Confusion Matrix

		Actual Labels	
		0	1
Prediction	0	404	29
	1	30	194

4.2 Classification for Diabetic Patients using Cascaded K-Means and Decision Tree C4.5

For the result reproduction of the second paper, two stages of processing were done to the original dataset. The first stage is data preprocessing and second stage is classification. Approach was used exactly the same as the original paper. For the first stage preprocessing, all the data which contained zero in columns glucose, body mass index, diastolic blood pressure, skin thickness reading and serum insulin level were deleted and we got 392 data samples. For the second stage classification, k-means clustering was firstly applied in order to determine the wrongly classified labels and the samples with wrongly classified labels were deleted. Then according to the decision rule in the original paper, the continuous data is converted to categorical data. At last, Decision Tree C4.5 was used for classification and the training-testing ratio was 60-40. The classification accuracy of the reproduction is 94.17%.

The confusion matrix of our reproduction is shown in table 2.

Table 2: Paper 2 Result Reproduction Confusion Matrix

		Actual Labels	
		0	1
Prediction	0	82	7
	1	0	31

5 New Approaches

The new approaches section will introduce the classifiers and preprocessing techniques we experimented with and show the results of our new approaches. We experimented with 5 classifiers and 4 preprocessing techniques in total. All the classifiers were paired with all the preprocessing techniques to find the best classification method.

5.1 Classifiers

Generally, five new classifiers have been tried for this dataset, including naïve-bayes, k nearest neighbors, neural network, boosting and semi-supervised learning.

5.1.1 Naïve-Bayes

Naïve-Bayes is the most commonly used classifier for classification when assuming the parameters are independent. It is suitable for out dataset and worthwhile to try it which can provide us a baseline for other methods. Five-cross validation were used for Naïve-Bayes and final best classification accuracy was 75.13%.

5.1.2 K Nearest Neighbors

K nearest neighbors is also a very commonly used classifier. Since the 8 attributes in the dataset does not have obvious correlations between each other, we would think K nearest neighbors could be a good classifier. The distance computed is the L2 distance. For the hyper-parameter k, we experimented from 1 to 50 and choose 22 since in produced the highest accuracy. The highest accuracy achieved was 75.92%.

5.1.3 Neural Network

Neural Network has been known as a way to approximate any mathematical distribution with a simple math structure. However, the structure of the neural network is hard to define. We experimented with neural network structures with 1 to 3 hidden layers and 1 to 20 hidden units in each layer. In result, the structure with 1 hidden layer and 7 hidden units fully connected perceptron produces the best results: 76.70% accuracy.

5.1.4 Boosting

Boosting is a method based on bagging and gradually learns the model from each sampling and has been a high performance classifier for many datasets. Commonly used boosting classifiers include adaptive boosting (adaboost) and extreme gradient boosting (xgboost). After experimentations, we found that extreme gradient boosting classification performs better on this dataset. Therefore, the extreme gradient boosting classification was used and find tunning of the parameters were also conducted. The accuracy of xgboost is 78.39%.

5.1.5 Semi-supervised Learning

Semi-supervised Learning is a model which has been rarely tried before on this dataset. The reason for that might be people often relate Semi-Supervised Learning with time-dependent model. However, the difference of the value in the same column can actually be treated as distance between different data point. And using this distance, we can construct a transition matrix for the Markov chain and use that matrix for the purpose of Semi-Supervised Learning. To have the best result of Semi-Supervised Learning, four different preprocessing methods were tried and two kernels (RBF and KNN) are tried. And for each kernel, hundreds of iterations were run in order to find the best parameter. Finally, we got the best setting for Semi-Supervised Learning and got a classification accuracy of 79.83%.

5.2 Preprocessing

For preprocessing, four preprocessing techniques were done to the original dataset, including no preprocessing, normalization, discretization and combining discretization and normalization.

Normalization is scaling the attributes from their original range to range 0 to 1. The attributes include all the 8 health condition attributes.

Discretization is changing the continuous values of the dataset to categorical based on the interval suggested by professionals. We chose different combinations of the preprocessing method for each model.

6 Results and Analysis

We have paired the five different new approaches for classification (naïve-bayes, k nearest neighbors, neural network, boosting and semi-supervised learning) with the four preprocessing methods (no preprocessing, normalization, discretization, discretization and normalization). The results are shown in Table 3. The results were computed by five fold cross validation.

Table 3: New Approaches Accuracy Results

	Naïve-Bayes	K Nearest Neighbors	Neural Network	Boosting	Semi-Supervised Learning
No Preprocessing	75.13%	74.62%	68.78%	78.26%	67.74%
Normalization	75.13%	75.92%	77.09%	78.39%	78.62%
Discretization	72.00%	67.97%	71.22%	74.87%	69.35%
Discretization and Normalization	72.00%	72.66%	76.70%	74.87%	79.83%

From Table 3, we can see that different classifiers pairs best with different preprocessing techniques. We think it is because different mathematical models are used in different classifiers and the mathematical models can define how they perform to different preprocessing techniques.

