**Report of COMP3008 Assignment 2**

**Aim**

This study aims to let student have better understanding of AI’s classification algorithm ( K-Nearest Neighbour and Naïve Bayes), including how to implement algorithms to predict result from dataset and how to validate them through stratified cross validation method. What’s more, this assignment also tested students’ proficiency of using Weka, the machine learning software.

The reason why this assignment is important is that student will be inspired by comparing different machine learning algorithm and have more deep understanding of how AI works. Additionally, students can compare their algorithm to Weka algorithm so they will have better self-cognition and broaden their horizons on machine learning and artificial intelligent.

**Data**

The dataset we use is ‘Pima Indians Diabetes Database’ from National Institute of Diabetes and Digestive and Kidney Diseases originally edited on 9 May 1990, and for the usage of this assignment, this dataset was modified in March 2015.

This dataset originally has 8 attributes excluding class attributes:

1. Number of times pregnant

2. Plasma glucose concentration a 2 hours in an oral glucose tolerance test

3. Diastolic blood pressure (mm Hg)

4. Triceps skin fold thickness (mm)

5. 2-Hour serum insulin (mu U/ml)

6. Body mass index (weight in kg/(height in m)^2)

7. Diabetes pedigree function

8. Age (years)

Then we used Weka to normalised the value in the dataset by restricting them from 0 to 1. After the Correlation-based Feature Selection method (CFS) was applied to the dataset, five attributes which have greater influence on deciding the class were left:

2. Plasma glucose concentration a 2 hours in an oral glucose tolerance test

5. 2-Hour serum insulin (mu U/ml)

6. Body mass index (weight in kg/(height in m)^2)

7. Diabetes pedigree function

8. Age (years)

**CFS**

Results and Discussion

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| % | ZeroR | 1R | 1NN | 5NN | NB | DT | MLP | SVM | RF |
| No feature selection | 65.1042 | 70.8333 | 67.8385 | 74.4792 | 75.1302 | 71.7448 | 75.3906 | 76.3021 | 74.8698 |
| CFS | 65.1042 | 70.8333 | 69.0104 | 74.4792 | 76.3021 | 73.3073 | 75.7813 | 76.6927 | 75.9115 |

|  |  |  |  |
| --- | --- | --- | --- |
|  | My1NN | My5NN | MyNB |
| No feature selection | 82.3721 | 86.1731 | 96.8924 |
| CFS | .8628 | 86.1731 | 97.1324 |

Discussion of the result

The original data is normalized by weka first for presenting more accurate result. To deal with the normalised dataset for applying 10-fold cross validation for My1NN, My5NN and MyNB, the folds will be stratified first in Excel.Because 268 is not divisible by 10, there are 8 folds have 27 ‘Yes’ cases and 2 for 26.

A support column is used to random print numbers between 0 to 1 and then the function ‘INDEX(A$2:A$501,MATCH(LARGE($J$2:$J$501,ROW(A1)),$J$2:$J$501,))’ is used to put the value in column A depends on the large size of the random value in support column for ‘No’ cases. After that, apply the formula to every line below to shuffle the data for ‘No’ cases. This is also used to find others value which correspond to column A. After applying previous steps for ‘Yes’ cases, every 50 lines for ‘No’ cases and every 26-27 lines for ‘Yes’ are taken for every fold.

The 10-fold cross validation will be applied by python program to test the difference on accuracy of prediction using Naive Bayes and N-Nearest Neighbour algorithm. The output is put beside the actual output and using an ‘if’ function to calculate the difference in Excel. The accuracy is defined by calculating the division of correct prediction and total cases.

The accuracy using weka and own python algorithm are shown in previous chart. Our own work can reach an accuracy over 95 percentage, but only around 70 percentage for weka. The reason why it happened can be that our own algorithm performs a better functions on prediction and normalization. Own classifier targets at the data from 0 to 1 and does the comparison and prediction in that range better. That is also an important reason for the consequence.

ZeroR:

1R:

k-Nearest Neighbor (1NN):

k-Nearest Neighbor (5NN):

Naïve Bayes (NB):

Decision Tree (DT):

Multi-Layer Perceptron (MLP):

Support Vector Machine (SVM):

Random Forest (RF):

Effect of CFS

According to the results we got, we can simply conclude that CFS can help to improve the accuracy of some classification algorithm. For example we from the result we using Weka, the accuracy results of 1NN, NB, DT, MLP, SVM and RF all improved. And for the results we get from our own algorithm, My1NN and MyNB both had improvement on accuracy.

From the results we get so far, CFS is beneficial. It did not cause any decrease of accuracy compared to the No feature selection values meanwhile it increased the performance of some classifier to get better accurate results.

The reason why we get better accuracy from the dataset which has applied CFS is probably that for a realistic dataset, it could have several independent variables which may correlated to a dependent variable, then the coefficients in a regression model will tend to be unstable. however , the CFS process will recognise the few most significant variables (predictors) which help in predicting the outcome. There will be no more irrelevant or not important attributes to mislead the predicting algorithm.

Comparison between algorithms

From the result we get so far, it is apparent that the algorithms we implemented are better performed then Weka. Our algorithms all have accuracy greater than 80%, however the best-performed algorithm in Weka is SVM and it only get 76.3021% of accuracy for not applying CFS and 76.6927% for applying CFS. The algorithm has worst performance in Weka is ZeroR, only get 65.1024% of accuracy. What’s more, by comparing 1NN to 5NN, it is found that commonly 5NN will conduct a better result than 1NN. Possibly it is because in KNN algorithm, when the K is bigger, it will produce more neighbors to vote for the prediction, and the more of candidates voting, the greater of the chance for it to get the correct result (prediction).

**Conclusion**

Result conclusion

To be done

Future work suggestion

From this study, we get pretty good results of predicting class from the given dataset (accuracy above 80% for both algorithm implemented) compared to Weka. However, it could only be suitable for this particular case. The algorithm still has a huge room to learn and grow, for a real AI algorithm, it need to continue learning and be adaptive for all situation. For this time, the case is to identify diabete and our algorithm works fine on it, but there is still a possibility that for some of other realistic case that KNN or NB might not work that well. In order to refine the classifier, more algorithm for more situation is needed, so that people can get more specific prediction using machine learning algorithm.

**Reflection**

For this assignment, Vicii Sun (SID:470035270) and Jacky Li (SID:470011746) worked as a group. For the classifier implementing part, Vicii finished K-Nearest Neighbour algorithm and Jacky completed Naïve Bayes algorithm. Later on, Vicii and Jacky produced all the assignment deliverables together and they contributed equally. From the process of this study, their knowledge of machine learning and AI classification algorithm was broden, they both have practiced Weka and handled data well.

**Appendix**

The source code for algorithm and additional sources for doing this report are attached in the subfolder.zip.