# K-Nearest Neighbor (KNN) Machine Learning Algorithm

Background: KNN algorithm is a simple, easy to implement supervised machine learning algorithm that can be used to solve both regression and classification problems. The KNN algorithm assumes that similar things are near each other.

How KNN algorithm works:

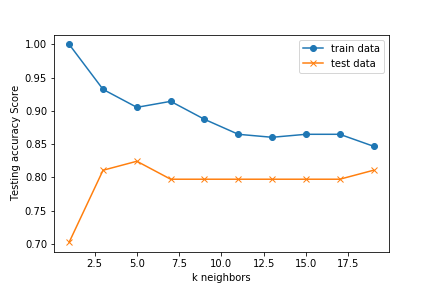
1. Load the data
2. Initialize K to chosen number of neighbors
3. For each instance in the data:
   1. Calculate the distance between the query example and the current example from the data.
   2. Add the distance and the index of the example to an ordered collection
4. Sort the ordered collection of distances and indices from smallest to largest (in ascending order) by the distances
5. Pick the first K entries from the sorted collection
6. Get the labels of the selected K entries
7. If regression, return the mean of the K labels
8. If classification, return the mode of the K labels

Implementation:

We used KNN as a classifier algorithm, and used the heart disease dataset found in the UCI Machine Learning website (<https://archive.ics.uci.edu/ml/datasets/Heart+Disease>)

After splitting the heart disease data into train and test sets, the data was scaled and encoded. A KNN classifier model was setup, we looped through a list of K values to determine the best accuracy score.

The results showed that the K=5 results in the best score for both train and test data.



The overall accuracy of the model was 0.784.

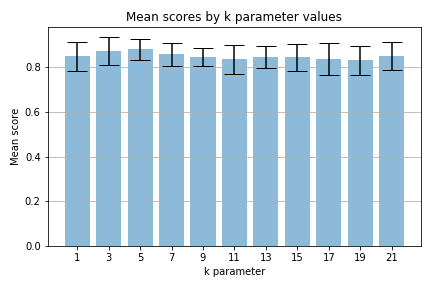
Algorithm modification and tuning: Given the limited data samples, K-Fold Cross-Validation was used to resample and evaluate the KNN model. Cross-validation is primary used in machine learning estimate the skills of the model on unseen data. The general procedure is as follows:

1. Shuffle the dataset randomly
2. Split the dataset into k groups
3. For each unique group
   1. Take the group as a hold out or test data
   2. Take the remaining groups as a training data set
   3. Fit a model on the training set and evaluate it on the test set
   4. Retain the evaluation score and discard the model
4. Summarize the skill of the model using the sample of model evaluation scores

When using K-Fold Cross-validation procedure, it is important to note that each observation in the dataset is assigned to an individual group and stays in that group for the duration of the procedure. This means that each sample is given the opportunity to be used in the hold out set 1 time and used to train the model k-1 times. K values [1,3,5,7,9.11,13,15,17,19,21], were used in this procedure.

In addition, GridSearchCV along with cross-validation was used to evaluate the model across various parameters.

The graph below shows mean accuracy across the different k values.



Conclusion: KNN is very simple to implement and robust with regards to the search space; for instance, the classes do not have to be linearly separable. There are only two parameters to tune (k and distance metric.). The main disadvantage of the algorithm is that it is expensive testing of each new observation, due to the need to compute its distance from all known observation which becomes problematic with a dataset with a large number of features.