Marmara University Faculty of Engineering

CSE4062 – Data Science, Spring2020

Group7

"DRIVER DROWSINESS DETECTION"

Delivery #5 - Report

CSE	Mahmut AKTAŞ	aktasmahmut97@gmail.com	150115010
CSE	Mustafa Abdullah HAKKOZ	mustafa.hakkoz@gmail.com	150117509
ME	Ozan Berke YABAR	ozanberkeyabar@gmail.com	150416822
MME	Ece HARPUTLU	harputlue@gmail.com	150515038
EE	Nurettin ABACI	abacinurettin@gmail.com	150715035

Submitted to: Assoc. Prof. Murat Can Ganiz

- 06.05.2020 -

DESCRIPTIVE ANALYSIS

Data Setup

After loading data, we apply PCA to our data so we reduced feature number to 9 to 3 for visualization purposes. So, we named our new 3 features as “PCA_feature1”, “PCA_feature2” and “PCA_feature3”.

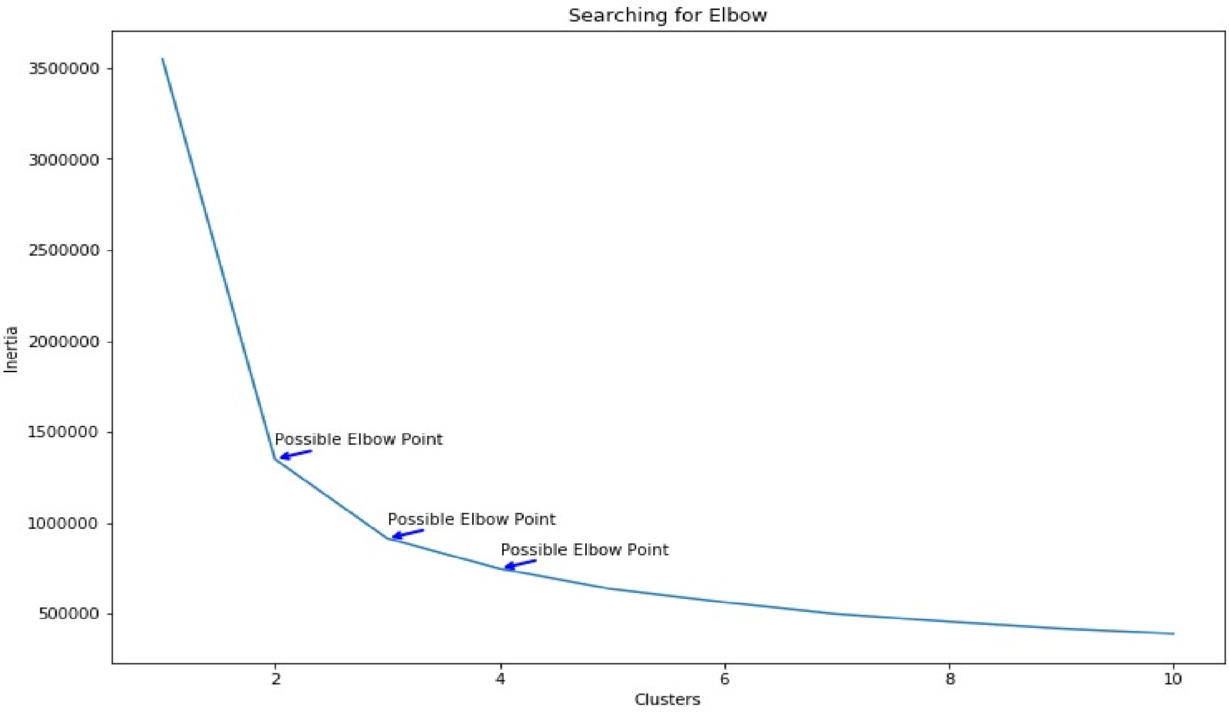
We choose 3 different clustering models of Sklean: **KMeans, AgglomerativeClustering and DBSCAN**. Kmeans runs without any problem on our dataset which has a size of (610K x 3) but other two models give memory errors. So, we use resample method to choose random (and stratified) 10K sample to work on manageable data.

Part – 1

We run 3 clustering models, tried different methods for finding best parameters and draw a table for statistics for this part.

