**Week2 Assignment**

Use the "from the expert" (FTE) jupyter notebook as a starter for this assignment, and ask your instructor questions if you need help.

Use the `churn\_data.csv` file to carry out a similar data cleaning and preparation as we did in the FTE. Specifically, at least complete these minimum requirements:

- Check for outliers in numeric data, and deal with them if needed

- Check for missing values, and decide how to deal with them if needed

- Convert categorical columns to numeric values

- Create at least one new feature by combining multiple columns. For example, you could calculate the ratio of total charges to tenure. Create at least one plot for your new feature.

- Save the data to a csv (or another filetype of your choice) for use next week.

- Write a short analysis at the end of the notebook describing your findings and what you did.

You can do more data cleaning, preparation, and EDA beyond these basic requirements if you want to learn more and develop your data science skills. For example, you could use a box-cox transformation on the numeric data or try other outlier methods.

**Filtering data with pandas**

**Checking for outliers** - deal with them: drop, clip, or treat as missing

**Missing values**

- check for them, deal with them

- drop rows with missing values

- imputation - fill with median, mean, mode, or ML

**Convert categorical to numeric**

- binary encoding

- label encoding

- one-hot encoding

**Feature engineering**

- mathematical transforms (standardize, Yeo-Johnson, log transform, etc)

- combining features

- extracting features from datetimes

Some of the above topics will be covered in the optional advanced section at the bottom of the notebook.

**Other topics we won't cover here, but are part of data cleaning**

- looking for odd values that signify or should be missing values

- e.g. -999, -1, and sometimes 0s are actually missing values

- cleaning inconsistencies in categorical columns

- e.g. Male, male, and MALE could all be in a gender column but should all be mapped to a consistent value

**# Analysis and Summary**

**## Value Counts for Churn**

The value counts for the Churn column are crucial in understanding the distribution of customers based on their churn status.

There are 5174 customers who did not churn (Churn: No) and 1869 customers who churned (Churn: Yes). This information provides a baseline understanding of the dataset's churn distribution, which is essential for subsequent analyses related to customer retention and predictive modeling.

**## Filtering Data**

Data is filtered based on the Churn column, creating two separate DataFrames

```shell

churn\_df: Contains data for customers who churned.

long\_tenure\_df: Contains data for customers with tenure greater than 50.

```

This enables a focused analysis on specific segments of the customer base, such as those who have churned or those with longer tenure.

**## Boxplot for Numeric Columns**

The boxplot visually represents the distribution of numeric columns (tenure, MonthlyCharges, and TotalCharges).

Identifying potential outliers is essential for understanding data quality and potential data issues. The boxplot aids in visualizing the spread of data and potential extreme values.

**## Handling Outliers**

Outliers are clipped for each numeric column using the interquartile range (IQR) method.

Outliers can significantly impact statistical analyses. By handling outliers, the dataset becomes more robust, and subsequent analyses are less influenced by extreme values.

**## Missing Values Check and Handling**

There are 11 missing values in the TotalCharges column.

Missing values are dropped using df.dropna(inplace=True).

**## Categorical to Numeric Conversion**

The Churn and PhoneService columns are converted to numeric. Converting categorical variables to numeric is necessary for many machine learning algorithms. It facilitates quantitative analysis and model training.

**## Feature Engineering**

Log transformation of MonthlyCharges and the creation of two new features:

```shell

"TotalCharges\_Tenure\_Ratio" (Ratio of TotalCharges to Tenure)

"MonthlyCharges\_to\_TotalCharges\_Ratio" (Ratio of MonthlyCharges to TotalCharges)

```

Feature engineering helps create additional relevant features that may improve the performance of predictive models. Log transformation is often used to handle skewed data, and the new ratios provide insights into the relationships between variables.

**## Scatter Plots and Histograms for Feature Relationships**

Visualizations are created to explore relationships, such as TotalCharges\_Tenure\_Ratio and MonthlyCharges\_to\_TotalCharges\_Ratio.

Visualizing relationships helps in understanding patterns, correlations, and potential dependencies between features, aiding further analysis and model development.

**## Advanced Outlier Detection - Boxplot Analysis**

The boxplot for scaled numeric data, as part of advanced outlier detection, provides insights into the distribution of standardized values for different features.

\*\*Tenure\*\*

The boxplot shows that most values are clustered around the median, with a few outliers at both ends. This suggests that while the majority of customers have similar tenure values, there are some extreme cases.

\*\*PhoneService\*\*

Being a binary variable (0 or 1), the boxplot here indicates that the values are concentrated at the extremes (0 and 1), with no apparent outliers.

\*\*MonthlyCharges\*\*

The boxplot for MonthlyCharges shows a relatively symmetric distribution with a few outliers on the higher side. This suggests that most customers have similar monthly charges, but there are some with significantly higher charges.

\*\*TotalCharges\*\*

Similar to MonthlyCharges, TotalCharges exhibit a symmetric distribution with a few outliers on the higher side. This implies that while most customers have similar total charges, there are outliers with exceptionally high total charges.

\*\*Churn\*\*

Being a binary variable (0 or 1), the boxplot for Churn indicates that the values are concentrated at the extremes (0 and 1), with no apparent outliers.

\*\*MonthlyCharges\_to\_TotalCharges\_Ratio\*\*

The boxplot reveals a distribution of the ratio, showing the spread and presence of outliers. Most values are clustered around the median, but there are some customers with significantly different ratios, indicating potential variations in spending patterns.

**## Filling Missing Values with ML (KNNImputer)**

The imputed values are as - Customer 7795-CFOCW: The missing tenure value was imputed as 48.0 – and so on.

**## Merged DataFrames**

The merged DataFrame combines the imputed tenure values with the original DataFrame, resulting in a dataset with non-null tenure values.

**## Value Counts for Contract and PaymentMethod**

The value counts provide information about the distribution of customers based on the Contract and PaymentMethod columns. It shows the count of customers for each unique value.

**## Yeo-Johnson Transform**

The Yeo-Johnson transform is applied to the tenure column. The transformed data is centered around 0 and standardized, making the distribution more symmetric and normal. The density plot shows the before-and-after comparison, highlighting the transformation's impact on the distribution.