# COMP3608 Assignment 2 - Report

## Aim

This study aims to evaluate the precision of Weka's internal classifiers designed for generic data to the Naive Bayes and Decision Tree classifiers programmed specifically for the Pima Indian Dataset. Given the extent of the dataset, finding an efficient classifier that can correctly predict whether a patient has diabetes will provide us with valuable information about which personal characteristics and medical measurements strongly correlate with the possibility of having diabetes. These findings can be used in further research to explain the correlations and potentially find a biological or genetic link between personal traits and diabetes.

## Data

The dataset provided is the Pima Indian Diabetes dataset. It contains 768 instances described by 8 numeric attributes. There are two classes - **yes** and **no**; the class shows if the person shows signs of diabetes or not. Each entry in the dataset corresponds to a patient's record; the attributes are personal characteristics and test measurements. The patients are from Pima Indian heritage. The patient attributes in the dataset are:

- Number of times pregnant
- Plasma glucose concentration a 2 hours in an oral glucose tolerance test
- Diastolic blood pressure (mm Hg)
- Triceps skinfold thickness (mm)
- 2-Hour serum insulin (mu U/ml)
- Body mass index (weight in kg/(height in m)^2)
- Diabetes pedigree function
- Age (years)

The attributes were provided in two forms - numeric data corresponding to the results provided by the tests and discrete data which gave each data numeric data point a value of "low", "medium", "high" or "very high". In this study, the numeric data was normalized along each column, such that every data point was in the range [0,1].

The normalized numeric data and discrete data also went through Correlation-based Feature Selection using Best-First Search as the search method. The following were selected as features that are highly correlated with the class and uncorrelated with each other:

- Plasma glucose concentration
- 2-Hour serum insulin
- Body mass index
- Diabetes pedigree function
- Age

# Results

## Classifier Accuracy

| Numeric Data | ZeroR | 1R | 1NN | 5NN | NB | MLP | SVM | MyNB |
|---|---|---|---|---|---|---|---|---|
| No feature selection | 65.1% | 70.8% | 67.8% | 74.5% | 75.1% | 75.4% | 76.3% | 75.4% |
| CFS | 65.1% | 70.8% | 69.0% | 74.5% | 76.3% | 75.8% | 76.7% | 76.7% |

| Nominal Data | DT unpruned | DT pruned | MyDT |
|---|---|---|---|
| No feature selection | 75% | 75.4% | 74.1% |
| CFS | 79.4% | 79.4% | 78.5% |

# DT Diagrams

## DT unpruned

```
att2 = high
|   att6 = high
|   |   att4 = high
|   |   |   att1 = low
|   |   |   |   att7 = high
|   |   |   |   |   att8 = high: yes (16.0/5.0)
|   |   |   |   |   att8 = low
|   |   |   |   |   |   att3 = high: yes (11.0/5.0)
|   |   |   |   |   |   att3 = low: no (5.0/2.0)
|   |   |   |   att7 = low
|   |   |   |   |   att3 = high: no (43.0/19.0)
|   |   |   |   |   att3 = low: yes (10.0/4.0)
|   |   |   att1 = high
|   |   |   |   att3 = high: yes (29.0/8.0)
|   |   |   |   att3 = low
|   |   |   |   |   att7 = high: no (2.0)
|   |   |   |   |   att7 = low: yes (3.0)
|   |   att4 = low: no (13.0/4.0)
|   att6 = low: no (29.0/4.0)
att2 = low
|   att6 = high
|   |   att5 = high
|   |   |   att8 = high
|   |   |   |   att7 = high: yes (7.0/3.0)
|   |   |   |   att7 = low: no (28.0/4.0)
|   |   |   att8 = low: no (43.0/4.0)
|   |   att5 = low: no (48.0/2.0)
|   att6 = low: no (66.0)
att2 = very high
|   att5 = high
|   |   att6 = high: yes (103.0/16.0)
|   |   att6 = low
|   |   |   att8 = high: yes (12.0/3.0)
|   |   |   att8 = low: no (4.0/1.0)
|   att5 = low: no (3.0/1.0)
att2 = medium
|   att8 = high
|   |   att5 = high
|   |   |   att6 = high
|   |   |   |   att7 = high: yes (37.0/10.0)
|   |   |   |   att7 = low
|   |   |   |   |   att3 = high: no (57.0/24.0)
|   |   |   |   |   att3 = low
|   |   |   |   |   |   att4 = high: yes (15.0/7.0)
|   |   |   |   |   |   att4 = low: no (3.0/1.0)
|   |   |   att6 = low: no (27.0/3.0)
|   |   att5 = low: no (8.0)
|   att8 = low
|   |   att6 = high
|   |   |   att1 = low
|   |   |   |   att4 = high
|   |   |   |   |   att7 = high
|   |   |   |   |   |   att3 = high: no (17.0/2.0)
|   |   |   |   |   |   att3 = low: yes (7.0/3.0)
|   |   |   |   |   att7 = low: no (54.0/8.0)
|   |   |   |   att4 = low: no (24.0/1.0)
|   |   |   att1 = high: yes (2.0/1.0)
|   |   att6 = low: no (42.0/1.0)
```

