

Background

In this assessment, I adapt a dataset named "E-Commerce Customer." The assessment will apply three tools: Talend Preparation, Talend Data Integration, and SAS Enterprise Miner. These three tools will be used to carry out tasks such as data import and preprocessing, conduct decision tree analysis, and apply various ensemble methods.

Dataset description

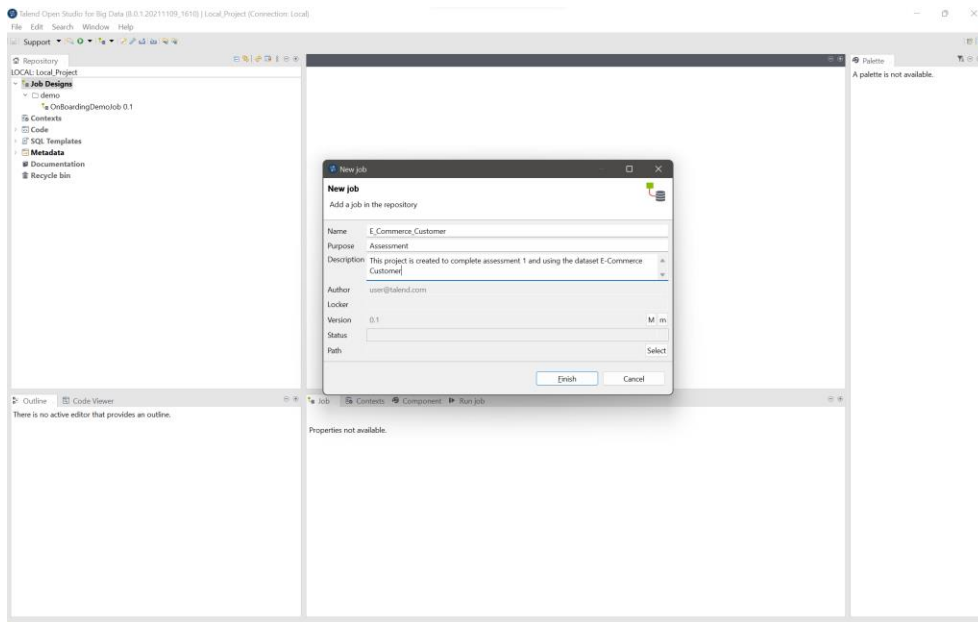
The dataset "E-Commerce Customer" records customer transactions on an e-commerce platform, including various customer attributes and purchase history in 2023. It has 12 attributes that include Customer ID, Age, Gender, Location, Membership Level, Total Purchases, Total Spent, Favorite Category, Last Purchase Date, Occupation, Website Visits Frequency, and Churn. The dataset comprises 570 rows.

Variable	Data Type	Description
Customer ID	Numeric	Unique identifier for each customer.
Age	Numeric	Age of the customer.
Gender	String	Gender of the customer.
Location	String	State of the customer base in Malaysia
Membership Level	String	Membership level label in Bronze, Silver, Gold, Platinum.
Total Purchases	Numeric	Total number of purchases made by the customer in a year.
Total Spent	Numeric	Total amount spent by the customer in a year.
Favorite Category	String	The category in which the customer most frequently shops labelled in Electronics, Clothing, Home Goods.
Last Purchase Date	Date	The date of the last purchase.
Occupation	String	Customer's occupation.
Website Visits Frequency	String	Frequency of customer visits the website.
Churn	Numeric	Indicates whether the customer has stopped purchasing (1 for churned, 0 for active).

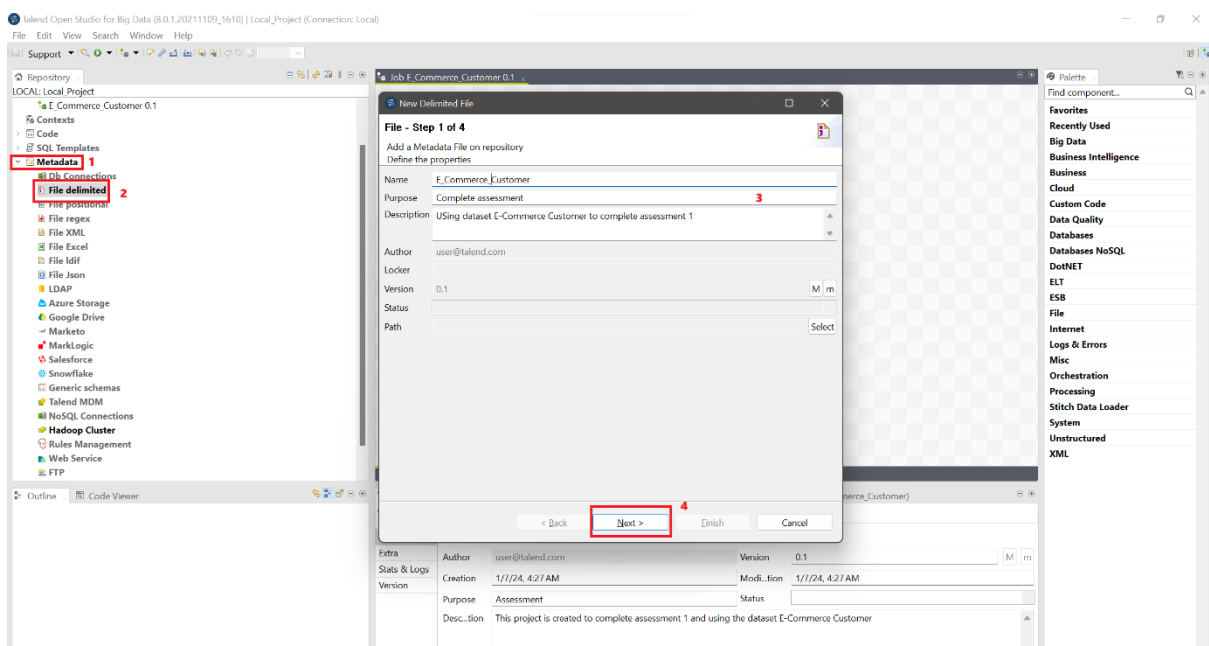
Steps

The assessment start with the help of Talend data Integration for data preprocessing.

- a. To start up, I created a project in Talend data integration.



- b. To extract data from CSV file, I went to Metadata, then right click File delimited. It will pop out a new window for me to fill up the 'Name', 'Purpose' and 'Description'. After that click 'Next'.



- c. I browse for my data set. The File viewer give me a preview of my import data. Then click 'Next'

New Delimited File

File - Step 2 of 4

Add a Metadata File on repository
Define the path of the file and the format settings

File Settings

Server: Localhost 127.0.0.1

File: C:/Users/User/Downloads/E-Commerce Customer.csv Browse...

Format: UNIX 1

File Viewer

```
CustomerID, Age, Gender, Location, MembershipLevel, TotalPurchases, TotalSpent, FavoriteCategory, LastPurchaseDate, Occupation, WebsiteVisits
1, 30, Male, Selangor, Gold, 10, 500, Electronics, 12/1/2023, Engineer, High, 0
2, 25, Female, Kuala Lumpur, Silver, 5, 50, Clothing, 11/15/2023, Student, Medium, 1
3, 40, Male, Penang, Bronze, 15, 800, HomeGoods, 12/20/2023, Doctor, Low, 0
4, 35, Female, Johor, Platinum, 20, 1200, Electronics, 12/10/2023, Lawyer, High, 0
5, 28, Male, Melaka, Silver, 8, 400, Clothing, 12/5/2023, Teacher, Medium, 1
6, 45, Female, Negeri Sembilan, Gold, 12, 700, HomeGoods, 11/30/2023, Artist, High, 0
7, 32, Male, Terengganu, Bronze, 18, 950, Electronics, 12/25/2023, IT Analyst, Medium, 0
8, 27, Female, Pahang, Silver, 7, 350, Clothing, 11/28/2023, Nurse, Low, 1
```

2

3 Next > Finish Cancel

- d. In next page, I change the 'Field Separator' to 'comma', tick 'header' then insert 1, and tick 'set heading row as column names'. Lastly, click 'Refresh Preview' to see the preview of my data. Once confirm I click 'Next'.

New Delimited File

File - Step 3 of 4

Add a Metadata File on repository
Define the setting of the parse job

File Settings

Encoding: US-ASCII 1

Field Separator: Comma 2 Corresponding Character: ""

Row Separator: Standard EOL Corresponding Character: "\n"

Escape Char Settings

☐ CSV ☒ Delimited

Escape Char: Empty

Text Enclosure: Empty

☐ Split row before field

Rows To Skip

If any rows must be ignored, specify the following parameters

Header: 1 3

Footer: ☐

☐ Skip empty row

Limit Of Rows

If the number of lines must be limited, specify this number

Limit:

Preview Output

☒ Set heading row as column names 4 Refresh Preview

CustomerID	Age	Gender	Location	MembershipLevel	TotalPurchases	TotalSpent	FavoriteCategory	LastPurchaseDate	Occupation	WebsiteVisits
1	30	Male	Selangor	Gold	10	500	Electronics	12/1/2023	Engineer	High
2	25	Female	Kuala Lumpur	Silver	5	50	Clothing	11/15/2023	Student	Medium
3	40	Male	Penang	Bronze	15	800	HomeGoods	12/20/2023	Doctor	Low
4	35	Female	Johor	Platinum	20	1200	Electronics	12/10/2023	Lawyer	High
5	28	Male	Melaka	Silver	8	400	Clothing	12/5/2023	Teacher	Medium
6	45	Female	Negeri Sembilan	Gold	12	700	HomeGoods	11/30/2023	Artist	High
7	32	Male	Terengganu	Bronze	18	950	Electronics	12/25/2023	IT Analyst	Medium
8	27	Female	Pahang	Silver	7	350	Clothing	11/28/2023	Nurse	Low

5 Next > Finish Cancel

- e. In the next page I change the data type for attribute LastPurchaseDate to date and adjust the pattern. Once done I proceed click finish.

