ASSIGNMENT 4

BUAN 6341: Applied Machine Learning

Madhav Sampath Gubbala (MSG170330)

Introduction:

The goal here is to implement K-means and Expectation Maximization. This report includes data analysis, model building and experimentation by tuning the parameters. Through the experimentation the best model is identified.

In this assignment we use 2 data sets, the description of the respective data sets is given below:

GPU Run Time:

This data set measures the running time of a matrix-matrix product A*B = C, where all matrices have size 2048 x 2048, using a parameterizable SGEMM GPU kernel with 241600 possible parameter combinations.
This dataset is used for implementing ANN and K Nearest Neighbors which is downloaded from UCI Machine Learning repository. The details about the dataset are mentioned below:

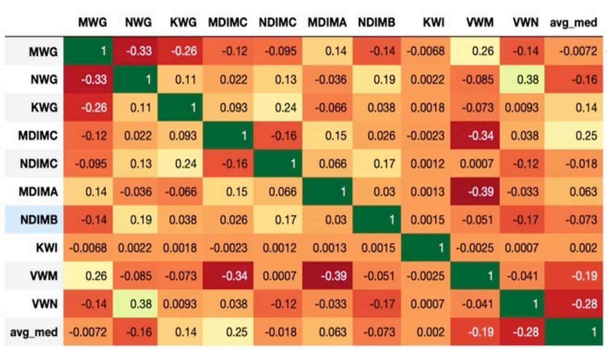
- Dataset consists of 241600 observations on 14 variables.
- Dependent variable is the average of Run1 (ms), Run2 (ms), Run3 (ms), Run4 (ms) columns which is 'avg_med'.
- Data split is done by 70:30 ratio for training and testing data set with no missing values.

Mobile Price Classification:

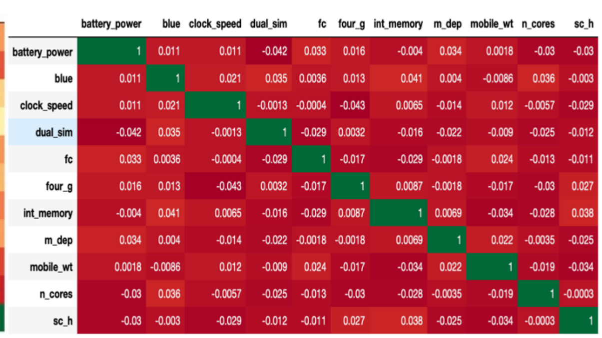
Link: https://www.kaggle.com/iabhishekofficial/mobile-price-classification
This data set measures the Price of the mobile phone with different features with 2500 possible parameter combinations. The price of data set is classified into two factors.
This dataset is used for implementing the ANN and K Nearest Neighbors algorithms is downloaded from Kaggle. The details about the dataset are mentioned below:

- Dataset consists of 2500 observations on 21 variables
- Dependent variable is the price of the mobile phone.
- Data split is done by 70:30 ratio for training and testing data set with no missing values.

<u>Correlation:</u> In SVM and Decision Tree algorithms correlation is important factor. For SVM, the effect is similar to that of multicollinearity in linear regression for linear kernel. For decision trees, no assumptions will be made by the algorithm on relation between features. If highly correlated information gain will be less.



	MWG	NWG	KWG	MDIMC	NDIMC	MDIMA	NDIMB	KWI	VWM	VWN	avg_med
MWG	1	-0.33	-0.26	-0.12	-0.095	0.14	-0.14	-0.0068	0.26	-0.14	-0.0072
NWG	-0.33	1	0.11	0.022	0.13	-0.036	0.19	0.0022	-0.085	0.38	-0.16
KWG	-0.26	0.11	1	0.093	0.24	-0.066	0.038	0.0018	-0.073	0.0093	0.14
MDIMC	-0.12	0.022	0.093	1	-0.16	0.15	0.026	-0.0023	-0.34	0.038	0.25
NDIMC	-0.095	0.13	0.24	-0.16	1	0.066	0.17	0.0012	0.0007	-0.12	-0.018
MDIMA	0.14	-0.036	-0.066	0.15	0.066	1	0.03	0.0013	-0.39	-0.033	0.063
NDIMB	-0.14	0.19	0.038	0.026	0.17	0.03	1	0.0015	-0.051	-0.17	-0.073
KWI	-0.0068	0.0022	0.0018	-0.0023	0.0012	0.0013	0.0015	1	-0.0025	0.0007	0.002
VWM	0.26	-0.085	-0.073	-0.34	0.0007	-0.39	-0.051	-0.0025	1	-0.041	-0.19
VWN	-0.14	0.38	0.0093	0.038	-0.12	-0.033	-0.17	0.0007	-0.041	1	-0.28
avg_med	-0.0072	-0.16	0.14	0.25	-0.018	0.063	-0.073	0.002	-0.19	-0.28	1

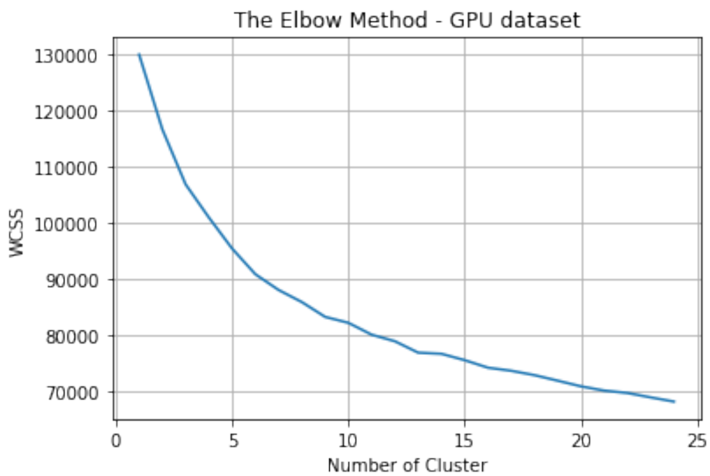


	battery_power	blue	clock_speed	dual_sim	fc	four_g	int_memory	m_dep	mobile_wt	n_cores	sc_h
battery_power	1	0.011	0.011	-0.042	0.033	0.016	-0.004	0.034	0.0018	-0.03	-0.03
blue	0.011	1	0.021	0.035	0.0036	0.013	0.041	0.004	-0.0086	0.036	-0.003
clock_speed	0.011	0.021	1	-0.0013	-0.0004	-0.043	0.0065	-0.014	0.012	-0.0057	-0.029
dual_sim	-0.042	0.035	-0.0013	1	-0.029	0.0032	-0.016	-0.022	-0.009	-0.025	-0.012
fc	0.033	0.0036	-0.0004	-0.029	1	-0.017	-0.029	-0.0018	0.024	-0.013	-0.011
four_g	0.016	0.013	-0.043	0.0032	-0.017	1	0.0087	-0.0018	-0.017	-0.03	0.027
int_memory	-0.004	0.041	0.0065	-0.016	-0.029	0.0087	1	0.0069	-0.034	-0.028	0.038
m_dep	0.034	0.004	-0.014	-0.022	-0.0018	-0.0018	0.0069	1	0.022	-0.0035	-0.025
mobile_wt	0.0018	-0.0086	0.012	-0.009	0.024	-0.017	-0.034	0.022	1	-0.019	-0.034
n_cores	-0.03	0.036	-0.0057	-0.025	-0.013	-0.03	-0.028	-0.0035	-0.019	1	-0.0003
sc_h	-0.03	-0.003	-0.029	-0.012	-0.011	0.027	0.038	-0.025	-0.034	-0.0003	1

Task 1:

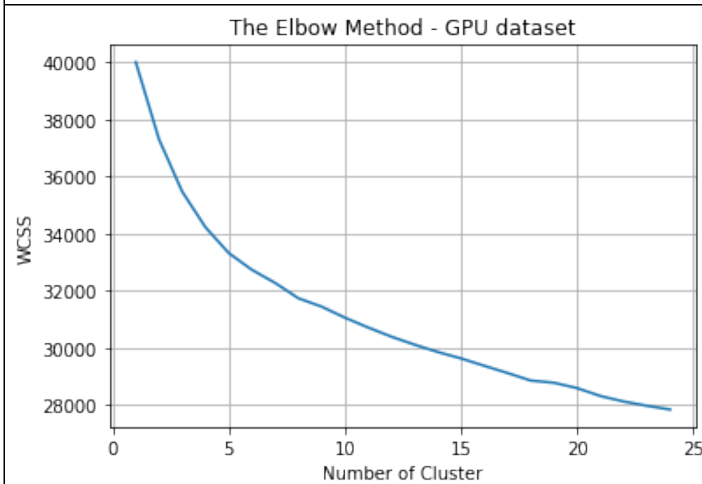
Run the clustering algorithms on your datasets and describe your observations (with plots).

K-Means Clustering:



GPU dataset:

- From the plot we could observe that the elbow shape is seen at cluster 10.
- After 10 the error doesn't seem to decrease significantly. let's take the number of clusters to be 10



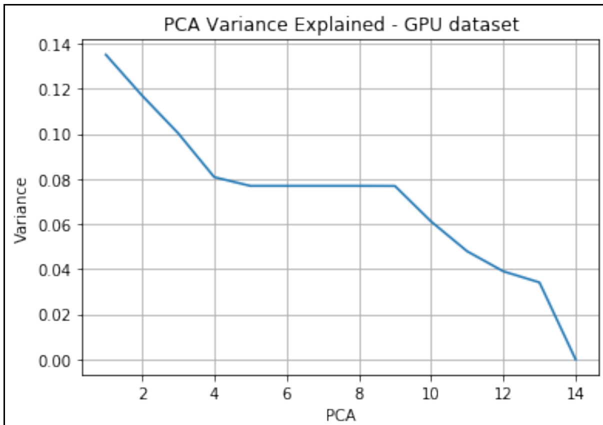
Mobile Price Dataset:

- Similarly, we can observe the elbow curve at clusters = 5 since there is steep decrease in WCSS from 40000 to 33000.
- Hence optimal number of clusters = 5

Task 2:

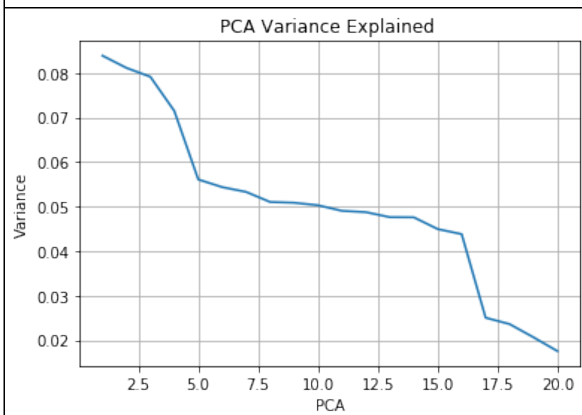
Apply the dimensionality reduction algorithms on your datasets and describe your observations (with plots).

Dimension Reduction (PCA)

<table>
<tr>
<td>


</td>
<td>

I implemented PCA as part of the Dimensionality Reduction concept.
We could see that the variance of the dataset explained by the 1st component estimates to about 15%.
10 components explain most of the variance in the dataset.
I choose first 10 components to pass it to the K – Means algorithm.

</td>
</tr>
<tr>
<td>


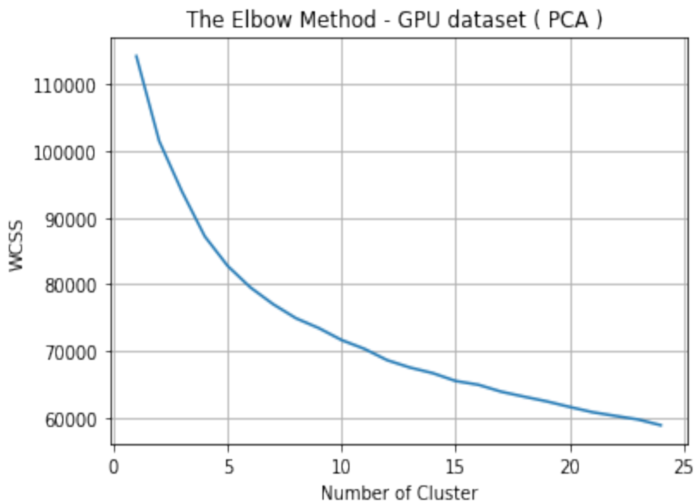
</td>
<td>
<u>Mobile Price Dataset:</u>

We can observe from the plot that 15 PCs can explain around 85% variance in the data.
Hence I chose 15 PCs for modelling and excluding 5 – dimension reduction

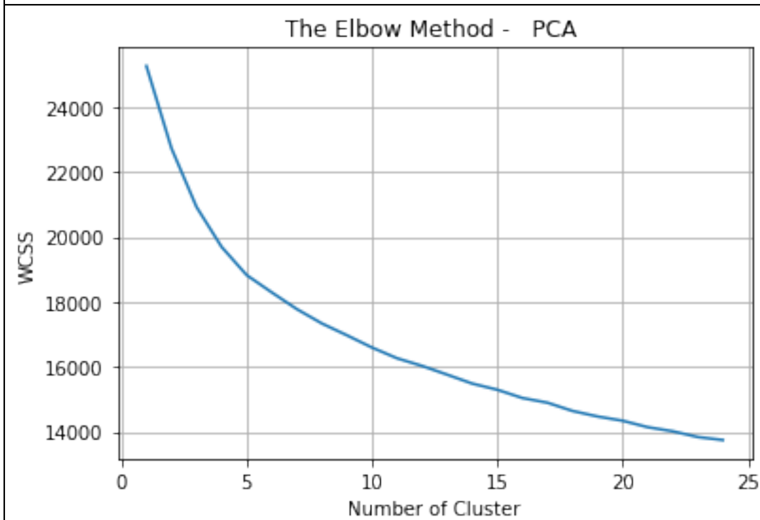
</td>
</tr>
</table>

Task 3:

Run the clustering algorithms again, this time after applying dimensionality reduction. Describe the difference compared to previous experimentation (with plots).



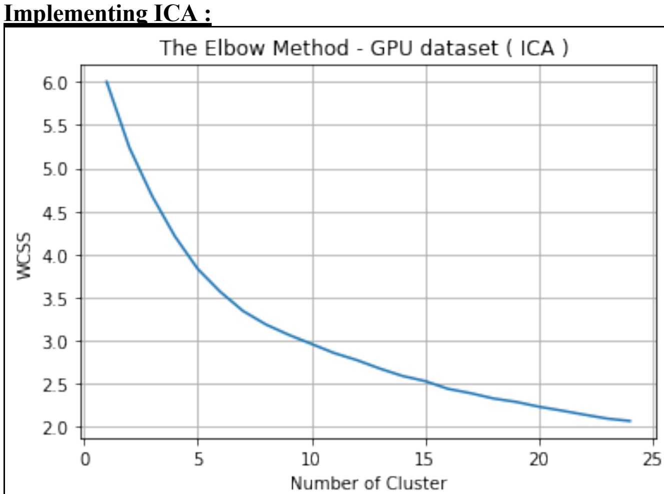
- After implementing the PCA and reduced the components to a total of 10 , reran the K-Means to find the number of clusters required.
- The difference of 10000 WCSS spotted between with PCA and without PCA.
- One observed difference is that we could find reduced **Within Cluster Sum of Squares** with PCA.



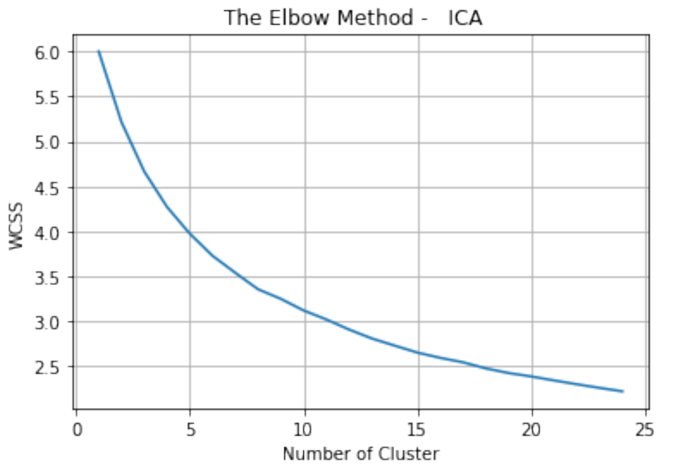
Mobile Price Dataset:

- Similarly, we can see that the curve flattens and elbow curve can be observed at k=5
- Hence the optimal K = 5 which is similar to the model run without applying PCA
- Only difference here is WCSS is much lower compared to previous model

Implementing ICA :



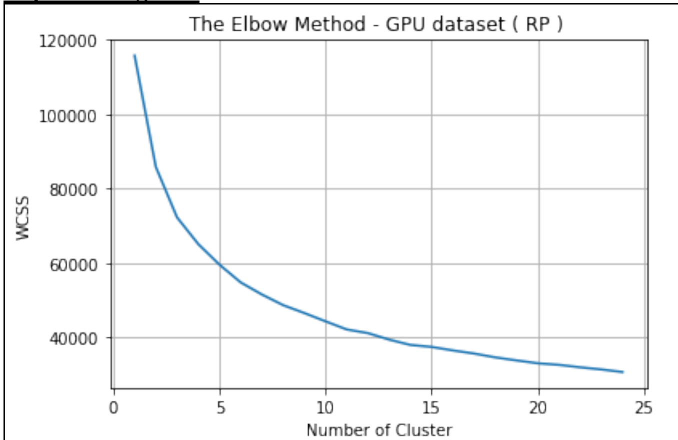
- we set the n_components to be 12 and have developed the ICA to extract the features. which gives the best mutually independence.
- WCSS has the value of 3 for 10 cluster configuration after the implementation of ICA.



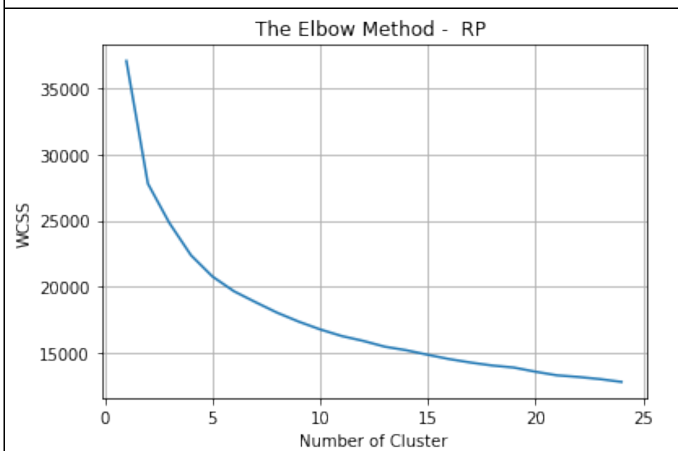
Mobile Price Dataset:

- Same can be observed in this graph as well.
- Optimal K can be found at 5 with WCSS of 4

Implementing RP :



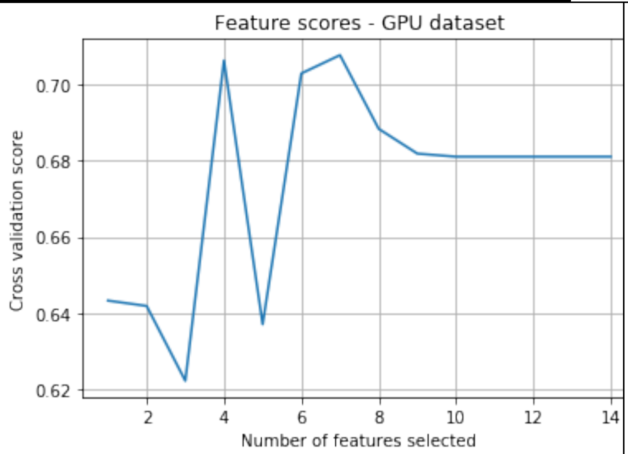
- After implementing the Randomized Projections WCSS value for cluster configuration 10 is 50,000 .
- Without this implementation , the cluster 10 WCSS value was found to be 80,000 .
- The WCSS for ICA implementation is low and this defines better cluster classification .

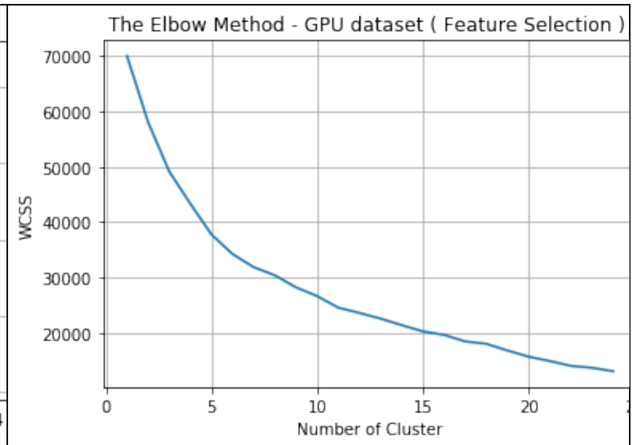


Mobile Price Dataset:

- Theoretically, Randomized projections provides compact clusters.
- Same can observed from WCSS values which are lower than other techniques.
- Here optimal K is also found at 5

Implementing Feature Selection (using Decision Tree):





The max value of Cross validation score can be observated at number of features = 7

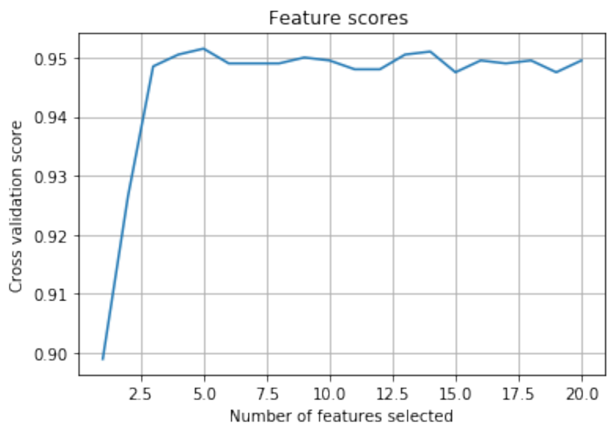
Therefore selected 7 most important features and re-run the k-means clustering algorithm for optimal clusters

From the above plot, the optimal number of clusters is still same but WCSS is decreased to 35000.

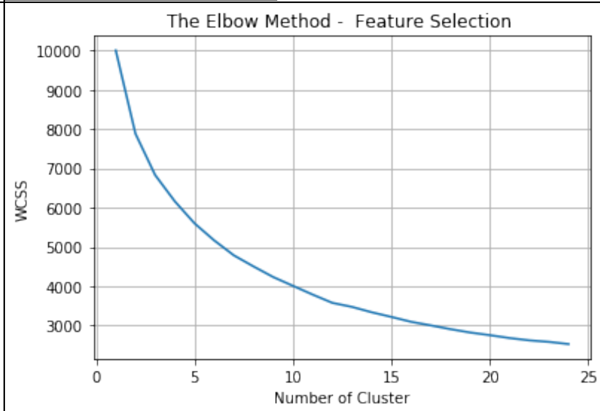
Therefore, feature selection has created compact clusters.

Mobile Price Dataset:

Mobile Price Dataset:

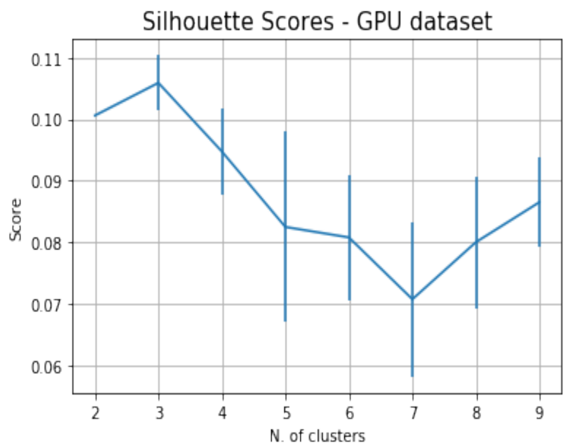
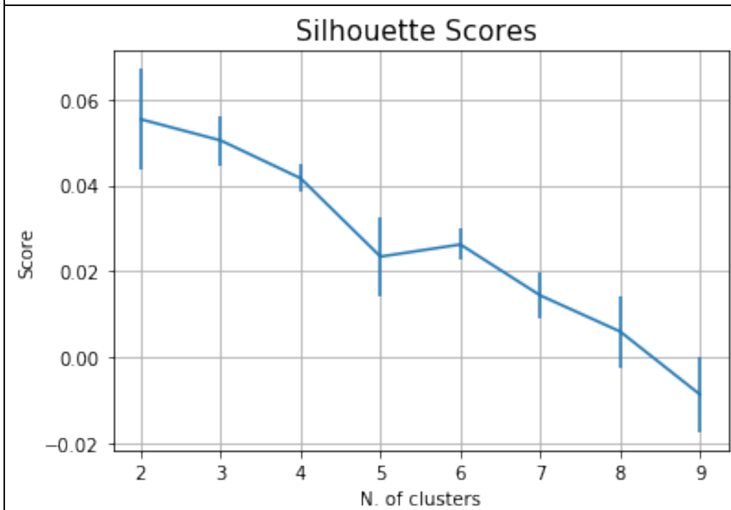


Here, The max value of Cross validation score can be observated at number of features = 5

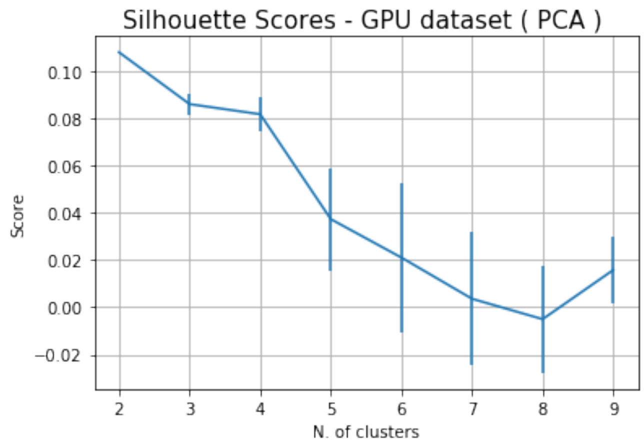


Optimal clusters is still same at 5. Here, feature selections has created compact clusters because there is much reduction in WCSS.

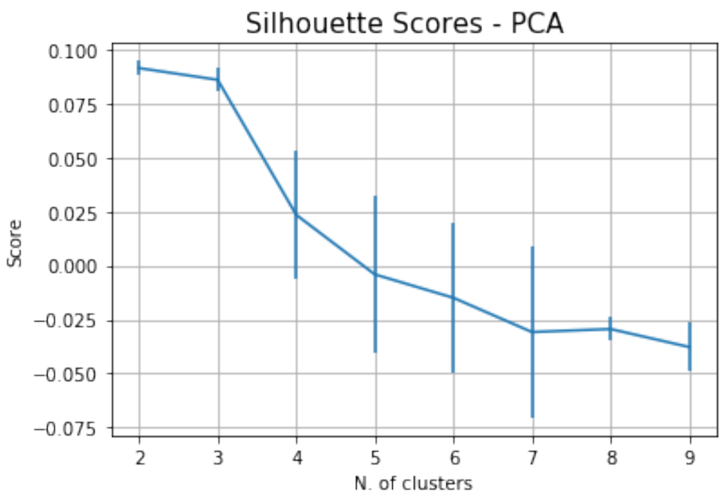
EXPECTATION MAXIMUM

	• Silhouette score is used to compare clusters • We can observe from the plot that the score gradually decreased till 5 clusters. • Since EM does soft clustering, it is trying to restrict no of clusters whereas K-Means does hard clustering. Therefore, we see much more clusters in k-Means
	**Mobile Price Dataset:** • Optimal clusters can be found at k=3. Theoretically , EM produces elongated clusters but here is provided much better compact clusters.

TASK 1: EM with PCA



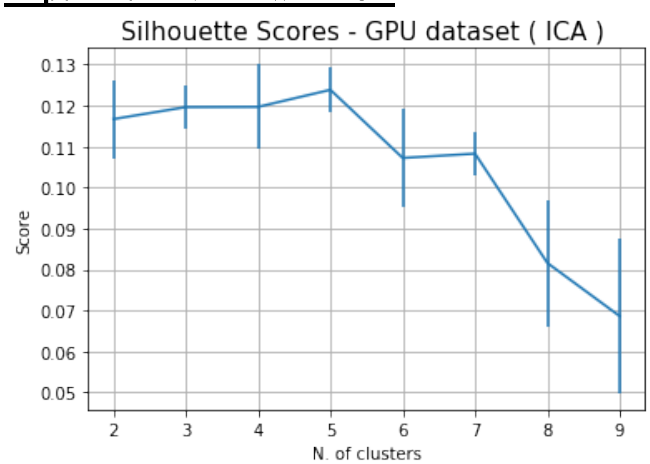
- Theoretically, higher the silhouette score the optimal the configuration
- There is significant decrease of score compared to original dataset for same number of clusters.
- This explains that PCA didn't handle clusters properly



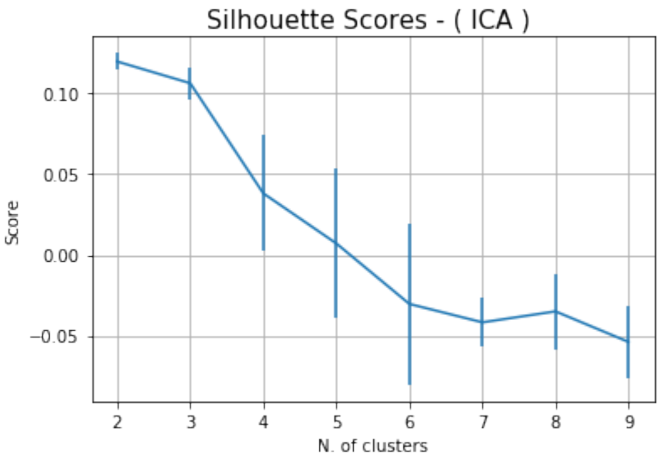
<u>Mobile Price Dataset:</u>

- Performance is similar to the model without PCA.
- Higher Silhouette score can be observed at optimal clusters i.e., k = 3

<u>Experiment 2: EM with ICA</u>



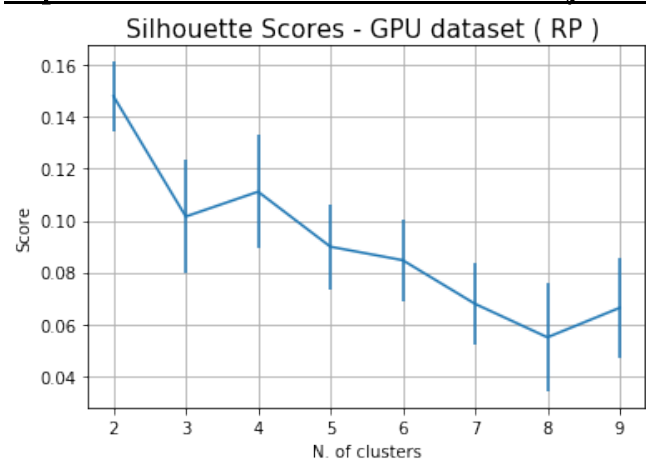
- Silhouette score gradually increased till 5 clusters. So the optimal number of clusters can be observed at 5
- There is an increase of score compared to original dataset for 5 clusters.



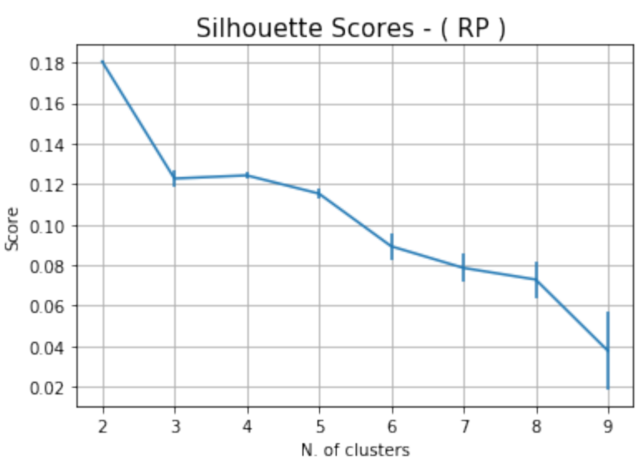
Mobile Price Dataset:

- There is gradual decrease in Silhouette score observed in the plot.
- There is also much decrease in score compared to other techniques for k=3

Experiment 3: EM with Randomized Projections



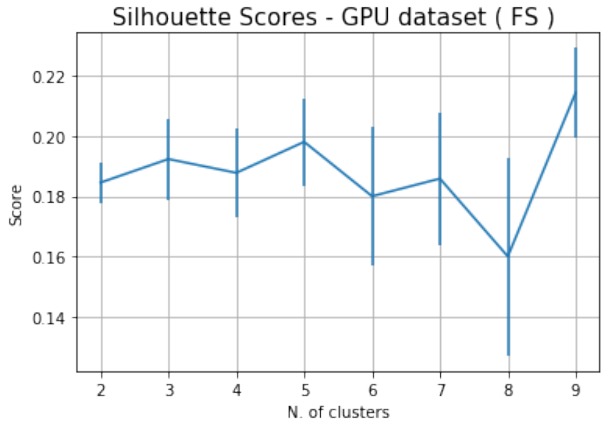
- There is gradual decrease of silhouette score till 8 clusters
- There is not much decrease in score compared to PCA for 8 clusters
- Optimal number of clusters is 2 in this scenario.



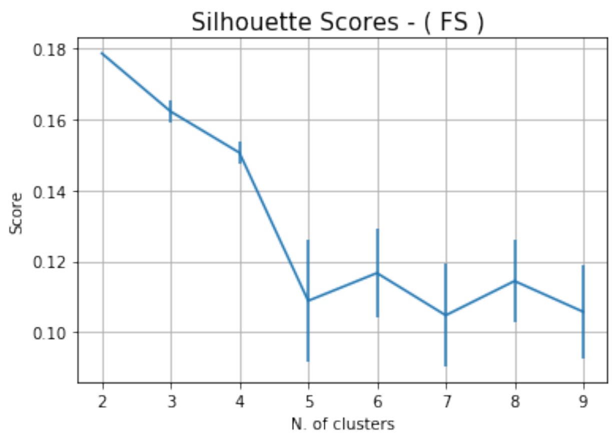
Mobile Price Dataset:

- Similar performance is observed from Randomized Projections.
- Not much improvement with base model.

Experiment 4: EM with Feature Selection



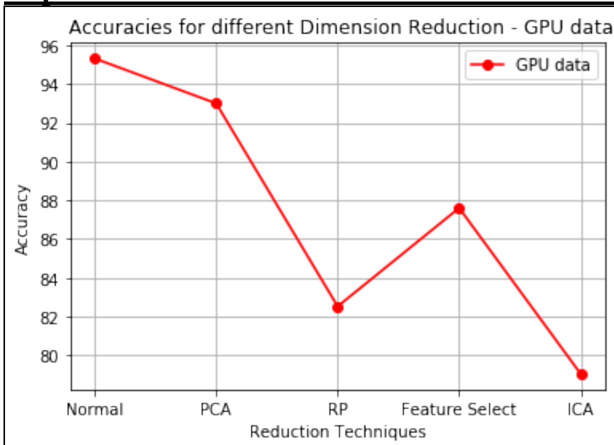
- There is much increase in Silhouette score compared to other dimension reduction techniques
- Even with 8 clusters, we achieved 0.16 score which is almost double the score of other techniques



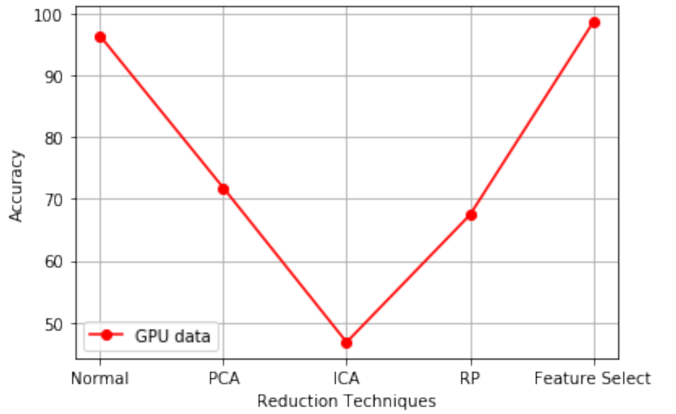
Mobile Price Dataset:

- There is much increase in Silhouette score compared to other dimension reduction techniques
- Even with 5 clusters, we achieved 0.12 score which is almost double the score of other techniques

Experiment 5: ANN with different dimension reduction techniques



- Neural Network able to produce an accuracy of 95%.
- This is because all features were used, and the data explained 100% variance . So, we got higher accuracy
- PCA was also able to get accuracies closer to 93% for GPU data whereas Feature Selection got closer with accuracy of 98% for mobile price data .



Mobile Price Dataset:

- Lowest accuracy for ICA is due to dataset doesn't contain mixed signals , so the independent features were not available to be found.

Experiment 6: ANN with cluster labels

- In this experiment , I have used the clustering on GPU dataset and used the output class (cluster) label (cluster numbers such as 0 , 1, 2, 3 etc.) and fed it as input to the Neural Network with the original output variable (0s and 1s) as the predictor (dependent variable)

```
Accuracy score :  0.6972
[[1502   80]
 [ 677  241]]
```

Mobile Price Dataset:

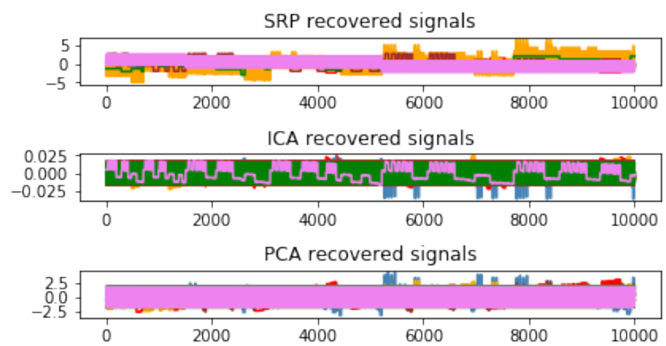
```
0.558
[[171  63]
 [158 108]]
```

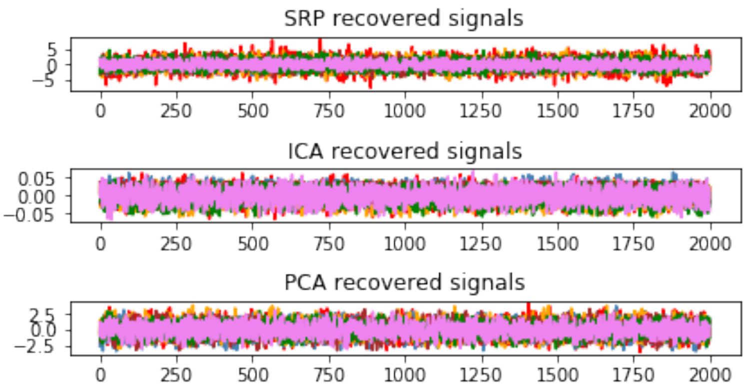
- In this experiment , I have used the clustering on Mobile price dataset and used the output class (cluster) label (cluster numbers such as 0 , 1, 2, 3 etc.) and fed it as input to the Neural Network with the original output variable (0s and 1s) as the predictor (dependent variable)

RP vs ICA vs PCA PLOTS

The different colors represent the different clusters which are formed with the respective dimensionality reduction techniques .

- For RP, we can see that 1 cluster is dominant
- For ICA, there are 2 clusters that are dominant
- As for PCA, we can see that 1 cluster is dominant and many data points are assigned to just one cluster





Mobile Price Dataset:

- For 3 techniques, we can observe that 1 cluster is dominating ie., many data points belongs to that cluster.

- Explanations of your methods. For example, how did you choose k?

  The number of K in clustering was obtained from the elbow charts plotted. For K-Means clustering , the WCSS scores were compared and the K was found where the WCSS didn't reduce significantly even though even we increased the cluster number . So we have to choose the K where the WCSS reduced significantly and then reduces gradually after that.
  For Expectation Maximization , This algorithm only guarantee that we land to a local optimal point, but it do not guarantee that this local optima is also the global one. And so, if the algorithm starts from different initialization points, in general it lands into different configurations. We should use Silhouette scores. The highest Silhouette scores mean that value of clusters should be used.

- What type of clusters did you get? Did they line up with the class labels? If not, did they line up naturally? Were they compact or not? Why do you think you got these types of clusters? How did you pick the features generated by ICA and RP? Compare and contrast the different algorithms .

  Our datasets are multi-dimensional ,we can't just plot the clusters based on random X and Y features . That wouldn't explain the variance properly. So it's very difficult to check whetherour clusters are good and also the shape of the clusters.

- When you reproduced your clustering experiments on the datasets projected onto the new spaces created by ICA, PCA and RP, did you get the same clusters as before? Different clusters? Why? Why not? Compare and contrast the different algorithms.

  We got the same clusters as before . But one major difference was that the newly created features resulted in less error values . For the K-Means the WCSS vales were way below in PCA , ICA and RP when compared to the original set . Hence , the newly identified features did well in clustering and thus the accuracy improved for these models .

  For Expectation Maximization also , we got the same number of clusters . As observed , the Silhouette Scores increased for the newly created feature space for PCA , ICA and RP . The scores were significantly higher for transformed features rather than the original feature set .

- When you re-ran your neural network algorithms were there any differences in performance? Speed? Anything at all?

  The training time went down considerably for the transformed features when compared to the training time it took for Neural Network without any feature transformation . The algorithm ran faster and more efficiently when we used the transformed feature set .