# D209 Data Mining 2

Professor Keiona Middleton

Mackenzie Simon

# Part 1 Research Question:

A1: As an analyst, our goal when looking at the Telecommunications Churn Data set is to figure out how to predict customer churn. We will figure this out by focusing on our K means clustering classification model when targeting our dependent variable Churn. From creating a valid model, we can then use this for future data in order to predict whether a customer will churn. After running our k nearest means classifier, we can answer what to invest in to minimize customer churn.

A2: The goal of our analysis is to build a k means model for predicting customer churn. Once we build our k nearest neighbor model, we can then test to see if our model is valid. If our model is valid, we know what variables will help us in predicting churn.

# Part 2 Method Justification

B1: K means is an unsupervised learning algorithm which allows us to determine how to label our data. Our K means model works by classifying and assigning each data point to a cluster. Clustering is the process of grouping data samples together based on their shared features. Through grouping the data, we establish the training phase of the algorithm. New data is then assigned to a cluster based on the distance from the mean of each cluster. Al-Masri (2019) describes K means as using the distance between points as a measure of similarity based on k averages. We expect our K means model to accurately predict Churn based on our input parameters.

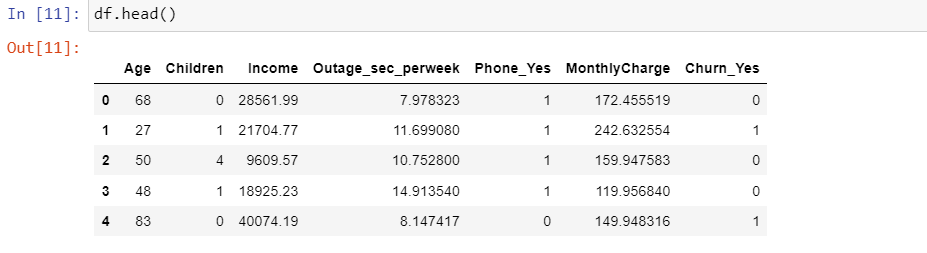
B2: K means is a non-parametric based algorithm and does not require any assumptions on data distribution. The k means model is typically used with unsupervised data.

B3: Some of the benefits of using Python are the libraries created for data analytics. In this project we used NumPy for indexing and arrays, Pandas for data formatting, and SciKit for data modeling. With SciKit learn we scale our data, construct a training/test model, build a K means algorithim, create a classification report and confusion matrix. Zhidkov, R (2021) described Python is also open source and has a large community for problem solving. Python also is extremely useful when analyzing large datasets due to its speed and processing. Python is useful because our data has 10,000 rows and 50 columns.

# Part 3 Data Preparation:

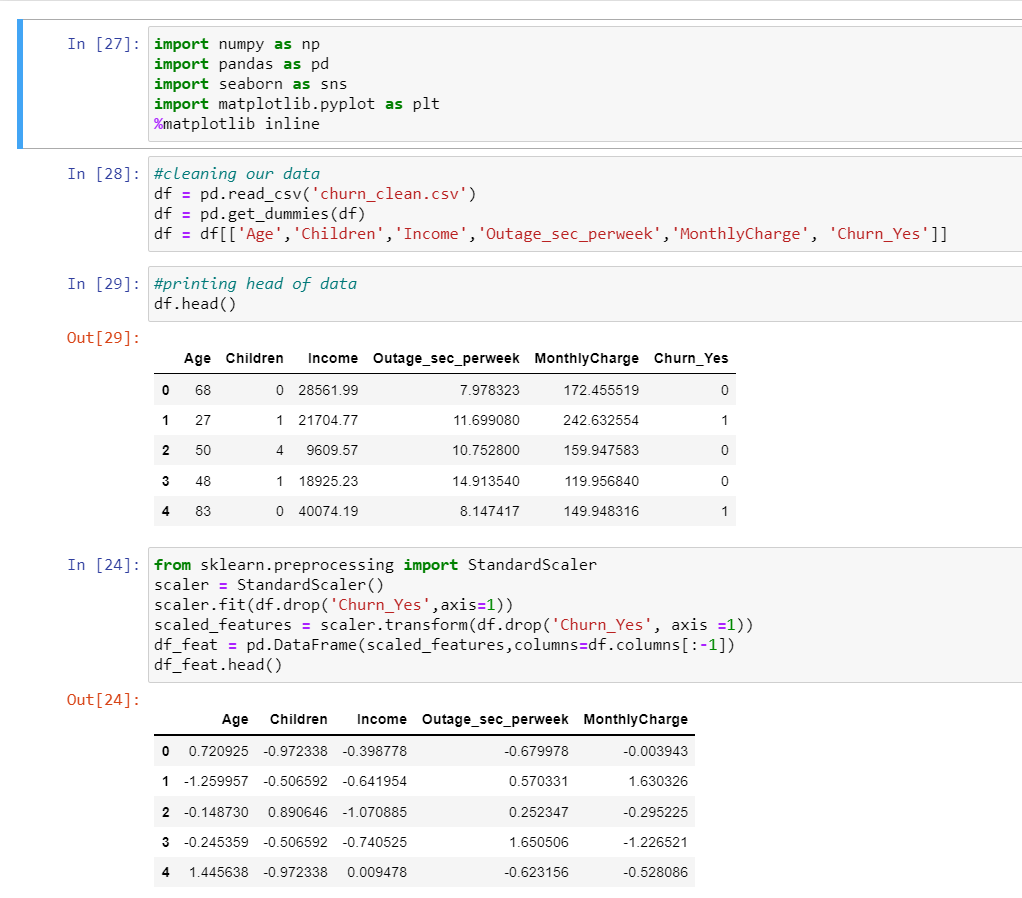
C1: One important pre-classification step for our K means model is that we must scale our data. Data scaling can be achieved by normalizing or standardizing input and output variables.

