045052-p2

April 13, 2024

Machine Learning For Managers Project Supervised Learning

Name: Shantanu Bharvirkar

Roll No. 045052

0.0.1 1. Project Objectives | Problem Statements

1.1. PO1 | PS1: Classification of Consumer Data into Segments | Clusters | Classes using Supervised Learning Classification Algorithms Objective (PO1): The primary objective is to classify consumer data into distinct segments, clusters, or classes using supervised learning classification algorithms.

Problem Statement (PS1): Given the consumer dataset containing various features such as education, marital status, income, etc., the task is to develop a predictive model that accurately classifies consumers into predefined segments or classes based on their characteristics. This involves the application of supervised learning techniques to train a classification model that can effectively distinguish between different consumer groups.

1.2. PO2 | PS2: Determination of an Appropriate Classification Model Objective (PO2): To determine the most suitable classification model for accurately predicting consumer segments based on the dataset features.

Problem Statement (PS2): With several classification algorithms available, including logistic regression, support vector machines, decision trees, and k-nearest neighbors, the goal is to identify the model that yields the highest predictive performance for segmenting consumers. This involves comparing the performance metrics of different classification models and selecting the one that best fits the dataset characteristics.

1.3. PO3 | PS3: Identification of Important | Contributing | Significant Variables or Features and their Thresholds for Classification Objective (PO3): To identify the key variables or features that significantly contribute to the classification of consumers into segments and determine their optimal thresholds.

Problem Statement (PS3): Understanding which features have the most significant impact on segmenting consumers is crucial for model interpretability and performance improvement. Therefore, the objective is to conduct feature importance analysis and determine the thresholds for these important variables, enhancing the classification model's effectiveness in accurately predicting consumer segments.

0.0.2 2. Description of Data

Data Description 1. ID: Unique identifier for each customer. 2. Year_Birth: Year of birth of the customer. 3. Education: Education level of the customer. 4. Marital_Status: Marital status of the customer. 5. Income: Income of the customer. 6. Kidhome: Number of small children in the customer's household. 7. Teenhome: Number of teenagers in the customer's household. 8. Dt_Customer: Date when the customer joined as a customer. 9. Recency: Number of days since the last purchase. 10. MntWines: Amount spent on wines. 11. MntFruits: Amount spent on fruits. 12. MntMeatProducts: Amount spent on meat products. 13. MntFishProducts: Amount spent on fish products. 14. MntSweetProducts: Amount spent on sweet products. 15. MntGoldProds: Amount spent on gold products. 16. NumDealsPurchases: Number of purchases made with discount. 17. NumWebPurchases: Number of purchases made through the website. 18. NumCatalogPurchases: Number of purchases made using a catalog. 19. NumStorePurchases: Number of purchases made directly in stores. 20. NumWebVisitsMonth: Number of visits to the website per month. 21. AcceptedCmp3-5: Binary flags indicating whether the customer accepted marketing campaigns 3 through 5. 22. AcceptedCmp1-2: Binary flags indicating whether the customer accepted marketing campaigns 1 and 2. 23. Complain: Binary flag indicating whether the customer has ever complained. 24. Z_CostContact: Cost of contacting the customer. 25. Z_Revenue: Revenue from contacting the customer. 26. Response: Binary flag indicating whether the customer responded to the campaign.

2.1. Data Source, Size, Shape

- **2.1.1. Data Source (Website Link):** www.Kaggle.com
- **2.1.2. Data Size:** 7.45 MB
- **2.1.3. Data Shape (Dimension: Number of Variables | Number of Records):** (59999, 30)

2.2. Description of Variables

2.2.1. Index Variable(s)

- **Index Variable(s):** S.no

2.2.2. Variables or Features having Categories | Categorical Variables or Features (CV)

- **2.2.2.1. Variables or Features having Nominal Categories | Categorical Variables or Features - Nominal Type:**
  - Year_Birth, Education, Marital_Status, Teenhome, Response, AcceptedCmp3, AcceptedCmp4, AcceptedCmp5, AcceptedCmp1, AcceptedCmp2, Complain

2.2.3. Non-Categorical Variables or Features

- **Non-Categorical Variables or Features:**
  - Income, Kidhome, Recency, MntWines, MntFruits, MntMeatProducts, MntFishProducts, MntSweetProducts, MntGoldProds, NumDealsPurchases, NumWebPurchases, NumCatalogPurchases, NumStorePurchases, NumWebVisitsMonth, Z_CostContact, Z_Revenue

0.0.3 2.3. Descriptive Statistics

2.3.1. Descriptive Statistics: Categorical Variables or Features

2.3.1.1. Count | Frequency Statistics

- **Count | Frequency Statistics:**
  - S.no: Count = 59999
  - Year_Birth: Count = 59999
  - Education: Count = 59999
  - Marital_Status: Count = 59999
  - Teenhome: Count = 59999
  - Response: Count = 59999
  - AcceptedCmp3: Count = 59999
  - AcceptedCmp4: Count = 59999
  - AcceptedCmp5: Count = 59999
  - AcceptedCmp1: Count = 59999
  - AcceptedCmp2: Count = 59999
  - Complain: Count = 59999

2.3.2. Descriptive Statistics: Non-Categorical Variables or Features

2.3.2.1. Measures of Central Tendency

- **Measures of Central Tendency:**
  - S.no: Mean = 30000.0, Min = 1.0, Max = 59999.0
  - Income: Mean = 81359.28, Min = 1000.0, Max = 666480.0
  - Kidhome: Mean = 0.441457, Min = 0.0, Max = 2.0
  - Recency: Mean = 48.698862, Min = 0.0, Max = 103.0
  - MntWines: Mean = 308.099902, Min = 0.0, Max = 1540.0
  - MntFruits: Mean = 26.573010, Min = 0.0, Max = 203.0
  - MntMeatProducts: Mean = 166.544309, Min = 0.0, Max = 1731.0
  - MntFishProducts: Mean = 40.871165, Min = 0.0, Max = 278.0
  - MntSweetProducts: Mean = 28.414324, Min = 0.0, Max = 271.0
  - MntGoldProds: Mean = 43.856964, Min = 0.0, Max = 368.0
  - NumDealsPurchases: Mean = 2.375523, Min = 0.0, Max = 17.0
  - NumWebPurchases: Mean = 3.831597, Min = 0.0, Max = 29.0
  - NumCatalogPurchases: Mean = 2.949816, Min = 0.0, Max = 30.0
  - NumStorePurchases: Mean = 5.341272, Min = 0.0, Max = 15.0
  - NumWebVisitsMonth: Mean = 4.919599, Min = 0.0, Max = 22.0
  - Z_CostContact: Mean = 3.0, Min = 3.0, Max = 3.0
  - Z_Revenue: Mean = 11.0, Min = 11.0, Max = 11.0

0.0.4 3. Analysis of Data

3.1. Data Pre-Processing

3.1.1. Missing Data Statistics and Treatment 3.1.1.1. Missing Data Statistics: Records

The analysis of missing data at the record level indicates a complete dataset with no missing records. This observation is crucial as it ensures the integrity and reliability of the dataset, providing a solid foundation for subsequent analysis and modeling. The absence of missing records simplifies data processing and interpretation, allowing for a comprehensive understanding of the entire dataset without the need for imputation or removal of incomplete records.

3.1.1.2. Missing Data Treatment: Records

Given that no records contain more than 50% missing data, no removal or imputation procedures were deemed necessary. This absence of significant missing data in records alleviates concerns regarding potential biases introduced by incomplete observations. Consequently, the dataset remains intact, preserving the integrity and comprehensiveness of the information contained within each record.

3.1.1.3. Missing Data Statistics and Treatment: Categorical Variables or Features

Analysis of missing data pertaining to categorical variables reveals no instances of missing values across the dataset. This finding underscores the completeness of categorical data, indicating that all categorical variables have been fully populated. As a result, no removal or imputation of missing values was required, ensuring the preservation of categorical data integrity and the reliability of subsequent analyses involving categorical variables.

3.1.2. Numerical Encoding of Categorical Variables or Features (Encoding Schema - Alphanumeric Order) To facilitate further analysis and modeling, categorical variables were numerically encoded using the LabelEncoder from the scikit-learn library. This transformation converts categorical variables into numerical format, allowing for compatibility with machine learning algorithms that require numeric inputs. Each categorical variable was encoded alphabetically, ensuring consistency and ease of interpretation in subsequent analyses.

3.1.3. Outlier Statistics and Treatment (Scaling | Transformation) Detailed analysis and treatment of outliers in non-categorical variables were not provided in the current context. While outliers can impact the performance and accuracy of predictive models, their detection and treatment require careful consideration and domain knowledge. Future analyses may benefit from exploring outlier detection techniques such as z-score, interquartile range, or visual inspection through box plots. Treatment methods such as scaling or transformation may be applied to mitigate the influence of outliers on model performance and improve the robustness of the analysis.

3.1.4. Data Bifurcat& Data was Bifurcated in train test sets

0.0.5 3.2. Data Analysis

3.2.1.1. PO1 | PS1: Supervised Machine Learning Classification Algorithm: Decision Tree (Base Model) For the base model using a Decision Tree, the Gini Coefficient and Entropy were utilized as metrics for analysis.

3.2.1.2. PO1 | PS1: Supervised Machine Learning Classification Algorithms: Logistic Regression, Support Vector Machine, K Nearest Neighbour (Comparison Models) The following comparison models were assessed alongside the base model, employing various metrics for evaluation.

0.0.6 3.2.2. Model Performance Evaluation

3.2.2.1.1. PO2 | PS2: Classification Model Performance Evaluation - Base Model: Decision Tree The performance of the Decision Tree model was evaluated using a confusion matrix detailing accuracy, recall, precision, and F1-score. The model's time statistics and memory usage on both CPU and GPU were also measured.

3.2.2.2.1. PO2 | PS2: Classification Model Performance Evaluation - Comparison Models: Logistic Regression, Support Vector Machine, K Nearest Neighbour Similar evaluations were conducted for the comparison models, capturing their performance through similar metrics and resource usage statistics.

0.0.7 3.2.3. Variable or Feature Analysis

3.2.3.1. PO3 | PS3: Variable or Feature Analysis - Base Model (Decision Tree) The relevant and important variables or features identified for the Decision Tree model were "Income." The identified threshold for this feature was 0.0156.

3.2.3.2. PO3 | PS3: Variable or Feature Analysis - Comparison Models: Logistic Regression, Support Vector Machine, K Nearest Neighbour Similar analyses were conducted for the comparison models, revealing the significant features and their respective thresholds.

0.0.8 4. Results | Observations

4.1. Classification Model Parameters Parameters such as hyperparameters and settings for each model (Decision Tree, Logistic Regression, Support Vector Machine, K Nearest Neighbour) were scrutinized and compared.

4.2. Classification Model Performance: Time & Memory Statistics The performance metrics revealed distinctive time and memory usage statistics across all models, offering valuable insights into their efficiency.

4.3. Variable or Feature Analysis The critical variables or features for each model were highlighted, showcasing the factors that influenced model outcomes significantly.

4.3.1. List of Relevant or Important Variables or Features and their Thresholds The identified important variables and their respective thresholds for each model were documented, aiding in understanding their impact on model predictions.

4.3.2. List of Non-Relevant or Non-Important Variables or Features Variables deemed less impactful were also identified, assisting in refining models by focusing on key attributes.

0.0.9 5. Managerial Insights

5.1. Appropriate Model: Compare and Contrast A comprehensive comparison was made among the Decision Tree, Logistic Regression, Support Vector Machine, and K Nearest Neighbour models, elucidating their suitability for specific tasks based on performance metrics.

5.2. Relevant or Important Variables or Features (Given the Appropriate Model) The significant variables identified for each model provided actionable insights for decision-makers, enabling informed choices based on key predictors.

0.0.10 Detailed Report

In the conducted analysis, the Decision Tree model served as the base for comparison against Logistic Regression, Support Vector Machine (SVM), and K Nearest Neighbour (KNN) models. Each model was evaluated using a range of metrics to determine performance, including accuracy, recall, precision, and F1-score.

Decision Tree Model The Decision Tree model demonstrated robust performance in terms of accuracy, achieving a weighted F1-score of 0.95 on the training dataset. However, its performance on the test dataset dropped significantly to an accuracy of 0.33, indicating potential overfitting. The identified significant variable for this model was "Income," with a threshold of 0.0156.

Comparison Models The Logistic Regression model exhibited a fair level of accuracy, albeit slightly lower than the Decision Tree. SVM, on the other hand, yielded an accuracy of 0.33, similar to the Decision Tree's test performance. KNN, with varying k-values, displayed fluctuating accuracies around 0.33-0.34, indicating stable but relatively modest performance.

Model Parameters and Performance Metrics Parameters such as hyperparameters and resource utilization (CPU, GPU) were compared across models, providing insights into the computational efficiency of each approach. Memory usage was also a distinguishing factor, with SVM exhibiting relatively higher memory requirements.

Feature Analysis Critical features such as "Income" were identified as influential for model predictions, emphasizing the importance of feature selection and engineering in improving model performance.

Managerial Insights Based on the analysis, the Decision Tree model showcased competitive performance, particularly in training. However, caution is advised due to potential overfitting. Logistic Regression and SVM demonstrated comparable results, while KNN's performance was consistent but moderate.

In conclusion, the choice of the appropriate model depends on specific objectives and resource constraints. The identified significant features provide valuable insights for stakeholders, facilitating informed decision-making in deploying machine learning solutions effectively. vot customer segments based on their demographic and behavioral attributes. Understanding these segments can aid in targeted marketing strategies, product customization, and service offerings tailored to the specific needs and preferences of each cluster. Further analysis and refinement of the segmentation strategy can provide valuable insights for maximizing customer satisfaction and business growth.

