D212 Data Mining II Performance Task 1

Sean Simmons

WDU Data Analytics

MSDA D212

February 2023

“Scenario 1

One of the most critical factors in customer relationship management that directly affects a company’s long-term profitability is understanding its customers. When a company can better understand its customer characteristics, it is better able to target products and marketing campaigns for customers, resulting in better profits for the company in the long term.

You are an analyst for a telecommunications company that wants to better understand the characteristics of its customers. You have been asked to use principal component analysis (PCA) to analyze customer data to identify the principal variables of your customers, ultimately allowing better business and strategic decision-making.”

Part I: Research Question

A. Describe the purpose of this data mining report by doing the following:

1. Propose **one** question relevant to a real-world organizational situation that you will answer using **one** of the following clustering techniques:

   - *k*-means
   - hierarchical

2. Define **one** goal of the data analysis. Ensure that your goal is reasonable within the scope of the scenario and is represented in the available data.

1. One research question relevant to the telecommunications company in scenario one is, "Can we identify and summarize unique clusters from the continuous variables in the customer dataset using kmeans clustering?" I will use kmeans clustering to answer this question.

2. One goal of this analysis is to identify and summarize distinct clusters according to the continuous variables in the customer dataset using k-means clustering. This goal is relevant in scenario one because doing so would help the organization to better understand its customers characteristics. It is also reasonable within the scope of the data because we are provided 10,000 rows of customer information that includes 10 unique continuous variables.

Part II: Technique Justification

B. Explain the reasons for your chosen clustering technique from part A1 by doing the following:

1. Explain how the clustering technique you chose analyzes the selected dataset. Include expected outcomes.
2. Summarize **one** assumption of the clustering technique.

3. List the packages or libraries you have chosen for Python or R, and justify how *each* item on the list supports the analysis.

1. K-means clustering is an unsupervised machine learning technique designed to identify clusters of variables in a dataset through grouping based on similar properties. Specifically, k-means uses expectation-maximization to group variables into clusters by similar characteristics and identify relevance to a specific center variable (Arvai 2023). In this specific dataset, k-means will be used to identify and summarize distinct clusters. We expect to see, with some value of confidence from the distortion number, the optimal number of clusters.
2. One assumption of this clustering technique is that all variables have the same variance, and it is a spherical distribution in nature.
3. The packages I have chosen to use in my python code are as follows:
   a. Pandas- To import and handle the churn dataset given to me by the organization.
   b. Numpy - To perform mathematical operations with the dataset
   c. Matplotlib, including subpackages - To perform variable analysis and provide visualizations for the models, including the scree plot and histograms.
   d. Sklearn, including subpackages - To perform K-means classification and modeling. This has subpackages that allow the data to be split into training and testing sets and then fit to the K-means model and tested for accuracy. Essentially, these packages performed the entirety of the data modeling and analysis.
   e. Yellowbrick - To run the elbow method calculations to determine the optimal number of clusters

f. Visualizer - To visualize and fit our final model with the distortion score to find the optimal number of clusters.

Part III: Data Preparation

C. Perform data preparation for the chosen dataset by doing the following

1. Describe **one** data preprocessing goal relevant to the clustering technique from part A1.

2. Identify the initial dataset variables that you will use to perform the analysis for the clustering question from part A1, and label *each* as continuous or categorical.

3. Explain *each* of the steps used to prepare the data for the analysis. Identify the code segment for *each* step.

4. Provide a copy of the cleaned dataset.

1. One data preprocessing goal relevant to the k-means clustering technique used is to scale our continuous variables. I scaled the variables using the standardscaler() package.

2. The initial variables I will use to perform the analysis for the clustering and their labels are the continuous variables from the customer dataset, listed below:

Variable Name	Data Type
Outage_Sec_perweek	Continuous

Tenure	Continuous
MonthlyCharge	Continuous
Bandwidth_GB_year	Continuous
Email	Continuous
Yearly_equip_failure	Continuous
Contacts	Continuous
Children	Continuous
Age	Continuous
Income	Continuous

3. The steps used to prepare the data analysis are labeled alphabetically, which are labeled the same way in the code file. Here is a summary of the steps:
   a. Import the data into my coding environment
   b. View the data type and summary information to prepare for the modeling
   c. Check for missing values
   d. Drop noncontinuous columns
   e. Check summary statistics, such as mean/median
   f. Perform univariate analysis of the continuous variables through their histograms
   g. Standardize the values using the standardscaler package
   h. Extract data set to csv file
   i. List features that will be used in the dataset (all remaining features)
4. The copy of the cleaned data set is “D212_churn_Task1.csv”

Initial analysis from the code:

```
Floats
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 7 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   Lat                10000 non-null  float64
 1   Lng                10000 non-null  float64
 2   Income             10000 non-null  float64
 3   Outage_sec_perweek 10000 non-null  float64
 4   Tenure             10000 non-null  float64
 5   MonthlyCharge      10000 non-null  float64
 6   Bandwidth_GB_Year  10000 non-null  float64
dtypes: float64(7)
memory usage: 547.0 KB
None
Integers
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 16 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   CaseOrder            10000 non-null  int64
 1   Zip                  10000 non-null  int64
 2   Population           10000 non-null  int64
 3   Children             10000 non-null  int64
 4   Age                  10000 non-null  int64
 5   Email                10000 non-null  int64
 6   Contacts             10000 non-null  int64
 7   Yearly_equip_failure 10000 non-null  int64
 8   Item1                10000 non-null  int64
 9   Item2                10000 non-null  int64
 10  Item3                10000 non-null  int64
 11  Item4                10000 non-null  int64
 12  Item5                10000 non-null  int64
 13  Item6                10000 non-null  int64
 14  Item7                10000 non-null  int64
 15  Item8                10000 non-null  int64
dtypes: int64(16)
memory usage: 1.2 MB
None
```

```
Objects
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 27 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Customer_id       10000 non-null  object
 1   Interaction       10000 non-null  object
 2   UID               10000 non-null  object
 3   City              10000 non-null  object
 4   State             10000 non-null  object
 5   County            10000 non-null  object
 6   Area              10000 non-null  object
 7   TimeZone          10000 non-null  object
 8   Job               10000 non-null  object
 9   Marital           10000 non-null  object
 10  Gender            10000 non-null  object
 11  Churn             10000 non-null  object
 12  Techie            10000 non-null  object
```

PA_D209_C

```
 13  Contract          10000 non-null  object
 14  Port_modem        10000 non-null  object
 15  Tablet            10000 non-null  object
 16  InternetService   10000 non-null  object
 17  Phone             10000 non-null  object
 18  Multiple          10000 non-null  object
 19  OnlineSecurity    10000 non-null  object
 20  OnlineBackup      10000 non-null  object
 21  DeviceProtection  10000 non-null  object
 22  TechSupport       10000 non-null  object
 23  StreamingTV       10000 non-null  object
 24  StreamingMovies   10000 non-null  object
 25  PaperlessBilling  10000 non-null  object
 26  PaymentMethod     10000 non-null  object
dtypes: object(27)
memory usage: 2.1+ MB
None
```

```
Dataset Information
<bound method DataFrame.info of       CaseOrder Customer_id
Interaction  \
0             1     K409198  aa90260b-4141-4a24-8e36-b04ce1f4f77b
1             2     S120509  fb76459f-c047-4a9d-8af9-e0f7d4ac2524
2             3     K191035  344d114c-3736-4be5-98f7-c72c281e2d35
3             4      D90850  abfa2b40-2d43-4994-b15a-989b8c79e311
4             5     K662701  68a861fd-0d20-4e51-a587-8a90407ee574
...         ...         ...                                   ...
9995       9996     M324793  45deb5a2-ae04-4518-bf0b-c82db8dbe4a4
9996       9997     D861732  6e96b921-0c09-4993-bbda-a1ac6411061a
9997       9998     I243405  e8307ddf-9a01-4fff-bc59-4742e03fd24f
9998       9999     I641617  3775ccfc-0052-4107-81ae-9657f81ecdf3
9999      10000      T38070  9de5fb6e-bd33-4995-aec8-f01d0172a499

                                   UID          City State  \
0     e885b299883d4f9fb18e39c75155d990   Point Baker    AK
1     f2de8bef964785f41a2959829830fb8a  West Branch    MI
2     f1784cfa9f6d92ae816197eb175d3c71       Yamhill    OR
3     dc8a365077241bb5cd5ccd305136b05e       Del Mar    CA
4     aabb64a116e83fdc4befc1fbab1663f9    Needville    TX
...                                ...           ...   ...
9995  9499fb4de537af195d16d046b79fd20a   Mount Holly    VT
9996  c09a841117fa81b5c8e19afec2760104   Clarksville    TN
9997  9c41f212d1e04dca84445019bbc9b41c      Mobeetie    TX
9998  3e1f269b40c235a1038863ecf6b7a0df    Carrollton    GA
9999  0ea683a03a3cd544aefe8388aab16176  Clarkesville    GA

                    County    Zip       Lat        Lng  ...  MonthlyCharge  \
0     Prince of Wales-Hyder  99927  56.25100 -133.37571  ...     172.455519
1                   Ogemaw  48661  44.32893  -84.24080  ...     242.632554
2                  Yamhill  97148  45.35589 -123.24657  ...     159.947583
3                San Diego  92014  32.96687 -117.24798  ...     119.956840
4                Fort Bend  77461  29.38012  -95.80673  ...     149.948316
...                    ...    ...       ...        ...  ...            ...
9995               Rutland   5758  43.43391  -72.78734  ...     159.979400
9996            Montgomery  37042  36.56907  -87.41694  ...     207.481100
9997               Wheeler  79061  35.52039 -100.44180  ...     169.974100
9998               Carroll  30117  33.58016  -85.13241  ...     252.624000
9999             Habersham  30523  34.70783  -83.53648  ...     217.484000

     Bandwidth_GB_Year Item1 Item2  Item3  Item4  Item5 Item6 Item7 Item8
0           904.536110     5     5      5      3      4     4     3     4
```

1	800.982766	3	4	3	3	4	3	4	4
2	2054.706961	4	4	2	4	4	3	3	3
3	2164.579412	4	4	4	2	5	4	3	3
4	271.493436	4	4	4	3	4	4	4	5
...	...	...	...	...	...	...	...	...	...
9995	6511.252601	3	2	3	3	4	3	2	3
9996	5695.951810	4	5	5	4	4	5	2	5
9997	4159.305799	4	4	4	4	4	4	4	5
9998	6468.456752	4	4	6	4	3	3	5	4
9999	5857.586167	2	2	3	3	3	3	4	1

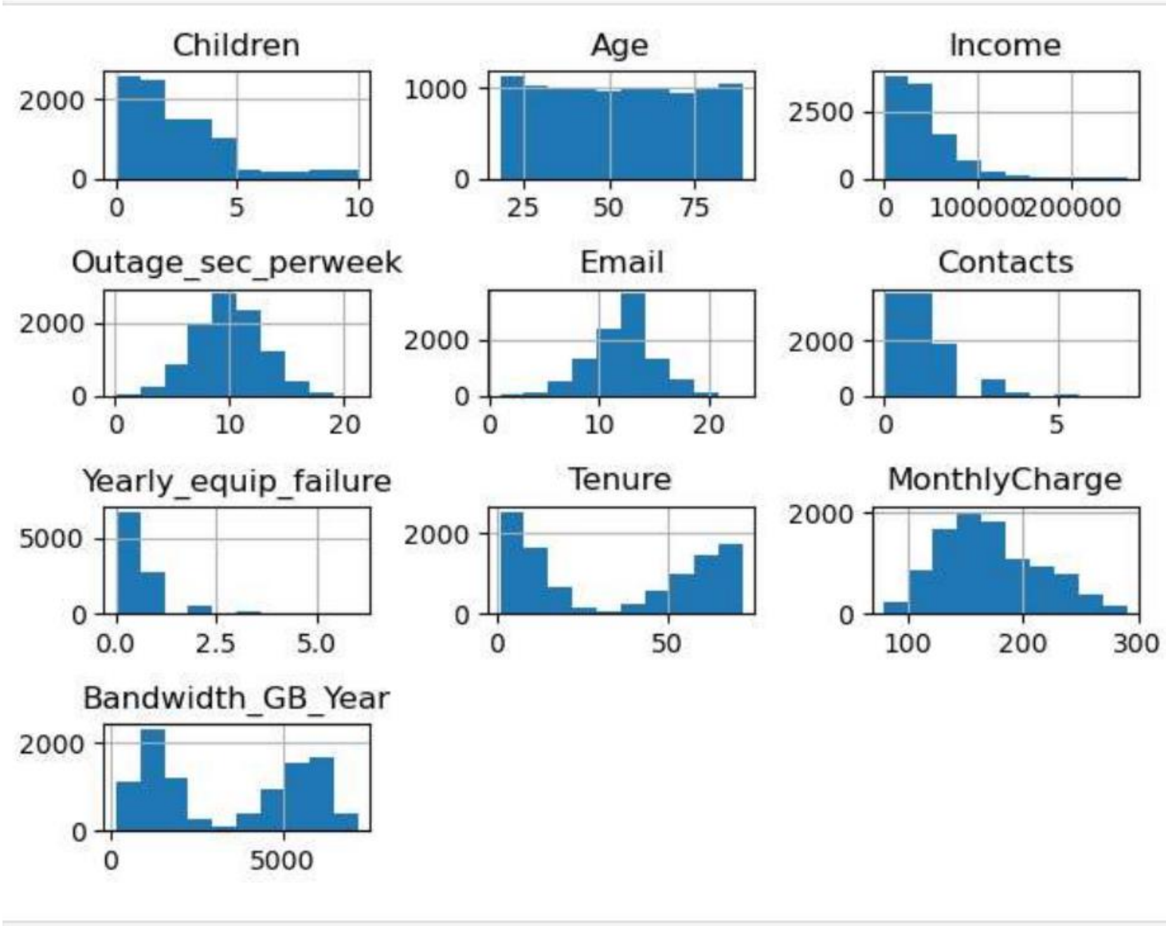
Missing Values:

```
CaseOrder                0
Customer_id              0
Interaction              0
UID                      0
City                     0
State                    0
County                   0
Zip                      0
Lat                      0
Lng                      0
Population               0
Area                     0
TimeZone                 0
Job                      0
Children                 0
Age                      0
Income                   0
Marital                  0
Gender                   0
Churn                    0
Outage_sec_perweek       0
Email                    0
Contacts                 0
Yearly_equip_failure     0
Techie                   0
Contract                 0
Port_modem               0
Tablet                   0
InternetService          0
Phone                    0
Multiple                 0
OnlineSecurity           0
OnlineBackup             0
DeviceProtection         0
TechSupport              0
StreamingTV              0
StreamingMovies          0
PaperlessBilling         0
PaymentMethod            0
Tenure                   0
MonthlyCharge            0
Bandwidth_GB_Year        0
item1_responses          0
item2_fixes              0
item3_replacements       0
item4_reliability        0
item5_options            0
item6_respectfulness     0
item7_courteous          0
item8_listening          0
dtype: int64
```

```
#e: Summary statistics
print (churn_df.mean())
#Median Values in the Distribution)
print (churn_df.median())
```

```
Children                    2.087700
Age                        53.078400
Income                  39806.926771
Outage_sec_perweek         10.001848
Email                      12.016000
Contacts                    0.994200
Yearly_equip_failure        0.398000
Tenure                     34.526188
MonthlyCharge             172.624816
Bandwidth_GB_Year        3392.341550
dtype: float64
Children                    1.000000
Age                        53.000000
Income                  33170.605000
Outage_sec_perweek         10.018560
Email                      12.000000
Contacts                    1.000000
Yearly_equip_failure        0.000000
Tenure                     35.430507
MonthlyCharge             167.484700
Bandwidth_GB_Year        3279.536903
dtype: float64
```

Univariate analysis



Part IV: Analysis

D. Perform the data analysis and report on the results by doing the following:

1. Describe the analysis technique you used to appropriately analyze the data. Include screenshots of the intermediate calculations you performed.

2. Provide the code used to perform the clustering analysis technique from part 2.

1. The K-means clustering method used here was facilitated through the packages in sklearn. The data first had to be scaled to work in this model using the standard scaler package. After retrieving the results, the analysis technique I used to appropriately analyze the data is the elbow method. The elbow method is used to find the optimal k value by finding where the clustering method falls off when adding more cluster groups to refine the results and looking at the distortion value. The 10 variables were all continuous variables in the dataset; children, age, income, outage_sec_perweek, email, contacts, yearly_equip_failure, tenure, monthlycharge, and bandwidth_GB_Year. The calculations are shown below:
    a. **Distortion:** The average of the squared distances from the cluster centers of the respective clusters. The Euclidean distance metric is used, seen in the distortion score with elbow method for kmeans clustering graph.
    b. **Inertia:** The sum of squared distances of samples to their closest cluster center. We want a low inertia for our model.

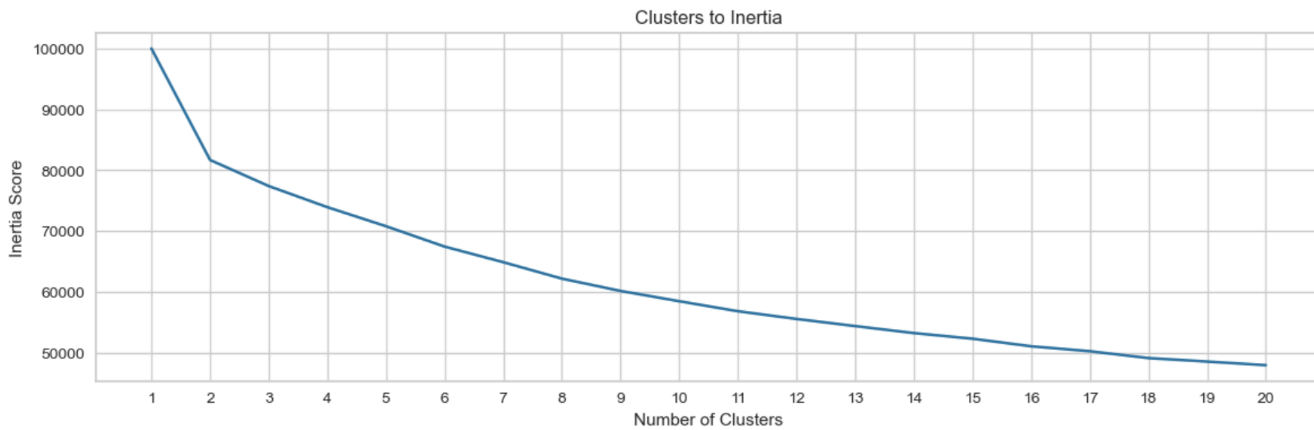
```
#Use the standardscaler package to standardize our values
num_col = churn_df.columns[churn_df.dtypes.apply(lambda c: np.issubdtype(c, np.number))]
scaler = StandardScaler()
churn_df[num_col] = scaler.fit_transform(churn_df[num_col])
#Check for scaling
print(churn_df)
```

```
      Children       Age    Income  Outage_sec_perweek     Email  Contacts  \
0    -0.972338  0.720925 -0.398778           -0.679978 -0.666282 -1.005852
1    -0.506592 -1.259957 -0.641954            0.570331 -0.005288 -1.005852
2     0.890646 -0.148730 -1.070885            0.252347 -0.996779 -1.005852
3    -0.506592 -0.245359 -0.740525            1.650506  0.986203  1.017588
4    -0.972338  1.445638  0.009478           -0.623156  1.316700  1.017588
...        ...       ...       ...                 ...       ...       ...
9995  0.424900 -1.453214  0.564456           -0.196888 -0.005288  1.017588
9996  0.890646 -0.245359 -0.201344           -1.095915  0.986203  1.017588
9997 -0.506592 -0.245359  0.219037           -1.146198 -0.666282 -1.005852
9998 -0.506592 -0.680187 -0.820588            0.695616  0.655706  0.005868
9999 -0.506592 -1.211643 -1.091760            0.589028  1.647197  0.005868

      Yearly_equip_failure    Tenure  MonthlyCharge  Bandwidth_GB_Year  ...  \
0                 0.946658 -1.048746      -0.003943          -1.138487  ...
1                 0.946658 -1.262001       1.630326          -1.185876  ...
2                 0.946658 -0.709940      -0.295225          -0.612138  ...
3                -0.625864 -0.659524      -1.226521          -0.561857  ...
4                 0.946658 -1.242551      -0.528086          -1.428184  ...
...                    ...       ...            ...                ...  ...
9995             -0.625864  1.273401      -0.294484           1.427298  ...
9996             -0.625864  1.002740       0.811726           1.054194  ...
9997             -0.625864  0.487513      -0.061729           0.350984  ...
9998             -0.625864  1.383018       1.863005           1.407713  ...
9999             -0.625864  1.090120       1.044672           1.128163  ...
```

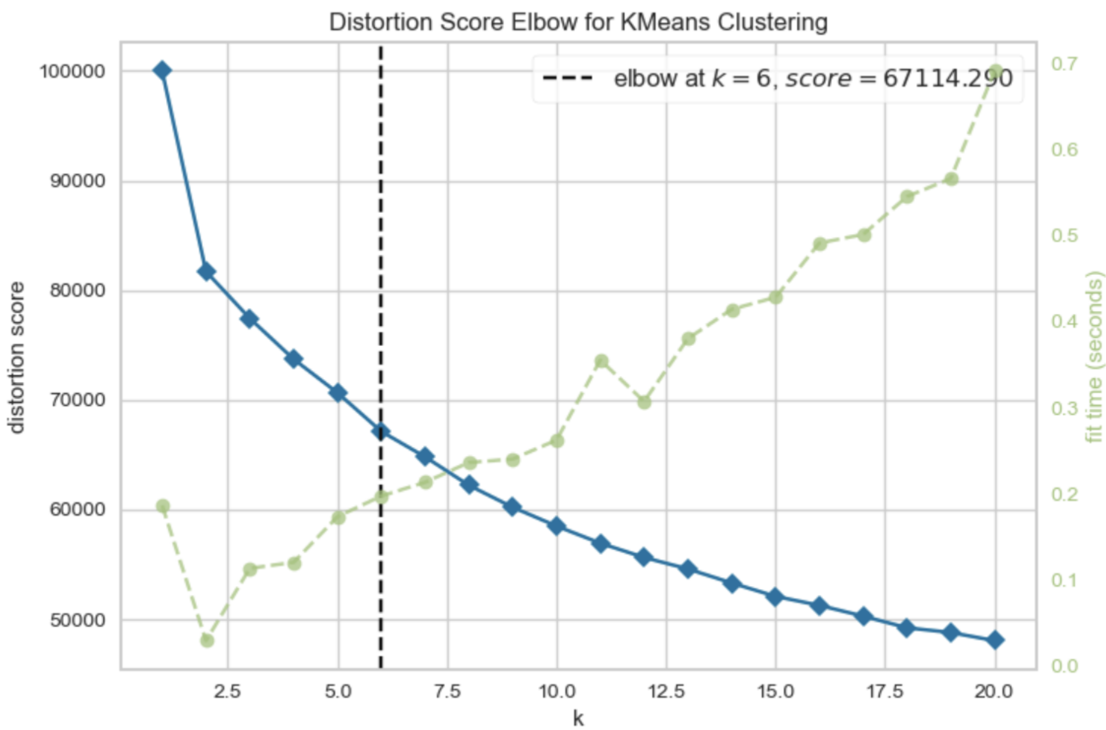
```
# Running K means with 20 clusters and showing the intertia's for all variable
no_of_clusters = range(1,21)
inertia = []

for n in no_of_clusters:
    kmeans = KMeans(n_clusters=n, random_state=540)
    kmeans = kmeans.fit(churn_df)
    i = kmeans.inertia_
    inertia.append(i)
    print("The innertia for :", n, "Clusters is:", i)
```

```
The innertia for : 1 Clusters is: 99999.99999999996
The innertia for : 2 Clusters is: 81703.413078988
The innertia for : 3 Clusters is: 77425.4402132015
The innertia for : 4 Clusters is: 73959.82018226922
The innertia for : 5 Clusters is: 70833.22073670026
The innertia for : 6 Clusters is: 67487.58432099453
The innertia for : 7 Clusters is: 64931.872782177554
The innertia for : 8 Clusters is: 62231.800062614515
The innertia for : 9 Clusters is: 60215.56940972775
The innertia for : 10 Clusters is: 58529.625701689525
The innertia for : 11 Clusters is: 56886.418556358716
The innertia for : 12 Clusters is: 55621.522667391066
The innertia for : 13 Clusters is: 54450.59410218633
The innertia for : 14 Clusters is: 53302.211625020325
The innertia for : 15 Clusters is: 52380.42953437563
The innertia for : 16 Clusters is: 51136.34551451661
The innertia for : 17 Clusters is: 50327.10139537831
The innertia for : 18 Clusters is: 49201.84730703119
The innertia for : 19 Clusters is: 48631.67281016224
The innertia for : 20 Clusters is: 48038.9004242199
```

```
# Creating the scree plot for Intertia - elbow method
fig, (ax1) = mpl.subplots(1, figsize=(14,4))
num = np.arange(len(no_of_clusters))
ax1.plot(num, inertia)
ax1.set_xticks(num)
ax1.set_xticklabels(no_of_clusters)
mpl.xlabel('Number of Clusters')
mpl.ylabel('Inertia Score')
mpl.title("Clusters to Inertia")
mpl.show()
```



```
model = KMeans()
#k is range of number of clusters. We are using a wide range to find the ideal number of clusters
visualizer = KElbowVisualizer(model, k=(1,21), timings= True,figsize=(20,10))
visualizer.fit(churn_df)
visualizer.show()
```




```
<AxesSubplot:title={'center':'Distortion Score Elbow for KMeans Clustering'}, xlabel='k', ylabel='distortion score'>
```

```
#From this, we gather the best number of clusters is 6
```

