EDP Presentation

Project 2 Findings

Outline of Content

- Pre-Processing
- Considerations in design
- The Classifier
- Performance
- Limitations
- Improvements
- Questions

Pre-Processing

Why are we Missing Data?
Handling Nan Values

Removing Nan

- Good when we have lots of data or if a row contains a lot of Nan values.
- Bad when the dataset is incomplete, and a large portion of the rows have missing values.
- Can create Bias if the dataset has Nan values that are due to outside influence.

Imputing values

- Good for handling data that has scattered Nan values
- Bad if the Nan values are due to outside influences.
- Creates bias depending on method used.

Mean vs Median imputation

- Choose the method depending on the type of data, and whether or not the data has outliers.
- Median imputation preserves median and Mean imputation preserves mean.
- If data has lots of outliers use median imputation.
- If data is well formed use mean imputation.

- For my classifier used median imputation with mean centred.
- Difference between mean and median imputation on the yeast dataset was small.
- Chose this since we had other classifiers to compare mine with later

Mean centred

- Subtracts a value from each point so that the mean becomes 0
- This allows for easier interpretation since 0 is now in our dataset
- Left: Median imputed, Middle: median imputed & mean centred
- Right: Original

	mcg	gvh
count	1484.000000	1484.000000
mean	0.497628	0.499643
std	0.131472	0.121954
min	0.110000	0.130000
50%	0.480000	0.490000
max	1.000000	1.000000

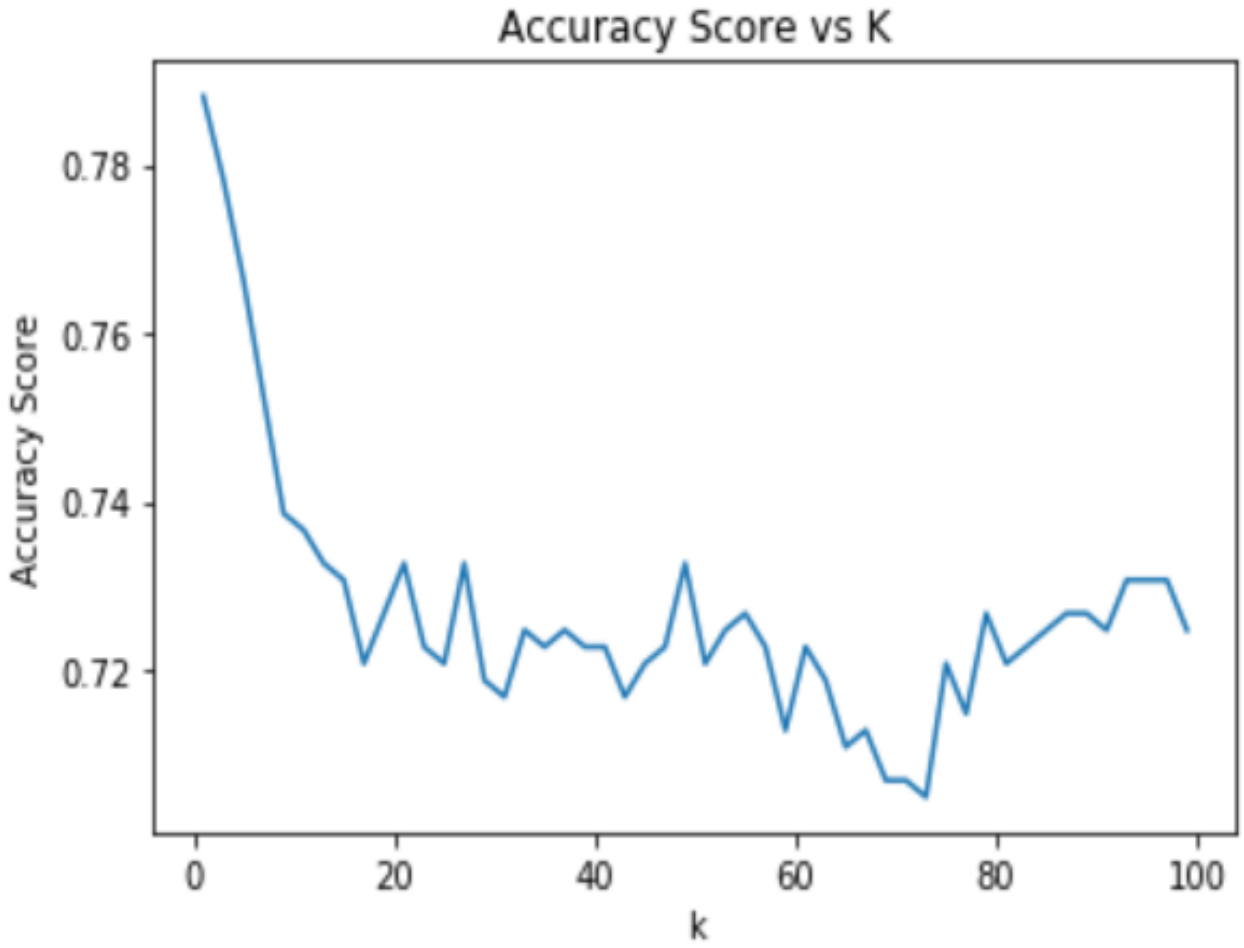
	1	2
1484.000000	1484.000000	
0.000000	-0.000000	
0.131472	0.121954	
-0.387628	-0.369643	
-0.017628	-0.009643	
0.502372	0.500357	

	mcg	gvh
count	1352.000000	1449.000000
mean	0.499349	0.499876
std	0.137625	0.123410
min	0.110000	0.130000
50%	0.480000	0.490000
max	1.000000	1.000000

