UNIVERSITI
MALAYA

WQD7005 DATA MINING
2023/2024 S1

ALTERNATIVE ASSESSMENT 1

Matric Number: S2164046

Name: HU LIANGLIANG

Case Study: E-Commerce Customer Behaviour Analysis

Tasks Data Import and Preprocessing:

- Import your dataset into SAS Enterprise Miner, handle missing values, and specify variable roles. [15 marks]
- Decision Tree Analysis: Create a decision tree model in SAS Enterprise Miner to analyse customer behaviour. [20 marks]
- Ensemble Methods: Apply Bagging and Boosting, using the Random Forest algorithm as a Bagging example. [10 marks]

Deliverables:

- A report detailing each step of the process, including the rationale behind your choices and any challenges faced.
- An analysis of the decision tree and ensemble methods, with insights into customer behavior and suggestions for business strategy. [5 marks]

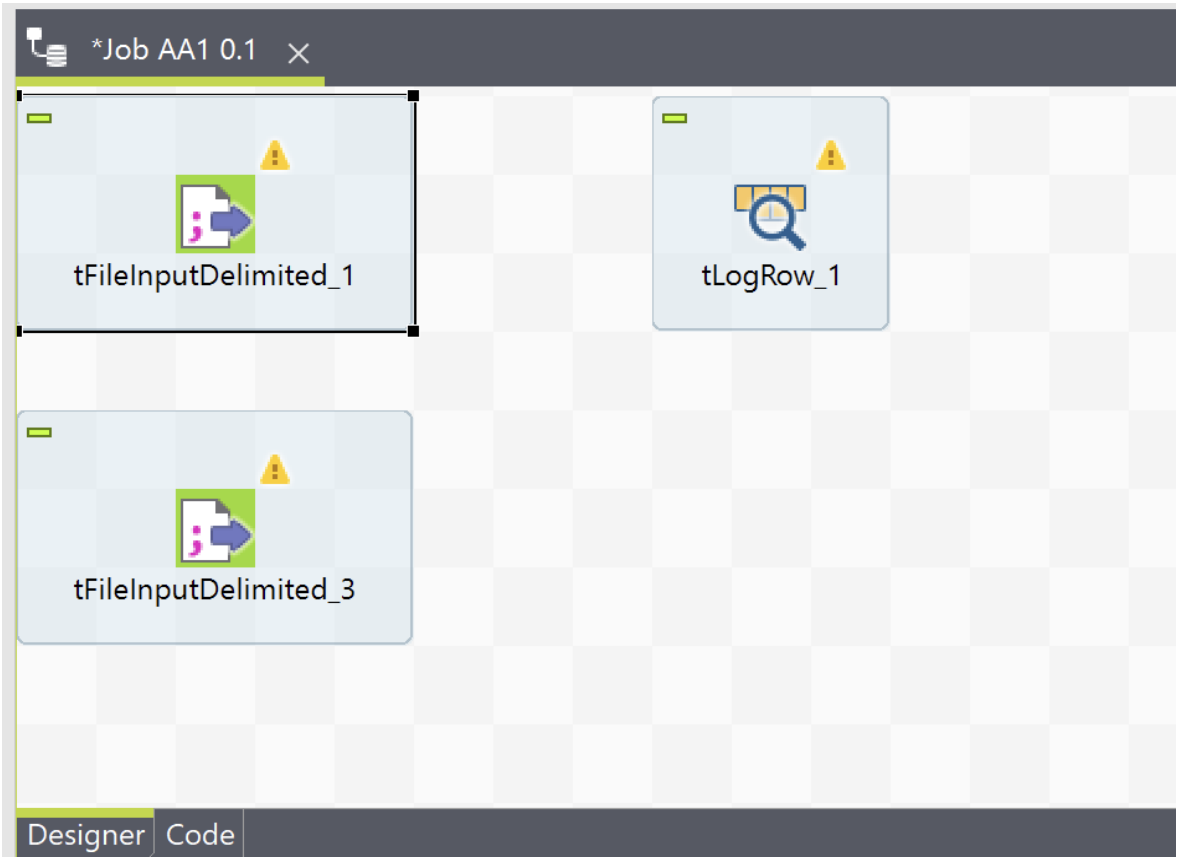
I will sequentially process the dataset in the following order to conduct effective analysis of e-commerce customer behavior. I will use Data Integration (DI) to integrate the entire dataset, employ Data Preparation (DP) for initial transformation and exploration of the data, and finally utilize SAS EM to handle tasks such as imputing missing values and specifying variable roles. This comprehensive approach is aimed at facilitating subsequent decision tree analysis and employing various ensemble methods.

I randomly generated a dataset according to different rules, aiming to produce results that meet the task requirements and closely resemble real-life scenarios.

First, let's preview the datasets. There are two distinct datasets available—one containing detailed customer personal information and the other containing customer purchase information. To proceed, we need to merge these two disparate datasets.

	CustomerID	Age	Gender	Location	Membersh								
1	85751	41	3561.19	Home Goc	Professiona	9	0	#######	78				
2	86667	49	Female	Shenzhen	Bronze	85705	46	3842.98	Books	Professiona	6	0	#######	87
3	86533	21	Male	Guangzho	Gold	86155	10	440.58	Clothing	Unemploy	7	1	2023/2/4	331
4	85972	24	Female	Hangzhou	Bronze	85984	23	1854.61	Sports	Unemploy	2	0	2023/9/2	121
5	86227	64	Male	Chongqin	Bronze	85985	42	5119.77	Home Goc	Self-Emplo	9	0	2023/9/4	119
6	86346	23	Male	Chongqin	Bronze	86557	87	35488.83	Home Goc	Self-Emplo	18	0	#######	127
7	86645	32	Male	Wuhan	Silver	86462	26	2751.18	Electronics	Self-Emplo	29	0	2023/7/2	183
8	85701	65	Female	Beijing	Gold	85719	38	1680.13	Electronics	Unemploy	4	0	#######	87
9	85905	38	Female	Wuhan	Bronze	85777	93	17380.37	Electronics	Self-Emplo	9	1	#######	277
10	85980	45	Male	Shenzhen	Bronze	86434	52	7977.06	Clothing	Self-Emplo	16	0	#######	133

1. Data Integration (DI)

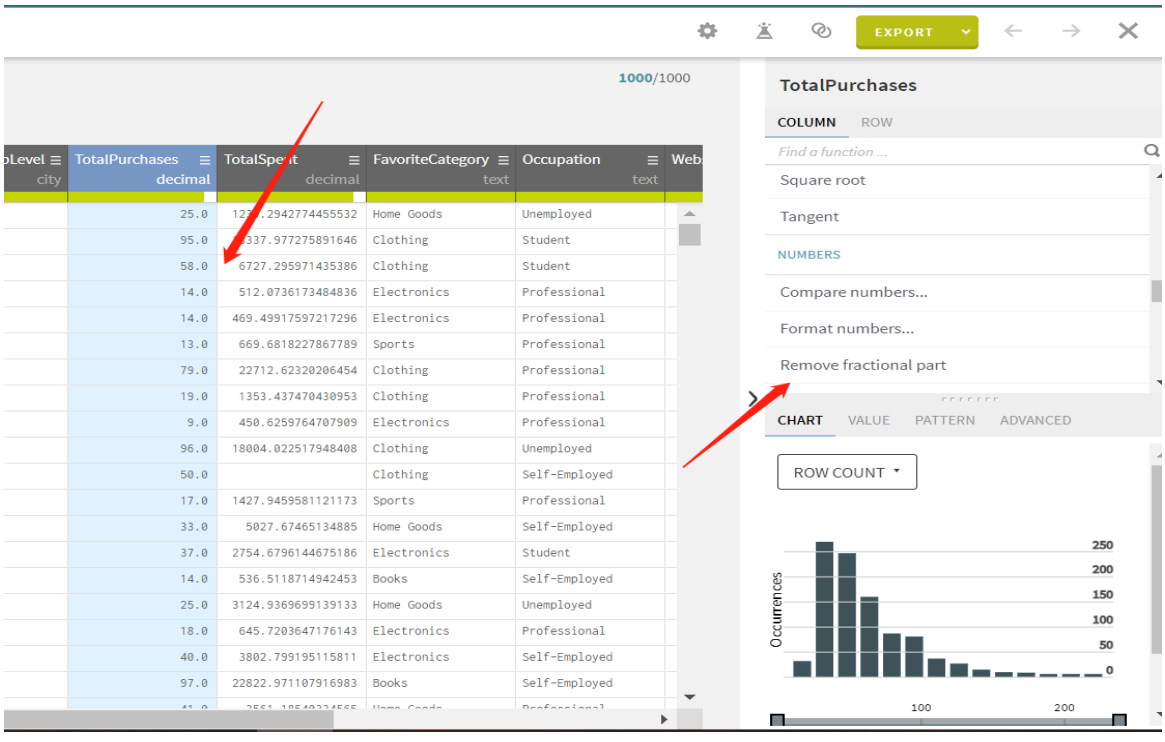


Connecting two different datasets

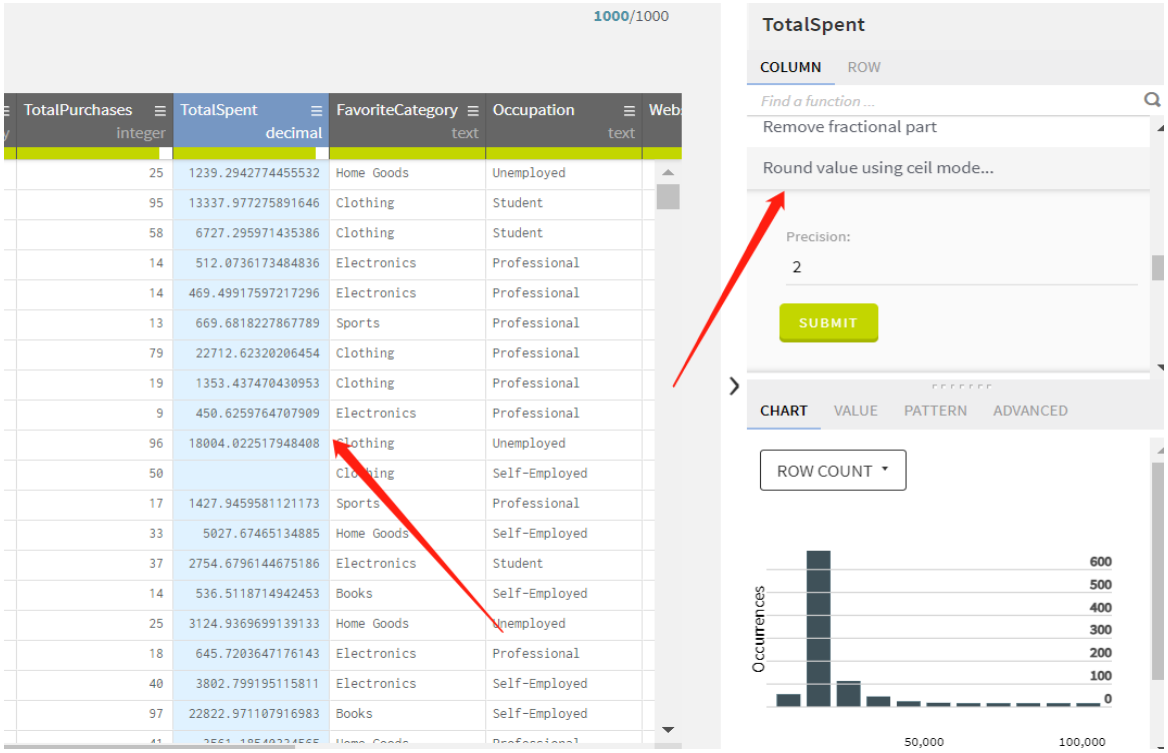
2. Data Preparation (DP)

After merging the distinct datasets, we need to conduct initial exploration and processing of the data.

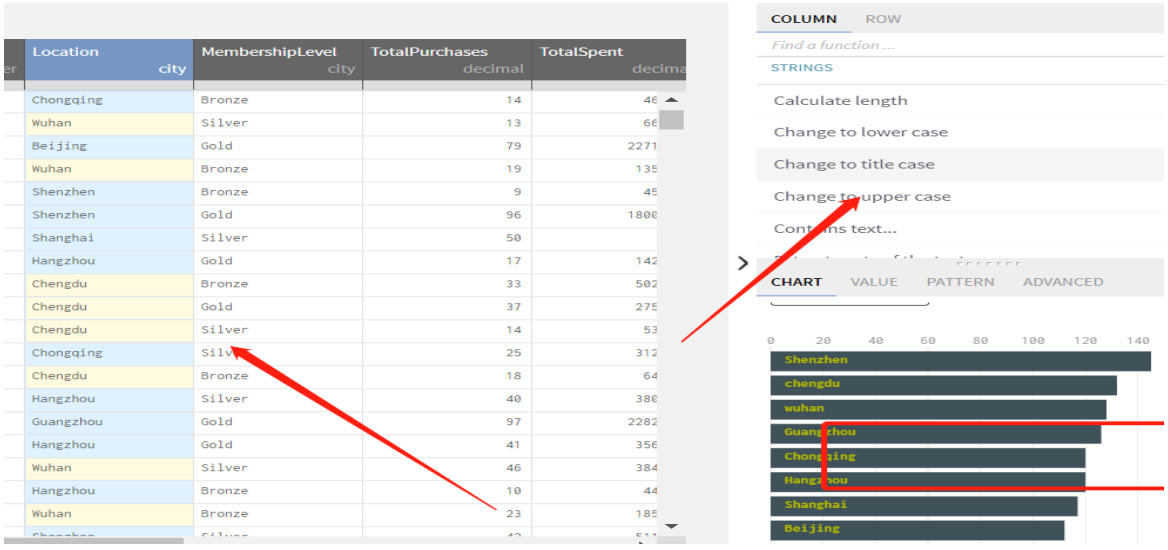
- 2.1 It is not reasonable for TotalPurchases to have decimal points, so I removed the decimal points from TotalPurchases.



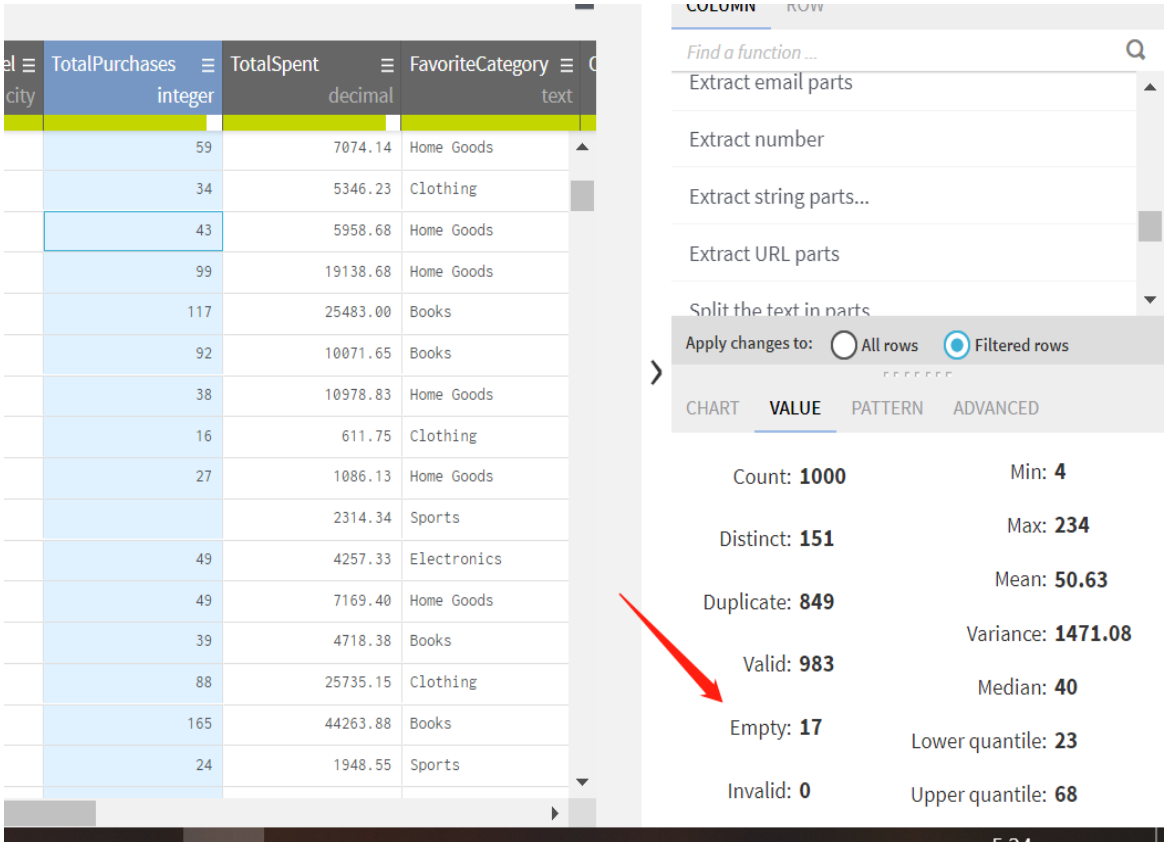
- 2.2 Many decimal places in TotalSpent are not reasonable, so I retained two decimal places for TotalSpent.

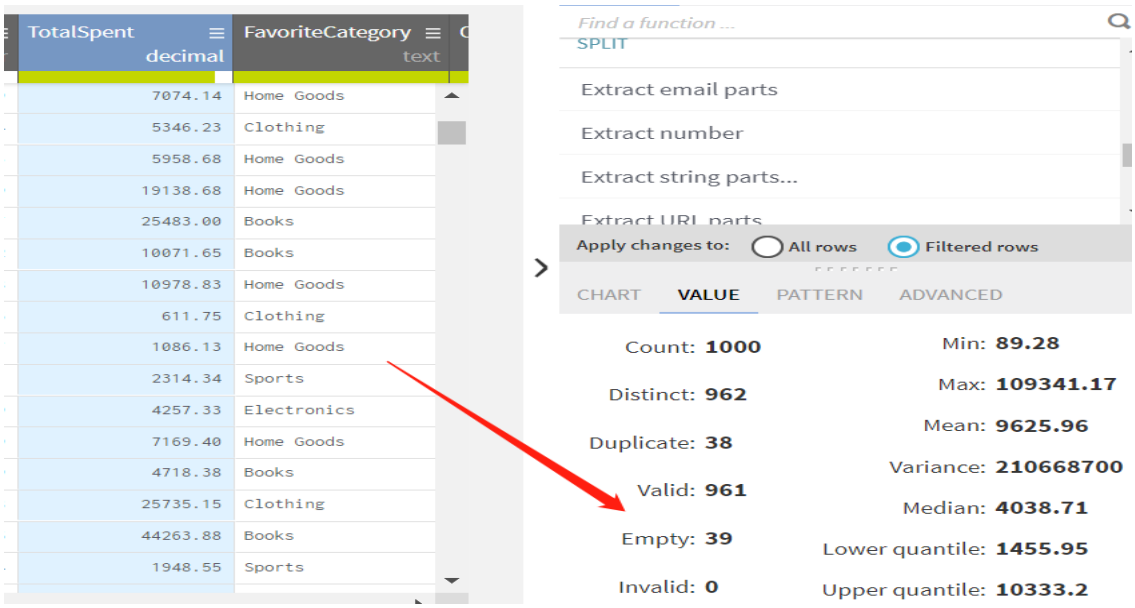


- 2.3 The capitalization of city names in the location data is inconsistent, with some having uppercase initial letters and others having lowercase initial letters. This is not reasonable, so I have standardized all city names to have uppercase initial letters.

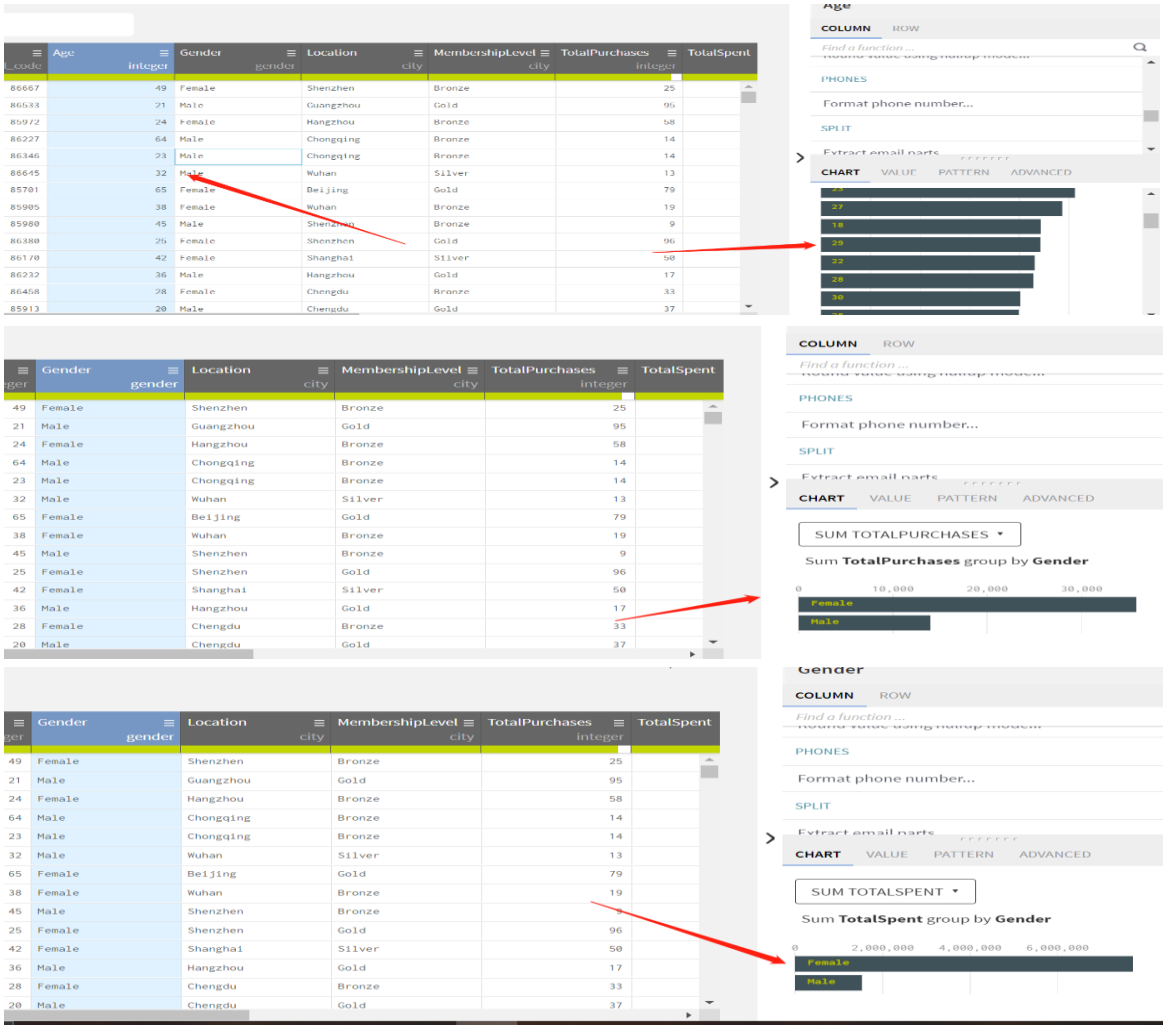


- 2.4 I observed that both TotalPurchases and TotalSpent have missing values, with 17 and 39 missing values respectively. I won't handle them at this point; instead, I will examine the overall structure of the data inside SAS EM before proceeding with any imputation or treatment.

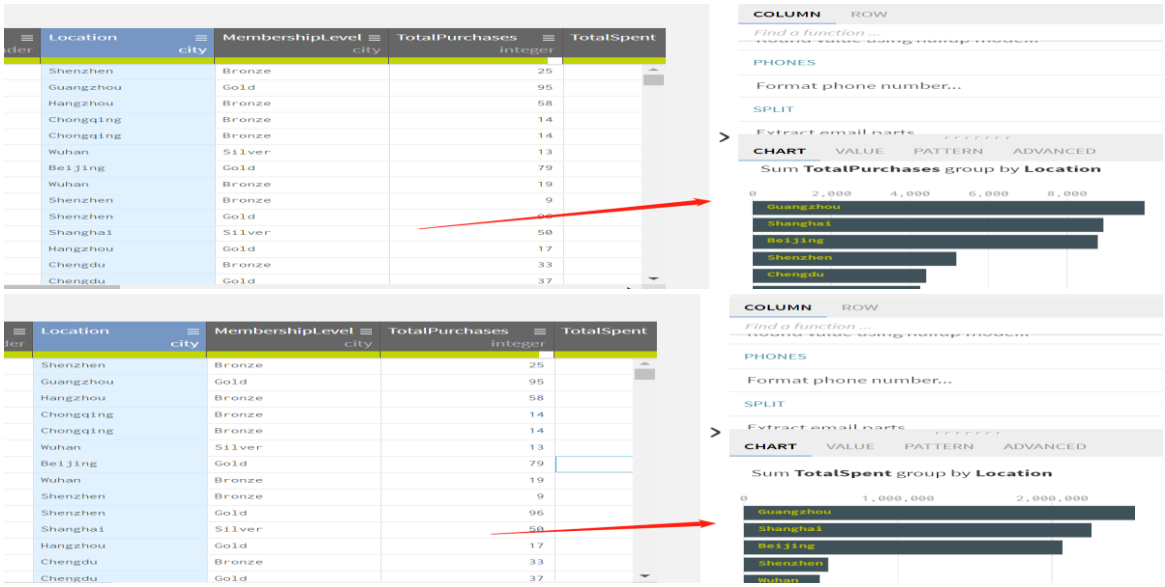




- 2.5 Observing this dataset, it appears that the majority of individuals are young, and the spending capacity of females is significantly higher than that of males. This aligns well with real-world scenarios.

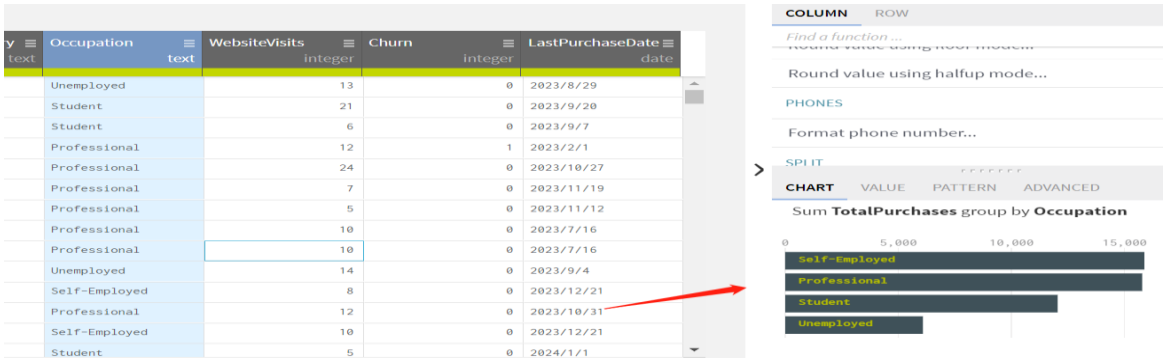


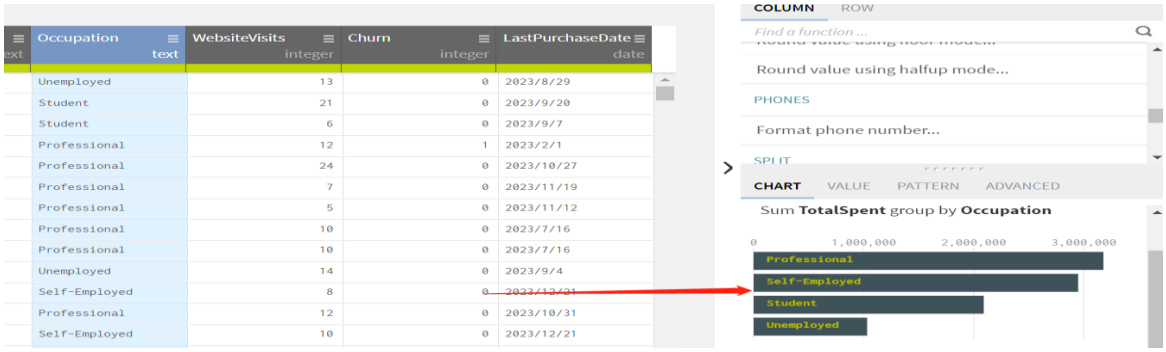
- 2.6 It can be observed that major cities like Beijing, Guangzhou, and Shanghai exhibit significantly higher purchase frequencies and total expenditures compared to other cities.

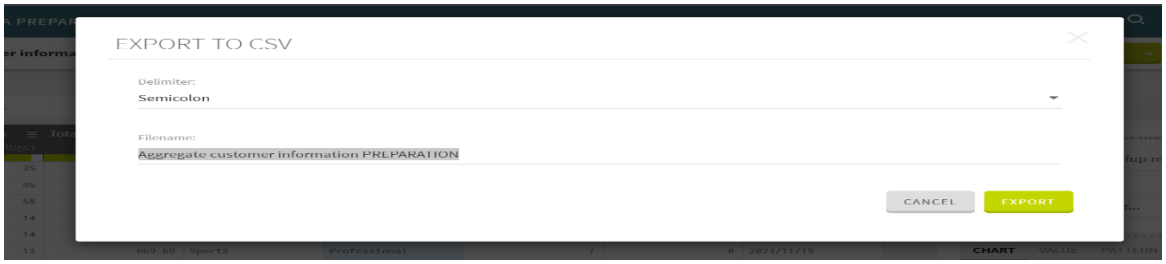


- 2.7 It is noticeable that individuals with employment or student status tend to have significantly higher purchase frequencies and total expenditures compared

to those who are unemployed.

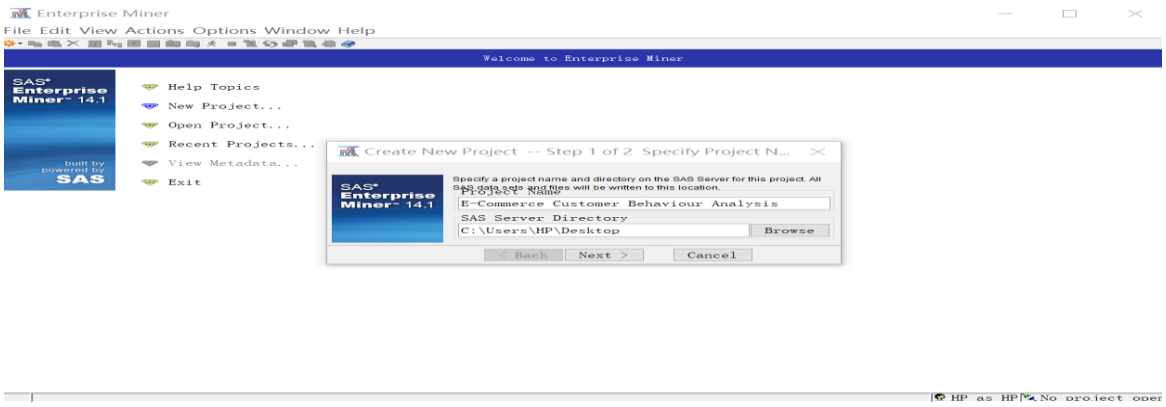


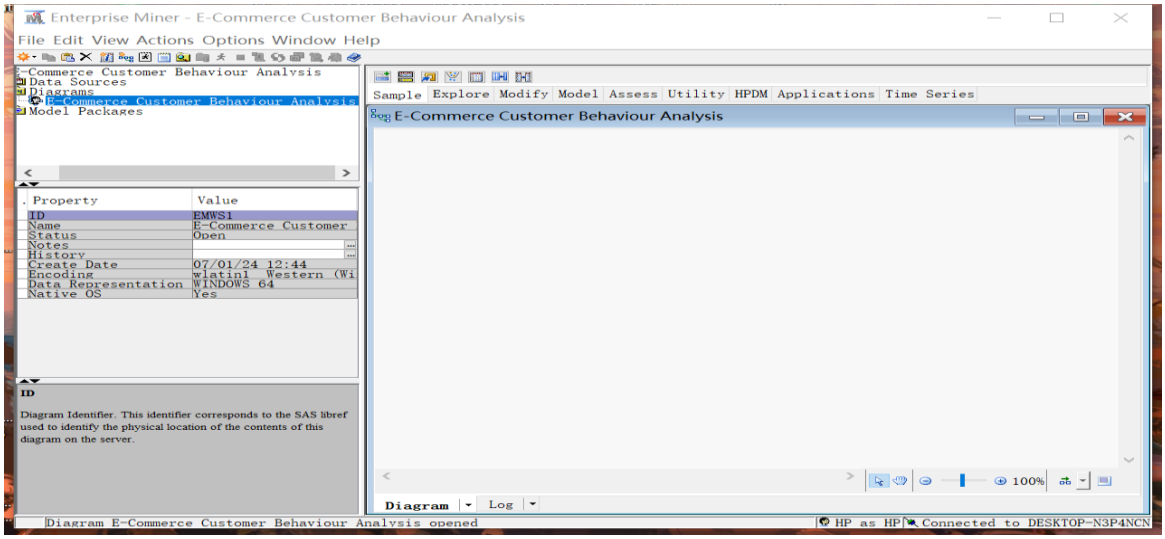




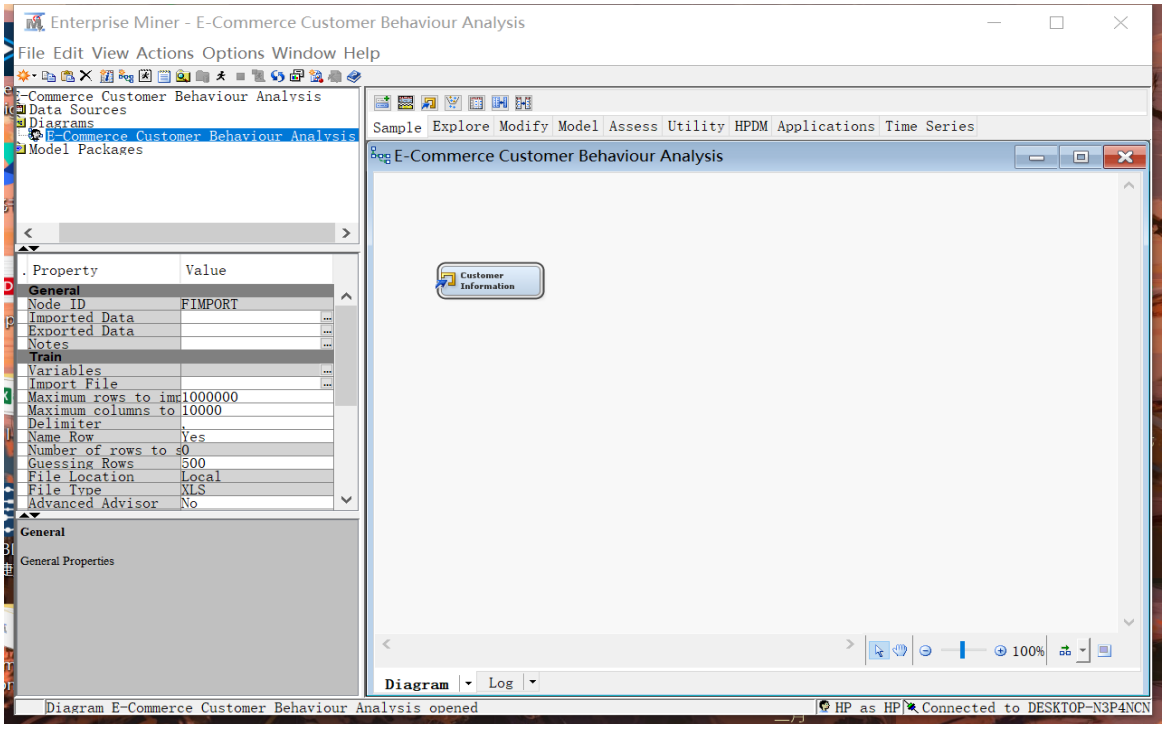
3. SAS EM

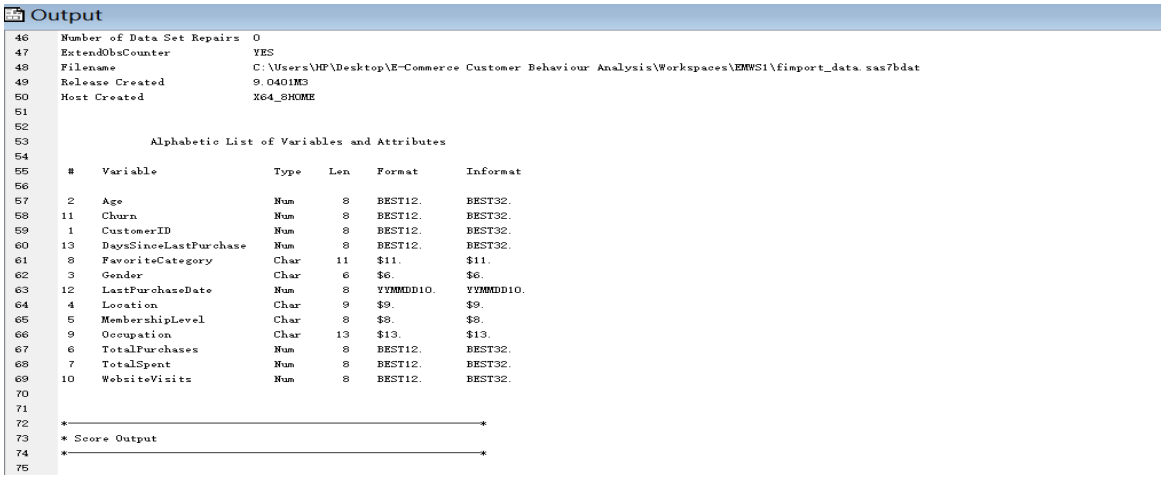
3.1 Creating Project Files





3.2 Import of data and initial checking

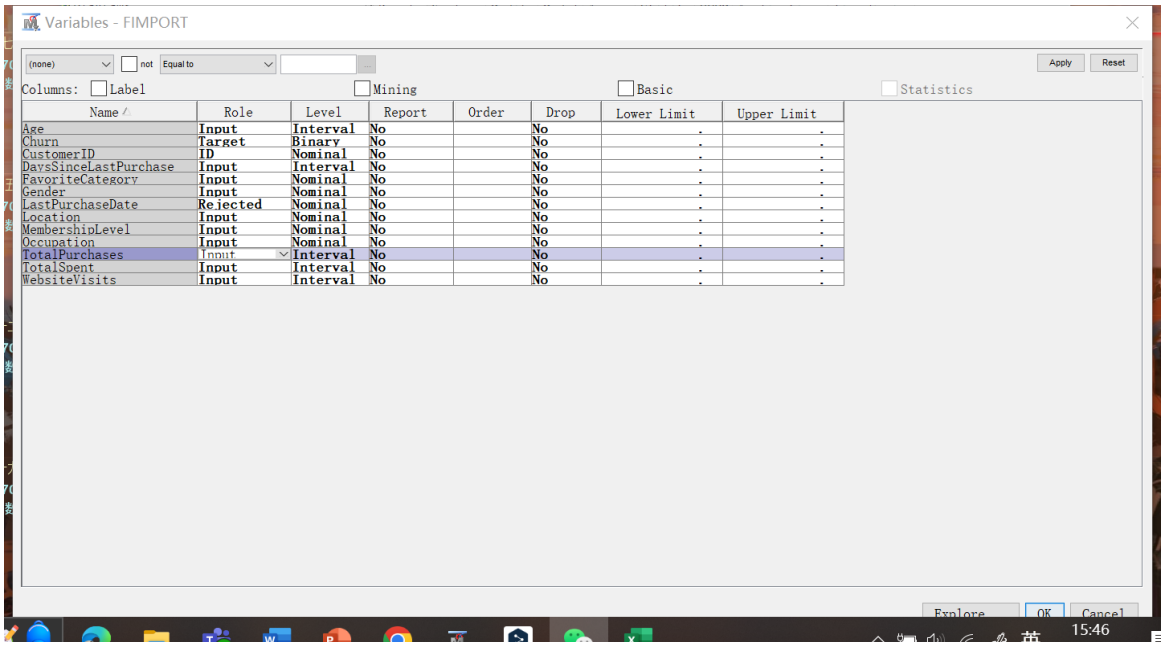




3.3 specify variable roles

reason:

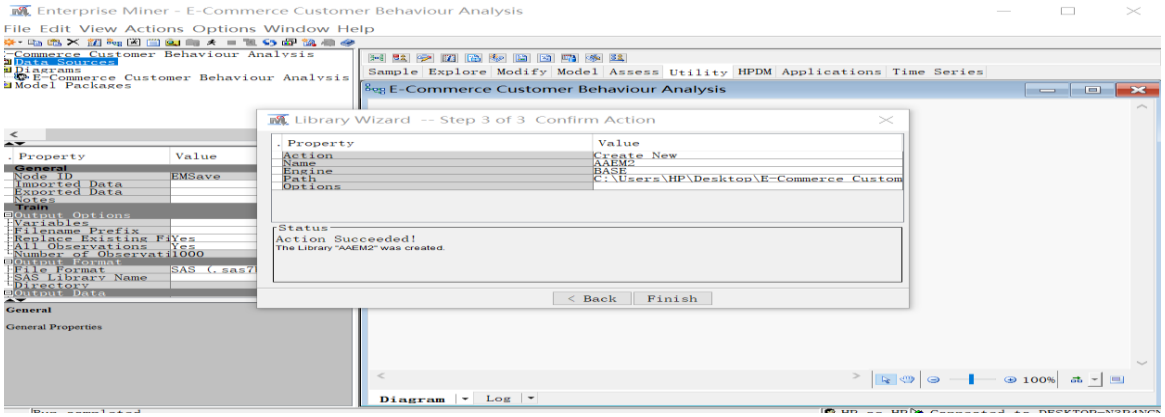
- CustomerID: This is a unique identifier for each customer and holds no predictive value for the model. It should be set as ID and not used as a feature in decision tree or random forest models.
- Age: This is a continuous numerical variable and should be set as Interval, serving as an input feature.
- Gender, Location, MembershipLevel, FavoriteCategory, Occupation: These are categorical variables and should be set as Nominal, serving as input features.
- TotalPurchases, TotalSpent, and WebsiteVisits: These are continuous numerical variables and should be set as Interval, serving as input features.
- Churn: This is the target variable indicating whether a customer has churned or not. Considering its binary nature (0 or 1), it should be set as Target and Binary.
- LastPurchaseDate: This is a date variable. For constructing decision trees and random forest models, date variables are typically not directly used. Since it has already been transformed into days since the last purchase, it can be set as Rejected.

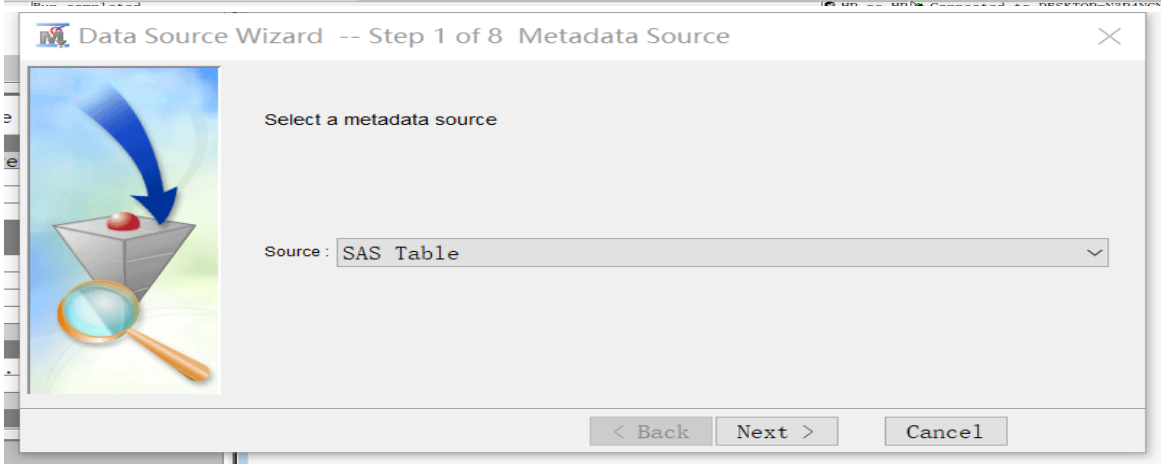


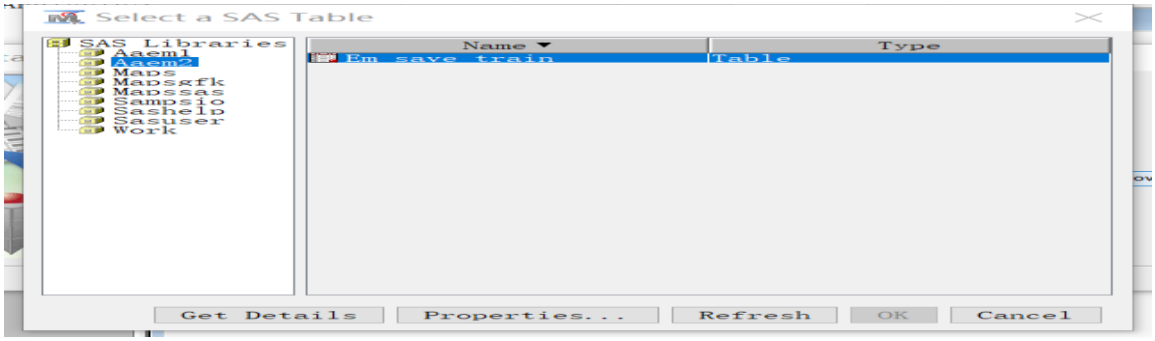
Name	Role	Level	Report	Order	Drop	Lower Limit	Upper Limit
Age	Input	Interval	No		No	.	.
Churn	Target	Binary	No		No	.	.
CustomerID	ID	Nominal	No		No	.	.
DaysSinceLastPurchase	Input	Interval	No		No	.	.
FavoriteCategory	Input	Nominal	No		No	.	.
Gender	Input	Nominal	No		No	.	.
LastPurchaseDate	Rejected	Nominal	No		No	.	.
Location	Input	Nominal	No		No	.	.
MembershipLevel	Input	Nominal	No		No	.	.
Occupation	Input	Nominal	No		No	.	.
TotalPurchases	Input	Interval	No		No	.	.
TotalSpent	Input	Interval	No		No	.	.
WebsiteVisits	Input	Interval	No		No	.	.

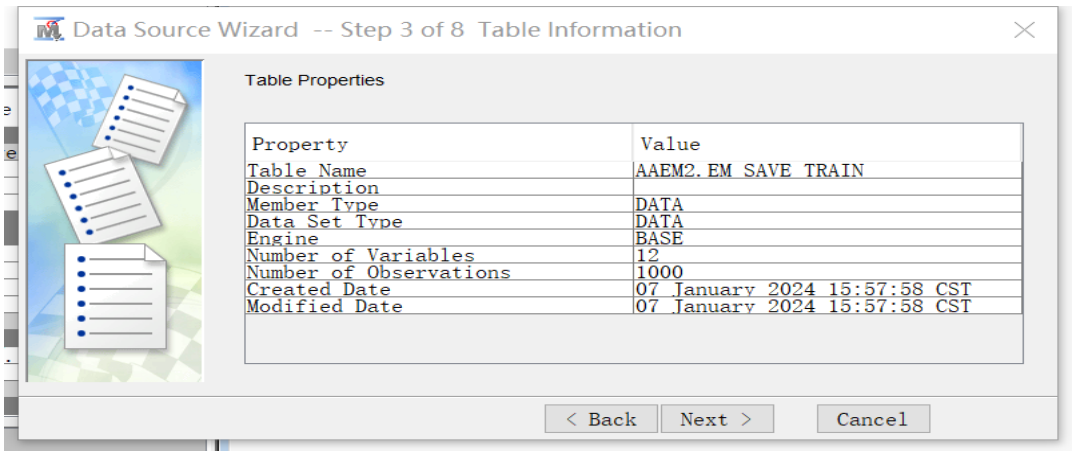
```
*------------------------------------------------------------*



Exported Attributes for TRAIN Port

                 Measurement    Frequency
Role               Level          Count

ID               NOMINAL            1
INPUT            INTERVAL           5
INPUT            NOMINAL            5
REJECTED         NOMINAL            1
TARGET           BINARY             1
```

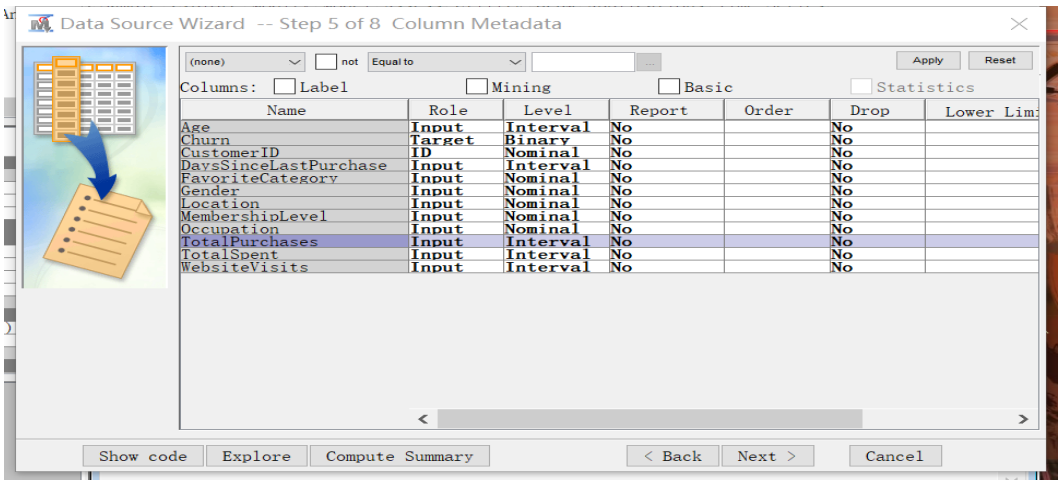
3.4 Creating Data Sources

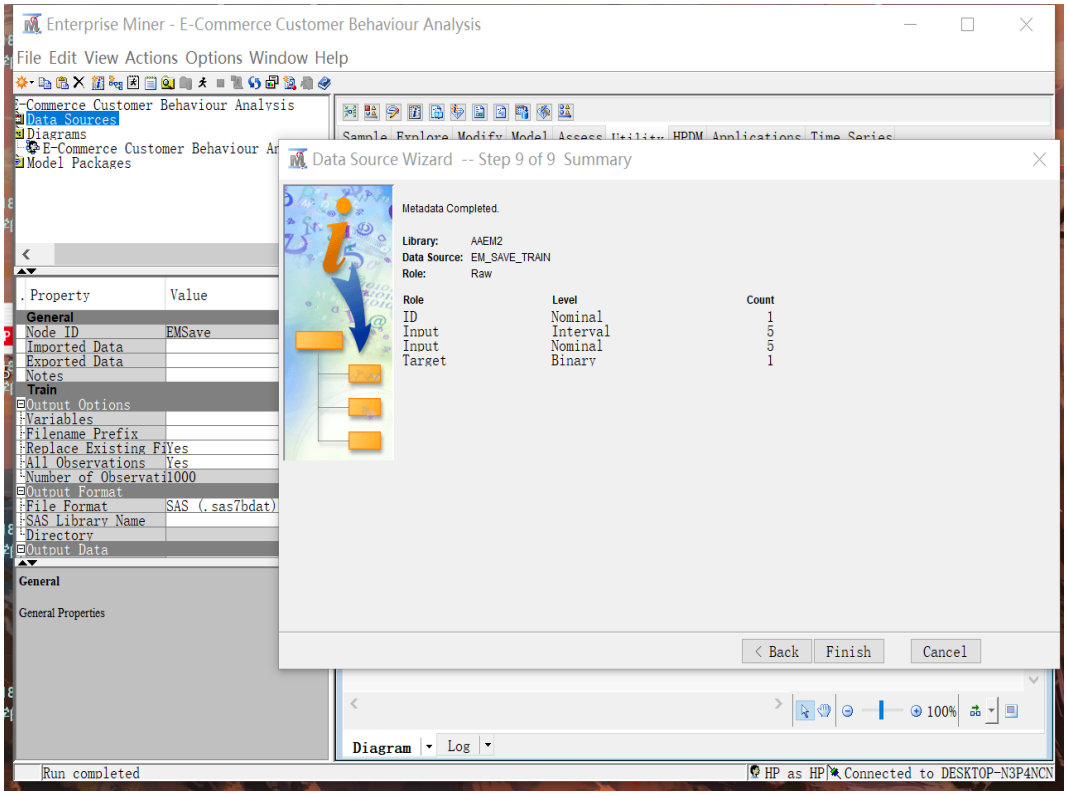










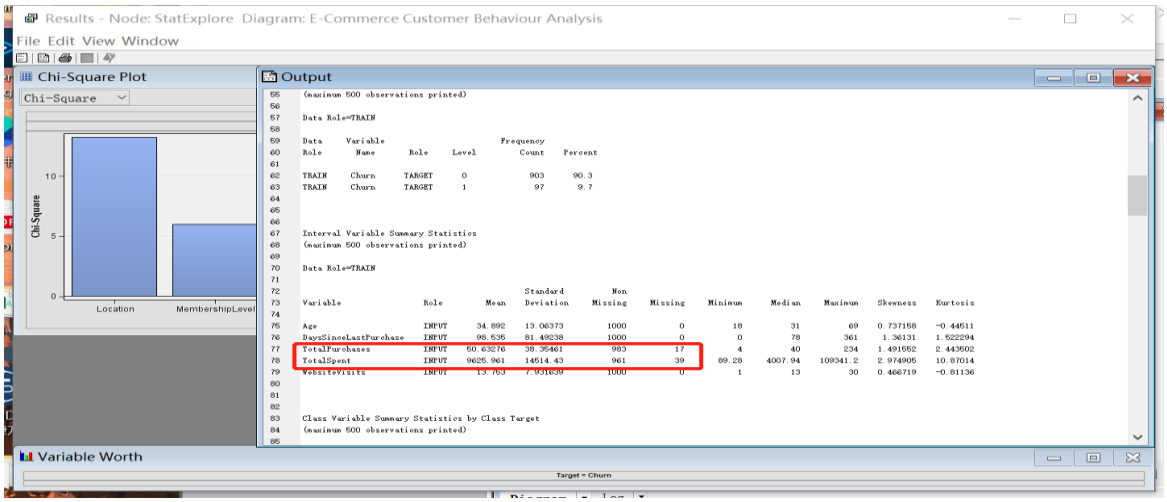


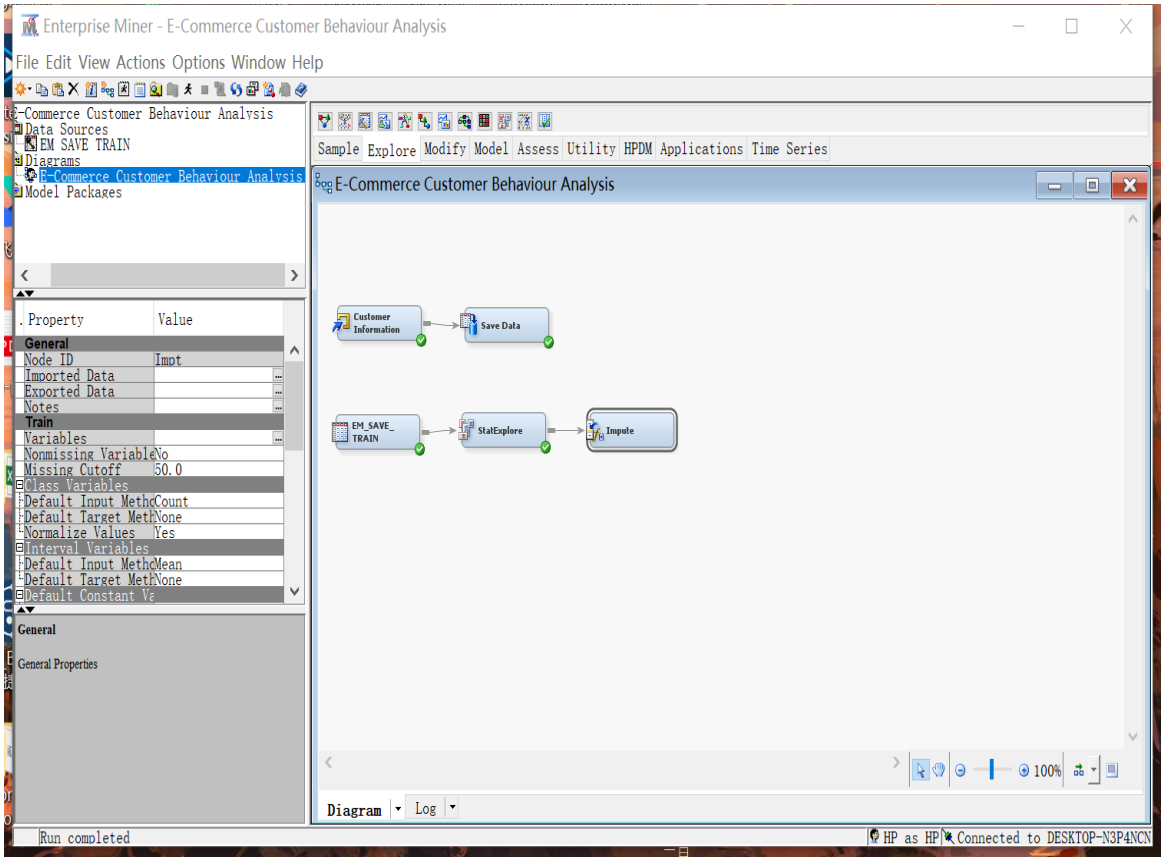
3.5 Processing data for missing values

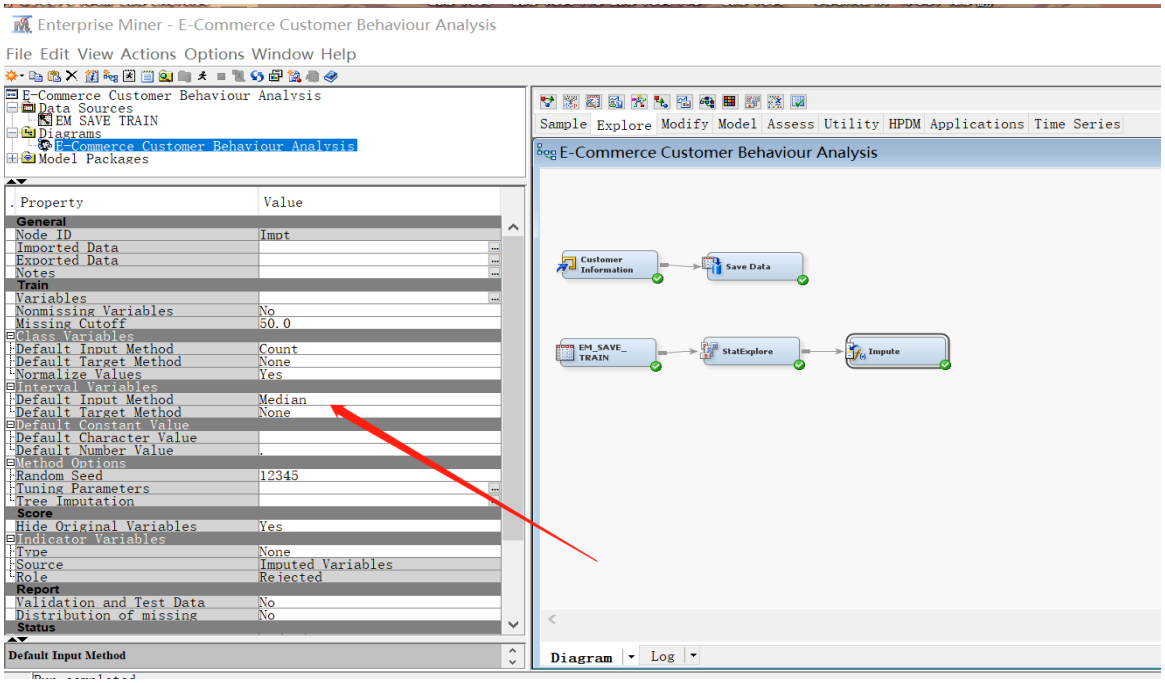
Choose median to due missing values

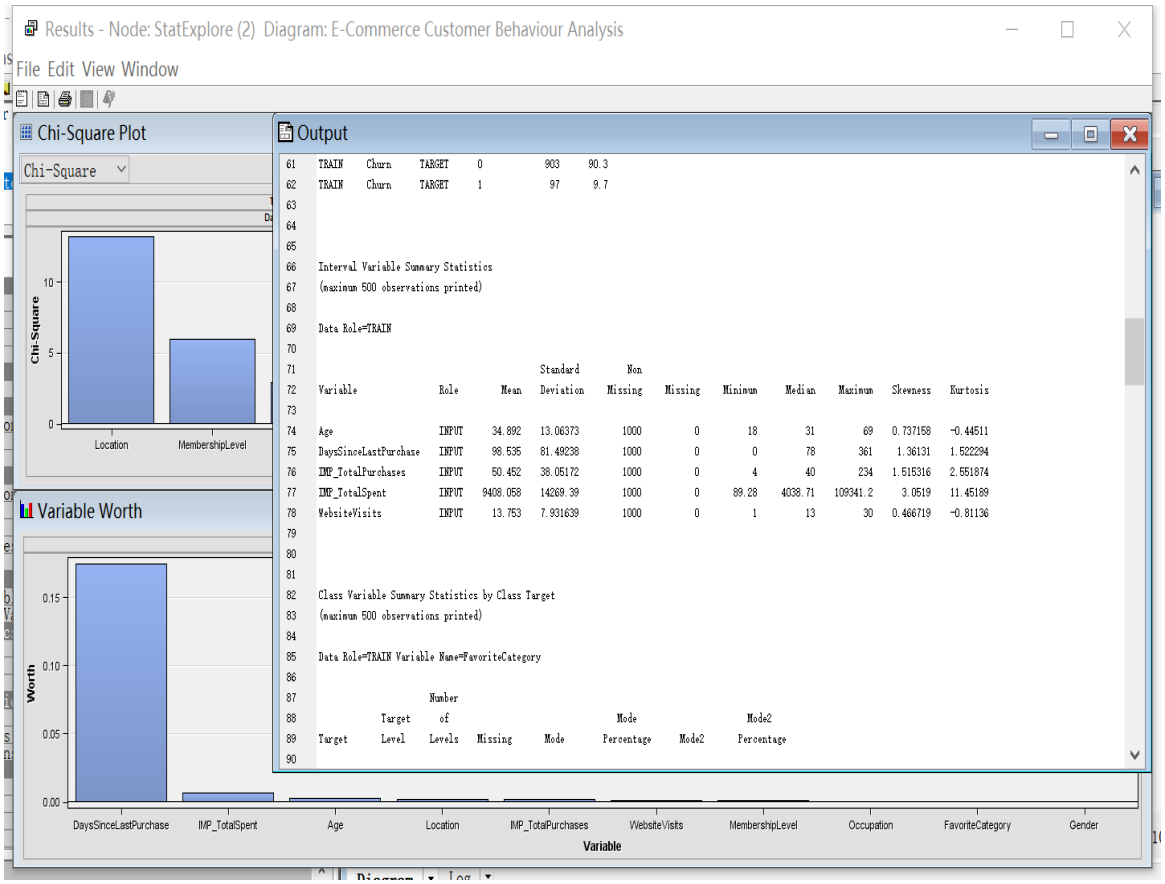
Reason： After checking the data structure found that there are TotalPurchases and

TotalSpent so that there are two columns of missing values, histogram to check the distribution of data is indeed more skewed, so the use of the median to fill in the missing values, which is particularly suitable for dealing with skewed distribution (i.e., asymmetric or long-tailed distribution) of numerical data. The median is less susceptible to extreme values or outliers than the mean.


Results - Node: StatExplore Diagram: E-Commerce Customer Behaviour Analysis
File Edit View Window
Chi-Square Plot
Chi-Square
Chi-Square
Location
MembershipLevel
Output
(maximum 600 observations printed)
Data Role=TRAIN
Data Role Variable Name Role Level Frequency Count Percent
TRAIN Churn TARGET 0 903 90.3
TRAIN Churn TARGET 1 97 9.7
Interval Variable Summary Statistics
(maximum 600 observations printed)
Data Role=TRAIN
Variable Role Mean Standard Deviation Non Missing Missing Minimum Median Maximum Skewness Kurtosis
Age INPUT 34.892 13.06373 1000 0 18 31 69 0.737168 -0.44611
DaysSinceLastPurchase INPUT 98.535 81.49238 1000 0 0 78 361 1.36131 1.522294
TotalPurchases INPUT 60.63276 38.35461 983 17 4 40 234 1.491652 2.443502
TotalSpent INPUT 9625.961 14514.43 961 39 89.28 4007.94 109341.2 2.974905 10.87014
WebsiteVisits INPUT 13.753 7.951639 1000 0 1 13 30 0.466719 -0.81136
Class Variable Summary Statistics by Class Target
(maximum 600 observations printed)
Variable Worth
Target = Churn


Enterprise Miner - E-Commerce Customer Behaviour Analysis
File Edit View Actions Options Window Help
E-Commerce Customer Behaviour Analysis
Data Sources
EM SAVE TRAIN
Diagrams
E-Commerce Customer Behaviour Analysis
Model Packages
Property
Value
General
Node ID
Imported Data
Exported Data
Notes
Train
Variables
Nonmissing Variable
Missing Cutoff
50.0
Class Variables
Default Input Meth
Default Target Meth
Normalize Values
Yes
Interval Variables
Default Input Meth
Mean
Default Target Meth
None
Default Constant Va
General
General Properties
Sample
Explore
Modify
Model
Assess
Utility
HPDM
Applications
Time Series
E-Commerce Customer Behaviour Analysis
Customer Information
Save Data
EM_SAVE_ TRAIN
StatExplore
Impute
100%
Diagram
Log
Run completed
HP as HP
Connected to DESKTOP-N3P4NCN

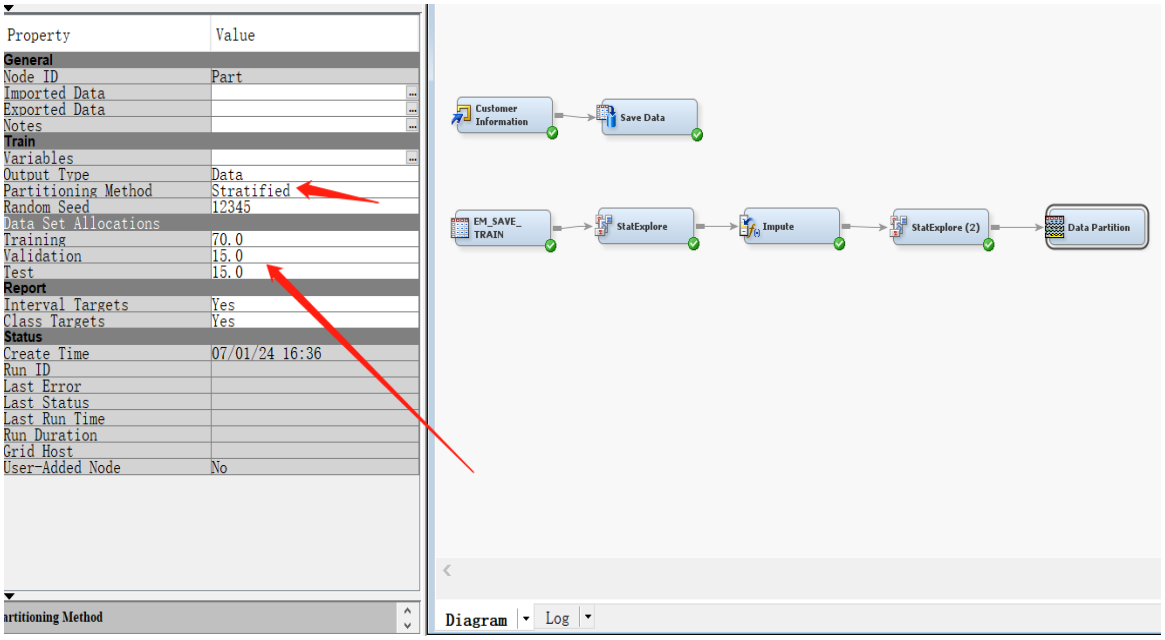

Enterprise Miner - E-Commerce Customer Behaviour Analysis
File Edit View Actions Options Window Help
E-Commerce Customer Behaviour Analysis
Data Sources
EM SAVE TRAIN
Diagrams
E-Commerce Customer Behaviour Analysis
Model Packages
Property
Value
General
Node ID
Imput
Imported Data
Exported Data
Notes
Train
Variables
Nonmissing Variables
No
Missing Cutoff
50.0
Class Variables
Default Input Method
Count
Default Target Method
None
Normalize Values
Yes
Interval Variables
Default Input Method
Median
Default Target Method
None
Default Constant Value
Default Character Value
Default Number Value
Method Options
Random Seed
12345
Tuning Parameters
Tree Imputation
Score
Hide Original Variables
Yes
Indicator Variables
Type
None
Source
Imputed Variables
Role
Rejected
Report
Validation and Test Data
No
Distribution of missing
No
Status
Default Input Method
Sample Explore Modify Model Assess Utility HPDM Applications Time Series
E-Commerce Customer Behaviour Analysis
Customer Information
Save Data
EM_SAVE_ TRAIN
StatExplore
Impute
Diagram
Log

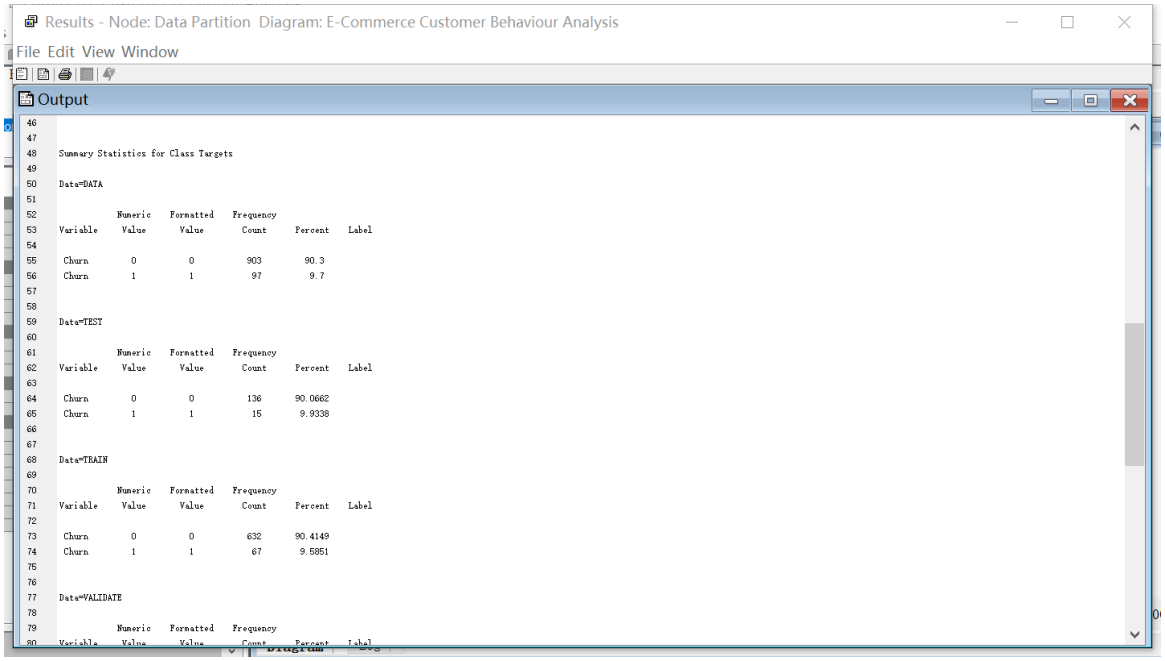

Results - Node: StatExplore (2) Diagram: E-Commerce Customer Behaviour Analysis
File Edit View Window
Chi-Square Plot
Chi-Square
Chi-Square
Location
MembershipLevel
Variable Worth
Worth
DaysSinceLastPurchase
IMP_TotalSpent
Age
Location
IMP_TotalPurchases
WebsiteVisits
MembershipLevel
Occupation
FavoriteCategory
Gender
Variable
Output
TRAIN Churn TARGET 0 903 90.3
TRAIN Churn TARGET 1 97 9.7
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 34.892 13.06373 1000 0 18 31 69 0.737158 -0.44511
DaysSinceLastPurchase INPUT 98.535 81.49238 1000 0 0 78 361 1.36131 1.522294
IMP_TotalPurchases INPUT 50.452 38.05172 1000 0 4 40 234 1.515316 2.551874
IMP_TotalSpent INPUT 9408.058 14269.39 1000 0 89.28 4038.71 109341.2 3.0519 11.45189
WebsiteVisits INPUT 13.753 7.931639 1000 0 1 13 30 0.466719 -0.81136
Class Variable Summary Statistics by Class Target
(maximum 500 observations printed)
Data Role=TRAIN Variable Name=FavoriteCategory
Target Target Level Number of Levels Missing Mode Mode Percentage Mode2 Mode2 Percentage

3.6 Data Partitioning

Now, let's proceed with data partitioning before the modeling phase. We'll split the dataset into training, validation, and test sets. These sets are used for training the model, tuning parameters, and evaluating the final model performance. The suggested partitioning is Training: 70.0%, Validation: 15%, Test: 15%, and the Partitioning Method should be set to Stratified.

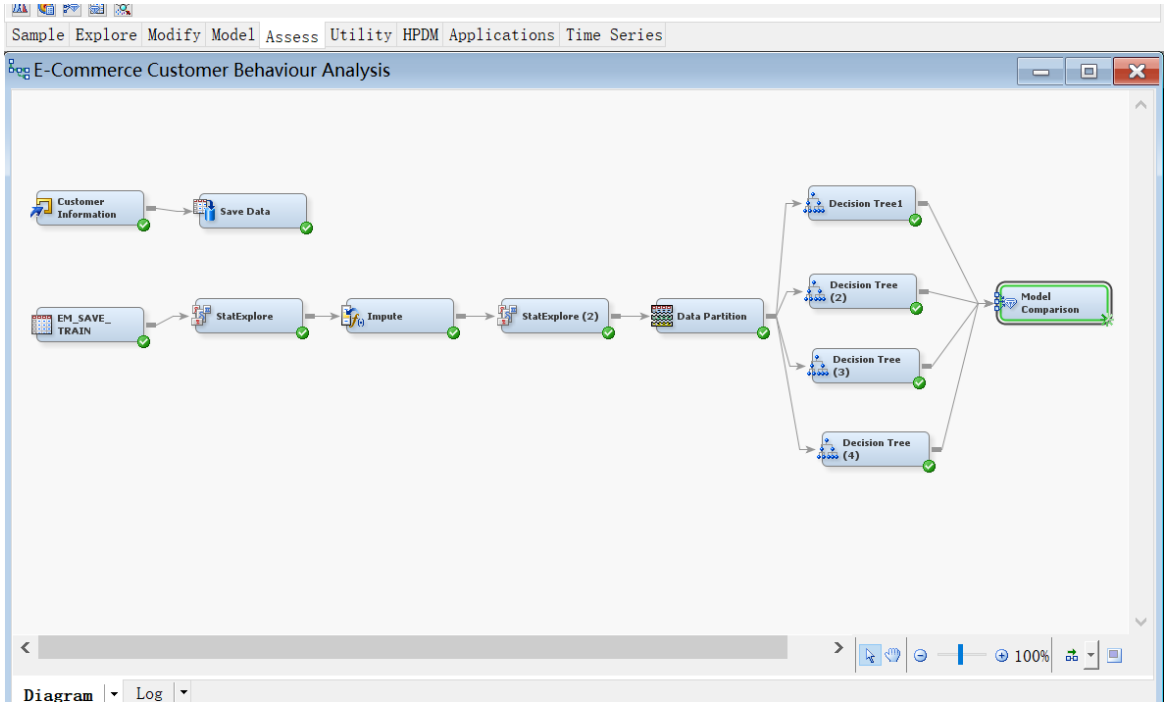
Reason: Considering the dataset size of only 1000 data points, if you need a larger training set to avoid overfitting, selecting 70% for training is a reasonable choice. It's also wise to use the "Default" partitioning method, especially when dealing with imbalanced class distribution in the target variable (Churn). This helps maintain a closer representation of the overall class proportions within each partition.

