Clustering and Randomised Algorithms

*Statistic for Computer Science*

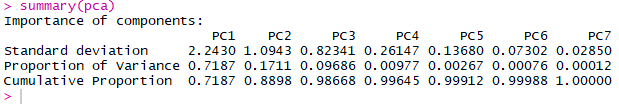
# Part 1

## Section 1

### task 1

1. PCA results using the first 7 attributes:





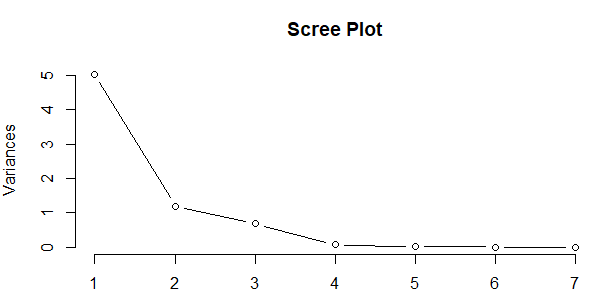
1. Eigenvalues for each of the attributes in descending order:



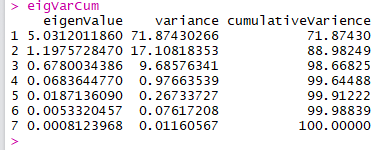
|  |  |
| --- | --- |
| **PC1** | 5.0312011860 |
| **PC2** | 1.1975728470 |
| **PC3** | 0.6780034386 |
| **PC4** | 0.0683644770 |
| **PC5** | 0.0187136090 |
| **PC6** | 0.0053320457 |
| **PC7** | 0.0008123968 |

1. Scree plot using the eigenvalues found above:





1. After performing the PCA the 7 original scaled attributes have been combined to form 4 leaner combinations or Principal Components. However, the first Principal Component is generally the most important as it explains most of the data, followed by Principal Component 2 then 3 and so on. Using the summary function for the PCA we get 3 values; the standard deviation, proportion of variance and the cumulative proportion. The standard deviation is the standard deviation of the data along a single principal component; a measure of the variability across that principal component. The proportion of variance represent the proportion of all the variability in the original data explained by the principal component for instance PC1 explains 71.87% of the data followed by PC2 which explains 17.11%, followed by PC3 which explains 9.69%. The cumulative proportion of PC1 explains 71.87% of the data, PC1 and PC2 combined explain 88.98%, PC1, PC2 and PC3 combined explain 98.67% of the data. This goes on until a 100% of the data is explained after combining all 7 principal components. All this is represented in the scree plot above. The table below shows the eigenvalue, percentage of variance and percentage of cumulative variance for each principal component.



From this table it is apparent that the first principal component is the best one as at explains the data the most from all 7 principal components.

The following table show the first 10 rows of the data set with the scores from the first 2 principal components in the last 2 columns.

## Section 2

### Task 1

Clustering is the process of partitioning a data set into homogeneous groups based on given features such that similar objects are kept in a group while dissimilar objects are placed in a different group. It is the most important unsupervised learning problem as it deals with finding structure in a collection of unlabelled data.

There are many clustering algorithms that can be used in any given scenario. The most popular ones include k-means, hierarchical clustering, fuzzy c-means and mixture of gaussians.

**K-means** is perhaps the most common algorithm used because it is a simple and easy way to classify a given data set through a certain number of clusters. The main idea is to define k centres, one for each cluster. These centres should be placed in a calculated way since different location cause different results. The better choice is to place them as far away as possible from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centre. When no point is pending, the first step is complete, and an early group age is done. At this point we re-calculate k new centroids as barycentre of the clusters resulting from the previous step. After we have these k new centroids, a new binding must be done between the same data set points and the nearest new centre. Here a loop is generated and as a result we notice that the k centres change their location step by step until no more changes are done. This is a randomised clustering algorithm as the centres are randomly selected.

K-means is Fast, robust and easier to understand. It is Relatively efficient and Gives best result when data set are distinct or well separated from each other. However, it requires specification of the number of clusters. it is not invariant to non-linear transformations therefore with different representation of data we get different results. The Euclidean distance measures can unequally weight underlying factors. This algorithm is Unable to handle noisy data and outliers and it fails for non-linear data set

**Hierarchical** clustering is an algorithm that builds a hierarchy of clusters. This algorithm starts with all the data points assigned to a cluster of their own. Then two nearest clusters are merged into the same cluster. The algorithm terminates when there is only a single cluster left. This is a non-randomised clustering algorithm. An advantage to hierarchical clustering is that it does not require information about the number of clusters that are needed. It is also easy to implement and gives best result in some cases. On the downside you can never undo what was previously done and a time complexity of at least O(n2 log n) is required (n being the number of data points). Moreover, no objective function is directly minimized and sometimes it is difficult to identify the correct number of clusters by the dendrogram.

Differences between the above algorithms:

* Hierarchical clustering can’t handle big data well unlike K Means clustering since the time complexity of K Means is linear (O(n)) while that of hierarchical clustering is quadratic (O(n2))
* Since K Means clustering starts with a random choice of clusters, the results produced by running the algorithm multiple times might differ. While results in Hierarchical clustering are reproducible
* K Means works well when the shape of the clusters is hyper spherical (2D circle, 3D sphere)
* K Means requires prior knowledge of K i.e. no. of clusters you want to divide your data into. But, you can stop at whatever number of clusters you find appropriate in hierarchical clustering by interpreting the dendrogram

The **fuzzy c-means** clustering algorithm works by assigning membership to each data point corresponding to each cluster centre based on distance between the cluster centre and the data point. The closer the data is to the cluster centre, the closer is its membership towards that particular cluster centre. Fuzzy c-means allows one piece of data to belong to two or more clusters. This method is frequently used in pattern recognition. Fuzzy c-means gives best result for overlapped data set and is comparatively better then k-means algorithm. Unlike k-means where data point must exclusively belong to one cluster centre here a data point is assigned membership to each cluster centre as a result of which a data point may belong to more than one cluster. Disadvantages to this algorithm include the fact that the number of clusters need to be specified. Also, with lower value of β (the termination criterion between [0, 1]) we get better results but at the expense of more number of iteration. The Euclidean distance measures can unequally weight underlying factors.

### Task 2

### Task 3

### Task 4

# part 2

## Section 1

### Task 1

### Task 2

### Task 3

## Section 2

### Task 1

### Task 2

# references