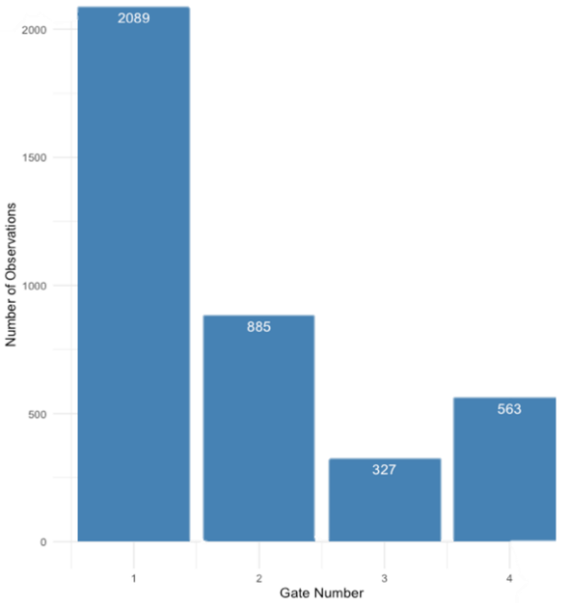
STU33011 – MLA Assignment – Jack Cleary 19333982

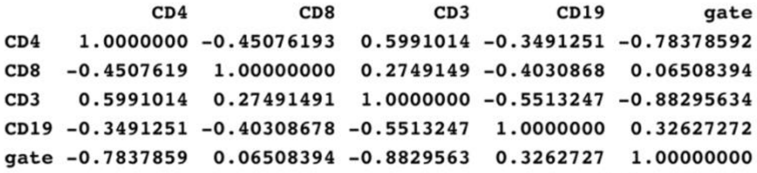
Descriptive Analysis

In this analysis we will look at the Healthy Flow dataset (subset), which consists of 3,864 observations and 4 attributes. Each of these 4 attributes take the form of continuous numeric variables. Also included is a gate variable, which separates our observations into one of four gates based on the results of a gating analysis that was carried out on the data. The purpose of our analysis of this dataset is twofold; one, to make use of unsupervised learning methods to identify subsets of the data which are similar to the identified gated populations, and two, to make use of supervised learning methods to accurately predict which data are assigned to the identified gates.



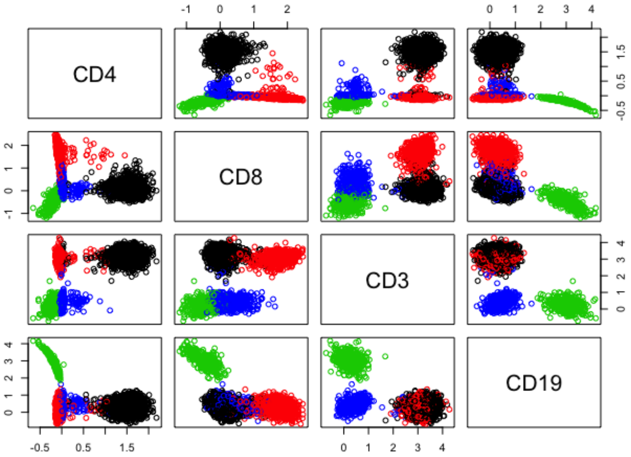
Each population of cells has unique characteristics with respect to the protein markers CD4, CD8, CD3 & CD19. The purpose of the different gates is to identify distinct populations of such cells within a dataset. As we can see from Figure 1.1 over, a large portion of the data was put in gate 1, and only a very small portion in gate 3. This indicates that, at least in the context of our sample, Population 1 is a much more common population than any other, and Population 3 is a lot more rare.

Some interesting observations can be made from the Correlation Matrix shown below, the first being the high correlation between protein markers CD3 and CD4. This high correlation indicates that a relationship may exist between the pair, however I have quite a limited knowledge on the topic and so I cannot be certain. Another point we may take from this matrix is the very high negative correlation between the CD3 marker and the Gate variable, which would indicate that a high CD3 value is probably a characteristic of Population 1. Similarly, the high correlation between CD4 and the Gate variable would imply that a high CD4 value is typical of Population 1 or 2. Lastly, the negative correlation between CD19 and the other three variables would suggest a difference in this protein marker and the other three.



```
            CD4         CD8        CD3       CD19        gate
CD4   1.0000000 -0.45076193  0.5991014 -0.3491251 -0.78378592
CD8  -0.4507619  1.00000000  0.2749149 -0.4030868  0.06508394
CD3   0.5991014  0.27491491  1.0000000 -0.5513247 -0.88295634
CD19 -0.3491251 -0.40308678 -0.5513247  1.0000000  0.32627272
gate -0.7837859  0.06508394 -0.8829563  0.3262727  1.00000000
```

