Digital Assignment II
A study on Clustering

Ishita Jaju
16BCE1059

1 The Dataset

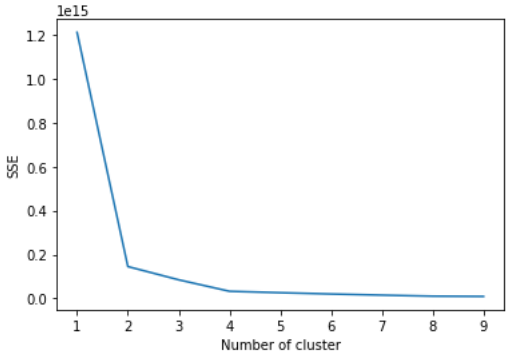
The dataset contains various values of different students in a college. There are no given labels in the dataset, it is an example of completely unsupervised learning and clustering analysis. It is called 'College Scorecard', so the results must group similar students together.

1.1 Preprocessing

The dataset initally contained 1725 columns with 7803 tuples, with plenty of null values. After removing them and filtering out the columns, we are left with 282 colums and 1319 row values.

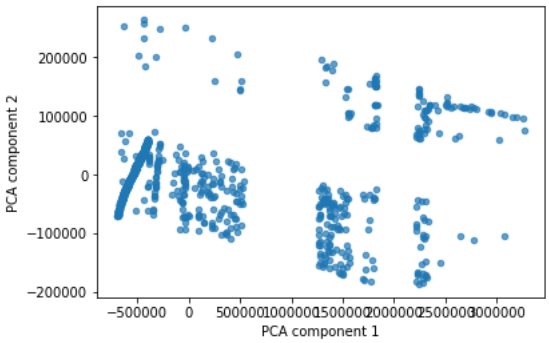
2 Ideal number of clusters using the Elbow Plot

The elbow plot with a filter of K-Means clustering gives us the number of clusters as '4'.



3 Visualization

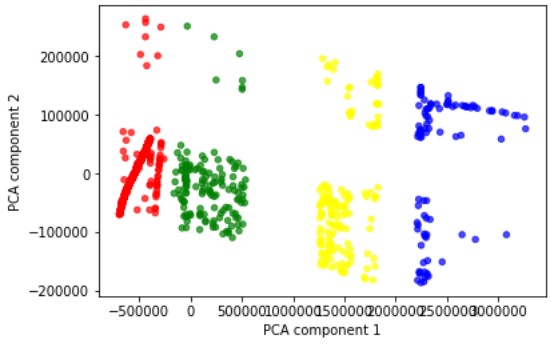
Lets visualize the data for easier understanding with the help of PCA, reducing it to 2 dimensions. It looks like there can be 4 distinct clusters, just like it was



concluded in the elbow plot.

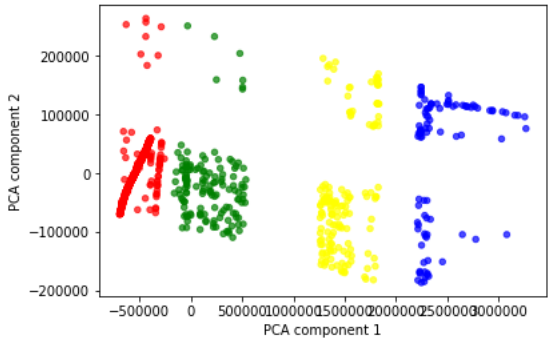
4 K-Means

After applying the K-Means algorithm, we get 4 clusters which separate distinctly like so:



5 AGNES

Agglomerative clustering gives similar results.



On comparing the results of K-Means with AGNES, we find that both give exactly the same results, as they give the tuples the same label.

```
print(kmeans.labels_)
print(clustering.labels_)
```

```
[3 0 0 ... 3 3 3]
[3 0 0 ... 3 3 3]
```

```
from sklearn.metrics import accuracy_score
accuracy_score(kmeans.labels_,clustering.labels_)
```

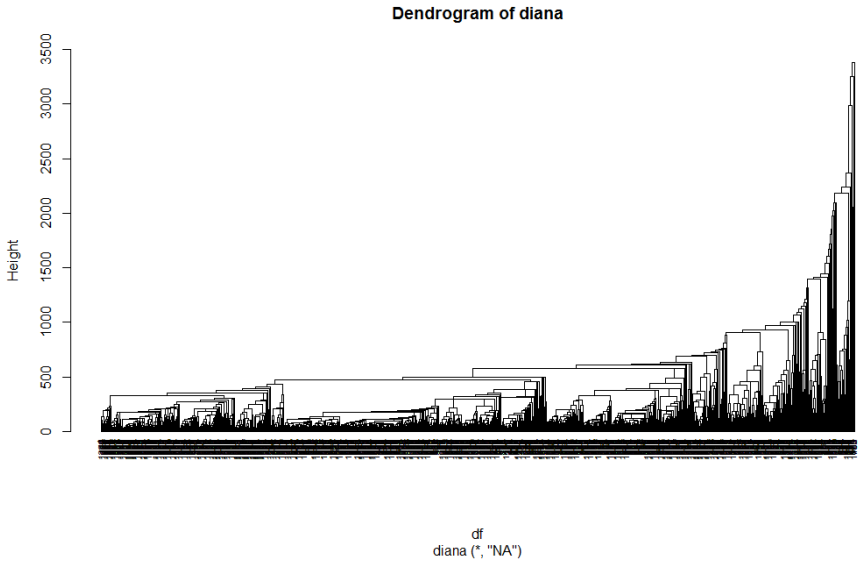
```
1.0
```

6 DIANA

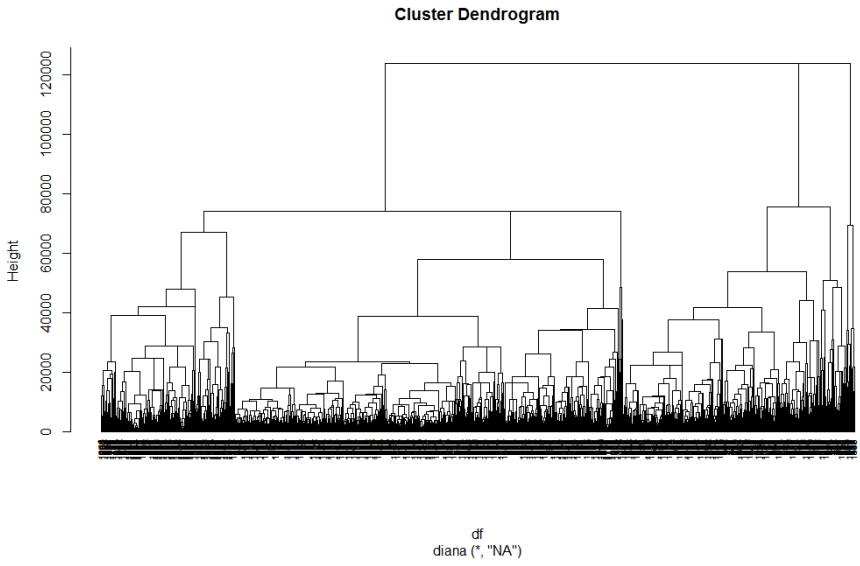
For divisive clustering, RStudio is used. Here, the metric is manhattan distance

```
> dv <- diana(df, metric="manhattan",stand=TRUE)
> pltree(dv, cex = 0.6, hang = -1,main = "Dendrogram of diana")
```

instead of the usual euclidean distance. On plotting this, we get the following dendogram:



On changing the distance metric back to Euclidean, the following dendogram is obtained:



```
> dv$dc
[1] 0.9564514
```

The metric dc is the divisive coefficient. It gives a numerical value of the amount of clustering structure found.

In the dendrograms displayed above, each leaf corresponds to one observation.

7 SOM

For SOM, we need a visualization. So we do PCA to reduce to 3 dimensions and then run SOM on it.

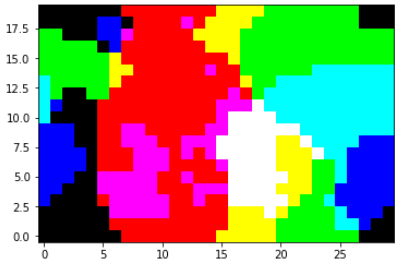
```
pca3 = PCA(n_components=3)
pca3.fit(df)
T3 = pca3.transform(df)
T3 = pd.DataFrame(T3)
```

```
somt = SimpleSOMMapper((20, 30), 400, learning_rate=0.05)
somt.train(T3.values)
```

```
plt.imshow(somt.K, origin='lower')
```

```
Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0..255] for in
```

```
<matplotlib.image.AxesImage at 0x7f540fd9aa20>
```



8 Inference

K-means clustering is a type of unsupervised learning, which is used with unlabeled data. Rather than defining groups before looking at the data, clustering helps find analyze the groups that have formed organically. K is the number of clusters. While choosing K, the user needs to run the K-means clustering algorithm for a range of K values and compare the results. In general, there is no method for determining exact value of K, but an accurate estimate can be found by methods like the elbow method, as above. This works on the basis that the metric to compare results across different values for K is the mean distance between data points and their cluster centroid. since increasing the number of clusters will always reduce the distance to data points, increasing K will always decrease this metric.

Agglomerative clustering works in a bottom-up manner. Each object is initially considered as a single-element cluster in the leaf. At each step, the clusters that are the most similar are combined into a new bigger cluster, whic are nodes. This procedure is iterated until all points are member of just one single big cluster which is the root). The result is a tree which can be plotted as a dendrogram. On the other hand, divisive hierarchical clustering works in a top-down manner. It is an inverse order of AGNES. It begins with the root, in which all objects are included in a single cluster. At each step of iteration, the most heterogeneous cluster is divided into two. The process is iterated until all objects are in their own cluster.

From our experiments, we find that agglomerative clustering is good at identifying small clusters and divisive hierarchical clustering is good at identifying large clusters.

The dendograms formed by the agglomerative and divise clustering can tell a lot of things. The height of the cut to the dendrogram controls the number of clusters obtained. It plays the same role as the k in k-means clustering.

Self Organizing Maps are helpful in uncovering categories in large datasets. It uses neurons and through multiple iterations, neurons on the grid will gradually cluster around areas with high density of data points. So these areas have many neurons. The plot that we got above is in the form of a heat map. The red areas signify greater concentration of data points (hotter) and neurons and the blue areas have few (colder).