Naïve-Bayes pairs best with no preprocessing because it is based on the Bayesian models and the assumption of underlying distributions. Therefore, normalization is ineffective to Naïve-Bayes, as it only sales the models but does not change the model's probabilities. Also, discretization would worsen the results as it would lose some part of the model distribution.

K nearest neighbors pairs best with normalization because it computes the distance between the data and label the testing data as its neighbors' labels. The attributes has different ranges. For example, body mass index ranges from 8.2 to 67.1 and plasma glucose range from 44.0 to 199.0. These ranges could effect the distances computed. Therefore, normalizing the data

could help with the performance of k nearest neighbors. On the other hand, discretization would lose some distance information and, thus, resulting in worsening the results.

Neural network pairs best with discretization and normalization because discretization could help neural networks better approximate the underlying distribution and normalization might help with faster convergence.

Boosting pairs best with normalization because boosting learns from a set of decision trees. The enhanced performance of boosting with the preprocessing could be the result of the preprocessing can improve decision trees prediction accuracy.

Semi-supervised learning pairs best with discretization and normalization. Since semi-supervised learning is also based on the distance between data, normalization increasing the prediction accuracy is similar to k nearest neighbors.

Among all the results in Table 3, the highest accuracy is 79.83% achieved by using Semi-supervised Learning paired with discretization and normalization. This model beats all the state-of-art models which don't change the missing value or deleting samples with missing value. Semi-supervised learning can achieve such high accuracy is actually out of expectation. Our assumption for this high accuracy is that the transition matrix of semi-supervised learning can accidentally fit these data better than the other model. Since most models achieves an accuracy between 73% to 80% without doing anything prior to the missing value. The model with better classification accuracy might not be a better classifier in this application area in general, but only to this dataset. However, the Semi-supervised Learning did achieve a best performance in this dataset and surprisingly, we couldn't find literatures which tried this approach before. The reason might be most people think Semi-supervised Learning is implemented by Markov chain and Markov chain is used mostly in time-dependent datasets. However, from our paper, we can show

that Semi-supervised Learning can have a great performance in time-independent dataset. Thus, more research can be done related to Semi-supervised Learning in bio-statistical learning area.

7 Conclusion

In conclusion, the first paper reproduction accuracy is 82.6%, while the original accuracy is 80%. For the second paper we reproduced, the original accuracy is 93.33% while our accuracy is 94.17%. This accuracy discrepancy is merely resulted by one less wrongly classified label. Therefore, the classification accuracy is almost the same with the original paper since we were using exactly the same method for classification. However, the decision rules of the decision tree are quite similar for the first layer but different for the next few layers. We believe the reason for this is that the size of samples used for constructing the tree is not big enough. As a result, when choosing different subset of the samples, the structure of the tree might vary a little bit.

8 Future Works

For future work, we will primarily focus on two aspects. Firstly, we would like to try more machine learning models and also the combinations of different models. Because we believe different models perform differently given different circumstances and implementing multiple layers of models for prediction might give us a surprising result. Secondly, we would like to try several approaches addressing the missing value issue. Several papers have already tried approaches other than simply deleting those samples with missing values like using K-NN algorithm to fill in those value. However, K-NN algorithm might fail to provide an accurate

estimation for the missing value when samples are hardly associated without the missing value and it's hard to say how confidence we are about the value we filled in. As a result, we would like to try using some probabilistic models for the missing value filling task and see the result.

9 Acknowledgement

We would like to thank Professor Predrag Jelenkovic for his awesome lectures that help us to get familiar with topics and tools of statistical learning domain. We also owe our sincere gratitude to the teaching assistant Ludwig Zhao and Tingkai Liu for their guidance in R and letting the course running smoothly the whole semester.

References

[1] "UCI Machine Learning Repository." *Archive.ics.uci.edu*.

[2] Bingley, Polly. "Clinical Applications Of Diabetes Antibody Testing." The Journal of Clinical Endocrinology & Metabolism 95.1 (2010): 25–33.

[3] Ilango, B. Sarojini, and N. Ramaraj. "A Hybrid Prediction Model With F-Score Feature Selection For Type II Diabetes Databases." Proceedings of the 1st Amrita ACM-W Celebration on Women in Computing in India - A2CWiC '10 (2010).

[4] Hussan, B.M. "Prediction Of Medical Data Using K-Means Algorithm." Basrah Journal of Science 30.1 (2012): 46-56.

[5] Karegowda, Asha Gowda, M. A. Jayaram, and A. S. Manjunath. ”Cascading K-Means With Ensemble Learning: Enhanced Categorization Of Diabetic Data.” Journal of Intelligent Systems 21.3 (2012).

[6] Breault, J.L. ”Data Mining Diabetic Databases: Are Rough Sets A Useful Addition?.” Print.

[7] Han, Jianchao, Juan C. Rodriguez, and Mohsen Beheshti. ”Diabetes Data Analysis And Prediction Model Discovery Using Rapidminer.” 2008 Second International Conference on Future Generation Communication and Networking (2008).

[8] Pujari, P., and G.G Vishwavidyalaya. ”Ensemble Data Mining Model For Classification Of Pima Indian Diabetes Data Set.”