New Delimited File

File - Step 4 of 4

Add a Schema on repository

Define the Schema

Name

Comment

Schema

Click to update schema preview

Guess

Description of the Schema

Column	K...	Type	<input checked="" type="checkbox"/> N.	Date Pattern (...)	Length	Precision	Defa...	Comm...
FavoriteCategory	<input type="checkbox"/>	String	<input checked="" type="checkbox"/>		11	0		
LastPurchaseDate	<input type="checkbox"/>	Date	<input checked="" type="checkbox"/>	"MM-dd-yyyy"		0		
Occupation	<input type="checkbox"/>	String	<input checked="" type="checkbox"/>		17	0		
WebsiteVisitsFreque...	<input type="checkbox"/>	String	<input checked="" type="checkbox"/>		6	0		
Churn	<input type="checkbox"/>	Integer	<input checked="" type="checkbox"/>		1	0		

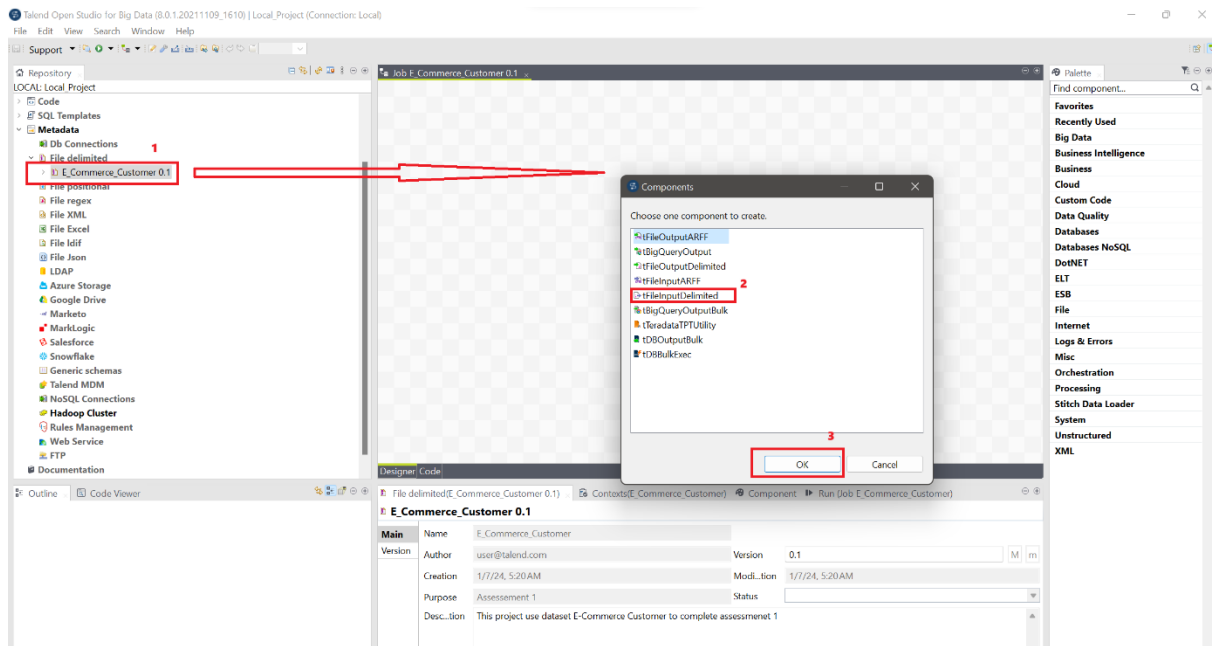
< Back

Next >

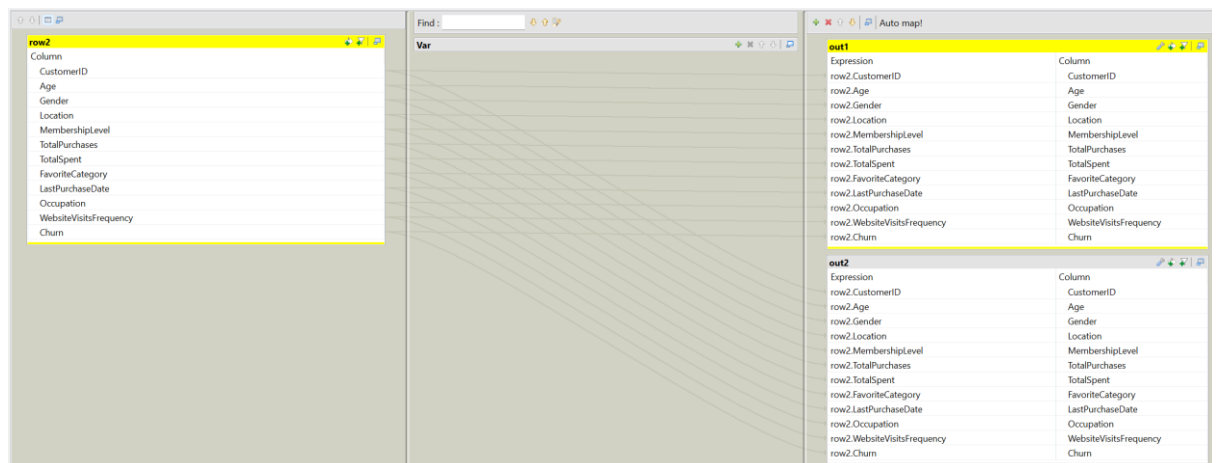
Finish

Cancel

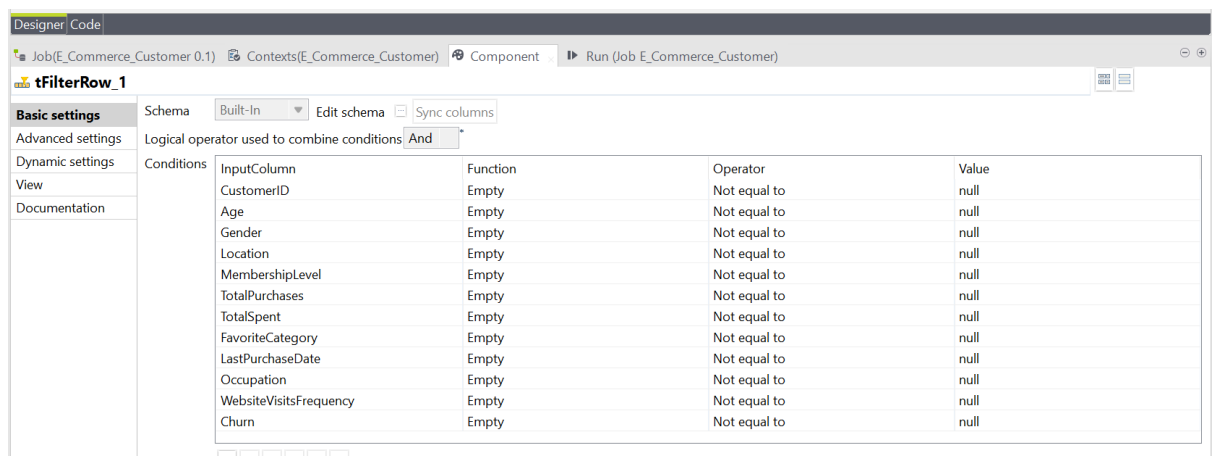
- f. Drag the delimited file to the canvas, a window pop out, I select 'tFileInputDelimited' and click OK.



- g. Then I add on component tLogRow to extract data and link it to tMap. In tmap I generate two output section for filter data in two scenario.



- h. In filter 1 I set filter for each attribute for getting dataset with no missing value.



- i. In filter 2 I set filter for each attribute for getting dataset with missing value.

Designer | Code

Job(E_Commerce_Customer 0.1) Contexts(E_Commerce_Customer) Component Run (Job E_Commerce_Customer)

tFilterRow_2

Basic settings Schema Built-In Edit schema Sync columns

Advanced settings Logical operator used to combine conditions Or

Dynamic settings Conditions

InputColumn	Function	Operator	Value
CustomerID	Empty	Equals	null
Age	Empty	Equals	null
Gender	Empty	Equals	null
Location	Empty	Equals	null
MembershipLevel	Empty	Equals	null
TotalPurchases	Empty	Equals	null
TotalSpent	Empty	Equals	null
FavoriteCategory	Empty	Equals	null
LastPurchaseDate	Empty	Equals	null
Occupation	Empty	Equals	null
WebsiteVisitsFrequency	Empty	Equals	null
Churn	Empty	Equals	null

- j. After that I link both filter to tFileOutputDelimited respectively. The file name and setting for each output file as below.