Correlation Matrix



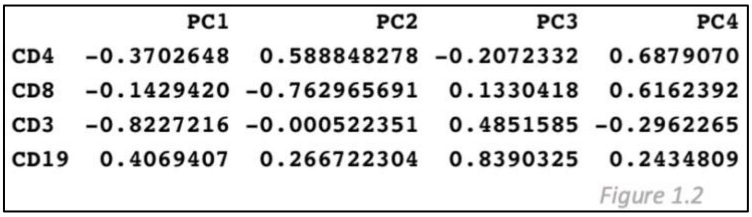
Inspection of the this pairs plot also reveals some information about the dataset. This tells us about relationships between each of the four variables, coloured by gate number. The black data points represents gate 1, the red represents gate 2, the green represents gate 3 and the blue represents gate 4. Firstly, we can note that gate 1 has high levels of CD4 and CD3, lower levels of CD8 and very low levels of CD19. Similarly, gate 2 has high levels of CD8 and CD3 but low levels of CD4 and lower levels of CD19. Lastly, gate 3 has low levels of all variables except for CD19, where it has the highest. Due to my limited knowledge on the topic, I cannot infer much from this however we can now talk with confidence about the characteristics of each gate with respect to the 4 variables.

Unsupervised Learning Methods

I will now use three different unsupervised learning methods in order to try and identify subsets of the data which are similar to the gated populations provided in this dataset. Unsupervised learning is a type of machine learning in which the algorithm is not provided with any pre-assigned labels or scores for the training data. I will omit the “gate” variable when running these analyses and later compare my results to the identified gated populations.

Principal Component Analysis (PCA)

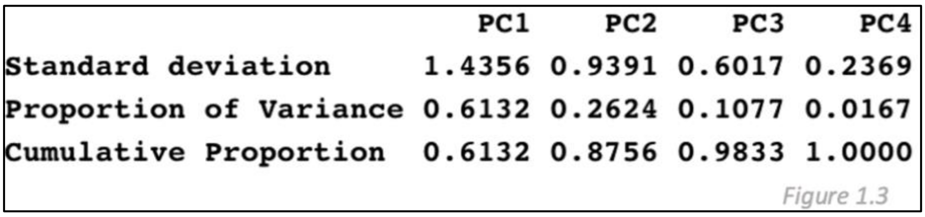
PCA is a method used to re-express the data with as much variation as possible in terms of a few linear combinations of the existing variables. It can be used to show any association among variables. The variables in our dataset have already been standardized and so there will be no need to scale the data. Running this analysis generates four Principal Components (PCs), which are ordered by decreasing importance. In Figure 1.2, we can see the loadings of each of the original variables for the four different PCs. We see that for PC1, variables CD4, CD8 & CD3 all have negative values while CD19 has a positive value. So therefore, cells with high CD4, CD3 and CD8 values will have a lower, probably negative first principal component. I mentioned above how these three characteristics seem to be somewhat different to the CD19 variable, and hence perhaps a low first principal component is typical of a particular population of cells.



```
            PC1          PC2        PC3        PC4
CD4  -0.3702648  0.588848278 -0.2072332  0.6879070
CD8  -0.1429420 -0.762965691  0.1330418  0.6162392
CD3  -0.8227216 -0.000522351  0.4851585 -0.2962265
CD19  0.4069407  0.266722304  0.8390325  0.2434809
```

Figure 1.2

Figure 1.3 below provides a summary of the variance explained by each of the PCs. As we can see, the third row shows the cumulative proportion of the variance explained by each additional Principal Component. For example, the first two PCs explain 87.56% of the variance in the data.



```
                          PC1    PC2    PC3    PC4
Standard deviation     1.4356 0.9391 0.6017 0.2369
Proportion of Variance 0.6132 0.2624 0.1077 0.0167
Cumulative Proportion  0.6132 0.8756 0.9833 1.0000
```

Figure 1.3

In order to choose an optimal number of principal components, we will now examine the “Scree Plot” shown below in Figure 1.4, which graphs the variance explained by each PC. Searching for an elbow in the graph, we see a kink at PC2. This reduces our dimensions by a satisfactory amount, and we can see from Figure 1.3 above that we also account for 87.56% of the variation in the data. This is a suitable amount to be able to draw results and so we plot the two principal components, colouring the observations by gate number.