K-Means

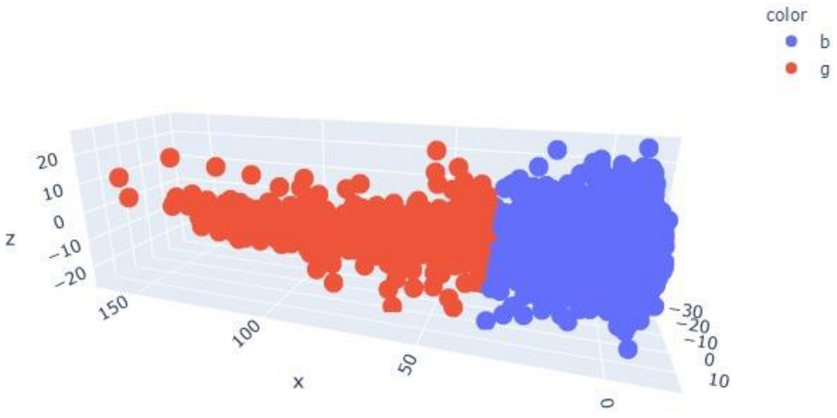
"inertia_" attribute provides sum of squared distances of samples to their closest cluster center. We can plot it and decide on the parameter "n_clusters" by using elbow method.



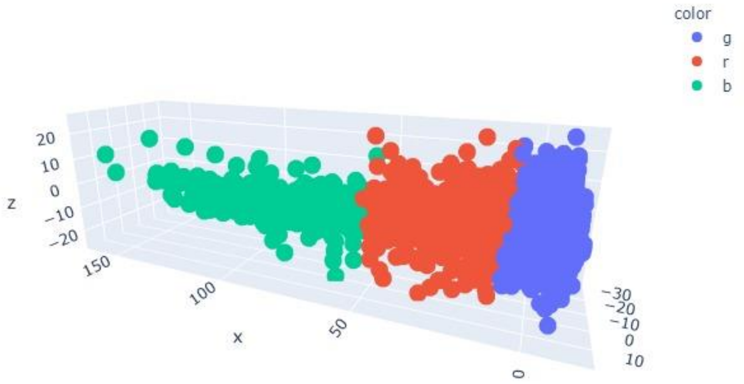
We tried possible elbow points by creating plots. We used **plotly** library for 3D plotting.

Cluster Plots

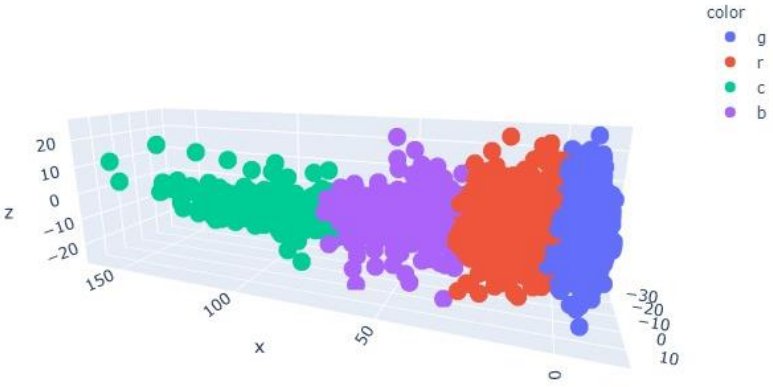
2 Clusters



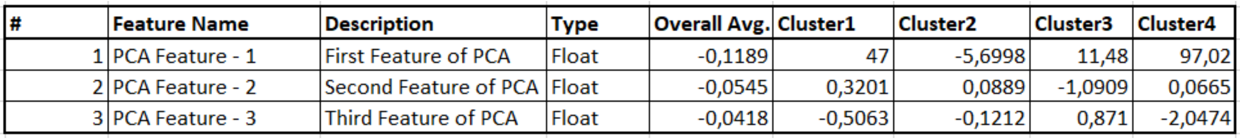
3 Clusters



4 Clusters



It seems like 4 clusters work best. Let's see our table for K-Means:

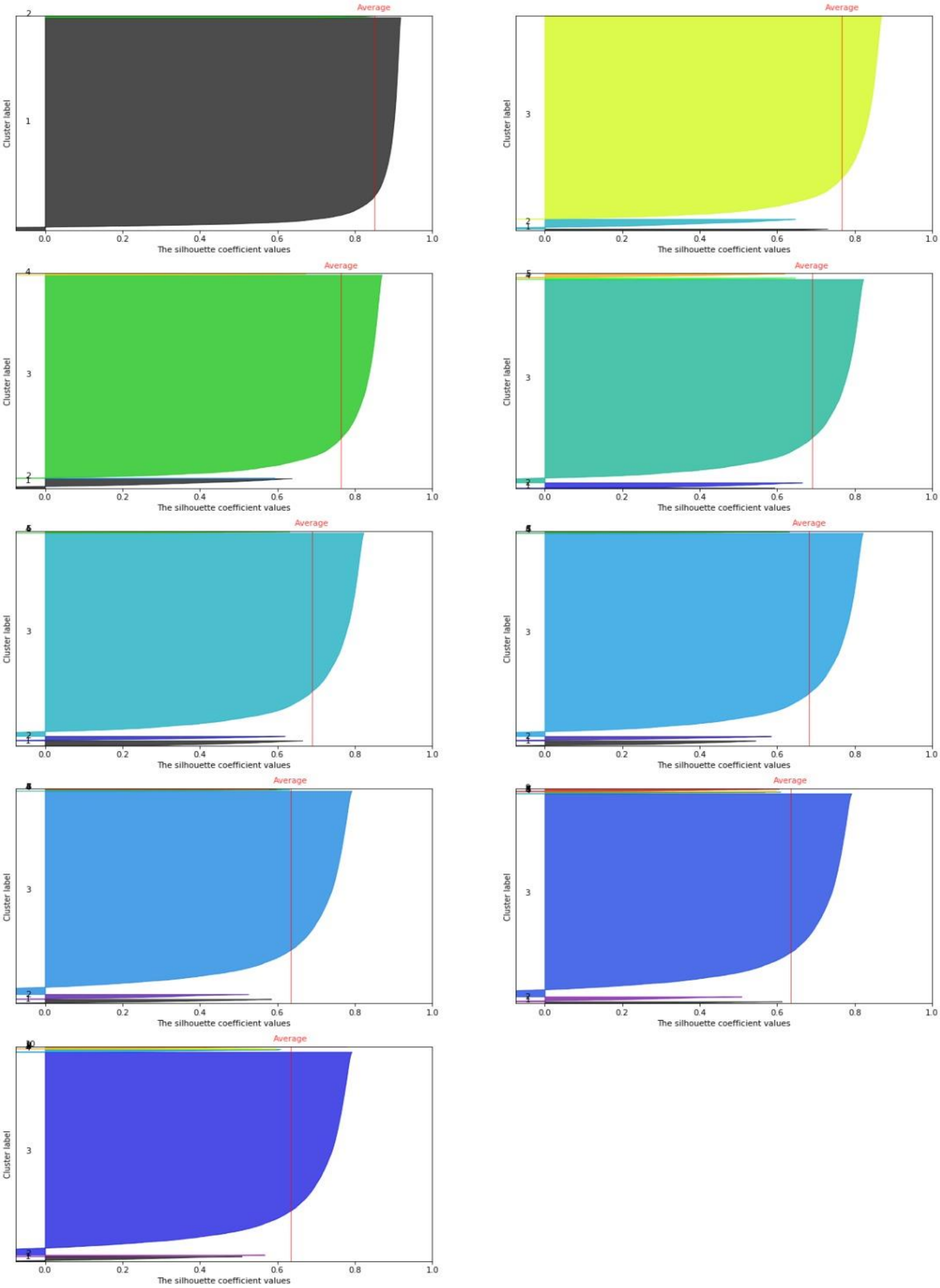


#	Feature Name	Description	Type	Overall Avg.	Cluster1	Cluster2	Cluster3	Cluster4
1	PCA Feature - 1	First Feature of PCA	Float	-0,1189	47	-5,6998	11,48	97,02
2	PCA Feature - 2	Second Feature of PCA	Float	-0,0545	0,3201	0,0889	-1,0909	0,0665
3	PCA Feature - 3	Third Feature of PCA	Float	-0,0418	-0,5063	-0,1212	0,871	-2,0474

AGNES

There's no inertia (Sum of squared distances of samples to their closest cluster center) attribute of AgglomerativeClustering class so we used silhouette coefficient (best:1, worst:-1) to select cluster number of AGNES.

