Topic:

“Analyzing and Predicting Churn in E-commerce Customer Behavior.”

1. Data collection:

- For this project, I utilized secondary data by acquiring a dataset from kaggle.com.
- The initial Python step involved importing the Pandas library to read the CSV file named 'E-commerce'.
- which comprises 5630 rows and 20 columns.

2. Data preprocessing:

- Initially, the data preprocessing involved importing the Pandas library for reading the CSV file and the NumPy library for numerical calculations.
- After specifying the file path, the code 'dataset.isna().sum()' was used to identify the total number of missing values in each column. Subsequently, variables were segregated based on their qualitative and quantitative nature.
- Descriptive statistics such as mean, median, mode, IQR, max, min, etc., were computed to gain insights into the dataset, facilitating the identification and handling of outliers and missing values.
- In my dataset, some missing values were replaced with either 0 or the median for numerical variables.
- Finally, all variables were consolidated and saved in the same CSV file.

3. Univariate and bivariate:

- Utilizing libraries such as Pandas, NumPy, and Matplotlib, where Matplotlib is employed for visualizing statistical graphs, I imported the dataset and excluded the index column.
- The Seaborn library was then used to create a distribution plot depicting the normal curve with probability density function (pdf) and

cumulative distribution function (cdf) for the 'Warehouse to Home' variable.

- Mean, standard deviation, and the area between specified ranges were computed using pdf and cdf.
- Subsequently, covariance analysis was performed, indicating the directional relationship between variables - a negative value suggests an inverse relationship, while a positive value indicates a positive correlation.
- Further exploration involved calculating correlation coefficients, visually assessed through scatter plots to identify linear relationships. Multicollinearity was addressed using the Variance Inflation Factor.
- The analysis extended to unpaired t-tests, revealing a p-value less than 0.05, leading to the rejection of the null hypothesis and indicating no significant gender-based differences in churn rates.
- Paired t-tests were also conducted, resulting in the rejection of the null hypothesis, signifying no significant distinction between orders placed with and without coupons for phone transactions.
- Additionally, one-way and two-way ANOVA were carried out using the Scipy Stats library.
- The distribution of the 'Hour Spent on App by the Customer' and 'Cashback Amount' columns was visualized using Seaborn's histplot.

4. Feature selection and model save:

- Importing essential libraries such as Pandas, NumPy, and Matplotlib, along with specific modules from the Scikit-learn library (e.g., model_selection, preprocessing, linear_model, ensemble, metrics), I initiated the feature engineering and selection process.
- The code includes the import of classes like 'StandardScaler,' 'GridsearchCV,' 'SelectKBest,' and models such as 'LinearRegression,' 'SVM,' 'DecisionTreeRegressor,' and 'RandomForestRegressor.'
- The feature engineering commenced with the creation of a 'StandardScaler' function and a unified 'train and test split' function.

- Feature selection was implemented using the 'SelectKBest' algorithm. Subsequently, hyperparameter tuning was performed to determine the best-fitted model.
- The evaluation of models was based on the R-squared score, and the results were organized into a dataframe for comparison.
- A 'save model' function was developed to facilitate the storage of the selected model into a file using the 'pickle' module.
- Additionally, the code includes a function to retrieve the selected features by accessing the 'get_support' attribute.

5. In deployment phase:

- In the deployment phase, the saved models were loaded, and new data or user inputs were imported.
- The next step involved utilizing these loaded models to predict the output using the best-performing model.
    -