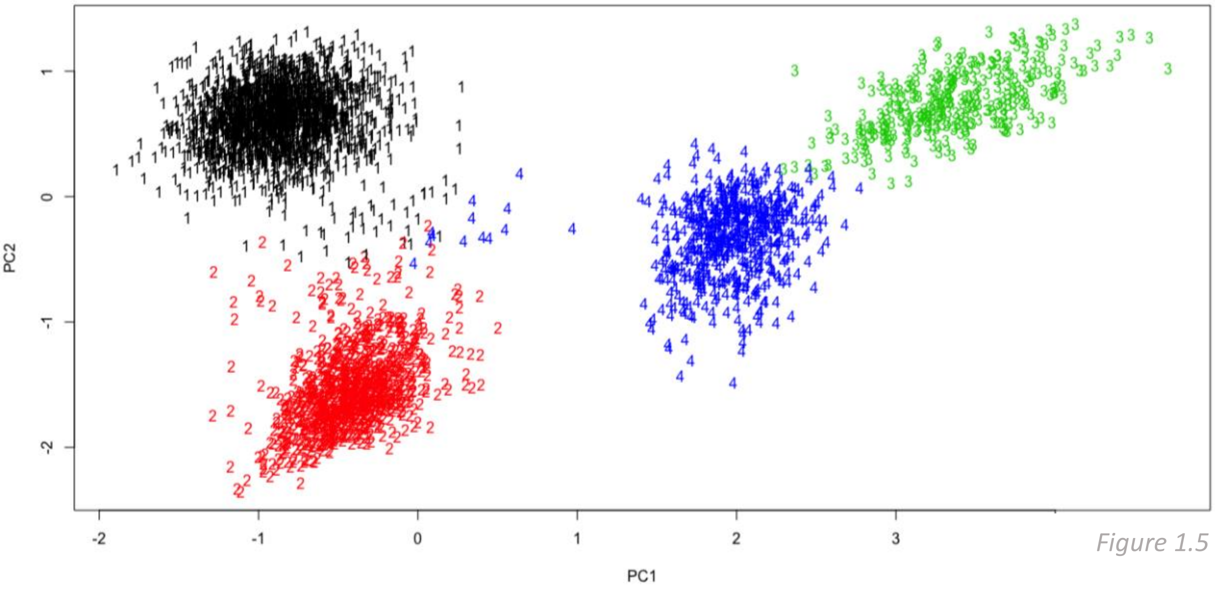
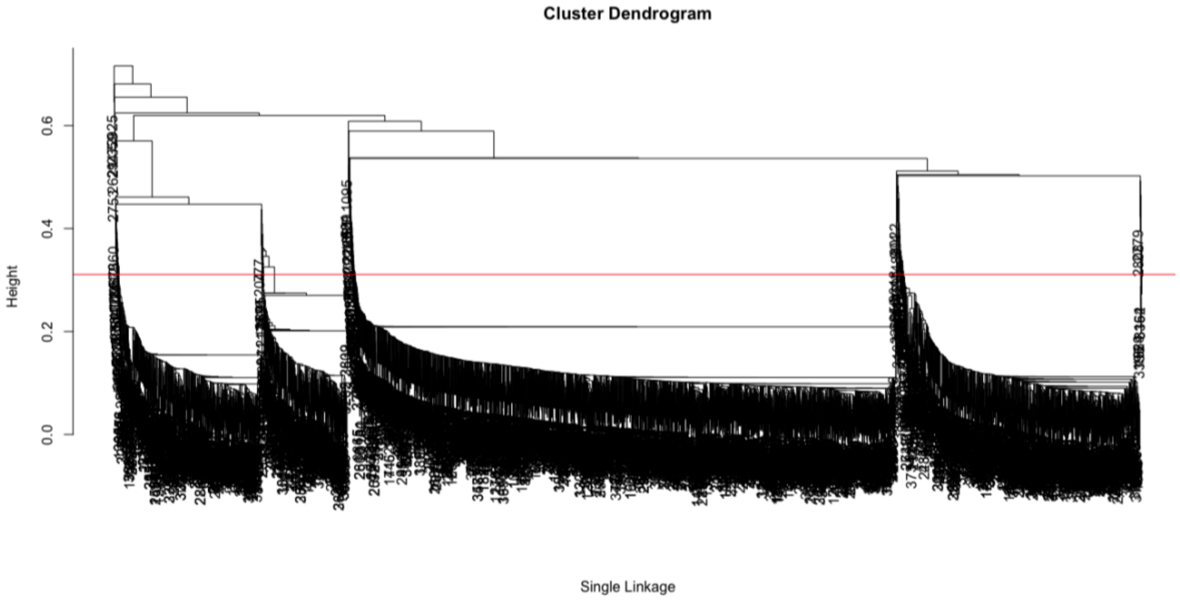


