AllLife Bank

Customer Segmentation

PG-DSBA Project 7
Eric Green
May 2021

Background

- AllLife Bank wants to focus on its credit card customer base in the next financial year
- The marketing research team indicates that market penetration can and should be improved by running personalized marketing campaigns to target new credit card customers in addition to upselling card services to existing bank customers
- The marketing research team also indicates that customers perceive a poor experience with the bank's credit card support services
- The operations team wants to improve the service delivery model to answer customers queries faster and more reliably to improve the customer support experience
- Marketing and Operations leadership have commissioned a customer segmentation analysis to be performed as the basis for strategic business insights and recommendations to achieve above goals of the bank

Business Objectives

Identify different segments/groups of existing customer based on their spending patterns as well as past interactions with the bank

- ✓ **Growth Objective** - Increase credit card market share by moving customers from other providers to ours and converting existing bank customers
- ✓ **New Marketing Campaigns** - Customer segmentation to inform personalized credit card marketing campaigns for new and existing customers
- ✓ **Service Optimization** - Optimization strategy to improve customer service delivery speed and quality
- ✓ Business strategy recommendations to accomplish above

Analysis & Modeling Objectives

- Exploratory data analysis to understand the customer data
- K-means clustering to identify customer groups to target
- Hierarchical clustering to identify customer groups to target
- Clustering optimization & comparison
- Dimensional reduction & testing with PCA
- Linear Regression modeling & testing

Raw Data Summary

Raw Variables

0 **Sl_No** 660 non-null int64
1 **Customer Key** 660 non-null int64
2 **Avg_Credit_Limit** 660 non-null int64
3 **Total_Credit_Cards** 660 non-null int64
4 **Total_visits_bank** 660 non-null int64
5 **Total_visits_online** 660 non-null int64
6 **Total_calls_made** 660 non-null int64

Null Values	Duplicates
Sl_No 0 Customer Key 0 Avg_Credit_Limit 0 Total_Credit_Cards 0 Total_visits_bank 0 Total_visits_online 0 Total_calls_made 0	Sl_No 0 Customer Key 0 Avg_Credit_Limit 0 Total_Credit_Cards 0 Total_visits_bank 0 Total_visits_online 0 Total_calls_made 0

cc-data.xlsx shape: (660, 7)

Raw Summary Statistics

	count	mean	std	min	25%	50%	75%	max
Sl_No	660.0	330.500000	190.669872	1.0	165.75	330.5	495.25	660.0
Customer Key	660.0	55141.443939	25627.772200	11265.0	33825.25	53874.5	77202.50	99843.0
Avg_Credit_Limit	660.0	34574.242424	37625.487804	3000.0	10000.00	18000.0	48000.00	200000.0
Total_Credit_Cards	660.0	4.706061	2.167835	1.0	3.00	5.0	6.00	10.0
Total_visits_bank	660.0	2.403030	1.631813	0.0	1.00	2.0	4.00	5.0
Total_visits_online	660.0	2.606061	2.935724	0.0	1.00	2.0	4.00	15.0
Total_calls_made	660.0	3.583333	2.865317	0.0	1.00	3.0	5.00	10.0

Observations

- Raw data shape is 660 rows x 7 columns
- The raw data tidy with no null/NA values, duplicate rows, placeholder or obvious anomaly values
- 5 point summary of raw column vectors indicate realistic probability densities per column vectors
- No raw data value cleansing was needed or performed

Data Preprocessing

Processed shape: (660, 5)

Processed Features

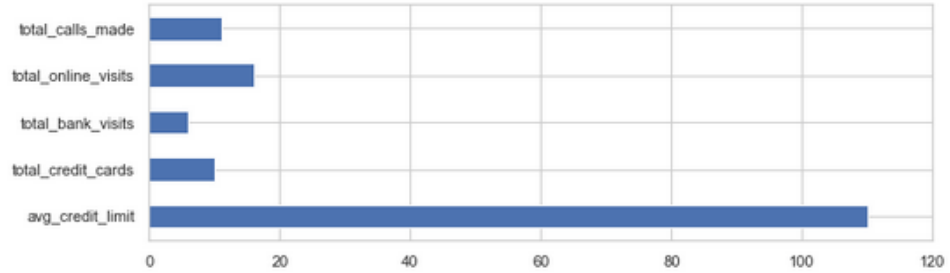
Column names were changed to be intuitive to work with downstream

avg_credit_limit 660 non-null int64
total_credit_cards 660 non-null int64
total_bank_visits 660 non-null int64
total_online_visits 660 non-null int64
total_calls_made 660 non-null int64

```
Uniques

 avg_credit_limit       110
total_credit_cards      10
total_bank_visits        6
total_online_visits     16
total_calls_made        11
dtype: int64
```

value uniqueness across columns

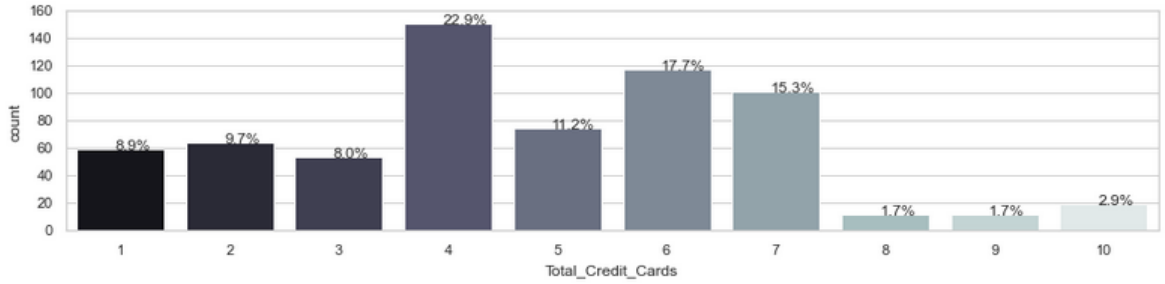


Observations

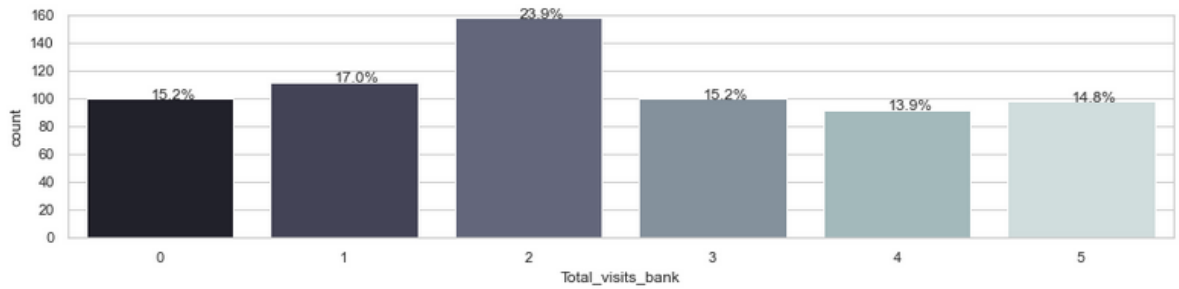
- The raw data is quite clean and does not require significant modification
- The raw data is confined to cc customer behavior and does not contain general customer attributes (hence the need for clustering)
- Raw data shape is 660 rows x 7 columns, all column vectors are integer
- SI_NO (serial number) and Customer Key columns were removed as they are not needed
- Columns were renamed to be intuitive to work with in downstream data engineering
- The magnitude of **avg_credit_limit** dominates the scale of all other columns
- All values present in the data look plausible and to be reasonably expected
- Initial preprocessing reduced data to shape 660 rows x 5 columns after removing non-value add variables

