Machine Learning
Project 1
Report: K-NN Classifier

Similarity Metric: Euclidean Distance

The similarity metric should take into consideration the kind of variable (discrete, continuous, nominal) and what you are trying to measure. Care should be take when deciding this heuristic as it can greatly influence the performance on the classifier.

For the implementation of K-Nearest-Neighbours Classification technique on the Abalone data set, Euclidean distance was chosen. The majority are continuous values with an exception for the sex attribute, which is nominal. This attribute was ignored in determining the distance between each of the test instances and each of the training instances in the final implementation of K-NN. At one stage this was accounted for, with a penalty for having the incorrect sex. However, change in classifier performance after evaluation was negligible so this was removed for simplification. The Euclidean distance was simple to implement with the dataset provided.

Validation Framework: Accuracy, Precision

The evaluation techniques implemented include Accuracy and Precision. Accuracy is simply the number of times the classifier correctly predicts a class label for all its predictions. Precision is the positive prediction value of the classifier. That is

$$Precision = \frac{TP}{TP + FP}$$

Where TP = True Positives and

FP = False Positives

A True Positive in this classifier was chosen to be the case where the classifier chose 'Young' and the actual class label was 'Young'. Whereas a False Positive was the case where the classifier chose 'Old' and the actual class label was 'Old'.

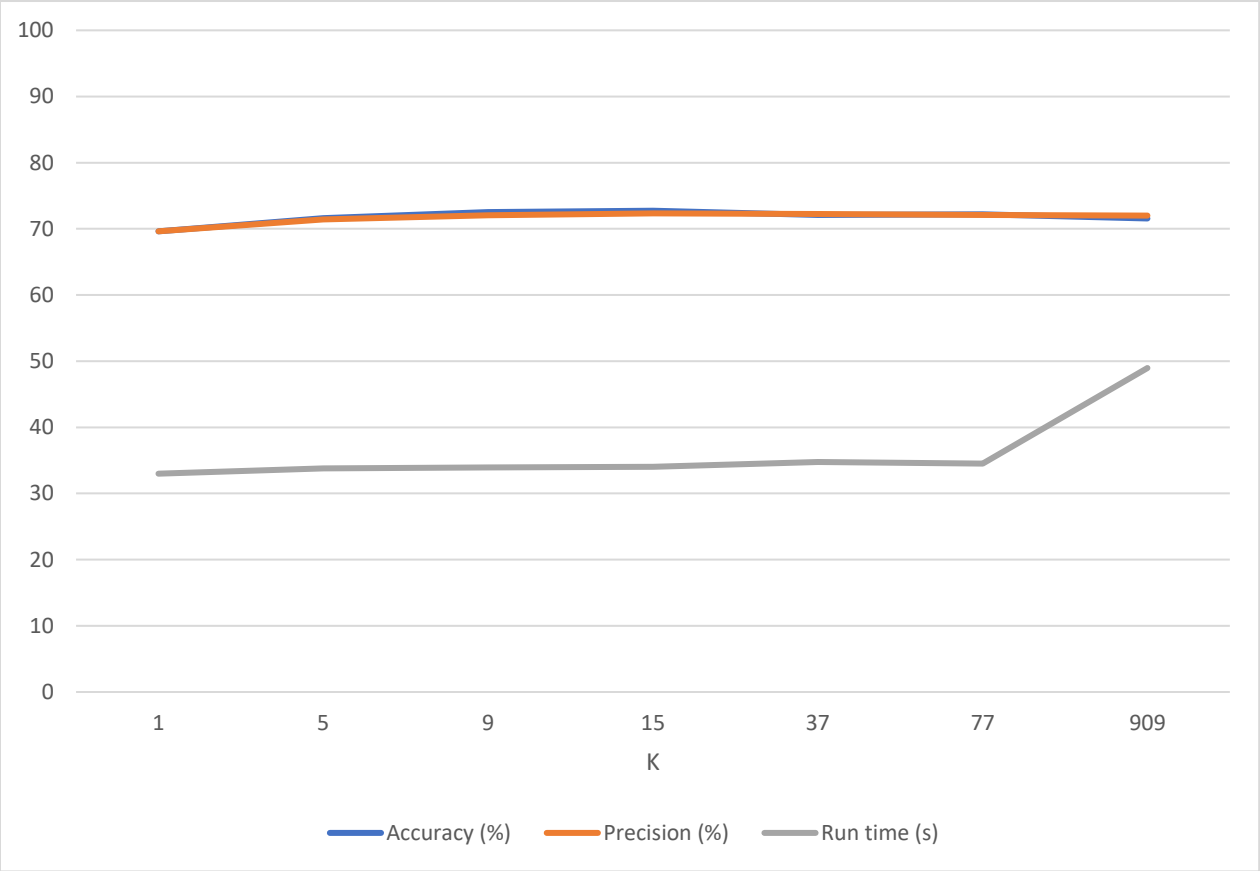
The system is mostly interested in a positive prediction.

Choosing the Value of K

The following is some program execution data with a 33:67 Test-Training Split over varying K values

K	1	5	9	15	37	77	909
Accuracy (%)	69.62119	71.60135	72.54475	72.73827	72.10934	72.18191	71.57716
Precision (%)	69.62452	71.42339	72.06098	72.33326	72.24849	72.08992	72.00931
Run time (s)	32.99504	33.8034	33.93667	34.02006	34.74105	34.51583	48.96246

A graph displaying changes with Accuracy, Precision and Run time, over the increasing K values



K is the number of nearest neighbours we choose to extract from the training set to use for voting. A good number for K is case specific.

There is no completely optimal value of K, as found through running some tests with different K values. This is mostly a cause of the K-NN classifier heuristics being very simple.

By increasing K, the K-Nearest-Neighbours classifier should become increasingly like the ZeroR Classifier. ZeroR classifier simply predicts the majority category. A large K - including too many nearest neighbours - is non-specific, increases run time and essentially defeats the purpose of the algorithm. Class boundaries become less distinct as noise is reduced.

With decreasing K, the classifier performance should be lower due to noise (overfitting). We run the risk of losing information about other close data points. For example when K=1 this is now a simply Nearest-Neighbour classifier.

It is difficult to select the highest performing K on evaluation functions without testing a range of K values. The best performing K on this dataset after testing was K=15 on Precision and K=9 on Accuracy. Although the differences were only fractions of a percent, I would be confident in good value for K being around this mark. This may have to do with the size of the dataset but should be examined on a case by case basis.

Improvements and Recommendations

Representations can support performance. The visualization or representation of the data should be easy to understand and informative. The Abalone dataset has 8 attributes, effectively 8 dimensions. It would hard to represent this graphically in a fathomable dimension, and staring at thousands of instances of Abalone is not in any way informative. Perhaps mapping it to a lower dimension in a scatter plot would suffice. We can gain more insight about the data set through machine learning.

To improve the performance of the classifier, it would benefit to more closely examine each attribute and its relevance. Since the assignment release we have further learnt about weighting significance of certain attributes – special weighting techniques could be implemented to further enhance the accuracy. Different voting methods could be implemented and tested also.

In summary, the K-NN classifier is a very basic classification method where an object is classified by a majority vote of its neighbors. And even on approaching domination of vote by majority, the classifier still performs at a reasonably stable accuracy and precision.