We can see the distortion value is 67114.290 and the final number of clusters is 6.

2. The code used to perform the clustering analysis technique from part 2 is included in "D212_Task1_Code.ipynb"

Part V: Data Summary and Implications

E. Summarize your data analysis by doing the following:

1. Explain the accuracy of your clustering technique.

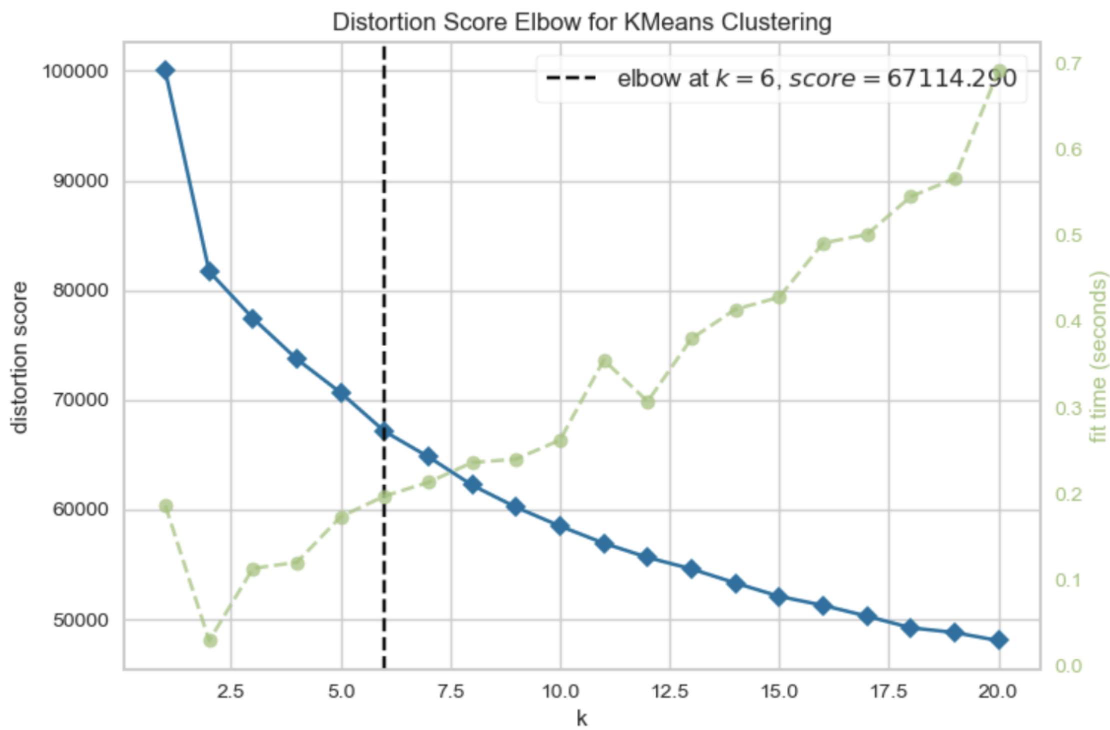
2. Discuss the results and implications of your clustering analysis.

3. Discuss **one** limitation of your data analysis.

4. Recommend a course of action for the real-world organizational situation from part A1 based on your results and implications discussed in part E2.

1. K-means does not lend itself to having a simple accuracy percentage, due to the nature of the learning algorithm and clustering. Therefore, the accuracy of my clustering technique is measured by the elbow method of the best number of clusters to use and the fit from the scree plot from above using distortion and inertia. We can see that 6 (k) is the best number of clusters to use and provides the most accurate clustering available through this method from the following accuracy calculation:

```
model = KMeans()
#k is range of number of clusters. We are using a wide range to find the ideal number of clusters
visualizer = KElbowVisualizer(model, k=(1,21), timings= True,figsize=(20,10))
visualizer.fit(churn_df)
visualizer.show()
```




```
<AxesSubplot:title={'center':'Distortion Score Elbow for KMeans Clustering'}, xlabel='k', ylabel='distortion score'>
```

```
#From this, we gather the best number of clusters is 6
```

2. The results of my clustering analysis show that we have identified 6 distinct clusters with k-means clustering. Using more than 6 will lower the accuracy, and using less clusters will not provide enough information. The implications are that we need to use this clustering method in response to specific organizational needs for information about customer characteristics.

   Centroids are datapoints representing the center of the cluster, inertia is the sum of squared distances of samples to their closest cluster center. As more centroids are added to the model, the distance from each point to its closest centroid will decrease. The elbow point, seen in the graph earlier in this report, is the point where utility is lost. We want a low inertia for our model and at 6 clusters we have a much lower inertia value than 1 cluster and after 6 it does not seem to drop drastically as clusters are added. The implication is that if more clusters are added, it is not brining enough utility to our model to justify their additions.

The image below shows our centroids. There are six clusters, and the given vectors of each are the centers of those clusters. The implications are that for a new datapoint, you could check to see which centroid is the closest and you can determine the new point cluster from this method.

```
kmeans.cluster_centers_
```

```
array([[-0.24939141,  0.03711216, -0.2143031 ,  0.01458996,  0.00602228,
         0.01753045, -0.62586353,  0.97892015,  0.01757937,  0.96329265],
       [-0.19016244, -0.87139478, -0.19182939,  0.0347376 ,  0.03352157,
        -0.09454968, -0.07701984, -0.96581335, -0.05529049, -0.94376698],
       [-0.1633452 ,  0.0409786 ,  2.36007471, -0.08326729, -0.1028044 ,
        -0.02114724, -0.10938787, -0.04905871, -0.07002836, -0.05637491],
       [-0.19475816,  0.87358519, -0.23655372, -0.03075561,  0.02332034,
         0.11908768, -0.0520549 , -0.96405336,  0.07356023, -0.98429463],
       [ 2.59479546, -0.1522096 , -0.04879073, -0.00342237,  0.04365792,
        -0.06973608, -0.05112404,  0.0987422 , -0.06818963,  0.18105469],
       [-0.18627526,  0.01441845, -0.1535159 ,  0.00941905, -0.06161765,
        -0.01838097,  1.41883312,  0.88885447,  0.01243503,  0.87773115]])
```

3. One limitation of my data analysis is that the number of clusters can vary and limits the performance of this model. The qualitative aspect of choosing the right number of clusters and possibly limiting the information we get or lowering accuracy makes the k-means clustering method a little less reliable than one would want.

4. A course of action I recommend is based on the clusters identified. The organization should put more resources into analyzing the model with 6 clusters to investigate how

they can understand customer characteristics and therefore, be more informed. I would recommend more investigation be put into cluster methods and running the model many more times to fine tune the results.

Part VI: Demonstration

F. Provide a Panopto video recording that includes a demonstration of the functionality of the code used for the analysis and a summary of the programming environment.

Link to my panopto Video:

https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=e50fb572-0dc5-4878-89c7-afb701779f85

G. Record the web sources used to acquire data or segments of third-party code to support the analysis. Ensure the web sources are reliable.

References

Arvai, K. (n.d.). *K-Means Clustering in Python: A Practical Guide – Real Python*. Realpython.com. https://realpython.com/k-means-clustering-python/

H. Acknowledge sources, using in-text citations and references, for content that is quoted, paraphrased, or summarized.

References

Arvai, K. (n.d.). *K-Means Clustering in Python: A Practical Guide – Real Python*. Realpython.com. https://realpython.com/k-means-clustering-python/

I. Demonstrate professional communication in the content and presentation of your submission.

This aspect of the rubric is evaluated through the entirety of this report and I hope professionalism has shown continuously.