# An Investigation Into K-Means Clustering in the Iris Data Set with R

The code presented uses the “Hartigan-Wong” Algorithm of k-means clustering as this is the default method used by the *kmeans* function in R and there is no argument supplied alter the default behaviour. In R The Hartigan-Wong method initializes by first selecting *k* points from the data set provided at random where *K* is the number of clusters to be searched for, to be the initial centroids (the points which will become the estimated centre of the clusters). In this case *K* is equal to three so three points will be chosen for the initial centroids and three clusters will be determined (RDocumentation, 2022). This differs from the description of the Hartigan Wong Algorithm given by CITE which states that each data point is initially assigned to one of *K* “Classes”(Clusters) and an initial centroid for each cluster is calculated as the mean of these points meaning they are all in roughly the same area.

As the *kmeans* function in R does not allow for visualisation of the clustering process it will be assumed that the points are assigned at random to each centroid chosen as is described in CITE. Figure 1 shows initial cluster assignment and centroid selection as performed in R and Figure 2 (overleaf) shows the same following the initialisation procedure described in CITE.

Chart, scatter chart

Description automatically generated

Figure 1: Initialisation of Clusters by Selecting Points at random as Centroids and randomly assigning all points to one (R)

It will be assumed that the R implementation uses the Hartigan-Wong algorithm as described in CITE following initialisation. The function *hw\_kmeans* (Appendix 1) was created to allow this process to be visualised and examined under this assumption. The *hw\_kmeans* function attempts to group the points into clusters by assigning each point to the cluster which results in the lowest overall sum of squares of error for the clustering as a whole. To do this Each point is assigned to each cluster in turn. For each cluster assignment the means are calculated as the “new” centroids for each cluster and the sum of squares for the entire clustering is calculated using:

Calc

Chart, scatter chart

Description automatically generated

Figure 2: Initialisation by Assigning each Point to A Cluster at Random and Calculating the Mean as the Centroid

Including the point being tested. the assignment resulting the lowest sum of squares result is chosen to be the new cluster for the point being tested. This process is repeated for each point and the points are iterated over until the centroids do not move for an entire iteration indicating that an optimal clustering has been found. This is repeated from the initialisation step as many times as is requested to account for the (pseudo)randomness of placing the initial clusters with the clustering with the lowest sum of squares value being returned. Figures 3 and 4 show the results of this as performed by the *hw\_kmeans* function using random point selection and mean calculation as methods for determining the intial centroids respectively.

Figure 3: Final Clustering Using Random Point Selection Initialisation

Figure 4: Final Clustering Using Mean of Assigned Points Initialisation

As can be seen from Figures 3 and 4 the initialisation method does not seem to make a great deal of difference to the eventual clustering. This is because even if the clusters are selected as random points from the provided data sets, they are then immediately calculated as the mean of the points assigned as part of the algorithm, which is how the “mean of assigned points” inittialisation method begins.

The *hw\_kmeans* method allows for the animation of the clustering process which can then be converted to video[[1]](#footnote-1) using the *FFMPEG* the command for which can be seen in Appendix 2.

What does Hartigan-Wong do to cluster points?

How does this affect the Clusters? Points aren’t always in the closest cluster

## References

## Figure References

## Bibliography

## Appendices

1. Hartigan Wong Clustering Animation video: <https://youtu.be/FlV48DzRGjI> [↑](#footnote-ref-1)