Designing the Classifier

- Considered two types of Classifiers with and without feature engineering
- k-NN vs Decision Tree
- Choose k-NN as it outperforms the Decision Tree

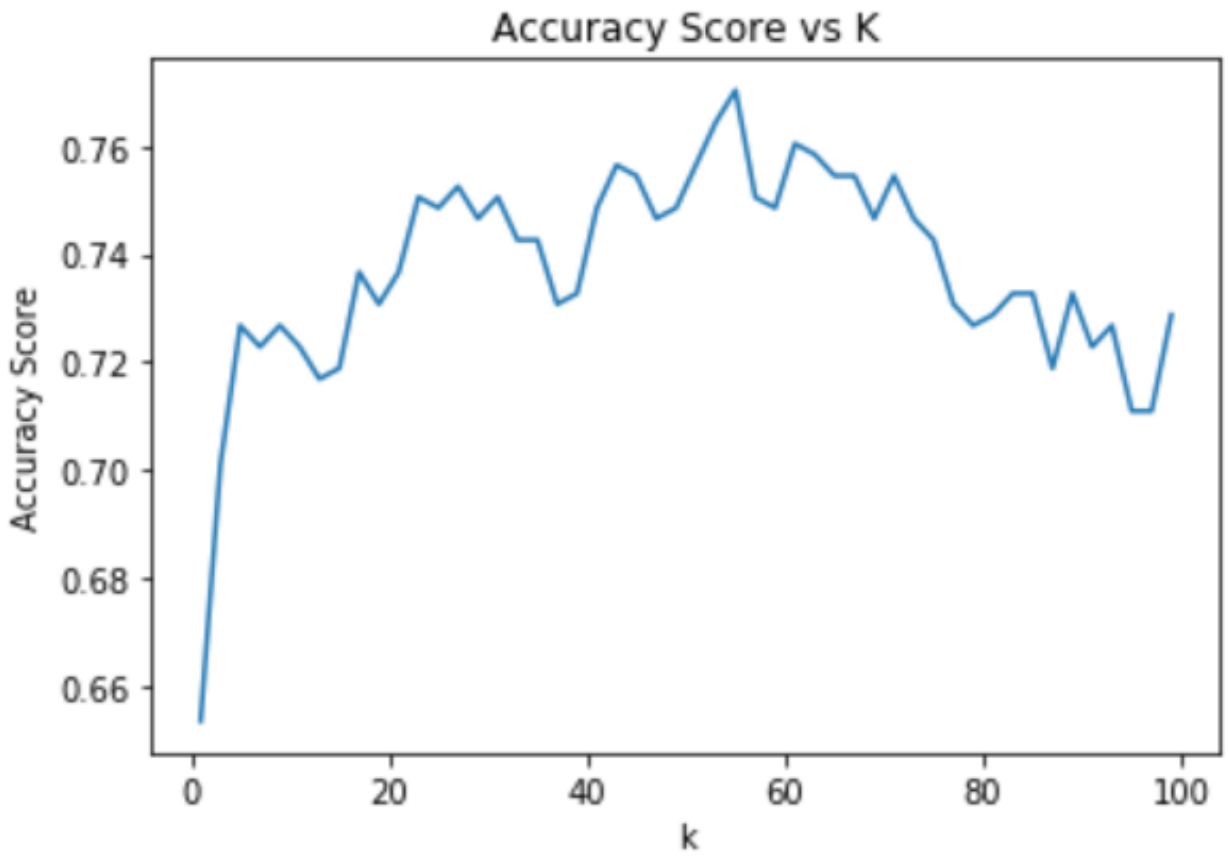
```
Accuracy Score K-NN 5:  0.7663366336633664
Accuracy Score K-NN 10:  0.7326732673267327
Accuracy Score Decision Tree:  0.7465346534653465
Highest Accuracy Score K 0.7881188118811882
best accuracy score varying depth of dt:  0.7544554455445
```



Feature Engineering

- Cluster labels vs Interaction
- Figure: k-NN with feature engineering
- Tested interaction by seeing how much mutual information was gained.

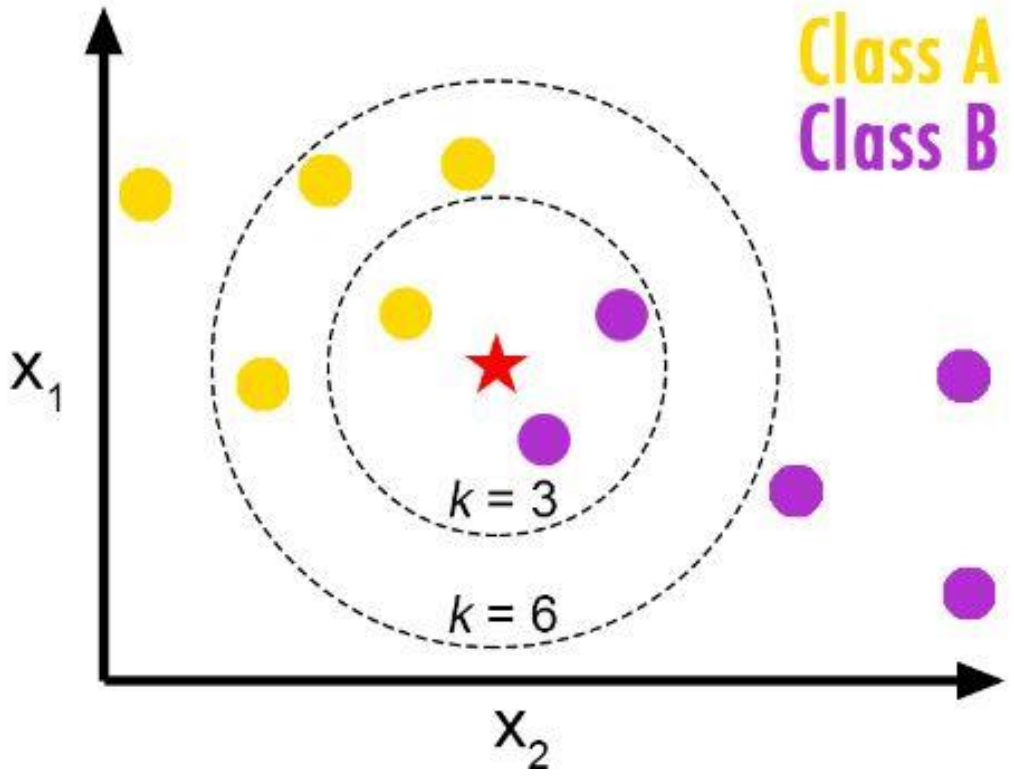
```
Accuracy Score kmeans cluster: 0.4871287128712871

Accuracy Score K-NN 5:  0.7267326732673267
highest score: 0.7702970297029703
```



How k-NN based classifier works

- Find the k nearest neighbours of the point you want to classify.
- Remarks:

1) Choose odd K value for 2 class problem
2) Don’t Use K that’s a multiple of the number of classes
3) Main issue with k-NN is the complexity in searching for nearest neighbours (On large Datasets)



Parameters of classifier

- K is the only parameter that can be varied (with ease)
- Impact of K:

1) K too large, data is classified into the majority class
2) K too small, highly variable, unstable decision boundary. Small changes to training set brings large changes in classification

- Selecting K:

1) Create validation set (from a portion of training data), vary k and observe training. This can lead to validation error
2) Take k to be the square root of n (give or take 1)

- I took k = 5, performed better than other k's
- Can also change features which model is based on i.e which interaction pairs to include.

Performance

- **Classification Accuracy:** Overall, how often is the classifier correct?
- **Classification Error:** Overall, how often is the classifier incorrect?
- **Sensitivity:** When the actual value is positive, how often is the prediction correct?
- **Specificity:** When the actual value is negative, how often is the prediction correct?
- **False Positive Rate:** When the actual value is negative, how often is the prediction incorrect?
- **Precision:** When a positive value is predicted, how often is the prediction correct?

```
[[ 42 115]
 [ 22 326]]
 TN:  42  FP:  115  FN:  22 TP:  326
accuracy score using formula: 0.7287128712871287
accuracy score using metrics: 0.7287128712871287
Classification error: 0.2712871287128713
Sensitivity: 0.9367816091954023
False Positive Rate: 0.732484076433121
Precision: 0.739229024943310 7
```

The type of data that we have doesn't require anything except that the classifier is correct. So we value accuracy the most.

Generalizing & Limitations

- The classifier would likely maintain a decent accuracy if applied to other datasets.
- Main influences on accuracy would be due to the imputation that was required. If the new Dataset was complete there would be some bias from the initial imputation which could cause a loss in accuracy. Likewise if the data required lots of pre-processing, we may see accuracy go either way.

- **Limitations:**

1) Optimal number of neighbors?
2) Missing Value treatment
3) Outlier sensitivity due to neighbour based algorithm
4) Imbalanced data causes problems
5) K-NN slows with size; as dataset grows efficiency of algorithm declines very fast.

Improvements & better classifier

- Improvement can be made in the selection of the interaction pairs to include in the model. Find a better measure than mutual information to test for interaction significance.
- **Performance of a better classifier**
- **Multilayer Feed-forward Neural Networks and Supervised Learning.**
- A feedforward neural network is an artificial neural network wherein connections between the nodes do not form a cycle.
- This class of networks consists of multiple layers of computational units, usually interconnected in a feed-forward way. Each neuron in one layer has directed connections to the neurons of the subsequent layer.

http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.2.4228&rep=rep1&type=pdf

Aristoklis D. Anastasiadis, George D. Magoulas, and Xiaohui Liu

Questions?