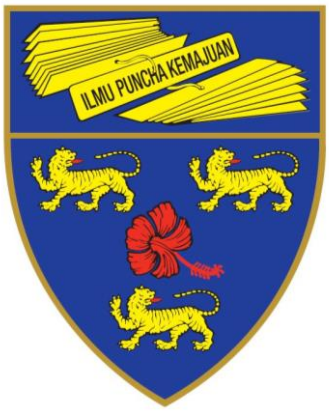


WQD7005 DATA MINING

ALTERNATIVE ASSESSMENT

CASE STUDY: E-COMMERCE CUSTOMER BEHAVIOR ANALYSIS

Student Name	**:**	Sarvinnah Kajandren
Matric Number	**:**	22051663
Module Name	**:**	Data Mining
Module Code	**:**	WQD7005
Lecturer Name	**:**	Dr Teh Ying Wah

1.0 Project Overview

1.1 Case Study description

In the dynamic world of e-commerce, businesses looking to gain a competitive edge now mostly depend on leveraging the convergence of varied information and deriving meaningful insights. This study explores the complexities of consumer behavior analysis by utilizing the combined powers of Talend Data Integration and SAS Enterprise Miner. Comprehensive data interpretation is facilitated by Talend Data Preparation, and data from several sources is streamlined through the integration process. SAS Enterprise Miner's subsequent analysis is designed to help with trend identification, identifying customer behavior influencers, and developing strategic suggestions to improve e-commerce decision-making.

The case study explores in-depth consumer behavior analysis in the e-commerce industry. The research aims to provide actionable insights by utilizing the capabilities of SAS Enterprise Miner, Talend Data Integration, and Talend Data Preparation. Predicting customer churn help companies navigate the digital marketplace are the main goals.

1.2 Objectives

This case study's main goal is to use SAS Enterprise Miner to analyze an integrated e-commerce dataset in-depth, with a particular emphasis on churn prediction and customer behavior analysis. The research attempts to get actionable insights that can guide strategic decision-making for companies involved in the digital marketplace using sophisticated analytics approaches.

1.3 Role of SAS e-miner, Talend Data Integration and Data preparation

Throughout this analysis, there are several tools that were integrated to get an accurate result by the end of this study. This section explains roles of each tool that were used:

1.3.1 Jupyter Notebook

The main tool used in this case study to create a synthetic dataset using the Faker package was Jupyter Notebook. Python scripts were run to take use of Jupiter’s interactive and iterative features and use Faker's capabilities to create a realistic and varied e-commerce dataset. This synthetic dataset was enhanced with fictitious but realistic consumer traits, and it was essential in validating and improving data cleaning algorithms and expanding the source dataset for a more thorough examination of e-commerce customer behavior. The smooth integration of the Faker library was made possible by the Jupyter Notebook environment, which enhanced the dataset production process's efficiency and transparency relative to the larger analytical goals.

1.3.2 Data Talend Integration

The customer information file and the customer behavior file are two crucial datasets that were merged in this case study, and Talend Data Integration was crucial in making this happen. The integration process was carefully planned to

bring information from these many sources together in a smooth manner, making use of Talend's strong capabilities. While the customer behavior file offered insights into past purchases and preferences, the customer details file included basic information like membership credentials and demographics. The mapping and modification of data fields was made easier by Talend's user-friendly graphical interface, which guaranteed a standardized and unified dataset. The case study aimed to provide a complete dataset that encompasses consumer traits and behavior through the implementation of this integration methodology. The logs produced throughout the integration process are an invaluable tool that guarantees the integrity of the combined dataset, reveals information about the merging procedures, and identifies any irregularities. This combined dataset provides a consolidated picture of consumer data for thorough e-commerce behavior analysis, which is crucial for further studies in SAS Enterprise Miner.

1.3.3 Talend Data Preparation

Within the framework of this case study, Talend Data Preparation was important in improving the comprehension of the dataset. A thorough examination of the dataset was carried out using Talend Data Preparation to learn more about its composition, caliber, and possible areas for development. The interactive and visual data exploration made possible by the platform's user-friendly interface allowed for a deeper look at certain qualities, distribution patterns, and the existence of missing information. Talend Data Preparation made it easier to find and deal with outliers, inconsistent data, and missing data, which improved the overall quality of the data. The summary statistics and visualizations that were produced gave the analyst a detailed understanding of the dataset and enabled them to decide on the best preprocessing and data cleaning techniques. The goal of the case study was to guarantee that the future analysis in SAS Enterprise Miner is based on a strong foundation of well-understood and well-prepared data by utilizing Talend Data Preparation for a thorough examination.

1.3.4 SAS Enterprise Miner

SAS Enterprise Miner was essential to the thorough examination of the e-commerce dataset in this case study. One of the most important steps in data preparation was filling in the missing values, which the platform helped with. To properly handle missing data, the case study made use of sophisticated methodologies and SAS's strong capabilities. In addition, a decision tree model was developed using SAS Enterprise Miner, which offered insights into the variables affecting consumer behavior. The investigation was extended to incorporate ensemble methods employing the Random Forest algorithm, including Bagging and Boosting. By using an ensemble

technique, the models' prediction ability was increased, which strengthened our understanding of customer behavior. Ultimately, SAS Enterprise Miner functioned as the analytical engine for analyzing the data and drawing significant business conclusions. The platform's toolkit enabled a thorough examination of the dataset, aiding in the process of making decisions regarding the behavior of online shoppers.

2.0 Dataset Structure

Column name	Description
CustomerID	It serves as a unique identifier for each customer and is not used as a target or input for modeling.
Age	Age is a continuous variable and can be used as an input for modeling.
Gender	Gender is a categorical variable and can be used as an input for modeling.
Country	Country is a categorical variable and can be used as an input for modelling
Membership level	Membership level is a categorical variable and can be used as an input for modelling
TotalPurchases	Total purchases is a continuous variable and can be used as an input for modeling.
TotalSpent	Total spent is a continuous variable and can be used as an input for modelling
FavoriteCategory	Favorite category is a categorical variable and can be used as an input for modelling
LastPurchaseDate	It can be used as an input for modeling if transformed appropriately
Occupation	Occupation is a categorical variable and can be used as an input for modelling
WebsiteVisitsFrequency	Website visits frequency is a categorical variable and can be used as an input for modelling
Item	Item is a categorical variable and can be used as an input for modelling
Churn	Churn is the target variable indicating customer churn (1 for churned, 0 for active)

3.0 Code and scripts

3.1 Data generation with Jupyter Notebook and Faker library

There are total of two csv files generated for this case study. The first file was about customer details such as customer ID, age, gender, and location. The second file was about customer behavior where it includes details such as membership level, item, churn etc.

(i) Faker library is imported to generate synthetic data

```
[ ] !pip install faker

    Collecting faker
      Downloading Faker-22.0.0-py3-none-any.whl (1.7 MB)
         ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.7/1.7 MB 7.6 MB/s eta 0:00:00
    Requirement already satisfied: python-dateutil>=2.4 in /usr/local/lib/python3.10/dist-packages (from faker) (2.8.2)
    Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.4->faker) (1.16.0)
    Installing collected packages: faker
    Successfully installed faker-22.0.0
```

(ii) Customer details dataset

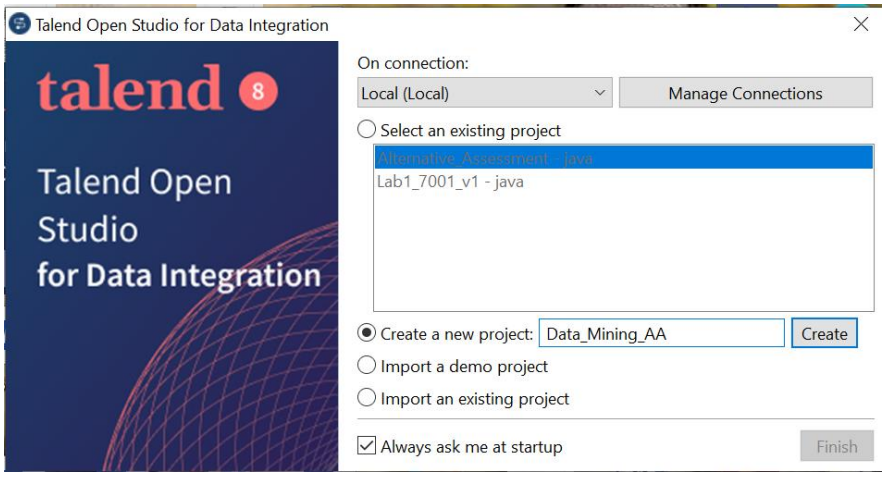
```
import csv
import random

# Set the seed for reproducibility
random.seed(42)

# Generate a random Asian_Country data with at least 1000 rows
data = []

for customer_id in range(1, 1001):
    age = random.randint(18, 65)
    gender = random.choice(['Male', 'Female', 'Non-Binary'])
    country = random.choice(['China', 'India', 'Indonesia', 'Pakistan', 'Bangladesh', 'Japan', 'Philippines', 'Vietnam', 'Turkey', 'Iran'])

    # Introduce missing values intentionally
    if random.random() < 0.1:  # 10% chance of missing values
        asian_country = None

    data.append([customer_id, asian_country])

# Write the dataset to a CSV file
with open('asian_countries.csv', 'w', newline='') as csvfile:
    csvwriter = csv.writer(csvfile)
    # Write header
    csvwriter.writerow(['customer_id', 'asian_countries'])
    # Write data
    csvwriter.writerows(data)
```

(iii) Customer behavior dataset

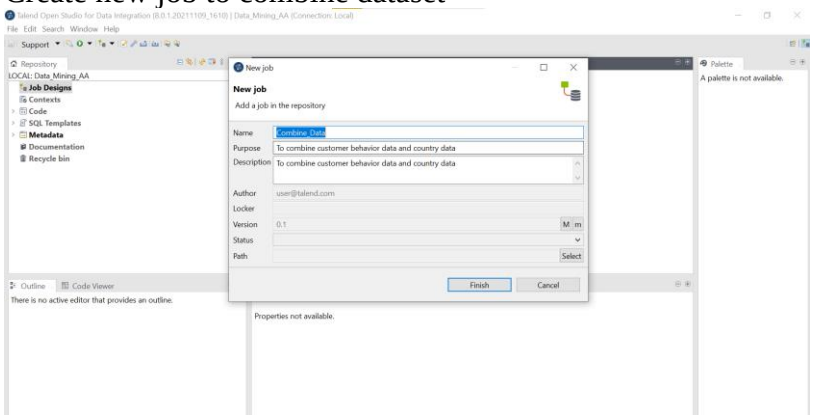
```
import csv
from faker import Faker
import random
from datetime import datetime, timedelta

fake = Faker()

# Set the seed for reproducibility
random.seed(42)

# Generate a synthetic dataset with at least 1000 rows
data = []

for _ in range(1000):
    customer_id = _ + 1
    membership_level = random.choice(['Bronze', 'Silver', 'Gold', 'Platinum'])
    total_purchases = random.randint(1, 50)
    total_spent = random.uniform(50, 2000)
    favorite_category = random.choice(['Electronics', 'Clothing', 'Home Goods'])
    last_purchase_date = (datetime.now() - timedelta(days=random.randint(1, 365))).strftime('%Y-%m-%d')
    occupation = fake.job()
    website_visits_frequency = random.choice(['Daily', 'Weekly', 'Bi-weekly', 'Monthly'])

    # Introduce missing values intentionally
    if random.random() < 0.1:  # 10% chance of missing values
        age = None
        location = None
        membership_level = None

    # Determine item related to the favorite category
    if favorite_category == 'Electronics':
        item = random.choice(['Laptop', 'Smartphone', 'Headphones', 'Smartwatch'])
    elif favorite_category == 'Clothing':
        item = random.choice(['T-Shirt', 'Jeans', 'Sneakers', 'Dress'])
    elif favorite_category == 'Home Goods':
        item = random.choice(['Couch', 'Coffee Maker', 'Bedding Set', 'Lamp'])
    else:
        item = 'Unknown'

    churn = random.choice([0, 1])

    data.append([customer_id, age, gender, location, membership_level, total_purchases, total_spent,
                 favorite_category, last_purchase_date, occupation, website_visits_frequency, item, churn])

# Write the dataset to a CSV file
with open('ecommerce_customer_behavior_dataset.csv', 'w', newline='') as csvfile:
    csvwriter = csv.writer(csvfile)
    # Write header
    csvwriter.writerow(['CustomerID', 'Age', 'Gender', 'Location', 'MembershipLevel', 'TotalPurchases',
                        'TotalSpent', 'FavoriteCategory', 'LastPurchaseDate', 'Occupation',
                        'WebsiteVisitsFrequency', 'Item', 'Churn'])
    # Write data
    csvwriter.writerows(data)
```

3.2 Integration Workflow in Talend Data Integration

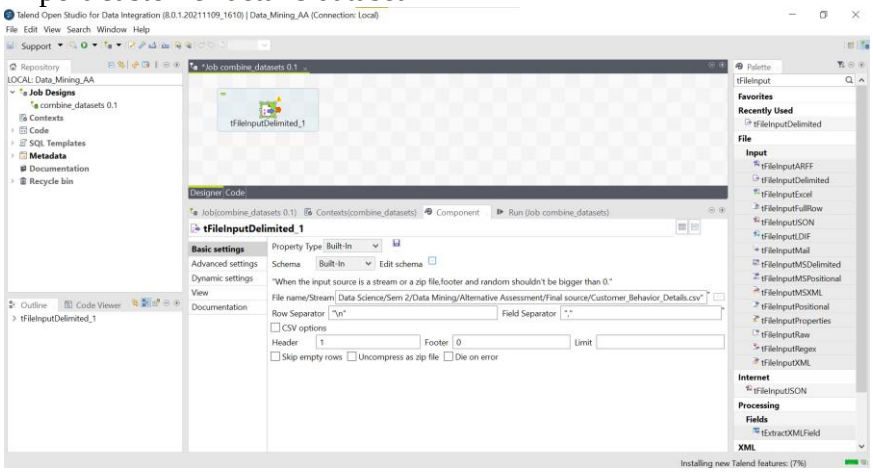
(i) Create new project



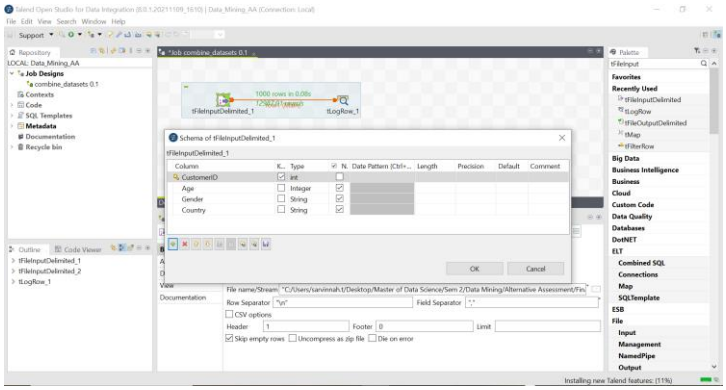
(ii) Create new job to combine dataset



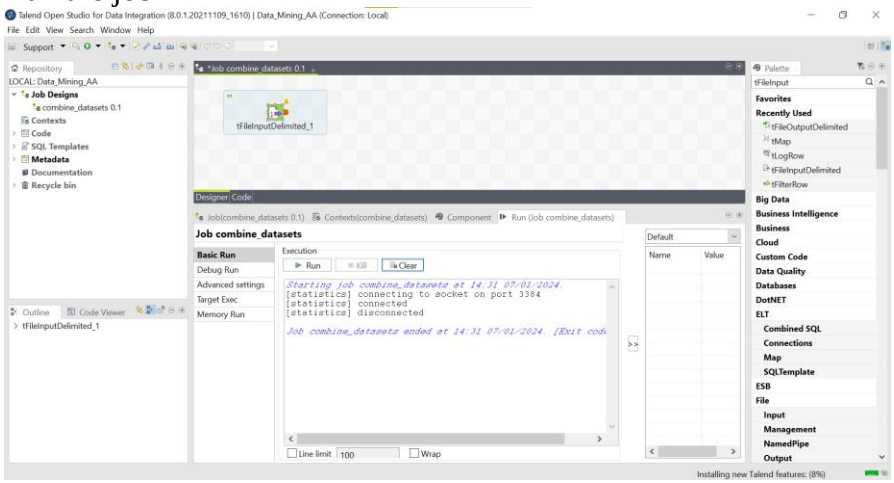
(iii) Import customer details dataset



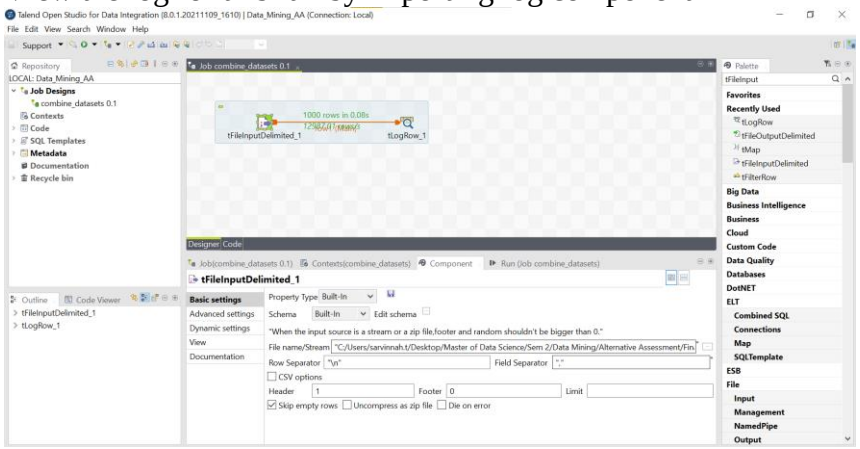
(iv) Create schema for this dataset



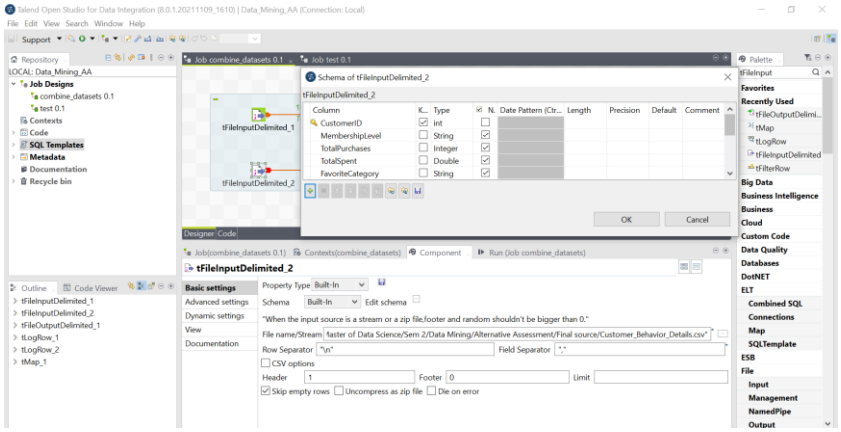
(v) Run the job



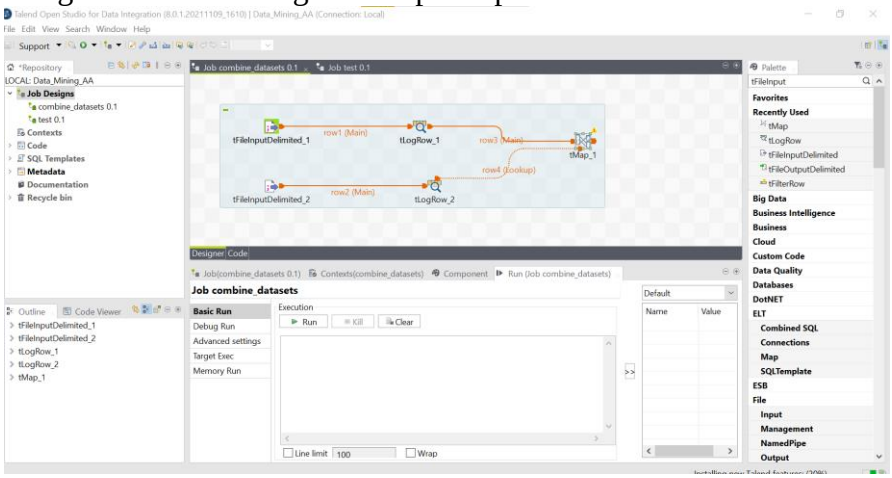
(vi) View the log for this run by importing log component



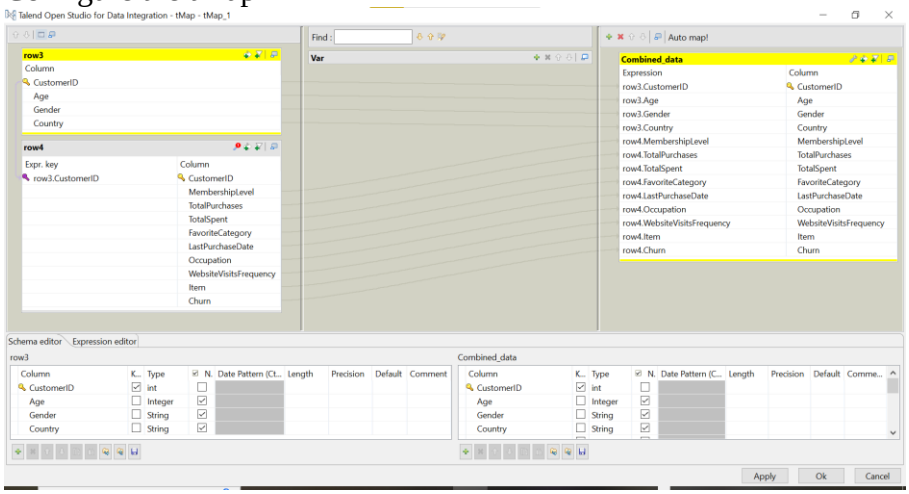
(vii) Repeat the same steps for Customer behavior dataset



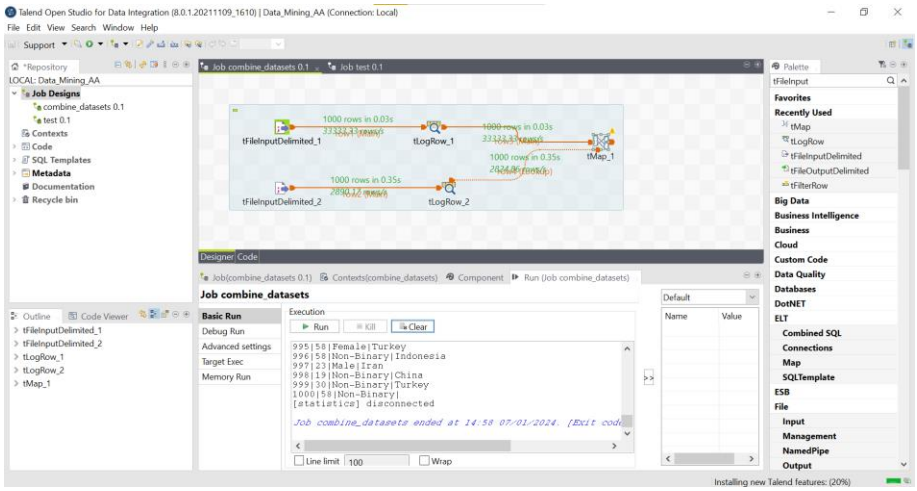
(viii) Merge the dataset using the tMap component



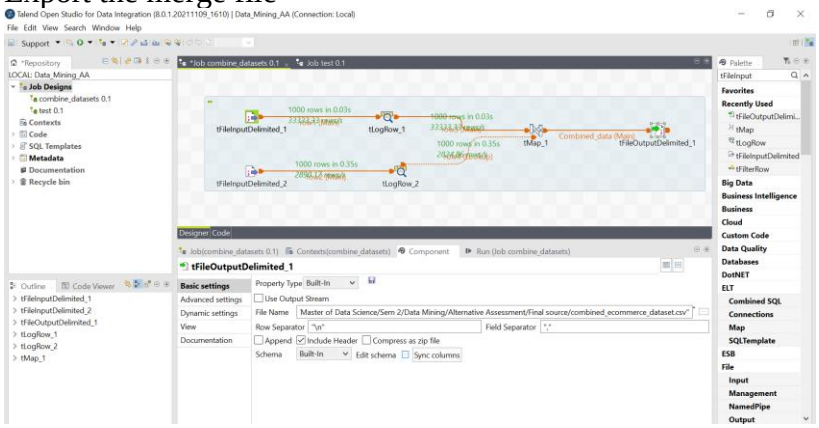
(ix) Configure the tMap



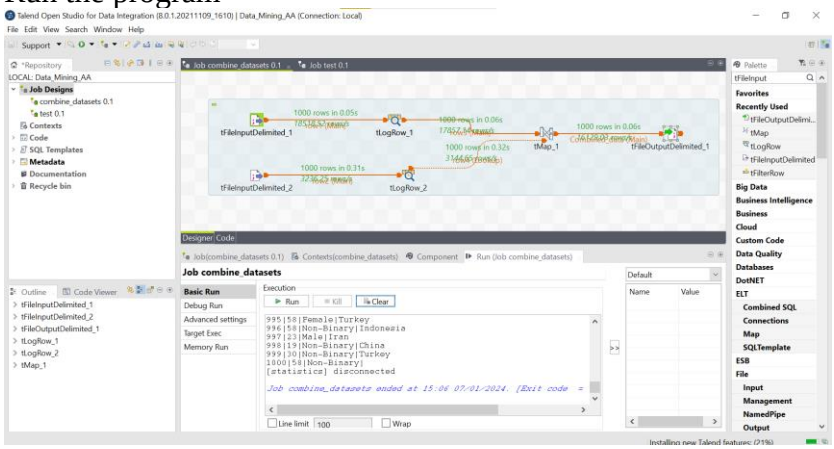
(x) Run the job



(xi) Export the merge file



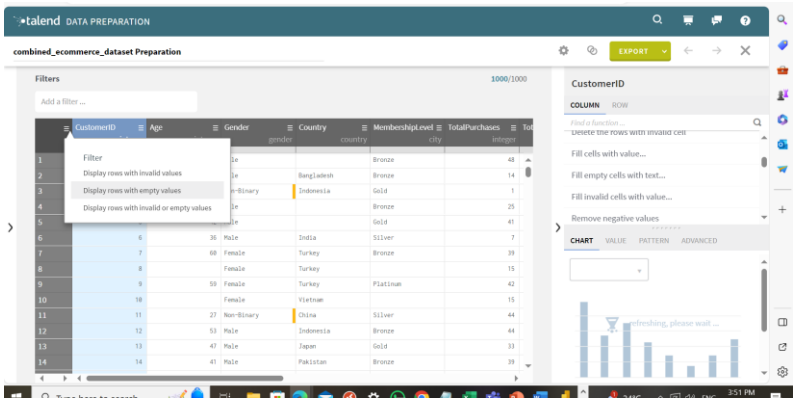
(xii) Run the program



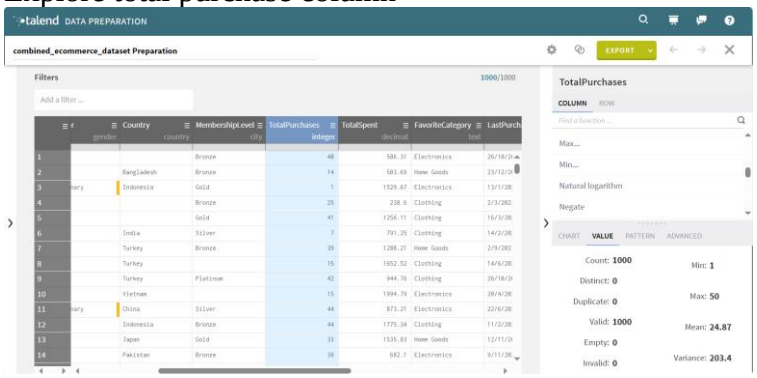
3.3 Data exploration with Talend data preparation

This tool is used to Explore the combined file that is exported from Talend Data Integration. This tool helped in understanding each column better which helps during analysis phase in SAS. Below are few samples of exploration:

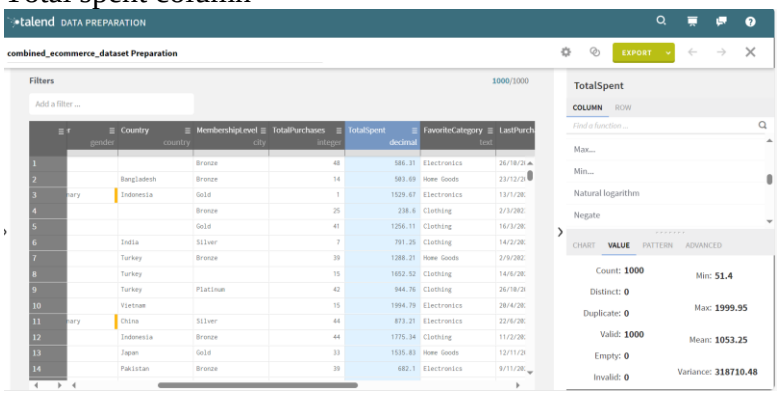
(i) Explore data with empty values



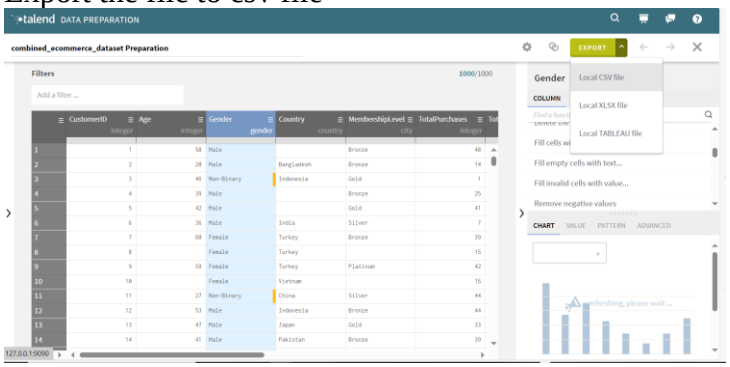
(ii) Explore total purchase column

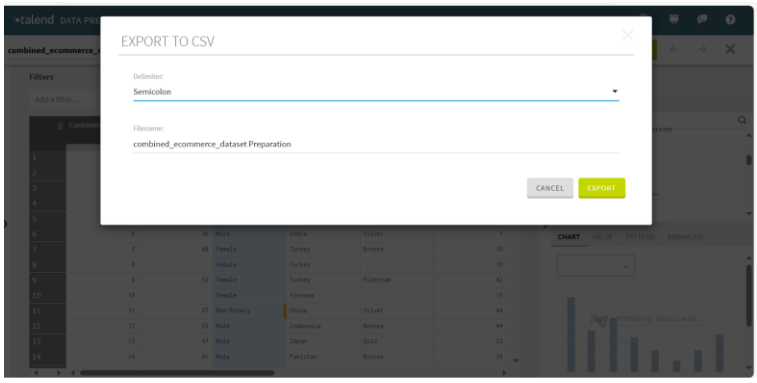


(iii) Total spent column



(iv) Export the file to csv file



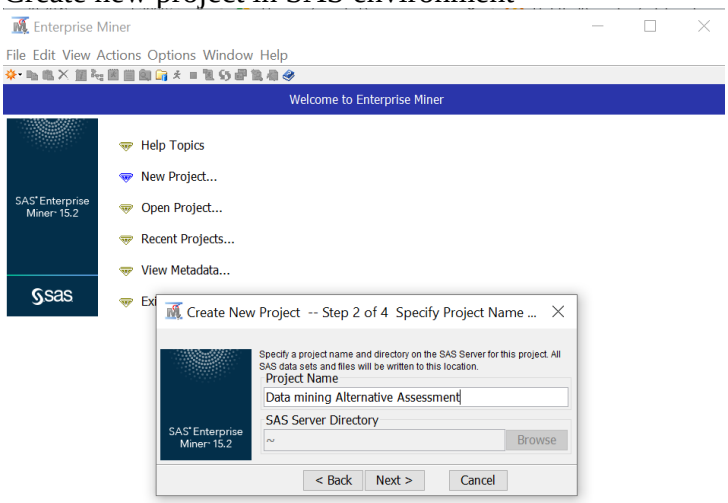


3.4 Data mining models in SAS e-miner

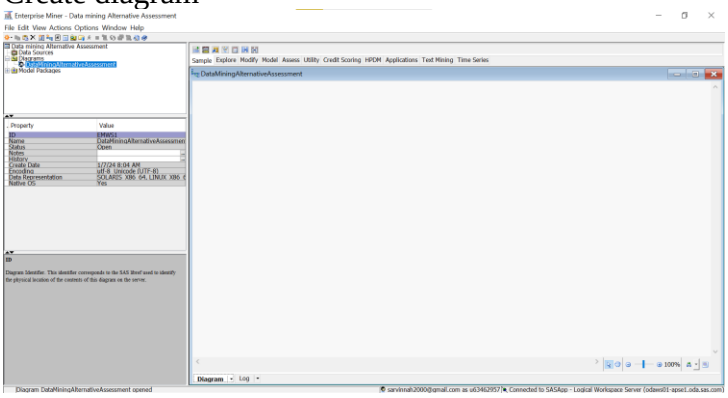
By applying sophisticated data mining algorithms to the e-commerce dataset, SAS Enterprise Miner significantly contributed to the study. After addressing missing variables, a Decision Tree model is developed to visualize consumer behavior. To increase prediction accuracy, I also investigated ensemble methods like bagging and boosting, which use the Random Forest algorithm. I found trends thanks to these models, particularly in terms of anticipating customer churn. The procedure ran smoothly because of SAS Enterprise Miner's user-friendly interface, which also offered insightful information for making strategic decisions in the e-commerce industry.

3.4.1 Prepare the environment for analysis of the ecommerce dataset

(i) Create new project in SAS environment

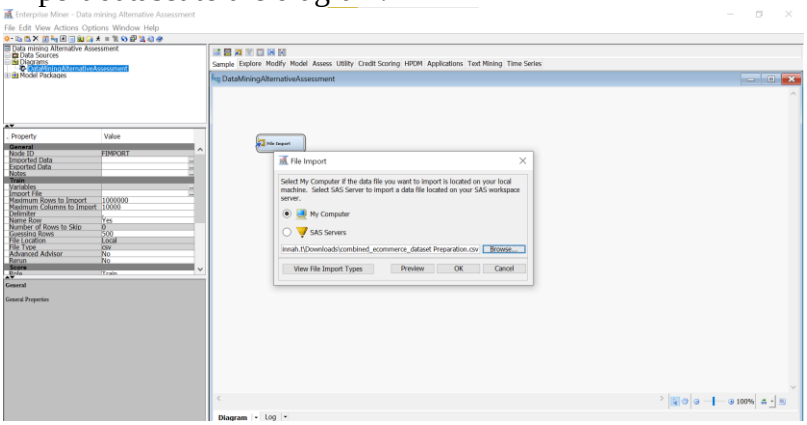


(ii) Create diagram



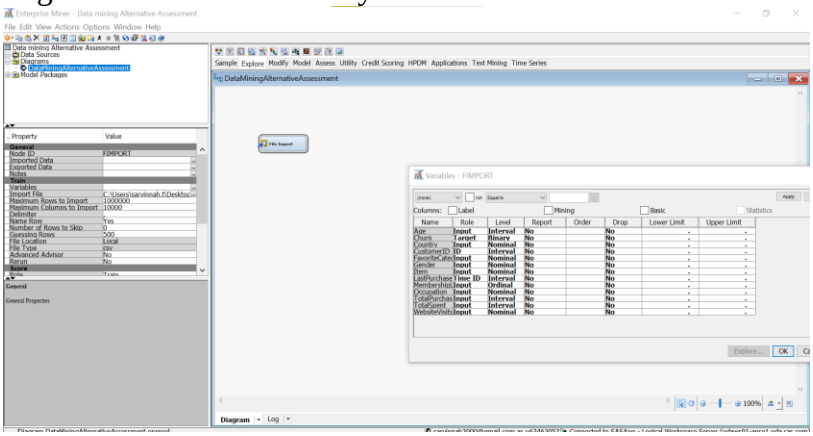
3.4.2 Data Import and Preprocessing: Import your dataset into SAS Enterprise Miner, handle missing values, and specify variable roles

(i) Import dataset to the diagram.



(ii) Assigning role to each variable

For this case study, none of the variables were removed as each variable was required to do the analysis. Churn column is given target role for this case study to understand customer behavior.



(iii) Understand the dataset using StatExplore node.

(a) Variable Summary

The Role column here explains the role of each variable in the dataset. Next, the measurement level column specifies the measurement level of each variable and finally the count column shows the frequency of each variable's role and level

Variable Summary

Role	Measurement Level	Frequency Count
ID	INTERVAL	1
INPUT	INTERVAL	3
INPUT	NOMINAL	6
INPUT	ORDINAL	1
TARGET	BINARY	1

(b) Variable level summary

Shows the number of distinct levels for every variable and their corresponding roles.

Variable Levels Summary
(maximum 500 observations printed)

Variable	Role	Frequency Count
Churn	TARGET	2
CustomerID	ID	1000

(c) Class Variable summary statistics

Provide statistics, such as mode, mode percentage, and mode2 (the second most common category), for categorical variables found in the training data.

Class Variable Summary Statistics
(maximum 500 observations printed)

Data Role=TRAIN

Data Role	Variable Name	Role	Number of Levels	Missing	Mode	Mode Percentage	Mode2	Mode2 Percentage
TRAIN	Country	INPUT	11	94	India	10.20	Japan	9.80
TRAIN	FavoriteCategory	INPUT	3	0	Clothing	34.40	Electronics	33.80
TRAIN	Gender	INPUT	3	0	Non-Binary	36.00	Female	32.00
TRAIN	Item	INPUT	12	0	Smartphone	9.30	Jeans	9.10
TRAIN	MembershipLevel	INPUT	5	98	Bronze	23.90	Silver	23.00
TRAIN	Occupation	INPUT	502	0	Archivist	0.60	Commissioning editor	0.60
TRAIN	WebsiteVisitsFrequency	INPUT	4	0	Daily	25.60	Weekly	25.30
TRAIN	Churn	TARGET	2	0	TRUE	50.50	FALSE	49.50

(d) Distribution of Class Target and Segment Variables

Displays the distribution of the training data's target variable, or churn, with the count and percentage for each class.

Distribution of Class Target and Segment Variables
(maximum 500 observations printed)

Data Role=TRAIN

Data Role	Variable Name	Role	Level	Frequency Count	Percent
TRAIN	Churn	TARGET	TRUE	505	50.5
TRAIN	Churn	TARGET	FALSE	495	49.5

(e) Interval Variable Summary Statistics

Provides summary statistics such as mean, standard deviation, minimum, maximum, skewness, and kurtosis for continuous variables (e.g., Age, TotalPurchases, TotalSpent) as part of the training data.

Interval Variable Summary Statistics
(maximum 500 observations printed)

Data Role=TRAIN

Variable	Role	Mean	Standard Deviation	Non Missing	Missing	Minimum	Median	Maximum	Skewness	Kurtosis
Age	INPUT	41.66741	13.89313	902	98	18	41	65	0.022519	-1.19492
TotalPurchases	INPUT	24.868	14.26177	1000	0	1	24	50	0.066162	-1.17217
TotalSpent	INPUT	1053.248	564.5445	1000	0	51.4	1071.34	1999.95	-0.02772	-1.21065

(f) Class Variable Summary Statistics by Class Target
Displays categorical variable data, such as mode, mode %, mode2, and mode2 percentage for each class, based on the target variable (Churn).

Class Variable Summary Statistics by Class Target
(maximum 500 observations printed)

Data Role=TRAIN Variable Name=Country

Target	Target Level	Number of Levels	Missing	Mode	Mode Percentage	Mode2	Mode2 Percentage
Churn	FALSE	11	39	India	10.71	Indonesia	10.10
Churn	TRUE	11	55		10.89	Japan	10.89
OVERALL		11	94	India	10.20	Japan	9.80

Data Role=TRAIN Variable Name=FavoriteCategory

Target	Target Level	Number of Levels	Missing	Mode	Mode Percentage	Mode2	Mode2 Percentage
Churn	FALSE	3	0	Clothing	34.34	Home Goods	33.54
Churn	TRUE	3	0	Electronics	35.45	Clothing	34.46
OVERALL		3	0	Clothing	34.40	Electronics	33.80

Data Role=TRAIN Variable Name=Gender

Target	Target Level	Number of Levels	Missing	Mode	Mode Percentage	Mode2	Mode2 Percentage
Churn	FALSE	3	0	Non-Binary	37.58	Female	31.92
Churn	TRUE	3	0	Non-Binary	34.46	Male	33.47
OVERALL		3	0	Non-Binary	36.00	Female	32.00

Data Role=TRAIN Variable Name=Item

Target	Target Level	Number of Levels	Missing	Mode	Mode Percentage	Mode2	Mode2 Percentage
Churn	FALSE	12	0	Couch	9.70	Smartphone	9.29
Churn	TRUE	12	0	Jeans	9.31	Smartphone	9.31
OVERALL		12	0	Smartphone	9.30	Jeans	9.10

Data Role=TRAIN Variable Name=MembershipLevel

Target	Target Level	Number of Levels	Missing	Mode	Mode Percentage	Mode2	Mode2 Percentage
Churn	FALSE	5	50	Bronze	24.24	Silver	23.03
Churn	TRUE	5	48	Bronze	23.56	Silver	22.97
OVERALL		5	98	Bronze	23.90	Silver	23.00

Data Role=TRAIN Variable Name=Occupation

Target	Target Level	Number of Levels	Missing	Mode	Mode Percentage	Mode2	Mode2 Percentage
Churn	FALSE	347	0	Medical secretary	1.01	Financial manager	0.81
Churn	TRUE	339	0	Commissioning editor	1.19	Artist	0.99
OVERALL		502	0	Archivist	0.60	Commissioning editor	0.60

Data Role=TRAIN Variable Name=WebsiteVisitsFrequency

Target	Target Level	Number of Levels	Missing	Mode	Mode Percentage	Mode2	Mode2 Percentage
Churn	FALSE	4	0	Daily	26.87	Monthly	25.25
Churn	TRUE	4	0	Weekly	27.33	Daily	24.36
OVERALL		4	0	Daily	25.60	Weekly	25.30

(g) Interval Variable Summary Statistics by Class Target

Gives summary statistics for continuous variables, such as the median, mean, standard deviation, skewness, and kurtosis for each class, depending on the target variable (Churn).

Interval Variable Summary Statistics by Class Target
(maximum 500 observations printed)

Data Role=TRAIN Variable=Age

Target	Target Level	Median	Missing	Non Missing	Minimum	Maximum	Mean	Standard Deviation	Skewness	Kurtosis	Role	Label
Churn	FALSE	41	50	445	18	65	41.72809	13.65568	0.063824	-1.14472	INPUT	Age
Churn	TRUE	42	48	457	18	65	41.60832	14.13522	-0.01284	-1.24269	INPUT	Age
OVERALL		41	98	902	18	65	41.66741	13.89313	0.022519	-1.19482	INPUT	Age

Data Role=TRAIN Variable=TotalPurchases

Target	Target Level	Median	Missing	Non Missing	Minimum	Maximum	Mean	Standard Deviation	Skewness	Kurtosis	Role	Label
Churn	FALSE	24	0	495	1	50	24.85657	14.52799	0.09176	-1.19478	INPUT	TotalPurchases
Churn	TRUE	25	0	505	1	50	24.87921	14.01032	0.038511	-1.15121	INPUT	TotalPurchases
OVERALL		24	0	1000	1	50	24.868	14.26177	0.066162	-1.17217	INPUT	TotalPurchases

Data Role=TRAIN Variable=TotalSpent

Target	Target Level	Median	Missing	Non Missing	Minimum	Maximum	Mean	Standard Deviation	Skewness	Kurtosis	Role	Label
Churn	FALSE	1100.6	0	495	58.46	1999.95	1072.446	572.9001	-0.08128	-1.23905	INPUT	TotalSpent
Churn	TRUE	1034.35	0	505	51.4	1996.39	1034.43	556.157	0.023264	-1.1733	INPUT	TotalSpent
OVERALL		1071.34	0	1000	51.4	1999.95	1053.248	564.5445	-0.02772	-1.21065	INPUT	TotalSpent

(h) Chi-Square Statistics

Shows the chi-square statistics for the correlation between the target variable (Churn) and the category variables. The degrees of freedom (Df), probability (Prob), and chi-square value are among the results of the chi-square test, which evaluates the independence of variables.

Chi-Square Statistics
(maximum 500 observations printed)

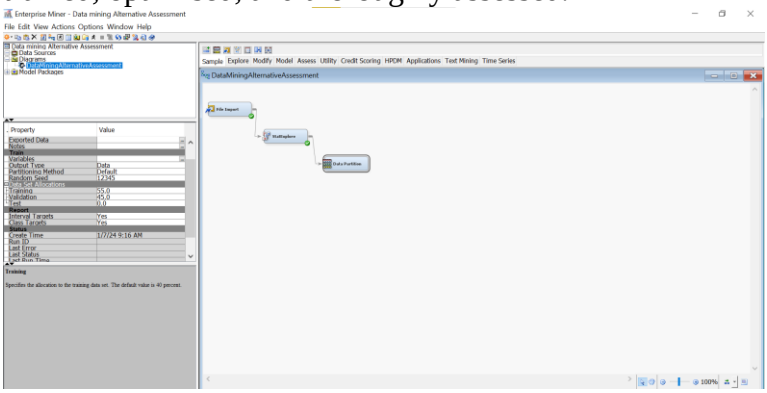
Data Role=TRAIN Target=Churn

Input	Chi-Square	Df	Prob
Occupation	520.0853	501	0.2688
Country	7.7898	10	0.6494
Item	5.7240	11	0.8911
Age	5.1543	5	0.3973
TotalSpent	3.0191	4	0.5546
WebsiteVisitsFrequency	2.4182	3	0.4903
TotalPurchases	1.9037	4	0.7535
FavoriteCategory	1.7465	2	0.4176
Gender	1.3626	2	0.5059
MembershipLevel	0.3542	4	0.9861

(iv) Data Partition

The dataset is systematically divided into three subsets during the "Data Partition" phase: Training Data (55.0%), Validation Data (45.0%), and Test Data (0.0%). Machine learning models are trained using 55% of the Training Data, and their performance is independently optimised and fine-tuned on 45% of the Validation Data. To guarantee strong generalisation, it is usually essential to evaluate the final model on completely fresh and unknown data, even though the Test Data is presently set at 0%. Given this, carefully choosing which test set to

include might improve the model's dependability in practical applications. By using this data partitioning technique, the model's performance in each project is guaranteed to be well-trained, optimised, and thoroughly assessed.



a) Partition summary
   - EMWS1.Stat_TRAIN: This dataset includes all 1000 observations in their entirety. It functions as the initial dataset for analysis and model construction.
   - EMWS1.Part_TRAIN (TRAIN): With 550 observations (55% of the total), this training dataset is a subset of the original dataset. Its purpose is to train models for machine learning.
   - Part_VALIDATE (EMWS1.Part_VALIDATE): With 450 observations (45% of the total), this validation dataset is an additional subset of the original dataset. It is used separately to evaluate and optimise the models' functionality.

Partition Summary

Type	Data Set	Number of Observations
DATA	EMWS1.Stat_TRAIN	1000
TRAIN	EMWS1.Part_TRAIN	550
VALIDATE	EMWS1.Part_VALIDATE	450

b) Summary statistics for class targets
   - Data=DATA: The class distribution in the original dataset is displayed in this section. It reveals that 50.5% of observations for Churn are labelled as "TRUE" and 49.5% as "FALSE."
   - Data=TRAIN: 278 observations are labelled as "TRUE" (50.55%) and 272 observations are labelled as "FALSE" (49.45%) for Churn in the training dataset.
   - Data=VALIDATE: For Churn, there are 227 observations labelled as "TRUE" (50.44%) and 223 observations labelled as "FALSE" (49.56%) in the validation dataset.

Summary Statistics for Class Targets

Data=DATA

Variable	Numeric Value	Formatted Value	Frequency Count	Percent	Label
Churn	.	FALSE	495	49.5	
Churn	.	TRUE	505	50.5	

Data=TRAIN

Variable	Numeric Value	Formatted Value	Frequency Count	Percent	Label
Churn	.	FALSE	272	49.4545	
Churn	.	TRUE	278	50.5455	

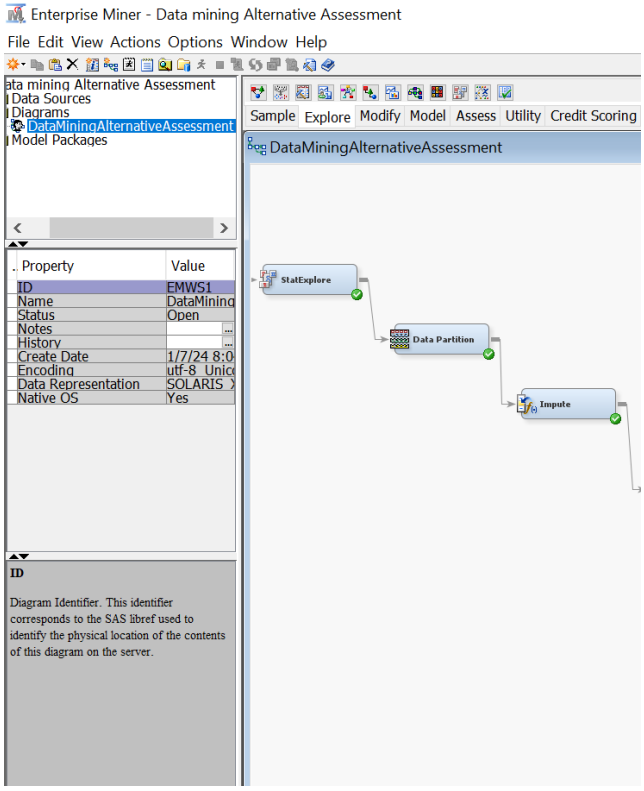
Data=VALIDATE

Variable	Numeric Value	Formatted Value	Frequency Count	Percent	Label
Churn	.	FALSE	223	49.5556	
Churn	.	TRUE	227	50.4444	

(v) Handle missing values

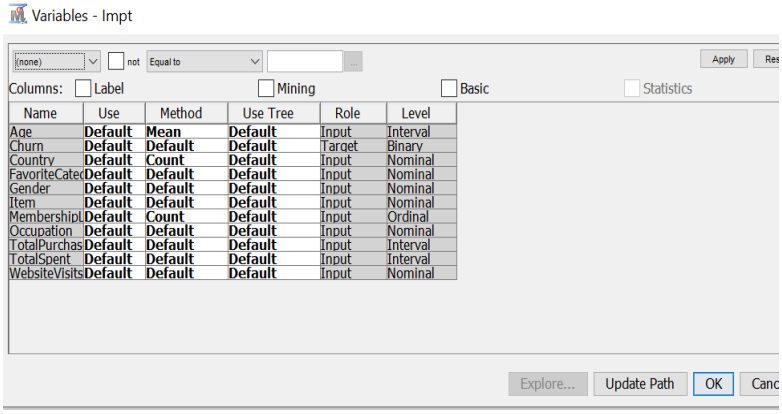
In the stateExplorer, there are few columns that have missing values. These missing values were handled by using impute node in which missing value for numerical variables were replaced by using mean and categorical variables were replaced by using mode value. Below are the further steps in handling missing values in variables.

(a) Impute node is implemented



(b) Edit variables to impute:

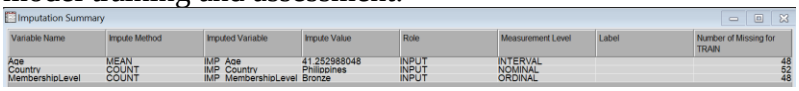
Since the empty value is in Age, Country and Membership column only, the method for this column are altered in which for numerical column the empty value replaced with mean and for non-numerical columns I have used count method.



Name	Use	Method	Use Tree	Role	Level
Age	Default	Mean	Default	Input	Interval
Churn	Default	Default	Default	Target	Binary
Country	Default	Count	Default	Input	Nominal
FavoriteCateg	Default	Default	Default	Input	Nominal
Gender	Default	Default	Default	Input	Nominal
Item	Default	Default	Default	Input	Nominal
MembershipL	Default	Count	Default	Input	Ordinal
Occupation	Default	Default	Default	Input	Nominal
TotalPurchas	Default	Default	Default	Input	Interval
TotalSpent	Default	Default	Default	Input	Interval
WebsiteVisits	Default	Default	Default	Input	Nominal

(c) Output of impute method

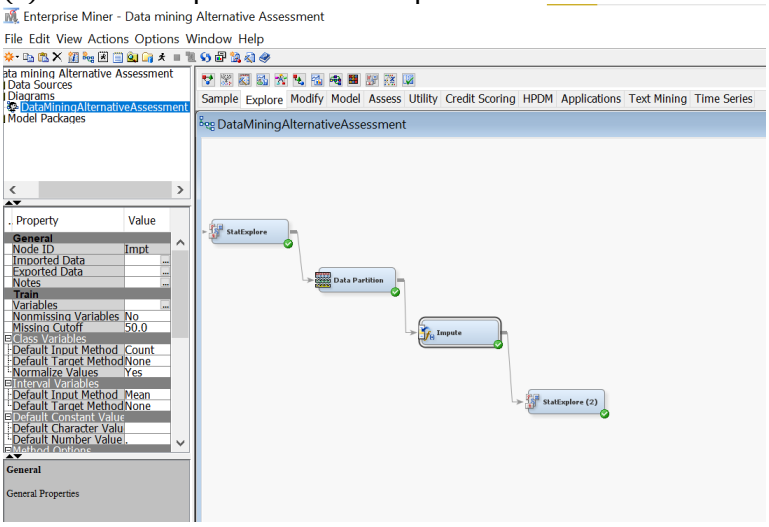
The number of missing values for each variable in the training data is displayed in the table. For example, the "Country" variable has missing values in 52 observations, and the "Age" and "MembershipLevel" variables have missing values in 48 observations. To handle missing data and make sure the dataset is full and prepared for further analysis, imputation is a necessary procedure. The selected imputation techniques, such mean and count imputation, help to preserve the data's usefulness and integrity for model training and assessment.



Variable Name	Impute Method	Imputed Variable	Impute Value	Role	Measurement Level	Label	Number of Missing for TRAIN
Age	MEAN	IMP_Age	41.252988048	INPUT	INTERVAL		48
Country	COUNT	IMP_Country	Philippines	INPUT	NOMINAL		52
MembershipLevel	COUNT	IMP_MembershipLevel	Bronze	INPUT	ORDINAL		48

(vi) Verify missing values after imputation method using statExplorer

(a) Join statExplorer node to impute node



(b) Output

Stat shows that there are no missing values in the dataset. The dataset is now clean to proceed with the modelling part.

Class Variable Summary Statistics
(maximum 500 observations printed)

Data Role=TRAIN

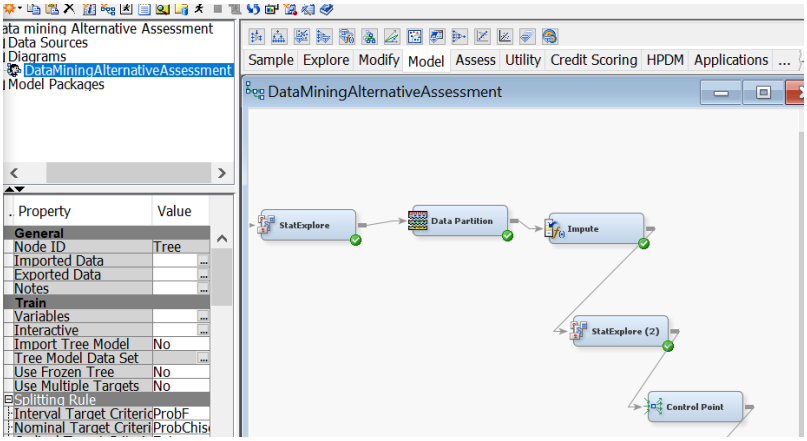
Data Role	Variable Name	Role	Number of Levels	Missing	Mode	Mode Percentage	Mode2	Mode2 Percentage
TRAIN	FavoriteCategory	INPUT	3	0	Clothing	36.55	Electronics	33.45
TRAIN	Gender	INPUT	3	0	Non-Binary	37.45	Male	31.64
TRAIN	IMP_Country	INPUT	10	0	Philippines	19.64	China	9.82
TRAIN	IMP_MembershipLevel	INPUT	4	0	Bronze	33.64	Silver	24.00
TRAIN	Item	INPUT	12	0	Jeans	11.09	Dress	9.82
TRAIN	Occupation	INPUT	374	0	Artist	0.91	Archivist	0.73
TRAIN	WebsiteVisitsFrequency	INPUT	4	0	Weekly	26.55	Daily	26.18
TRAIN	Churn	TARGET	2	0	TRUE	50.55	FALSE	49.45

3.4.3 Decision Tree Analysis: Create a decision tree model in SAS Enterprise Miner to analyze customer behavior.

Decision Tree Analysis is used in the E-Commerce Customer Behavior Analysis case study to carefully analyze the information and identify trends that affect customer attrition. The decision tree model offers a framework that is easy to understand and visualize by classifying clients according to factors such as age, gender, membership level, and buying behavior. This study makes it possible to estimate the likelihood of customer attrition, pinpoints crucial characteristics that influence consumer behavior (such as spending and favorite category) and provides clear insights for the development of successful company strategies. Businesses may reduce churn and increase overall profitability by customizing marketing campaigns, optimizing promotions, and improving customer experiences by understanding the decision criteria that contribute to churn.

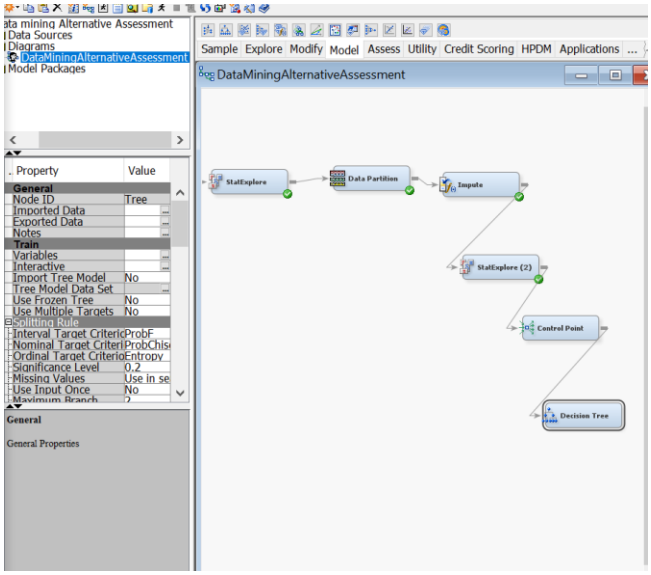
(a) Control point

The Control Point node is crucial in directing the flow of data and conducting analyses depending on predetermined criteria in the model flow developed for the SAS Enterprise Miner case study on E-Commerce Customer Behavior Analysis. This node is put in a strategic manner to control the conditional execution of tasks, such as directing the data flow to various process segments according to the results of previous models. For example, based on the decision tree model's anticipated chance of client attrition, it may steer the analysis in a different direction. Through the utilization of the Control Point node, the model flow becomes more flexible and responsive, enabling customized interventions, error management, and dynamic modifications in reaction to the changing circumstances and discoveries made during the procedure.



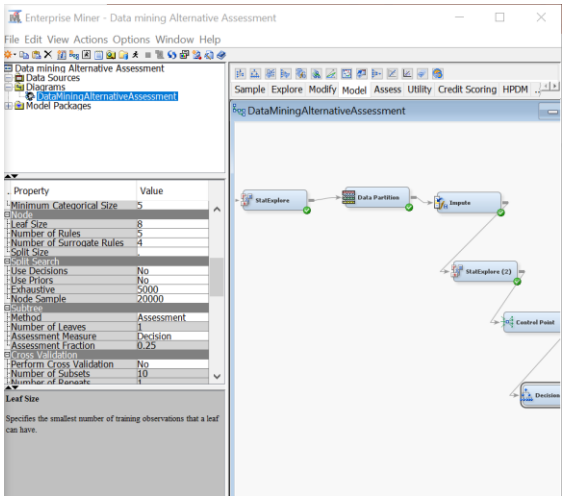
(b) Decision tree analysis

i. The decision tree node is connected to the central point node



ii. Decision tree configuration.

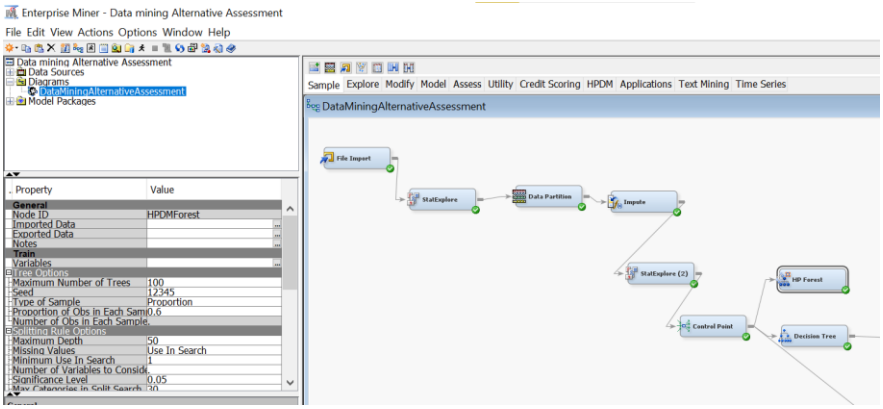
The maximum depth property for the decision tree is set to 10 to allow the tree to be trained with up to 10 generations of splits. Next the leaf size to 8 which means each leaf will require minimum of 8 training observations. Finally, set the number of surrogate rules to 4. This is to ensure that if the splitting relies on an input with missing values, then the software is configured to us up to four surrogate rules to determine splits in non-leaf nodes. The assessment measure property is set to Decision in which the decision tree will seek to maximize profit in the validation data. The output of the model is explained in the results and analysis section.



3.4.4 Ensemble Methods: Apply Bagging and Boosting, using the Random Forest algorithm as a Bagging example.

i. Random Forest (Bagging)

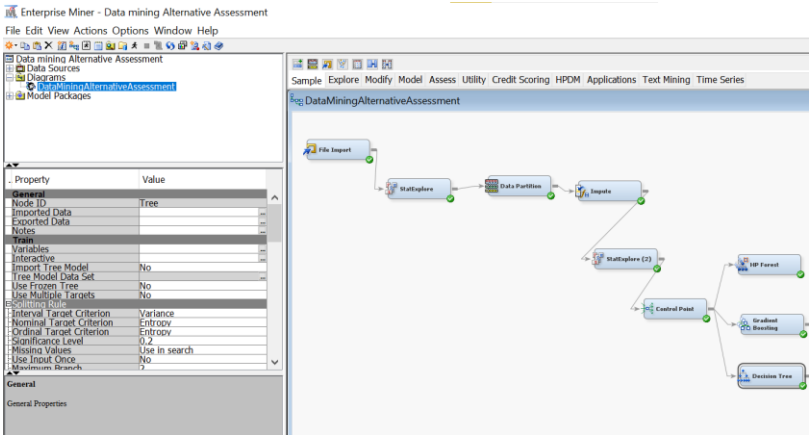
In this case study, the customer churn forecasting model's accuracy and resilience are improved by Random Forest using bagging technique. Using several subsets of the training data, this ensemble learning approach creates different decision trees while incorporating unpredictability into the voting and feature selection processes. Randomness encourages generalization to previously unknown data and reduces overfitting. The Random Forest model for analyzing customer turnover offers a more consistent and dependable approach by merging forecasts from several trees. The model is a useful tool for companies looking to comprehend and successfully handle customer turnover because of its evaluation metrics, which include assessment score distributions and misclassification rates. These metrics provide insights into the model's performance on both training and validation datasets.



ii. Gradient Boosting (Boosting)

Gradient Boosting is used as a potent ensemble learning strategy in the case study to improve the customer churn forecasting model's accuracy. With this approach, a sequence of decision trees is constructed one after the other, with each tree concentrating on fixing mistakes from the preceding one.

Gradient Boosting adjusts to the complexity of the data by iteratively fine-tuning the model depending on its performance, enabling it to identify complicated patterns associated with customer attrition. An evaluation of the model's performance on training and validation datasets may be gained by examining its evaluation metrics, which include misclassification rates and assessment score distributions. Businesses looking for a more complex method of customer churn research will find that Gradient Boosting is a useful tool as it improves predicting skills by utilizing the strengths of several weak learners.



4.0 Results and Analysis

4.1 Model Outputs

(I) Decision Tree

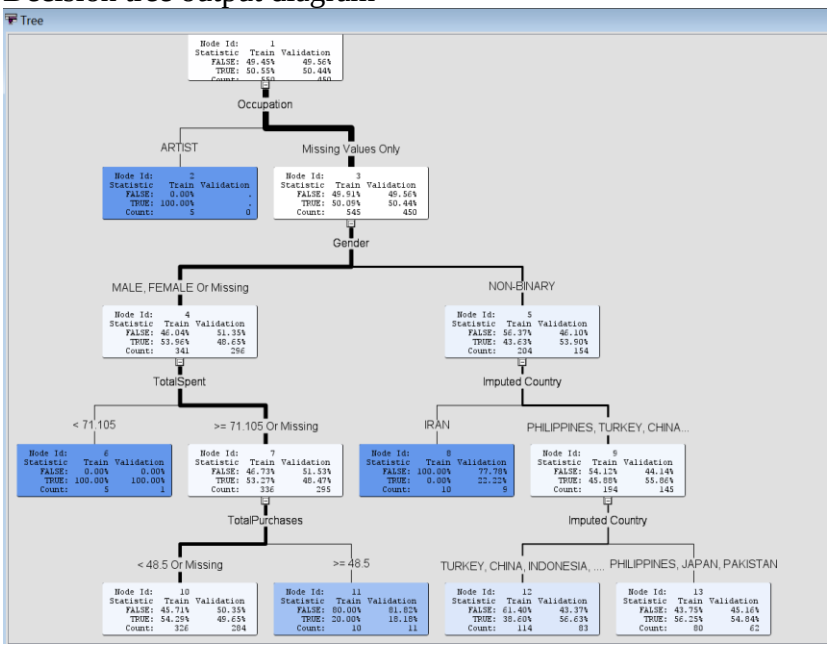
The Decision Tree Analysis's result offers a thorough grasp of the variables affecting customer churn in the case study. The dataset is used to create the decision tree structure, which shows the important factors that go into determining a customer's likelihood of churning or not. Every variable is given weight in the analysis, emphasizing how each affects the model's ability to predict outcomes accurately. The factors imputed age, amount spent, and imported country are found to be very important in predicting customer attrition.

The tree structure emerges as a sequence of splits, each of which represents a condition that directs the customer classification into different segments. The attributes of clients that fit into various categories are shown by the leaf nodes of the tree. The model's performance is quantified by fit statistics, classification tables, and evaluation score rankings, which highlight the model's accuracy in churn prediction. Furthermore, each feature's significance is emphasized in the variable importance section, where the most significant predictors are the imputed age, imputed nation, and total spent.

Businesses may use this Decision Tree Analysis as a valuable tool to understand the complex patterns of customer churn and, as a result, make well-informed decisions on client retention tactics. Strategic planning is

made easier by the quantification of model performance and the clear visualization of decision rules, which empowers companies to better satisfy customers and reduce the loss of customers.

(a) Decision tree output diagram



(b) Model Event

The target variable 'Churn' is binary, with two levels ('FALSE' and 'TRUE'), and the analysis is carried out in decreasing order, according to the Model Events section.

Model Events

Target	Event	Measurement Level	Number of Levels	Order	Label
Churn	TRUE	BINARY	2	Descending	

(c) Variable importance

The input variables are ranked in the Variable Importance section according to how much of a contribution they make to the model. The input variables with greater importance ratios are 'IMP_Country' (Imputed Country), 'TotalSpent,' and 'IMP_Age' (Imputed Age). Each leaf node's training observations are detailed in the Tree Leaf Report, along with the proportion of actual churn cases.

Variable Importance

Variable Name	Label	Number of Splitting Rules	Number of Surrogate Rules	Importance	Validation Importance	Ratio of Validation to Training Importance
IMP_Country	Imputed Country	2	0	1.0000	0.7894	0.7894
TotalSpent		1	2	0.9030	0.6147	0.6807
IMP_Age	Imputed Age	0	2	0.8994	0.7718	0.8581
TotalPurchases		1	1	0.7657	1.0000	1.3061
Gender		1	0	0.6269	0.0000	0.0000
Occupation		1	0	0.5967	0.0000	0.0000
WebsiteVisitsFrequency		0	1	0.5005	0.0000	0.0000

(d) Fit statistics

Metrics like misclassification rate, maximum absolute error, and sum of squared errors are provided for both training and validation data in Fit Statistics, which assess the model's performance. For every dataset role, the Classification Table offers a thorough analysis of true positives, true negatives, false positives, and false negatives.

Fit Statistics

Target=Churn Target Label=' '

Fit Statistics	Statistics Label	Train	Validation
NOBS	Sum of Frequencies	550.00	450.000
MISC	Misclassification Rate	0.42	0.493
MAX	Maximum Absolute Error	0.80	1.000
SSE	Sum of Squared Errors	258.41	227.401
ASE	Average Squared Error	0.23	0.253
RASE	Root Average Squared Error	0.48	0.503
DIV	Divisor for ASE	1100.00	900.000
DFT	Total Degrees of Freedom	550.00	.

(e) Assessment Score Rankings

The model's predictive ability at different decision levels is indicated by the Assessment Score Rankings, which display cumulative gain, lift, and other metrics at different tree depths. Based on the given posterior probabilities for the training and validation datasets, the Posterior Probability Ranges and Distribution provide information on the probability of customer attrition.

Assessment Score Rankings

Data Role=TRAIN Target Variable=Churn Target Label=' '

Depth	Gain	Lift	Cumulative Lift	% Response	Cumulative % Response	Number of Observations	Mean Posterior Probability
5	42.1987	1.42199	1.42199	71.8750	71.8750	28	0.71875
10	27.0234	1.11286	1.27023	56.2500	64.2045	27	0.56250
15	21.7144	1.11286	1.21714	56.2500	61.5211	28	0.56250
20	18.4513	1.08420	1.18451	54.8015	59.8717	27	0.54801
25	16.2124	1.07417	1.16212	54.2945	58.7401	28	0.54294
30	14.7732	1.07417	1.14773	54.2945	58.0126	27	0.54294
35	13.7060	1.07417	1.13706	54.2945	57.4732	28	0.54294
40	12.9342	1.07417	1.12934	54.2945	57.0831	27	0.54294
45	12.3113	1.07417	1.12311	54.2945	56.7683	28	0.54294
50	11.8308	1.07417	1.11831	54.2945	56.5254	27	0.54294
55	11.4229	1.07417	1.11423	54.2945	56.3192	28	0.54294
60	11.0952	1.07417	1.11095	54.2945	56.1536	27	0.54294
65	10.8075	1.07417	1.10808	54.2945	56.0082	28	0.54294
70	10.5697	1.07417	1.10570	54.2945	55.8880	27	0.54294
75	10.3560	1.07417	1.10356	54.2945	55.7799	28	0.54294
80	8.4816	0.79811	1.08482	40.3407	54.8325	27	0.40341
85	6.5598	0.76360	1.06560	38.5965	53.8611	28	0.38596
90	4.9126	0.76360	1.04913	38.5965	53.0285	27	0.38596
95	3.3839	0.76360	1.03384	38.5965	52.2559	28	0.38596
100	0.0000	0.34452	1.00000	17.4139	50.5455	27	0.17414

Data Role=VALIDATE Target Variable=Churn Target Label=' '

Depth	Gain	Lift	Cumulative Lift	% Response	Cumulative % Response	Number of Observations	Mean Posterior Probability
5	12.6036	1.12604	1.12604	56.8022	56.8022	23	0.58152
10	10.7006	1.08711	1.10701	54.8387	55.8423	22	0.56250
15	9.2710	1.06474	1.09271	53.7103	55.1212	23	0.55825
20	6.6188	0.98421	1.06619	49.6479	53.7833	22	0.54294
25	4.9502	0.98421	1.04950	49.6479	52.9415	23	0.54294
30	3.8862	0.98421	1.03886	49.6479	52.4048	22	0.54294
35	3.0906	0.98421	1.03091	49.6479	52.0035	23	0.54294
40	2.5199	0.98421	1.02520	49.6479	51.7156	22	0.54294
45	2.0554	0.98421	1.02055	49.6479	51.4813	23	0.54294
50	1.7001	0.98421	1.01700	49.6479	51.3020	22	0.54294
55	1.3960	0.98421	1.01396	49.6479	51.1486	23	0.54294
60	1.1535	0.98421	1.01154	49.6479	51.0263	22	0.54294
65	0.9390	0.98421	1.00939	49.6479	50.9181	23	0.54294
70	0.7632	0.98421	1.00763	49.6479	50.8294	22	0.54294
75	0.6038	0.98421	1.00604	49.6479	50.7490	23	0.54294
80	0.9700	1.06596	1.00970	53.7716	50.9337	22	0.45018
85	1.6477	1.12255	1.01648	56.6265	51.2756	23	0.38596
90	2.2239	1.12255	1.02224	56.6265	51.5663	22	0.38596
95	2.7629	1.12255	1.02763	56.6265	51.8382	23	0.38596
100	0.0000	0.46248	1.00000	23.3297	50.4444	22	0.13509

(II) Random Forest

Out of a possible 100 trees, a total of 13 trees were used to train the Random Forest model. To forecast customer attrition, the model combines nominal, ordinal, and interval input data. A node size of 100, a leaf size of 1, and a Gini split criterion are important factors. The more trees added to the model, the better its performance is measured by log loss, average square error, and misclassification rate; 13 trees yielded the best results.

(a) Variable importance

"TotalPurchases," "IMP_Age," and "IMP_MembershipLevel" are important predictors for detecting likely churners, according to the variable importance study. Each variable's Gini value indicates how much it contributes to lowering impurity in the decision trees.

Loss Reduction Variable Importance

Variable	Number of Rules	Gini	OOB Gini	Valid Gini	Margin	OOB Margin	Valid Margin
TotalPurchases	1	0.000513	0.00021	-0.00097	0.001027	0.00062	-0.00038
IMP_Age	0	0.000000	0.00000	0.00000	0.000000	0.00000	0.00000
TotalSpent	0	0.000000	0.00000	0.00000	0.000000	0.00000	0.00000
WebsiteVisitsFrequency	0	0.000000	0.00000	0.00000	0.000000	0.00000	0.00000
Item	0	0.000000	0.00000	0.00000	0.000000	0.00000	0.00000
Occupation	0	0.000000	0.00000	0.00000	0.000000	0.00000	0.00000
IMP_MembershipLevel	1	0.000644	-0.00098	-0.00033	0.001288	-0.00050	0.00028
FavoriteCategory	1	0.000823	-0.00192	-0.00073	0.001646	-0.00103	0.00023
IMP_Country	1	0.001444	-0.00215	-0.00032	0.002888	-0.00064	0.00094
Gender	12	0.011062	-0.01010	-0.01743	0.022124	0.00087	-0.00646

(b) Assessment score ranking

Response rates, cumulative gain, and lift at various tree levels offer information on how well the model can distinguish between churn and non-churn cases. At a depth of 5, the model performs at its best and captures the most important patterns in the data.

Assessment Score Rankings

Data Role=TRAIN Target Variable=Churn Target Label=' '

Depth	Gain	Lift	Cumulative Lift	% Response	Cumulative % Response	Number of Observations	Mean Posterior Probability
5	17.5150	1.17515	1.17515	59.3985	59.3985	28	0.57561
10	17.5150	1.17515	1.17515	59.3985	59.3985	27	0.57561
15	17.5150	1.17515	1.17515	59.3985	59.3985	28	0.57561
20	17.5150	1.17515	1.17515	59.3985	59.3985	27	0.57561
25	16.4042	1.12040	1.16404	56.6313	58.8370	28	0.57339
30	11.5693	0.86857	1.11569	43.9024	56.3932	27	0.56322
35	9.9067	1.00109	1.09907	50.6007	55.5528	28	0.56190
40	9.4747	1.06387	1.09475	53.7736	55.3345	27	0.56127
45	9.1260	1.06387	1.09126	53.7736	55.1582	28	0.56127
50	8.8571	1.06387	1.08857	53.7736	55.0223	27	0.56127
55	8.2967	1.02793	1.08297	51.9573	54.7391	28	0.55109
60	7.7825	1.02012	1.07783	51.5625	54.4792	27	0.54888
65	8.6610	1.19014	1.08661	60.1563	54.9232	28	0.49836
70	8.4177	1.05191	1.08418	53.1695	54.8002	27	0.44613
75	8.1423	1.04356	1.08142	52.7473	54.6610	28	0.44562
80	8.3053	1.10798	1.08305	56.0036	54.7434	27	0.42929
85	6.0339	0.70341	1.06034	35.5540	53.5953	28	0.42272
90	3.5024	0.59623	1.03502	30.1366	52.3158	27	0.41652
95	2.6521	0.87620	1.02652	44.2881	51.8860	28	0.40996
100	0.0000	0.48628	1.00000	24.5791	50.5455	27	0.39509

Data Role=VALIDATE Target Variable=Churn Target Label=' '

Depth	Gain	Lift	Cumulative Lift	% Response	Cumulative % Response	Number of Observations	Mean Posterior Probability
5	0.88106	0.99119	0.99119	50.0000	50.0000	23	0.57561
10	0.88106	0.99119	0.99119	50.0000	50.0000	22	0.57561
15	0.88106	0.99119	0.99119	50.0000	50.0000	23	0.57561
20	0.88106	0.99119	0.99119	50.0000	50.0000	22	0.57561
25	1.42928	0.96425	0.98571	48.6413	49.7235	23	0.56753
30	2.01297	0.94989	0.97987	47.9167	49.4290	22	0.56322
35	2.21667	0.96588	0.97783	48.7231	49.3263	23	0.56220
40	2.18370	0.98053	0.97816	49.4624	49.3429	22	0.56127
45	2.15686	0.98053	0.97843	49.4624	49.3564	23	0.56127
50	2.13633	0.98053	0.97864	49.4624	49.3668	22	0.56127
55	2.39561	0.95068	0.97604	47.9565	49.2360	23	0.55642
60	2.98065	0.90424	0.97019	45.6140	48.9409	22	0.54888
65	3.49835	0.90424	0.96502	45.6140	48.6797	23	0.54888
70	1.90896	1.19259	0.98091	60.1595	49.4815	22	0.46044
75	1.63048	1.02183	0.98370	51.5459	49.6220	23	0.44271
80	0.53280	1.33769	1.00533	67.4788	50.7132	22	0.43147
85	0.60877	1.01798	1.00609	51.3514	50.7515	23	0.42423
90	0.35662	0.82837	0.99643	41.7866	50.2646	22	0.42163
95	0.59157	0.95271	0.99408	48.0590	50.1460	23	0.41175
100	0.00000	1.11509	1.00000	56.2500	50.4444	22	0.39356

(c) Assessment score distribution

The posterior probability distribution shows how confident the model

is in its ability to forecast churn within probability ranges. Interestingly, most forecasts are within the probability range of 0.55-0.60.

Assessment Score Distribution

Data Role=TRAIN Target Variable=Churn Target Label=' '

Posterior Probability Range	Number of Events	Number of Nonevents	Mean Posterior Probability	Percentage
0.55-0.60	154	126	0.56837	50.9091
0.50-0.55	33	31	0.54888	11.6364
0.40-0.45	85	95	0.42973	32.7273
0.35-0.40	6	20	0.39464	4.7273

Data Role=VALIDATE Target Variable=Churn Target Label=' '

Posterior Probability Range	Number of Events	Number of Nonevents	Mean Posterior Probability	Percentage
0.55-0.60	118	121	0.56754	53.1111
0.50-0.55	26	31	0.54888	12.6667
0.40-0.45	70	61	0.42949	29.1111
0.35-0.40	13	10	0.39373	5.1111

(d) Classification table

The model's performance is summarised in the classification table, which shows the number and percentage of occurrences for both the training and validation datasets that were categorised correctly and erroneously. It draws attention to how well the model can differentiate between false positives, false negatives, real positives, and true negatives.

Classification Table

Data Role=TRAIN Target Variable=Churn Target Label=' '

Target	Outcome	Target Percentage	Outcome Percentage	Frequency Count	Total Percentage
FALSE	FALSE	55.8252	42.2794	115	20.9091
TRUE	FALSE	44.1748	32.7338	91	16.5455
FALSE	TRUE	45.6395	57.7206	157	28.5455
TRUE	TRUE	54.3605	67.2662	187	34.0000

Data Role=VALIDATE Target Variable=Churn Target Label=' '

Target	Outcome	Target Percentage	Outcome Percentage	Frequency Count	Total Percentage
FALSE	FALSE	46.1039	31.8386	71	15.7778
TRUE	FALSE	53.8961	36.5639	83	18.4444
FALSE	TRUE	51.3514	68.1614	152	33.7778
TRUE	TRUE	48.6486	63.4361	144	32.0000

Event Classification Table

Data Role=TRAIN Target=Churn Target Label=' '

False Negative	True Negative	False Positive	True Positive
91	115	157	187

Data Role=VALIDATE Target=Churn Target Label=' '

False Negative	True Negative	False Positive	True Positive
83	71	152	144

This Random Forest model shows promise as a predictive tool, especially when it comes to finding important characteristics linked to customer churn. The thorough assessment of variable significance and model performance offers insightful information for strategic decision-making.

(III) Gradient Boosting

(a) Variable importance

With an important ratio of 1.00000, the variable significance analysis reveals "Item" to be the most significant predictor. "IMP_Country" (0.82464), "TotalPurchases" (0.66321), "TotalSpent" (0.64842), "IMP_Age" (0.59199), "Gender" (0.36132), and "WebsiteVisitsFrequency" (0.21674) are the next most significant predictors. The relative contributions of each variable to the predicted performance of the model are indicated by these values.

Variable Importance

Obs	NAME	LABEL	NRULES	IMPORTANCE	VIMPORTANCE	RATIO
1	Item		38	1.00000	1.00000	1.00000
2	IMP_Country	Imputed Country	27	0.82464	0.69283	0.84016
3	TotalPurchases		19	0.66321	0.63448	0.95668
4	TotalSpent		22	0.64842	0.39086	0.60279
5	IMP_Age	Imputed Age	15	0.59199	0.64181	1.08417
6	Gender		5	0.36132	0.00000	0.00000
7	WebsiteVisitsFrequency		2	0.21674	0.00000	0.00000
8	IMP_MembershipLevel	Imputed MembershipLevel	2	0.20409	0.00000	0.00000

(b) Model fit statistics

A high degree of accuracy is indicated by the low misclassification rate, which is 0.28 for the training dataset and 0.453 for the validation dataset. A quantifiable indicator of prediction accuracy is provided by the average squared error (ASE), which is 0.21 for the training set and 0.259 for the validation set, respectively. The model's overall fit is shown by the root average squared error (RASE), which is 0.45 for training and 0.509 for validation.

Fit Statistics

Target=Churn Target Label=' '

Fit Statistics	Statistics Label	Train	Validation
NOBS	Sum of Frequencies	550.00	450.000
SUMW	Sum of Case Weights Times Freq	1100.00	900.000
MISC	Misclassification Rate	0.28	0.453
MAX	Maximum Absolute Error	0.72	0.763
SSE	Sum of Squared Errors	226.72	232.728
ASE	Average Squared Error	0.21	0.259
RASE	Root Average Squared Error	0.45	0.509
DIV	Divisor for ASE	1100.00	900.000
DFT	Total Degrees of Freedom	550.00	.

(c) Classification table

The model accurately detects 71.79% of non-churn instances and 72.30% of churn cases in the training dataset. The accurate identification rates in the validation dataset are 59.03% for churn and 54.63% for non-churn. These numbers provide a thorough analysis of the predicted accuracy of the model.

Classification Table

Data Role=TRAIN Target Variable=Churn Target Label=' '

Target	Outcome	Target Percentage	Outcome Percentage	Frequency Count	Total Percentage
FALSE	FALSE	71.4815	70.9559	193	35.0909
TRUE	FALSE	28.5185	27.6978	77	14.0000
FALSE	TRUE	28.2143	29.0441	79	14.3636
TRUE	TRUE	71.7857	72.3022	201	36.5455

Data Role=VALIDATE Target Variable=Churn Target Label=' '

Target	Outcome	Target Percentage	Outcome Percentage	Frequency Count	Total Percentage
FALSE	FALSE	54.6341	50.2242	112	24.8889
TRUE	FALSE	45.3659	40.9692	93	20.6667
FALSE	TRUE	45.3061	49.7758	111	24.6667
TRUE	TRUE	54.6939	59.0308	134	29.7778

(d) Event Classiication table

In the training dataset, the model accurately identifies 193 true negative cases and 201 true positive instances while examining churn events. It accurately identifies 112 true negative events and 134 genuine positive instances in the validation dataset.

Event Classification Table

Data Role=TRAIN Target=Churn Target Label=' '

False Negative	True Negative	False Positive	True Positive
77	193	79	201

Data Role=VALIDATE Target=Churn Target Label=' '

False Negative	True Negative	False Positive	True Positive
93	112	111	134

(e) Assessment score ranking

Across various tree depths, cumulative gain, lift, and response metrics are displayed. The cumulative gain is 97.84% and the lift is 1.97842, for example, for a tree depth of 5, reflecting the relative improvement in prediction accuracy over a random model.

Assessment Score Rankings

Data Role=TRAIN Target Variable=Churn Target Label=' '

Depth	Gain	Lift	Cumulative Lift	% Response	Cumulative % Response	Number of Observations	Mean Posterior Probability
5	97.8417	1.97842	1.97842	100.000	100.000	28	0.71734
10	76.2590	1.53877	1.76259	77.778	89.091	27	0.66410
15	66.8545	1.48381	1.66854	75.000	84.337	28	0.63701
20	65.4676	1.61204	1.65468	81.481	83.636	27	0.61131
25	59.1336	1.34250	1.59134	67.857	80.435	28	0.58693
30	57.0743	1.46549	1.57074	74.074	79.394	27	0.56929
35	54.7881	1.41316	1.54788	71.429	78.238	28	0.55683
40	51.9784	1.31894	1.51978	66.667	76.818	27	0.54046
45	49.1790	1.27184	1.49179	64.286	75.403	28	0.52475
50	43.8849	0.95257	1.43885	48.148	72.727	27	0.50979
55	38.4239	0.84789	1.38424	42.857	69.967	28	0.49568
60	36.6906	1.17240	1.36691	59.259	69.091	27	0.48400
65	33.1840	0.91855	1.33184	46.429	67.318	28	0.47006
70	28.9825	0.73275	1.28983	37.037	65.195	27	0.45510
75	23.5912	0.49460	1.23591	25.000	62.470	28	0.44422
80	19.1547	0.51292	1.19155	25.926	60.227	27	0.42805
85	14.9849	0.49460	1.14985	25.000	58.120	28	0.40760
90	9.9121	0.21982	1.09912	11.111	55.556	27	0.38600
95	4.7842	0.14132	1.04784	7.143	52.964	28	0.34480
100	0.0000	0.07327	1.00000	3.704	50.545	27	0.27420

Data Role=VALIDATE Target Variable=Churn Target Label=' '

Depth	Gain	Lift	Cumulative Lift	% Response	Cumulative % Response	Number of Observations	Mean Posterior Probability
5	22.4287	0.77571	0.77571	39.1304	39.1304	23	0.70763
10	20.7048	0.81097	0.79295	40.9091	40.0000	22	0.65587
15	15.4574	0.94809	0.84543	47.8261	42.6471	23	0.62457
20	3.0837	1.35162	0.96916	68.1818	48.8889	22	0.60370
25	1.7582	1.03428	0.98242	52.1739	49.5575	23	0.58639
30	4.5521	0.81097	0.95448	40.9091	48.1481	22	0.57180
35	0.8811	1.20667	0.99119	60.8696	50.0000	23	0.55975
40	2.4229	1.26151	1.02423	63.6364	51.6667	22	0.54488
45	0.5838	0.86190	1.00584	43.4783	50.7389	23	0.53328
50	3.0837	1.26151	1.03084	63.6364	52.0000	22	0.52160
55	7.1124	1.46524	1.07112	73.9130	54.0323	23	0.50765
60	3.5242	0.63076	1.03524	31.8182	52.2222	22	0.49117
65	0.8104	0.68952	1.00810	34.7826	50.8532	23	0.47734
70	1.9509	1.17141	1.01951	59.0909	51.4286	22	0.46056
75	0.2919	0.77571	1.00292	39.1304	50.5917	23	0.44360
80	1.8722	1.26151	1.01872	63.6364	51.3889	22	0.42778
85	1.9657	1.03428	1.01966	52.1739	51.4360	23	0.40937
90	1.8111	0.99119	1.01811	50.0000	51.3580	22	0.38853
95	0.9716	0.86190	1.00972	43.4783	50.9346	23	0.35316
100	0.0000	0.81097	1.00000	40.9091	50.4444	22	0.29117

(f) Assessment score distribution

There are ranges in the posterior probability distribution for churn forecasts. For example, in the training dataset, the mean probability in the range of 0.57225 corresponds to 15.45% of cases with a posterior probability between 0.55 and 0.60.

Assessment Score Distribution

Data Role=TRAIN Target Variable=Churn Target Label=' '

Posterior Probability Range	Number of Events	Number of Nonevents	Mean Posterior Probability	Percentage
0.75-0.80	2	0	0.77329	0.3636
0.70-0.75	19	0	0.72083	3.4545
0.65-0.70	28	6	0.66982	6.1818
0.60-0.65	40	12	0.62591	9.4545
0.55-0.60	61	24	0.57225	15.4545
0.50-0.55	51	37	0.52389	16.0000
0.45-0.50	50	52	0.47592	18.5455
0.40-0.45	21	62	0.42843	15.0909
0.35-0.40	5	37	0.37964	7.6364
0.30-0.35	0	19	0.32869	3.4545
0.25-0.30	1	18	0.28020	3.4545
0.20-0.25	0	5	0.23274	0.9091

Data Role=VALIDATE Target Variable=Churn Target Label=' '

Posterior Probability Range	Number of Events	Number of Nonevents	Mean Posterior Probability	Percentage
0.75-0.80	1	1	0.75927	0.4444
0.70-0.75	3	8	0.71884	2.4444
0.65-0.70	11	11	0.67421	4.8889
0.60-0.65	27	21	0.62324	10.6667
0.55-0.60	39	38	0.57432	17.1111
0.50-0.55	53	32	0.52710	18.8889
0.45-0.50	28	41	0.47774	15.3333
0.40-0.45	35	34	0.42721	15.3333
0.35-0.40	19	18	0.37780	8.2222
0.30-0.35	7	10	0.32484	3.7778
0.25-0.30	3	8	0.27966	2.4444
0.20-0.25	1	1	0.24945	0.4444

4.2 Model Comparison

Three distinct models—Decision Tree, HP Forest, and Gradient Boosting (Boost)—are thoroughly evaluated in the model comparison result. The goal of the study is to evaluate the models' performance using many metrics, and it focuses on both the training and validation datasets.

The Boost model has the greatest KS Statistic (0.459) and Cutoff (0.515) for the Bin-Based Two-Way Kolmogorov-Smirnov and KS Probability Cutoffs. This shows that the Boost model outperforms the other models in terms of discriminatory power, achieving a stronger distinction between the positive and negative classes based on the training data.

Fit Statistics Table
Target: Churn

Data Role=Train

Statistics	Boost	Tree	HPDMForest
Train: Bin-Based Two-Way Kolmogorov-Smirnov Probability Cutoff	0.53	0.54	0.43
Train: Kolmogorov-Smirnov Statistic	0.46	0.16	0.14
Train: Average Squared Error	0.21	0.23	0.25
Train: Roc Index	0.80	0.60	0.58
Train: Cumulative Percent Captured Response	17.63	12.70	11.75
Train: Percent Captured Response	7.55	5.46	5.77
Selection Criterion: Valid: Misclassification Rate	0.45	0.49	0.52
Train: Total Degrees of Freedom	550.00	550.00	.
Train: Frequency of Classified Cases	.	.	550.00
Train: Divisor for ASE	1100.00	1100.00	1100.00
Train: Gain	76.26	27.02	17.52
Train: Gini Coefficient	0.60	0.20	0.16
Train: Bin-Based Two-Way Kolmogorov-Smirnov Statistic	0.45	0.16	0.13
Train: Kolmogorov-Smirnov Probability Cutoff	0.52	0.39	0.43
Train: Cumulative Lift	1.76	1.27	1.18
Train: Lift	1.54	1.11	1.18
Train: Maximum Absolute Error	0.72	0.80	0.60
Train: Misclassification Rate	0.28	0.42	0.45
Train: Sum of Frequencies	550.00	550.00	550.00
Train: Root Average Squared Error	0.45	0.48	0.50
Train: Cumulative Percent Response	89.09	64.20	59.40
Train: Percent Response	77.78	56.25	59.40
Train: Sum of Squared Errors	226.72	258.41	270.80
Train: Sum of Case Weights Times Freq	1100.00	.	.
Train: Number of Wrong Classifications	.	.	248.00

Data Role=Valid

Statistics	Boost	Tree	HPDMForest
Valid: Kolmogorov-Smirnov Statistic	0.093	0.054	0.047
Valid: Average Squared Error	0.259	0.253	0.257
Valid: Roc Index	0.511	0.521	0.488
Valid: Bin-Based Two-Way Kolmogorov-Smirnov Probability Cutoff	0.507	0.386	0.424
Valid: Cumulative Percent Captured Response	7.930	11.070	9.912
Valid: Percent Captured Response	3.965	5.315	4.846
Valid: Frequency of Classified Cases	.	.	450.000
Valid: Divisor for VASE	900.000	900.000	900.000
Valid: Gain	20.705	10.701	0.881
Valid: Gini Coefficient	0.021	0.042	-0.025
Valid: Bin-Based Two-Way Kolmogorov-Smirnov Statistic	0.079	0.053	0.010
Valid: Kolmogorov-Smirnov Probability Cutoff	0.499	0.201	0.448
Valid: Cumulative Lift	0.793	1.107	0.991
Valid: Lift	0.811	1.087	0.991
Valid: Maximum Absolute Error	0.763	1.000	0.615
Valid: Misclassification Rate	0.453	0.493	0.522
Valid: Sum of Frequencies	450.000	450.000	450.000
Valid: Root Average Squared Error	0.509	0.503	0.507
Valid: Cumulative Percent Response	40.000	55.842	50.000
Valid: Percent Response	40.909	54.839	50.000
Valid: Sum of Squared Errors	232.728	227.401	231.655
Valid: Sum of Case Weights Times Freq	900.000	.	.
Valid: Number of Wrong Classifications	.	.	235.000

As we go on to the Model Fit Statistics, we see some significant variations amongst the models, including misclassification rates, Gini coefficients, and cumulative lift. With a lower misclassification rate (0.45333) than Decision Tree and HP Forest, Boost performs better in the training dataset. Boost has a significantly higher Gini coefficient (0.60), which suggests more discriminating capacity. Additionally, the model effectiveness metric, cumulative lift, is highest for Boost (1.76).

Fit Statistics
Model Selection based on Valid: Misclassification Rate (_VMISC_)

Selected Model	Model Node	Model Description	Valid: Misclassification Rate	Train: Average Squared Error	Train: Misclassification Rate	Valid: Average Squared Error
Y	Boost	Gradient Boosting	0.45333	0.20611	0.28364	0.25859
	Tree	Decision Tree	0.49333	0.23492	0.41818	0.25267
	HPDMForest	HP Forest	0.52222	0.24618	0.45091	0.25739

But when evaluating the models on the validation dataset, Boost continues to outperform Decision Tree (0.41818) and HP Forest (0.45091) with a reduced misclassification rate (0.28364). The fact that Boost's cumulative lift is still the greatest at 0.793 indicates that the model is still good at identifying positive situations.

Fit Statistics
Model Selection based on Valid: Misclassification Rate (_VMISC_)

Selected Model	Model Node	Model Description	Valid: Misclassification Rate	Train: Average Squared Error	Train: Misclassification Rate	Valid: Average Squared Error
Y	Boost	Gradient Boosting	0.45333	0.20611	0.28364	0.25859
	Tree	Decision Tree	0.49333	0.23492	0.41818	0.25267
	HPDMForest	HP Forest	0.52222	0.24618	0.45091	0.25739

For both training and validation datasets, a thorough analysis of model predictions is given in the Event Classification Table. To help with a more detailed knowledge of model behaviour, it discloses the amount of true positives, true negatives, false positives, and false negatives for each model.

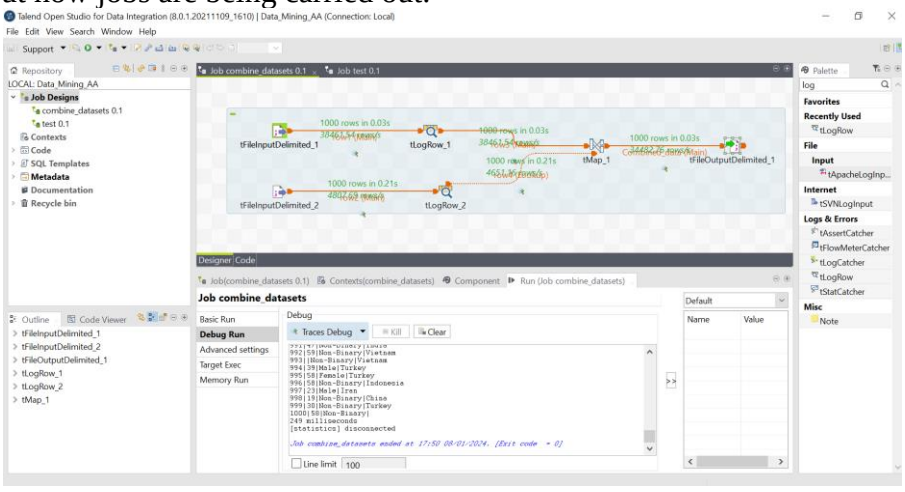
Event Classification Table
Model Selection based on Valid: Misclassification Rate (_VMISC_)

Model Node	Model Description	Data Role	Target	Target Label	False Negative	True Negative	False Positive	True Positive
Boost	Gradient Boosting	TRAIN	Churn		77	193	79	201
Boost	Gradient Boosting	VALIDATE	Churn		93	112	111	134
HPDMForest	HP Forest	TRAIN	Churn		91	115	157	187
HPDMForest	HP Forest	VALIDATE	Churn		83	71	152	144
Tree	Decision Tree	TRAIN	Churn		46	88	184	232
Tree	Decision Tree	VALIDATE	Churn		51	52	171	176

To sum up, the study shows that Gradient Boosting (Boost) performs better than the other models on a number of different measures. As a result of its better cumulative lift, reduced misclassification rates, and higher KS Statistic, it is the model of choice for predicting the target variable (Churn) in this situation. According to the findings, Boost offers a reliable and practical solution for the given classification assignment.

4.3 Data Integration logs

The logging feature in Talend Data Integration is intended to give information about how jobs are being carried out.



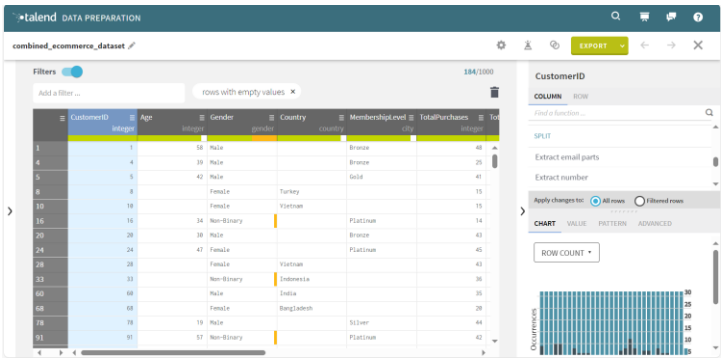
(a) Duration of Time:
The phrase "249 milliseconds" denotes how long the task or process took to complete. It shows that it took 249 milliseconds to do the task.

(b) Disconnection and Statistics:
The words "[statistics] disconnected" suggest that the job's logging or statistics part is coming to an end. The message "Job combine_datasets ended" denotes that the "combine_datasets" job has been completed. Given that many systems interpret an exit value of 0 as properly completed, the "[Exit code = 0]" implies that the job finished successfully.

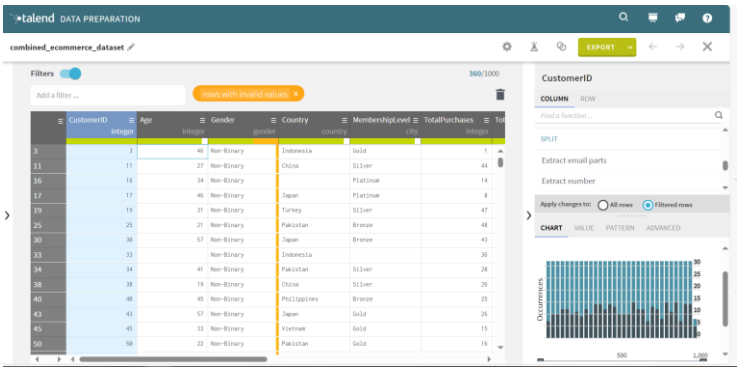
4.4 Data Quality reports

The quality of the dataset is checked in Talend data preparation before it is cleaned in the SAS studio.

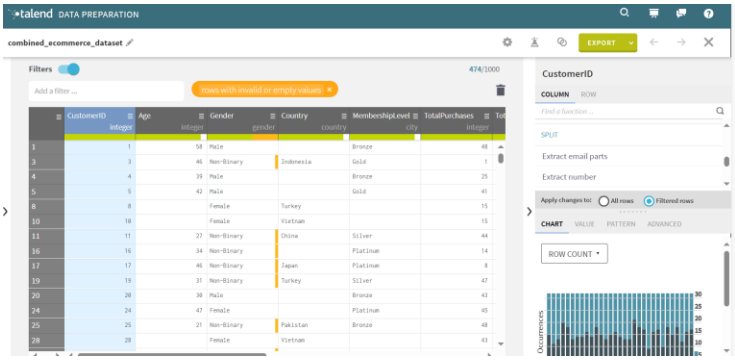
(a) Identify rows with empty columns
This filter helps in identifying all the columns that have empty rows. It helps to identify that the column Age, Country and Membership level has empty rows.



(b) Identify rows with invalid values
This filter helps in identifying all the columns that have invalid rows. It helps to identify that the gender column has invalid column has it shows Non-binary value.



(c) Identify rows with both invalid and empty rows
This filter helps to point out the overall view of the dataset that is not suitable to do analysis



5.0 Reflection and learning outcomes

With machine learning models, this case study offered insights into the subtleties and complexity of customer churn prediction, making it an invaluable learning experience. During the process, several significant thoughts and learning objectives were apparent.

5.1 Challenges Faced

One significant obstacle that was faced was the requirement to properly manage missing data. Because there were missing values in the dataset, careful imputation techniques were needed to preserve data integrity. It was also difficult to evaluate and maximize the performance of several machine learning models, particularly when deciding which one was best for the job at hand. A constant problem was making sure the model could be applied to fresh data without overfitting.

5.2 How those challenges were overcome

Imputation approaches, such as mean or median imputation for numerical variables and mode imputation for categorical characteristics, were utilized to address the missing data. Important information in the dataset might be preserved by inputting missing values. A methodical strategy was employed, utilizing decision trees, random forests, gradient boosting, and HP Forest, to address the model selection difficulty. A thorough grasp of the advantages and disadvantages of each model was possible via comparing how well they performed on different measures. The Gradient Boosting model, which showed better predicting abilities, was chosen with the assistance of this iterative procedure.

In addition, reducing overfitting required careful hyperparameter adjustment, such as changing the gradient boosting model's learning rates and tree depths. To make sure the model could be applied to data that had not yet been seen, cross-validation techniques were also used. To increase the robustness of the model, regularization approaches including feature significance analysis and pruning were used.