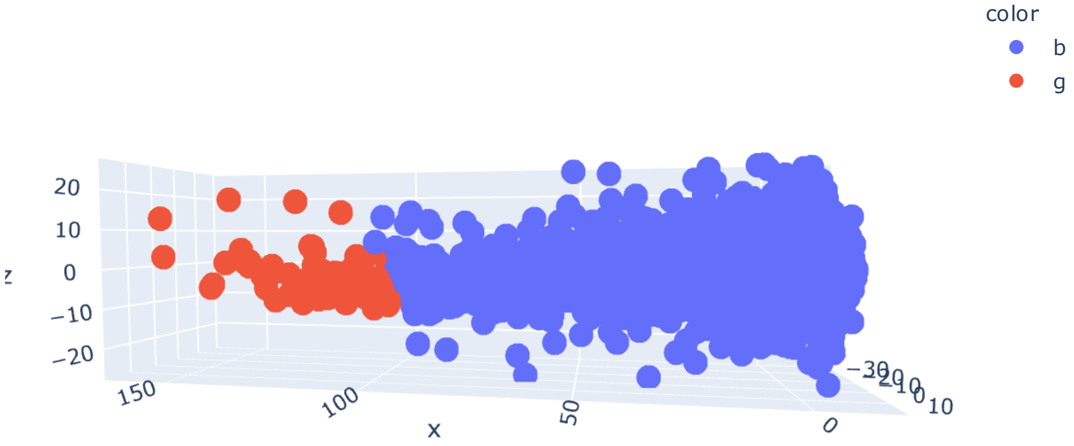
Silhouette Plots



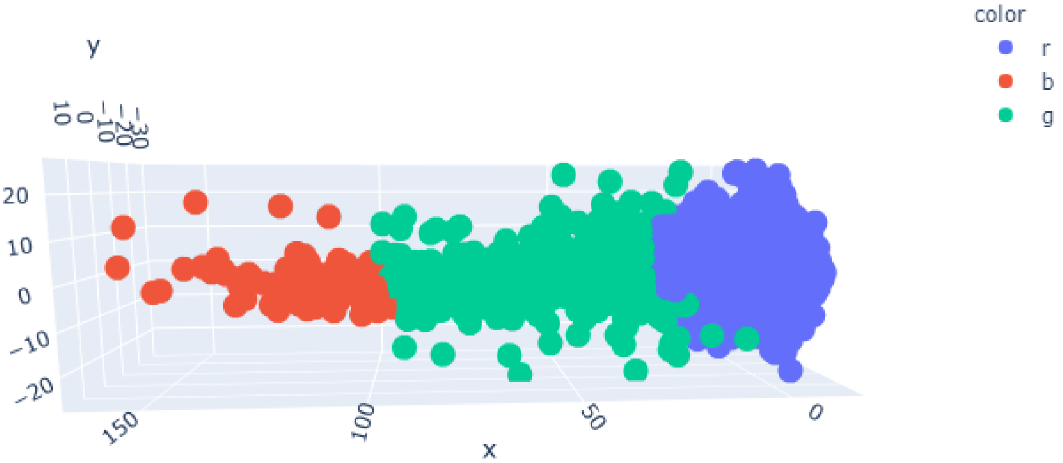
It seems like none of the plots gave a good result in the manner of cluster sizes but we choose the models with 2, 3 or 4 clusters to plot and to choose best one manually.

Cluster Plots

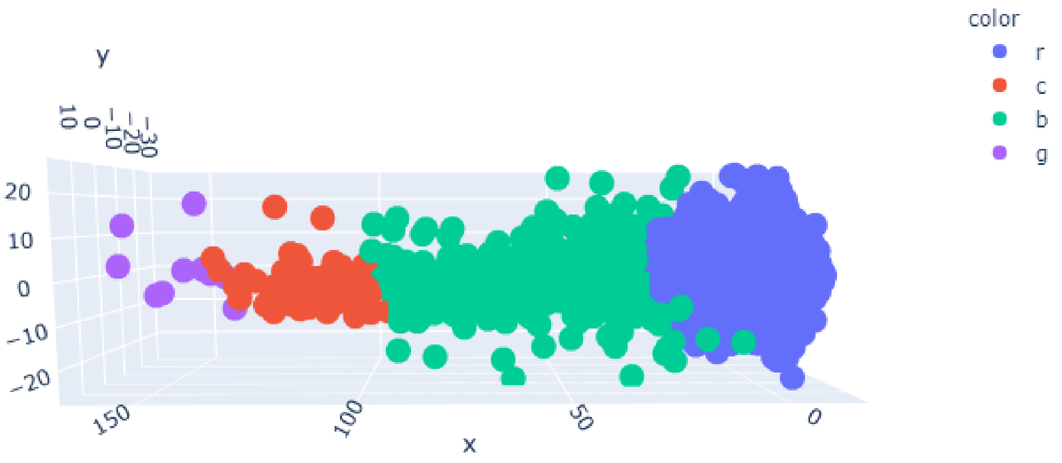
2 Cluster



3 Cluster



4 Cluster



It seems like 3 is the best amongst them (depending on the significant silhouette values). That's why we choose to work with n_clusters = 3.

