Aim

This study is aim to evaluate the performance of commonly used classifier algorithms, K-Nearest Neighbour and Naïve Bayes algorithm using stratified cross-validation method. Evaluating the effectiveness and accuracy of these algorithm on predicting the class and how much they correlate with other features.

It is crucial to evaluate the performance and accuracy on these algorithms as it helps us to find the best model that represents and show us insights on how the model will work in the future.

Data

**What is our Dataset ?**

The dataset is the Pima Indian Diabetes dataset, in which featured **768 instances** described by **8 numerics attributes**. There are **two classes** - **yes** and **no.** Each entry in the database corresponds to a patient’s record. The patients are from the Pima Indian heritage, hence the name of the dataset.

**What is Correlation-based feature selection(CFS) ?**

Correlation-based feature selection is a method of selecting a subset of the original attributes. It searches for the best subset of features, where best is defined by a heuristic which considers how good the individual features are at predicting the class and how much they correlate with the other features. Good subsets of features contain features that are highly correlated with the class and uncorrelated with each other.

Attribute selected by the CFS in this study are as follow:

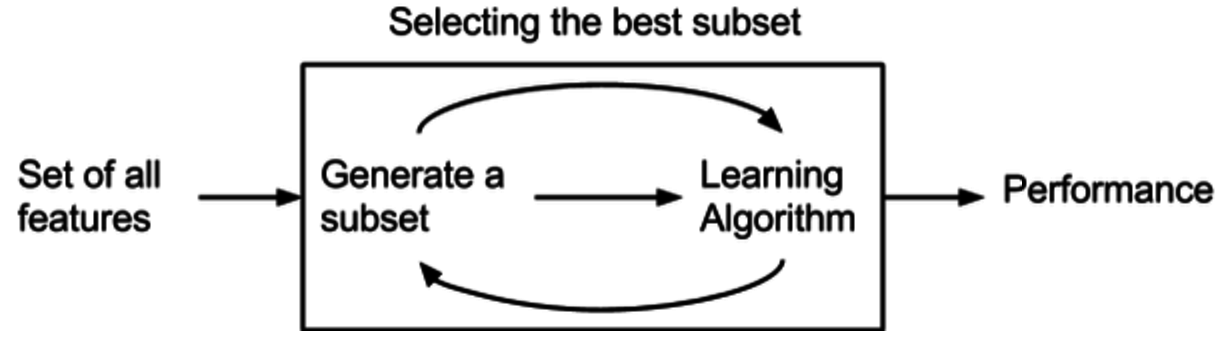
1 ) Plasma glucose concentration a 2 hours in an oral glucose tolerance test

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

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

4 ) Diabetes pedigree function

5) Age (years)



Results and discussion

In this section, we will compare the performance of the classifiers and discuss the effect of feature selection and whether it is beneficial to use feature selection or not.

*Table 1. Weka Classifier Performance*

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | ZeroR | 1R | | 1NN | 5NN | | NB | DT | | MLP | SVM |
| No feature selection | 65.10 % | 70.83 % | | 67.84 % | 74.48 % | | 74.74 % | 72.53 % | | 75.39 % | 76.30 % |
| CFS | 65.10 % | 70.83 % | | 69.01% | 74.48% | | 76.30% | 73.18% | | 75.78% | 76.69% |

*Table 2. Performance of our own KNN and NB implementation*

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | My1NN | | | My5NN | | | MyNB | | |
| No feature selection | | | 68.36 % | | | 75.39 % | | | 75.00 % | | |
| CFS | | | 68.23 % | | | 75.13 % | | | 76.43 % | | |

***Compare performance of the classifiers, with and without feature selection***

From the above table, we can see that the accuracy of most classifier with Feature selection is similar to that of those without feature selection. In the test using weka, a predefined data mining application, we can see not all classifier method experience a significant growth as ZeroR and 1R both have the same accuracy as before. From Table 1, We can observe 1NN, NB, DT, MLP and SVM experience some slight improved accuracy. We will now take a closer look at 1NN and 5NN, despite we are using the same algorithm, the number of nearest neighbours affect the accuracy of the method. We can see slight improvement over 1NN but not of 5NN.