Raw Data Inspection - Count Distributions

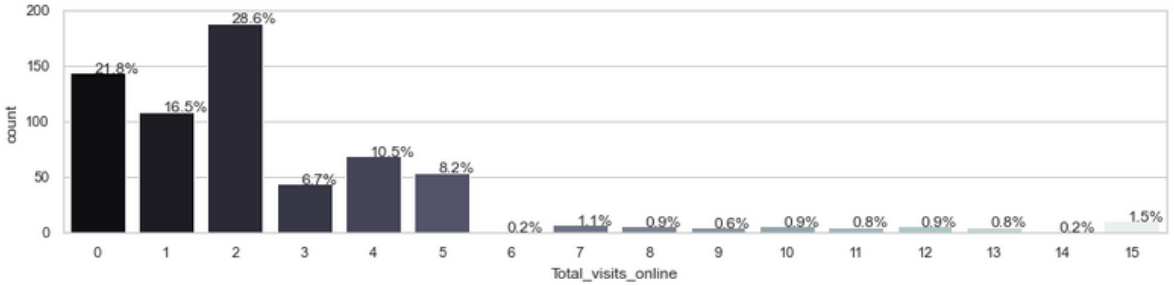
93.7% of all customers have less than 8 cards



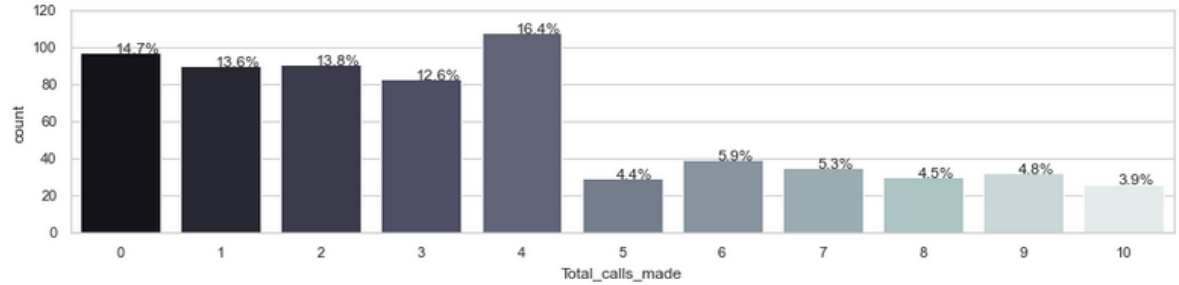
56.8% of customers visited the bank 2 or fewer times



6 of 7 (~85%) of customers had 5 or less online visits

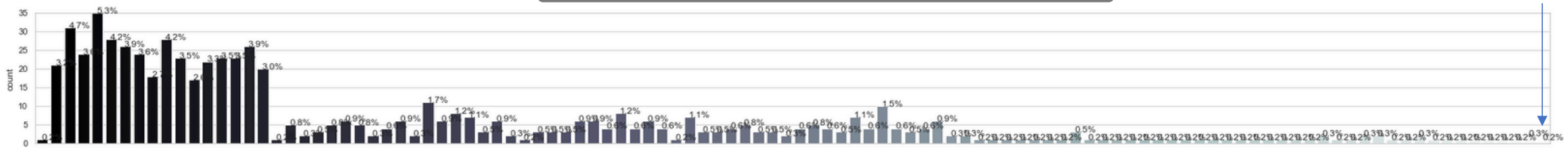


29% of customers made 5 or more support calls



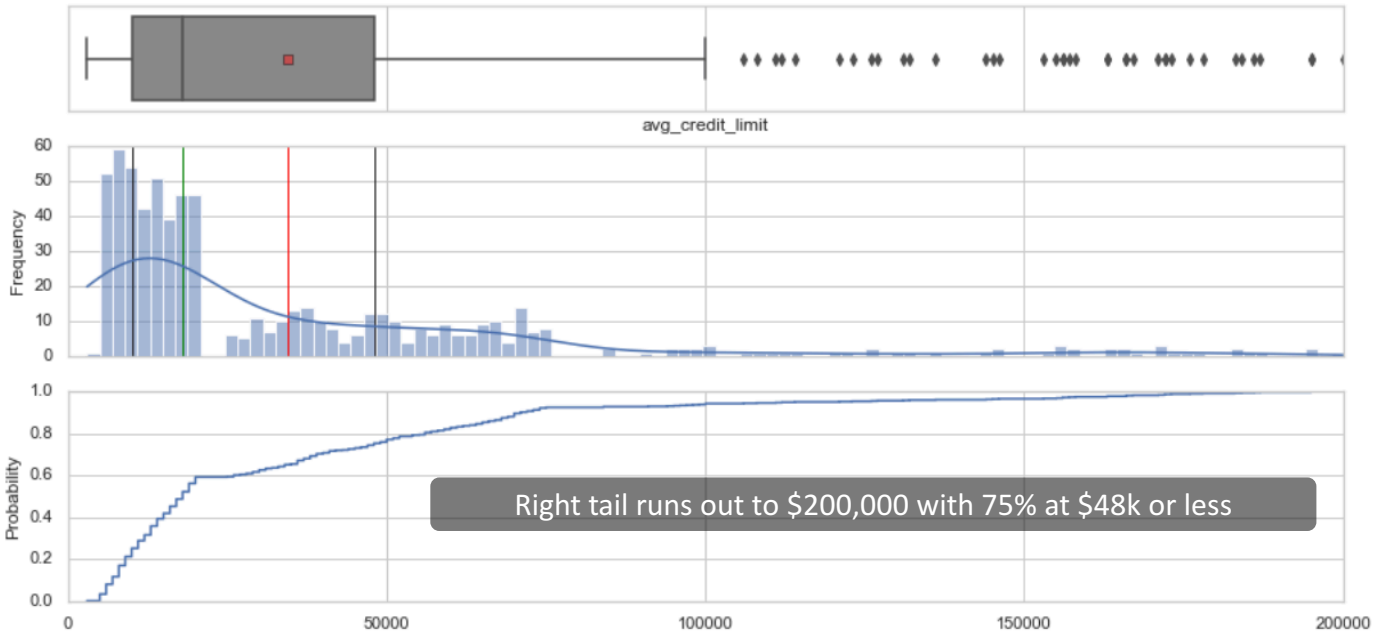
Half of customers have an average credit limit at or below 18K

200k

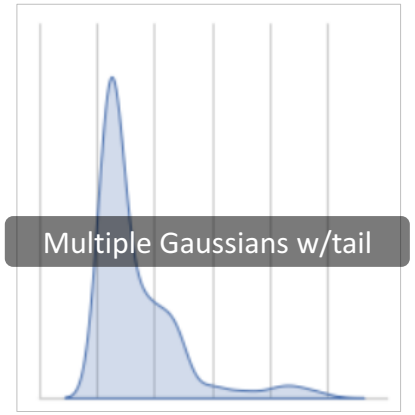


Univariate EDA on Column Vectors

avg_credit_limit



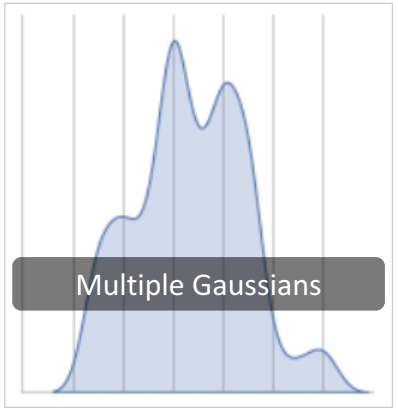
avg_credit_limit



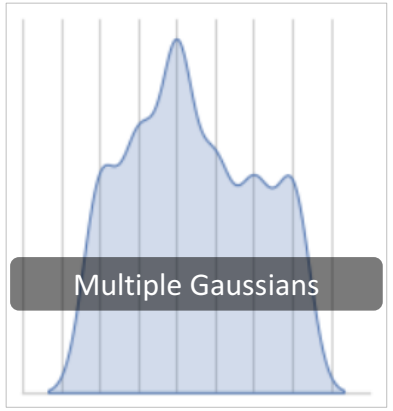
mean	34574.242424
std	37625.487804
min	3000.000000
25%	10000.000000
50%	18000.000000
75%	48000.000000
max	200000.000000