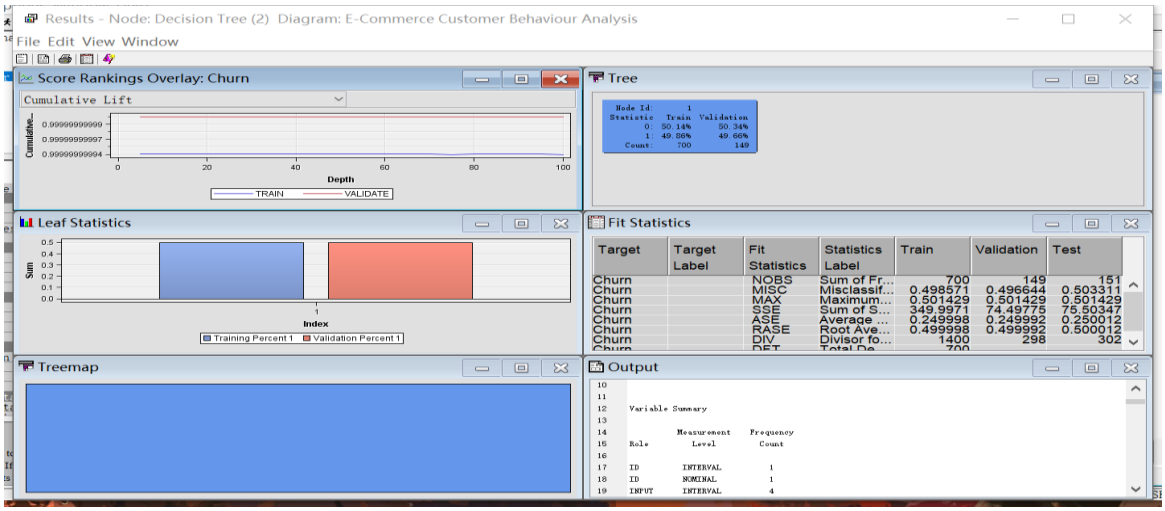


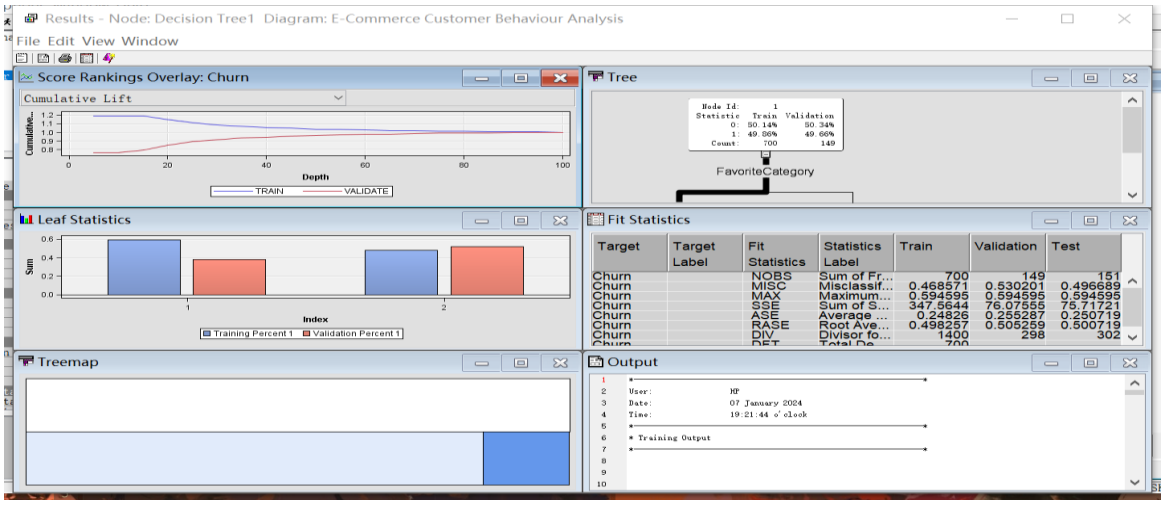


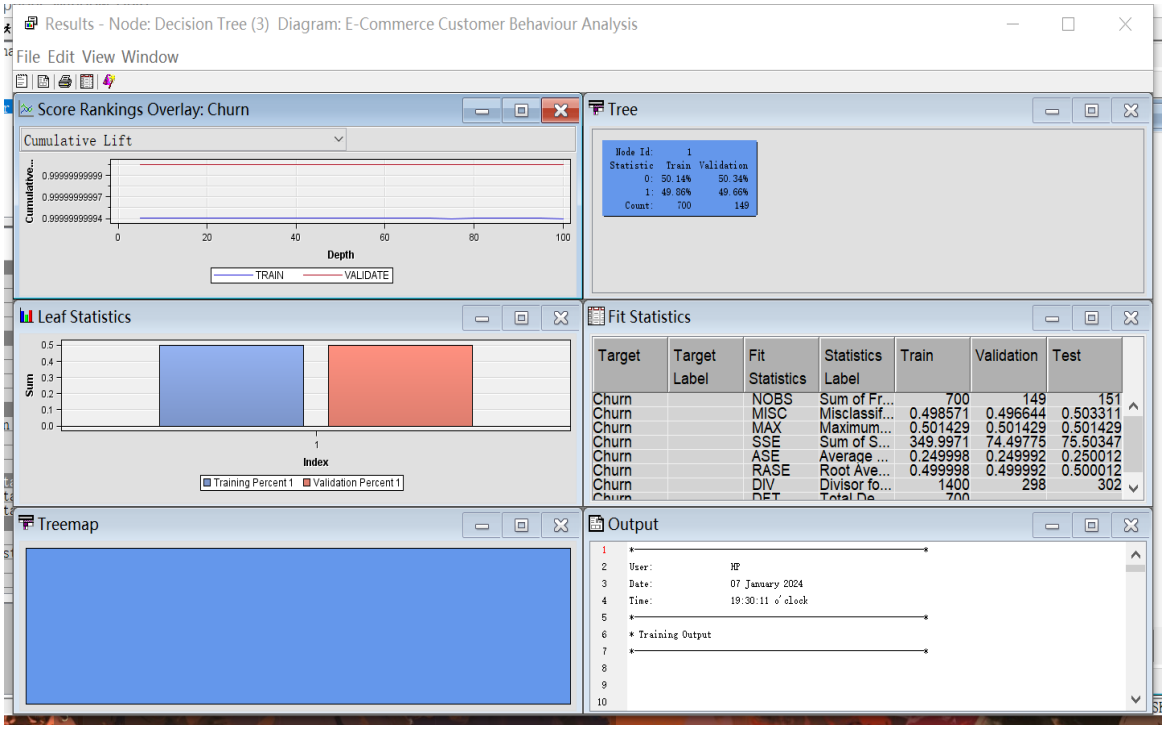
3.7 Create a decision tree model

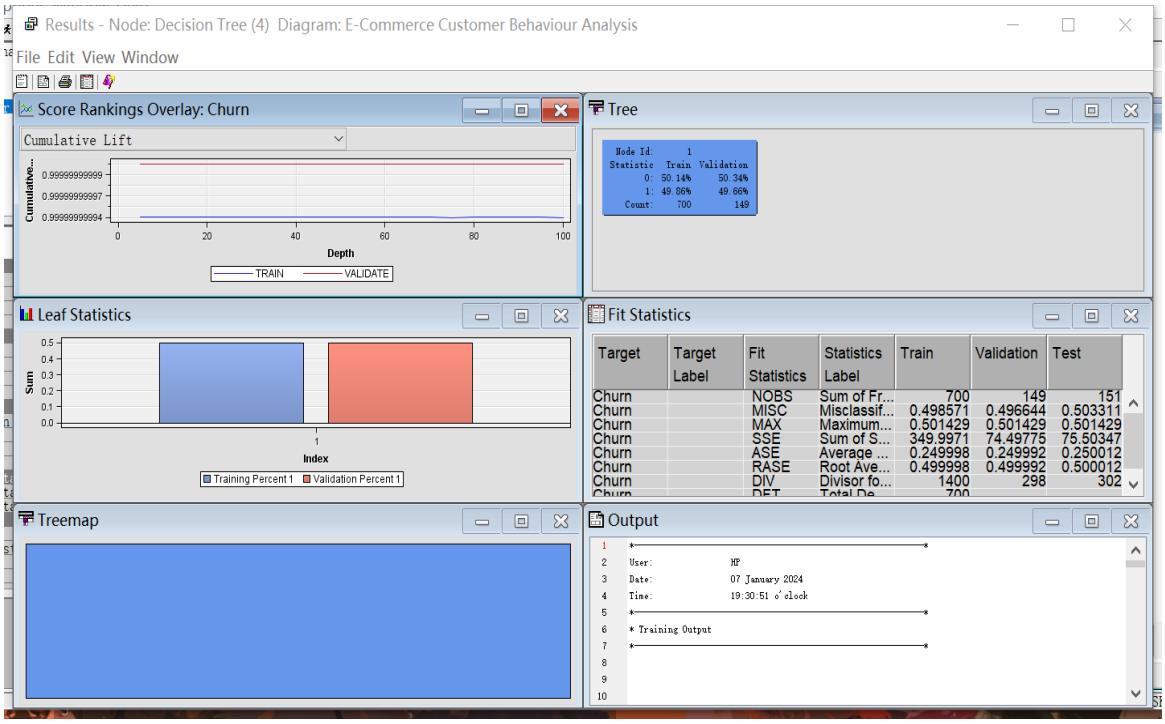
I choose different assessment measure and create four different tree

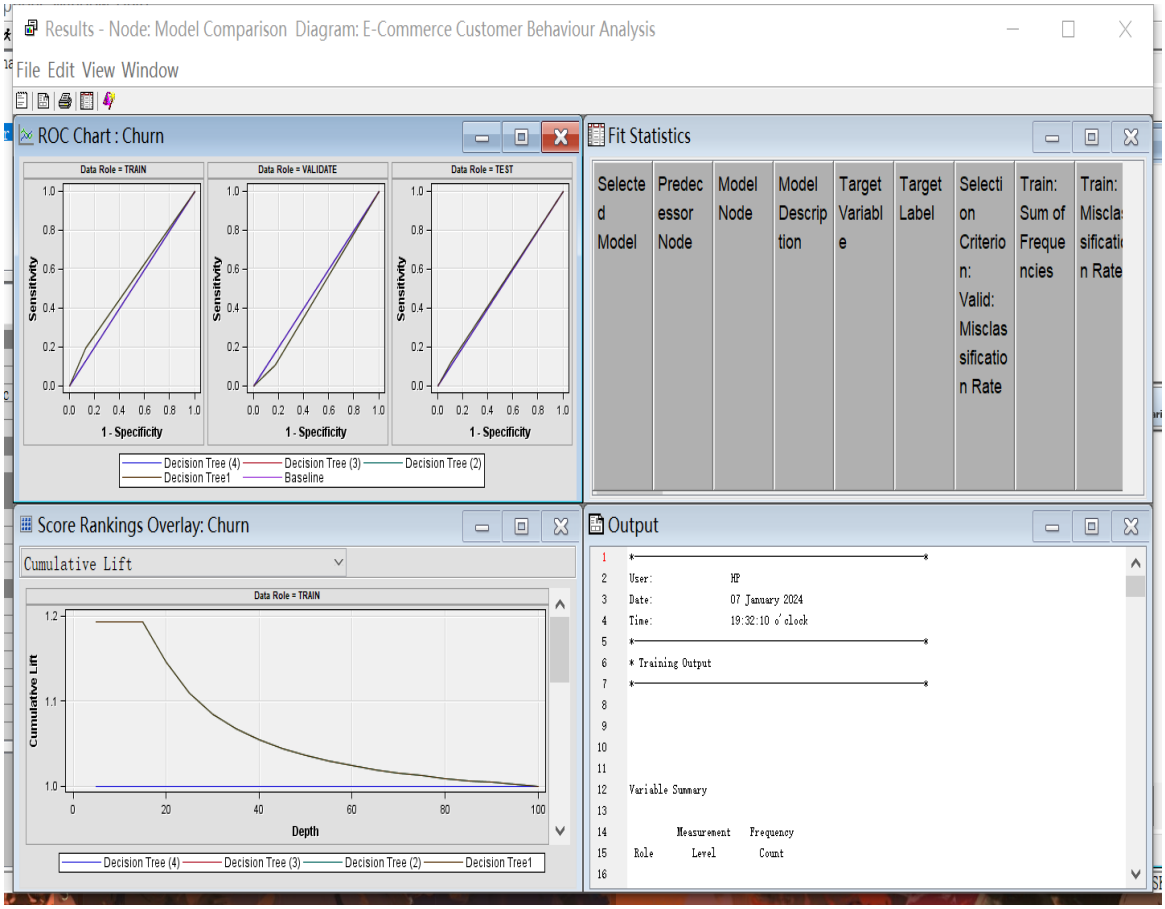







Results - Node: Decision Tree (3) Diagram: E-Commerce Customer Behaviour Analysis
File Edit View Window
Score Rankings Overlay: Churn
Cumulative Lift
Cumulative...
0.99999999999
0.99999999997
0.99999999994
0 20 40 60 80 100
Depth
TRAIN VALIDATE
Tree
Node Id: 1
Statistic Train Validation
0: 50.14% 50.34%
1: 49.86% 49.66%
Count: 700 149
Leaf Statistics
Sum
0.5 0.4 0.3 0.2 0.1 0.0
1
Index
Training Percent 1 Validation Percent 1
Fit Statistics
Target | Target Label | Fit Statistics | Statistics Label | Train | Validation | Test
Churn | | NOBS | Sum of Fr... | 700 | 149 | 151
Churn | | MISC | Misclassif... | 0.498571 | 0.496644 | 0.503311
Churn | | MAX | Maximum... | 0.501429 | 0.501429 | 0.501429
Churn | | SSE | Sum of S... | 349.9971 | 74.49775 | 75.50347
Churn | | ASE | Average ... | 0.249998 | 0.249992 | 0.250012
Churn | | RASE | Root Ave... | 0.499998 | 0.499992 | 0.500012
Churn | | DIV | Divisor fo... | 1400 | 298 | 302
Treemap
Output
* Training Output
User: HP
Date: 07 January 2024
Time: 19:30:11 o'clock


Results - Node: Decision Tree (4) Diagram: E-Commerce Customer Behaviour Analysis
File Edit View Window
Score Rankings Overlay: Churn
Cumulative Lift
Cumulative...
0.99999999999
0.99999999997
0.99999999994
0 20 40 60 80 100
Depth
TRAIN VALIDATE
Tree
Node Id: 1
Statistic Train Validation
0: 50.14% 50.34%
1: 49.86% 49.66%
Count: 700 149
Leaf Statistics
Sum
0.5 0.4 0.3 0.2 0.1 0.0
1
Index
Training Percent 1 Validation Percent 1
Fit Statistics
Target | Target Label | Fit Statistics | Statistics Label | Train | Validation | Test
Churn | | NOBS | Sum of Fr... | 700 | 149 | 151
Churn | | MISC | Misclassif... | 0.498571 | 0.496644 | 0.503311
Churn | | MAX | Maximum... | 0.501429 | 0.501429 | 0.501429
Churn | | SSE | Sum of S... | 349.9971 | 74.49775 | 75.50347
Churn | | ASE | Average ... | 0.249998 | 0.249992 | 0.250012
Churn | | RASE | Root Ave... | 0.499998 | 0.499992 | 0.500012
Churn | | DIV | Divisor fo... | 1400 | 298 | 302
Treemap
Output
* Training Output
User: HP
Date: 07 January 2024
Time: 19:30:51 o'clock



- **Model Selection:**

Model selection is based on the Validation Misclassification Rate (VMISC), showing the performance of four different decision tree models. Tree3 and Tree4 have the lowest misclassification rates, but according to the event classification table, these models seem to have not captured any positive events (Churn=1). This suggests that the models might simply be predicting all cases as the majority class (Churn=0).

- **Fit Statistics:**

Kolmogorov-Smirnov Statistic (KS Statistic): For Tree3 and Tree4, the KS statistics for the training, validation, and test sets are all 0, indicating that these models lack discriminative power.

Roc Index: The index is approximately 0.5 for all models, implying that the predictive ability of the models is not different from random guessing.

Cumulative Percent Captured Response: In Tree3 and Tree4, this metric is 0, suggesting that the models did not capture any response from positive events.

Event Classification Table: Tree3 and Tree4 did not correctly predict any cases with Churn=1 in both the training and validation sets, while Tree and Tree2, although capturing some Churn=1 cases, still exhibit overall poor performance.

- **Based on these outputs, here is an assessment of how the decision tree models impact customer behavior analysis:**

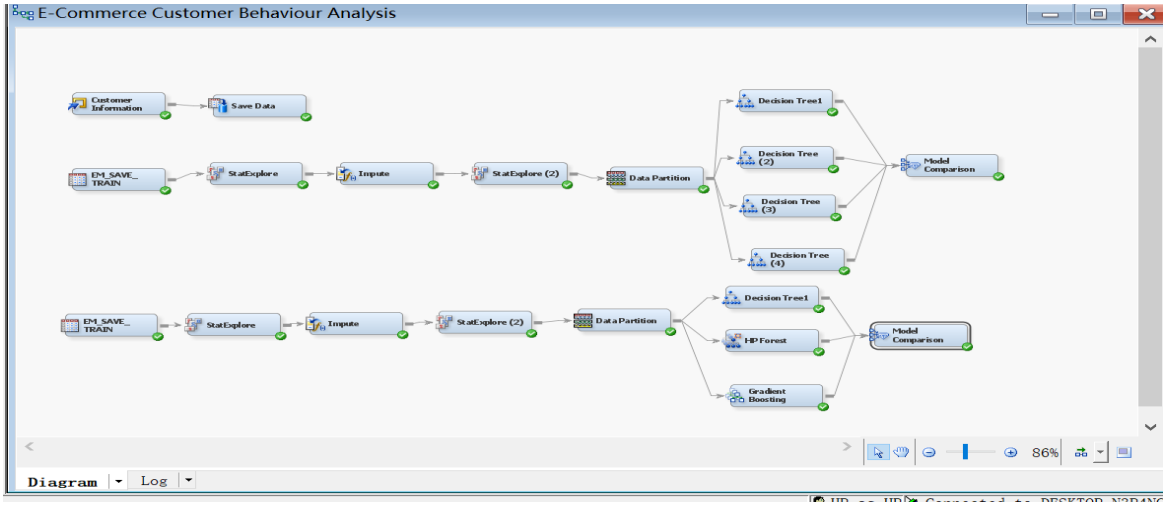
These models struggled with handling imbalanced datasets, as accuracy is not a good metric in such cases. Even if the models predict no churn for all customers (Churn=0), accuracy may appear high.

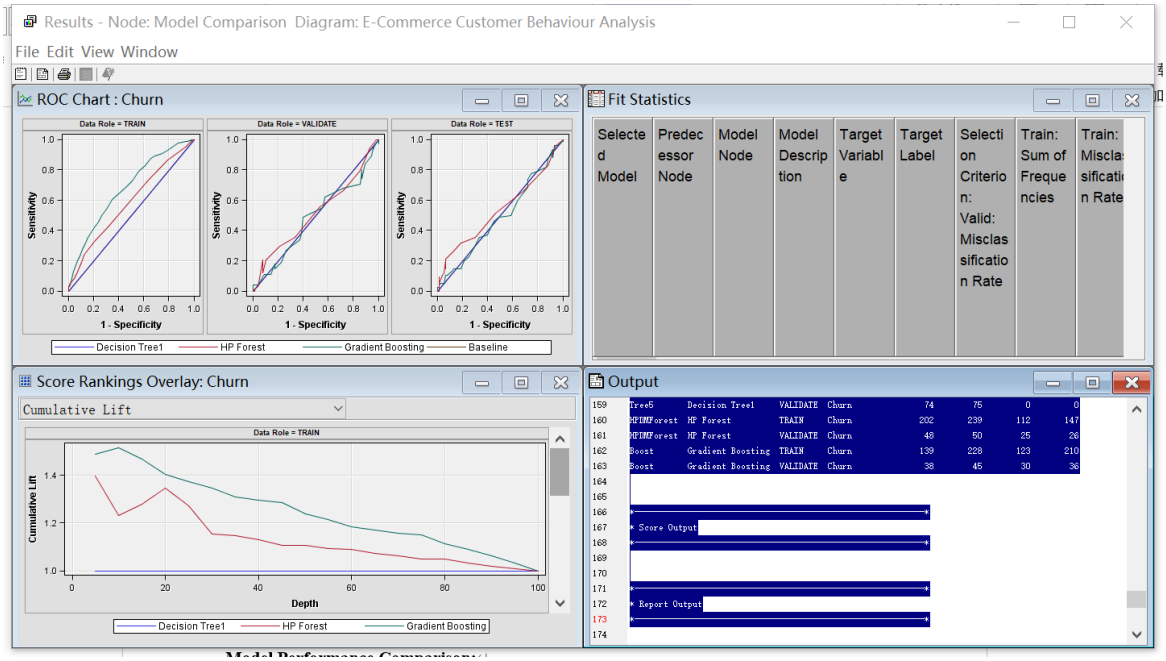
Tree3 and Tree4 may be too simplistic, lacking sufficient complexity to learn patterns in the data, and they might be disregarding the minority class (Churn=1).
While Tree and Tree2 attempted to distinguish between the two classes, they still need adjustments to improve their ability to identify the minority class.

- **Recommendations:**

To enhance predictive capabilities for customer churn behavior, consider employing resampling techniques to balance classes, adjusting model complexity, or trying different model parameters.
Further fine-tuning and optimization of these decision tree models are necessary. Explore different model structures, parameter settings, or try alternative model types such as random forests or gradient boosting models, which may exhibit better robustness on imbalanced datasets.

3.8 Apply Bagging and Boosting, using the Random Forest algorithm as a Bagging example.





We can observe the performance metrics of three models based on different decision tree

algorithms: Gradient Boosting (Boost), HP Forest (HPDMForest), and a single decision tree (Tree5). Here is an analysis of these results:

Model Performance Comparison:

- **Gradient Boosting (Boost):** It has the lowest misclassification rate on the validation set (0.37429), and its performance in terms of average square error (0.25801) and ROC index (0.479) is also good. This indicates that the Boost model has the overall best performance among the three models.
- **HP Forest (HPDMForest):** This model has a misclassification rate of 0.44857 on the validation set, with a slightly better average square error (0.24902) than Boost, but a lower ROC index (0.515) compared to the Boost model.
- **Decision Tree (Tree5):** It performs the worst among the three models, with a misclassification rate of 0.49857 on the validation set. Both the average square error (0.24999) and ROC index (0.500) are the lowest.

Classification Event Table:

- The Boost model correctly predicted 210 cases of Churn=1 in the training set and 36 cases in the validation set.
- The HPDMForest model correctly predicted 147 cases of Churn=1 in the training set and 26 cases in the validation set.
- The Tree5 model did not correctly predict any cases of Churn=1 in both the training and validation sets.

Customer Behavior Analysis:

- The Boost model seems to capture behavioral patterns that may lead to customer churn most effectively. Its predictive ability for Churn=1 is superior to the other two models, suggesting that the Boost model can better identify key factors that may lead to customer churn.
- Although the HPDMForest model has some capability in predicting Churn=1, its performance is not as good as the Boost model. This could be because the model did not effectively capture all relevant behavioral patterns.
- The performance metrics of the Tree5 model suggest that it did not capture any patterns related to customer churn, making it potentially less useful for analyzing customer behavior.

Conclusion:

In summary, the Boost model performs the best among these three models, indicating that it might be the optimal choice for analyzing and predicting customer behavior. However, it's essential to note that despite the good performance on the validation set, the ROC index is still close to 0.5, suggesting room for improvement in predictive capability. Further exploration may be needed, such as adjusting model parameters, trying other advanced algorithms, and considering strategies to balance the dataset to enhance the model's predictive ability for the minority class.