***CFS - Effect of feature selection***

In Weka, CFS contains a few feature selections that gives higher compatibility to various different datasets. Those features include settings for missing value, number of threads used, and number of non-improving nodes search before terminate (searchTerm). Of those feature selections, CFS can run on large dataset as well as incomplete dataset. From the default selection of CFS, numThreads = 1, poolSize = 1, and searchTerm = 5. Due to the possible reason that our dataset is small enough, those feature selections does not make significant changes to the CFS results. However, in my opinion those values are well selected by default to fit our small dataset.

**C*ompare Our own KNN and NB against Weka***

In this study, we also implemented our own version of K-Nearest Neighbour and Naive Bayes Classifier. The result of the test on our own version of classifier is attached above as Table 2. Comparing to Weka’s KNN and NB, we can observe that our accuracy of 1NN, 5NN and NB without feature selection are all slightly higher than that of Weka’s version of KNN and NB. However, when we are comparing these with feature selection, we can observe a slight decrease in accuracy for my1NN and MyNB against that of weka’s. But a slight improvement of accuracy for my5NN against that of Weka’s 5NN.

***Is feature Selection Beneficial ?***

Though the improved accuracy of these classifiers are not significant, it proved that by using feature selection does improve the accuracy of the classifiers, but highly depending on the methods it adopted.

From the aspect of accuracy, feature selection benefit is insignificant, as the improvement will always be depending on the method used and also the sample size and also from above the overall improvement of accuracy is insignificant for some method and some with no improvement at all or even some with worse accuracy. Therefore, in terms of improving accuracy, feature selection is not beneficial.

Although it is not beneficial in improving accuracy on methods, it does have some benefits when using feature selection.

First of all, Improved run time when dealing with a large dataset. By reducing some feature read into the classifier, the run time will significantly reduced in later stages. In which, will enable user to run the algorithm at a higher complexity and do more evaluation.

Secondly, Easier of human operator to comprehend the result. The smaller set of data will allow human operator to have a better sense on what is going in the dataset,making it easier for him/her to explain the finding to client/professors in less complicated language.

Last but not least, Reduce noise in the large dataset. Dataset which are large in size will always contain some sort of noise in the data. By using feature selection, we can eliminate those noise and prevent having a false sense of good accuracy when running the method on a dataset without feature selection.

Conclusion

In conclusion, we can observe that feature selection is beneficial for some methods of classifier, but highly depending on the size of the dataset and the method used. In general, if we want to have higher accuracy, feature selection is not a must, but should be used when dealing with a large dataset as mentioned above. However, we must keep in mind when using feature selection, for some method, it is detrimental to their accuracy. Therefore, we must balance the benefit and impact to the accuracy before deciding whether to use feature selection or not.

Reflection

Chung Lai Lam, 450181104

This project is both interesting and rewarding. In the process of coding my own classifier and working with a teammate, which simulates a real world working environment, I have gain deeper understanding on how classifier works and does different way of manipulating the dataset impact the accuracy of different classifier. Not to mention, it is exciting to see that our own classifier slightly works better than weka’s implementation and really highlight the whole assignment. I choose this unit out of interest and I am glad that I learn more about different Machine Learning algorithm, gains hands on experience while doing this assignment and most importantly, inspire me trying to learn and research more about Machine Learning when doing the assignment.

Chih-Jung Liao, 450535064

Throughout this assignment, I gained deeper knowledge while processing data practically with different kinds of classifiers; understanding the accuracy, efficiency and the design of each of them is also interesting. They are unique and have different strengths, some perform quicker but inaccurate, some might be slower but more accurate. Furthermore, by implementing the classifiers ourselves, I understood deeper and often could relate real world situations with those classifiers. For instance, estimating weather report, predict game movements or anything you like if you collect data for it.