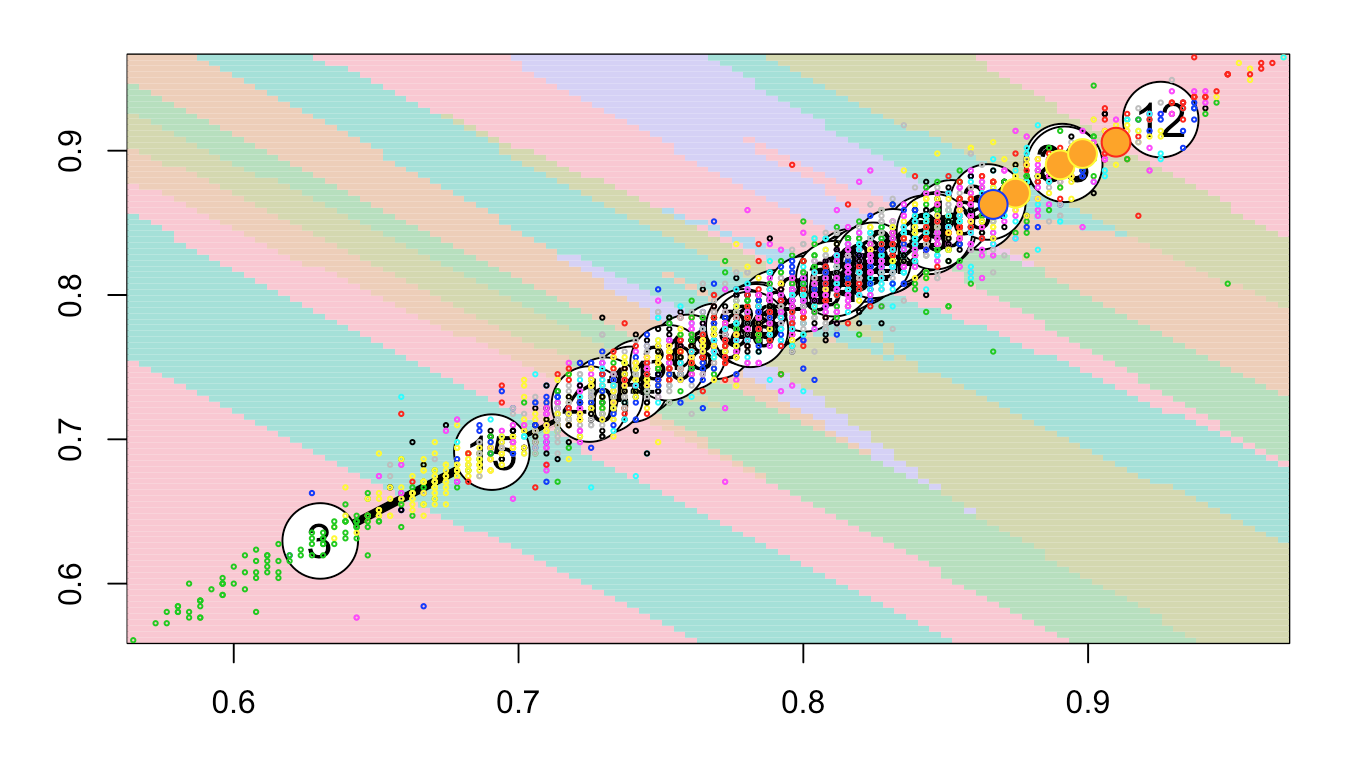
K-means: Hieroglyph classification

Before beginning the process of clustering, I spent quite some time manually running through all the images in the Hieroglyphics folder. From my rough overview, there were close to 25-30 different, discernable symbols in the data. However, before I could apply any clustering, I had to read in all the png files (4415 files) into RStudio. I did this just once in a loop and saved the resulting object as a csv file. However, the csv format added an additional index column (which caused exponentially higher within cluster SS and made predictions impossible – took me over an hour to figure this out) which had to be taken out separately. Though I had a rough estimate for the clusters, I decided to check my intuition using the within cluster sum of squares (WSS) for upto 60 clusters (twice my initial prediction) for the kmeans algorithm on the image data. However, it was tough to tell from the graph which k would really be suitable since I could neither figure out a specific “elbow” point nor did the WSS have a local minimum before 60. I thus chose 3 trial k values: 20,40 and 60. The hypothesis was that 20 clusters would be inaccurate in clustering the more than 20 symbols while 60 would over-classify and create multiple clusters of the same symbol. 40 was a tentative middle ground. I ran the k-means for all the three cluster numbers and found 40 to work the best of the lot (based on manual sifting). As hypothesized, 60 had too many similar clusters while 20 had too few. I also tried to visualize the clusters through a bivariate plot which looked at the first two (scaled) principle components and plotted the clusters based on those. While the two PCAs explained 42% of the total variability in the data, there was still significant overlap within the clusters in the two dimensions to derive any meaning. As a safeguard measure, I reran the k-means for 30 clusters which again was similar to the result of 20 clusters.



I then proceeded to the prediction images with the 40 cluster model and used a minimization function to minimize the squared distances from the point to the cluster’s centre. This approach turned out to be fairly accurate as 1,2 and 5, which were relatively similar to one another went into a single cluster while 3 (correctly) went to a ‘knife’ symbol cluster and 4 went to a ‘bird’ cluster. However, in order to verify this, I also used the *kmeans.predict* function from the *SwarmSVM* package which gave me the exact same cluster grouping.

In order to attempt another visual depiction of how the predicted values faired, I used the *kcca* method from the *Cluster* library which helped me plot the clusters and identify where the predicted values (bigger orange dots) lied relatively to those clusters.



Based on my 40 cluster analysis, I realized that shapes are better classified than objects. Clusters like 4,17 and 35 which comprise of circles and rectangles are much better sorted compared to some other ones like 1 and 2. However, cluster of distinct objects, like birds or knives were also very accurate in sorting (see 5,12,13,19,27,40). I also realized that 40 might not have been the exact number as some of the clusters had similar symbols in them (see 1-33 and 35-37).