The number of Gaussians (modes) gives us an intuitive sense of how many clusters potentially exist within the data

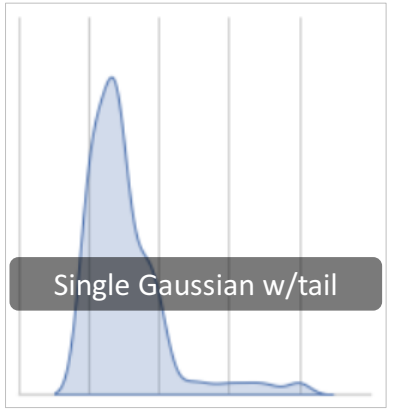
We can use this intuition when comparing against number of clusters in top-down or bottom-up clustering analysis



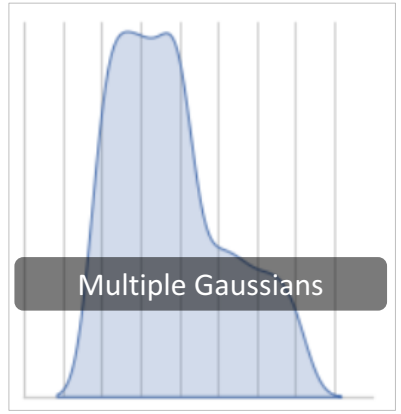
total_credit_cards



total_bank_visits



total_online_visits

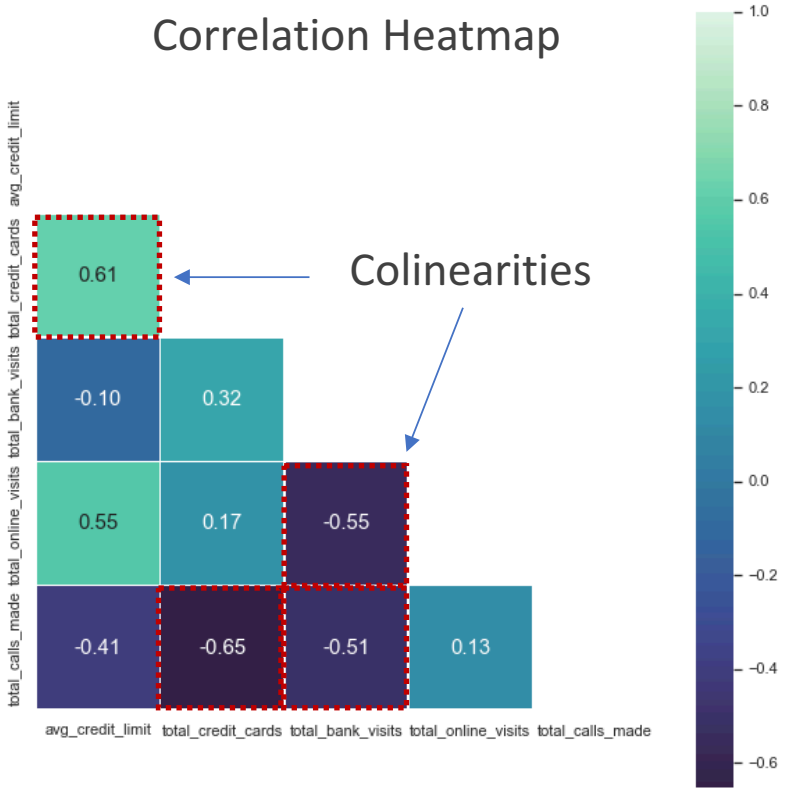


total_calls_made

Summary Observations on Preprocessed Data

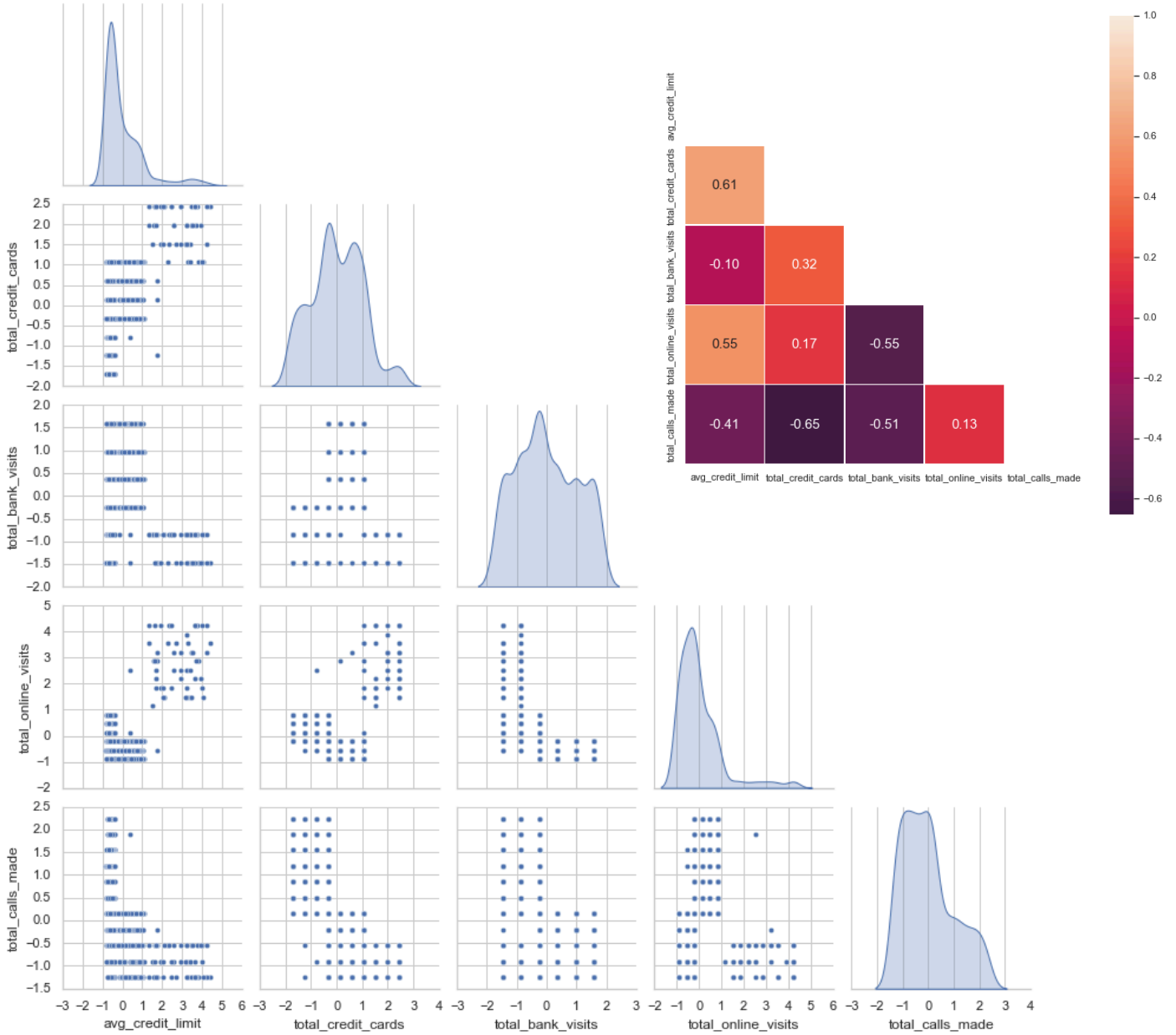
Understanding central tendency, spread and probability densities

1. avg_credit_limit
    1. Magnitude dominates the scale of all other columns
    2. Has a long right tail indicated outliers outside of the box whiskers
    3. Vector appears to be multimodal having at least 2 distinct gaussians
    4. Max avg_credit_limit observed is large at 200k (through is plausible for ultra wealthly individuals)
2. total_credit_cards
    1. No customer has more than 10 total_credit_cards
    2. 80% of customers have 6 cards or less
    3. A random customer is mostly likely to 4 cards
    4. 50% of customers have less than 4 cards while the other 50% have more than 4 cards
3. total_bank_visits
    1. No customer has had more than 5 bank visits
    2. Mean banks visits per customer is 2.4
4. total_online_visits
    1. 75% of customers have 4 or fewer online visits
    2. Customers with 8 or more online visits are reasonable outliers visible in the long right tail
5. total_calls_made
    1. No customer has made more than 10 calls
    2. 75% of customers have called 4 times or less
6. KDE pairplots shows low-to moderate correlation block patterns and KDEs highlight the multimodal nature of the frequency distributions
7. Heatmap shows moderate correlation between:
    1. avg_credit_limit & total_credit_cards (.61)
    2. avg_credit_limit & total_online_visits (.55)
    3. total_bank_visits & total_online_visits (-.55)
    4. total_bank_visits & total_calls_made (-.51)
    5. total_credit_cards & total_calls_made (-.65)
    6. Covariance between these variables all make sense logically

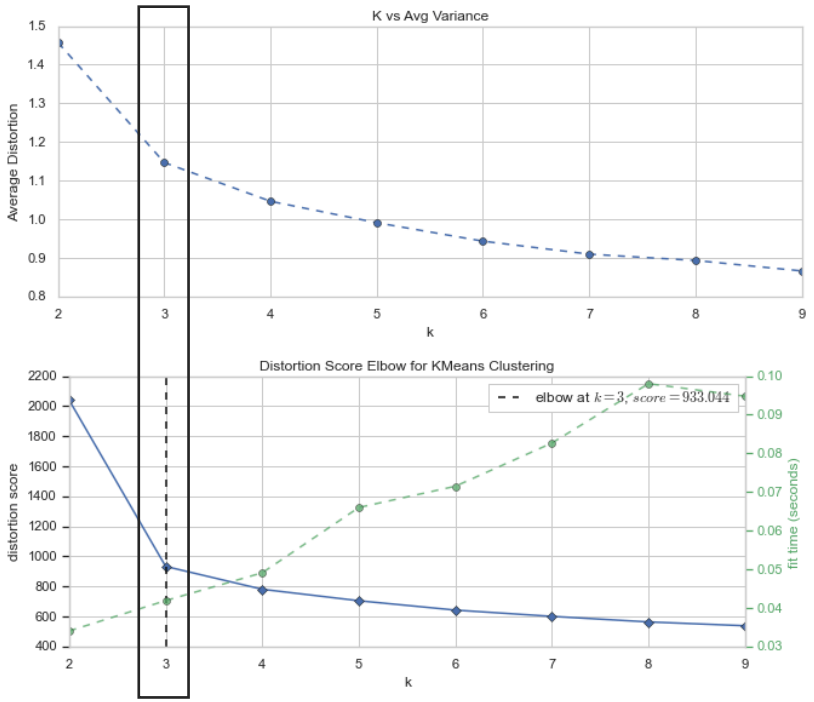


Z Score Standardization

- ✓ Here we transform the features into a (unit-less) value of standard deviations each point is away from its mean (the shape of the data is preserved exactly)
- ✓ This transform renders a mean of 0 and standard deviation of 1 and prevents large magnitude column vectors from having a dominating influence on cluster model fitting

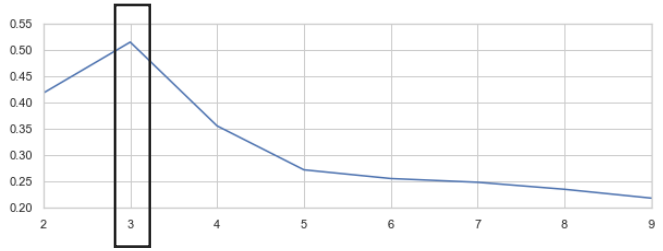


KMeans Clustering – Elbow Method & Silhouette Analysis

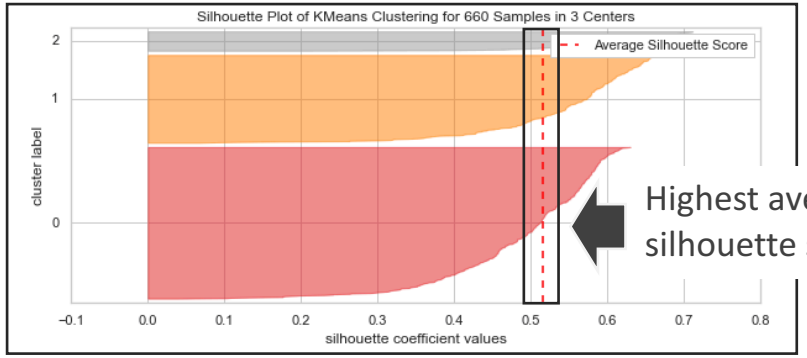


Silhouette Coefficient

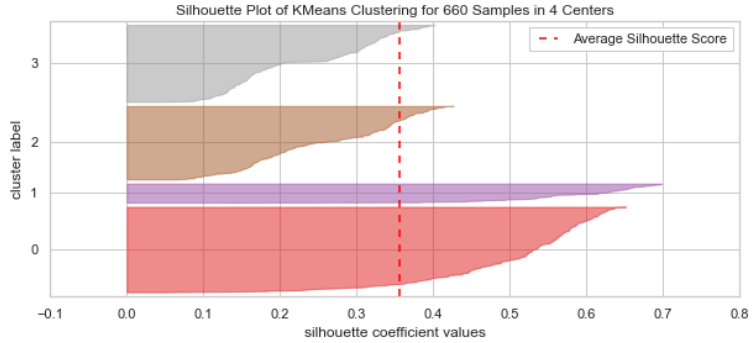
The best value is 1 and the worst value is -1. Values near 0 indicate overlapping clusters. Negative values generally indicate that a sample has been assigned to the wrong cluster, as a different cluster is more similar



For n_clusters = 2, silhouette score is 0.41842496663215445)
For n_clusters = 3, silhouette score is 0.5157182558881063)
For n_clusters = 4, silhouette score is 0.3556670619372605)
For n_clusters = 5, silhouette score is 0.2717470361089752)
For n_clusters = 6, silhouette score is 0.25508747605095977)
For n_clusters = 7, silhouette score is 0.2479547871676149)
For n_clusters = 8, silhouette score is 0.2345439760452223)
For n_clusters = 9, silhouette score is 0.2176620570412375)



Highest average silhouette score

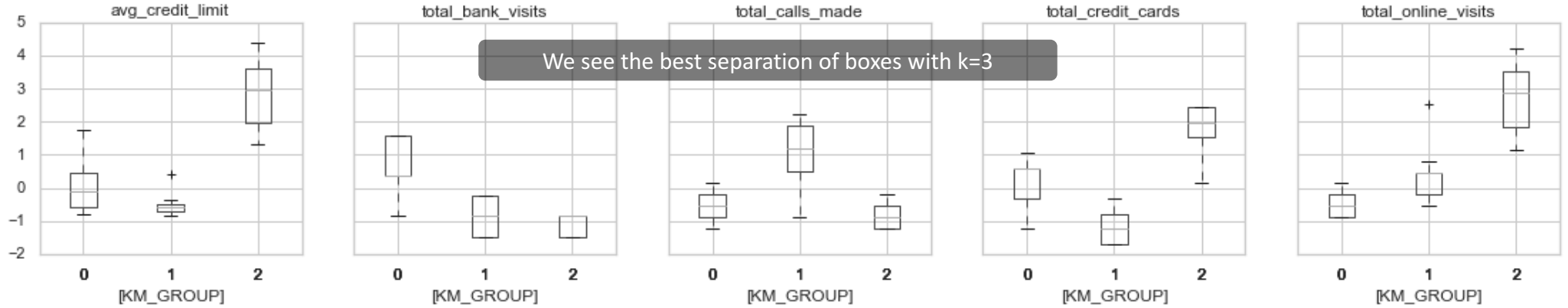


Observations

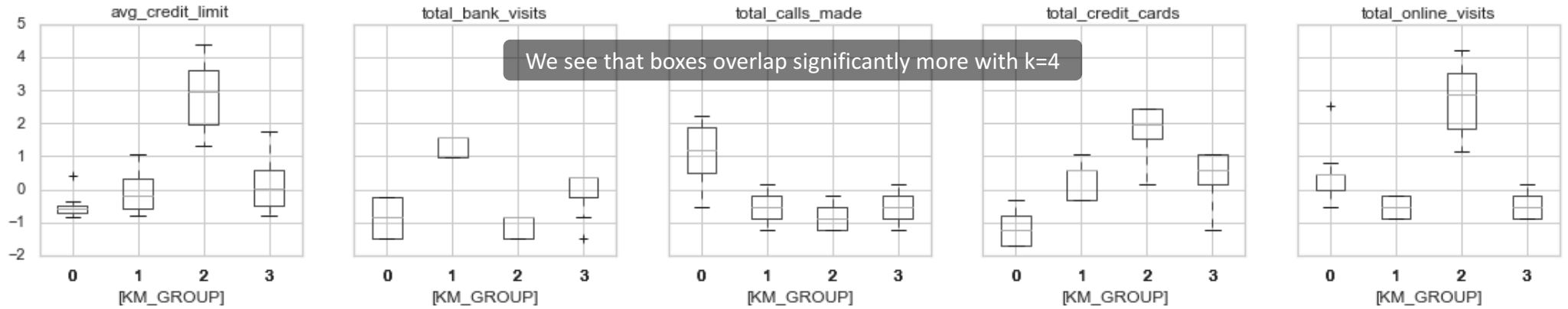
- Using KMeans clustering based on Euclidean distance, we determine the elbow to be at 3 clusters (the elbow is where a cluster's ability to explain the variance begins to more sharply reduce compared to prior)
- From the above charts, we will do further cluster modeling using a k value of 3 or 4, where we should get the most bang for the buck
- Silhouette visualizer and silhouette coefficient indicate that the best number of clusters to use is 3
- The elbow and silhouette methods agree on 3 as the best number of clusters

KMeans Clustering – Boxplot Distinctions

KM_GROUPs: 3 (k)



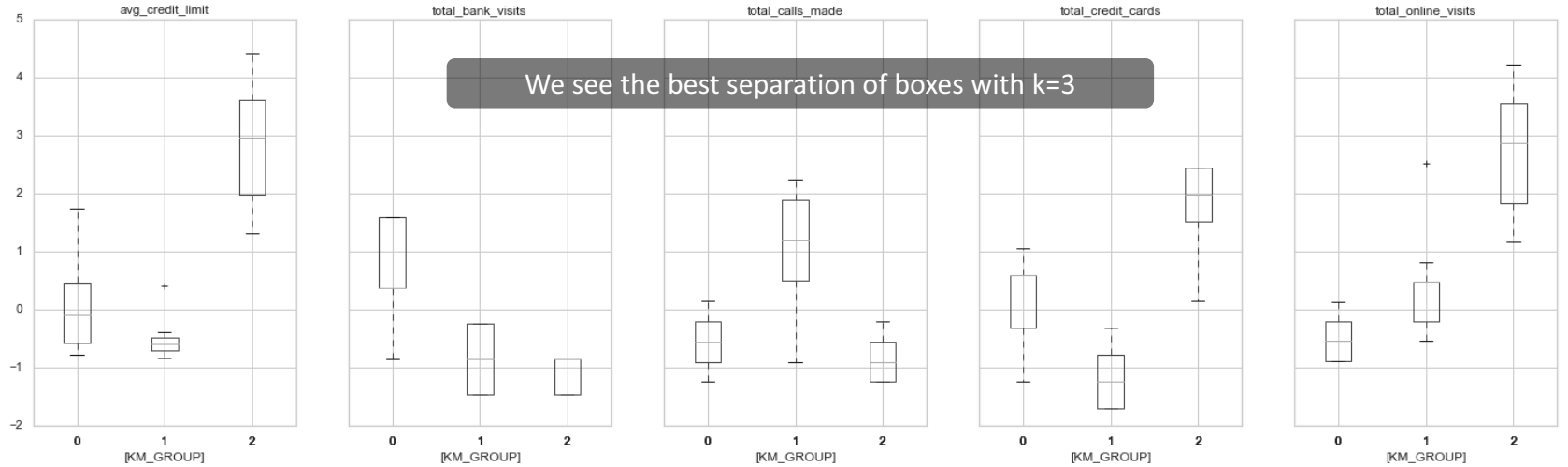
KM_GROUPs: 4 (k)



KMeans Clustering (k=3)

Top Down clustering

3 most distinction clusters (KMeans)



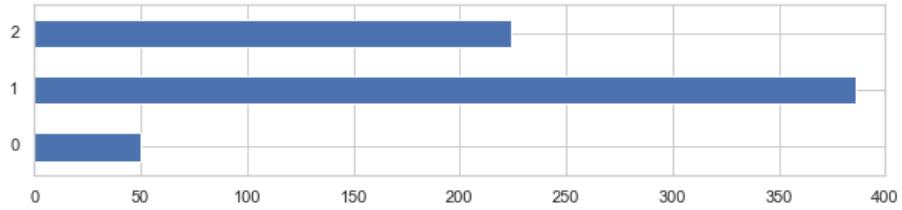
We see the best separation of boxes with k=3

Observations on KMeans Clustering (k=3)

- From the visual inspection of the 3 cluster boxplots, we can observe that:
  - **Wealthy Tech Savvy GROUP**
    - High Credit, Online User, Low Support
  - **Low Credit User GROUP**
    - Low Credit, Caller, High Support
  - **Bank Traditionalist GROUP**
    - Medium Credit, Bank Visitor, Medium Support
- WTS group represents 386 customers (58%)
- LCU group represents 224 customers (34%)
- BT group represents 50 customers (7%)

Cluster sizes

```
0     50
1    386
2    224
Name: KM_GROUP, dtype: int64
```



Maximums

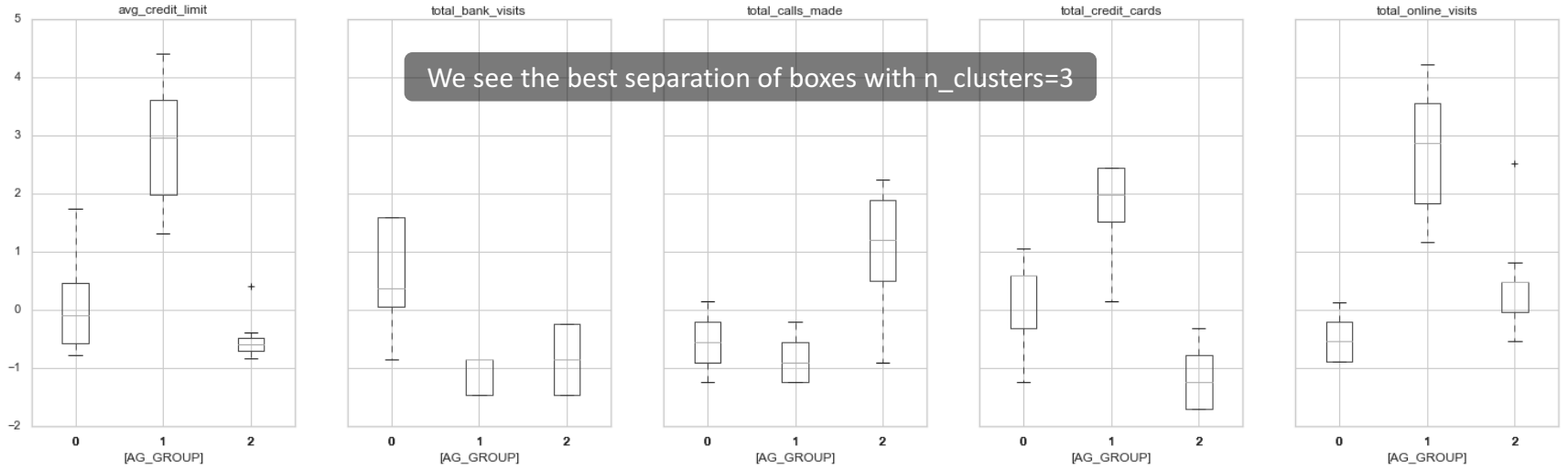
KM_GROUP	avg_credit_limit	total_credit_cards	total_bank_visits	total_online_visits	total_calls_made	G_COUNT
0	2.831764	1.862226	-1.105763	2.827319	-0.874330	50
1	-0.021062	0.373690	0.666395	-0.553672	-0.553005	386
2	-0.595796	-1.059623	-0.901518	0.322997	1.148109	224

Above we can see that GROUP 0 has the highest value for 3 of 5 dimensions. This is what characterizes the cluster.

Agglomerative Clustering (n_clusters=3)

Bottom Up clustering

3 most distinction clusters (Agglomerative)



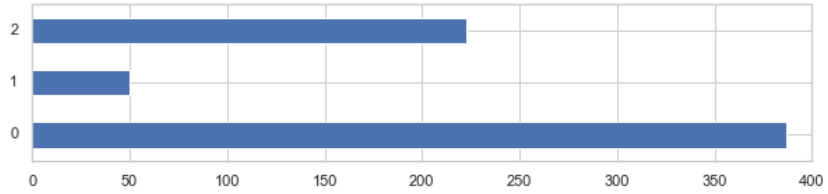
We see the best separation of boxes with n_clusters=3

Observations on Agglomerative Clustering (n_clusters=3)

- From the visual inspection of the 3 cluster boxplots, we can observe that Agglomerative clustering produces grouping results very similar to KMeans clustering

Cluster sizes

```
0     50
1    386
2    224
Name: KM_GROUP, dtype: int64
```

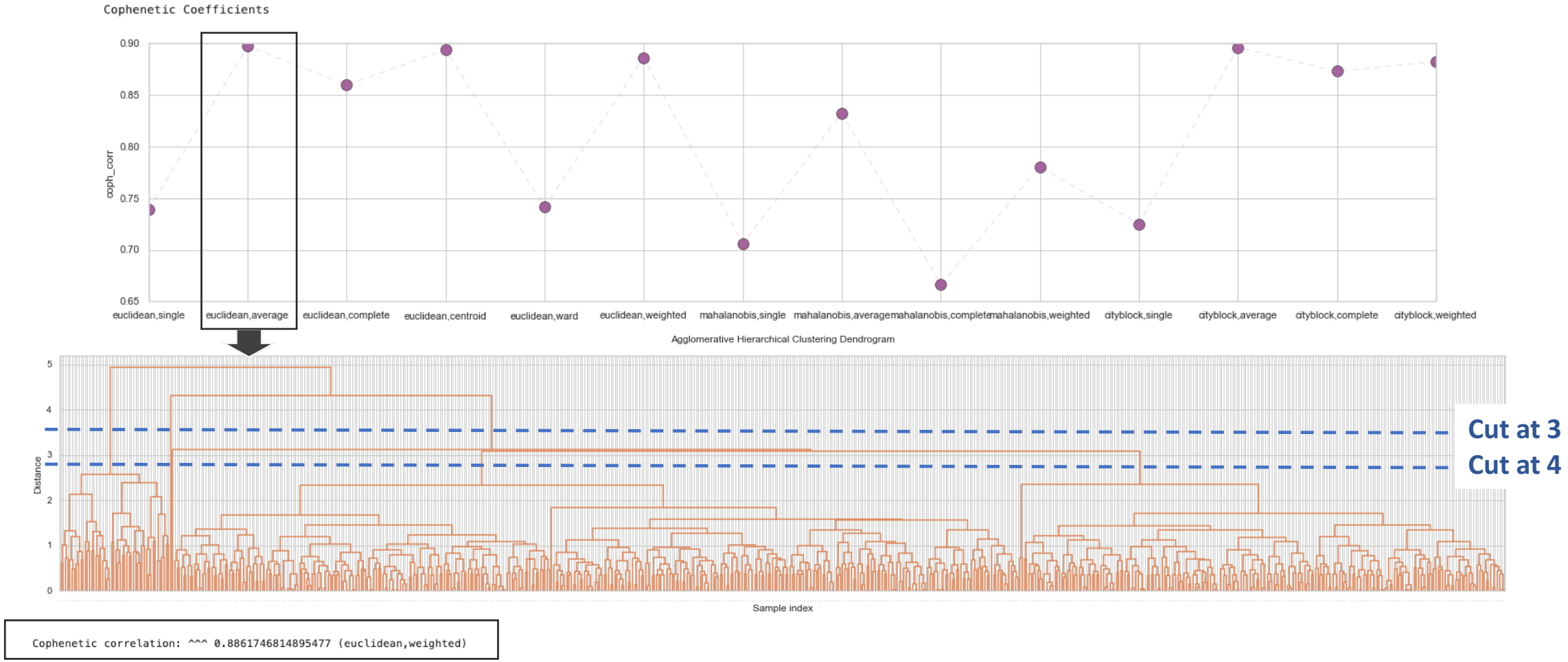


Maximums

AG_GROUP	avg_credit_limit	total_credit_cards	total_bank_visits	total_online_visits	total_calls_made	A_COUNT
0	-0.022902	0.371882	0.664034	-0.552775	-0.551200	387
1	2.831764	1.862226	-1.105763	2.827319	-0.874330	50
2	-0.595179	-1.062913	-0.904453	0.325372	1.152605	223

Above we can see that GROUP 1 has the highest value for 3 of 5 dimensions. This is what characterizes the cluster.

Cophenetic Coefficient & Dendrogramic Clustering




```
Cophenetic correlation: ^^^ 0.8861746814895477 (euclidean,weighted)
```

Observations on Cophenetic Coefficient & Dendrograms

- Here we allow unsupervised dendrograms to determine what the clustering should be (bottom up) and then we can "cut" at a specific distance level on the y axis to get the clusters
- We are looking for the closest value we can find to 1 for the cophenetic coefficient across the dendrograms
- Observing the plot above, we can see that the divisive algorithm using **Euclidean/average** gives the highest cophenetic coefficient of .897
- We therefore take dendrogram #2 in the list as the one that maximizes the amount of variance explained within each cluster

Principal Component Analysis - Fitting

Original df: (660, 6)