Figure 1.5

Plotting the first two principal components as in Figure 1.5 depicts four very clear clusters, representing the four gated populations. As we can see above, a negative first principal component is typical of a large portion of the dataset, gates 1 and 2. This is as expected from our earlier analysis of the loadings of PC1. The first two components do a very good job of structuring the data into 4 separate and very distinct populations, and hence this explains the dataset very well.

Hierarchical Clustering

Hierarchical clustering is a method that uncovers group structures by clustering the data into a tree-like structure over a series of steps in which the most similar observations are joined together. To determine the similarity between observations, we interestingly use dissimilarity measures. For my analysis I used the Euclidean distance measure, because this dataset only contains continuous variables and hence Euclidean is appropriate. I then clustered the data using complete, average and single linkages and compared the resulting dendrograms and clusters. I cut the trees at the recommended height which is equal to $\bar{h} + 3s_h$ ($\bar{h}$ is the average height at which groups are joined,

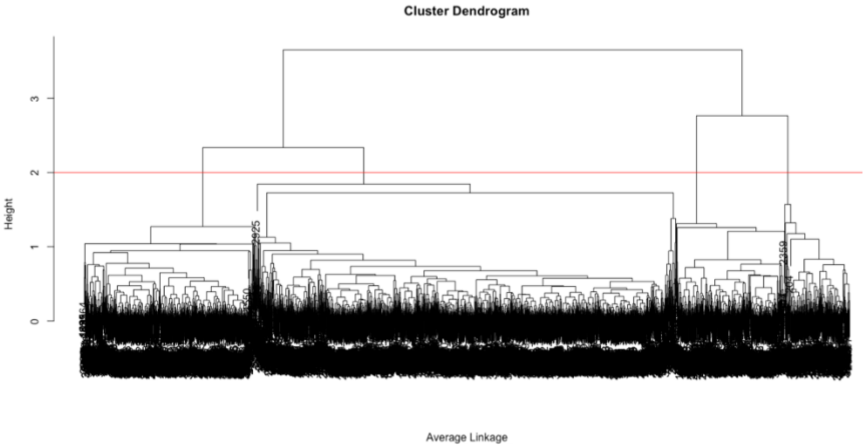


s_h is the standard deviation of these heights). Using single linkage resulted in 83 different clusters and comparing these with the original gated populations yielded an Adjusted Rand Index of 0.9475. A Rand Index of 1 would indicate that the groupings are exactly the same, and zero would indicate that not one data point within the groupings agrees. 0.9475 was the highest ARI that was achieved using this method, and the resulting Dendrogram can be seen above (please note the dendrogram is not very clear due to the size of the dataset).

Of the 83 clusters formed, 75 of them contain only one single observation and another 4 contain either 2 or 3 observations. If we ignore these clusters as outliers, we are left with 4 clusters as shown below. This table gives a comparison between the original gated populations (G1, G2, G3 and G4) and the new clusters (C1, C2, C3 and C4). As we can see, bar a slight overlap between C1 and C4, the results are almost perfect, hence the extremely high Adjusted Rand Index.

Cluster/Gate	G1	G2	G3	G4
C1	2035	0	0	0
C2	0	0	318	0
C3	0	0	0	536
C4	26	857	0	0

For the purpose of being thorough, I then tried using a standard cut off point of k = 4 for the trees, to agree with my earlier Principal Component Analysis which showed 4 clear groupings. Using single, average and complete linkage again, I found that average linkage would split the data into four distinct groups and again, comparing with the original gated populations this yielded a Rand Index of 0.9886 and an adjusted Rand index of 0.9756, the highest I had achieved yet. The Dendrogram for average linkage can be seen here, along with a red line indicating a potential cut off height to yield 4 clusters. Once more, we compare these resulting clusters to the original gated populations in the table below:



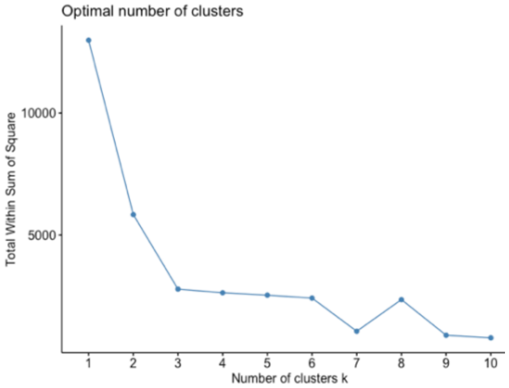
Cluster/Gate	G1	G2	G3	G4	
C1	2087	14	0	14	2115
C2	0	0	327	1	328
C3	0	0	0	548	548
C4	2	871	0	0	873
	2089	885	327	563	

As we can see, this again results in very similar groupings to the original gated populations. Using Average Linkage results in cleaner groupings as this measure is more robust to outliers than the single linkage used above. We can also say that using 4 clusters significantly improves this method's ability to recognize groupings within the dataset which correspond to the gated populations. Overall Hierarchical Clustering did a very good job of identifying subsets of the data which are similar to the identified gated populations.

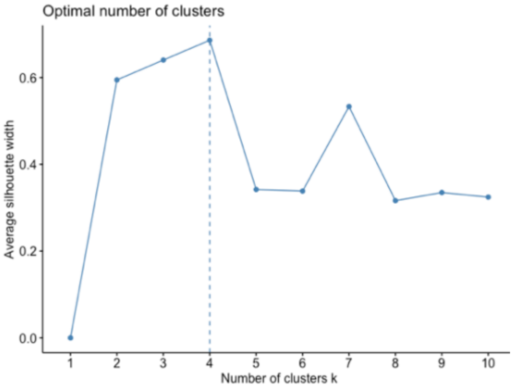
K-Means Clustering

The next unsupervised learning method that I will be employing is K-means clustering. Similar to Hierarchical, this method attempts to split the data into groups of similar observations. Rather than using a hierarchical process, it is iterative so at each stage we have a fixed number of clusters *k,* and we assign each observation to the cluster with the closest centroid. K-means also uses Euclidean distances, which suits our dataset as each of our variables are continuous.

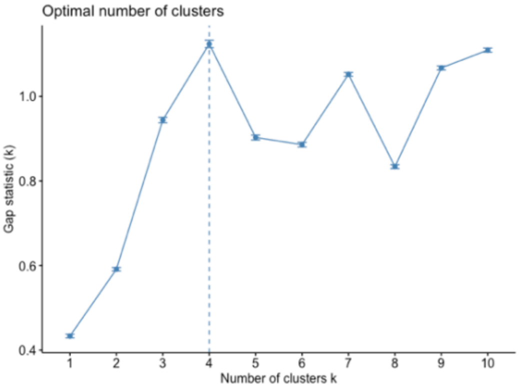
The first step is to determine the optimal number of clusters, or the optimal value for k. There are three methods we can use to do this, known as Within Sum of Squares method, the Silhouette method and the Gap Statistic. The graphs for these three methods can be seen below, and overall they point to an optimal k = 4. For the WSS graph, we look for a kink or an “elbow” in the graph. The Silhouette method we look to maximise the Average Silhouette Width, and the Gap Statistic we look to maximise the Gap Statistic.