C2: Our variables for the K means model are listed below. We will use Age (continuous), Income (continuous), outage seconds per week (continuous), Monthly Charge (Continuous) as our X or predictor variables. We will use Churn (categorical) as our dependent or target variable



C3:

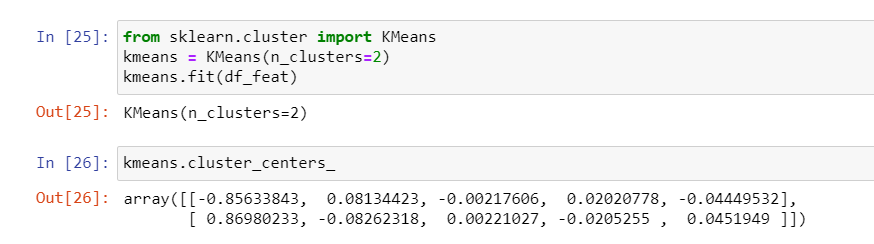
For my data preparation I used the panda’s library to import the csv data. I started by converting all the binary columns into dummy variables. I then looked through the data and choose what columns I thought would be useful for predicting churn. I made sure that these variables were continuous and would be useful for clustering. I then chose to drop any of the columns I deemed not useful to our model. I proceeded to standardize our data using the preprocessing package from sklearn making sure to avoiding scaling our target variable Churn\_Yes. In order to make this work, we had to make a scaler model and fit our data to the model. We then transformed our fit model and set the scaled data to a variable called df\_feat.



# Part 4 Analysis:

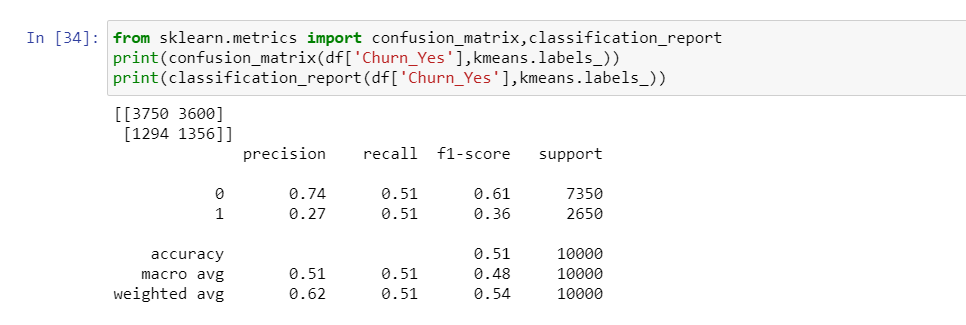
D1: For our k means algorithm, we import Kmeans from Sci Kit Learn. After we have our library imported, we can then set our k means to the number of clusters. For our model we used N\_clusters=2 because our target variable Churn\_Yes is binary. We then fit k means model to the scaled data df\_feat. After we have our k means model setup, we can view the cluster centers. Once we have model completed, we can check the accuracy by using a confusion matrix and classification report from sklearn.

D2:



# Part 5 Data Summary and Implications:

E1: From our classification report, we have a macro average of .51 for precision, .51 for recall, and .48 for F-1 score. Overall, our accuracy for F1 is .51. I would suggest looking for a target variable that allows us for higher levels of clustering. I would avoid using this model.



E2: From our results, I would avoid using K means clustering for our data set. The issue with using K means clustering is we typically use the algorithm with unsupervised data. In our case, we have labeled data so in a sense we are working backwards. There are other algorithms that allow us to build off supervised learning data and give us a better model. Another issue is k means algorithms are heavily influenced by outliers. Data must be relatively similar or else our algorithm will classify data points wrong.

3. One of the limitations for our data analysis is we are only trying to use two clusters for predicting Churn because our target variable is binary. This model would be a lot better if we were targeting a variable that allows us to use more clusters given the high dimensionality of the data.

4. A course of action would be to avoid using K means modeling for this data set. I would recommend reducing the variables down even further given the high dimensionality. An alternative suggestion would be to use an algorithm that works with labeled data such as K nearest neighbors.

# References

Al-Masri, A. (2019, May 15). *How Does k-Means Clustering in Machine Learning Work?* Medium. https://towardsdatascience.com/how-does-k-means-clustering-in-machine-learning-work-fdaaaf5acfa0.

Zhidkov, R. (2021, January 10). *Why Python is Essential for Data Analysis*. RTInsights. https://www.rtinsights.com/why-python-is-essential-for-data-analysis/#:~:text=The%20object%2Doriented%20programming%20language,streamline%20large%20complex%20data%20sets.&text=Being%20fast%2C%20Python%20jibes%20well,not%20limited%20to%20scientific%20computing.

**Resources for Python Libraries:**

https://matplotlib.org/

https://numpy.org/

<https://pandas.pydata.org/>

https://scikit-learn.org/stable/

https://seaborn.pydata.org/