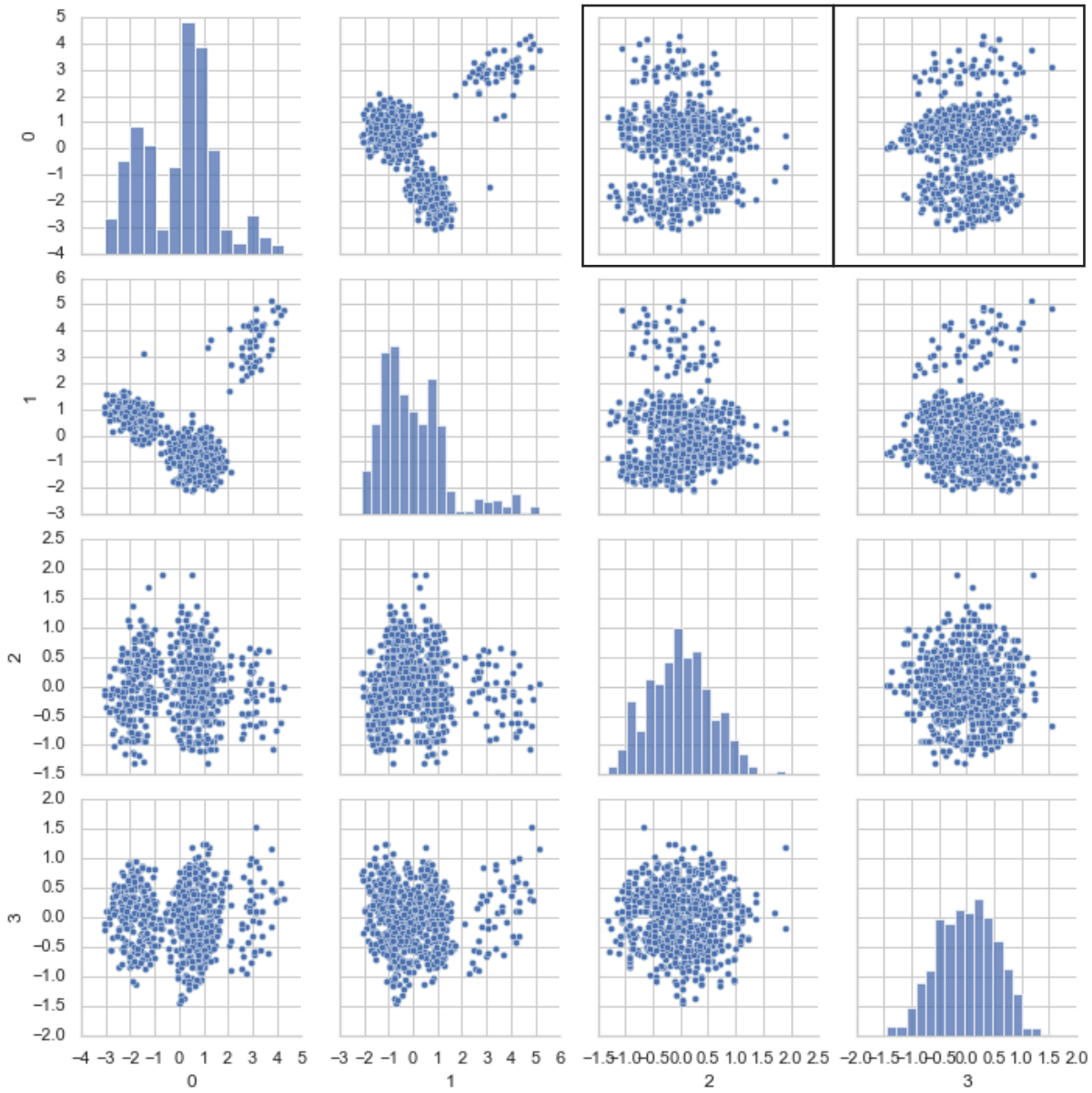
	avg_credit_limit	total_credit_cards	total_bank_visits	total_online_visits	total_calls_made	AG_GROUP
0	1.740187	-1.249225	-0.860451	-0.547490	-1.251537	0
1	0.410293	-0.787585	-1.473731	2.520519	1.891859	2
2	0.410293	1.058973	-0.860451	0.134290	0.145528	0
3	-0.121665	0.135694	-0.860451	-0.547490	0.145528	0
4	1.740187	0.597334	-1.473731	3.202298	-0.203739	1

Transform

PCA df: (660, 5)

	pc1	pc2	pc3	pc4	GROUP
0	0.501291	0.524829	1.895862	1.200582	0
1	-1.459560	3.105588	-0.906802	0.411052	2
2	0.525795	0.823979	0.089030	-1.033119	0
3	-0.362294	0.128123	0.558215	-0.729885	0
4	1.266228	3.668708	-0.099999	0.505571	1

Distinct bi-variate clusters



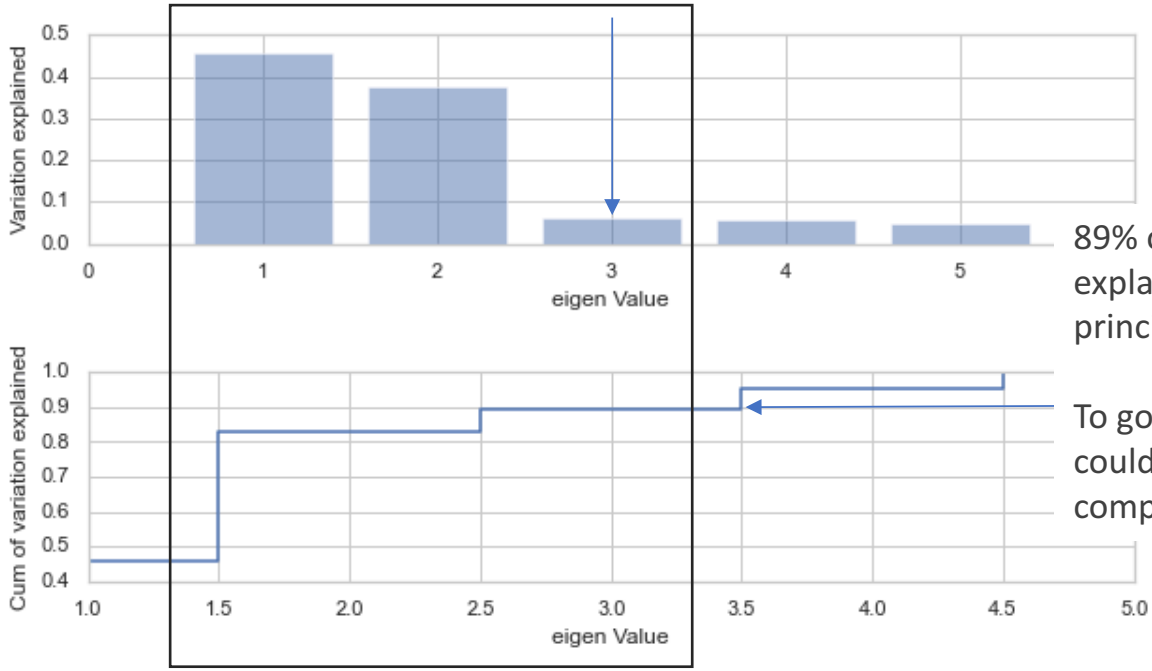
Observations on PCA fitting

- Principal components 1 and 3 and principal components 1 and 4 show the most distinct separation of clusters
- This indicates that pc3 and pc4 account for most of the variation among the clusters

Principal Component Analysis – Dimensional Reduction

Principal Components

	pc1	pc2	pc3	pc4	GROUP
0	0.501291	0.524829	1.895862	1.200582	0
1	-1.459560	3.105588	-0.906802	0.411052	2
2	0.525795	0.823979	0.089030	-1.033119	0
3	-0.362294	0.128123	0.558215	-0.729885	0
4	1.266228	3.668708	-0.099999	0.505571	1
5	-1.997304	0.665038	0.260150	-0.729732	2
6	1.147495	3.349963	0.369645	0.821562	1
7	-0.695324	0.064917	1.898606	-0.194751	2
8	-1.258442	0.248553	1.691161	0.091647	2
9	-1.747152	0.418565	0.361157	-1.129226	2
10	-1.113141	1.244488	0.414248	-0.473047	2
11	-1.854932	0.738637	0.389683	-0.596368	2
12	-2.364975	0.806227	-0.880027	0.778877	2
13	-1.977146	0.910306	0.198443	0.736248	2
14	-1.606327	0.305570	-0.205700	0.533929	2
15	-2.081541	1.351322	0.208921	0.030433	2
16	-1.346965	0.152206	0.847559	0.282234	2
17	-1.571898	0.663617	1.119221	0.060229	2
18	-1.812809	-0.016226	-0.231463	0.275613	2
19	-1.409952	1.215485	-0.663577	-0.372958	2

