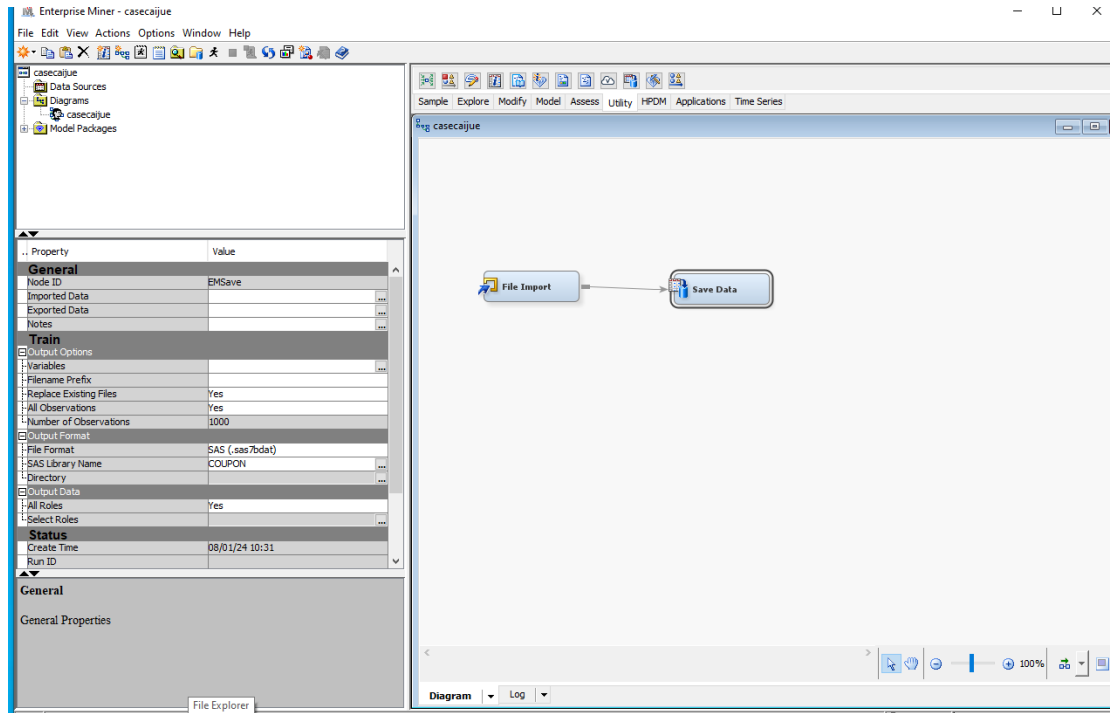


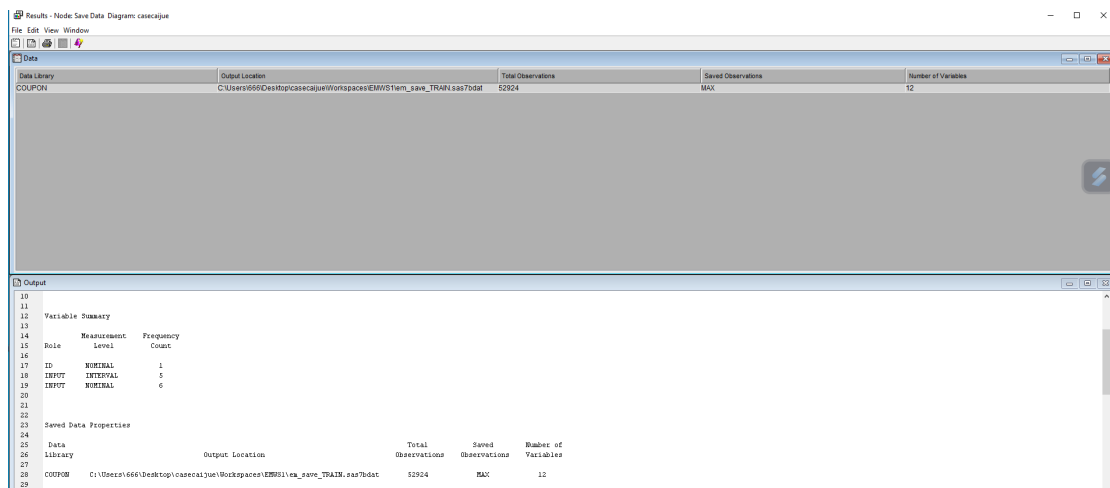
# Exploration in SAS

22078878 CAI JUE

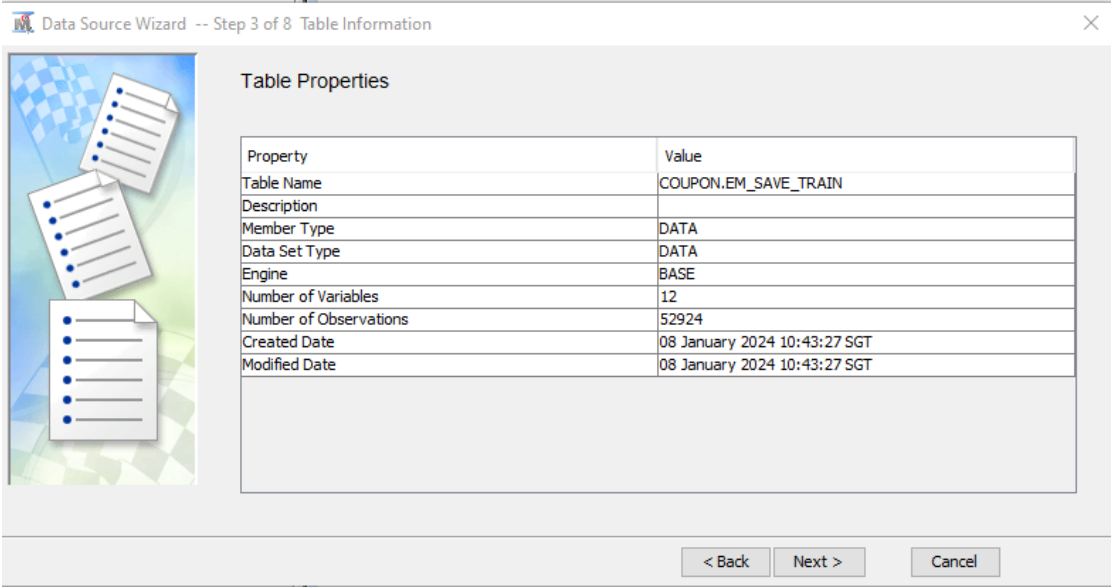
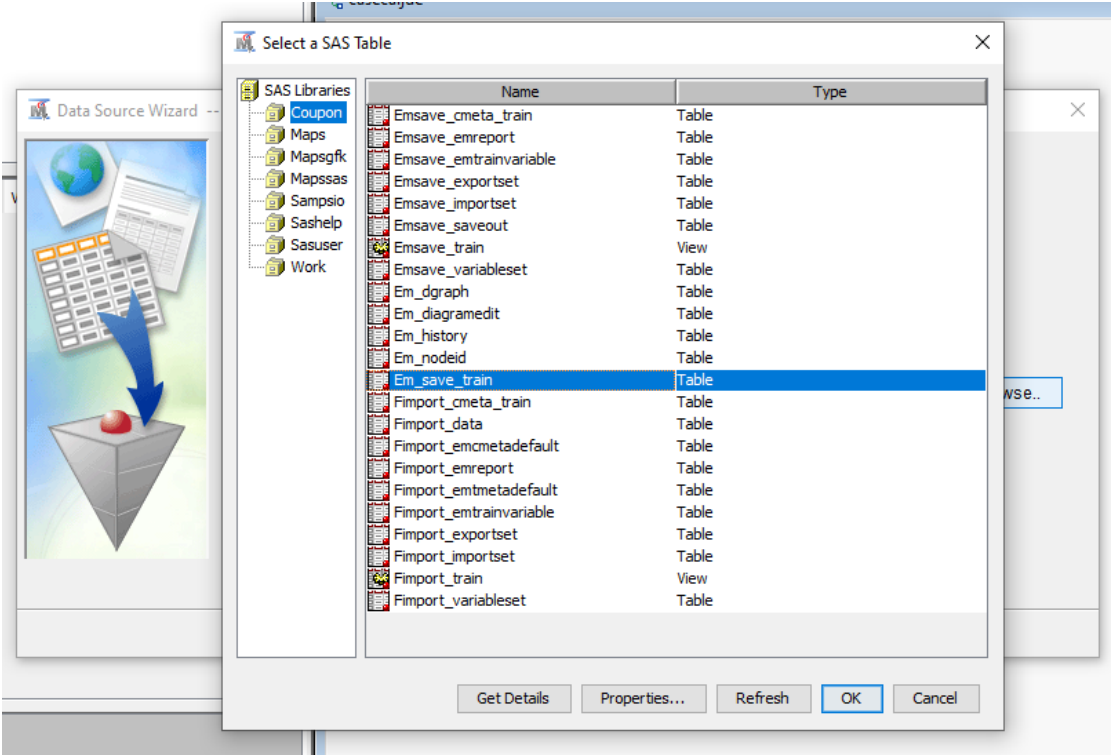
## 1.Data imported and saved under COUPON's library



## 2.Check the name of the save and open this, em\_save\_TRAIN.sas7bat, for the next step of exploration.



3. Select this file to explore



## Basic Version:

Data Source Wizard -- Step 5 of 8 Column Metadata

(none) ☐ not Equal to ☐ Apply Reset

Columns: ☐ Label ☐ Mining ☐ Basic ☐ Statistics

Name	Role	Level	Report	Order	Drop	Lower Limit	Upper Limit
Avg_Price	Input	Interval	No		No	.	
Coupon_Status	Input	Nominal	No		No	.	
CustomerID	Input	Interval	No		No	.	
Delivery_Charge	Input	Interval	No		No	.	
Gender	Input	Nominal	No		No	.	
Location	Input	Nominal	No		No	.	
Product_Category	Input	Nominal	No		No	.	
Product_SKU	Input	Nominal	No		No	.	
Quantity	Input	Interval	No		No	.	
Tenure_Months	Input	Interval	No		No	.	
Transaction_Date	Input	Nominal	No		No	.	
Transaction_ID	ID	Nominal	No		No	.	

Show code Explore Compute Summary < Back Next > Cancel

## Advanced Version:

Data Source Wizard -- Step 5 of 8 Column Metadata

(none) ☐ not Equal to ☐ Apply Reset

Columns: ☐ Label ☐ Mining ☐ Basic ☐ Statistics

Name	Role	Level	Report	Order	Drop	Lower Limit	Upper Limit
Avg_Price	Input	Interval	No		No	.	
Coupon_Status	Input	Binary	No		No	.	
CustomerID	Input	Interval	No		No	.	
Delivery_Charges	Input	Interval	No		No	.	
Gender	Input	Binary	No		No	.	
Location	Input	Nominal	No		No	.	
Product_Category	Input	Nominal	No		No	.	
Product_SKU	Rejected	Nominal	No		No	.	
Quantity	Input	Interval	No		No	.	
Tenure_Months	Input	Interval	No		No	.	
Transaction_Date	Rejected	Nominal	No		No	.	
Transaction_ID	ID	Interval	No		No	.	

Show code Explore Refresh Summary < Back Next > Cancel

## After manual adjustment:

Data Source Wizard -- Step 5 of 8 Column Metadata

(none) ☐ not Equal to ☐ Apply Reset

Columns: ☐ Label ☐ Mining ☐ Basic ☐ Statistics

Name	Role	Level	Report	Order	Drop	Lower Limit	Upper Limit
Avg_Price	Input	Interval	No		No	.	
Coupon_Status	Target	Binary	No		No	.	
CustomerID	ID	Nominal	No		No	.	
Delivery_Charges	Input	Interval	No		No	.	
Gender	Input	Binary	No		No	.	
Location	Input	Nominal	No		No	.	
Product_Category	Input	Nominal	No		No	.	
Product_SKU	Rejected	Nominal	No		No	.	
Quantity	Input	Interval	No		No	.	
Tenure_Months	Input	Interval	No		No	.	
Transaction_Date	Time ID	Nominal	No		No	.	
Transaction_ID	ID	Nominal	No		No	.	

Show code Explore Refresh Summary < Back Next > Cancel

## Renamed Coupon\_decision for better readability.

Data Source Wizard -- Step 8 of 9 Data Source Attributes

You may change the name and the role, and can specify a population segment identifier for the data source to be created.

Name :

Role :

Segment :

Notes :

< Back   Next >   Cancel

## Final presentation

Data Source Wizard -- Step 9 of 9 Summary

Metadata Completed.

Library: COUPON  
Data Source: Coupon\_decision  
Role: Raw

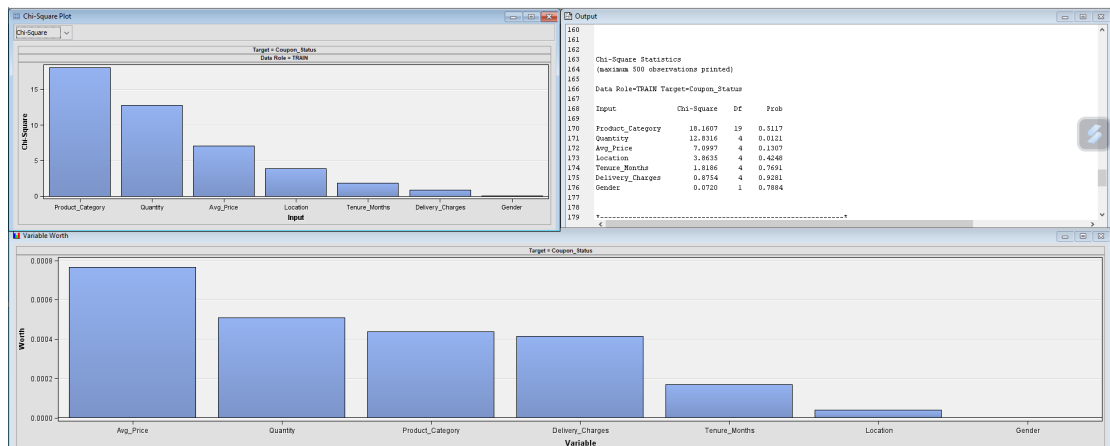
Role	Level	Count
ID	Nominal	2
Input	Binary	1
Input	Interval	4
Input	Nominal	2
Rejected	Nominal	1
Target	Binary	1
Time ID	Nominal	1

< Back   Finish   Cancel

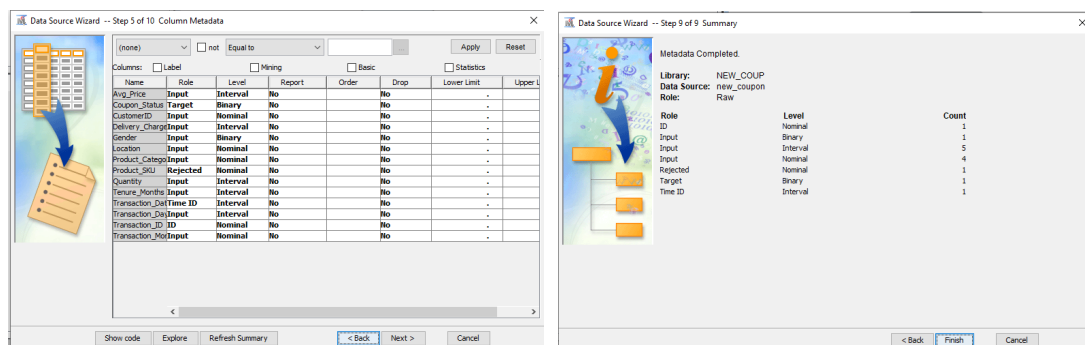
## 4.使用 StatExplore



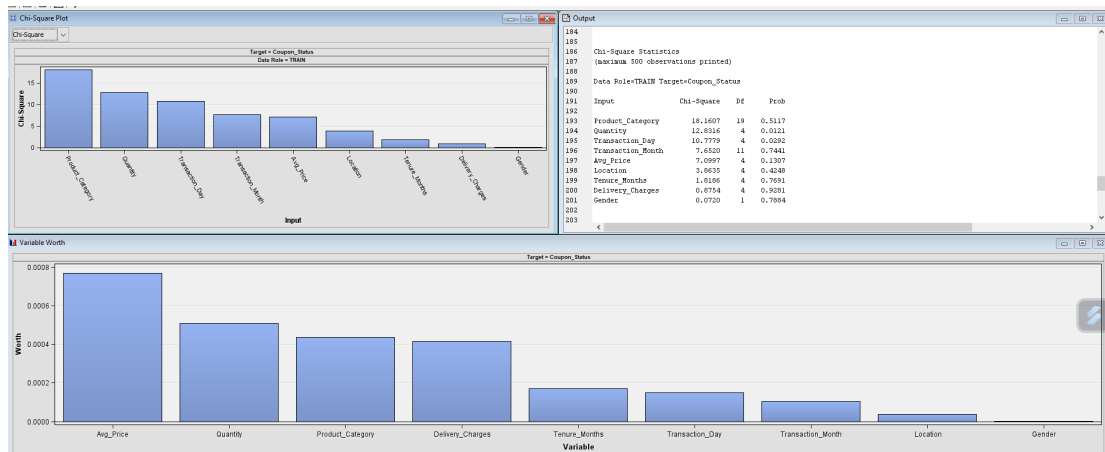
Looking at the relationship between INPUT and TARGET, since the cardinality shows no obvious relationship and only QUANTITY presents a certain relationship, I considered the possibility that the month of purchase and the day of the month might also have an effect, so I felt that from creating a new SOURCE for the comparison, splitting the date of the DATA in it into month and day, made it possible to explore more relationships. Because the current simple exploration of the relationships presented is not promising.



Split in python and form new dataset and new source “new\_coupon” in SAS.



As you can see my conjecture is valid and a new significance variable appears which is the trading day, but the model construction is complex so I will look at the results of the two performed decision trees.



**5. Running the first DECISION TREE reveals that 1 is not OUTCOME (Explore solutions)**

Classification Table

Data Role=TRAIN Target Variable=Coupon\_Status Target Label=' '

Target	Outcome	Target Percentage	Outcome Percentage	Frequency Count	Total Percentage
0	0	66.1722	100	14008	66.1722
1	0	33.8278	100	7161	33.8278

Data Role=VALIDATE Target Variable=Coupon\_Status Target Label=' '

Target	Outcome	Target Percentage	Outcome Percentage	Frequency Count	Total Percentage
0	0	66.1691	100	10505	66.1691
1	0	33.8309	100	5371	33.8309

**Problem1-solution based on dataset:** Adjusting the data structure to continue the exploration, I found a complication in my dataset is that “CustomerID” and “TransactionID” are duplicates, so I need to create a key combination TransactionDd+Produc\_SKU to uniquely identify each of the Combined\_IDS. for each piece of data, which can then be used for better modeling. Then delete transactionid+Produc\_SKU. Done in python.

```
data['Combined_Key'] = data['Transaction_ID'].astype(str) + "-" + data['Product_SKU']
```

```
data.to_csv('datasetfinal_new2.csv', index=False)
```

```
data = data.drop(['Transaction_ID', 'Product_SKU'], axis=1)
```

The new data variables are as follows:

Data Source Wizard -- Step 5 of 8 Column Metadata

(none) ☐ not Equal to ☐ Apply ☐ Reset

Columns: ☐ Label ☐ Mining ☐ Basic ☐ Statistics

Name	Role	Level	Report	Order	Drop	Lower Limit	Upper Limit
Avg_Price	Input	Interval	No		No	-	-
Combined_Key	ID	Nominal	No		No	-	-
Coupon_Status	Target	Binary	No		No	-	-
CustomerID	Input	Interval	No		No	-	-
Delivery_Charges	Input	Interval	No		No	-	-
Gender	Input	Binary	No		No	-	-
Location	Input	Nominal	No		No	-	-
Product_Category	Input	Nominal	No		No	-	-
Quantity	Input	Interval	No		No	-	-
Tenure_Months	Input	Interval	No		No	-	-
Transaction_Date	Time ID	Nominal	No		No	-	-
Transaction_Day	Input	Nominal	No		No	-	-
Transaction_Month	Input	Nominal	No		No	-	-

Show code Explore Refresh Summary < Back Next > Cancel

Data Source Wizard -- Step 9 of 9 Summary

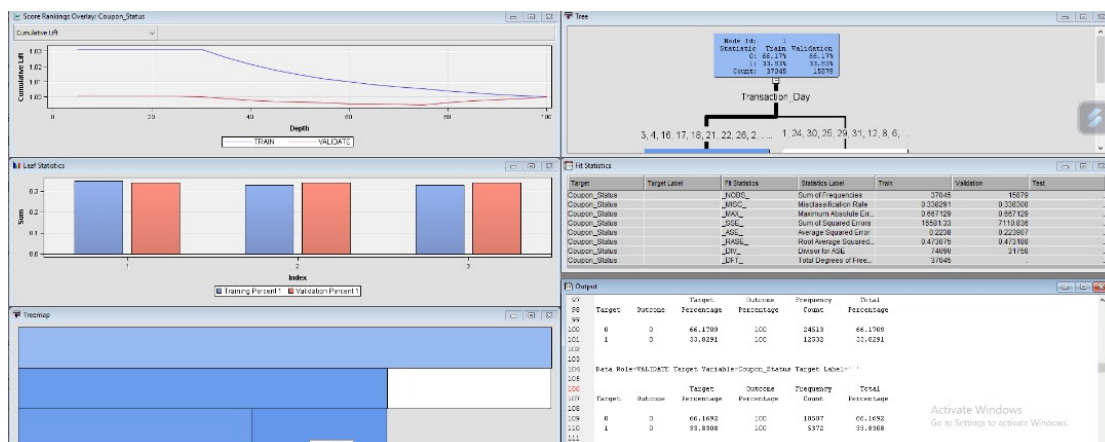
Metadata Completed.

Library: NEW\_COUP  
Data Source: new2\_coupon  
Role: Raw

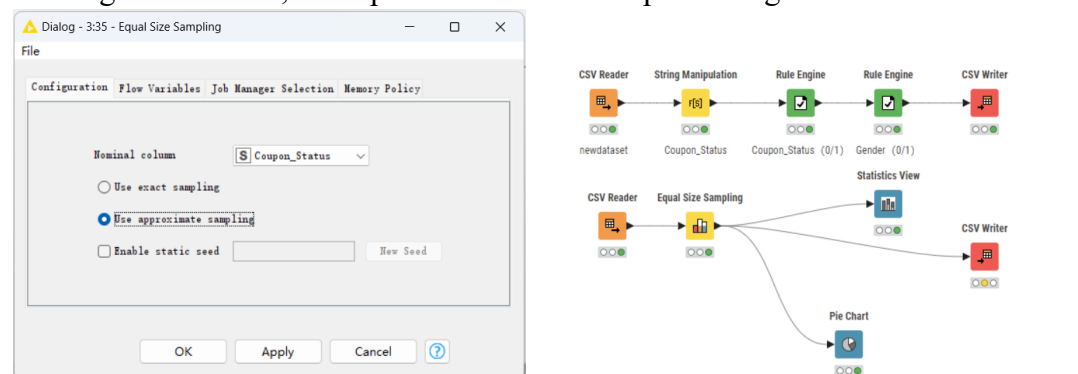
Role	Level	Count
ID	Nominal	1
Input	Binary	1
Input	Interval	5
Input	Nominal	4
Target	Binary	1
Time ID	Nominal	1

< Back Finish Cancel

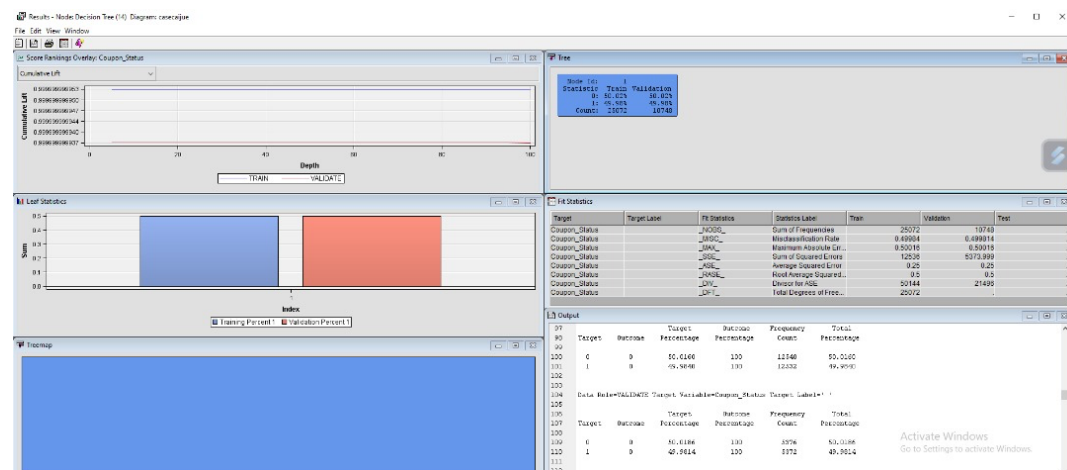
Running the decision tree, the problem is still not solved.



**Problem2-solution based on dataset:** Checked the 0 and 1 of the coupon status and found that as a TARGET it's 0 and 1 are not balanced, maybe that's why the 0 can't be read and explored, so I solved the data imbalance by exploring it. Since I am not skilled in using sas software, I completed the imbalance processing of the data in KINME.



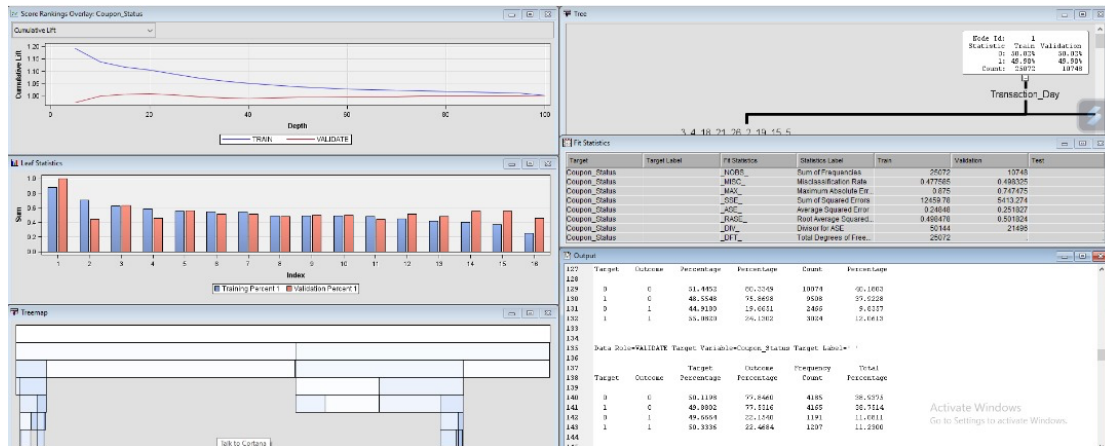
After exporting the new dataset BALANCE and repeating the above steps, the rendering is still poor.





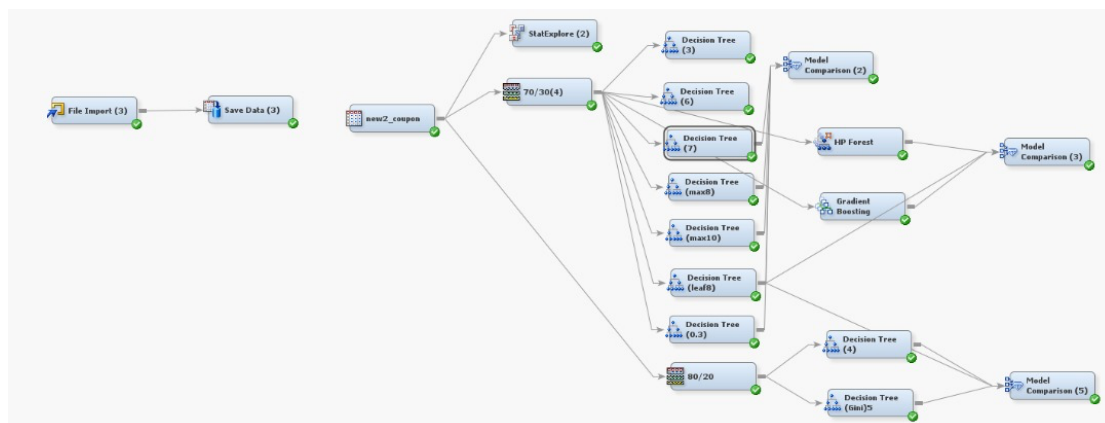
### Solution3--solution based on decision tree setting. - Resolved

Experimenting with the third possibility, the data was too noisy for another segmentation method, using gini, it was found that a certain amount of decision making occurred although the values were still poor, but it was decided to adjust the decision tree parameters to continue to improve.



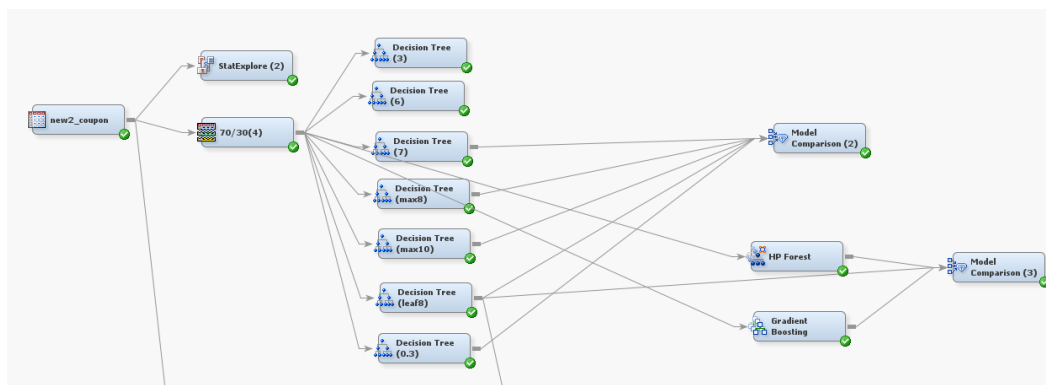
## 6. Explore decision tree results on both datasets.

### 6.1 new2\_coupon



#### ● Data set split 70/30

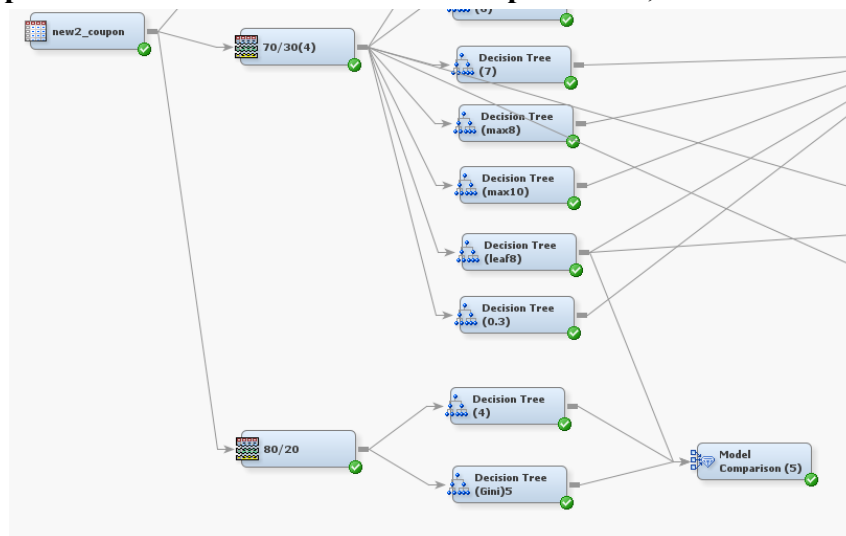
With a 70/30 score dataset, adjusted for multiple branchleaf values and depth values, Gini leaf8 length10 performed the best.



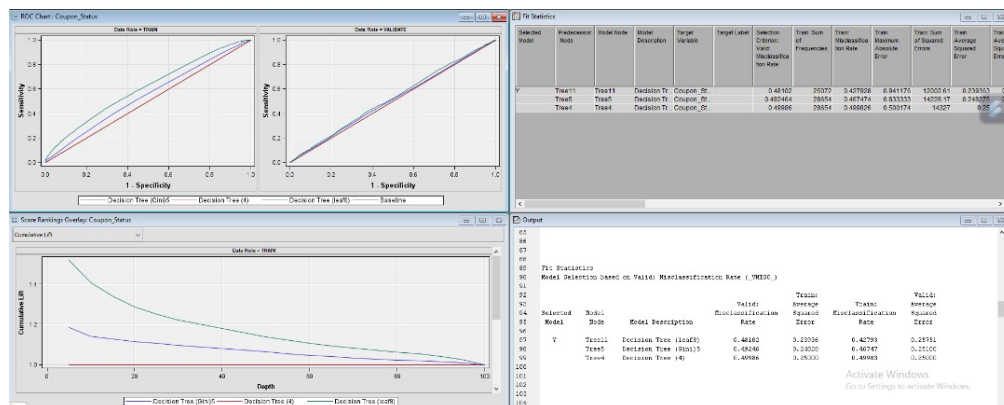
## Model Comparison (2) Outcome.

