Zacs part

Aim

The aim is to compare the performance of machine learning algorithms in a prediction task. The task is to predict the likelihood of diabetes in female Pima peoples. This study highlights the model’s performance in terms of accuracy and uses this performance to explain their possible application in medicine.

1NN and 5NN

Our implementation did about as well as weka’s for 1NN, while weka did about 4% better with feature selection and 6% without feature selection for 5NN. Overall the accuracy is quite high for both; mainly due to normalising the data so that all values lie between 0 and 1. This makes the distance calculations not outweigh each other for different classes which reduces the likelihood of overfitting to a particular feature based on the metric used to measure it; e.g. features like age will always have a difference of at least one where as Plasma Glucose Concentration can have differences far smaller. By normalising the differences can be equated so that the metric doesn’t mislead the distance function. This does create some issues though, because it effectively equates the importance of all the features which is most likely not the case; as some features will be of greater importance than others. Feature selection alleviates this problem a little bit but still has the problem of the selected features having different importance which our implementation will ignore. In general, KNNs computation time can be quite high for larger datasets but by using feature selection KNN models can be built and tested quite efficiently if being used at a larger scale.

MLP

The MLP model did very well in terms of accuracy and compared to most neural nets is quite simple too. The only work comes from configuring starting weights and number of hidden layers which is automated by Weka. MLP did quite well but there may be better methods in the neural net family as its possible for our MLP to have reached a local optimum in terms of reducing the error instead of the global optimum. More testing should be done (especially out of sample) before it should be used as we do not know if this MLP is overfitted to the training data; meaning it won’t handle new data points well.

SVM

SVM had the highest accuracy with or without feature selection. SVM can generate non-linear decision boundaries by leveraging the kernel function making it computationally efficient. Because it tries to maximise the margin between points it can lead to a very effective classifier which is reflected in the results. Maximising the margin of the decision boundary makes it far easier to classify the fringe cases then a classifier like KNN as to do so in KNN you need to expand your neighbours but in doing so you expand the region of search which can lead to including points which lie to far away from the new data point; swinging the vote and hence the prediction. Due to its ability to deal with fringe cases it is likely for SVM to generalise to new data however this was not tested. SVM is also quite efficient in terms of training being much faster than MLP and in terms of cost and performance the trade off is quite good due to its high performance.

Reflection

For me the most important thing was the ease of implementation for some of these methods. KNN does not take a lot of work to get going and its predictive power is quite good considering the amount of work you do. I found it really interesting to see how it compared to wekas performance as it only did a bit worse which was nice to see but I’m interested to see in how you can make it even better (maybe experiment with different distance metrics). It was also good to see all these different methods we’ve learnt about being put in practise as it lets you test your understanding while also giving you an insight into how their strengths and weaknesses translate into the real world or with real data.

Conclusion

Our results show that some classifiers outperform others however this is by no means a suitable comparison of all the classifiers we used as each has its own set of strengths and weaknesses which translates to performance for different kinds of data or prediction problems. What our report does show is that for data with minimal noise, some class imbalance (500 vs 267) and a relatively small set of features; SVM can outperform unsupervised methods like KNN or supervised methods like MLP etc as well as rule-based algorithms like 0R and 1R. Its important to note that this can easily change based on the balance in the dataset or the performance metric being used. Furthermore, because the performance is only measured for in sample performance, we cannot make any comment on their performance in practise, although there in sample performance is a good indicator of what to expect. Its also important to consider the computational cost of running these algorithms as in practise they may not be suitable due to how slow or expensive they are – its important to strike a balance by consulting with domain experts on the resources available, the performance needed etc.

In our case, SVM seemed to perform the best after removing some attributes. Although its performance was the highest there is still a large room for improvement and because the classifier could potentially affect a person’s life (if used for prognostics or screening) it is paramount that it can be as accurate and reliable as possible.

Going forward, expanding the dataset would be a good first step, as right now it only looks at female patients and only adults. Expanding the set of features or doing feature selection with a domain expert may also lead to better success in terms of prediction. Currently, none of the models have an output to explain the relationship between features (except for 1R) which makes its practical use limited as well.