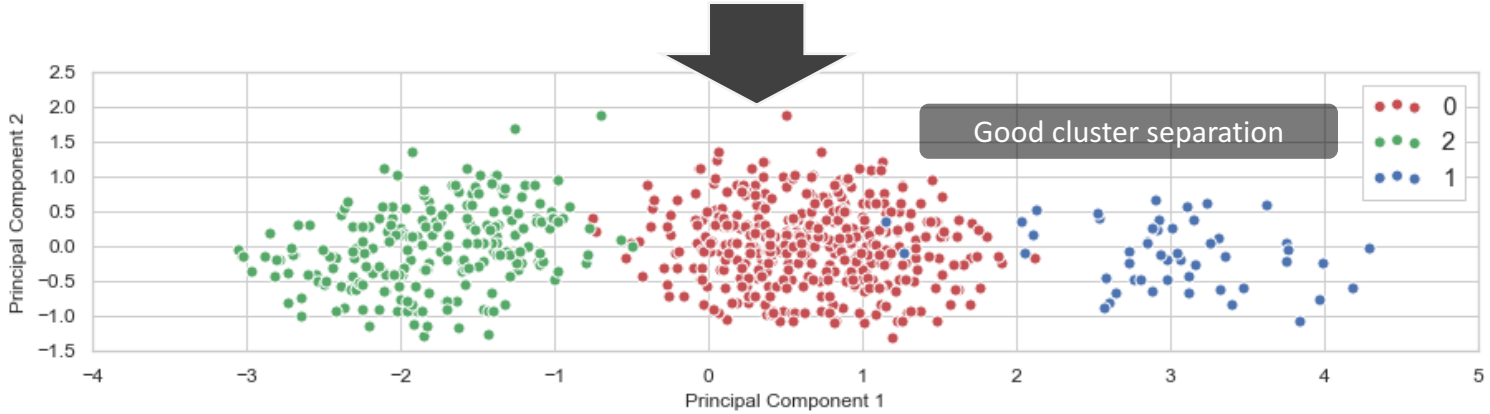


89% of variance explained with only 3 principal components

To go above 90% we could add a 4th principal component to the model

Observations on PCA reduction

- We are able to reduce the size and dimensionality of the data with PCA with very small loss of efficacy in subsequent regression modeling performance compared with using the Z score standardized dimensions as model inputs



Linear Regression Modeling

Observations in Linear Regression modeling (z score features vs. PCA reduction)

- PCA allows us to reduce the number of dimensions used for modeling without significant loss of predictive efficacy
- We observe the principal components "pc3" and "pc4" describe over 80% of the variance
- We went from 5 (z score) features/dimensions to 4 principal components
- Principal component tuning has allowed us to drop 1 more dimension, for a total of 3, and still retain the ability to predict out of sample groups without significant additional error

```
* Principal components used:  Index(['pc2', 'pc3', 'pc4'], dtype='object')

        Linear Regression on pca_data_df with pc1 removed

                R^2 (in sample): 0.36925707241769035
                R^2 (out of sample): 0.344257670070644

* Principal components used:  Index(['pc1', 'pc3', 'pc4'], dtype='object')

        Linear Regression on pca_data_df with pc2 removed

                R^2 (in sample): 0.518523999201375
                R^2 (out of sample): 0.42641083451413386

* Principal components used:  Index(['pc1', 'pc2', 'pc4'], dtype='object')

        Linear Regression on pca_data_df with pc3 removed

                R^2 (in sample): 0.8538150961535741
                R^2 (out of sample): 0.848548139662444

* Principal components used:  Index(['pc1', 'pc2', 'pc3'], dtype='object')

        Linear Regression on pca_data_df with pc4 removed

                R^2 (in sample): 0.8518709431676426
                R^2 (out of sample): 0.8443501561517556
```

Linear Regression on ag_df5: on 5 features

```
R^2 (in of sample): 0.8637389615261618
R^2 (out of sample): 0.8485060100241324
```

In sample R^2 is comparably close to linear regression results obtained by modeling the z scored features

Out of sample R^2 error between using fewer principal components vs. all z features is not statistically significant

Here we can see a runtime benefit of reducing dimensionality with PCA if we chose to make predictions in the future with supervised learning methods e.g. linear regression

LR on z scores, out-of-sample R^2: .848506 **VS** LR on PCA out-of-sample R^2: .84854

error (delta) = -3.399999999997849e-05)

Summary of Findings & Insights

Insights on Cluster Analysis / Segmentation, Dimension Reduction & Modeling

1. KMeans (top down clustering)
    1. Elbow Method identified that 3 as the optimal value for KMeans clustering
    2. Silhouette Visualization confirmed 3 as the optimal value for KMeans clustering
    3. Boxplot Visualization in addition confirmed 3 as the optimal number of clusters

2. Cophenetic Coef / Dendrograms (bottom up clustering)
    1. Euclidean (distance) average (method) gives the highest cophenetic coefficient of .897, and thus provides the best groupings

3. PCA (variation accounting)
    1. 5 source data dimensions were transformed into 4 principal components which desribe the clusters effectively
    2. PCA is effective in reducing dimensionality and size of model inputs
        1. This has greatest benefits when the scale of data analysis increases by orders of magnitude

4. Linear Regression (supervised prediction using cluster labels)
    1. We prefer use of principal components as model inputs to minimize data complexity and speed up run-time execution

Cluster Profiling is used differentiate clusters/groups by identifying the unique dimensional characteristics of each cluster/group. To accomplish this, we look for maximums and other summary statistics of the dimensions of the clusters.

Key Findings

1.Marketing

1. KMean, Agglomerative and Cophenetic methods all agree that the standardized dimensions separate into 3 distinct clusters/groups
2. Cluster analysis identified 3 distinct Customer Segments
    1. Segment / **Group 1 - Wealthy Tech Savvy** (High Credit, Online User, Low Support)
    2. Segment / **Group 2 - Low Credit User** (Low Credit, Caller, High Support)
    3. Segment / **Group 3 - Bank Traditionalist** (Medium Credit, Bank Visitor, Medium Support)
3. By using Linear Regression, we have the ability to predict a customer segment (post clustering)

2.Service delivery

1. Low Credit User segment is the group that consumes the majority of support delivery (operational improvement initiatives)
2. Wealthy Tech Savvy segment like self-sufficient technology (digital transformation investment - Digital Experience)

Business Strategy Recommendations

1. Growth
    a. As part of a broader Digital Transformation strategy, bank website should be modernized and upgraded to provide a better user experience (UX) and more value than what competitors currently have
    Invest in development of deeper backend analytics to better understand customer behaviors
2. Marketing
    a. Segment / Group 1 - Wealthy Tech Savvy (High Credit, Online User, Low Support)
        - New User & Convert Existing Campaign
            - Social Media Awareness (Linkedin, Twitter, Instagram)
            - YouTube Ads (tag to videos focused wealthy tech savvy individuals)
    b. Segment / Group 2 - Low Credit User (Low Credit, Caller, High Support)
        - Make Happy Campaign
            - Email Newsletter messaging "what we're up to at AllBank" on improved customer support efficiencies for our valued customers
    c. Segment / Group 3 - Bank Traditionalist (Medium Credit, Bank Visitor, Medium Support)
        - Tradition Campaign
            - Upgrade in bank customer experience with smart and clean credit card offer banners (high-lighting our dedication to the customer experience and the bank's technology and support assets)
3. Service Delivery
    1. Defer capital investment and focus on removing waste in service delivery processes
    2. Identify all value chain service delivery processes and rank by cost to maintain
    3. Run Kaizen (improvement) projects based on cost rank with run bi-weekly strategy/risk reviews with leadership