WSS Method

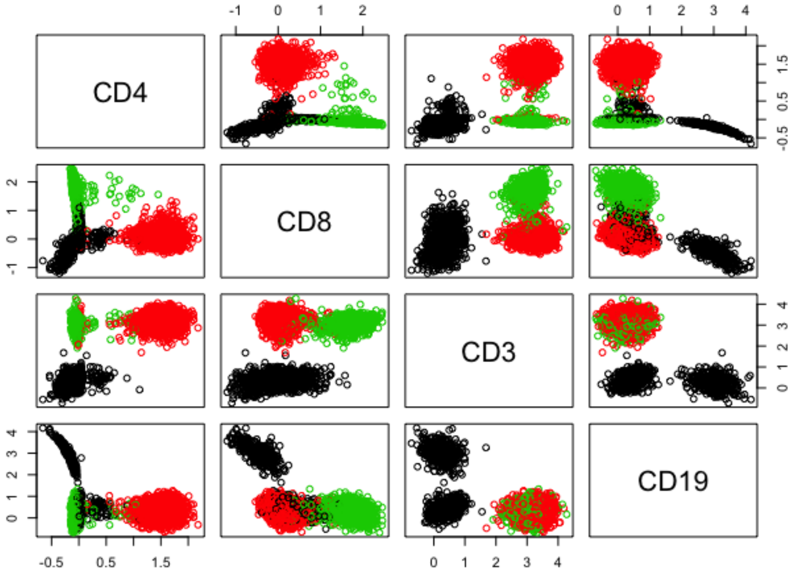


Silhouette Method

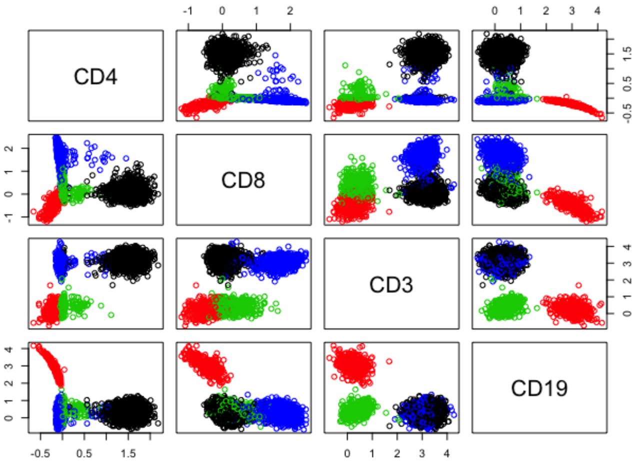


Gap Statistic

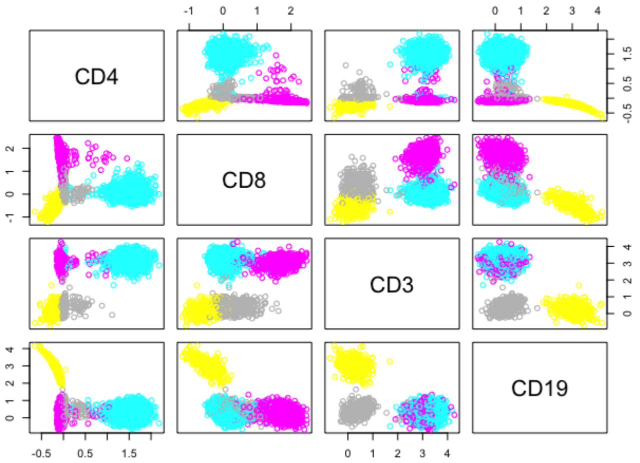
On inspection of the above graphs, we can see the WSS indicates 3 clusters and the other two indicate 4 clusters as the optimal number. I will use both 3 and 4 as my value for k and then compare the resulting clusters. Setting k = 3 firstly, yields three clusters of size 2090, 896 and 878. We can now run another pairs plot, this time coloured by the three clusters we have just created. What this shows us is that two of the clusters from our original pairs plot seem to have joined into one, the black grouping. For example in the CD3 v CD19 graph, the black cluster can be clearly split into two separate groupings, and similar could be done in most of the graphs. The Rand Index for these clusters versus the original groupings is 0.969 and adjusted Rand Index is 0.936. While this is a very high level of agreement, it is not as high as the Hierarchical clustering. We will now try setting k = 4 and see if this will be improved.



Using k = 4 and running the k-means algorithm yields 4 clusters of size 2086, 327, 553 and 898. This gives a Rand index of 0.994 and an adjusted Rand index of 0.987, the highest yet. If we look at the comparison pairs plots below, we can clearly see the similarity between our 4 new clusters, left, and our original gated populations, right. This is a great visualization of their resemblance, and demonstrates how accurately this method divides the dataset into groupings which are similar to the gated populations.