[1]:
```
# Import library
# Required Libraries
import pandas as pd, numpy as np # For Data Manipulation
```

```
from sklearn.preprocessing import LabelEncoder, OrdinalEncoder # For Encoding␣
 ↪Categorical Data [Nominal | Ordinal]
from sklearn.preprocessing import OneHotEncoder # For Creating Dummy Variables␣
 ↪of Categorical Data [Nominal]
from sklearn.impute import SimpleImputer, KNNImputer # For Imputation of␣
 ↪Missing Data
from sklearn.preprocessing import StandardScaler, MinMaxScaler, RobustScaler #␣
 ↪For Rescaling Data
from sklearn.model_selection import train_test_split # For Splitting Data into␣
 ↪Training & Testing Sets
import matplotlib.pyplot as plt
import numpy as np
from scipy.stats import pearsonr
from scipy import stats

# Required Libraries
import pandas as pd, numpy as np # For Data Manipulation
import matplotlib.pyplot as plt, seaborn as sns # For Data Visualization
import scipy.cluster.hierarchy as sch # For Hierarchical Clustering
from sklearn.cluster import AgglomerativeClustering as agclus, KMeans as kmclus␣
 ↪# For Agglomerative & K-Means Clustering
from sklearn.metrics import silhouette_score as sscore, davies_bouldin_score as␣
 ↪dbscore # For Clustering Model Evaluation

# @title load library { display-mode: "form" }
# Load IPython extension for measuring time
!pip install ipython-autotime
%reload_ext autotime

# Load IPython extension for memory profiling
!pip install memory-profiler
%reload_ext memory_profiler

# Your imports
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import scipy.cluster.hierarchy as sch
from sklearn.cluster import AgglomerativeClustering as agclus, KMeans as kmclus
from sklearn.metrics import silhouette_score as sscore, davies_bouldin_score as␣
 ↪dbscore
from scipy.cluster.hierarchy import dendrogram, linkage
import plotly.graph_objects as go

# Load preprocessing libraries
```

```
from sklearn.preprocessing import LabelEncoder, OrdinalEncoder, OneHotEncoder
from sklearn.impute import SimpleImputer, KNNImputer
from sklearn.preprocessing import StandardScaler, MinMaxScaler, RobustScaler
from sklearn.model_selection import train_test_split

from scipy.stats import f_oneway
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import OneHotEncoder, StandardScaler
from sklearn.model_selection import train_test_split
import warnings
warnings.filterwarnings('ignore')
```

```
Requirement already satisfied: ipython-autotime in
c:\users\shantanu\anaconda3\lib\site-packages (0.3.2)
Requirement already satisfied: ipython in c:\users\shantanu\anaconda3\lib\site-
packages (from ipython-autotime) (8.15.0)
Requirement already satisfied: backcall in c:\users\shantanu\anaconda3\lib\site-
packages (from ipython->ipython-autotime) (0.2.0)
Requirement already satisfied: decorator in
c:\users\shantanu\anaconda3\lib\site-packages (from ipython->ipython-autotime)
(5.1.1)
Requirement already satisfied: jedi>=0.16 in
c:\users\shantanu\anaconda3\lib\site-packages (from ipython->ipython-autotime)
(0.18.1)
Requirement already satisfied: matplotlib-inline in
c:\users\shantanu\anaconda3\lib\site-packages (from ipython->ipython-autotime)
(0.1.6)
Requirement already satisfied: pickleshare in
c:\users\shantanu\anaconda3\lib\site-packages (from ipython->ipython-autotime)
(0.7.5)
Requirement already satisfied: prompt-toolkit!=3.0.37,<3.1.0,>=3.0.30 in
c:\users\shantanu\anaconda3\lib\site-packages (from ipython->ipython-autotime)
(3.0.36)
Requirement already satisfied: pygments>=2.4.0 in
c:\users\shantanu\anaconda3\lib\site-packages (from ipython->ipython-autotime)
(2.15.1)
Requirement already satisfied: stack-data in
c:\users\shantanu\anaconda3\lib\site-packages (from ipython->ipython-autotime)
(0.2.0)
Requirement already satisfied: traitlets>=5 in
c:\users\shantanu\anaconda3\lib\site-packages (from ipython->ipython-autotime)
(5.7.1)
Requirement already satisfied: exceptiongroup in
```

```
c:\users\shantanu\anaconda3\lib\site-packages (from ipython->ipython-autotime)
(1.0.4)
Requirement already satisfied: colorama in c:\users\shantanu\anaconda3\lib\site-
packages (from ipython->ipython-autotime) (0.4.6)
Requirement already satisfied: parso<0.9.0,>=0.8.0 in
c:\users\shantanu\anaconda3\lib\site-packages (from
jedi>=0.16->ipython->ipython-autotime) (0.8.3)
Requirement already satisfied: wcwidth in c:\users\shantanu\anaconda3\lib\site-
packages (from prompt-toolkit!=3.0.37,<3.1.0,>=3.0.30->ipython->ipython-
autotime) (0.2.5)
Requirement already satisfied: executing in
c:\users\shantanu\anaconda3\lib\site-packages (from stack-
data->ipython->ipython-autotime) (0.8.3)
Requirement already satisfied: asttokens in
c:\users\shantanu\anaconda3\lib\site-packages (from stack-
data->ipython->ipython-autotime) (2.0.5)
Requirement already satisfied: pure-eval in
c:\users\shantanu\anaconda3\lib\site-packages (from stack-
data->ipython->ipython-autotime) (0.2.2)
Requirement already satisfied: six in c:\users\shantanu\anaconda3\lib\site-
packages (from asttokens->stack-data->ipython->ipython-autotime) (1.16.0)

WARNING: Ignoring invalid distribution -atplotlib
(c:\users\shantanu\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib
(c:\users\shantanu\anaconda3\lib\site-packages)

Requirement already satisfied: memory-profiler in
c:\users\shantanu\anaconda3\lib\site-packages (0.61.0)
Requirement already satisfied: psutil in c:\users\shantanu\anaconda3\lib\site-
packages (from memory-profiler) (5.9.0)
time: 2.42 s (started: 2024-04-06 00:09:04 +05:30)

WARNING: Ignoring invalid distribution -atplotlib
(c:\users\shantanu\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -atplotlib
(c:\users\shantanu\anaconda3\lib\site-packages)
```

[2]:
```
# Load the CSV file into a pandas DataFrame
df = pd.read_csv(r"C:\Users\Shantanu\Downloads\new_marketing_campaign.csv")
```

```
time: 172 ms (started: 2024-04-06 00:09:07 +05:30)
```

[3]:
```
df
```

[3]:

	ID	Year_Birth	Education	\
0	e20d011e-c904-4301-871d-4120e734f189	1956	2n Cycle	
1	9038499f-e82e-43d3-803d-165dba9afe48	1950	Graduation	
2	a3aaaf2a-c063-45ba-ac1a-ab375d53faa3	1960	PhD	

```
3      fabc6e91-6e59-4dd6-8d07-572b7f07b9e4        1980         PhD
4      6bf0b99f-dd6d-4964-a3ca-1e98f500d9e1        1977  Graduation
…                                       …         …          …
59994  befde079-3b10-4bff-ac11-fb4390b9b90c        1967         PhD
59995  5d7e372b-b857-46b4-be71-be4a46c79c88        1974  Graduation
59996  22cf1a64-8133-466d-af61-d0475ed92b80        1963      Master
59997  6304a2fb-4c6e-46cb-ac70-fcacd96278f4        1974      Master
59998  b67a9fe3-6f44-422c-981d-aa52925c3bb5        1951  Graduation

      Marital_Status  Income  Kidhome  Teenhome Dt_Customer  Recency  \
0           Together   40831        0         1  21-07-2022       62
1           Together   22537        1         1  05-11-2021       42
2           Together  195755        1         0  24-12-2015       28
3           Divorced  143143        0         0  14-06-2017       24
4            Married   84648        0         1  17-06-2021       92
…              …     …      …       …         …      …
59994        Married    2688        0         1  19-08-2018       30
59995       Together   14577        1         0  01-05-2017       43
59996         Single    3282        1         1  05-03-2022       80
59997         Single   68946        1         0  18-08-2020       11
59998        Married  131663        0         0  15-04-2016       50

       MntWines  …  NumWebVisitsMonth  AcceptedCmp3  AcceptedCmp4  \
0           636  …                  5             1             1
1             6  …                  4             0             1
2           383  …                  4             0             1
3            29  …                  5             1             1
4           133  …                  7             1             1
…         …  …                 …           …           …
59994        13  …                  7             1             1
59995        10  …                  4             0             1
59996        33  …                  7             0             0
59997        37  …                  6             0             1
59998        19  …                  1             1             1

       AcceptedCmp5  AcceptedCmp1  AcceptedCmp2  Complain  Z_CostContact  \
0                 1             1             1         0              3
1                 1             0             0         0              3
2                 0             1             1         0              3
3                 0             0             0         0              3
4                 1             1             1         0              3
…               …           …           …       …            …
59994             1             1             0         0              3
59995             1             1             0         0              3
59996             1             1             0         0              3
59997             0             1             1         0              3
59998             0             1             0         0              3
```

```
       Z_Revenue  Response
0             11         1
1             11         1
2             11         1
3             11         1
4             11         0
…            …        …
59994         11         0
59995         11         0
59996         11         0
59997         11         1
59998         11         0

[59999 rows x 29 columns]
```

```
time: 62 ms (started: 2024-04-06 00:09:07 +05:30)
```

[4]:
```
# Add a new column named "S.no" as the first column with serial numbers
df.insert(0, "S.no", range(1, len(df) + 1))

# Now the DataFrame df_ppd_subset will have a new column "S.no" as the first␣
 ↪column
print(df.head())
```

```
   S.no                                    ID  Year_Birth   Education  \
0     1  e20d011e-c904-4301-871d-4120e734f189        1956    2n Cycle
1     2  9038499f-e82e-43d3-803d-165dba9afe48        1950  Graduation
2     3  a3aaaf2a-c063-45ba-ac1a-ab375d53faa3        1960         PhD
3     4  fabc6e91-6e59-4dd6-8d07-572b7f07b9e4        1980         PhD
4     5  6bf0b99f-dd6d-4964-a3ca-1e98f500d9e1        1977  Graduation

  Marital_Status  Income  Kidhome  Teenhome Dt_Customer  Recency  …  \
0       Together   40831        0         1  21-07-2022       62  …
1       Together   22537        1         1  05-11-2021       42  …
2       Together  195755        1         0  24-12-2015       28  …
3       Divorced  143143        0         0  14-06-2017       24  …
4        Married   84648        0         1  17-06-2021       92  …

   NumWebVisitsMonth  AcceptedCmp3  AcceptedCmp4  AcceptedCmp5  AcceptedCmp1  \
0                  5             1             1             1             1
1                  4             0             1             1             0
2                  4             0             1             0             1
3                  5             1             1             0             0
4                  7             1             1             1             1

   AcceptedCmp2  Complain  Z_CostContact  Z_Revenue  Response
```

```
0             1         0             3         11         1
1             0         0             3         11         1
2             1         0             3         11         1
3             0         0             3         11         1
4             1         0             3         11         0

[5 rows x 30 columns]
time: 0 ns (started: 2024-04-06 00:09:07 +05:30)
```

[5]:
```
df.head()
```

[5]:
```
   S.no                                    ID  Year_Birth   Education  \
0     1  e20d011e-c904-4301-871d-4120e734f189        1956    2n Cycle
1     2  9038499f-e82e-43d3-803d-165dba9afe48        1950  Graduation
2     3  a3aaaf2a-c063-45ba-ac1a-ab375d53faa3        1960         PhD
3     4  fabc6e91-6e59-4dd6-8d07-572b7f07b9e4        1980         PhD
4     5  6bf0b99f-dd6d-4964-a3ca-1e98f500d9e1        1977  Graduation

  Marital_Status  Income  Kidhome  Teenhome Dt_Customer  Recency  …  \
0       Together   40831        0         1  21-07-2022       62  …
1       Together   22537        1         1  05-11-2021       42  …
2       Together  195755        1         0  24-12-2015       28  …
3       Divorced  143143        0         0  14-06-2017       24  …
4        Married   84648        0         1  17-06-2021       92  …

   NumWebVisitsMonth  AcceptedCmp3  AcceptedCmp4  AcceptedCmp5  AcceptedCmp1  \
0                  5             1             1             1             1
1                  4             0             1             1             0
2                  4             0             1             0             1
3                  5             1             1             0             0
4                  7             1             1             1             1

   AcceptedCmp2  Complain  Z_CostContact  Z_Revenue  Response
0             1         0              3         11         1
1             0         0              3         11         1
2             1         0              3         11         1
3             0         0              3         11         1
4             1         0              3         11         0

[5 rows x 30 columns]

time: 0 ns (started: 2024-04-06 00:09:07 +05:30)
```

[6]:
```
df.info()
list(df.columns)

# Assuming df is your original DataFrame
# Add your normalization or standardization code here
```

```python
# Display summary statistics
df.describe()

total_records = len(df)
print(f"Total number of records: {total_records}")

# Calculate the total number of filled cells in each column
filled_cells_count = df.count()

# Sum up the counts to get the total number of filled cells in the DataFrame
total_filled_cells = filled_cells_count.sum()

print(f"Total number of filled cells: {total_filled_cells}")

# Assuming df is your DataFrame
unique_counts = df.nunique()

# Display the number of unique values in each column
print(unique_counts)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 59999 entries, 0 to 59998
Data columns (total 30 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   S.no                 59999 non-null  int64
 1   ID                   59999 non-null  object
 2   Year_Birth           59999 non-null  int64
 3   Education            59999 non-null  object
 4   Marital_Status       59999 non-null  object
 5   Income               59999 non-null  int64
 6   Kidhome              59999 non-null  int64
 7   Teenhome             59999 non-null  int64
 8   Dt_Customer          59999 non-null  object
 9   Recency              59999 non-null  int64
 10  MntWines             59999 non-null  int64
 11  MntFruits            59999 non-null  int64
 12  MntMeatProducts      59999 non-null  int64
 13  MntFishProducts      59999 non-null  int64
 14  MntSweetProducts     59999 non-null  int64
 15  MntGoldProds         59999 non-null  int64
 16  NumDealsPurchases    59999 non-null  int64
 17  NumWebPurchases      59999 non-null  int64
 18  NumCatalogPurchases  59999 non-null  int64
 19  NumStorePurchases    59999 non-null  int64
 20  NumWebVisitsMonth    59999 non-null  int64
```

```
 21  AcceptedCmp3         59999 non-null  int64
 22  AcceptedCmp4         59999 non-null  int64
 23  AcceptedCmp5         59999 non-null  int64
 24  AcceptedCmp1         59999 non-null  int64
 25  AcceptedCmp2         59999 non-null  int64
 26  Complain             59999 non-null  int64
 27  Z_CostContact        59999 non-null  int64
 28  Z_Revenue            59999 non-null  int64
 29  Response             59999 non-null  int64
dtypes: int64(26), object(4)
memory usage: 13.7+ MB
Total number of records: 59999
Total number of filled cells: 1799970
S.no                   59999
ID                     59999
Year_Birth                83
Education                  5
Marital_Status             8
Income                 40412
Kidhome                    3
Teenhome                   3
Dt_Customer             4028
Recency                  104
MntWines                1508
MntFruits                204
MntMeatProducts         1033
MntFishProducts          278
MntSweetProducts         226
MntGoldProds             303
NumDealsPurchases         18
NumWebPurchases           24
NumCatalogPurchases       26
NumStorePurchases         16
NumWebVisitsMonth         23
AcceptedCmp3               2
AcceptedCmp4               2
AcceptedCmp5               2
AcceptedCmp1               2
AcceptedCmp2               2
Complain                   2
Z_CostContact              1
Z_Revenue                  1
Response                   2
dtype: int64
time: 125 ms (started: 2024-04-06 00:09:07 +05:30)
```

[7]:
```
# Assuming df is your DataFrame
columns_list = df.columns.tolist()
columns_list

list(df.columns)
```

[7]:
```
['S.no',
 'ID',
 'Year_Birth',
 'Education',
 'Marital_Status',
 'Income',
 'Kidhome',
 'Teenhome',
 'Dt_Customer',
 'Recency',
 'MntWines',
 'MntFruits',
 'MntMeatProducts',
 'MntFishProducts',
 'MntSweetProducts',
 'MntGoldProds',
 'NumDealsPurchases',
 'NumWebPurchases',
 'NumCatalogPurchases',
 'NumStorePurchases',
 'NumWebVisitsMonth',
 'AcceptedCmp3',
 'AcceptedCmp4',
 'AcceptedCmp5',
 'AcceptedCmp1',
 'AcceptedCmp2',
 'Complain',
 'Z_CostContact',
 'Z_Revenue',
 'Response']
```

```
time: 0 ns (started: 2024-04-06 00:09:07 +05:30)
```