Here is our table for Agnes:

#	Feature Name	Description	Type	Overall Avg.	Cluster1	Cluster2	Cluster3
1	PCA Feature - 1	First Feature of PCA	Float	-0,1189	117	52,72	-3,5779
2	PCA Feature - 2	Second Feature of PCA	Float	-0,0545	-1,6655	0,2685	-0,0567
3	PCA Feature - 3	Third Feature of PCA	Float	-0,0418	-2,1689	-1,2396	0,0323

DBSCAN

The maximum distance between two samples for one to be considered as in the neighborhood of the other (default: 0.5).

min_samples: The number of samples (or total weight) in a neighborhood for a point to be considered as a core point. This includes the point itself (default: 5).

Cluster Plots

2 Clusters

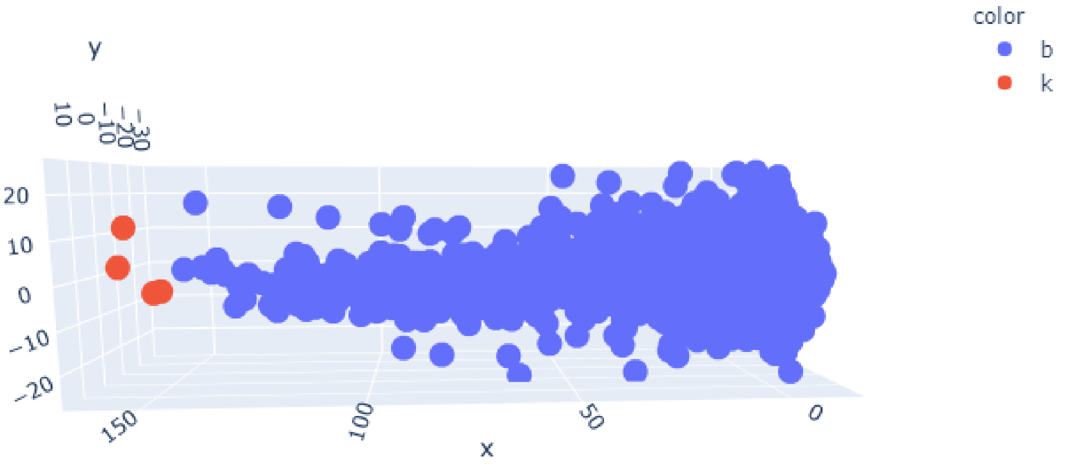
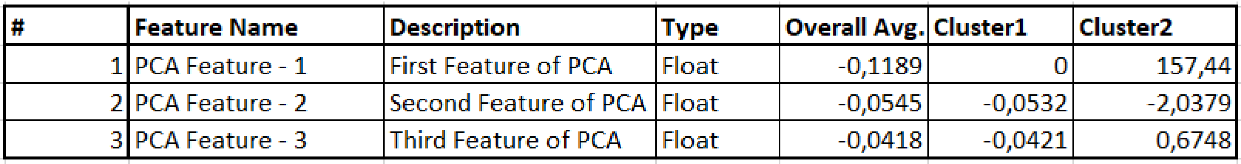


Table for DBSCAN method:

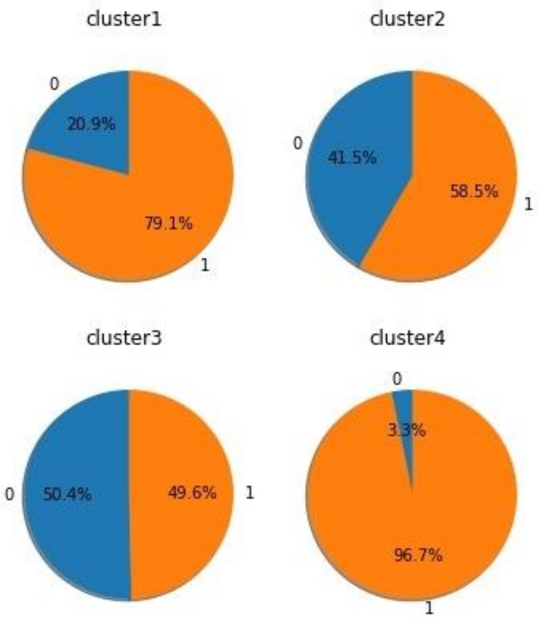


#	Feature Name	Description	Type	Overall Avg.	Cluster1	Cluster2
1	PCA Feature - 1	First Feature of PCA	Float	-0,1189	0	157,44
2	PCA Feature - 2	Second Feature of PCA	Float	-0,0545	-0,0532	-2,0379
3	PCA Feature - 3	Third Feature of PCA	Float	-0,0418	-0,0421	0,6748

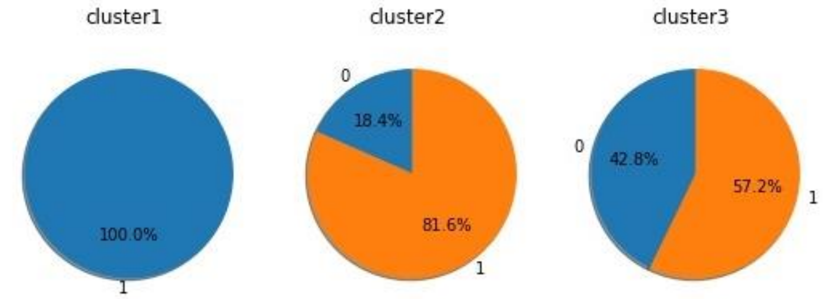
PART – 2

For this part we plot the pie charts for class distribution of each clustering models.

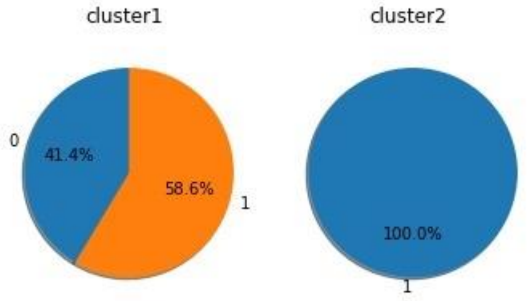
K-Means with 4 clusters



Agnes with 3 clusters

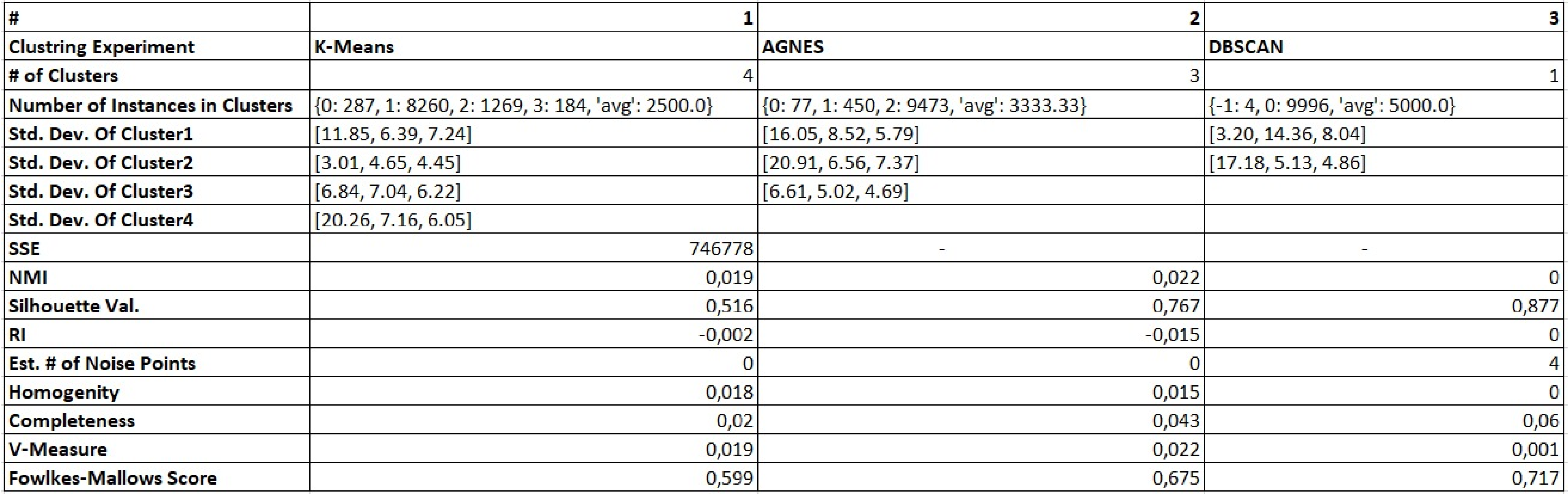


DBSCAN with 2 clusters



PART – 3

For this part, we draw a table showing evaluation metrics.



#	1	2	3
Clustring Experiment	K-Means	AGNES	DBSCAN
# of Clusters	4	3	1
Number of Instances in Clusters	{0: 287, 1: 8260, 2: 1269, 3: 184, 'avg': 2500.0}	{0: 77, 1: 450, 2: 9473, 'avg': 3333.33}	{-1: 4, 0: 9996, 'avg': 5000.0}
Std. Dev. Of Cluster1	[11.85, 6.39, 7.24]	[16.05, 8.52, 5.79]	[3.20, 14.36, 8.04]
Std. Dev. Of Cluster2	[3.01, 4.65, 4.45]	[20.91, 6.56, 7.37]	[17.18, 5.13, 4.86]
Std. Dev. Of Cluster3	[6.84, 7.04, 6.22]	[6.61, 5.02, 4.69]	
Std. Dev. Of Cluster4	[20.26, 7.16, 6.05]		
SSE	746778	-	-
NMI	0,019	0,022	0
Silhouette Val.	0,516	0,767	0,877
RI	-0,002	-0,015	0
Est. # of Noise Points	0	0	4
Homogenity	0,018	0,015	0
Completeness	0,02	0,043	0,06
V-Measure	0,019	0,022	0,001
Fowlkes-Mallows Score	0,599	0,675	0,717

CONCLUSION

We compared best of 3 models by using several metrics:

- Estimated number of clusters
- Number of instances in cluster
- Estimated number of noise points
- Standard deviation of clusters
- Homogeneity: For perfect clustering, each cluster contains only members of a single class.
- Completeness: For perfect clustering, all members of a given class are assigned to the same cluster.
- V-measure: Harmonic mean of Homogeneity and Completeness.
- Adjusted Rand Index: Given the knowledge of the ground truth class, the adjusted Rand index is a function that measures the similarity of the two assignments, ignoring permutations and with chance normalization.
- Adjusted Mutual Information: Given the knowledge of the ground truth class, the Mutual Information is a function that measures the agreement of the two assignments, ignoring permutations.
- Fowlkes-Mallows score: Geometric mean of precision and recall.
- Silhouette Coefficient: If the ground truth labels are not known, the Silhouette Coefficient is calculated using the mean intra-cluster distance (a) and the mean nearest-cluster distance (b) for each sample by (b - a) / max(a, b)
- Inertia (SSE): Inertia measures the internal cluster sum of squares (sum of squares is the sum of all residuals). Inertia is utilized to measure how related clusters are amongst themselves, the lower the inertia score the better. However, it is important to note that inertia heavily relies on the assumption that the clusters are convex (of spherical shape). DBSCAN and AGNES does not necessarily divide data into spherical clusters, therefore inertia is not a good metric to use for evaluating DBSCAN and AGNES models (which is why I did not include inertia in the code above). Inertia is more often used in other clustering methods, such as K-means clustering. Thus, inertia_ attribute is only provided for K-means in SKLearn.

In our results, only metrics don't require ground-truth labels are Silhouette Coefficient and inertia. For Silhouette Coefficient, DBSCAN takes ahead, then comes AGNES and followed by K-MEAN. Its results are correlated with Fowlkes-Mallows score although they require different inputs.

But from the perspective of Homogeneity and Completeness principles, AGNES performs best. Following is K-MEANS and DBSCAN. The results are similar for Mutual

Information.

And finally, for Rand Index, AGNES performs best again. Followings are K-MEANS and DBSCAN.

But one important note, for best parameters of DBSCAN, the model produces only 1 cluster so most of metrics generates null values for it. Thus, it’s better not taking DBSCAN into consideration in the evaluation phase.