## DT pruned

```
att2 = high
|   att6 = high
|   |   att4 = high: yes (119.0/51.0)
|   |   att4 = low: no (13.0/4.0)
|   att6 = low: no (29.0/4.0)
att2 = low: no (192.0/14.0)
att2 = very high: yes (122.0/24.0)
att2 = medium
|   att8 = high
|   |   att6 = high
|   |   |   att7 = high: yes (37.0/10.0)
|   |   |   att7 = low: no (80.0/33.0)
|   |   att6 = low: no (30.0/3.0)
|   att8 = low: no (146.0/17.0)
```

## MyDT

[1, {"high": [5, {"high": [7, {"high": [6, {"high": [2, {"high": [0, {"high": [4, {"high": [3, {"high": "yes", "low": "yes"}], "low": "yes"}], "low": [4,
{"high": [3, {"high": "yes", "low": "yes"}], "low": "yes"}]}], "low": [0, {"high": "no", "low": [4, {"high": [3, {"high": "yes", "low": "yes"}], "low":
"yes"}]}]}], "low": [4, {"high": [3, {"high": [2, {"high": [0, {"high": "yes", "low": "no"}], "low": [0, {"high": "yes", "low": "yes"}]}], "low": [2,
{"high": [0, {"high": "no", "low": "yes"}], "low": "no"}]}], "low": "yes"}]}], "low": [3, {"high": [6, {"high": [2, {"high": [4, {"high": [0, {"high": "yes",
"low": "yes"}], "low": "yes"}], "low": [4, {"high":[0, {"high": "no", "low": "no"}], "low": "no"}]}], "low": [2, {"high": [4, {"high": [0, {"high": "no",
"low": "no"}], "low": "no"}], "low": [4, {"high": [0, {"high": "yes", "low": "yes"}], "low": "yes"}]}]}], "low": [6, {"high": [4, {"high": [2, {"high": "no",
"low": [0, {"high": "yes", "low": "yes"}]}], "low": "no"}], "low": "no"}]}]}], "low": [3, {"high": [4, {"high": [6, {"high": "no", "low": [7, {"high": [2,
{"high": [0, {"high": "no", "low": "no"}], "low": "no"}], "low": [2, {"high": "no", "low": [0, {"high": "yes", "low": "yes"}]}]}]}], "low": [6, {"high":
"yes", "low": "no"}]}], "low": "no"}]}], "very high": [4, {"high": [5, {"high": [0, {"high": [6, {"high": "yes", "low": [2, {"high": [7, {"high": [3,
{"high": "yes", "low": "yes"}], "low": "yes"}], "low": [7, {"high": [3, {"high": "yes", "low": "yes"}], "low": "yes"}]}]}], "low": [7, {"high": [6, {"high":
[3, {"high": [2, {"high": "yes", "low": "yes"}], "low": "yes"}], "low": [3, {"high": [2, {"high": "yes", "low": "yes"}], "low": [2, {"high": "yes", "low":
"yes"}]}]}], "low": [6, {"high": "yes", "low": [3, {"high": [2, {"high": "yes", "low": "yes"}], "low": [2, {"high": "yes", "low": "no"}]}]}]}]}], "low": [7,
{"high": [3, {"high": [0, {"high": [6, {"high": [2, {"high": "yes", "low": "yes"}], "low": "yes"}], "low": [6, {"high": "yes", "low": [2, {"high": "yes",
"low": "yes"}]}]}], "low": "yes"}], "low": [6, {"high": "no", "low": [3, {"high": "no", "low": [2, {"high": [0, {"high": "yes", "low": "yes"}], "low":
"yes"}]}]}]}]}], "low": [6, {"high": "yes", "low": "no"}]}],"low": [5, {"high": [4, {"high": [7, {"high": [6, {"high": [2, {"high": [3, {"high": [0, {"high":
"yes", "low": "no"}], "low": [0, {"high": "no", "low": "yes"}]}], "low": "yes"}], "low": [3, {"high": [0, {"high": [2, {"high": "no", "low": "no"}], "low":
[2, {"high": "no", "low": "no"}]}], "low": "no"}]}], "low": [2, {"high": "no", "low": [3, {"high": [6, {"high": [0, {"high": "no", "low": "no"}], "low": [0,
{"high": "no", "low": "no"}]}], "low": "no"}]}]}], "low": [2, {"high": [7, {"high": "no", "low": [3, {"high": [6, {"high": "yes", "low": [0, {"high": "no",
"low": "no"}]}], "low": "no"}]}], "low": "no"}]}], "low": "no"}], "medium": [7, {"high": [5, {"high": [6, {"high": [0, {"high": "yes", "low": [3, {"high":
[2, {"high": [4, {"high": "yes", "low": "yes"}], "low": [4, {"high": "yes", "low": "yes"}]}], "low": "yes"}]}], "low": [4, {"high": [2, {"high": [0, {"high":
[3, {"high": "no", "low": "no"}], "low": [3, {"high": "no", "low": "yes"}]}], "low": [3, {"high": [0, {"high": "yes", "low": "yes"}], "low": [0, {"high":
"no", "low": "no"}]}]}], "low": "no"}]}], "low": [2, {"high": [0, {"high": "no", "low": [6, {"high": "no", "low": [3, {"high": [4, {"high": "no", "low":
"no"}], "low": "no"}]}]}], "low": [0, {"high": "yes", "low": [3, {"high": "no", "low": [6, {"high": [4, {"high": "no", "low": "no"}], "low": "no"}]}]}]}]}],
"low": [5, {"high": [3, {"high": [0, {"high": [6, {"high": "yes", "low": [4, {"high": [2, {"high":"yes", "low": "yes"}], "low": "yes"}]}], "low": [6,
{"high": [2, {"high": [4, {"high": "no", "low": "no"}], "low": [4, {"high": "yes", "low": "yes"}]}], "low": [2, {"high": [4, {"high": "no", "low": "no"}],
"low": [4, {"high": "no", "low": "no"}]}]}]}], "low": [6, {"high": [2, {"high": "no", "low": [4, {"high": [0, {"high": "no", "low": "no"}], "low": "no"}]}],
"low": "no"}]}], "low": [6, {"high": [4, {"high": "no", "low": [2, {"high": "no", "low": [3, {"high": "yes", "low": [0, {"high": "yes", "low": "yes"}]}]}]}],
"low": "no"}]}]}]}]

NOTE: numbers 0,1,2...7 correspond to the index of the column to look up.
Pls forgive

# Discussion

The discussion will use ZeroR as the baseline for comparing all other classifiers. ZeroR correctly classified 65.1% of the data in 10 fold stratified cross validation. All classifiers performed better than ZeroR.

## Classifiers

Using numeric data without feature selection, SVM, MLP and Naive Bayes perfomed best, correctly classifying 76.3%, 75.4% and 75.1% respectively. This is unsurprising as these three methods use the most sophisticated statistical analysis out of the classification algorithms used in this study. It is worth noting that Multi Layer Perceptron took by far the most time to train and classify using 10 folds.

1 Nearest Neighbour performed the worst, classifying only 67.8% correctly which is only 2.7% better than ZeroR - within a margin of error. This is most likely due to the fact that 1NN will optimize only to 1 nearest point, why may be noise, as a result making suboptimal classification decisions. Even 1R performed better, with 70.8% correct classification. 5NN on the other hand easily outperformed 1R, classifying 74.5% correctly - very close to the top 3 performers.

When using CFS on numeric data, significant increases in accuracy were observed in 1NN and Naive Bayes, both increasing by 1.2%. Both of these classifiers take into account all data and make statistical decisions based on probabilities that are on a continuous scale, making it possible for noisy data to sway the classifier to an incorrect decision. As a result, using CFS would have significantly reduced noisy data, as there are less attributes to consider and the attributes are uncorrelated with each other. Given CFS data, Naive Bayes outperforms MLP by 0.5%, perhaps due to the fact that MLP can only classify linearly divisible data.

The custom Naive Bayes implementation performed marginally better than Weka's - only 0.3% more correct classifications on data with no feature selection and 0.4% on data with feature selection. This could be attributed to two factors - Python's float precision may be different to Weka's (Java implementation) which would result in test cases with very similar probabilities to get classified differently as the lack of float precision would sway the classifier towards a particular class. We could also postulate that Weka uses Laplace corrections automatically, which would add a miniscule amount of noise to the data, resulting in more misclassification than the custom Naive Bayes implementation. The increase in percentage difference using CFS data further reinforces this hypothesis that the custom Naive Bayes implementation is less sensitive to noise than Weka's.

Using discrete data without feature selection, the pruned decision tree performed marginally better with a 0.4% increase in accuracy. This is due to the fact that pruning a DT removes branches which provide little classification power, as a result reducing overfitting - making the final DT more resistant to noisy data when training and classifying.

However, when employing CFS both implementations correctly classified significantly more instances than without feature selection. The intuition behind this involves the fact that DTs tend to overfit for the given data, which means that reducing the scope of the data (as such generating a smaller tree) improved the classifiers' resistance to noise, hence increasing their accuracy. Furthermore, both pruned and unpruned DTs correctly classified the same number of instances, indicating that CFS removed the same noise from the data as would be removed from pruning, as such nullifying the gains seen in pruning DTs.

The custom Decision Tree implementation performed worse than Weka's counterparts - correctly classifying 1.3% less instances with no feature selection and 0.9% with feature selection. It is also evident that MyDT produced a larger tree than Weka's unpruned implementation. These results strongly indicate an error in the implementation of MyDT, as the algorithm used for constructing the decision tree would be the same and as such the decision tree should also be the same. There is also evidence that MyDT is more sensitive to noise, as there was a reduction in differences of incorrectly classified instances using CFS. Since CFS is used to reduce noise, it implies that MyDT performs relatively better with data that has low noise.

## CFS

As seen above, employing CFS improved some classifiers' accuracies. The biggest gains were seen on discrete data, as the small selection of values (only "low" or "high" in most cases) for each attribute significantly increased noise on those attributes. Having CFS remove these therefore greatly improved the classifiers' accuracy. It is then can be concluded that CFS is a good technique to employ when the data has a lot of noise and/or the classifier used could be sensitive to noise.

CFS attribute selection was intuitive and predictable.  It is well known that patients with diabetes have higher than usual levels of glucose, which makes Plasma glucose concentration a well correlated attribute with the class. Diabetes patients also tend to have unusual levels of insulin, which means that 2-hour insulin serum could be closely related to the class. Mainstream media points out that living a healthy lifestyle reduces one's chances of diabetes which made the Body Mass Index an obvious attribute to include, while insurance companies tend to give higher premiums to older people and those with family history of diabetes, which means that including both Age and the Diabetes Pedigree Function made intuitive sense.

However it is surprising that both glucose concentration and insulin were picked by the CFS as the human body uses insulin to control the blood glucose levels, which would indicate that the two are closely correlated with each other.
Another unusual CFS decision was to include both BMI and glucose levels. Intuitively one would assume that people with higher BMI lead unhealthier lifestyles, resulting in increased blood glucose levels. However both were included in the CFS, indicating that there is no strong correlation between these two attributes.

# Conclusion

In conclusion, the most accurate classifiers were Weka's Decision trees using discrete data passed through CFS. Both the pruned and unpruned DT correctly classified on average 79.4% of instances using 10 fold stratified cross validation. This indicates that there is correlation between the chances of an individual having diabetes and their general well being, age, family history and other genetic components. Since using discrete data produced better overall results, we can further conclude that only simple medical tests are needed to determine if an individual is at risk.

Due to multiple unexpected results found in this study, the following future work could be beneficial:

- Study the effect of reducing noise in data to the classification accuracy of pruned and unpruned DTs
- Study any correlation between the blood glucose levels and insulin levels in patients with diabetes
- Study correlation between BMI and blood glucose.

# Reflection

This assignment has taught us how to implement basic classification algorithms and evaluate them. The most valuable things we learnt included the need to have a baseline when comparing classifiers and why ZeroR is a good baseline. It was also surprising to see how CFS improved the accuracy of some classifiers but not others, paying special attention to how CFS greatly improved discrete data and nullified any gains of pruning the DT.

Another key takeaway from this assignment involved the use of stratification in 10 fold cross validation. We found it to be a very innovative way of using all test data in such a way as to maximise its utility for both training and testing purposes.