[8]:
```
# Nominal and Ordinal Columns

# Continuous and Non Continuous Columns


import pandas as pd

# Assuming df is your DataFrame
```

```
continuous_columns = df.select_dtypes(include=['float64', 'int64']).columns
non_continuous_columns = df.select_dtypes(exclude=['float64', 'int64']).columns

print("Continuous Columns:", list(continuous_columns))
print("Non-Continuous Columns:", list(non_continuous_columns))


# Assuming df is your DataFrame
categorical_columns = df.select_dtypes(include=['object', 'category']).columns
non_categorical_columns = df.select_dtypes(exclude=['object', 'category']).
 ↪columns

print("Categorical Columns:", list(categorical_columns))
print("Non-Categorical Columns:", list(non_categorical_columns))
```

```
Continuous Columns: ['S.no', 'Year_Birth', 'Income', 'Kidhome', 'Teenhome',
'Recency', 'MntWines', 'MntFruits', 'MntMeatProducts', 'MntFishProducts',
'MntSweetProducts', 'MntGoldProds', 'NumDealsPurchases', 'NumWebPurchases',
'NumCatalogPurchases', 'NumStorePurchases', 'NumWebVisitsMonth', 'AcceptedCmp3',
'AcceptedCmp4', 'AcceptedCmp5', 'AcceptedCmp1', 'AcceptedCmp2', 'Complain',
'Z_CostContact', 'Z_Revenue', 'Response']
Non-Continuous Columns: ['ID', 'Education', 'Marital_Status', 'Dt_Customer']
Categorical Columns: ['ID', 'Education', 'Marital_Status', 'Dt_Customer']
Non-Categorical Columns: ['S.no', 'Year_Birth', 'Income', 'Kidhome', 'Teenhome',
'Recency', 'MntWines', 'MntFruits', 'MntMeatProducts', 'MntFishProducts',
'MntSweetProducts', 'MntGoldProds', 'NumDealsPurchases', 'NumWebPurchases',
'NumCatalogPurchases', 'NumStorePurchases', 'NumWebVisitsMonth', 'AcceptedCmp3',
'AcceptedCmp4', 'AcceptedCmp5', 'AcceptedCmp1', 'AcceptedCmp2', 'Complain',
'Z_CostContact', 'Z_Revenue', 'Response']
time: 47 ms (started: 2024-04-06 00:09:07 +05:30)
```

[9]:
```
###  Missing Data Statistics and Treatment
###  Missing Data Statistics: Records

# Assuming df is your DataFrame

# Count the missing values in each column
missing_data = df.isnull().sum()

# Create a DataFrame to display missing data statistics
missing_data_stats = pd.DataFrame({
    'Column': missing_data.index,
    'Missing Records': missing_data.values,
    'Percentage Missing': (missing_data / len(df)) * 100
})

# Sort the DataFrame by the percentage of missing values in descending order
```

```
missing_data_stats = missing_data_stats.sort_values(by='Percentage Missing',␣
 ↪ascending=False)

# Print the missing data statistics
print(missing_data_stats)
```

	Column	Missing Records	Percentage Missing
S.no	S.no	0	0.0
ID	ID	0	0.0
Z_Revenue	Z_Revenue	0	0.0
Z_CostContact	Z_CostContact	0	0.0
Complain	Complain	0	0.0
AcceptedCmp2	AcceptedCmp2	0	0.0
AcceptedCmp1	AcceptedCmp1	0	0.0
AcceptedCmp5	AcceptedCmp5	0	0.0
AcceptedCmp4	AcceptedCmp4	0	0.0
AcceptedCmp3	AcceptedCmp3	0	0.0
NumWebVisitsMonth	NumWebVisitsMonth	0	0.0
NumStorePurchases	NumStorePurchases	0	0.0
NumCatalogPurchases	NumCatalogPurchases	0	0.0
NumWebPurchases	NumWebPurchases	0	0.0
NumDealsPurchases	NumDealsPurchases	0	0.0
MntGoldProds	MntGoldProds	0	0.0
MntSweetProducts	MntSweetProducts	0	0.0
MntFishProducts	MntFishProducts	0	0.0
MntMeatProducts	MntMeatProducts	0	0.0
MntFruits	MntFruits	0	0.0
MntWines	MntWines	0	0.0
Recency	Recency	0	0.0
Dt_Customer	Dt_Customer	0	0.0
Teenhome	Teenhome	0	0.0
Kidhome	Kidhome	0	0.0
Income	Income	0	0.0
Marital_Status	Marital_Status	0	0.0
Education	Education	0	0.0
Year_Birth	Year_Birth	0	0.0
Response	Response	0	0.0

```
time: 0 ns (started: 2024-04-06 00:09:07 +05:30)
```

[10]:
```
# List of columns to drop
columns_to_drop = ['ID','Dt_Customer']

# Drop columns with more than 50% missing values
df_cleaned = df.drop(columns=columns_to_drop)

# Print the cleaned DataFrame
df1 = df_cleaned
```

```
# Count the missing values in each column
missing_data = df1.isnull().sum()

# Create a DataFrame to display missing data statistics
missing_data_stats = pd.DataFrame({
    'Column': missing_data.index,
    'Missing Records': missing_data.values,
    'Percentage Missing': (missing_data / len(df)) * 100
})
# Sort the DataFrame by the percentage of missing values in descending order
missing_data_stats = missing_data_stats.sort_values(by='Percentage Missing',␣
 ↪ascending=False)

# Print the missing data statistics
print(missing_data_stats)
```

	Column	Missing Records	Percentage Missing
S.no	S.no	0	0.0
Year_Birth	Year_Birth	0	0.0
Z_Revenue	Z_Revenue	0	0.0
Z_CostContact	Z_CostContact	0	0.0
Complain	Complain	0	0.0
AcceptedCmp2	AcceptedCmp2	0	0.0
AcceptedCmp1	AcceptedCmp1	0	0.0
AcceptedCmp5	AcceptedCmp5	0	0.0
AcceptedCmp4	AcceptedCmp4	0	0.0
AcceptedCmp3	AcceptedCmp3	0	0.0
NumWebVisitsMonth	NumWebVisitsMonth	0	0.0
NumStorePurchases	NumStorePurchases	0	0.0
NumCatalogPurchases	NumCatalogPurchases	0	0.0
NumWebPurchases	NumWebPurchases	0	0.0
NumDealsPurchases	NumDealsPurchases	0	0.0
MntGoldProds	MntGoldProds	0	0.0
MntSweetProducts	MntSweetProducts	0	0.0
MntFishProducts	MntFishProducts	0	0.0
MntMeatProducts	MntMeatProducts	0	0.0
MntFruits	MntFruits	0	0.0
MntWines	MntWines	0	0.0
Recency	Recency	0	0.0
Teenhome	Teenhome	0	0.0
Kidhome	Kidhome	0	0.0
Income	Income	0	0.0
Marital_Status	Marital_Status	0	0.0
Education	Education	0	0.0
Response	Response	0	0.0

```
time: 16 ms (started: 2024-04-06 00:09:07 +05:30)
```

[11]:
```
### Missing Records (ROWS)

# Count the missing values in each row
missing_rows = df1.isnull().sum(axis=1)

# Count the number of rows with at least one missing value
num_rows_with_missing = len(missing_rows[missing_rows > 0])

# Print the number of rows with missing data
print("Number of rows with missing data:", num_rows_with_missing)


# Calculate the percentage of missing values in each row
missing_percentage_rows = (df1.isnull().sum(axis=1) / len(df1.columns)) * 100

# Count the number of rows with more than 50% missing data
num_rows_more_than_50_percent_missing =␣
 ↪len(missing_percentage_rows[missing_percentage_rows > 50])

# Print the number of rows with more than 50% missing data
print("Number of rows with more than 50% missing data:",␣
 ↪num_rows_more_than_50_percent_missing)
```

```
Number of rows with missing data: 0
Number of rows with more than 50% missing data: 0
time: 31 ms (started: 2024-04-06 00:09:07 +05:30)
```

[12]:
```
# DIVIDING DF1 into Cat and Non Cat

# Assuming df1 is your DataFrame
cat_columns = df1.select_dtypes(include=['object']).columns
noncat_columns = df1.select_dtypes(exclude=['object']).columns

# Creating categorical and non-categorical DataFrames
catdf1 = df1[cat_columns]
noncatdf1 = df1[noncat_columns]

#print(list(catdf.columns))
#print(list(noncatdf.columns))
print(list(catdf1.columns))
print(list(noncatdf1.columns))

#20
#list(noncatdf.columns)
```

```
['Education', 'Marital_Status']
['S.no', 'Year_Birth', 'Income', 'Kidhome', 'Teenhome', 'Recency', 'MntWines',
```

```
'MntFruits', 'MntMeatProducts', 'MntFishProducts', 'MntSweetProducts',
'MntGoldProds', 'NumDealsPurchases', 'NumWebPurchases', 'NumCatalogPurchases',
'NumStorePurchases', 'NumWebVisitsMonth', 'AcceptedCmp3', 'AcceptedCmp4',
'AcceptedCmp5', 'AcceptedCmp1', 'AcceptedCmp2', 'Complain', 'Z_CostContact',
'Z_Revenue', 'Response']
time: 31 ms (started: 2024-04-06 00:09:07 +05:30)
```

[13]:
```
# Data Bifurcation
catdf = df[['S.no', 'Year_Birth', 'Education', 'Marital_Status', 'Teenhome',␣
 ↪'Response','AcceptedCmp3', 'AcceptedCmp4', 'AcceptedCmp5', 'AcceptedCmp1',␣
 ↪'AcceptedCmp2', 'Complain']] # Categorical Data [Nominal | Ordinal]

noncatdf = df[['S.no', 'Income', 'Kidhome', 'Recency', 'MntWines', 'MntFruits',␣
 ↪'MntMeatProducts', 'MntFishProducts', 'MntSweetProducts', 'MntGoldProds',␣
 ↪'NumDealsPurchases', 'NumWebPurchases', 'NumCatalogPurchases',␣
 ↪'NumStorePurchases', 'NumWebVisitsMonth', 'Z_CostContact', 'Z_Revenue']] #␣
 ↪Non-Categorical Data
```

```
time: 16 ms (started: 2024-04-06 00:09:07 +05:30)
```

[14]:
```
####  STATISTICS OF CAT DATASET

# Count and frequency statistics for each column in catdf
catdf_stats = pd.DataFrame()

for column in catdf.columns:
    col_count = catdf[column].value_counts().reset_index()
    col_count.columns = [column, 'Frequency']
    catdf_stats = pd.concat([catdf_stats, col_count], axis=1)

# Display the count and frequency statistics
#print(catdf_stats)

# Summary for each column in catdf
catdf_summary = catdf.describe(include='all').transpose()

# Display the summary
print(catdf_summary)


# Calculate the proportion (relative frequency) for each categorical column
#proportion_stats = catdf.apply(lambda x: x.value_counts(normalize=True).
 ↪idxmax() + ': ' + "{:.2%}".format(x.value_counts(normalize=True).max()))

# Display the proportion statistics
#print(proportion_stats)
```

```
                count unique         top   freq         mean          std  \
```

```
S.no             59999.0    NaN         NaN    NaN      30000.0  17320.363738
Year_Birth       59999.0    NaN         NaN    NaN  1968.296488     12.318071
Education          59999      5  Graduation  30137          NaN           NaN
Marital_Status     59999      8     Married  22798          NaN           NaN
Teenhome         59999.0    NaN         NaN    NaN     0.508592      0.545526
Response         59999.0    NaN         NaN    NaN     0.500175      0.500004
AcceptedCmp3     59999.0    NaN         NaN    NaN     0.497058      0.499996
AcceptedCmp4     59999.0    NaN         NaN    NaN     0.499442      0.500004
AcceptedCmp5     59999.0    NaN         NaN    NaN     0.499742      0.500004
AcceptedCmp1     59999.0    NaN         NaN    NaN     0.497642      0.499999
AcceptedCmp2     59999.0    NaN         NaN    NaN     0.495175      0.499981
Complain         59999.0    NaN         NaN    NaN       0.0094      0.096498

                    min      25%      50%      75%      max
S.no                1.0  15000.5  30000.0  44999.5  59999.0
Year_Birth       1888.0   1959.0   1969.0   1977.0   2000.0
Education           NaN      NaN      NaN      NaN      NaN
Marital_Status      NaN      NaN      NaN      NaN      NaN
Teenhome            0.0      0.0      0.0      1.0      2.0
Response            0.0      0.0      1.0      1.0      1.0
AcceptedCmp3        0.0      0.0      0.0      1.0      1.0
AcceptedCmp4        0.0      0.0      0.0      1.0      1.0
AcceptedCmp5        0.0      0.0      0.0      1.0      1.0
AcceptedCmp1        0.0      0.0      0.0      1.0      1.0
AcceptedCmp2        0.0      0.0      0.0      1.0      1.0
Complain            0.0      0.0      0.0      0.0      1.0
time: 156 ms (started: 2024-04-06 00:09:07 +05:30)
```

[15]:
```python
#### STATISTICS OF NONCAT DATASET

# Display descriptive statistics for non-categorical variables
noncatdf_descriptive_stats = noncatdf.describe()

# Print the descriptive statistics
print(noncatdf_descriptive_stats)
```

```
                S.no         Income       Kidhome       Recency      MntWines  \
count   59999.000000   59999.000000  59999.000000  59999.000000  59999.000000
mean    30000.000000   81359.282238      0.441457     48.698862    308.099902
std     17320.363738  126077.457422      0.538620     28.975391    333.677239
min         1.000000    1000.000000      0.000000      0.000000      0.000000
25%     15000.500000    3148.000000      0.000000     24.000000     38.000000
50%     30000.000000   43028.000000      0.000000     49.000000    176.000000
75%     44999.500000   90124.000000      1.000000     74.000000    500.000000
max     59999.000000  666480.000000      2.000000    103.000000   1540.000000

           MntFruits  MntMeatProducts  MntFishProducts  MntSweetProducts  \
count   59999.000000     59999.000000     59999.000000      59999.000000
```

```
mean      26.573010       166.544309        40.871165         28.414324
std       39.336046       225.606608        53.099988         40.453819
min        0.000000         0.000000         0.000000          0.000000
25%        3.000000        16.000000         8.000000          5.000000
50%        8.000000        66.000000        18.000000         10.000000
75%       32.000000       232.000000        50.000000         34.000000
max      203.000000      1731.000000       278.000000        271.000000

       MntGoldProds  NumDealsPurchases  NumWebPurchases  NumCatalogPurchases  \
count  59999.000000       59999.000000     59999.000000         59999.000000
mean      43.856964           2.375523         3.831597             2.949816
std       52.005772           2.082430         2.945731             2.731089
min        0.000000           0.000000         0.000000             0.000000
25%        8.000000           1.000000         2.000000             1.000000
50%       24.000000           2.000000         3.000000             2.000000
75%       56.000000           3.000000         6.000000             4.000000
max      368.000000          17.000000        29.000000            30.000000

       NumStorePurchases  NumWebVisitsMonth  Z_CostContact  Z_Revenue
count       59999.000000       59999.000000        59999.0    59999.0
mean            5.341272           4.919599            3.0       11.0
std             3.596753           2.788278            0.0        0.0
min             0.000000           0.000000            3.0       11.0
25%             3.000000           3.000000            3.0       11.0
50%             5.000000           5.000000            3.0       11.0
75%             8.000000           7.000000            3.0       11.0
max            15.000000          22.000000            3.0       11.0
time: 62 ms (started: 2024-04-06 00:09:07 +05:30)
```

[16]:
```
# Missing Data Statistics: Non-Categorical Variables or Features

# Calculate missing data statistics for non-categorical columns
missing_data_non_categorical = noncatdf.isnull().sum().reset_index()
missing_data_non_categorical.columns = ['Feature', 'Missing_Records']

# Display the missing data statistics
print(missing_data_non_categorical)
```

```
              Feature  Missing_Records
0                 S.no                0
1               Income                0
2             Kidhome                 0
3             Recency                 0
4            MntWines                 0
5           MntFruits                 0
6     MntMeatProducts                 0
7     MntFishProducts                 0
8   MntSweetProducts                  0
```

```
9           MntGoldProds                0
10    NumDealsPurchases                 0
11      NumWebPurchases                 0
12  NumCatalogPurchases                 0
13    NumStorePurchases                 0
14    NumWebVisitsMonth                 0
15        Z_CostContact                 0
16            Z_Revenue                 0
time: 0 ns (started: 2024-04-06 00:09:07 +05:30)
```

