DSC PROJECT GROUP 12

Name1: Nehal Singh Parmar

Roll No:187141

Name2:Shanu kumar

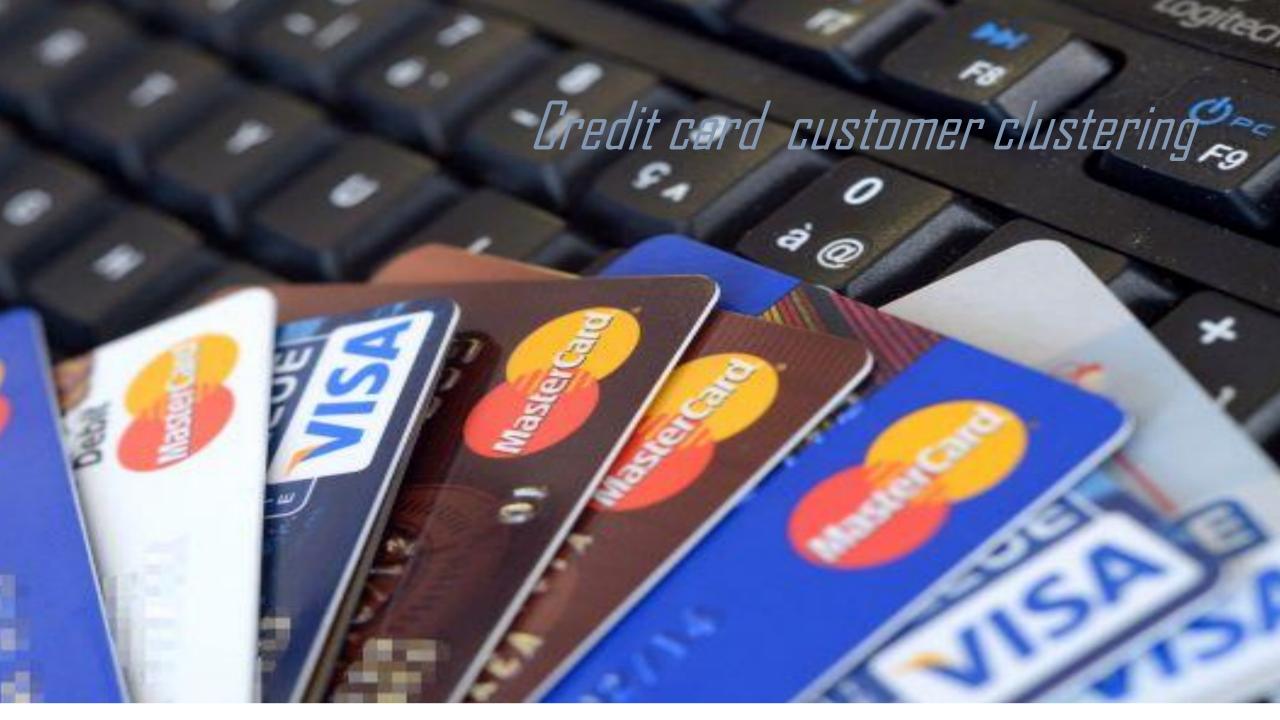
Roll No:187159

Name3:Bhanuprakash kasibisi

Roll No:187128

Name4:Hitender Singh Shekhawat

Roll No.:187122





- → The number one challenge faced by marketers is to understand who they are selling to.
- → When you know your buyers' personas you can tailor your targeting and offerings to increase their satisfaction and your revenue as a result.
- → When you already have a pool of customers and enough data on them, it can be very useful to segment them.
- →Here, we are going to see how we can use clustering to segment some credit card customers.



->The problem described in this dataset requires us to extract segments of customers depending on their behaviour patterns provided in the dataset, to focus marketing strategy of the company on a particular segment.

->DATA:-The credit card data has 17 attributres for each customer which include the balance (credit owed by the customer), cash advance (when a customer withdraws cash using the credit card), the customer's credit limit, minimum payment, percentage of full payments and tenure.

Accuracy = Samples correctly classified/Total number of samples $F_{\beta}=(I+\beta^2)^*$ precision*recall/($(\beta^2 * precision)$ +recall)

METRICS

- Accuracy(precision)= 0.5554
- $F_{\beta} = 0.8620$

Note: β =2,recall=1

DATA PREPROCESSING

- DATA ANALYSIS
- NORMALISATION

IMPLEMENTATION

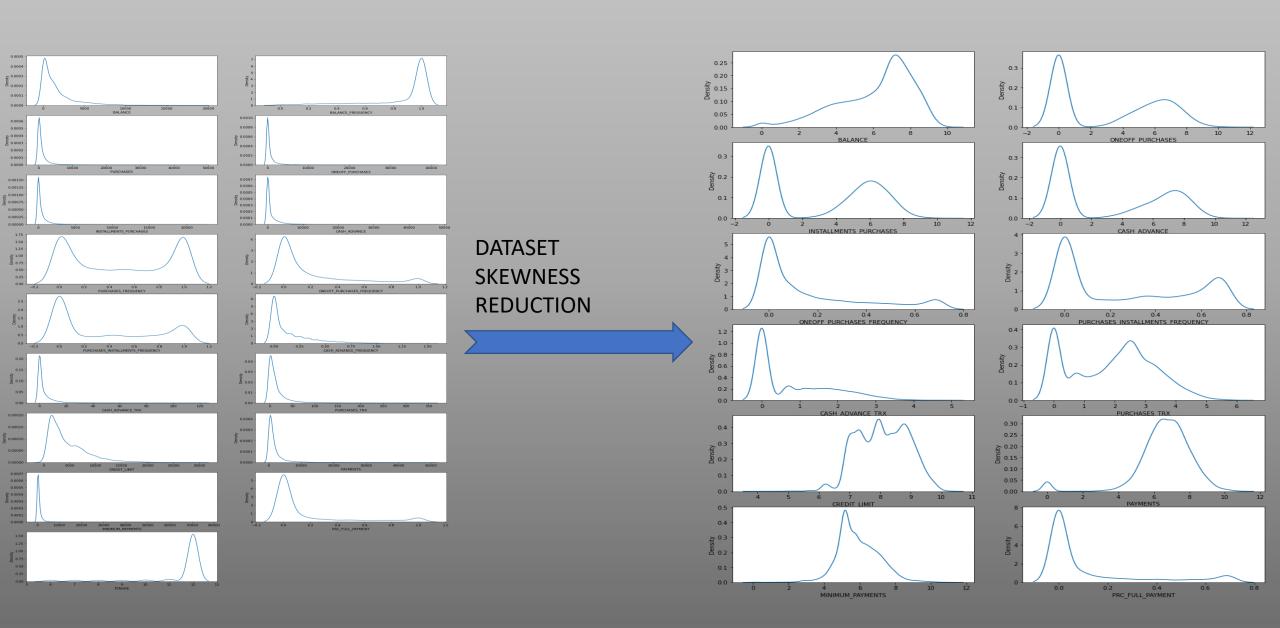
- SPLITTING DATA
- CREATING TRAINING & amp; amp; PREDICTION PIPELINE
- REFINEMENT
- FEATURE IMPORTANCE

STEP WISE APPROACH

Data taken

Data	columns (total 18 columns):				
#	Column	Non-Null Count	Dtype		
0	CUST_ID	8950 non-null	object		
1	BALANCE	8950 non-null	float64		
2	BALANCE_FREQUENCY	8950 non-null	float64		
3	PURCHASES	8950 non-null	float64		
4	ONEOFF_PURCHASES	8950 non-null	float64		
5	INSTALLMENTS_PURCHASES	8950 non-null	float64		
6	CASH_ADVANCE	8950 non-null	float64		
7	PURCHASES_FREQUENCY	8950 non-null	float64		
8	ONEOFF_PURCHASES_FREQUENCY	8950 non-null	float64		
9	PURCHASES_INSTALLMENTS_FREQUENCY	8950 non-null	float64		
10	CASH_ADVANCE_FREQUENCY	8950 non-null	float64		
11	CASH_ADVANCE_TRX	8950 non-null	int64		
12	PURCHASES_TRX	8950 non-null	int64		

DATA PROCESSING



CORRELATION IDENTIFICATION AMONG FEATURES

USING PCA FOR DIMENSIONALITY REDUCTION FOR INCREASING NUMBER OF CORRELATED FEATURES

from sklearn.decomposition import PCA

 $pca = PCA(n_components=0.95)$

X_red = pca.fit_transform(df)

		8.3T	70.25						1767							75	
BALANCE	- 1	0.65	0.17	0.18	-0.11	0.52	-0.094	0.16	-0.12	0.45	0.5	-0.017	0.3	0.43	0.76	-0.42	0.073
BALANCE_FREQUENCY	0.65	1	0.13	0.14	0.11	0.16	0.23	0.2	0.17	0.19	0.19	0.2	0.095	0.32	0.42	-0.097	0.12
PURCHASES	0.17	0.13	1	0.48	0.38	-0.16	0.39	0.5	0.31	-0.12	-0.14	0.53	0.3	0.35	0.13	0.18	0.086
ONEOFF_PURCHASES	0.18	0.14	0.48	1	0.16	-0.19	0.37	0.81	0.097	-0.12	-0.14	0.57	0.28	0.27	0.059	0.042	0.092
INSTALLMENTS_PURCHASES	-0.11	0.11	0.38	0.16	1	-0.39	0.79	0.19	0.91	-0.29	-0.34	0.78	0.11	0.17	-0.039	0.28	0.099
CASH_ADVANCE	0.52	0.16	-0.16	-0.19	-0.39	1	-0.43	-0.18	-0.37	0.77	0.9	-0.43	0.12	0.18	0.41	-0.33	-0.1
PURCHASES_FREQUENCY	-0.094	0.23	0.39	0.37	0.79	-0.43	1	0.5	0.87	-0.31	-0.37	0.91	0.11	0.18	-0.053	0.31	0.061
ONEOFF_PURCHASES_FREQUENCY	0.16	0.2	0.5	0.81	0.19	-0.18	0.5	1	0.14	-0.11	-0.15	0.62	0.29	0.27	0.04	0.14	0.081
PURCHASES_INSTALLMENTS_FREQUENCY	-0.12	0.17	0.31	0.097	0.91	-0.37	0.87	0.14	1	-0.27	-0.32	0.77	0.053	0.13	-0.04	0.26	0.071
CASH_ADVANCE_FREQUENCY	0.45	0.19	-0.12	-0.12	-0.29	0.77	-0.31	-0.11	-0.27	1	0.91	-0.3	0.14	0.19	0.36	-0.25	-0.13
CASH_ADVANCE_TRX	0.5	0.19	-0.14	-0.14	-0.34	0.9	-0.37	-0.15	-0.32	0.91	1	-0.36	0.12	0.2	0.42	-0.29	-0.08
PURCHASES_TRX	-0.017	0.2	0.53	0.57	0.78	-0.43	0.91	0.62	0.77	-0.3	-0.36	1	0.2	0.25	0.005	0.27	0.14
CREDIT_LIMIT	0.3	0.095	0.3	0.28	0.11	0.12	0.11	0.29	0.053	0.14	0.12	0.2	1	0.34	0.25	0.044	0.17
PAYMENTS	0.43	0.32	0.35	0.27	0.17	0.18	0.18	0.27	0.13	0.19	0.2	0.25	0.34	1	0.28	0.15	0.21
MINIMUM_PAYMENTS	0.76	0.42	0.13	0.059	-0.039	0.41	-0.053	0.04	-0.04	0.36	0.42	0.005	0.25	0.28	1	-0.37	0.15
PRC_FULL_PAYMENT	-0.42	-0.097	0.18	0.042	0.28	-0.33	0.31	0.14	0.26	-0.25	-0.29	0.27	0.044	0.15	-0.37	1	-0.014
TENURE		0.12	0.086	0.092	0.099	-0.1	0.061	0.081	0.071	-0.13	-0.08	0.14	0.17	0.21	0.15	-0.014	1
	BALANCE -	BALANCE_FREQUENCY -	PURCHASES -	ONEOFF_PURCHASES -	INSTALLMENTS_PURCHASES -	CASH_ADVANCE -	PURCHASES_FREQUENCY -	OFF_PURCHASES_FREQUENCY -	S_INSTALLMENTS_FREQUENCY -	CASH_ADVANCE_FREQUENCY -	CASH_ADVANCE_TRX -	PURCHASES_TRX -	CREDIT_LIMIT -	PAYMENTS -	MINIMUM PAYMENTS -	PRC_FULL_PAYMENT -	TENURE -
					INST/		2	OFF_PU	S_INST	CASH							

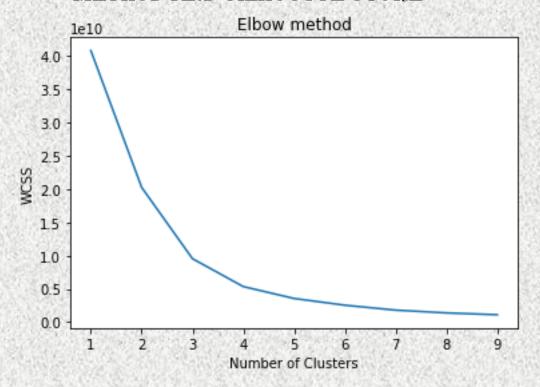
-1.0

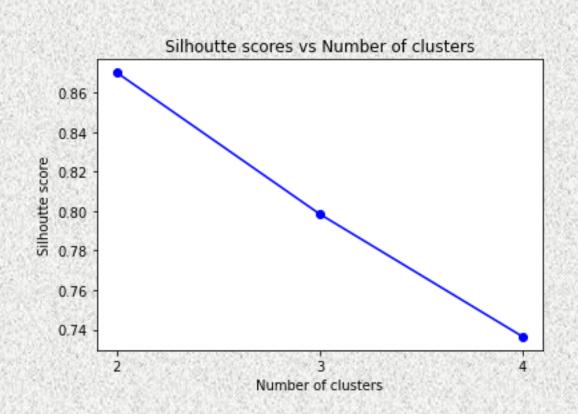
-0.8

- 0.6

MODEL TRAINING USING K-MEANS CLUSTERING

DECIDING NO.OF CLUSTERS USING ELBOW METHOD AND SILHOUTTE SCORE





Elbow is around 2,3,4 so using silhouette score for more accurately deciding on no.of clusters

N=2 has high silhouette score than other 2 alternatives

Model Evaluation and Inference.

```
from sklearn.metrics import silhouette_score

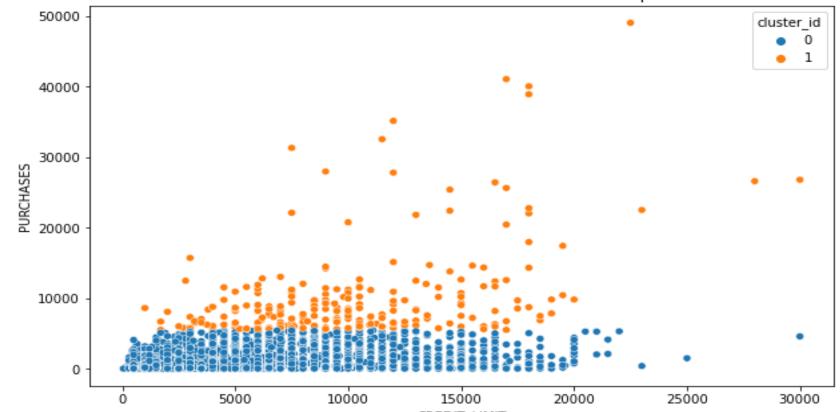
kmeans = KMeans(n_clusters=2, random_state=23)
kmeans.fit(X_red)

print('Silhoutte score of our model is ' + str(silhouette_score, kmeans.labels_)))
```

RESULTS

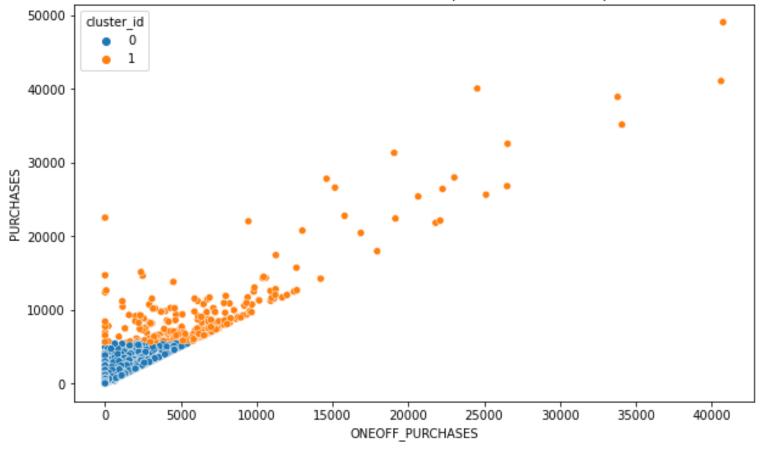
Silhoutte score of our model is 0.8700455999561516

Distribution of clusters based on Credit limit and total purchases



Distribution of clusters based on One off purchases and total purchases

->Looking at the 2 plots it can be concluded that our model has clustered customers with (low usage of credit card in one cluster)and customers with (higher usage) of clusters in other.



CONCLUSIONS

our clustering model will be able to segment our credit card users into distinctive groups. Some of these will be fairly classical such as the prime segment, revolvers and transactors. Understanding the behaviour of customers at this level of granularity is key to tailoring offers which improve customer retention and drive revenues