New Clusters



Original Groupings

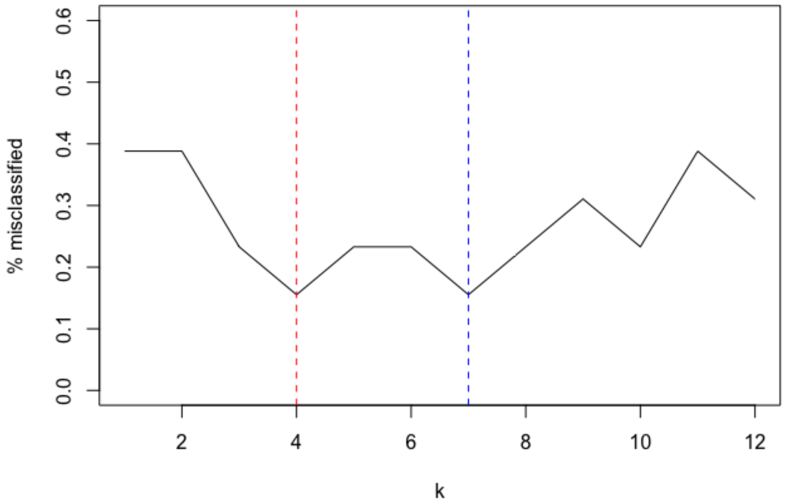
Supervised Learning Methods

For the next stage of my analysis, I will employ the use of Supervised learning methods to try to accurately predict which data is to be assigned to the identified gates. Supervised learning methods are defined by their use of labelled datasets. These labels (in this case the gate variable) are used to train or "supervise" algorithms into predicting outcomes accurately. The Supervised Learning Methods I will be using are K-Nearest Neighbours (KNN), Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA).

K-Nearest Neighbours (KNN)

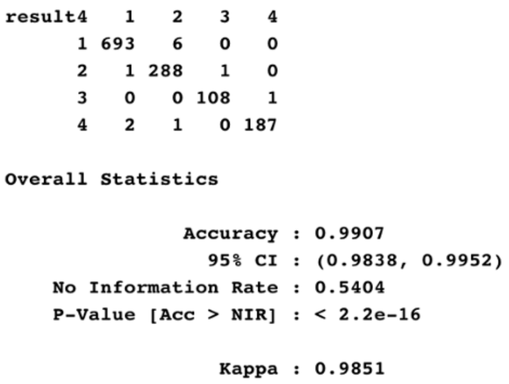
KNN is a non-parametric supervised learning method that will make use of both our dataset and the known gated populations in order to predict the "class" or gate of new observations. For this method, we split our data into three groups; training, test and validation. The training set is the "known" information which will be used to classify unlabelled points. The test set is the "new" data, points which we will be labelling. And the validation set consists of unlabelled points which we will use to calculate error.

For simplicity, I divided the gates equally among the three groups. So training set is made up of 697 observations from gate 1, 295 from gate 2, 109 from gate 3 and 188 from gate 4. Testing is made up of 696 from gate 1, 295 from gate 2, 109 from gate 3 and 188 from gate 4. And Validation is made up of 696 observations from gate 1, 295 from gate 2, 109 from gate 3 and 187 from gate 4. In order to select an optimal value for k, I ran the KNN algorithm for values of k between 1 and 50, and found the miscalculation rate for each k. The graph on the left shows the misclassifications plotted for each k from 1 to 12 (After 12 the graph went upwards rapidly so it is of little use). As we can see the values of k

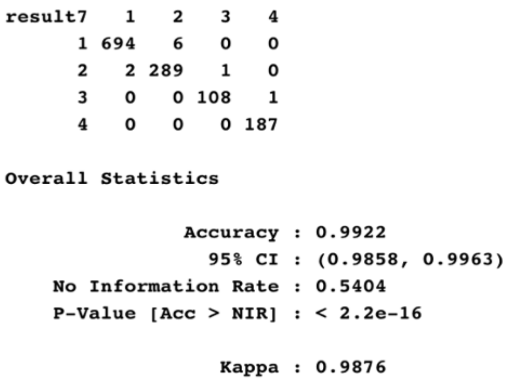


with the lowest miscalculation rates are 4 and 7 which have equal percentages of 15.52%. Hence, we will test both values of k and compare our results.

In order to test these values of k, we will re-run the KNN algorithm with the validation set. Firstly, setting k = 4 yields a misclassification rate of 0.93%, whereas k = 7 had a misclassification rate of 0.78%. Both of these are very low rates, indicating a very high degree of accuracy in the model. Below is the output of a confusion matrix for k = 4 and k = 7. As we can see, when k = 7 the predictions are slightly more accurate and so I would call this the optimal value for k. Overall, this method is extremely accurate in predicting the gate of an observation, and should only falsely predict about 0.78% of the time.



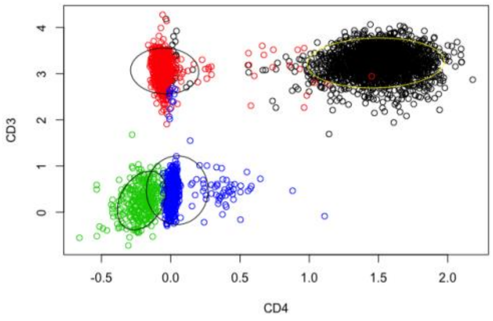
```
result4   1   2   3   4
      1 693   6   0   0
      2   1 288   1   0
      3   0   0 108   1
      4   2   1   0 187

Overall Statistics

               Accuracy : 0.9907
                 95% CI : (0.9838, 0.9952)
    No Information Rate : 0.5404
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.9851
```

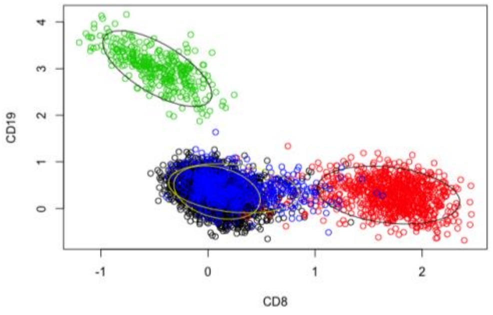


```
result7   1   2   3   4
      1 694   6   0   0
      2   2 289   1   0
      3   0   0 108   1
      4   0   0   0 187

Overall Statistics

               Accuracy : 0.9922
                 95% CI : (0.9858, 0.9963)
    No Information Rate : 0.5404
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.9876
```

Discriminant Analysis

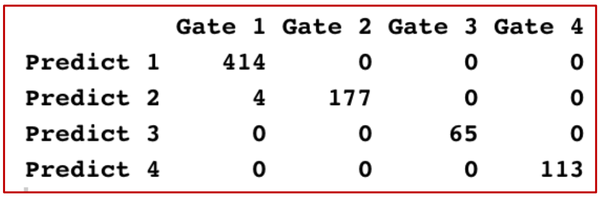
The final part of my analysis will use Discriminant Analysis. Discriminant Analysis is another classification method, similar to KNN, which reveals underlying structures in the dataset. It takes what we already know about existing data, the gate variable, and learns from this information, using it to predict what gate a new observation will go into. The two types of Discriminant Analysis that I will be consider are Linear (LDA) and Quadratic (QDA).





On examination of the plots over, showing the relationships between CD3 & CD4, and CD8 & CD19 respectively, we plot ellipses which are centred on the means of each grouping and shaped by their covariance matrices. The distinction between LDA and QDA is that LDA takes a pooled covariance, so that it utilises one shared covariance matrix, however QDA allows for two different covariances. In this case, as we can see from the overlaid ellipses, the covariances are not the same, for example in the first graph the black dots represent gate 1 and the yellow ellipse here is much wider than the other three ellipses in that graph. This indicates differing variances between variables and gates, and so LDA will most likely perform badly in this case. For this reason, I will only conduct and Quadratic Discriminant Analysis.

For QDA, similar to KNN, we will split up our data, however this time we do not require a validation set since



```
            Gate 1 Gate 2 Gate 3 Gate 4
Predict 1    414      0      0      0
Predict 2      4    177      0      0
Predict 3      0      0     65      0
Predict 4      0      0      0    113
```

observations in the test set will be assigned a probability and not an absolute assignment. Hence, we only require training and test sets. We will use an 80/20 split of the data and run our analysis. The table to the right is our output, which shows the predicted groupings versus actual for the test set. These predictions have 99.48% accuracy, only misclassifying 4 out of 773 observations (773 being 20% of 3864, our total). When I predicted prior probabilities using the test set, it generated probabilities for gates 1 to 4 of 0.54, 0.23, 0.08 and 0.15 respectively, which almost exactly mirrors the proportions of the overall dataset. The misclassification rate is only 0.52%, which is even more accurate then KNN and indicates that QDA is more accurate in predicting which data are assigned to the specified gated populations.

Conclusion

Throughout this report, I have analysed the Healthy Flow dataset. At first, I used charts, plots and other means to describe the data and visualise any relationships that may exist between variables.

I then used unsupervised learning methods to try to identify subsets of the data which were similar to the identified gated populations. I was able to do a very good job of this with all three methods that I employed, Principal Component Analysis, Hierarchical Clustering and K-Means Clustering. Four very distinct populations exist within this dataset and in particular, K-Means clustering did an excellent job of uncovering this with an adjusted Rand Index of 0.987, indicating extremely high agreement with the original gated populations.

Lastly, I used supervised learning methods to try to accurately predict which data are assigned to these specified gates. Both methods used – K-Nearest Neighbours and Quadratic Discriminant Analysis – also did a very good job of making accurate predictions, with misclassification rates of 0.78% and 0.52% respectively which are exceptionally low rates. Overall our outlined objectives were achieved successfully in this analysis.