Designer | Code

Job(E_Commerce_Customer 0.1) Contexts(E_Commerce_Customer) Component Run (Job E_Commerce_Customer)

tFileOutputDelimited_1

Basic settings Property Type Built-In

Advanced settings Use Output Stream

Dynamic settings File Name "C:/Users/User/Downloads/TOS_BD-20211109_1610-V8.0.1/workspace/clean_E_Commerce_Customer.csv"

View Row Separator "\n" Field Separator ","

Documentation Append Include Header Compress as zip file

Schema Built-In Edit schema Sync columns

Designer | Code

Job(E_Commerce_Customer 0.1) Contexts(E_Commerce_Customer) Component Run (Job E_Commerce_Customer)

tFileOutputDelimited_2

Basic settings Property Type Built-In

Advanced settings Use Output Stream

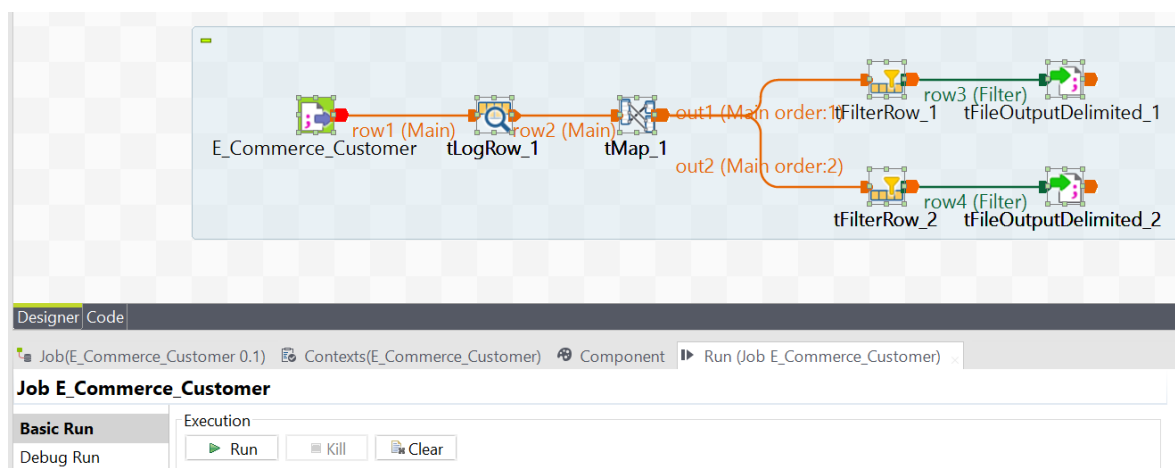
Dynamic settings File Name "C:/Users/User/Downloads/TOS_BD-20211109_1610-V8.0.1/workspace/missing_E_Commerce_Customer.csv"

View Row Separator "\n" Field Separator ","

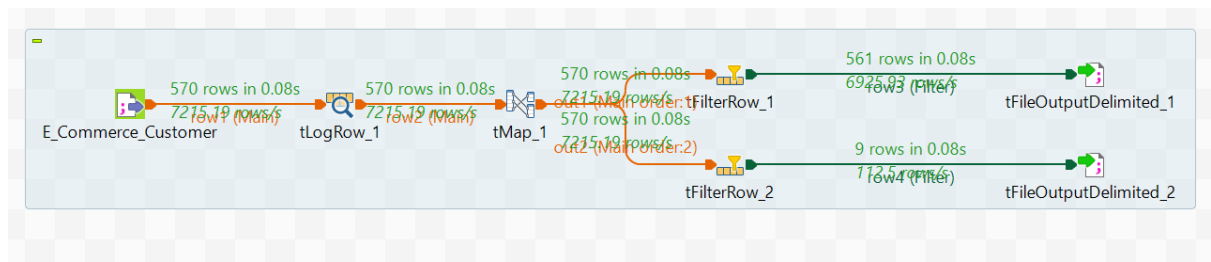
Documentation Append Include Header Compress as zip file

Schema Built-In Edit schema Sync columns

- k. The overall design as below. Once confirm I click 'Run'

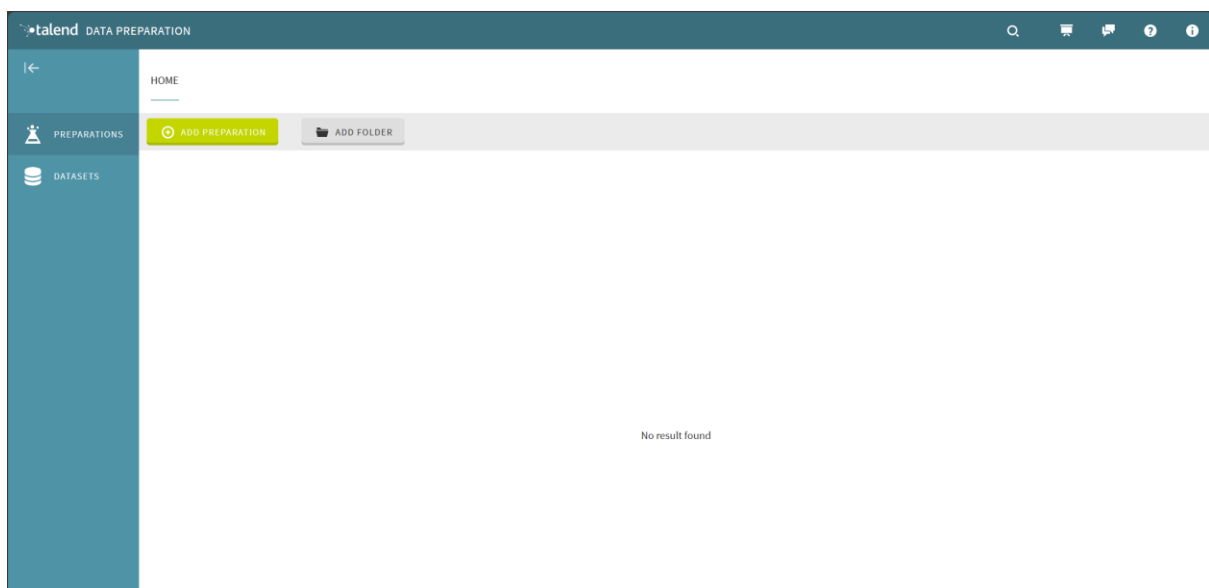


I. The outcome as below. From here I found there 9 three rows being filter due to missing data.



The assessment continue with the Talend preparation for deeper data preprocessing and understanding.

a) Import Data



- b) After successfully import data prio clean in Talend data integration , I start study the data
- From the value section, can see that there are 561 counts which means I have 561 rows left after the first round cleaning.

talend DATA PREPARATION

clean_E_Commerce_Customer Preparation

Filters

Add a filter ...

	CustomerID	Age	Gender	Location	MembershipLevel	TotalPurchases	TotalSpent	FavoriteCategory	LastPurchaseDate
	integer	integer	gender	city	city	integer	integer	text	date
1	1	38	Male	Selangor	Gold	10	500	Electronics	12/01/2023
2	2	25	Female	Kuala Lumpur	Silver	5	50	Clothing	11/15/2023
3	3	40	Male	Penang	Bronze	15	800	HomeGoods	12/20/2023
4	4	35	Female	Johor	Platinum	20	1200	Electronics	12/10/2023
5	5	28	Male	Melaka	Silver	8	400	Clothing	12/05/2023
6	6	45	Female	Negeri Sembilan	Gold	12	700	HomeGoods	11/30/2023
7	7	32	Male	Terengganu	Bronze	18	950	Electronics	12/25/2023
8	8	27	Female	Pahang	Silver	7	350	Clothing	11/20/2023
9	9	38	Male	Sabah	Gold	14	900	HomeGoods	12/10/2023
10	10	31	Female	Sarawak	Platinum	22	1500	Electronics	12/08/2023
11	11	22	Male	Perlis	Silver	3	200	Clothing	12/14/2023
12	12	29	Male	Selangor	Bronze	10	600	Electronics	12/12/2023
13	13	33	Female	Kuala Lumpur	Platinum	30	1500	Clothing	12/03/2023
14	14	42	Male	Penang	Gold	17	1100	HomeGoods	11/25/2023
15	15	36	Female	Johor	Silver	16	950	Clothing	12/28/2023
16	16	26	Male	Kuala Lumpur	Gold	8	400	Electronics	12/14/2023
17	17	34	Female	Selangor	Bronze	12	750	HomeGoods	11/30/2023
18	18	44	Male	Pahang	Silver	14	850	Clothing	12/10/2023
19	19	20	Female	Terengganu	Platinum	20	1300	Electronics	12/10/2023
20	20	31	Male	Perlis	Bronze	10	550	Electronics	12/15/2023
21	21	27	Female	Melaka	Silver	6	320	HomeGoods	12/01/2023
22	22	35	Male	Selangor	Platinum	18	1100	Clothing	11/25/2023
23	23	42	Female	Kuala Lumpur	Gold	4	900	Electronics	12/10/2023
24	24	29	Male	Penang	Silver	12	700	Clothing	11/30/2023
25	25	46	Female	Terengganu	Bronze	15	850	HomeGoods	12/05/2023
26	26	33	Male	Johor	Platinum	20	950	Electronics	12/20/2023

CustomerID

COLUMN ROW

Find a function ...

SUGGESTIONS

Compare numbers...

Add, multiply, subtract or divide...

BOOLEAN

Negate value

COLUMNS

Concatenate with...

Delete column

CHART VALUE PATTERN ADVANCED

Count: 561 Min: 1

Distinct: 0

Duplicate: 0 Max: 570

Valid: 561 Mean: 283.07

Empty: 0

Invalid: 0 Variance: 26715.08

c) Clean the data

- Notice there are some attributes wrongly label and missing value (white box)

CustomerID	Age	Gender	Location	MembershipLevel	TotalPurchases	TotalSpent	FavoriteCategory	LastPurchaseDate	Occupation	WebsiteVisitsFrequency	Churn
integer	integer	gender	city	city	integer	integer	text	date	job_title	last_name	boolean

1. Change the label

- For example change the label for attribute Membership level from city to text and the attribute website visit frequency level from last_name to text

Location	MembershipLevel	TotalPurchases	TotalSpent
			integer
Selangor			
Kuala Lumpur			
Penang			
Johor			
Melaka			
Negeri Sembilan			
Terengganu	Bronze		
Pahang	Silver		
Sabah	Gold		
Sarawak	Platinum		
Perlis	Silver		
Selangor	Bronze		

This column is a city

Duplicate column

Rename column

Create new column

Delete column

City 76.84 %

Last Name 56.84 %

First Name 27.19 %

Airport 20 %

Set as TEXT

Set as BOOLEAN

Set as DATE

Set as INTEGER

Set as DECIMAL

2. Remove missing data

- Notice there are white box at the corner of certain attribute. In the values section it show that out of 561 instance there are 2 empty cell in Favourite category and 6 missing in job_title. I decide to remove the row that missing data as my data consider big therefore deleted few rows will not effect my analysis.

- Notice there are data with similar text that might be due to typo during data entry. This were found in the attribute of location and Favourite category.

FIND AND GROUP SIMILAR TEXT

Replace all similar values with the right one (i.e. cluster on fuzzy matching)

<input checked="" type="checkbox"/>	These values have been found	This value will be kept
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Johor <input checked="" type="checkbox"/> Jehor	Replace value: <input type="text" value="Johor"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> KualaLumpur <input checked="" type="checkbox"/> Kuala Lumpur <input checked="" type="checkbox"/> KualailLumpur	Replace value: <input type="text" value="Kuala Lumpur"/>

SUBMIT

FIND AND GROUP SIMILAR TEXT

Replace all similar values with the right one (i.e. cluster on fuzzy matching)

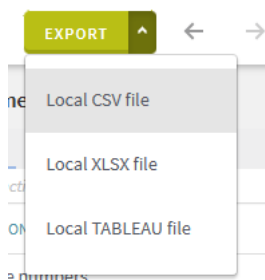
<input checked="" type="checkbox"/>	These values have been found	This value will be kept
<input checked="" type="checkbox"/>	<div><div><input checked="" type="checkbox"/> Home Goods</div><div><input checked="" type="checkbox"/> HomeGoods</div></div>	<div>Replace value:</div> <div>HomeGoods</div>

SUBMIT

d) After done cleaning the data we can see that the dataset had been reduce to 554 rows.

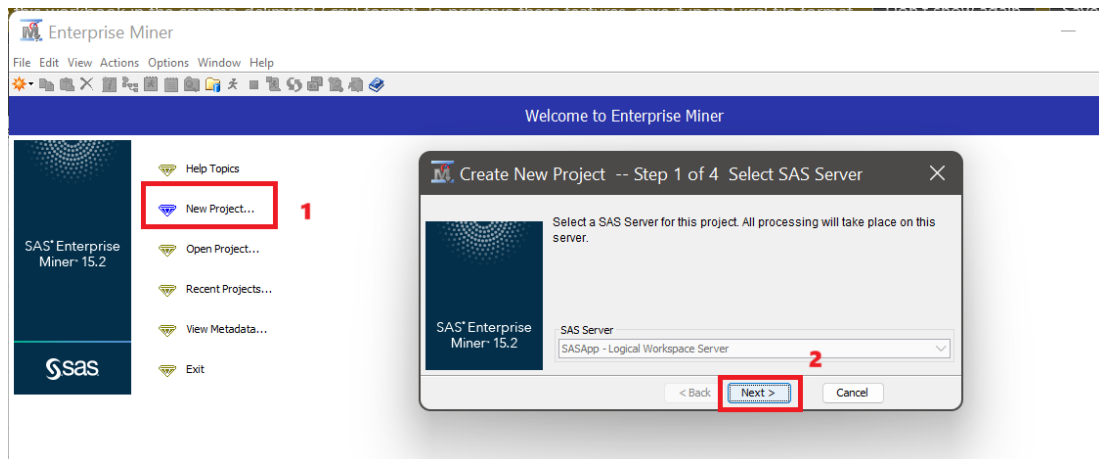
The screenshot shows the SAS Enterprise Miner interface with a project named 'clean_E_Commerce_Customer Preparation'. On the left, a list of cleaning steps is visible: 1. Change data type on column MembershipLevel; 2. Change data type on column WebsiteVisitsFrequency; 3. Delete the rows with empty cell on column FavoriteCategory; 4. Delete the rows with empty cell on column Occupation; 5. Find and group similar text on column Location; 6. Find and group similar text on column FavoriteCategory. A filter is applied: 'LastPurchaseDate: rows with valid values'. The main window displays a table with 26 rows and 7 columns: TotalPurchases, TotalSpent, FavoriteCategory, LastPurchaseDate, Occupation, WebsiteVisitsFrequency, and a hidden column. The table shows data for various products like Electronics, Clothing, and HomeGoods, with columns for purchase date, occupation, and website visits frequency. On the right, a 'FavoriteCategory' panel shows suggestions for filtering, including 'Delete these filtered rows', 'Keep these filtered rows', 'Change to upper case', 'Replace the cells that match...', 'Change to title case', and 'Change to lower case'. Below this, a 'BOOLEAN' section shows 'Apply changes to: All rows' and 'Filtered rows'. A 'VALUE' section shows statistics: Count: 554, Avg length: 9.43, Distinct: 3, Duplicate: 551, Valid: 554, Empty: 0, Invalid: 0, Min length: 8, Max length: 11.

e) Once confirm, I export the dataset in CSV file.

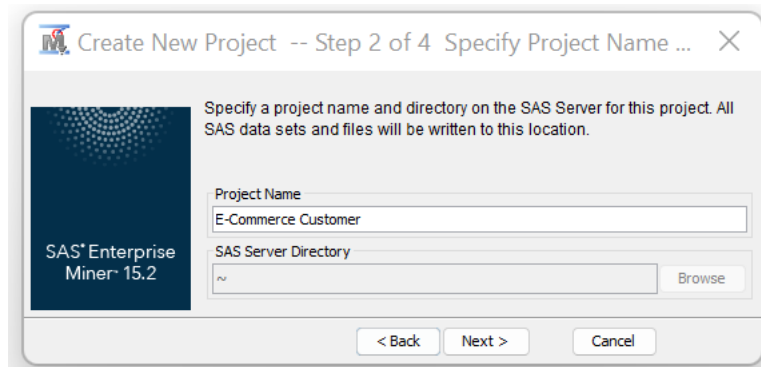


The project continued with tools SAS Enterprise Miner.

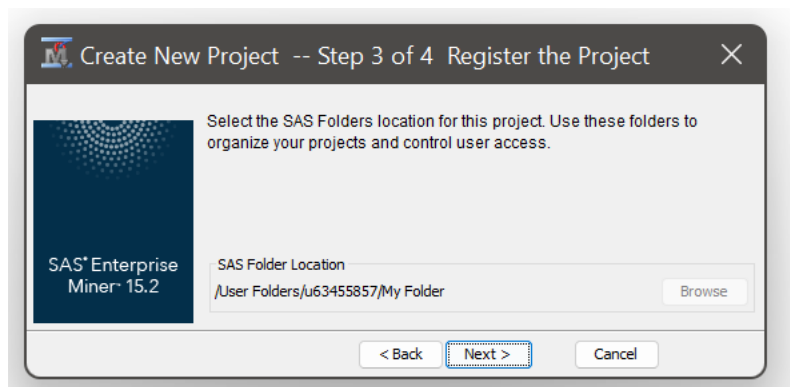
a. First, I create New project, a window will pop out and I click next.



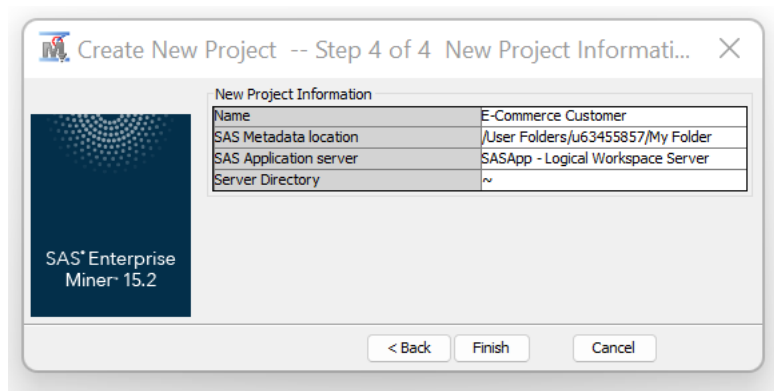
b. Insert Project name and click next.



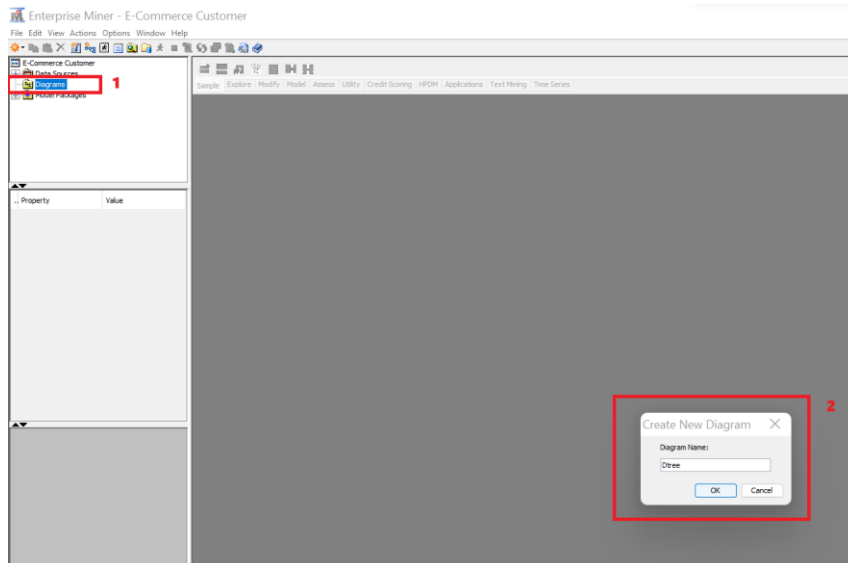
c. Click next 1 more time.



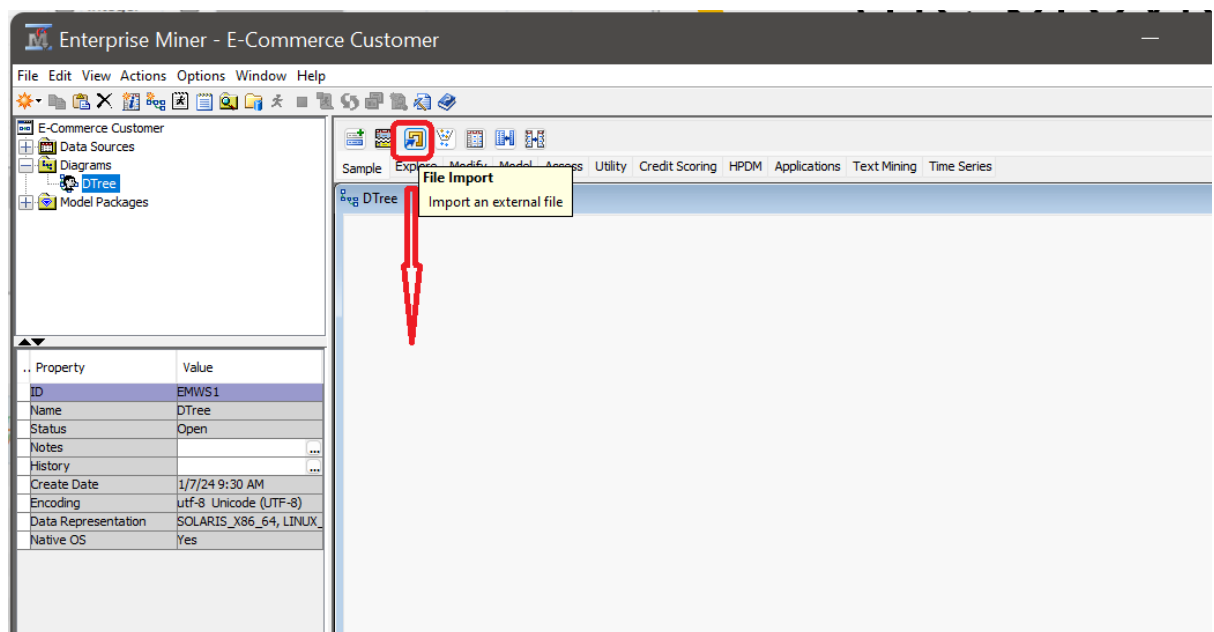
d. Then click finish.



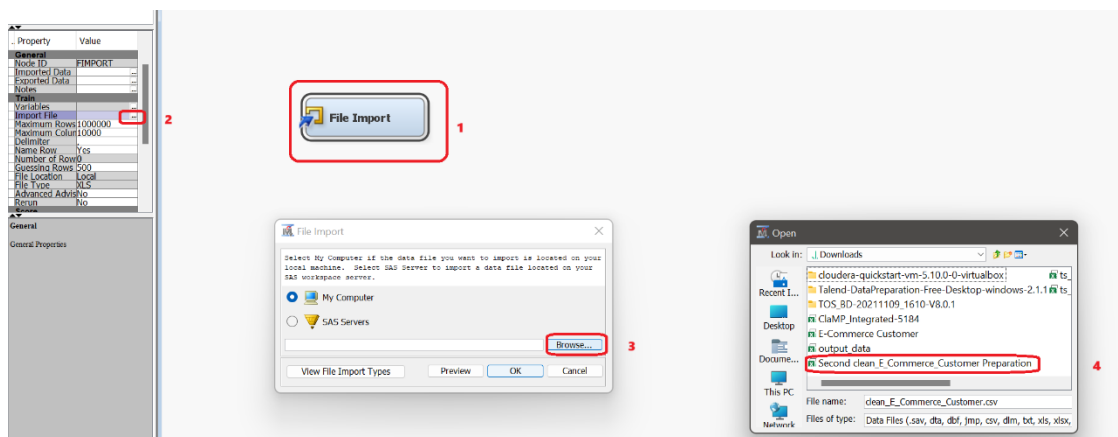
e. Through right click the 'Diagrams', I click create diagram and name it 'DTree'.



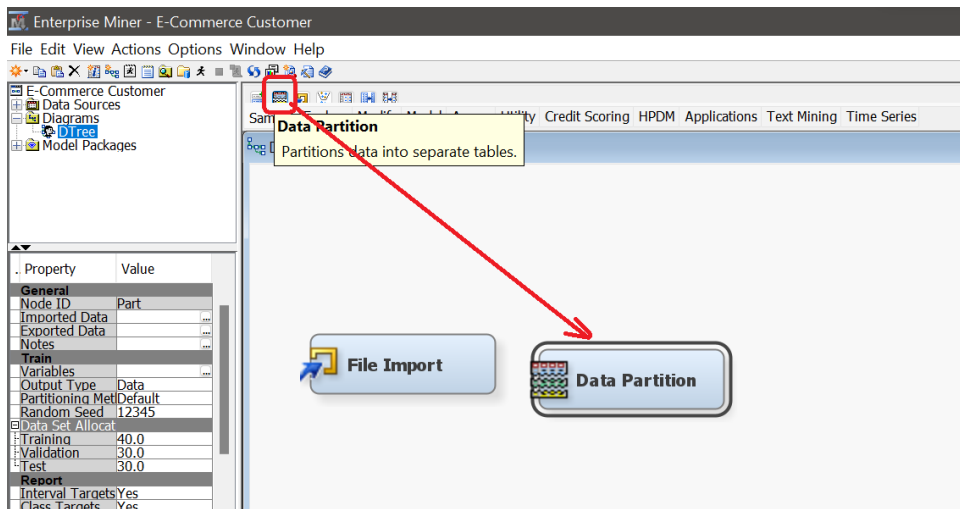
f. Drag the File import node down.



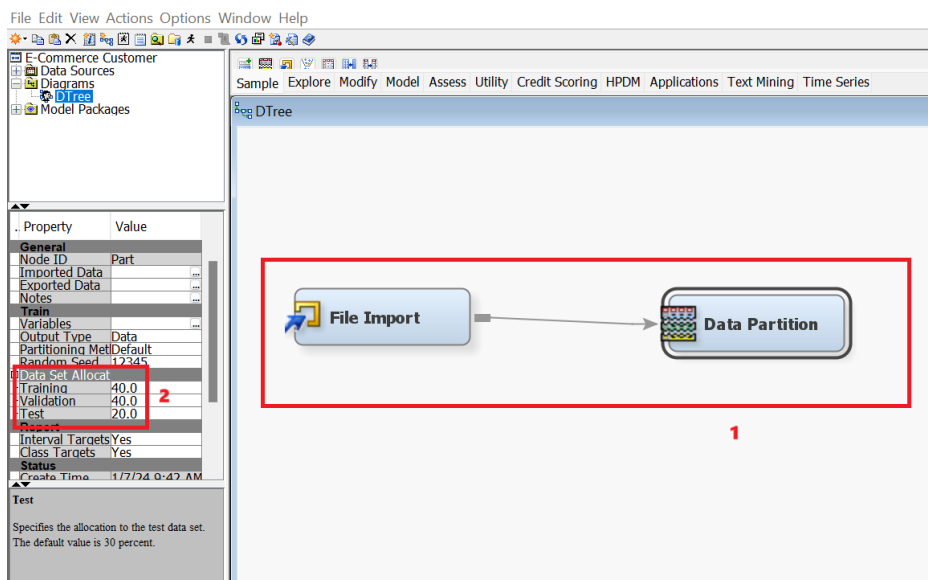
g. Click the 'File Import' node, then click the '...', click 'Browse' and select data to be use.



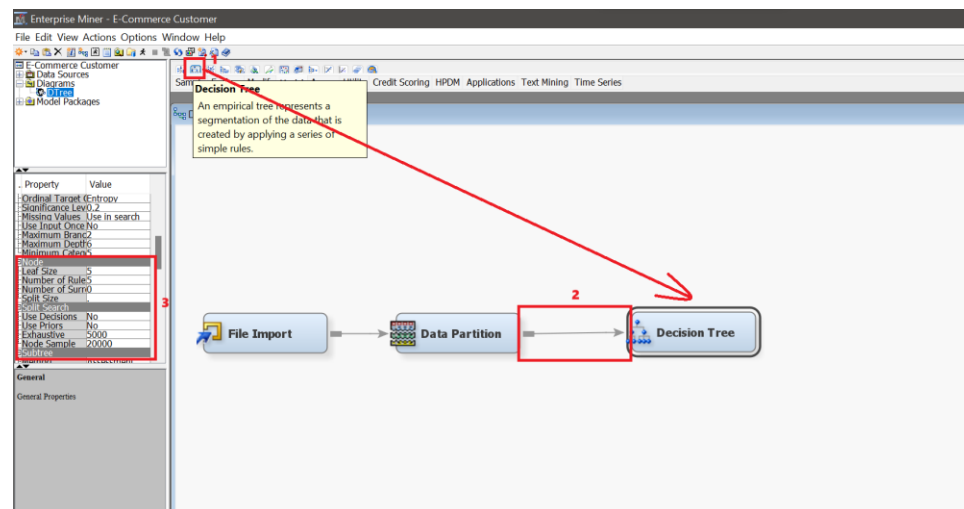
h. Drag the 'Data Partition' node down for splitting data into training, validating and testing.



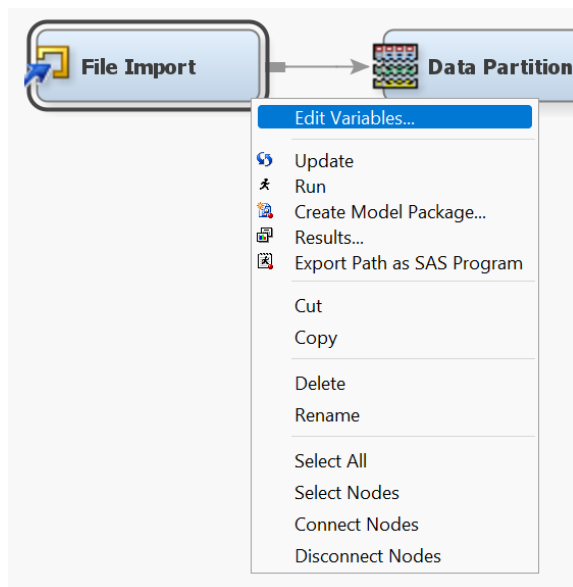
- i. Link both nodes together. Change the Dataset Allocation to 4:4:2 for Training, Validation and Test.



- j. Drag the Decision Tree node down and connect to Data partition. At the right corner, I adjust the maximum branch, Depth and categorical size.



- k. Right Click File Import node and select edit variable.



- l. Below is my adjustment for my variables. Churn will be my target as I want to predict whether a customer will churn or not. I drop customer id as it is not important in this project. Once confirm, I Click ok.

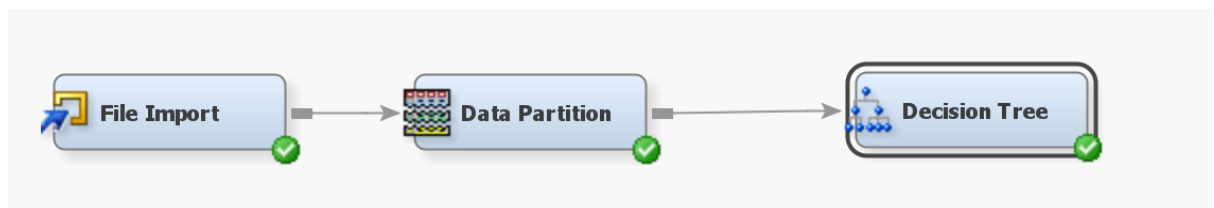
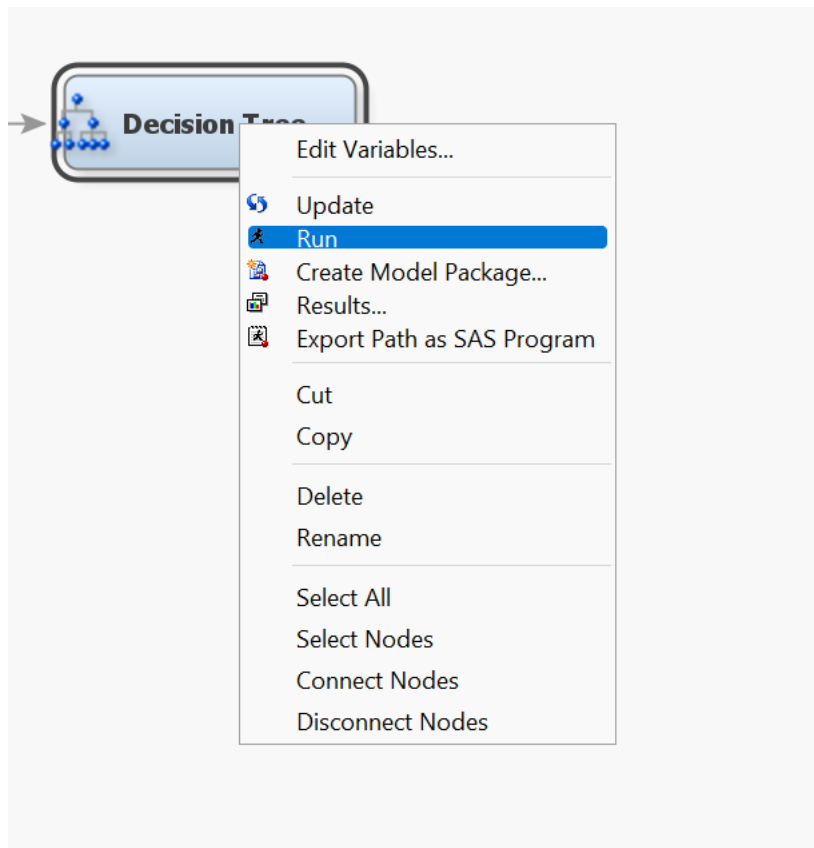
Variables - FIMPORT

(none) ☐ not Equal to ☐ Mining ☐ Basic

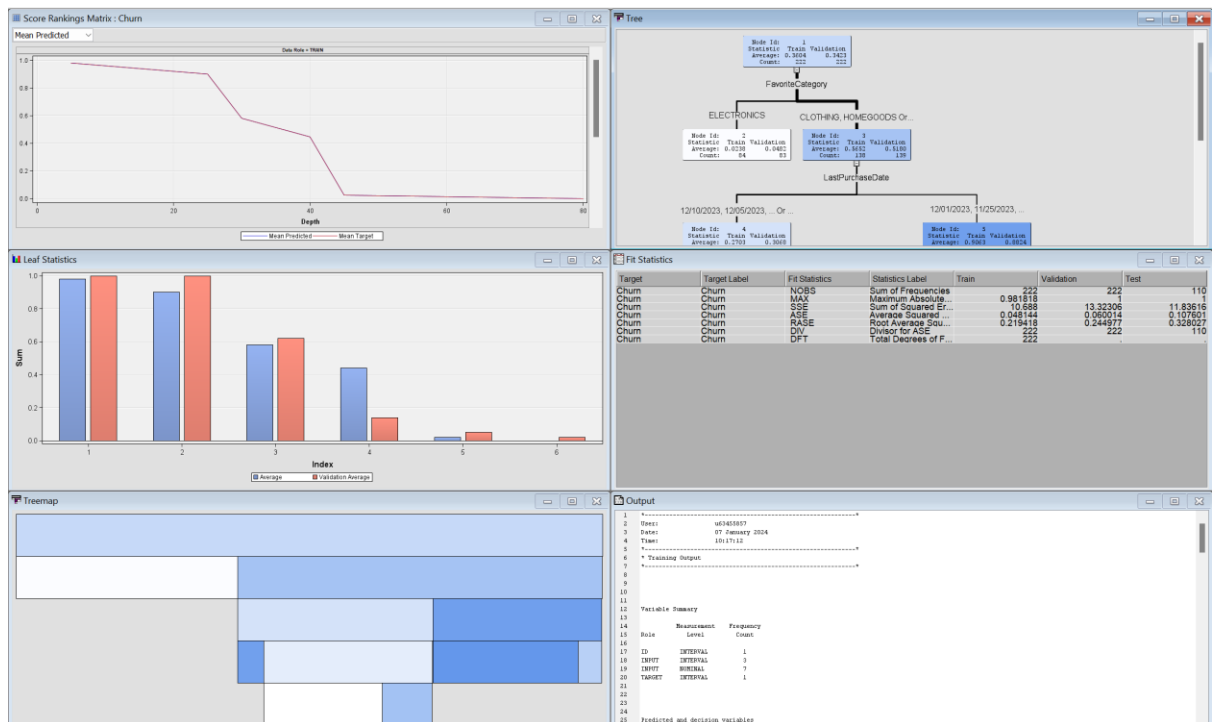
Columns: ☐ Label ☐ Mining ☐ Basic

Name	Role	Level	Report	Order	Drop	Lower Limit	Upper Limit
Age	Input	Interval	No		No	.	.
Churn	Target	Interval	No		No	.	.
CustomerID	Input	Interval	No		Yes	.	.
FavoriteCate	Input	Nominal	No		No	.	.
Gender	Input	Nominal	No		No	.	.
LastPurchase	Input	Nominal	No		No	.	.
Location	Input	Nominal	No		No	.	.
MembershipL	Input	Nominal	No		No	.	.
Occupation	Input	Nominal	No		No	.	.
TotalPurchas	Input	Interval	No		No	.	.
TotalSpent	Input	Interval	No		No	.	.
WebsiteVisits	Input	Nominal	No		No	.	.

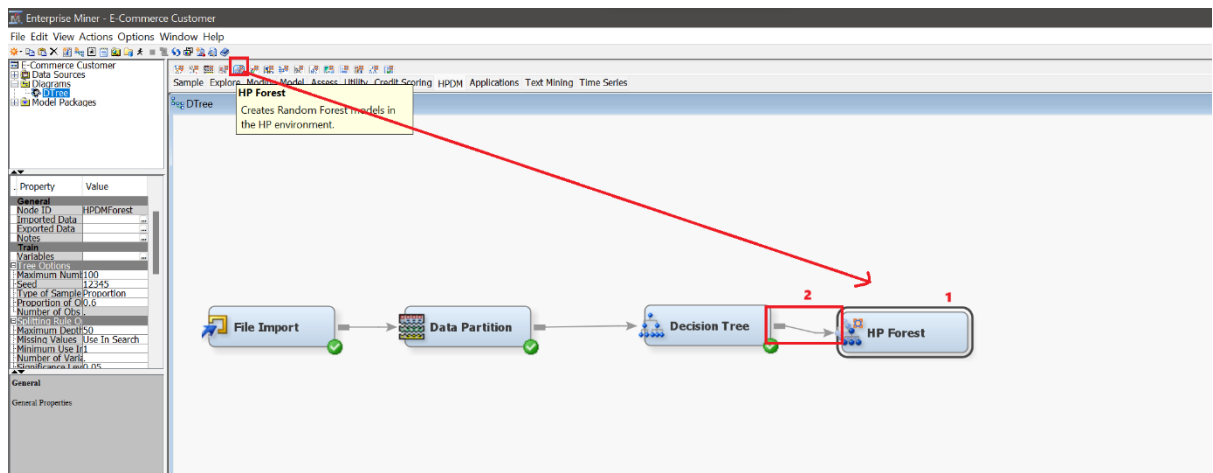
- m. Right click Decision Tree node and click Run.



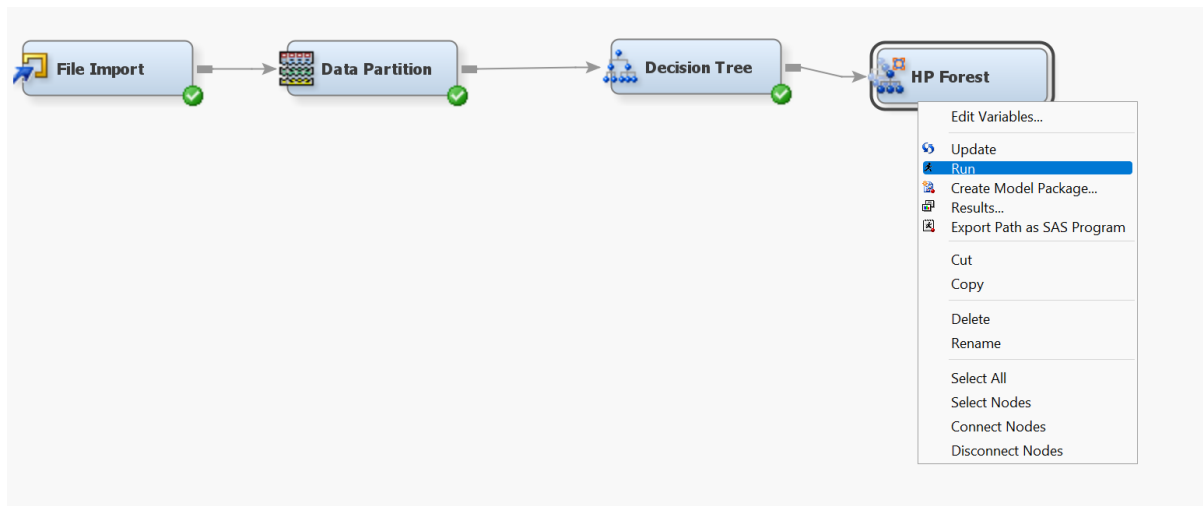
- n. Click result to see the outcome. From The decision tree it seems to have identified "FavoriteCategory" as the most important variable for predicting churn, followed by "LastPurchaseDate," "Age," and "Occupation."



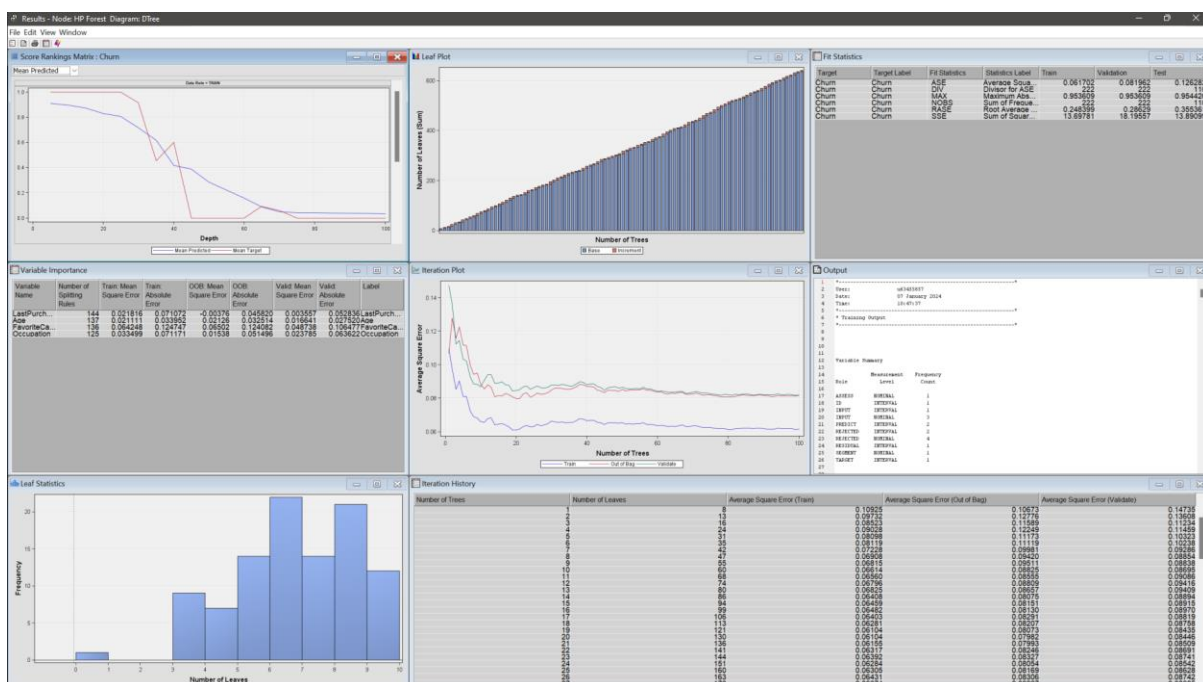
- o. To apply Bagging Random Forest algorithm involves utilizing the corresponding nodes in the diagram. I drag 'HP Forest' which stand for Random forest in Sas and connect it to decision tree model node. This mean that the Random Forest will be built based on the individual decision trees I created earlier.



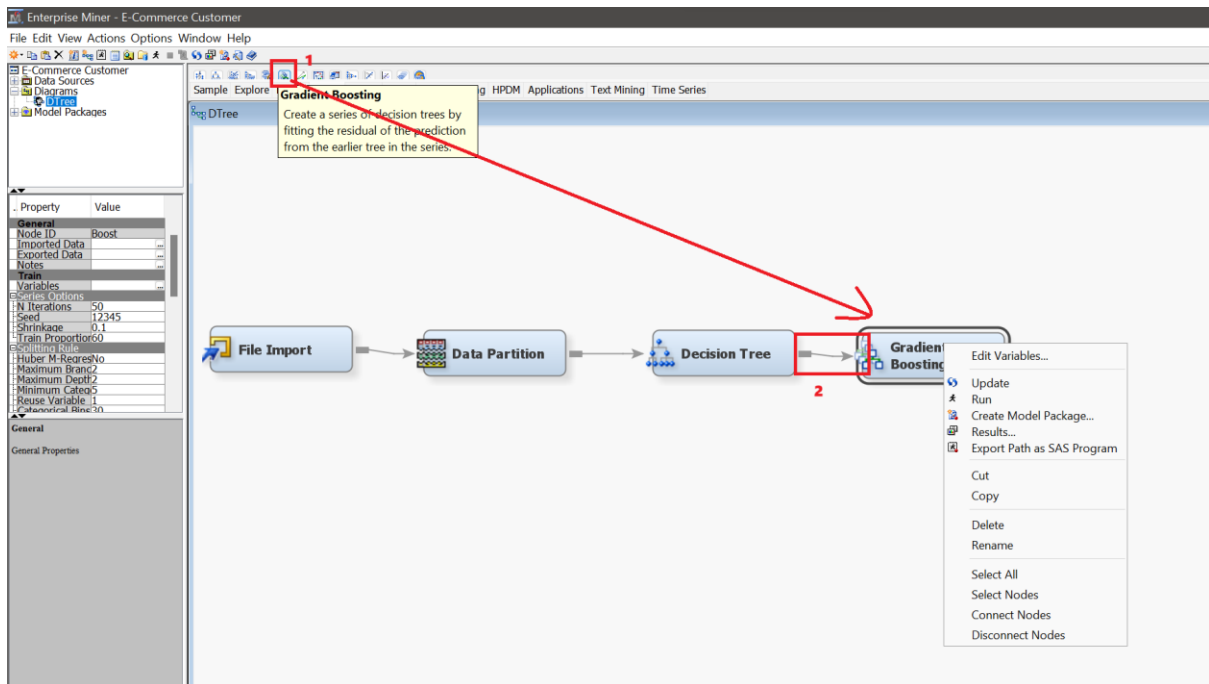
- p. Right click HP Forest and run



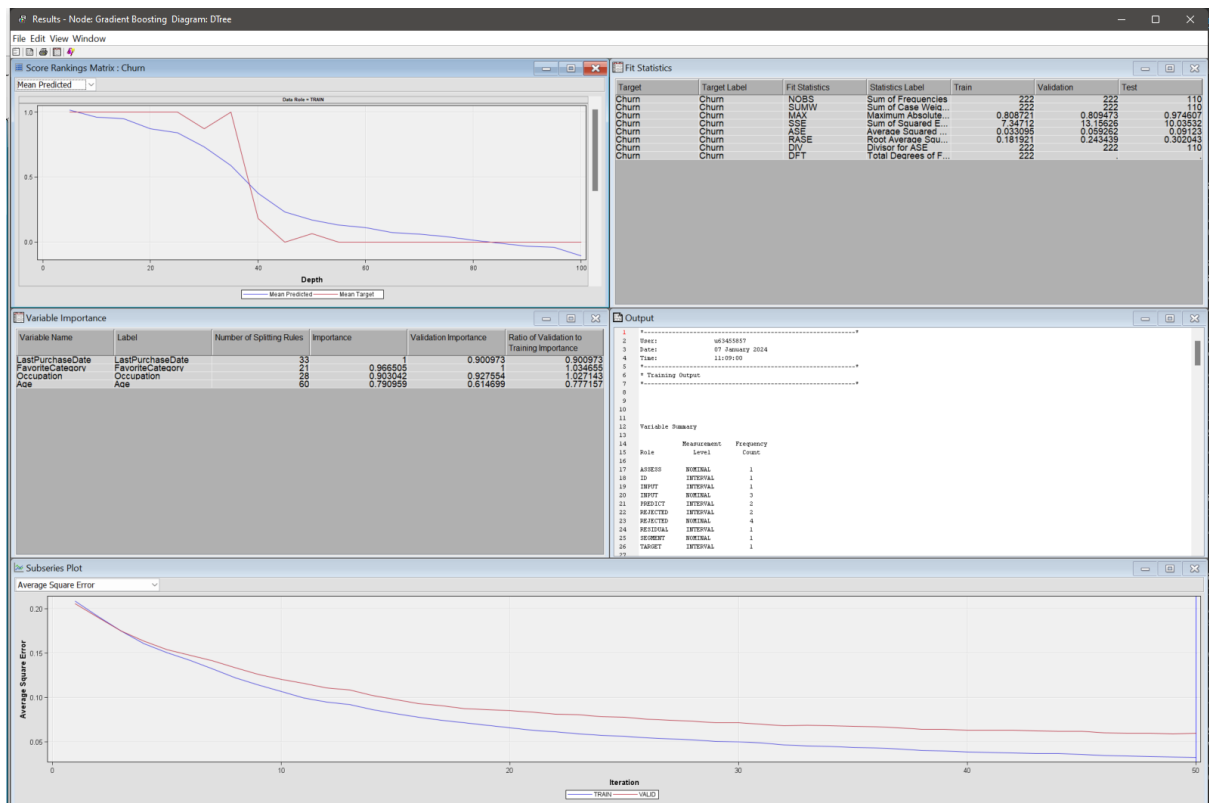
- q. Once it done, click result to see the outcome. The Random Forest model, employing 100 trees, was trained on 222 observations, demonstrating competitive fit statistics. The average square error decreased with an increasing number of trees, stabilizing at 0.06170. Key predictors for predicting churn included "FavoriteCategory," "Age," "Occupation," and "LastPurchaseDate." The model's assessment score rankings showcase its efficacy across various depth levels. The scoring process was efficient, taking minimal time. The model's predictive accuracy was assessed on training, validation, and test datasets, yielding Root Average Squared Errors of 0.248, 0.286, and 0.355, respectively. Overall, the Random Forest demonstrated robust predictive performance in identifying potential churn instances.



- r. To apply Boosting using Random Forest, I drag "Gradient Boosting" node from the onto the diagram then connect the Decision Tree model node to the Gradient Boosting node. This establishes the base learner for boosting. For the properties I follow the default. Once confirm right click 'Gradient Boosting' and click 'run'.

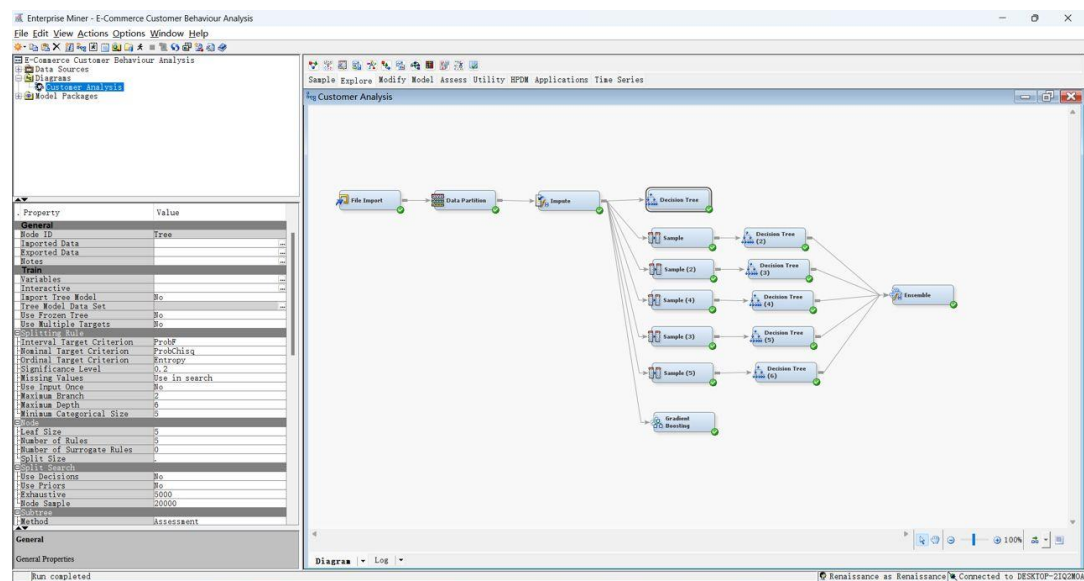


- s. Once done, click result to see the outcome. The analysis predicting customer churn revealed significant predictors. The key variables influencing the model include "LastPurchaseDate," "FavoriteCategory," "Occupation," and "Age." The Fit Statistics show good performance on the training set, with an average squared error of 0.033. The Assessment Score Rankings illustrate the model's ability to differentiate between churn and active customers across different depths. Variable Importance indicates "LastPurchaseDate" as the most influential. Overall, the model demonstrates promising predictive power, emphasizing the importance of recent purchase behavior and customer demographics in identifying potential churn.



There another way in getting the bagging process which is below thohrugh esemble. We create few samples with different percentage then link decision tree for analysis. For this case I repeated for 5 times. Lastly I connect all using the ensemble node.

The analyst trained a model ensemble, combining models labeled TREE2 to TREE6 using the average probability function. The training fit statistics indicate an average squared error (ASE) of 0.0318 and a root average squared error (RASE) of 0.1782. Assessment score rankings show varying prediction accuracies across different depths, with observations at depth 5 achieving a perfect match. The assessment score distribution illustrates the model's ability to differentiate between high and low predicted churn probabilities.



Result show as below.