Selected Model	Model Node	Model Description	Valid: Misclassification Rate	Train: Average Squared Error	Train: Misclassification Rate	Valid: Average Squared Error
Y	Tree11	Decision Tree (max10)	0.48102	0.23936	0.42793	0.25751
	Tree9	Decision Tree (max10)	0.48214	0.23923	0.43327	0.25687
	Tree10	Decision Tree (max8)	0.49088	0.24463	0.45333	0.25386
	Tree7	Decision Tree (7)	0.49833	0.24848	0.47758	0.25183

- Compare the difference between dataset splits 70/30, 80/20

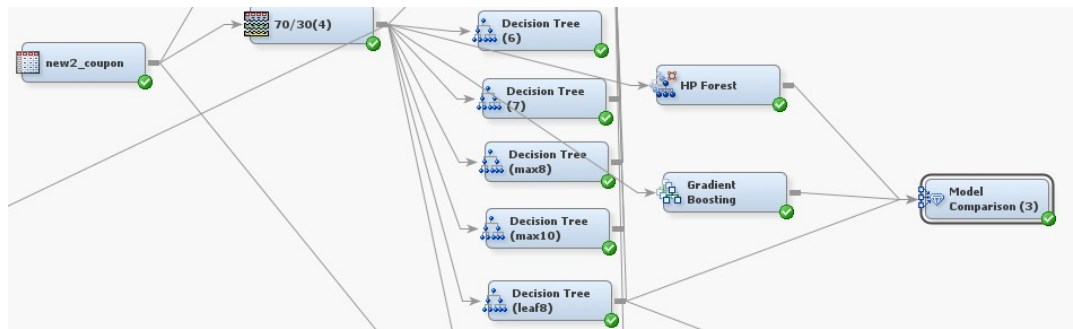


## Model Comparison (5) Outcome.



It was found that splitting the size of the other dataset does not drastically affect the accuracy of the model, so it is possible to use 70/30 to continue exploring, since splitting does not make sense.

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



## Model Comparison (3) Outcome.

### Fit Statistics

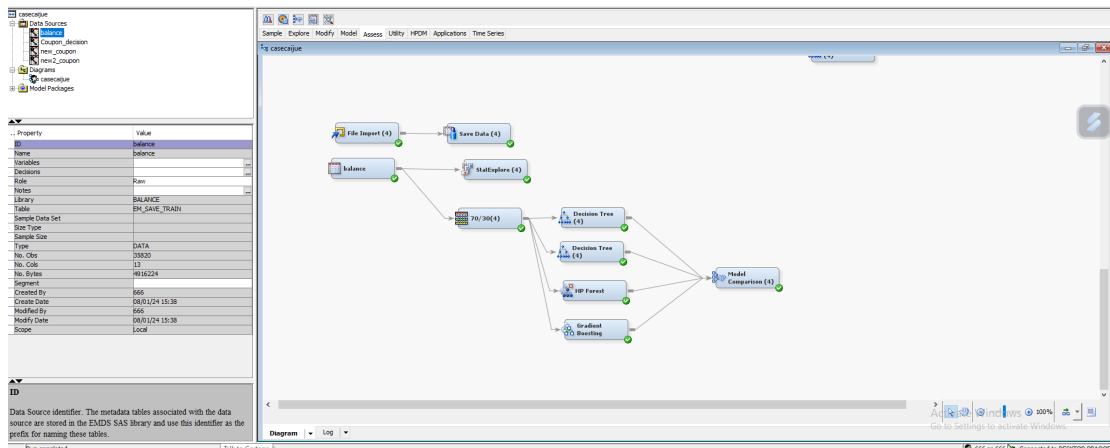
Model Selection based on Valid: Misclassification Rate (\_VMISC\_)

Selected Model	Model Node	Model Description	Valid: Misclassification Rate	Train: Average Squared Error	Train: Misclassification Rate	Valid: Average Squared Error
Y	Tree11	Decision Tree (leaf8)	0.48102	0.23936	0.42793	0.25751
	Boost	Gradient Boosting	0.50502	0.24981	0.48413	0.25012
	HPDMForest	HP Forest	0.50735	0.24970	0.48062	0.25025

Activate Windows  
Go to Settings to activate Windows.

Adding bagging and boost tends to make, accuracy improve, but sadly it didn't in my data project. Suggesting that I need to start this whole project with a review of the meaning of the dataset I'm exploring and whether the categorization really makes sense.

## 6.2 balance



### 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	Tree13	Decision Tree (4)	0.48809	0.23641	0.41644	0.26199
	Boost2	Gradient Boosting	0.49070	0.24759	0.44241	0.25009
	Tree12	Decision Tree (4)	0.49116	0.23888	0.42266	0.25973
	HPDMForest2	HP Forest	0.51107	0.24967	0.47818	0.25025

The data after balancing with KNIME is also still not good, there are many reasons for this, it could be that the specific technique used for balancing doesn't work and exploring this I think I need to go back to the beginning and reexamine all my work.

### General overview of the process:

