**K-Nearest Neighbours: Classification and Regression**

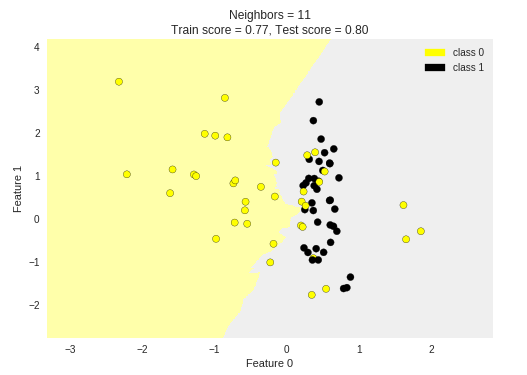
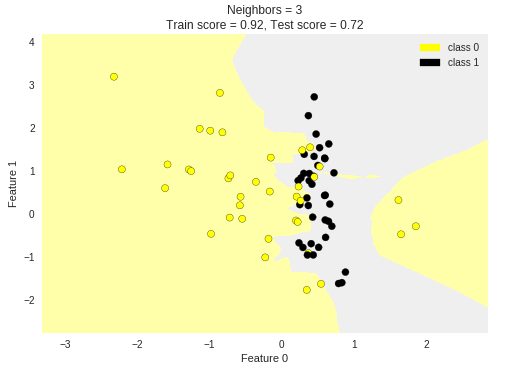
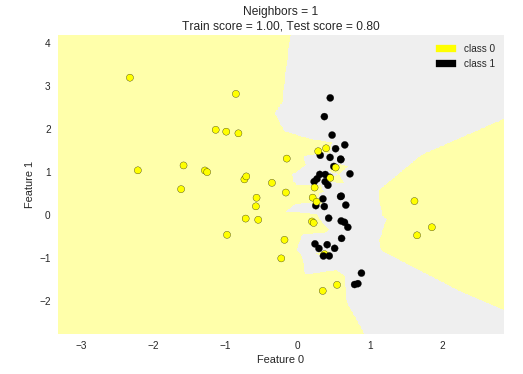
This supervised learning method can be used for **classification** and **regression**. KNN often gives potentially accurate but sometimes unstable predictions.

**Classification:**

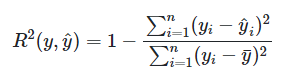
For KNN classification the algorithm memories the training set. To then classify a new instance, it follows three steps:

* Find the closest instances to that test point, (n\_neighors = n).
* Get the labels of these closest instances.
* Predict the labels by combining the labels of the closest point by simple majority vote.

A KNN with n=1 would generally be overfitting the data, and would result in the training score being 1, while the testing score is likely to be much lower. Reducing the n value will results in a less complex model that will be able to generalize well to new instances.

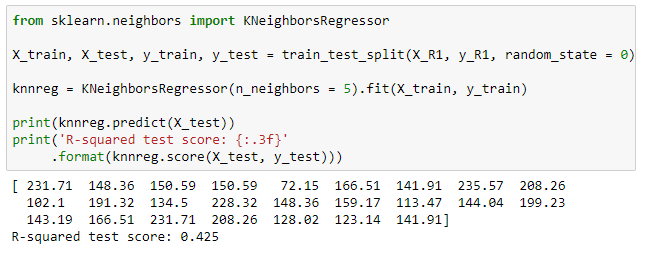


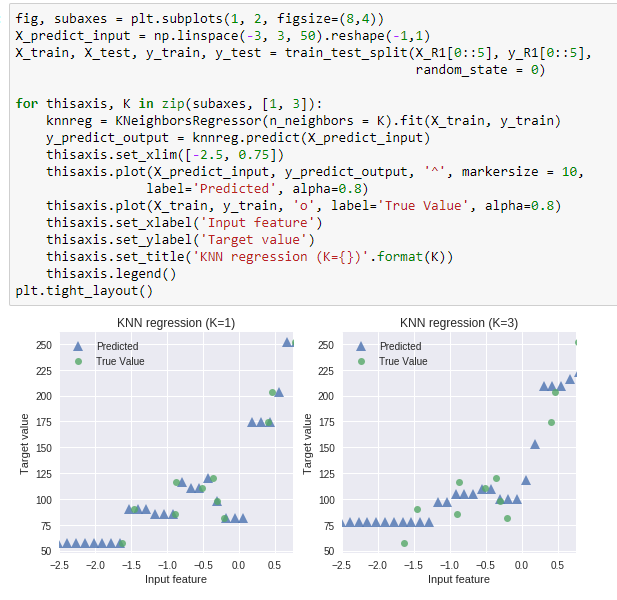
**R^2 Score:**

This is a way to assess how well a model performs, with a minimum of 0 and maximum of 1. Where y is the true value, y (hat) is the predicted value, y (bar) is the mean of the true values.

**Regression:**

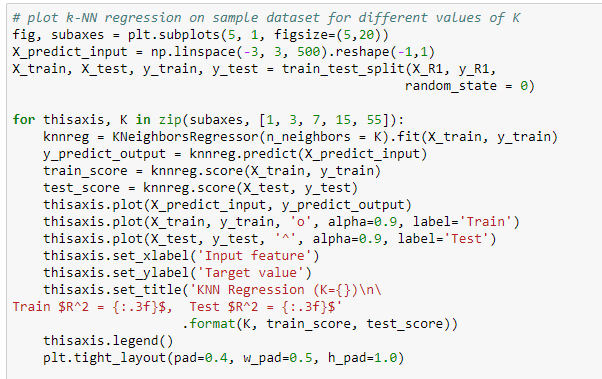
KNN regression work in a very similar way.

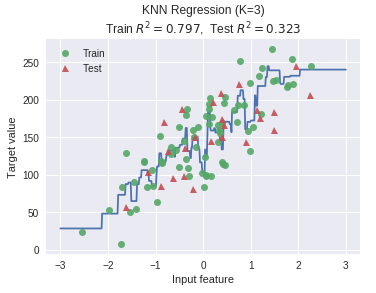
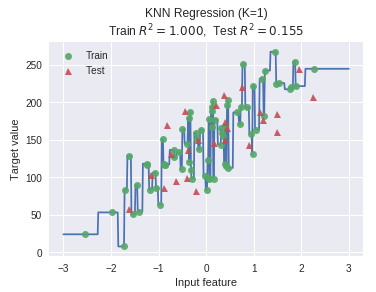
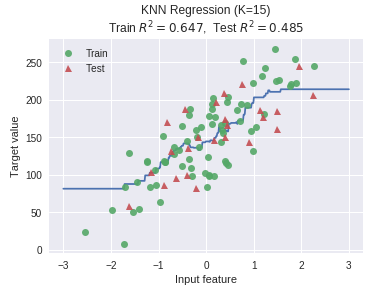
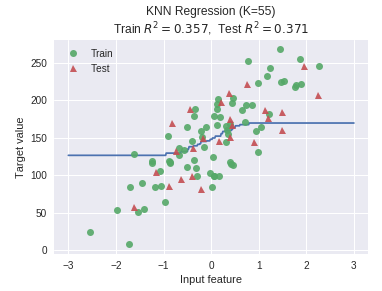




The above two figure show how the model predicts values based on the data. Given an input feature value the model then finds the closest point to that new value and then gives it a target value that is the mean of these closest points.

Regression model complexity as a function of N:



With smaller values of K we can see that the model doesn’t generalize well to new instances, and is very complex. On the otherside with a very high K we can see that the model starts to underfit the data and also results in poor R^2 scores for both training and testing.