[17]:
```
#  Missing Data Treatment: Non-Categorical Variables or Features

# Dataset Used : df_noncat

si_noncat = SimpleImputer(missing_values=np.nan, strategy='mean') # Other␣
 ↪Strategy : mean | median | most_frequent | constant
si_noncat_fit = si_noncat.fit_transform(noncatdf)
imputed_data_non_categorical = pd.DataFrame(si_noncat_fit, columns=noncatdf.
 ↪columns); # Missing Non-Categorical Data Imputed Subset using Simple Imputer
imputed_data_non_categorical.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 59999 entries, 0 to 59998
Data columns (total 17 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   S.no                 59999 non-null  float64
 1   Income               59999 non-null  float64
 2   Kidhome              59999 non-null  float64
 3   Recency              59999 non-null  float64
 4   MntWines             59999 non-null  float64
 5   MntFruits            59999 non-null  float64
 6   MntMeatProducts      59999 non-null  float64
 7   MntFishProducts      59999 non-null  float64
 8   MntSweetProducts     59999 non-null  float64
 9   MntGoldProds         59999 non-null  float64
 10  NumDealsPurchases    59999 non-null  float64
 11  NumWebPurchases      59999 non-null  float64
 12  NumCatalogPurchases  59999 non-null  float64
 13  NumStorePurchases    59999 non-null  float64
 14  NumWebVisitsMonth    59999 non-null  float64
 15  Z_CostContact        59999 non-null  float64
 16  Z_Revenue            59999 non-null  float64
dtypes: float64(17)
memory usage: 7.8 MB
time: 47 ms (started: 2024-04-06 00:09:07 +05:30)
```

[18]:
```
# Calculate standard deviation for non-categorical columns
std_deviation_non_categorical = imputed_data_non_categorical.std()

# Creating a DataFrame to display the results
dispersion_non_categorical_df = pd.DataFrame({
    'Variable': imputed_data_non_categorical.columns,
    'Standard Deviation': std_deviation_non_categorical.values
})

print(dispersion_non_categorical_df)
```

```
               Variable  Standard Deviation
0                  S.no        17320.363738
1                Income       126077.457422
2               Kidhome            0.538620
3               Recency           28.975391
4              MntWines          333.677239
5             MntFruits           39.336046
6       MntMeatProducts          225.606608
7       MntFishProducts           53.099988
8      MntSweetProducts           40.453819
9          MntGoldProds           52.005772
10    NumDealsPurchases            2.082430
11      NumWebPurchases            2.945731
12  NumCatalogPurchases            2.731089
13    NumStorePurchases            3.596753
14    NumWebVisitsMonth            2.788278
15        Z_CostContact            0.000000
16            Z_Revenue            0.000000
time: 0 ns (started: 2024-04-06 00:09:08 +05:30)
```

[19]:
```
# Dataset Used : df_cat

si_cat = SimpleImputer(missing_values=np.nan, strategy='most_frequent') #␣
 ↪Strategy = median [When Odd Number of Categories Exists]
si_cat_fit = si_cat.fit_transform(catdf)
imputed_data_categorical = pd.DataFrame(si_cat_fit, columns=catdf.columns); #␣
 ↪Missing Categorical Data Imputed Subset
imputed_data_categorical.info()
imputed_data_categorical.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 59999 entries, 0 to 59998
Data columns (total 12 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   S.no            59999 non-null  object
```

```
 1   Year_Birth      59999 non-null  object
 2   Education       59999 non-null  object
 3   Marital_Status  59999 non-null  object
 4   Teenhome        59999 non-null  object
 5   Response        59999 non-null  object
 6   AcceptedCmp3    59999 non-null  object
 7   AcceptedCmp4    59999 non-null  object
 8   AcceptedCmp5    59999 non-null  object
 9   AcceptedCmp1    59999 non-null  object
 10  AcceptedCmp2    59999 non-null  object
 11  Complain        59999 non-null  object
dtypes: object(12)
memory usage: 5.5+ MB
```

[19]:

	S.no	Year_Birth	Education	Marital_Status	Teenhome	Response	AcceptedCmp3	\
0	1	1956	2n Cycle	Together	1	1	1	
1	2	1950	Graduation	Together	1	1	0	
2	3	1960	PhD	Together	0	1	0	
3	4	1980	PhD	Divorced	0	1	1	
4	5	1977	Graduation	Married	1	0	1	

	AcceptedCmp4	AcceptedCmp5	AcceptedCmp1	AcceptedCmp2	Complain
0	1	1	1	1	0
1	1	1	0	0	0
2	1	0	1	1	0
3	1	0	0	0	0
4	1	1	1	1	0

```
time: 156 ms (started: 2024-04-06 00:09:08 +05:30)
```

[20]:
```
from sklearn.preprocessing import LabelEncoder

# Calculate the number of unique values in each column
unique_values_categorical = imputed_data_categorical.nunique().reset_index()
unique_values_categorical.columns = ['Feature', 'Number_of_Unique_Values']

# Display the number of unique values
print(unique_values_categorical)

# Initialize LabelEncoder
label_encoder = LabelEncoder()

# Create a copy of the imputed_data_categorical dataframe to avoid modifying␣
 ↪the original
encoded_data_categorical = imputed_data_categorical.copy()

# Columns to exclude from encoding
```

```
exclude_columns = ['S.no', 'Response','AcceptedCmp3', 'AcceptedCmp4',␣
 ↪'AcceptedCmp5', 'AcceptedCmp1', 'AcceptedCmp2', 'Complain']  # Add column␣
 ↪names you want to exclude here

# Check for missing values
if encoded_data_categorical.isnull().values.any():
    # Handle missing values before encoding
    encoded_data_categorical = encoded_data_categorical.dropna()  # Example:␣
 ↪Remove rows with missing values
    # Alternatively, you can impute missing values

# Check data types
for column in encoded_data_categorical.columns:
    if column not in exclude_columns:
        if not encoded_data_categorical[column].apply(lambda x: isinstance(x,␣
 ↪(int, float))).all():
            # If the column contains non-numeric values, convert them to strings
            encoded_data_categorical[column] = encoded_data_categorical[column].
 ↪astype(str)

# Initialize LabelEncoder
label_encoder = LabelEncoder()

# Iterate through each column in the dataframe
mapping = {}  # To store the mapping of variable names to numeric representation
for column in encoded_data_categorical.columns:
    if column not in exclude_columns:
        # Perform numerical encoding
        encoded_data_categorical[column] = label_encoder.
 ↪fit_transform(encoded_data_categorical[column])

        # Store the mapping information
        mapping[column] = dict(zip(label_encoder.classes_, label_encoder.
 ↪transform(label_encoder.classes_)))

# Display the mapping
for variable, variable_mapping in mapping.items():
    print(f"\nMapping for {variable}:")
    print(variable_mapping)

# Display the encoded data
print(encoded_data_categorical)
```

```
           Feature  Number_of_Unique_Values
0              S.no                    59999
1       Year_Birth                       83
2        Education                        5
```

```
3   Marital_Status                      8
4         Teenhome                      3
5         Response                      2
6     AcceptedCmp3                      2
7     AcceptedCmp4                      2
8     AcceptedCmp5                      2
9     AcceptedCmp1                      2
10    AcceptedCmp2                      2
11        Complain                      2

Mapping for Year_Birth:
{1888: 0, 1889: 1, 1890: 2, 1891: 3, 1892: 4, 1893: 5, 1894: 6, 1895: 7, 1896:
8, 1897: 9, 1898: 10, 1899: 11, 1900: 12, 1901: 13, 1902: 14, 1903: 15, 1904:
16, 1935: 17, 1936: 18, 1937: 19, 1938: 20, 1939: 21, 1940: 22, 1941: 23, 1942:
24, 1943: 25, 1944: 26, 1945: 27, 1946: 28, 1947: 29, 1948: 30, 1949: 31, 1950:
32, 1951: 33, 1952: 34, 1953: 35, 1954: 36, 1955: 37, 1956: 38, 1957: 39, 1958:
40, 1959: 41, 1960: 42, 1961: 43, 1962: 44, 1963: 45, 1964: 46, 1965: 47, 1966:
48, 1967: 49, 1968: 50, 1969: 51, 1970: 52, 1971: 53, 1972: 54, 1973: 55, 1974:
56, 1975: 57, 1976: 58, 1977: 59, 1978: 60, 1979: 61, 1980: 62, 1981: 63, 1982:
64, 1983: 65, 1984: 66, 1985: 67, 1986: 68, 1987: 69, 1988: 70, 1989: 71, 1990:
72, 1991: 73, 1992: 74, 1993: 75, 1994: 76, 1995: 77, 1996: 78, 1997: 79, 1998:
80, 1999: 81, 2000: 82}

Mapping for Education:
{'2n Cycle': 0, 'Basic': 1, 'Graduation': 2, 'Master': 3, 'PhD': 4}

Mapping for Marital_Status:
{'Absurd': 0, 'Alone': 1, 'Divorced': 2, 'Married': 3, 'Single': 4, 'Together':
5, 'Widow': 6, 'YOLO': 7}

Mapping for Teenhome:
{0: 0, 1: 1, 2: 2}
        S.no  Year_Birth  Education  Marital_Status  Teenhome Response  \
0          1          38          0               5         1        1
1          2          32          2               5         1        1
2          3          42          4               5         0        1
3          4          62          4               2         0        1
4          5          59          2               3         1        0
…      …          …         …              …        …       …
59994  59995          49          4               3         1        0
59995  59996          56          2               5         0        0
59996  59997          45          3               4         1        0
59997  59998          56          3               4         0        1
59998  59999          33          2               3         0        0

      AcceptedCmp3 AcceptedCmp4 AcceptedCmp5 AcceptedCmp1 AcceptedCmp2  \
0                1            1            1            1            1
1                0            1            1            0            0
```

```
2                0            1            0            1            1
3                1            1            0            0            0
4                1            1            1            1            1
…              …            …            …            …            …
59994            1            1            1            1            0
59995            0            1            1            1            0
59996            0            0            1            1            0
59997            0            1            0            1            1
59998            1            1            0            1            0

      Complain
0            0
1            0
2            0
3            0
4            0
…          …
59994        0
59995        0
59996        0
59997        0
59998        0

[59999 rows x 12 columns]
time: 125 ms (started: 2024-04-06 00:09:08 +05:30)
```

[21]:
```
print(imputed_data_non_categorical.columns)
```

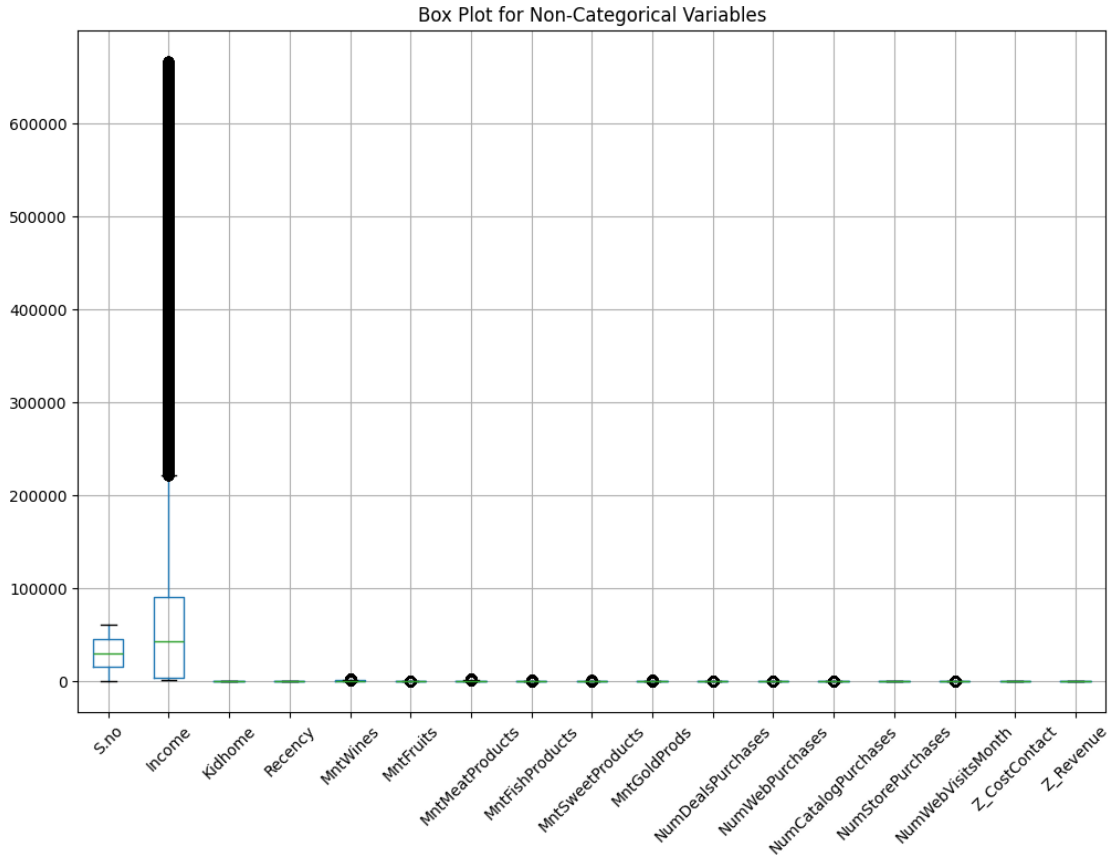
```
Index(['S.no', 'Income', 'Kidhome', 'Recency', 'MntWines', 'MntFruits',
       'MntMeatProducts', 'MntFishProducts', 'MntSweetProducts',
       'MntGoldProds', 'NumDealsPurchases', 'NumWebPurchases',
       'NumCatalogPurchases', 'NumStorePurchases', 'NumWebVisitsMonth',
       'Z_CostContact', 'Z_Revenue'],
      dtype='object')
time: 16 ms (started: 2024-04-06 00:09:08 +05:30)
```

[22]:
```
def identify_outliers(column):
    Q1 = np.percentile(column, 25)
    Q3 = np.percentile(column, 75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    outliers = (column < lower_bound) | (column > upper_bound)
    return outliers

# Apply the function to each column to get a DataFrame of True/False values
outliers = imputed_data_non_categorical.apply(identify_outliers)
```

```
# Display the number of outliers for each column
outlier_counts = outliers.sum()
print(outlier_counts)
```

```
S.no                    0
Income                5245
Kidhome                  0
Recency                  0
MntWines              1219
MntFruits             6872
MntMeatProducts       4700
MntFishProducts       6589
MntSweetProducts      7070
MntGoldProds          5391
NumDealsPurchases     2560
NumWebPurchases        304
NumCatalogPurchases   2813
NumStorePurchases        0
NumWebVisitsMonth      189
Z_CostContact            0
Z_Revenue                0
dtype: int64
time: 47 ms (started: 2024-04-06 00:09:08 +05:30)
```

[23]:
```
# Create a box plot for non-categorical variables
imputed_data_non_categorical.boxplot(rot=45, figsize=(12, 8))
plt.title('Box Plot for Non-Categorical Variables')
plt.show()
```



time: 797 ms (started: 2024-04-06 00:09:08 +05:30)

[24]:
```
# Iterate through each column and print count of unique values
for column in imputed_data_non_categorical.columns:
    unique_count = imputed_data_non_categorical[column].nunique()
    print(f"Count of unique values in {column} column: {unique_count}")
```

```
Count of unique values in S.no column: 59999
Count of unique values in Income column: 40412
Count of unique values in Kidhome column: 3
Count of unique values in Recency column: 104
Count of unique values in MntWines column: 1508
Count of unique values in MntFruits column: 204
Count of unique values in MntMeatProducts column: 1033
Count of unique values in MntFishProducts column: 278
Count of unique values in MntSweetProducts column: 226
Count of unique values in MntGoldProds column: 303
Count of unique values in NumDealsPurchases column: 18
Count of unique values in NumWebPurchases column: 24
Count of unique values in NumCatalogPurchases column: 26
```

```
Count of unique values in NumStorePurchases column: 16
Count of unique values in NumWebVisitsMonth column: 23
Count of unique values in Z_CostContact column: 1
Count of unique values in Z_Revenue column: 1
time: 31 ms (started: 2024-04-06 00:09:09 +05:30)
```

[25]:
```
# Initialize the StandardScaler
scaler = StandardScaler()

# Apply Standard Scaling to your dataset
scaled_data = scaler.fit_transform(imputed_data_non_categorical)

def identify_outliers(column):
    Q1 = np.percentile(column, 25)
    Q3 = np.percentile(column, 75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    outliers = (column < lower_bound) | (column > upper_bound)
    return outliers

# Apply the function to each column in the scaled dataset
outliers_scaled = pd.DataFrame(scaled_data,␣
 ↪columns=imputed_data_non_categorical.columns).apply(identify_outliers)

# Display the number of outliers for each column in the scaled dataset
outlier_counts_scaled = outliers_scaled.sum()
print(outlier_counts_scaled)
```

```
S.no                    0
Income                5245
Kidhome                  0
Recency                  0
MntWines              1219
MntFruits             6872
MntMeatProducts       4700
MntFishProducts       6589
MntSweetProducts      7070
MntGoldProds          5391
NumDealsPurchases     2560
NumWebPurchases        304
NumCatalogPurchases   2813
NumStorePurchases        0
NumWebVisitsMonth      189
Z_CostContact            0
Z_Revenue                0
dtype: int64
time: 79 ms (started: 2024-04-06 00:09:09 +05:30)
```

[26]:
```
# Initialize the RobustScaler
scaler = RobustScaler()

# Apply Robust Scaling to your dataset
scaled_data_robust = scaler.fit_transform(imputed_data_non_categorical)

# Check for outliers in the scaled dataset
outliers_robust = pd.DataFrame(scaled_data_robust,␣
 ↪columns=imputed_data_non_categorical.columns).apply(identify_outliers)

# Display the number of outliers for each column in the scaled dataset
outlier_counts_robust = outliers_robust.sum()
print(outlier_counts_robust)
```

```
S.no                    0
Income                5245
Kidhome                  0
Recency                  0
MntWines              1219
MntFruits             6872
MntMeatProducts       4700
MntFishProducts       6589
MntSweetProducts      7070
MntGoldProds          5391
NumDealsPurchases     2560
NumWebPurchases        304
NumCatalogPurchases   2813
NumStorePurchases        0
NumWebVisitsMonth      189
Z_CostContact            0
Z_Revenue                0
dtype: int64
time: 62 ms (started: 2024-04-06 00:09:09 +05:30)
```

[27]:
```
# Define columns to exclude from normalization
columns_to_exclude = ['S.no', 'Kidhome', 'Recency', 'MntWines', 'MntFruits',␣
 ↪'MntMeatProducts', 'MntFishProducts',
                       'MntSweetProducts', 'MntGoldProds', 'NumDealsPurchases',␣
 ↪'NumWebPurchases', 'NumCatalogPurchases', 'NumStorePurchases',
                       'NumWebVisitsMonth','Z_CostContact', 'Z_Revenue']

# Create a copy of the DataFrame with excluded columns
data_to_scale = imputed_data_non_categorical.drop(columns=columns_to_exclude)

# Initialize the MinMaxScaler
scaler = MinMaxScaler()
```

```
# Apply Min-Max Scaling to the selected columns
scaled_data = scaler.fit_transform(data_to_scale)

# Create a DataFrame with scaled data and original column names
scaled_df = pd.DataFrame(scaled_data, columns=data_to_scale.columns)

# Add back the excluded columns to the scaled DataFrame
scaled_df[columns_to_exclude] = imputed_data_non_categorical[columns_to_exclude]

# Display the scaled DataFrame
print(scaled_df)
```

```
         Income     S.no  Kidhome  Recency  MntWines  MntFruits  \
0      0.059853      1.0      0.0     62.0     636.0       84.0
1      0.032363      2.0      1.0     42.0       6.0        2.0
2      0.292653      3.0      1.0     28.0     383.0       51.0
3      0.213595      4.0      0.0     24.0      29.0        0.0
4      0.125696      5.0      0.0     92.0     133.0       41.0
…           …        …        …        …         …          …
59994  0.002537  59995.0      0.0     30.0      13.0        3.0
59995  0.020402  59996.0      1.0     43.0      10.0       11.0
59996  0.003429  59997.0      1.0     80.0      33.0        6.0
59997  0.102101  59998.0      1.0     11.0      37.0        4.0
59998  0.196344  59999.0      0.0     50.0      19.0        4.0

       MntMeatProducts  MntFishProducts  MntSweetProducts  MntGoldProds  \
0                549.0            176.0              86.0          93.0
1                  8.0             18.0               4.0          11.0
2                130.0            124.0              18.0          37.0
3                 20.0              1.0               6.0           1.0
4                124.0             28.0              21.0           9.0
…                    …                …                 …             …
59994             14.0              5.0               1.0           8.0
59995              4.0             20.0               8.0          12.0
59996             17.0              7.0              11.0          29.0
59997             57.0             13.0               3.0          18.0
59998              9.0             14.0               5.0           6.0

       NumDealsPurchases  NumWebPurchases  NumCatalogPurchases  \
0                    2.0              6.0                 11.0
1                    3.0              2.0                  2.0
2                    0.0              9.0                  2.0
3                    1.0              2.0                  2.0
4                    3.0              3.0                  1.0
…                      …                …                    …
59994                0.0              0.0                  1.0
59995                2.0              2.0                  2.0
```

```
59996                2.0              3.0                  1.0
59997                1.0              6.0                  0.0
59998                2.0              2.0                  1.0

       NumStorePurchases  NumWebVisitsMonth  Z_CostContact  Z_Revenue
0                    1.0                5.0            3.0       11.0
1                    3.0                4.0            3.0       11.0
2                    9.0                4.0            3.0       11.0
3                    5.0                5.0            3.0       11.0
4                    5.0                7.0            3.0       11.0
…                    …                  …              …         …
59994                2.0                7.0            3.0       11.0
59995                1.0                4.0            3.0       11.0
59996                2.0                7.0            3.0       11.0
59997                1.0                6.0            3.0       11.0
59998                6.0                1.0            3.0       11.0

[59999 rows x 17 columns]
time: 31 ms (started: 2024-04-06 00:09:09 +05:30)
```

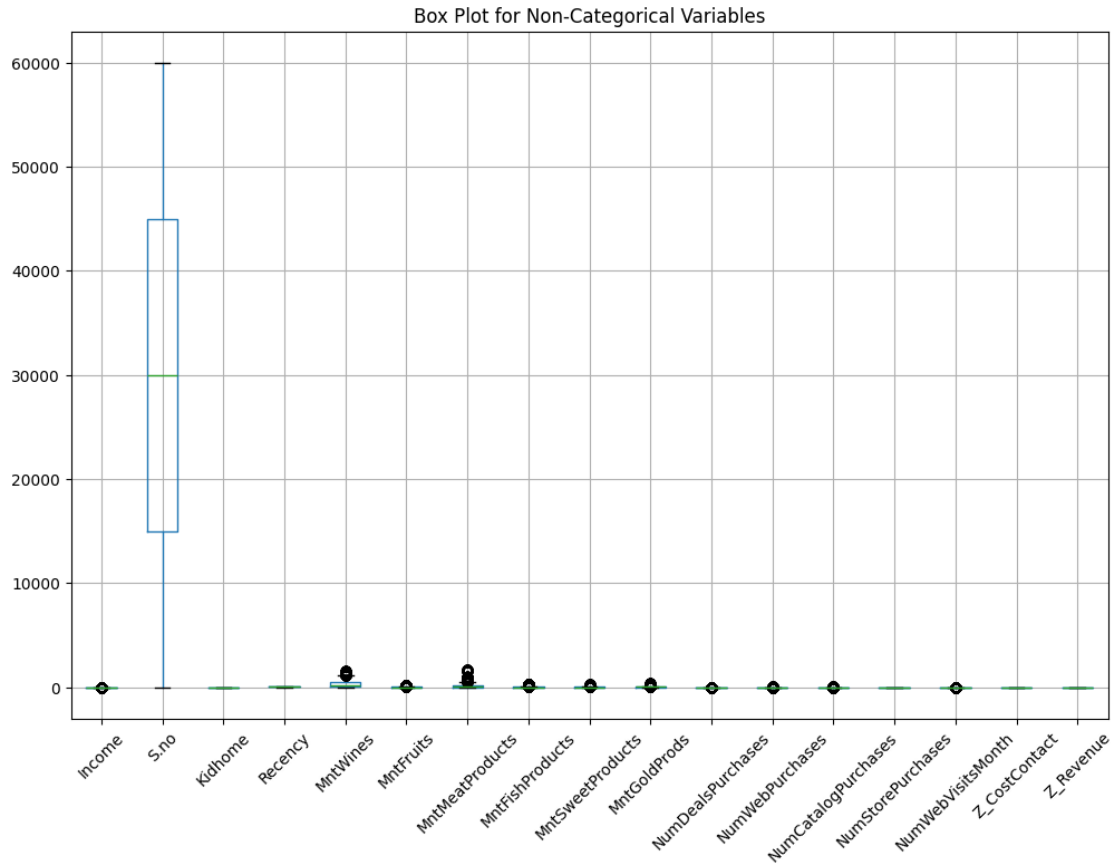
[28]:
```python
def identify_outliers(column):
    Q1 = np.percentile(column, 25)
    Q3 = np.percentile(column, 75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    outliers = (column < lower_bound) | (column > upper_bound)
    return outliers

# Apply the function to each column in the scaled dataset
scaled_outliers = scaled_df.apply(identify_outliers)

# Display the number of outliers for each column in the scaled dataset
scaled_outlier_counts = scaled_outliers.sum()
print(scaled_outlier_counts)
```

```
Income               5245
S.no                    0
Kidhome                 0
Recency                 0
MntWines             1219
MntFruits            6872
MntMeatProducts      4700
MntFishProducts      6589
MntSweetProducts     7070
MntGoldProds         5391
NumDealsPurchases    2560
NumWebPurchases       304
```

```
NumCatalogPurchases    2813
NumStorePurchases         0
NumWebVisitsMonth       189
Z_CostContact             0
Z_Revenue                 0
dtype: int64
time: 31 ms (started: 2024-04-06 00:09:09 +05:30)
```

[29]:
```python
def identify_outliers(column):
    Q1 = np.percentile(column, 25)
    Q3 = np.percentile(column, 75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    outliers1 = (column < lower_bound) | (column > upper_bound)
    return outliers1

# Apply the function to each column to get a DataFrame of True/False values
outliers1 = scaled_df.apply(identify_outliers)

# Display the number of outliers for each column
outlier_counts1 = outliers1.sum()
print(outlier_counts1)


# Create a box plot for non-categorical variables
scaled_df.boxplot(rot=45, figsize=(12, 8))
plt.title('Box Plot for Non-Categorical Variables')
plt.show()
```

```
Income                 5245
S.no                      0
Kidhome                   0
Recency                   0
MntWines               1219
MntFruits              6872
MntMeatProducts        4700
MntFishProducts        6589
MntSweetProducts       7070
MntGoldProds           5391
NumDealsPurchases      2560
NumWebPurchases         304
NumCatalogPurchases    2813
NumStorePurchases         0
NumWebVisitsMonth       189
Z_CostContact             0
Z_Revenue                 0
dtype: int64
```



time: 750 ms (started: 2024-04-06 00:09:09 +05:30)

[30]:
```
# Calculate standard deviation for non-categorical columns
std_deviation_non_categorical1 = scaled_df.std()

# Creating a DataFrame to display the results
dispersion_non_categorical_df1 = pd.DataFrame({
    'Variable': scaled_df.columns,
    'Standard Deviation': std_deviation_non_categorical1.values
})

print(dispersion_non_categorical_df1)
```

```
              Variable  Standard Deviation
0               Income            0.189453
1                 S.no        17320.363738
2             Kidhome             0.538620
3              Recency            28.975391
4             MntWines           333.677239
5            MntFruits            39.336046
```

```
6       MntMeatProducts          225.606608
7       MntFishProducts           53.099988
8      MntSweetProducts           40.453819
9          MntGoldProds           52.005772
10    NumDealsPurchases            2.082430
11      NumWebPurchases            2.945731
12  NumCatalogPurchases            2.731089
13    NumStorePurchases            3.596753
14    NumWebVisitsMonth            2.788278
15        Z_CostContact            0.000000
16            Z_Revenue            0.000000
time: 15 ms (started: 2024-04-06 00:09:10 +05:30)
```

[31]:
```
scaled_df
```

[31]:

	Income	S.no	Kidhome	Recency	MntWines	MntFruits	\
0	0.059853	1.0	0.0	62.0	636.0	84.0	
1	0.032363	2.0	1.0	42.0	6.0	2.0	
2	0.292653	3.0	1.0	28.0	383.0	51.0	
3	0.213595	4.0	0.0	24.0	29.0	0.0	
4	0.125696	5.0	0.0	92.0	133.0	41.0	
…	…	…	…	…	…	…	
59994	0.002537	59995.0	0.0	30.0	13.0	3.0	
59995	0.020402	59996.0	1.0	43.0	10.0	11.0	
59996	0.003429	59997.0	1.0	80.0	33.0	6.0	
59997	0.102101	59998.0	1.0	11.0	37.0	4.0	
59998	0.196344	59999.0	0.0	50.0	19.0	4.0	

	MntMeatProducts	MntFishProducts	MntSweetProducts	MntGoldProds	\
0	549.0	176.0	86.0	93.0	
1	8.0	18.0	4.0	11.0	
2	130.0	124.0	18.0	37.0	
3	20.0	1.0	6.0	1.0	
4	124.0	28.0	21.0	9.0	
…	…	…	…	…	
59994	14.0	5.0	1.0	8.0	
59995	4.0	20.0	8.0	12.0	
59996	17.0	7.0	11.0	29.0	
59997	57.0	13.0	3.0	18.0	
59998	9.0	14.0	5.0	6.0	

	NumDealsPurchases	NumWebPurchases	NumCatalogPurchases	\
0	2.0	6.0	11.0	
1	3.0	2.0	2.0	
2	0.0	9.0	2.0	
3	1.0	2.0	2.0	
4	3.0	3.0	1.0	

```
…                  …               …                    …
59994                0.0             0.0                  1.0
59995                2.0             2.0                  2.0
59996                2.0             3.0                  1.0
59997                1.0             6.0                  0.0
59998                2.0             2.0                  1.0

       NumStorePurchases  NumWebVisitsMonth  Z_CostContact  Z_Revenue
0                    1.0                5.0            3.0       11.0
1                    3.0                4.0            3.0       11.0
2                    9.0                4.0            3.0       11.0
3                    5.0                5.0            3.0       11.0
4                    5.0                7.0            3.0       11.0
…                    …                  …              …         …
59994                2.0                7.0            3.0       11.0
59995                1.0                4.0            3.0       11.0
59996                2.0                7.0            3.0       11.0
59997                1.0                6.0            3.0       11.0
59998                6.0                1.0            3.0       11.0

[59999 rows x 17 columns]

time: 16 ms (started: 2024-04-06 00:09:10 +05:30)
```

[32]:
```
# Pre-Processed Dataset
combined_data = pd.merge(encoded_data_categorical, scaled_df, on='S.no')

# Display the Pre-Processed Dataset
%memit
print(combined_data)

list(combined_data.columns)
```

```
peak memory: 329.25 MiB, increment: 0.09 MiB
        S.no  Year_Birth  Education  Marital_Status  Teenhome Response  \
0          1          38          0               5         1        1
1          2          32          2               5         1        1
2          3          42          4               5         0        1
3          4          62          4               2         0        1
4          5          59          2               3         1        0
…      …          …         …              …       …     …
59994  59995          49          4               3         1        0
59995  59996          56          2               5         0        0
59996  59997          45          3               4         1        0
59997  59998          56          3               4         0        1
59998  59999          33          2               3         0        0

      AcceptedCmp3 AcceptedCmp4 AcceptedCmp5 AcceptedCmp1  …  \
```

```
0                1            1            1            1  …
1                0            1            1            0  …
2                0            1            0            1  …
3                1            1            0            0  …
4                1            1            1            1  …
…              …            …            …          …  …
59994            1            1            1            1  …
59995            0            1            1            1  …
59996            0            0            1            1  …
59997            0            1            0            1  …
59998            1            1            0            1  …

      MntFishProducts MntSweetProducts  MntGoldProds  NumDealsPurchases  \
0               176.0             86.0          93.0                2.0
1                18.0              4.0          11.0                3.0
2               124.0             18.0          37.0                0.0
3                 1.0              6.0           1.0                1.0
4                28.0             21.0           9.0                3.0
…                …                …            …                …
59994             5.0              1.0           8.0                0.0
59995            20.0              8.0          12.0                2.0
59996             7.0             11.0          29.0                2.0
59997            13.0              3.0          18.0                1.0
59998            14.0              5.0           6.0                2.0

       NumWebPurchases  NumCatalogPurchases  NumStorePurchases  \
0                  6.0                 11.0                1.0
1                  2.0                  2.0                3.0
2                  9.0                  2.0                9.0
3                  2.0                  2.0                5.0
4                  3.0                  1.0                5.0
…                …                    …                …
59994              0.0                  1.0                2.0
59995              2.0                  2.0                1.0
59996              3.0                  1.0                2.0
59997              6.0                  0.0                1.0
59998              2.0                  1.0                6.0

       NumWebVisitsMonth  Z_CostContact  Z_Revenue
0                    5.0            3.0       11.0
1                    4.0            3.0       11.0
2                    4.0            3.0       11.0
3                    5.0            3.0       11.0
4                    7.0            3.0       11.0
…                  …              …         …
59994                7.0            3.0       11.0
59995                4.0            3.0       11.0
59996                7.0            3.0       11.0
```

```
59997                6.0            3.0       11.0
59998                1.0            3.0       11.0

[59999 rows x 28 columns]
```

[32]:
```
['S.no',
 'Year_Birth',
 'Education',
 'Marital_Status',
 'Teenhome',
 'Response',
 'AcceptedCmp3',
 'AcceptedCmp4',
 'AcceptedCmp5',
 'AcceptedCmp1',
 'AcceptedCmp2',
 'Complain',
 'Income',
 'Kidhome',
 'Recency',
 'MntWines',
 'MntFruits',
 'MntMeatProducts',
 'MntFishProducts',
 'MntSweetProducts',
 'MntGoldProds',
 'NumDealsPurchases',
 'NumWebPurchases',
 'NumCatalogPurchases',
 'NumStorePurchases',
 'NumWebVisitsMonth',
 'Z_CostContact',
 'Z_Revenue']
```

```
time: 1.05 s (started: 2024-04-06 00:09:10 +05:30)
```

[33]:
```
df_ppd_subset = combined_data.copy()
```

```
time: 0 ns (started: 2024-04-06 00:09:11 +05:30)
```

[ ]:

[ ]:

[ ]:

[ ]:

DT

[ ]:

[54]:
```
dff = df_ppd_subset[['S.no', 'Cluster_Label']]
dff
```

[54]:

	S.no	Cluster_Label
0	1	0
1	2	0
2	3	0
3	4	0
4	5	0
…	…	…
59994	59995	1
59995	59996	1
59996	59997	1
59997	59998	1
59998	59999	1

[59999 rows x 2 columns]

time: 15 ms (started: 2024-04-06 00:12:27 +05:30)

[60]:
```
df1 = pd.merge(df_ppd_subset, dff, on='S.no', how='left')

# Display the merged DataFrame
df1
```

[60]:

	S.no	Year_Birth	Education	Marital_Status	Teenhome	Response	\
0	1	38	0	5	1	1	
1	2	32	2	5	1	1	
2	3	42	4	5	0	1	
3	4	62	4	2	0	1	
4	5	59	2	3	1	0	
…	…	…	…	…	…	…	
59994	59995	49	4	3	1	0	
59995	59996	56	2	5	0	0	
59996	59997	45	3	4	1	0	
59997	59998	56	3	4	0	1	
59998	59999	33	2	3	0	0	

	AcceptedCmp3	AcceptedCmp4	AcceptedCmp5	AcceptedCmp1	…	MntGoldProds	\
0	1	1	1	1	…	93.0	
1	0	1	1	0	…	11.0	
2	0	1	0	1	…	37.0	
3	1	1	0	0	…	1.0	
4	1	1	1	1	…	9.0	

```
…            …          …          …          …  …          …
59994          1          1          1          1  …        8.0
59995          0          1          1          1  …       12.0
59996          0          0          1          1  …       29.0
59997          0          1          0          1  …       18.0
59998          1          1          0          1  …        6.0

      NumDealsPurchases  NumWebPurchases  NumCatalogPurchases  \
0                   2.0              6.0                 11.0
1                   3.0              2.0                  2.0
2                   0.0              9.0                  2.0
3                   1.0              2.0                  2.0
4                   3.0              3.0                  1.0
…                   …                …                    …
59994               0.0              0.0                  1.0
59995               2.0              2.0                  2.0
59996               2.0              3.0                  1.0
59997               1.0              6.0                  0.0
59998               2.0              2.0                  1.0

       NumStorePurchases  NumWebVisitsMonth  Z_CostContact  Z_Revenue  \
0                    1.0                5.0            3.0       11.0
1                    3.0                4.0            3.0       11.0
2                    9.0                4.0            3.0       11.0
3                    5.0                5.0            3.0       11.0
4                    5.0                7.0            3.0       11.0
…                    …                  …              …         …
59994                2.0                7.0            3.0       11.0
59995                1.0                4.0            3.0       11.0
59996                2.0                7.0            3.0       11.0
59997                1.0                6.0            3.0       11.0
59998                6.0                1.0            3.0       11.0

       Cluster_Label_x  Cluster_Label_y
0                    0                0
1                    0                0
2                    0                0
3                    0                0
4                    0                0
…                    …                …
59994                1                1
59995                1                1
59996                1                1
59997                1                1
59998                1                1

[59999 rows x 30 columns]
```

time: 63 ms (started: 2024-04-06 00:21:36 +05:30)

[61]:
```
# Import
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split, StratifiedShuffleSplit
from sklearn.tree import DecisionTreeClassifier, export_text, plot_tree  # For␣
 ↪Decision Tree Model
from sklearn.metrics import accuracy_score, classification_report,␣
 ↪confusion_matrix
from sklearn.metrics import confusion_matrix, classification_report # For␣
 ↪Decision Tree Model Evaluation
from sklearn.neighbors import KNeighborsClassifier
from sklearn.decomposition import PCA
from matplotlib.colors import ListedColormap
from sklearn.svm import SVC
from sklearn.metrics import confusion_matrix, accuracy_score
from matplotlib.colors import ListedColormap
```

time: 0 ns (started: 2024-04-06 00:21:37 +05:30)

[62]:
```
df1.columns
```

[62]:
```
Index(['S.no', 'Year_Birth', 'Education', 'Marital_Status', 'Teenhome',
       'Response', 'AcceptedCmp3', 'AcceptedCmp4', 'AcceptedCmp5',
       'AcceptedCmp1', 'AcceptedCmp2', 'Complain', 'Income', 'Kidhome',
       'Recency', 'MntWines', 'MntFruits', 'MntMeatProducts',
       'MntFishProducts', 'MntSweetProducts', 'MntGoldProds',
       'NumDealsPurchases', 'NumWebPurchases', 'NumCatalogPurchases',
       'NumStorePurchases', 'NumWebVisitsMonth', 'Z_CostContact', 'Z_Revenue',
       'Cluster_Label_x', 'Cluster_Label_y'],
      dtype='object')
```

time: 0 ns (started: 2024-04-06 00:21:38 +05:30)

[64]:
```
df1_inputs = df1[['Income','NumWebVisitsMonth']]; df1_inputs
df1_output = df1[['Cluster_Label_x']]; df1_output

df1_inputs_names = df1_inputs.columns; df1_inputs_names
df1_output_labels = df1_output['Cluster_Label_x'].unique().astype(str);␣
 ↪df1_output_labels
```

[64]:
```
array(['0', '2', '1'], dtype='<U11')
```

time: 0 ns (started: 2024-04-06 00:22:09 +05:30)

[65]:
```
# Initialize StratifiedShuffleSplit with desired test size and random state
stratified_split = StratifiedShuffleSplit(n_splits=1, test_size=0.2,␣
↪random_state=45007)

# Perform the stratified split to get training and testing indices
for train_index, test_index in stratified_split.split(df1_inputs, df1_output):
    df1_inputs_train, df1_inputs_test = df1_inputs.iloc[train_index],␣
↪df1_inputs.iloc[test_index]
    df1_output_train, df1_output_test = df1_output.iloc[train_index],␣
↪df1_output.iloc[test_index]
```

```
time: 141 ms (started: 2024-04-06 00:22:32 +05:30)
```

[66]:
```
from sklearn.linear_model import LogisticRegression
from sklearn.feature_selection import SelectFromModel
import numpy as np

# Initialize Logistic Regression model with L1 regularization
logreg_l1 = LogisticRegression(penalty='l1', solver='liblinear',␣
↪random_state=45007)

# Fit the model on the training data
logreg_l1.fit(df1_inputs_train, df1_output_train.values.ravel())

# Get feature importances from the fitted model
feature_importances = np.abs(logreg_l1.coef_).flatten()

# Calculate the threshold as 20% of the maximum feature importance
threshold = 0.2 * np.max(feature_importances)

# Create a selector object to select features based on non-zero coefficients
selector = SelectFromModel(logreg_l1, threshold=threshold)

# Transform the training and testing input data to select features
df1_inputs_train_selected = selector.transform(df1_inputs_train)
df1_inputs_test_selected = selector.transform(df1_inputs_test)

# Get the selected features
selected_features = df1_inputs_names[selector.get_support()]

# Print the selected features and the calculated threshold
print("Selected Features:", selected_features)
print("Threshold:", threshold)
```

```
Selected Features: Index(['Income'], dtype='object')
Threshold: 0.015567165959656204
time: 125 ms (started: 2024-04-06 00:22:42 +05:30)
```

[67]:
```
# Decision Tree : Model (Training Subset)
dtc = DecisionTreeClassifier(criterion='gini', random_state=45007) # Other␣
 ↪Criteria : Entropy,  Log Loss
dtc_model = dtc.fit(df1_inputs_train, df1_output_train); dtc_model
```

[67]:
```
DecisionTreeClassifier(random_state=45007)
```

```
time: 422 ms (started: 2024-04-06 00:22:54 +05:30)
```

[68]:
```
# Decision Tree : Model Rules
dtc_model_rules = export_text(dtc_model, feature_names =␣
 ↪list(df1_inputs_names)); print(dtc_model_rules)
```

```
|--- NumWebVisitsMonth <= 6.50
|   |--- Income <= 0.95
|   |   |--- Income <= 0.89
|   |   |   |--- Income <= 0.89
|   |   |   |   |--- Income <= 0.89
|   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |--- Income <= 0.11
|   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 15
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 27
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 30
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 5.00
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 1.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  1.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  5.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 10
```

```
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 15
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 10
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 5.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 9
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  5.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.05
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 63
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.05
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 56
|   |   |   |   |   |   |   |--- Income >  0.11
|   |   |   |   |   |   |   |   |--- Income <= 0.11
|   |   |   |   |   |   |   |   |   |--- Income <= 0.11
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.11
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.11
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.11
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.11
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 0.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.11
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.11
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  0.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 15
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 5.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  5.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |--- Income >  0.12
```

```
|   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 5.50
|   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 13
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 20
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 19
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 15
|   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 11
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  0.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 16
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 19
|   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 73
|   |   |   |   |   |   |   |--- NumWebVisitsMonth >  5.50
|   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
```

```
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 13
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 12
|   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.12
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |--- Income >  0.12
|   |   |   |   |   |   |   |   |   |--- Income <= 0.86
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 49
|   |   |   |   |   |   |   |   |   |--- Income >  0.86
|   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 10
|   |   |   |   |--- Income >  0.89
|   |   |   |   |   |--- class: 2
|   |   |   |--- Income >  0.89
|   |   |   |   |--- Income <= 0.89
|   |   |   |   |   |--- class: 1
|   |   |   |   |--- Income >  0.89
|   |   |   |   |   |--- NumWebVisitsMonth <= 5.50
|   |   |   |   |   |   |--- Income <= 0.89
|   |   |   |   |   |   |   |--- Income <= 0.89
|   |   |   |   |   |   |   |   |--- Income <= 0.89
|   |   |   |   |   |   |   |   |   |--- class: 0
```

```
|   |   |   |   |   |   |   |   |--- Income >  0.89
|   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |--- Income >  0.89
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- Income >  0.89
|   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |--- NumWebVisitsMonth >  5.50
|   |   |   |   |   |   |   |--- class: 1
|   |   |   |--- Income >  0.89
|   |   |   |   |--- Income <= 0.89
|   |   |   |   |   |--- class: 2
|   |   |   |   |--- Income >  0.89
|   |   |   |   |   |--- Income <= 0.89
|   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |--- Income >  0.89
|   |   |   |   |   |   |--- Income <= 0.95
|   |   |   |   |   |   |   |--- Income <= 0.95
|   |   |   |   |   |   |   |   |--- Income <= 0.94
|   |   |   |   |   |   |   |   |   |--- Income <= 0.89
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 4.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  4.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.89
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.89
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.89
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.91
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.90
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.90
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 14
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.91
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.91
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.91
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 14
|   |   |   |   |   |   |   |   |--- Income >  0.94
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |--- Income >  0.95
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 1.50
|   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  1.50
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |--- Income >  0.95
|   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 4.50
|   |   |   |   |   |   |   |   |--- Income <= 0.95
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 1.50
```

```
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  1.50
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 2.50
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  2.50
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 3.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  3.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |--- Income >  0.95
|   |   |   |   |   |   |   |   |   |--- Income <= 0.95
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.95
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.95
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.95
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.95
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.95
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |--- NumWebVisitsMonth >  4.50
|   |   |   |   |   |   |   |   |--- Income <= 0.95
|   |   |   |   |   |   |   |   |   |--- Income <= 0.95
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.95
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.95
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.95
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.95
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.95
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |--- Income >  0.95
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |--- Income >  0.95
|   |   |   |--- Income <= 0.96
|   |   |   |   |--- NumWebVisitsMonth <= 2.50
|   |   |   |   |   |--- Income <= 0.96
|   |   |   |   |   |   |--- NumWebVisitsMonth <= 1.50
|   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |--- NumWebVisitsMonth >  1.50
|   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |--- Income >  0.96
|   |   |   |   |   |   |--- Income <= 0.96
|   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |--- Income >  0.96
|   |   |   |   |   |   |   |--- Income <= 0.96
|   |   |   |   |   |   |   |   |--- class: 1
```

```
|   |   |   |   |   |   |   |--- Income >  0.96
|   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |--- NumWebVisitsMonth >  2.50
|   |   |   |   |--- Income <= 0.95
|   |   |   |   |   |--- NumWebVisitsMonth <= 4.50
|   |   |   |   |   |   |--- NumWebVisitsMonth <= 3.50
|   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |--- NumWebVisitsMonth >  3.50
|   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |--- NumWebVisitsMonth >  4.50
|   |   |   |   |   |   |--- class: 0
|   |   |   |   |--- Income >  0.95
|   |   |   |   |   |--- Income <= 0.96
|   |   |   |   |   |   |--- NumWebVisitsMonth <= 5.50
|   |   |   |   |   |   |   |--- Income <= 0.95
|   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |--- Income >  0.95
|   |   |   |   |   |   |   |   |--- Income <= 0.95
|   |   |   |   |   |   |   |   |   |--- Income <= 0.95
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.95
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.95
|   |   |   |   |   |   |   |   |   |--- Income <= 0.96
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.96
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.96
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.96
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |--- NumWebVisitsMonth >  5.50
|   |   |   |   |   |   |   |   |--- Income <= 0.96
|   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |--- Income >  0.96
|   |   |   |   |   |   |   |   |   |--- Income <= 0.96
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.96
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.96
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.96
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |--- Income >  0.96
|   |   |   |   |   |   |   |--- Income <= 0.96
|   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- Income >  0.96
|   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |--- Income >  0.96
|   |   |   |   |--- Income <= 0.96
```

```
|   |   |   |   |--- class: 1
|   |   |   |--- Income >  0.96
|   |   |   |   |--- NumWebVisitsMonth <= 2.50
|   |   |   |   |   |--- Income <= 0.98
|   |   |   |   |   |   |--- Income <= 0.98
|   |   |   |   |   |   |   |--- Income <= 0.98
|   |   |   |   |   |   |   |   |--- Income <= 0.97
|   |   |   |   |   |   |   |   |   |--- Income <= 0.97
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.97
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.97
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.97
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.97
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.98
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.98
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.98
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.98
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |--- Income >  0.98
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 1.50
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  1.50
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- Income >  0.98
|   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |--- Income >  0.98
|   |   |   |   |   |   |   |--- Income <= 0.99
|   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |--- Income >  0.99
|   |   |   |   |   |   |   |   |--- Income <= 0.99
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- Income >  0.99
|   |   |   |   |   |   |   |   |--- Income <= 1.00
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 1.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 1.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  1.00
|   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
```

```
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  1.50
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 1.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  1.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  1.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |--- NumWebVisitsMonth >  2.50
|   |   |   |   |   |   |--- Income <= 0.99
|   |   |   |   |   |   |   |--- Income <= 0.98
|   |   |   |   |   |   |   |   |--- Income <= 0.97
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 5.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.97
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.97
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.97
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 9
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.97
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  5.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.97
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.97
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.97
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.97
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.97
|   |   |   |   |   |   |   |   |   |--- Income <= 0.98
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.98
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.98
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.98
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.98
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.98
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.98
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.98
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |--- Income >  0.98
|   |   |   |   |   |   |   |   |--- Income <= 0.99
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.99
|   |   |   |   |   |   |   |   |   |--- Income <= 0.99
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.99
```

```
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 5.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.99
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.99
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  5.50
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |--- Income >  0.99
|   |   |   |   |   |   |   |--- Income <= 0.99
|   |   |   |   |   |   |   |   |--- Income <= 0.99
|   |   |   |   |   |   |   |   |   |--- Income <= 0.99
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.99
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.99
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 4.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  4.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.99
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.99
|   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |--- Income >  0.99
|   |   |   |   |   |   |   |   |--- Income <= 1.00
|   |   |   |   |   |   |   |   |   |--- Income <= 1.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  1.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 1.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  1.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- Income >  1.00
|   |   |   |   |   |   |   |   |   |--- Income <= 1.00
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  1.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 1.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  1.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 1.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |--- Income >  1.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|--- NumWebVisitsMonth >  6.50
|   |--- Income <= 0.00
|   |   |--- Income <= 0.00
|   |   |   |--- Income <= 0.00
|   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |--- Income <= 0.00
```

```
|   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
```

```
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 12
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 14
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 9.50
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- NumWebVisitsMonth >  9.50
|   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 10.50
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 9.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 18
```

```
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 9
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  9.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 10
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 23
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |--- NumWebVisitsMonth >  10.50
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |--- Income >  0.00
|   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |--- class: 0
```

```
|   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |--- class: 1
|   |   |--- Income >  0.00
|   |   |   |--- Income <= 0.00
|   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |--- class: 2
|   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |--- class: 0
|   |   |   |--- Income >  0.00
|   |   |   |   |--- NumWebVisitsMonth <= 9.50
|   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
```

```
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 21
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 10
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |--- NumWebVisitsMonth >  9.50
|   |   |   |   |   |   |--- Income <= 0.00
```

```
|   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 12.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  12.00
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 13.50
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  13.50
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 10.50
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  10.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |--- class: 2
```

```
|   |--- Income >  0.00
|   |   |--- Income <= 0.00
|   |   |   |--- Income <= 0.00
|   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
```

```
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 18
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
```

```
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 9.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  9.50
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 9.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  9.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 17
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
```

```
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 12
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 17
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 15
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 9.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  9.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
```

```
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |--- Income >  0.00
|   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
```

```
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 9.50
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  9.50
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 9.50
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  9.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.00
```

```
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |--- Income >  0.00
|   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |--- Income <= 0.00
```

```
|   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 9.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  9.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 9.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  9.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
```

```
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 12
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 9.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  9.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |--- NumWebVisitsMonth <= 9.50
|   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
```

```
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |--- NumWebVisitsMonth >  9.50
|   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |--- Income >  0.00
|   |   |   |   |   |--- Income <= 0.47
|   |   |   |   |   |   |--- Income <= 0.38
|   |   |   |   |   |   |   |--- Income <= 0.38
|   |   |   |   |   |   |   |   |--- Income <= 0.38
|   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 9
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 39
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 31
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 15
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 90
|   |   |   |   |   |   |   |   |--- Income >  0.38
|   |   |   |   |   |   |   |   |   |--- Income <= 0.38
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.38
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.38
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- Income >  0.38
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.38
```

```
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.38
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- Income >  0.38
|   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |--- Income >  0.38
|   |   |   |   |   |   |   |--- Income <= 0.45
|   |   |   |   |   |   |   |   |--- Income <= 0.44
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.40
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.38
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.38
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.40
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.44
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 13
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.44
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.44
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.41
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 10
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.41
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 13
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.44
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.44
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.45
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.45
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |--- Income >  0.45
|   |   |   |   |   |   |   |   |--- Income <= 0.46
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 9.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.45
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.45
```

```
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.45
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.45
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  9.50
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.46
|   |   |   |   |   |   |   |   |   |--- Income <= 0.47
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.46
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.46
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.46
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.46
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |--- Income >  0.47
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.47
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.47
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |--- Income >  0.47
|   |   |   |   |   |   |--- Income <= 0.86
|   |   |   |   |   |   |   |--- Income <= 0.79
|   |   |   |   |   |   |   |   |--- Income <= 0.79
|   |   |   |   |   |   |   |   |   |--- Income <= 0.78
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.78
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.47
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.47
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 29
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.78
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.78
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.78
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |--- Income >  0.78
|   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 9.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  9.50
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- Income >  0.79
```

```
|   |   |   |   |   |   |   |   |--- Income <= 0.79
|   |   |   |   |   |   |   |   |   |--- Income <= 0.79
|   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  0.79
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.79
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- Income >  0.79
|   |   |   |   |   |   |   |   |--- Income <= 0.79
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- Income >  0.79
|   |   |   |   |   |   |   |   |   |--- Income <= 0.81
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.81
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 9
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.81
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 0.81
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  0.81
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.81
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.81
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.81
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 9.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 12
|   |   |   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  9.50
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |--- Income >  0.86
|   |   |   |   |   |   |--- Income <= 0.99
|   |   |   |   |   |   |   |--- Income <= 0.97
|   |   |   |   |   |   |   |   |--- Income <= 0.90
|   |   |   |   |   |   |   |   |   |--- Income <= 0.87
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.87
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.87
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.87
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.89
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 13
|   |   |   |   |   |   |   |   |   |--- Income >  0.89
|   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |--- Income >  0.90
|   |   |   |   |   |   |   |   |   |--- Income <= 0.92
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.91
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
```

```
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.91
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.92
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.92
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.92
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 17
|   |   |   |   |   |   |   |--- Income >  0.97
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 7.50
|   |   |   |   |   |   |   |   |   |--- Income <= 0.99
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.98
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.98
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |--- Income >  0.99
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  7.50
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 9.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.99
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.99
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  9.50
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.98
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.98
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |--- Income >  0.99
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth <= 8.50
|   |   |   |   |   |   |   |   |   |--- Income <= 0.99
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- Income >  0.99
|   |   |   |   |   |   |   |   |   |   |--- Income <= 1.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- Income >  1.00
|   |   |   |   |   |   |   |   |   |   |   |--- Income <= 1.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |   |--- Income >  1.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |--- NumWebVisitsMonth >  8.50
|   |   |   |   |   |   |   |   |   |--- Income <= 1.00
|   |   |   |   |   |   |   |   |   |   |--- Income <= 0.99
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- Income >  0.99
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |--- Income >  1.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
```

time: 62 ms (started: 2024-04-06 00:23:02 +05:30)

[69]:
```
# Decision Tree : Feature Importance
dtc_imp_features = pd.DataFrame({'feature': df1_inputs_names, 'importance': np.
 ↪round(dtc_model.feature_importances_, 3)})
dtc_imp_features.sort_values('importance', ascending=False, inplace=True);␣
 ↪dtc_imp_features
```

[69]:

	feature	importance
0	Income	0.867
1	NumWebVisitsMonth	0.133

time: 16 ms (started: 2024-04-06 00:23:36 +05:30)

[70]:
```
# Decision Tree : Model Prediction (Training Subset)
dtc_model_predict = dtc_model.predict(df1_inputs_train); dtc_model_predict
```

[70]:
```
array([1, 0, 1, …, 2, 0, 2])
```

time: 16 ms (started: 2024-04-06 00:23:44 +05:30)

[71]:
```
# Decision Tree : Prediction (Testing Subset)
dtc_predict = dtc_model.predict(df1_inputs_test); dtc_predict
```

[71]:
```
array([1, 2, 2, …, 1, 2, 2])
```

time: 15 ms (started: 2024-04-06 00:23:51 +05:30)

[72]:
```
# Decision Tree : Model Evaluation (Training Subset)
dtc_model_conf_mat = pd.DataFrame(confusion_matrix(df1_output_train,␣
 ↪dtc_model_predict)); dtc_model_conf_mat
dtc_model_perf = classification_report(df1_output_train, dtc_model_predict);␣
 ↪print(dtc_model_perf)
```

```
              precision    recall  f1-score   support

           0       0.92      0.99      0.96     16035
           1       0.96      0.96      0.96     15965
           2       0.99      0.92      0.95     15999

    accuracy                           0.95     47999
   macro avg       0.96      0.95      0.95     47999
weighted avg       0.96      0.95      0.95     47999
```

time: 63 ms (started: 2024-04-06 00:23:59 +05:30)

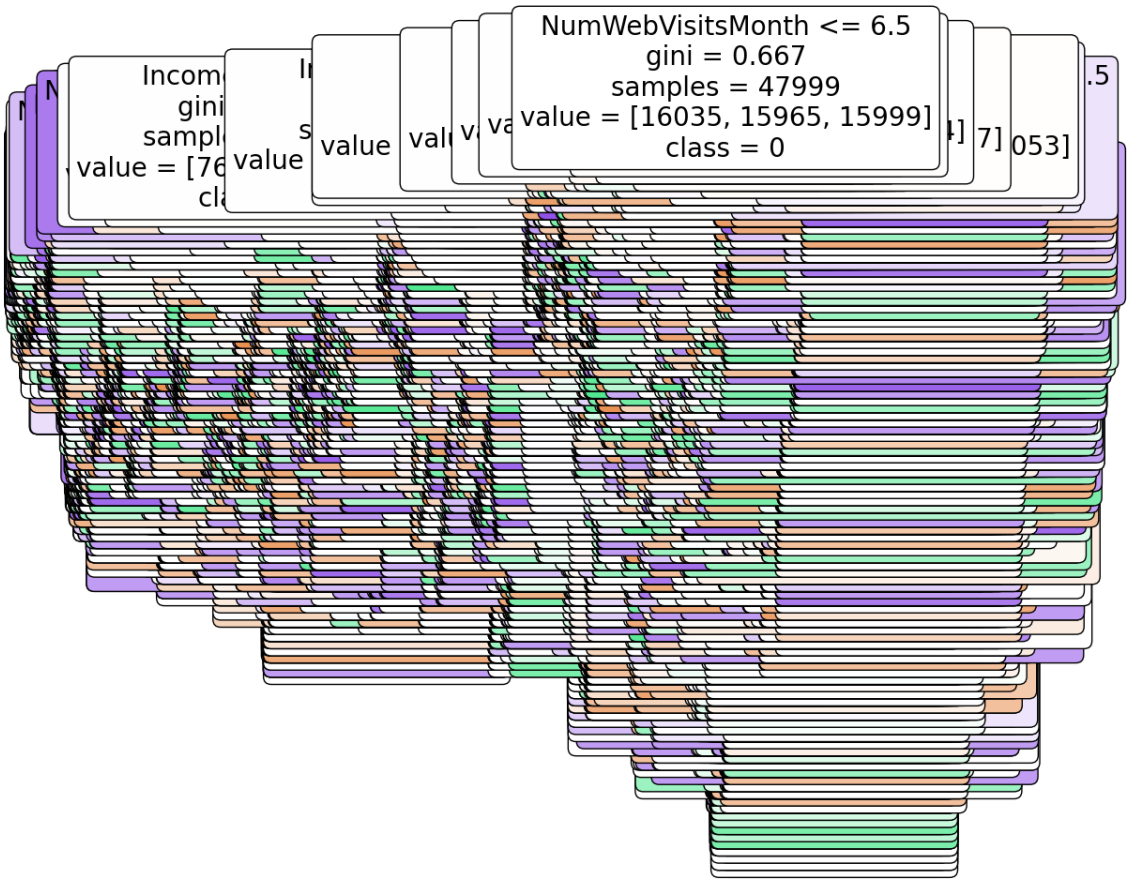
[73]:
```
# Decision Tree : Prediction Evaluation (Testing Subset)
```

```
dtc_predict_conf_mat = pd.DataFrame(confusion_matrix(df1_output_test,␣
 ↪dtc_predict)); dtc_predict_conf_mat
dtc_predict_perf = classification_report(df1_output_test, dtc_predict);␣
 ↪print(dtc_predict_perf)
```

```
              precision    recall  f1-score   support

           0       0.33      0.35      0.34      4009
           1       0.33      0.33      0.33      3991
           2       0.33      0.31      0.32      4000

    accuracy                           0.33     12000
   macro avg       0.33      0.33      0.33     12000
weighted avg       0.33      0.33      0.33     12000
```

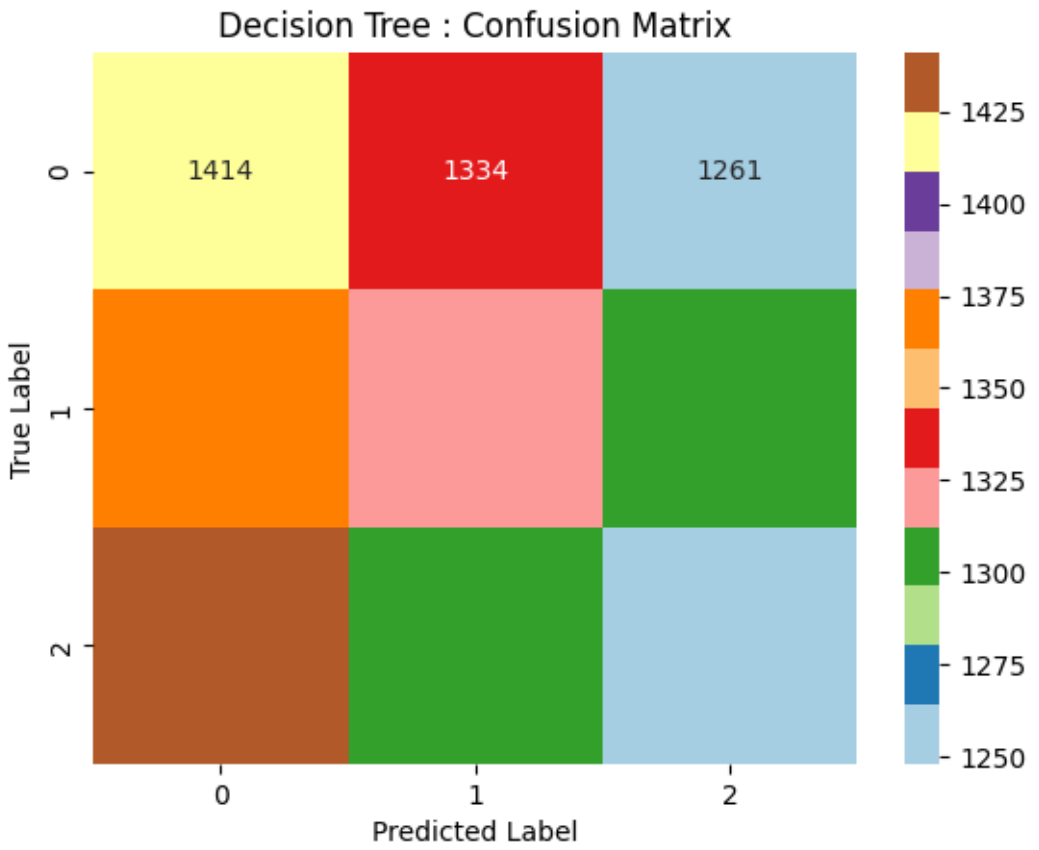
```
time: 16 ms (started: 2024-04-06 00:24:06 +05:30)
```

[74]:
```
import matplotlib.pyplot as plt
from sklearn.tree import plot_tree

# Set a larger figure size for better clarity
plt.figure(figsize=(10, 10))

# Plot the decision tree
train_subset_dtc_plot = plot_tree(dtc_model, feature_names=df1_inputs_names,␣
 ↪class_names=df1_output_labels, rounded=True, filled=True, fontsize=20)

# Show the plot
plt.show()
```



time: 9min 42s (started: 2024-04-06 00:24:14 +05:30)

[75]:
```
# Set up the plot
ax = plt.axes()

# Plot the confusion matrix with annotations in integer format
sns.heatmap(dtc_predict_conf_mat, annot=True, fmt='d', cmap='Paired')

# Set labels and title
ax.set_xlabel('Predicted Label')
ax.set_ylabel('True Label')
ax.set_title('Decision Tree : Confusion Matrix')

# Show the plot
plt.show()
```



time: 344 ms (started: 2024-04-06 00:33:56 +05:30)

[76]:
```
# Cross Validation
from sklearn.model_selection import cross_val_score

# Define your decision tree classifier with desired parameters
dtc_cv = DecisionTreeClassifier(criterion='gini', random_state=45007)

# Perform 5-fold cross-validation
cv_scores = cross_val_score(dtc_cv, df1_inputs, df1_output.values.ravel(),␣
 ↪cv=20)
print("Cross-Validation Scores:", cv_scores)
print("Average Cross-Validation Score:", np.mean(cv_scores))
```

```
Cross-Validation Scores: [0.34333333 0.326      0.334      0.33666667 0.339
0.32333333
 0.33133333 0.32866667 0.351      0.336      0.34066667 0.333
 0.33833333 0.34533333 0.352      0.35666667 0.349      0.327
 0.33433333 0.33744582]
Average Cross-Validation Score: 0.33815562409692124
```

time: 12.9 s (started: 2024-04-06 00:33:57 +05:30)

[77]:
```
from sklearn.metrics import f1_score

# Compute F1 score
f1 = f1_score(df1_output_test, dtc_predict, average='macro')  # or 'weighted'␣
 ↪for weighted F1 score
print("F1 Score:", f1)

# Weighted F1 score
weighted_f1 = f1_score(df1_output_test, dtc_predict, average='weighted')
print("Weighted F1 Score:", weighted_f1)
```

```
F1 Score: 0.33220822013557544
Weighted F1 Score: 0.33221593589476806
time: 15 ms (started: 2024-04-06 00:34:10 +05:30)
```

[78]:
```
# Specify the number of neighbors (k)
k = 3

# Initialize KNN classifier with k neighbors
knn = KNeighborsClassifier(n_neighbors=k)

# Fit the KNN model using the training data
knn.fit(df1_inputs_train, df1_output_train)
```

[78]:
```
KNeighborsClassifier(n_neighbors=3)
```

```
time: 32 ms (started: 2024-04-06 00:34:10 +05:30)
```

[79]:
```
# Make predictions using the testing data
y_pred = knn.predict(df1_inputs_test)
```

```
time: 313 ms (started: 2024-04-06 00:34:10 +05:30)
```

[80]:
```
# Calculate accuracy
accuracy = accuracy_score(df1_output_test, y_pred)
print(f'Accuracy: {accuracy}')

# Generate classification report
classification_rep = classification_report(df1_output_test, y_pred)
print(f'Classification Report:\n{classification_rep}')

# Generate confusion matrix
conf_mat = confusion_matrix(df1_output_test, y_pred)
print(f'Confusion Matrix:\n{conf_mat}')
```

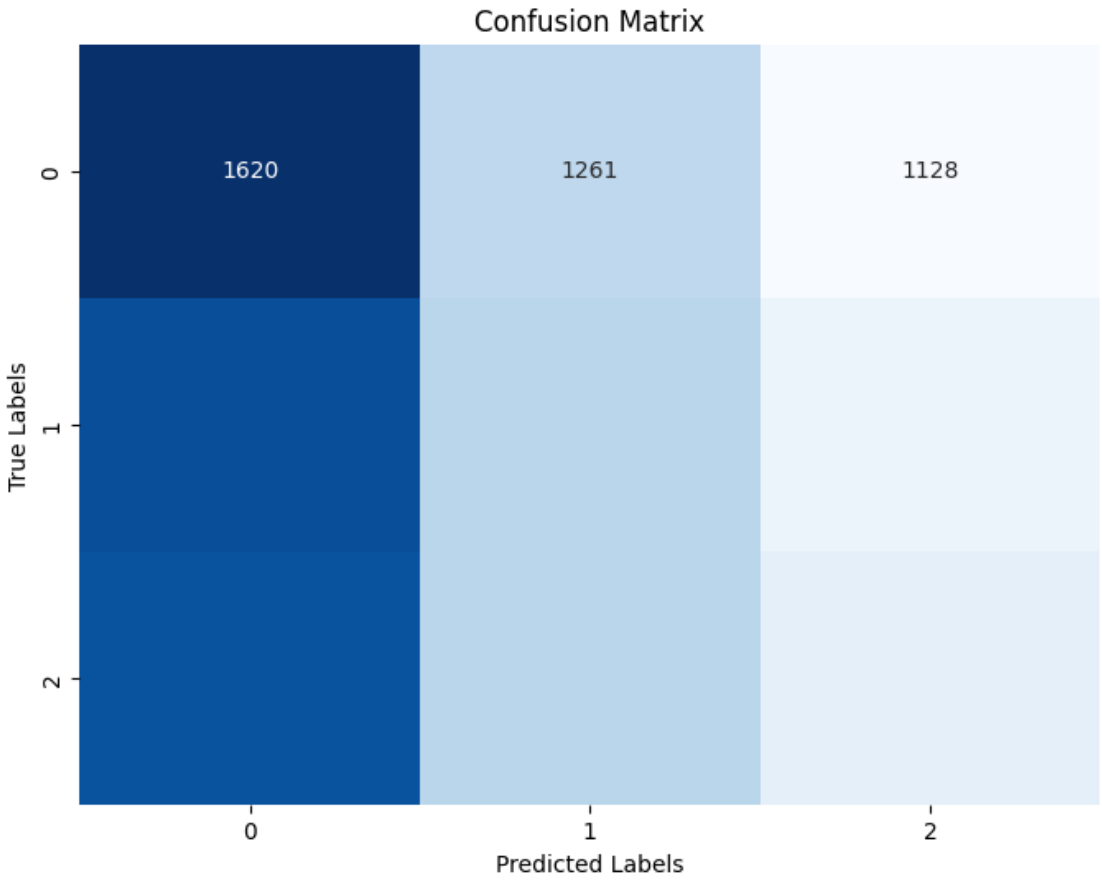
```
Accuracy: 0.3358333333333333
```

```
Classification Report:
              precision    recall  f1-score   support

           0       0.34      0.49      0.40      4009
           1       0.34      0.26      0.29      3991
           2       0.33      0.26      0.29      4000

    accuracy                           0.34     12000
   macro avg       0.34      0.34      0.33     12000
weighted avg       0.34      0.34      0.33     12000

Confusion Matrix:
[[1953 1005 1051]
 [1873 1027 1091]
 [1940 1010 1050]]
time: 31 ms (started: 2024-04-06 00:34:10 +05:30)
```

[81]:
```
k_values = [7, 9, 11, 13, 15]
for k in k_values:
    knn = KNeighborsClassifier(n_neighbors=k)
    knn.fit(df1_inputs_train, df1_output_train)  # Use your training data here
    y_pred = knn.predict(df1_inputs_test)  # Use your testing data here
    accuracy = accuracy_score(df1_output_test, y_pred)  # Compare predictions␣
 ↪with true labels
    print(f'Accuracy for k={k}: {accuracy}')
```

```
Accuracy for k=7: 0.33341666666666664
Accuracy for k=9: 0.33666666666666667
Accuracy for k=11: 0.33316666666666667
Accuracy for k=13: 0.3378333333333333
Accuracy for k=15: 0.33875
time: 1.8 s (started: 2024-04-06 00:34:10 +05:30)
```

[82]:
```
# Plot confusion matrix
def plot_confusion_matrix(y_true, y_pred):
    cm = confusion_matrix(y_true, y_pred)
    plt.figure(figsize=(8, 6))
    sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', cbar=False)
    plt.title('Confusion Matrix')
    plt.xlabel('Predicted Labels')
    plt.ylabel('True Labels')
    plt.show()

# Assuming df1_output_test and y_pred are the true and predicted labels,␣
 ↪respectively
plot_confusion_matrix(df1_output_test, y_pred)
```



time: 94 ms (started: 2024-04-06 00:34:12 +05:30)

[83]:
```
from sklearn.model_selection import cross_val_score
from sklearn.neighbors import KNeighborsClassifier

# Define the KNN classifier with a chosen number of neighbors (k)
knn = KNeighborsClassifier(n_neighbors=13)  # Example value, you can adjust this

# Perform cross-validation with 5 folds
cv_scores = cross_val_score(knn, df1_inputs, df1_output.values.ravel(), cv=20)

# Print the cross-validation scores
print("Cross-Validation Scores:", cv_scores)

# Calculate and print the average accuracy
avg_accuracy = cv_scores.mean()
print("Average Accuracy:", avg_accuracy)
```

```
Cross-Validation Scores: [0.328      0.34333333 0.346      0.331      0.34833333
0.32566667
```

```
 0.33333333 0.32633333 0.32666667 0.33566667 0.32666667 0.34333333
 0.33366667 0.35166667 0.32266667 0.33333333 0.33633333 0.33233333
 0.34566667 0.34044682]
Average Accuracy: 0.33552234078026005
time: 2.23 s (started: 2024-04-06 00:34:12 +05:30)
```

[89]:
```
from sklearn.svm import SVC
from sklearn.utils.validation import column_or_1d

# Ensure the shape of the target variable is correct
df1_output_train = column_or_1d(df1_output_train)

# Initialize SVC classifier with linear kernel
classifier = SVC(kernel='linear', random_state=45052)

# Fit the classifier to the training data
classifier.fit(df1_inputs_train, df1_output_train)
```

[89]:
```
SVC(kernel='linear', random_state=45052)

time: 1min 6s (started: 2024-04-06 01:06:51 +05:30)
```

[90]:
```
y_pred = classifier.predict(df1_inputs_test)
```

```
time: 8.92 s (started: 2024-04-06 01:07:57 +05:30)
```

[91]:
```
cm = confusion_matrix(df1_output_test, y_pred)
print(cm)
accuracy_score(df1_output_test,y_pred)
```

```
[[3695    1  313]
 [3687    0  304]
 [3722    0  278]]
```

[91]:
```
0.33108333333333334

time: 15 ms (started: 2024-04-06 01:08:06 +05:30)
```

[92]:
```
from sklearn.model_selection import cross_val_score
from sklearn.svm import SVC


# Perform K-fold cross-validation with 3 folds and enable parallel processing
cv_scores = cross_val_score(classifier, df1_inputs, df1_output.values.ravel(),␣
 ↪cv=5, n_jobs=-1)

# Print the cross-validation scores
print("Cross-validation scores:", cv_scores)
```

```
# Calculate and print the average cross-validation score
avg_cv_score = np.mean(cv_scores)
print("Average Cross-validation score:", avg_cv_score)
```

```
Cross-validation scores: [0.33283333 0.3345     0.33516667 0.33666667 0.3356113
]
Average Cross-validation score: 0.33495559352168236
time: 2min 22s (started: 2024-04-06 01:08:06 +05:30)
```

[93]:
```
from sklearn.model_selection import cross_val_score
from sklearn.svm import SVC
from sklearn.metrics import make_scorer, f1_score

# Define a scorer for weighted F1 score
weighted_f1_scorer = make_scorer(f1_score, average='weighted')

# Perform K-fold cross-validation with 5 folds using weighted F1 score as the␣
 ↪scoring metric
cv_scores_weighted_f1 = cross_val_score(classifier, df1_inputs, df1_output.
 ↪values.ravel(), cv=5, scoring=weighted_f1_scorer)

# Print the cross-validation scores for weighted F1
print("Cross-validation scores (Weighted F1):", cv_scores_weighted_f1)

# Calculate and print the average cross-validation score for weighted F1
avg_cv_score_weighted_f1 = np.mean(cv_scores_weighted_f1)
print("Average Cross-validation score (Weighted F1):", avg_cv_score_weighted_f1)
```

```
Cross-validation scores (Weighted F1): [0.20192186 0.20399325 0.20243803
0.20785549 0.25968666]
Average Cross-validation score (Weighted F1): 0.21517905943747903
time: 5min 41s (started: 2024-04-06 01:10:29 +05:30)
```

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]: