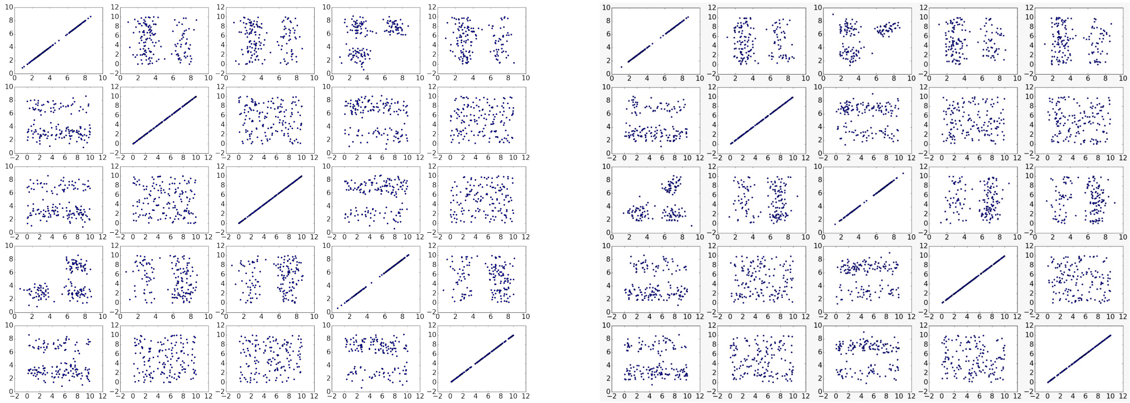
Coursework 1: Report

Symbols, Patterns and Signals
Dameli Ryspayeva
Joshua Van Leeuwen

Introduction

Feature Selection

Given the training set data we needed to identify which two features separate the classes of the data in order to know what attributes to use. To do this we visually inspected the data by plotting each pair of attribute values against each other. The results were:



Josh

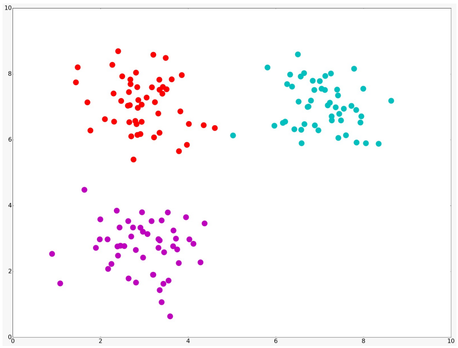
Dameli

Figure 1: Attribute Plots

Inspecting the graphs it is clear that in Figure 1, the attributes that separate the classes into clusters the most for the first set are the first and fourth attribute values whereas the first and fourth attribute values for the second set separate the data into clusters the most. This means that these attributes make up the features of their data sets.

Identifying the Classes

Now we have identified that there are three classes displayed by the three clusters we then applied K-means to the data set to obtain the 3 clusters. By using K-means we were able to identify which data points belongs to which clustering. With this information we were able to visualise the different groupings using colour as shown by Figure 2.



Josh

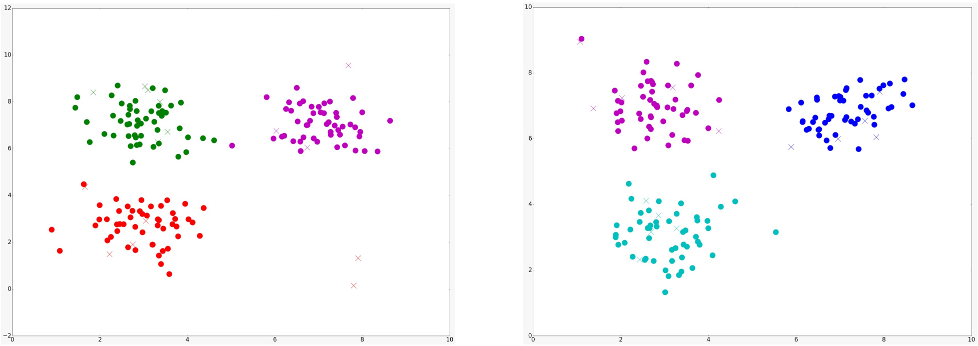
The result of visualising K-means and assigning the points to a class is what we expected as it has matched what we thought the clustering should be. K-means is a centroid-based clustering algorithm which partitions features into several sets of data that we refer to, as clusters, by minimising the within-cluster sum of squares. Mathematically, this is denoted as:

$$\boldsymbol{\arg\min_{S} \sum_{i=1}^{k} \sum_{x \epsilon S_i} \parallel x - \mu_i \parallel^2}$$

According to the number of clusters, n, the algorithm randomly initialises n points that are called centroids. Given that k-means is an iterative algorithm, it iteratively continues doing two steps: assigning a point to a cluster and moving centroids closer to the centre of this cluster. In our case, it will go through each point of the features set and will assign them to one of the cluster groups depending on which cluster is closer to that point. After that, it will recalculate the average of all points within a cluster and will assign the centroid to the current centre of the cluster. The two steps are repeated until the centroid value no longer changes.

Nearest-centroid Classification

When classifying new data points into our set one can use the nearest-centroid classification algorithm to fit them into a class.



Josh

Dameli

Figure 2: Nearest-centroid classification

As you can see from the left plot, their are two test data points which are outliers, being a great distance out from any cluster or other points. Even though it is clear that they do not belong in a cluster, and so class, they have still been assigned to the red class. This is because we have specified that there are three clusters that all data points belong to one.

Maximum-likelihood Classification

Discussion of Results

References

[1] K-Means