045005

April 13, 2024

1 REPORT by AGATHIYAN K (045005)

Prepared by - Agathiyan K (045005) PGDM - BDA04

Project Title: Demographic Patterns and Home Loan Applicant Profiling

1. Project Objectives | Problem Statements

1.1. PO1 | PS1: Classification of Consumer Data into Segments | Clusters | Classes using Supervised Learning Classification Algorithms

1.2. PO2 | PS2: Determination of an Appropriate Classification Model

1.3. PO3 | PS3: Identification of Important | Contributing | Significant Variables or Features and their Thresholds for Classification

2.2.2. Outcome Variable or Feature: OV

['action_taken_name']

The outcome variable, labeled as 'action_taken_name', represents the target variable in our analysis. It is the variable of interest that we aim to predict or classify using our machine learning models.

2.2.3. Input Variables or Features having Categories | Input Categorical Variables or Features (ICV)

['msamd_name', 'loan_type_name', 'loan_purpose_name', 'hud_median_family_income', 'loan_amount_000s']

PREPROCESSING REPORT is after the Managerial Insights

3.2. Data Analysis

3.2.1.1. PO1 | PS1:: Supervised Machine Learning Classification Algorithm | **Variable or Feature Analysis**: Decision Tree (Base Model)

Feature	Importance
loan_amount_000s	0.604
hud_median_family_income	0.162
msamd_name	0.148
loan_type_name	0.049
loan_purpose_name	0.038

Important Variables: - loan_amount_000s (0.604) - hud_median_family_income (0.162) - msamd_name (0.148)

Least-Important Variables: - loan_type_name (0.049) - loan_purpose_name (0.038)

Analysis of both the important and Least-important variables:

Important Variables Analysis:

loan_amount_000s: This variable has the highest importance, indicating that loan amount is the most influential factor in predicting the target variable. Customers applying for larger loan amounts may have different outcomes compared to those applying for smaller loans.

hud_median_family_income: The median family income is also significant, suggesting that applicants' financial stability or background plays a crucial role in loan approval decisions.

msamd_name: The Metropolitan Statistical Area/Division (MSAMD) name is also important, implying that regional differences may impact loan approval rates.

Least-Important Variables Analysis:

loan_type_name and loan_purpose_name: These variables have lower importance compared to the others. While still contributing to the model, they are not as influential in predicting the target variable. It's possible that the specific type or purpose of the loan has less impact on the loan approval decision compared to other factors like loan amount and income.

3.2.1.2. PO1 | PS1:: Supervised Machine Learning Classification Algorithms | **Variable or Feature Analysis**: {Logistic Regression | Support Vector Machine | K Nearest Neighbour} (Comparison Models)

Logistic Regression

Feature	Importance
hud_median_family_income	9.955654e-05
msamd_name	1.011932e-08
loan_purpose_name	1.903765e-09
loan_type_name	9.476882e-10
loan_amount_000s	6.487305e-12

Important Variables: - hud_median_family_income (9.955654e-05)

Least-Important Variables: - loan_type_name (9.476882e-10) - loan_purpose_name (1.903765e-09) - loan_amount_000s (6.487305e-12) - msamd_name (1.011932e-08)

Analysis of both the important and non-important variables:

Important Variables Analysis:

hud_median_family_income: This variable has the highest importance among the features, indicating its significance in determining loan approval.

Least-Important Variables Analysis:

msamd_name, loan_purpose_name, loan_type_name, and loan_amount_000s: These variables have very low importance according to the LR model, suggesting they may not strongly influence the loan approval decision.

Support Vector Machine

Feature	Importance
hud_median_family_income	73931.750494
msamd_name	6.869248
loan_purpose_name	1.421012
loan_type_name	0.592216
loan_amount_000s	0.005059

Important Variables: - hud_median_family_income (73931.750494)

Least-Important Variables: - msamd_name (6.869248) - loan_purpose_name (1.421012) - loan_type_name (0.592216) - loan_amount_000s (0.005059)

Analysis of both the important and Least-important variables:

Important Variables Analysis:

hud_median_family_income: This variable has the highest importance, indicating that family income is the most influential feature according to the SVM model.

Least-Important Variables Analysis:

msamd_name, loan_purpose_name, loan_type_name, and loan_amount_000s: These variables have significantly lower importance compared to hud_median_family_income, suggesting that they have little to no impact on the classification outcome according to the SVM model.

K Nearest Neighbour

feature	importance
hud_median_family_income	0.003083
loan_type_name	0.000383
msamd_name	0.000000
loan_purpose_name	-0.000375
loan_amount_000s	-0.000642

Important Variables: - hud_median_family_income (0.003083)

Least-Important Variables: - loan_type_name (0.000383) - msamd_name (0.000000) - loan_purpose_name (-0.000375) - loan_amount_000s (-0.000642)

Analysis of both the important and Least-important variables:

Important Variables Analysis:

hud_median_family_income: This variable has the highest importance, albeit very low compared to Decision Tree. It suggests that family income may still have some influence on the classification outcome, although it’s not as significant as in the Decision Tree model.

Non-Important Variables Analysis:

loan_type_name, msamd_name, loan_purpose_name, and loan_amount_000s: These variables have negligible importance, with some even having negative importance. This indicates that they have little to no impact on the classification outcome according to the KNN model.

3.2.2.1.1. PO2 | PS2:: Classification Model Performance Evaluation: Confusion Matrix {Accuray, Recall, Precision, F1-Score} **(Base Model: Decision Tree)**

Class	Precision	Recall	F1-Score	Support
0	0.58	0.44	0.50	63
1	0.66	0.34	0.45	121
2	0.68	0.59	0.63	2000
3	0.72	0.43	0.54	334
4	0.97	0.99	0.98	41861
5	0.92	0.26	0.41	582
6	1.00	0.38	0.56	13
7	1.00	0.88	0.94	26

	Precision	Recall	F1-Score	Support
Accuracy	0.95			45000
Macro Avg	0.82	0.54	0.63	45000
Weighted Avg	0.95	0.95	0.95	45000

Precision: Indicates the proportion of true positive predictions out of all positive predictions. For example, for class 0, 58% of the predicted instances were correct. Recall: Represents the proportion of true positive predictions out of all actual positive instances. For class 0, 44% of the actual instances were correctly predicted. F1-score: The harmonic mean of precision and recall, providing a balance between the two metrics. It is a measure of accuracy for each class. Support: Indicates the number of instances in each class. Accuracy: The overall proportion of correct predictions across all classes. In this case, the model achieved an accuracy of 95% on the training subset. Macro avg: The average of precision, recall, and F1-score across all classes, without considering class imbalance. Weighted avg: The weighted average of precision, recall, and F1-score, considering the number of instances in each class.

3.2.2.1.2. PO2 | PS2:: Classification Model Performance Evaluation: Time Statistics | (CPU | GPU) Memory Statistics **(Base Model: Decision Tree)**

Model	Memory Used (MB)
Decision Tree	722.75
KNN (k=5)	722.75
KNN (k=7)	722.75
KNN (k=9)	722.75
KNN (k=11)	722.75
KNN (k=13)	722.75
KNN (k=15)	722.75

Model	Memory Used (MB)
Logistic Regression	722.75
SVM	722.77

All algorithms have utilized approximately the same amount of memory, around 722.75 MB, except for the SVM model, which used slightly more memory at 722.77 MB. The memory usage across different algorithms is consistent, indicating that memory consumption may not be a differentiating factor when choosing between these algorithms. Factors other than memory usage, such as accuracy, training and prediction time, should be considered to select the most suitable algorithm for a particular task.

3.2.2.2.1. PO2 | PS2:: Classification Model Performance Evaluation: Confusion Matrix {Accuray, Recall, Precision, F1-Score} (Comparison Models: Logistic Regression | Support Vector Machine | K Nearest Neighbour)

KNN Accuracy: 0.9174666666666667 Classification Report:

Class	Precision	Recall	F1-Score	Support
0	0.00	0.00	0.00	21
1	0.00	0.00	0.00	40
2	0.23	0.12	0.16	666
3	0.02	0.01	0.01	112
4	0.94	0.98	0.96	13954
5	0.09	0.01	0.02	194
6	0.00	0.00	0.00	4
7	0.67	0.67	0.67	9

Accuracy
0.92

	Precision	Recall	F1-Score	Support
Macro Avg	0.24	0.22	0.23	15000
Weighted Avg	0.88	0.92	0.90	15000

Accuracy: The overall accuracy of the model on the test dataset is approximately 91.75%. Precision: For each class, precision measures the proportion of true positive predictions out of all positive predictions. In this case, precision varies across different classes, with class 4 having the highest precision (94%) and classes 0, 1, and 6 having precision values of 0%. Recall: Represents the proportion of true positive predictions out of all actual positive instances. Similar to precision, recall varies across classes, with class 4 having the highest recall (98%) and classes 0, 1, 3, and 6 having recall values of 0%. F1-score: The harmonic mean of precision and recall, providing a balance between the two metrics. It reflects the model's accuracy for each class, with class 4 having the highest F1-score (96%) and classes 0, 1, and 6 having F1-scores of 0%. Support: Indicates the number of instances in each class in the test dataset. Macro avg: The average of precision, recall, and F1-score across all classes, without considering class imbalance. The macro-average F1-score is 23%. Weighted avg: The weighted average of precision, recall, and F1-score, considering the number of instances in each class. The weighted average F1-score is 90%, indicating the overall performance of the model.

LR Accuracy: 0.9302666666666667 Classification Report:

	Precision	Recall	F1-Score	Support
0	0.00	0.00	0.00	21
1	0.00	0.00	0.00	40
2	0.00	0.00	0.00	666
3	0.00	0.00	0.00	112
4	0.93	1.00	0.96	13954
5	0.00	0.00	0.00	194
6	0.00	0.00	0.00	4
7	0.00	0.00	0.00	9

	Precision	Recall	F1-Score	Support
Accuracy	0.93			15000
Macro Avg	0.12	0.12	0.12	15000
Weighted Avg	0.87	0.93	0.90	15000

Accuracy: The overall accuracy of the model on the test dataset is approximately 93.03%. Precision: For each class, precision measures the proportion of true positive predictions out of all positive predictions. In this case, precision varies across different classes, with class 4 having the highest precision (93%) and classes 0, 1, 2, 3, 5, 6, and 7 having precision values of 0%. Recall: Represents the proportion of true positive predictions out of all actual positive instances. Similar to precision, recall varies across classes, with class 4 having the highest recall (100%) and classes 0, 1, 2, 3, 5, 6, and 7 having recall values of 0%. F1-score: The harmonic mean of precision and recall, providing a balance between the two metrics. It reflects the model's accuracy for each class, with class 4 having the highest F1-score (96%) and classes 0, 1, 2, 3, 5, 6, and 7 having F1-scores of 0%. Support: Indicates the number of instances in each class in the test dataset. Macro avg: The average of precision, recall, and F1-score across all classes, without considering class imbalance. The macro-average F1-score is 12%. Weighted avg: The weighted average of precision, recall, and F1-score, considering the number of instances in each class. The weighted average F1-score is 90%, indicating the overall performance of the model.

SVM Accuracy: 0.9302666666666667 Memory used (MB): 722.74609375 Classification Report:

Class	Precision	Recall	F1-Score	Support
0	0.00	0.00	0.00	21
1	0.00	0.00	0.00	40
2	0.00	0.00	0.00	666
3	0.00	0.00	0.00	112

Class	Precision	Recall	F1-Score	Support
4	0.93	1.00	0.96	13954
5	0.00	0.00	0.00	194
6	0.00	0.00	0.00	4
7	0.00	0.00	0.00	9

Metric	Value
Accuracy	0.93
Macro Avg	
Precision	0.12
Recall	0.12
F1-Score	0.12
Support	15000
Weighted Avg	
Precision	0.87
Recall	0.93
F1-Score	0.90
Support	15000

Accuracy: The accuracy of the SVM classifier on the test dataset is 93.03%. Memory used: The memory used by the SVM classifier during training and prediction is approximately 722.75 MB. Classification Report: Provides precision, recall, F1-score, and support for each class in the test dataset. It shows that the model performs poorly for most classes, with precision, recall, and F1-score values close to zero. Confusion Matrix: Shows the distribution of true positive, false positive, true negative, and false negative predictions for each class. In this case, the confusion matrix indicates that the model correctly predicts class 4 most of the time, while performance on other classes is poor.

3.2.2.2.2. PO2 | PS2:: Classification Model Performance Evaluation: Time Statistics | (CPU | GPU) Memory Statistics (Comparison Models: Logistic Regression | Support Vector Machine | K Nearest Neighbour)

Model	Memory Used (MB)
Decision Tree	722.75
KNN (k=5)	722.75
KNN (k=7)	722.75
KNN (k=9)	722.75
KNN (k=11)	722.75
KNN (k=13)	722.75
KNN (k=15)	722.75
Logistic Regression	722.75
SVM	722.77

Analysis:

All algorithms have utilized approximately the same amount of memory, around 722.75 MB, except for the SVM model, which used slightly more memory at 722.77 MB. The memory usage across different algorithms is consistent, indicating that memory consumption may not be a differentiating factor when choosing between these algorithms. Factors other than memory usage, such as accuracy, training and prediction time, should be considered to select the most suitable algorithm for a particular task.

3.2.3.1. PO3 | PS3:: Variable or Feature Analysis: Base Model (Decision Tree) 3.2.3.1.1. List of Relevant or Important Variables or Features and their Thresholds

loan_amount_000s: This variable is the most influential in predicting the target. Its high importance suggests that loan amount plays a significant role in determining the outcome.

hud_median_family_income: The median family income is also crucial, indicating that applicants' financial stability or background is an important factor. msamd_name: While not as influential as loan amount or income, the Metropolitan Statistical Area/Division (MSAMD) name still contributes significantly to the model, suggesting regional differences in loan approval.

3.2.3.1.2. List of Non-Relevant or Non-Important Variables or Features

loan_type_name: With an importance score of 0.049, it falls below the threshold of 0.1, indicating its lesser impact on predicting the target variable.

loan_purpose_name: Similarly, with an importance score of 0.038, it is considered non-important by the model.

3.2.3.2. PO3 | PS3:: Variable or Feature Analysis: Comparison Models (Logistic Regression | Support Vector Machine | K Nearest Neighbour) 3.2.3.2.1. List of Relevant or Important Variables or Features and their Thresholds

KNN loan_amount_000s hud_median_family_income msamd_name

LR loan_amount_000s hud_median_family_income msamd_name

SVM loan_amount_000s hud_median_family_income msamd_name

3.2.3.2.2. List of Non-Relevant or Non-Important Variables or Features

KNN loan_type_name loan_purpose_name

LR loan_type_name loan_purpose_name

SVM loan_type_name loan_purpose_name

4. Results | Observations

4.1. Classification Model Parameters: Base Model (Decision Tree) | Comparison Models (Logistic Regression | Support Vector Machine | K Nearest Neighbour) | Important Variables or Features and their Thresholds

Variables	Decision Tree	KNN	SVM	Logistic Regression
hud_median_family_income	0.162	0.003083	73931.750494	9.955654e-05
loan_amount_000s	0.604	-0.000642	0.005059	6.487305e-12
msamd_name	0.148	0.000000	6.869248	1.011932e-08
loan_type_name	0.049	0.000383	0.592216	9.476882e-10

Variables	Decision Tree	KNN	SVM	Logistic Regression
loan_purpose_name	0.038	-0.000375	1.421012	1.903765e-09

Analysis for Important Variables:

hud_median_family_income: SVM assigns the highest importance to median family income, followed by Decision Tree. KNN and Logistic Regression give comparatively lower importance.

loan_amount_000s: DT identifies loan amount as the most crucial variable, with SVM also considering it important. However, KNN and LR assign negligible importance to this variable.

msamd_name: SVM gives the highest importance to MSAMD name, while DT also finds it significant. KNN considers it less important, and LR assigns minimal importance.

loan_type_name and loan_purpose_name: These variables have varying importance across methods, with SVM assigning the highest importance to loan type and KNN giving the lowest importance to loan purpose.

All methods consider all variables to be important to some extent, so there are no non-important variables based on the criteria provided. However, the importance values for some variables are extremely low, indicating their limited impact on the prediction outcomes across all methods.

4.2. Classification Model Performance: Time, Memory Statistics and Accuracy [**Base Model (Decision Tree) | Comparison Models (Logistic Regression | Support Vector Machine | K Nearest Neighbour)**]

Model	Time Taken (s)	Memory Used (MB)	Accuracy
Decision Tree	0.087596	722.746094	0.899917
KNN (k=5)	0.047091	722.746094	0.919667
KNN (k=7)	0.046967	722.746094	0.925583
KNN (k=9)	0.047910	722.746094	0.927500
KNN (k=11)	0.045176	722.746094	0.928583
KNN (k=13)	0.086027	722.746094	0.931000
KNN (k=15)	0.079144	722.746094	0.931333
Logistic Regression	0.320378	722.746094	0.934083
SVM	13.512154	722.769531	0.934083

All algorithms utilize the same amount of memory: 722.746094 MB. Time Taken: Decision Tree is the fastest, followed by KNN with k=11 being the fastest among KNN variants. Logistic Regression is the slowest, while SVM takes significantly longer time compared to other algorithms.

Memory Used: All algorithms utilize the same amount of memory, indicating similar memory requirements.

Accuracy: Logistic Regression and SVM achieve the highest accuracy of 0.934083, followed closely by KNN with k=15 at 0.931333. Decision Tree has the lowest accuracy at 0.899917, but still performs reasonably well.

4.3. Variable or Feature Analysis: Base Model (Decision Tree) | Comparison Models (Logistic

Regression | Support Vector Machine | K Nearest Neighbour)

4.3.1. List of Relevant or Important Variables or Features and their Thresholds loan_amount_000s: This variable is the most influential in predicting the target. Its high importance suggests that loan amount plays a significant role in determining the outcome. hud_median_family_income: The median family income is also crucial, indicating that applicants' financial stability or background is an important factor. msamd_name: While not as influential as loan amount or income, the Metropolitan Statistical Area/Division (MSAMD) name still contributes significantly to the model, suggesting regional differences in loan approval.

4.3.2. List of Non-Relevant or Non-Important Variables or Features loan_type_name: With an importance score of 0.049, it falls below the threshold of 0.1, indicating its lesser impact on predicting the target variable. loan_purpose_name: Similarly, with an importance score of 0.038, it is considered non-important by the model.

5. Managerial Insights

5.1. Appropriate Model **{Decision Tree | Logistic Regression | Support Vector Machine | K Nearest Neighbour}** Time Taken (s): Decision Tree is the fastest among all algorithms, indicating its efficiency in training and prediction. This could be advantageous in scenarios requiring quick model development and deployment. KNN (k=11) exhibits the next best performance in terms of time taken, making it suitable for applications where moderate computational resources are available.

Memory Used (MB): All algorithms utilize the same amount of memory, suggesting that memory efficiency is not a distinguishing factor in choosing between these algorithms for this particular dataset.

Accuracy: Logistic Regression and SVM achieve the highest accuracy of 0.934083, followed closely by KNN (k=15) at 0.931333. These algorithms demonstrate robust performance in correctly predicting the target variable. Decision Tree achieves a lower accuracy of 0.899917 compared to other algorithms but still provides reasonable predictive performance.

Logistic Regression and SVM stand out as the top performers in terms of accuracy. These models are recommended when high predictive accuracy is critical, such as in financial risk assessment or medical diagnosis. Decision Tree, although less accurate, offers advantages in terms of speed and simplicity. It can be preferred for exploratory analysis, as it provides interpretable decision rules and insights into feature importance. KNN demonstrates competitive accuracy across different values of k, indicating its suitability for scenarios where local patterns or nearest neighbors' influence is significant.

Dataset-specific Considerations: Logistic Regression and SVM might perform better in this dataset due to their ability to handle complex relationships between features and the target variable. If the dataset contains non-linear relationships or interactions, SVM's kernel trick can effectively capture such patterns. KNN, being a non-parametric algorithm, can adapt well to the dataset's characteristics, particularly if there are regions with dense clusters of data points. Decision Tree may struggle if the dataset has intricate decision boundaries or noisy features, leading to lower accuracy compared to other algorithms. However, its simplicity and interpretability make it suitable for initial data exploration and generating actionable insights.

In summary, the choice of algorithm depends on the specific requirements of the task at hand, balancing factors such as accuracy, interpretability, computational resources, and the nature of the dataset. For this particular dataset, considering the trade-offs between accuracy and interpretability, Logistic Regression and SVM emerge as the top choices, providing both high accuracy and potential for model interpretation.

5.2. Relevant or Important Variables or Features (Given the Appropriate Model)

Variables	Decision Tree	KNN	SVM	Logistic Regression
hud_median_family_income	0.162	0.003083	73931.750494	9.955654e-05
loan_amount_000s	0.604	-0.000642	0.005059	6.487305e-12
msamd_name	0.148	0.000000	6.869248	1.011932e-08
loan_type_name	0.049	0.000383	0.592216	9.476882e-10
loan_purpose_name	0.038	-0.000375	1.421012	1.903765e-09

Loan Amount: Appropriate Algorithm: Logistic Regression or SVM may be suitable for modeling the relationship with loan amount, as they can handle continuous variables effectively and are robust to outliers. These algorithms can capture complex relationships between loan amount and the target variable, such as non-linearity or interaction effects.

HUD Median Family Income: Appropriate Algorithm: Logistic Regression or Decision Tree could be effective for modeling the impact of median family income. Logistic Regression can provide insights into the direction and magnitude of the relationship, while Decision Tree can identify threshold values or interactions with other variables.

MSAMD Name: Appropriate Algorithm: Decision Tree or Random Forest may be suitable for capturing the influence of MSAMD name. Decision Tree can identify specific MSAMD regions associated with certain outcomes, while Random Forest can enhance the robustness of predictions by aggregating multiple decision trees.

MODELS:

Decision Tree (DT): DT identifies “loan_amount_000s” as the most important variable for predicting the target outcome, followed by “hud_median_family_income” and “msamd_name”. DT offers a clear hierarchical structure for decision-making based on feature importance.

K-Nearest Neighbors (KNN): KNN assigns relatively lower importance to all variables compared to DT and SVM. KNN’s performance in identifying important variables might be affected by its reliance on local neighborhood information.

Support Vector Machine (SVM): SVM assigns the highest importance to “hud_median_family_income”, followed by “loan_amount_000s” and “msamd_name”. SVM’s ability to handle high-dimensional data and capture complex relationships might contribute to its effectiveness in identifying important variables.

Logistic Regression (LR): LR assigns extremely low importance values to all variables, indicating minimal impact on the prediction outcomes. LR’s linear decision boundary might result in underestimating the importance of certain variables compared to non-linear models like DT and SVM.

Conclusion:

SVM and DT appear to be the most effective models for predicting important variables in this scenario. SVM offers a robust approach for identifying influential features, particularly for cases where non-linear relationships exist between predictors and the target variable. DT provides an interpretable decision-making process, which can be advantageous for understanding the underlying logic behind variable importance. LR, while commonly used for its simplicity and interpretability, may not be the optimal choice for identifying important variables in this dataset due to its linear nature.

Overall, SVM and DT stand out as preferred choices for predicting important variables, considering their ability to capture complex relationships and provide transparent decision rules.

In summary, the choice of algorithm depends on the specific characteristics of the dataset, such as the nature of the variables, the complexity of the relationships, and the desired interpretability of the model. While KNN is a powerful algorithm for certain types of problems, it may not be the best choice for modeling the relationship between loan features and approval status in this particular dataset. For this dataset, Logistic Regression, SVM, and Decision Tree are recommended based on their ability to handle different types of variables and capture various aspects of the relationship between loan features and approval status.

2 PREPROCESSING REPORT

Prepared by - Agathiyan K (045005) PGDM - BDA04

Project Title: Demographic Patterns and Home Loan Applicant Profiling

1. Project Objectives | Problem Statements

1.1. PO1 | PS1: Classification of Consumer Data into Segments | Clusters | Classes using Supervised Learning Classification Algorithms

1.2. PO2 | PS2: Determination of an Appropriate Classification Model

1.3. PO3 | PS3: Identification of Important | Contributing | Significant Variables or Features and their Thresholds for Classification

2. Description of Data

2.1. Data Source, Size, Shape

2.1.1. Data Source (Website Link) https://www.kaggle.com/datasets/miker400/washington-state-home-mortgage-hdma2016 https://drive.google.com/file/d/13fp1-YgAuSiR_bWZwetJiEcJZOeDeAtu/view?usp=sharing

2.1.2. Data Size (in KB | MB | GB …)

30.1 MB

2.1.3. Data Shape (Dimension: Number of Variables | Number of Records) 39 Variables (39 columns)

Maximum number of rows 60,000

Total number of records: 60000

Total number of filled cells: 2038780

Missed cells: 301220

2.2. Description of Variables

2.2.1. Index Variable(s): I1, I2, … The index variable is S.no

2.2.2. Outcome Variable or Feature: OV

['action_taken_name']

The outcome variable, labeled as 'action_taken_name', represents the target variable in our analysis. It is the variable of interest that we aim to predict or classify using our machine learning models.

2.2.3. Input Variables or Features having Categories | Input Categorical Variables or Features (ICV)

['msamd_name', 'loan_type_name', 'loan_purpose_name', 'hud_median_family_income', 'loan_amount_000s']

2.2.3.1. Input Variables or Features having Nominal Categories | Categorical Variables or Features - Nominal Type: ICNV1, ICNV2, …

All the categorical variables available in the dataset for nominal variables

2.2.3.2. Input Variables or Features having Ordinal Categories | Categorical Variables or Features - Ordinal Type: ICOV1, ICOV2, …

No ordinal data available in the dataset

2.2.3. Input Non-Categorical Variables or Features: INCV1, INCV2, …

['hud_median_family_income', 'loan_amount_000s']

2.3. Descriptive Statistics

2.3.1. Descriptive Statistics: Outcome Variable or Feature (Categorical)

2.3.1.1. Count | Frequency Statistics count unique

Cluster_Label	Count
3	14162
1	14049
0	11435
4	10743
2	9611

2.3.1.2. Proportion (Relative Frequency) Statistics

Cluster_Label	Proportion
3	0.236033
1	0.234150
0	0.190583
4	0.179050
2	0.160183

2.3.2. Descriptive Statistics: Input Categorical Variables or Features

2.3.2.1. Count | Frequency Statistics

Variable	Count Unique
state_name	1
state_abbr	1
respondent_id	593
purchaser_type_name	10
property_type_name	3
preapproval_name	3
owner_occupancy_name	3
msamd_name	14
loan_type_name	4
loan_purpose_name	3
lien_status_name	4
hoepa_status_name	2
county_name	39
co_applicant_sex_name	5
co_applicant_ethnicity_name	5
applicant_sex_name	4
applicant_ethnicity_name	4
agency_name	6
agency_abbr	6
action_taken_name	8

Variable	Value
state_name	Washington
state_abbr	WA
respondent_id	32489
purchaser_type_name	Loan was not originated or was not sold in calendar year covered by the loan/application register
property_type_name	One-to-four family dwelling (other than manufactured housing)
preapproval_name	Not applicable
owner_occupancy_name	Owner-occupied as a principal dwelling
msamd_name	Seattle, Bellevue, Everett - WA
loan_type_name	Conventional
loan_purpose_name	Refinancing
lien_status_name	Secured by a first lien
hoepa_status_name	Not a HOEPA loan
county_name	King County
co_applicant_sex_name	No co-applicant
co_applicant_ethnicity_name	No co-applicant
applicant_sex_name	Male
applicant_ethnicity_name	Not Hispanic or Latino

Variable	Value
agency_name	Department of Housing and Urban Development
agency_abbr	HUD
action_taken_name	Loan originated

Variable	Frequency
state_name	60000
state_abbr	60000
respondent_id	5006
purchaser_type_name	16112
property_type_name	57630
preapproval_name	47832
owner_occupancy_name	53940
msamd_name	17965
loan_type_name	42917
loan_purpose_name	28576
lien_status_name	57046
hoepa_status_name	59996
county_name	12915
co_applicant_sex_name	26987
co_applicant_ethnicity_name	26987
applicant_sex_name	37070
applicant_ethnicity_name	44014
agency_name	27514
agency_abbr	27514
action_taken_name	55815

In the context of catdf dataset: state_name and state_abbr: These columns have only one unique value, which is "Washington" for state_name and "WA" for state_abbr. This suggests that these columns may not provide much information for analysis as they have constant values for all rows.

respondent_id: This column has 593 unique values, and the most frequent respondent_id is "32489" with a frequency of 5006. This column likely identifies different respondents.

Other categorical columns: Each column represents a categorical variable, and the summary provides information about the number of unique categories, the most frequent category (top), and its frequency.

action_taken_name: This column represents the action taken for the loan application. It has 8 unique values, and "Loan originated" is the most frequent action with a frequency of 55815.

2.3.2.2. Proportion (Relative Frequency) Statistics

Variable	Frequency
state_name	Washington: 100.00%
state_abbr	WA: 100.00%

Variable	Frequency
respondent_id	32489: 8.34%
purchaser_type_name	Loan was not originated or was not sold in cal…
property_type_name	One-to-four family dwelling (other than manufa…
preapproval_name	Not applicable: 79.72%
owner_occupancy_name	Owner-occupied as a principal dwelling: 89.90%
msamd_name	Seattle, Bellevue, Everett - WA: 34.80%
loan_type_name	Conventional: 71.53%
loan_purpose_name	Refinancing: 47.63%
lien_status_name	Secured by a first lien: 95.08%
hoepa_status_name	Not a HOEPA loan: 99.99%
county_name	King County: 21.56%
co_applicant_sex_name	No co-applicant: 44.98%
co_applicant_ethnicity_name	No co-applicant: 44.98%
applicant_sex_name	Male: 61.78%
applicant_ethnicity_name	Not Hispanic or Latino: 73.36%
agency_name	Department of Housing and Urban Development: 4…
agency_abbr	HUD: 45.86%
action_taken_name	Loan originated: 93.03%

2.3.3. Descriptive Statistics: Input Non-Categorical Variables or Features 2.3.3.1. Measures of Central Tendency

Variable	Count	Mean/Std/Min/25%/50%/75%/Max
tract_to_msamd_income	59878	Mean: 107.62, Std: 28.23, Min: 14.05, 25%: 88.97, 50%: 105.55, 75%: 123.33, Max: 257.14
population	59878	Mean: 5278.78, Std: 1716.10, Min: 98.00, 25%: 4070.00, 50%: 5145.00, 75%: 6382.00, Max: 13025.00
minority_population	59878	Mean: 23.24, Std: 14.42, Min: 2.04, 25%: 12.95, 50%: 19.42, 75%: 29.68, Max: 94.79
number_of_owner_occupied_units	59876	Mean: 1399.04, Std: 518.33, Min: 15.00, 25%: 1034.00, 50%: 1359.00, 75%: 1722.00, Max: 2997.00
number_of_1_to_4_family_units	59878	Mean: 1873.28, Std: 738.51, Min: 27.00, 25%: 1414.00, 50%: 1770.00, 75%: 2249.00, Max: 5893.00
loan_amount_000s	60000	Mean: 291.36, Std: 604.96, Min: 1.00, 25%: 170.00, 50%: 242.00, 75%: 337.00, Max: 55000.00

Variable	Count	Mean/Std/Min/25%/50%/75%/Max
hud_median_family_income	59878	Mean: 73869.41, Std: 12811.24, Min: 48700.00, 25%: 63100.00, 50%: 73300.00, 75%: 90300.00, Max: 90300.00
applicant_income_000s	53630	Mean: 112.82, Std: 122.86, Min: 1.00, 25%: 61.00, 50%: 89.00, 75%: 132.00, Max: 6161.00
sequence_number	60000	Mean: 77526.47, Std: 150515.70, Min: 1.00, 25%: 3231.50, 50%: 16481.00, 75%: 72762.25, Max: 1241590.00
census_tract_number	59878	Mean: 1750.60, Std: 3359.68, Min: 1.00, 25%: 114.02, 50%: 403.02, 75%: 713.10, Max: 9757.00
as_of_year	60000	Mean: 2016.00, Std: 0.00, Min: 2016.00, 25%: 2016.00, 50%: 2016.00, 75%: 2016.00, Max: 2016.00
application_date_indicator	60000	Mean: 0.03, Std: 0.23, Min: 0.00, 25%: 0.00, 50%: 0.00, 75%: 0.00, Max: 2.00

2.3.3.3. Correlation Statistics (with Test of Correlation)

3. Analysis of Data

3.1. Data Pre-Processing

3.1.1. Missing Data Statistics and Treatment

3.1.1.1.1. Missing Data Statistics: Records

- Number of rows with missing data: 60000
- Number of rows with more than 50% missing data: 0

3.1.1.1.2. Missing Data Treatment: Records

3.1.1.1.2.1. Removal of Records with More Than 50% Missing Data: None | R1, R2, …

No rows with more than 50% missing values

3.1.1.2.1. Missing Data Statistics: Categorical Variables or Features

Variable	Missing Records	Percentage Missing
denial_reason_name_3	59999	99.998333
denial_reason_name_2	59957	99.928333
denial_reason_name_1	59882	99.803333
rate_spread	58134	96.890000

Variable	Missing Records	Percentage Missing
edit_status_name	47549	79.248333
msamd_name	8381	13.968333
applicant_income_000s	6370	10.616667
number_of_owner_occupied_units	124	0.206667
census_tract_number	122	0.203333
tract_to_msamd_income	122	0.203333
hud_median_family_income	122	0.203333
number_of_1_to_4_family_units	122	0.203333
minority_population	122	0.203333
population	122	0.203333
county_name	92	0.153333

3.1.1.2.2. Missing Data Treatment: Categorical Variables or Features

3.1.1.2.2.1. Removal of Variables or Features with More Than 50% Missing Data: None | CV1, CV2, …

Removed the below columns as they have more than 50% data missing

• denial_reason_name_3 • denial_reason_name_2 • denial_reason_name_1 • rate_spread (non-cat) • edit_status_name

3.1.1.2.2.2. Imputation of Missing Data using Descriptive Statistics: Mode

3.1.1.3.1. Missing Data Statistics: Non-Categorical Variables or Features

Feature	Missing Records
tract_to_msamd_income	122
population	122
minority_population	122
number_of_owner_occupied_units	124
number_of_1_to_4_family_units	122
loan_amount_000s	0
hud_median_family_income	122
applicant_income_000s	6370
sequence_number	0
census_tract_number	122
as_of_year	0
application_date_indicator	0

3.1.1.3.2. Missing Data Treatment: Non-Categorical Variables or Features

3.1.1.3.2.1. Removal of Variables or Features with More Than 50% Missing Data: None | NCV1, NCV2, …

- rate_spread

3.1.1.3.2.2. Imputation of Missing Data using Descriptive Statistics: Mean | Median

- Imputing the missing values using mean

3.1.2. Numerical Encoding of Categorical Variables or Features (Encoding Schema - Alphanumeric Order)

(Encoding Schema - Alphanumeric Order)

Feature	Number of Unique Values
state_name	1
state_abbr	1
respondent_id	593
purchaser_type_name	10
property_type_name	3
preapproval_name	3
owner_occupancy_name	3
msamd_name	14
loan_type_name	4
loan_purpose_name	3
lien_status_name	4
hoepa_status_name	2
county_name	39
co_applicant_sex_name	5
co_applicant_ethnicity_name	5
applicant_sex_name	4
applicant_ethnicity_name	4
agency_name	6
agency_abbr	6
action_taken_name	8

We are converting the above variables into numeric format in the alpha numeric order

3.1.3. Outlier Statistics and Treatment (Scaling | Transformation)

3.1.3.1.1. Outlier Statistics: Non-Categorical Variables or Features

Outliers count for the Non-Categorical Variables

Feature	Number of Unique Values
tract_to_msamd_income	1309
population	553
minority_population	2641
number_of_owner_occupied_units	478
number_of_1_to_4_family_units	2150
loan_amount_000s	2467
hud_median_family_income	0
applicant_income_000s	3765
sequence_number	7898
census_tract_number	9391
as_of_year	0

Feature	Number of Unique Values
application_date_indicator	776

3.1.3.1.2. Outlier Treatment: Non-Categorical Variables or Features

3.1.3.1.2.1. Standardization: OV1, OV2, …

3.1.3.1.2.2. Normalization using Min-Max Scaler: OV3, OV4, …

3.1.3.1.2.3. Log Transformation: OV5, OV6, …

I performed scaling using normalization using min-max scaler. But post the scaling, bubbles were still visible in the box plot. This signifies that the outliers present in the non categorical datset are not heavily influenced by the scaling method. The count of outliers seems consistent across different scaling methods.

3.1.4. Data Bifurcation: Training & Testing Sets

[Bifurcation Schema: Random Sampling or Stratified Sampling (Based on Outcome Variable or Feature) with {70% | 75% | 80%} Data in Training Set and {30% | 25% | 20%} Data in Testing Set]

The dataset was systematically divided into two distinct subsets: a training set and a testing set. This division is crucial for evaluating the performance and generalization of machine learning models.

Bifurcation Schema Sampling Technique: Stratified Sampling Ratio: 80% of the data in the training set and 20% in the testing set Stratified Sampling Stratified sampling was employed to ensure that each subset (training and testing) maintains the same proportion of classes as the original dataset. This approach is particularly beneficial when dealing with imbalanced datasets or when preserving class representation is essential for model training and evaluation.

The 80-20 split ratio was chosen to allocate a significant portion of the data to the training set (80%), allowing models to learn from a substantial amount of information while retaining a separate testing set (20%) for unbiased evaluation and validation.

---

3 LOADING LIBRARIES

[1]:
```
# 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
```

```python
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
```

```
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

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


# 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 sklearn.model_selection import StratifiedShuffleSplit
```

```
Collecting ipython-autotime
  Downloading ipython_autotime-0.3.2-py2.py3-none-any.whl (7.0 kB)
Requirement already satisfied: ipython in /usr/local/lib/python3.10/dist-
packages (from ipython-autotime) (7.34.0)
Requirement already satisfied: setuptools>=18.5 in
```

```
/usr/local/lib/python3.10/dist-packages (from ipython->ipython-autotime)
(67.7.2)
Collecting jedi>=0.16 (from ipython->ipython-autotime)
  Downloading jedi-0.19.1-py2.py3-none-any.whl (1.6 MB)
                           1.6/1.6 MB
18.6 MB/s eta 0:00:00
Requirement already satisfied: decorator in
/usr/local/lib/python3.10/dist-packages (from ipython->ipython-autotime) (4.4.2)
Requirement already satisfied: pickleshare in /usr/local/lib/python3.10/dist-
packages (from ipython->ipython-autotime) (0.7.5)
Requirement already satisfied: traitlets>=4.2 in /usr/local/lib/python3.10/dist-
packages (from ipython->ipython-autotime) (5.7.1)
Requirement already satisfied: prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0 in
/usr/local/lib/python3.10/dist-packages (from ipython->ipython-autotime)
(3.0.43)
Requirement already satisfied: pygments in /usr/local/lib/python3.10/dist-
packages (from ipython->ipython-autotime) (2.16.1)
Requirement already satisfied: backcall in /usr/local/lib/python3.10/dist-
packages (from ipython->ipython-autotime) (0.2.0)
Requirement already satisfied: matplotlib-inline in
/usr/local/lib/python3.10/dist-packages (from ipython->ipython-autotime) (0.1.6)
Requirement already satisfied: pexpect>4.3 in /usr/local/lib/python3.10/dist-
packages (from ipython->ipython-autotime) (4.9.0)
Requirement already satisfied: parso<0.9.0,>=0.8.3 in
/usr/local/lib/python3.10/dist-packages (from jedi>=0.16->ipython->ipython-
autotime) (0.8.4)
Requirement already satisfied: ptyprocess>=0.5 in
/usr/local/lib/python3.10/dist-packages (from pexpect>4.3->ipython->ipython-
autotime) (0.7.0)
Requirement already satisfied: wcwidth in /usr/local/lib/python3.10/dist-
packages (from prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0->ipython->ipython-
autotime) (0.2.13)
Installing collected packages: jedi, ipython-autotime
Successfully installed ipython-autotime-0.3.2 jedi-0.19.1
Collecting memory-profiler
  Downloading memory_profiler-0.61.0-py3-none-any.whl (31 kB)
Requirement already satisfied: psutil in /usr/local/lib/python3.10/dist-packages
(from memory-profiler) (5.9.5)
Installing collected packages: memory-profiler
Successfully installed memory-profiler-0.61.0
time: 17.6 s (started: 2024-04-13 14:05:33 +00:00)
```

4 Uploading of Data

[2]:
```
import pandas as pd
import gdown

# Google Drive file ID
file_id = '13fp1-YgAuSiR_bWZwetJiEcJZOeDeAtu'

# Downloading the CSV file from Google Drive
url = f'https://drive.google.com/uc?id={file_id}'
csv_file_path = 'Washington_State_HDMA_Dataset with Cluster Label'
gdown.download(url, csv_file_path, quiet=False)

# Read the CSV file into a pandas DataFrame
df = pd.read_csv(csv_file_path)

# Display the first few rows of the DataFrame to verify the data
print(df.head())
```

```
Downloading…
From: https://drive.google.com/uc?id=13fp1-YgAuSiR_bWZwetJiEcJZOeDeAtu
To: /content/Washington_State_HDMA_Dataset with Cluster Label
100%|      | 32.6M/32.6M [00:00<00:00, 88.4MB/s]

   S.no  tract_to_msamd_income  rate_spread  population  minority_population  \
0     1             121.690002          NaN      8381.0            23.790001
1     2              83.370003          NaN      4915.0            23.990000
2     3              91.129997          NaN      5075.0            11.820000
3     4             146.169998          NaN      5032.0             8.590000
4     5             162.470001          NaN      5183.0            10.500000

   number_of_owner_occupied_units  number_of_1_to_4_family_units  \
0                          2175.0                         2660.0
1                          1268.0                         1777.0
2                          1136.0                         1838.0
3                          1525.0                         1820.0
4                          1705.0                         2104.0

   loan_amount_000s  hud_median_family_income  applicant_income_000s  …  \
0               227                   73300.0                  116.0  …
1               240                   57900.0                   42.0  …
2               241                   73300.0                  117.0  …
3               351                   73300.0                  315.0  …
4               417                   78100.0                  114.0  …

                        co_applicant_ethnicity_name census_tract_number  \
0                            Not Hispanic or Latino              413.27
1                                   No co-applicant             9208.01
```

```
2                              Not Hispanic or Latino              414.00
3  Information not provided by applicant in mail,…              405.10
4                              Not Hispanic or Latino              907.00

   as_of_year application_date_indicator applicant_sex_name  \
0        2016                          0             Female
1        2016                          0               Male
2        2016                          0               Male
3        2016                          0               Male
4        2016                          0             Female

                          applicant_ethnicity_name  \
0                           Not Hispanic or Latino
1                               Hispanic or Latino
2                           Not Hispanic or Latino
3  Information not provided by applicant in mail,…
4                           Not Hispanic or Latino

                                  agency_name agency_abbr action_taken_name  \
0         Consumer Financial Protection Bureau        CFPB   Loan originated
1  Department of Housing and Urban Development         HUD   Loan originated
2  Department of Housing and Urban Development         HUD   Loan originated
3         National Credit Union Administration       NCUA   Loan originated
4        Federal Deposit Insurance Corporation       FDIC   Loan originated

  Cluster_Label
0             4
1             3
2             4
3             4
4             4

[5 rows x 39 columns]
time: 4.31 s (started: 2024-04-13 14:05:51 +00:00)
```

```
<ipython-input-2-f0f2a90db5ca>:13: DtypeWarning: Columns (24,25,26) have mixed
types. Specify dtype option on import or set low_memory=False.
  df = pd.read_csv(csv_file_path)
```

[3]:

```python
df.info()
list(df.columns)

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

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

```python
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: 60000 entries, 0 to 59999
Data columns (total 39 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   S.no                            60000 non-null  int64
 1   tract_to_msamd_income           59878 non-null  float64
 2   rate_spread                     1866 non-null   float64
 3   population                      59878 non-null  float64
 4   minority_population             59878 non-null  float64
 5   number_of_owner_occupied_units  59876 non-null  float64
 6   number_of_1_to_4_family_units   59878 non-null  float64
 7   loan_amount_000s                60000 non-null  int64
 8   hud_median_family_income        59878 non-null  float64
 9   applicant_income_000s           53630 non-null  float64
 10  state_name                      60000 non-null  object
 11  state_abbr                      60000 non-null  object
 12  sequence_number                 60000 non-null  int64
 13  respondent_id                   60000 non-null  object
 14  purchaser_type_name             60000 non-null  object
 15  property_type_name              60000 non-null  object
 16  preapproval_name                60000 non-null  object
 17  owner_occupancy_name            60000 non-null  object
 18  msamd_name                      51619 non-null  object
 19  loan_type_name                  60000 non-null  object
 20  loan_purpose_name               60000 non-null  object
 21  lien_status_name                60000 non-null  object
 22  hoepa_status_name               60000 non-null  object
 23  edit_status_name                12451 non-null  object
 24  denial_reason_name_3            1 non-null      object
```

```
 25  denial_reason_name_2           43 non-null     object
 26  denial_reason_name_1           118 non-null    object
 27  county_name                    59908 non-null  object
 28  co_applicant_sex_name          60000 non-null  object
 29  co_applicant_ethnicity_name    60000 non-null  object
 30  census_tract_number            59878 non-null  float64
 31  as_of_year                     60000 non-null  int64
 32  application_date_indicator     60000 non-null  int64
 33  applicant_sex_name             60000 non-null  object
 34  applicant_ethnicity_name       60000 non-null  object
 35  agency_name                    60000 non-null  object
 36  agency_abbr                    60000 non-null  object
 37  action_taken_name              60000 non-null  object
 38  Cluster_Label                  60000 non-null  int64
dtypes: float64(9), int64(6), object(24)
memory usage: 17.9+ MB
Total number of records: 60000
Total number of filled cells: 2038780
S.no                              60000
tract_to_msamd_income              1327
rate_spread                         320
population                         1283
minority_population                1216
number_of_owner_occupied_units      996
number_of_1_to_4_family_units      1051
loan_amount_000s                   1344
hud_median_family_income             15
applicant_income_000s               807
state_name                            1
state_abbr                            1
sequence_number                   39340
respondent_id                       593
purchaser_type_name                  10
property_type_name                    3
preapproval_name                      3
owner_occupancy_name                  3
msamd_name                           14
loan_type_name                        4
loan_purpose_name                     3
lien_status_name                      4
hoepa_status_name                     2
edit_status_name                      1
denial_reason_name_3                  1
denial_reason_name_2                  8
denial_reason_name_1                  8
county_name                          39
co_applicant_sex_name                 5
co_applicant_ethnicity_name           5
```

```
census_tract_number              1108
as_of_year                          1
application_date_indicator          2
applicant_sex_name                  4
applicant_ethnicity_name            4
agency_name                         6
agency_abbr                         6
action_taken_name                   8
Cluster_Label                       5
dtype: int64
time: 351 ms (started: 2024-04-13 14:05:55 +00:00)
```

[4]:

```
# Importing necessary libraries
import pandas as pd

# Assuming df is your DataFrame and 'Cluster_Label' is the column of interest

# Calculate relative frequencies
relative_frequencies = df['Cluster_Label'].value_counts(normalize=True)

# Print relative frequencies
print(relative_frequencies)

# Assuming df is your DataFrame and 'Cluster_Label' is the column of interest

# Count the occurrences of each unique value
unique_value_counts = df['Cluster_Label'].value_counts()

# Print the count of unique values
print(unique_value_counts)
```

```
Cluster_Label
3    0.236033
1    0.234150
0    0.190583
4    0.179050
2    0.160183
Name: proportion, dtype: float64
Cluster_Label
3    14162
1    14049
0    11435
4    10743
2     9611
Name: count, dtype: int64
time: 10.4 ms (started: 2024-04-13 14:05:55 +00:00)
```

[5]:
```python
# Assuming df is your DataFrame
columns_list = df.columns.tolist()
columns_list

list(df.columns)
```

[5]:
```
['S.no',
 'tract_to_msamd_income',
 'rate_spread',
 'population',
 'minority_population',
 'number_of_owner_occupied_units',
 'number_of_1_to_4_family_units',
 'loan_amount_000s',
 'hud_median_family_income',
 'applicant_income_000s',
 'state_name',
 'state_abbr',
 'sequence_number',
 'respondent_id',
 'purchaser_type_name',
 'property_type_name',
 'preapproval_name',
 'owner_occupancy_name',
 'msamd_name',
 'loan_type_name',
 'loan_purpose_name',
 'lien_status_name',
 'hoepa_status_name',
 'edit_status_name',
 'denial_reason_name_3',
 'denial_reason_name_2',
 'denial_reason_name_1',
 'county_name',
 'co_applicant_sex_name',
 'co_applicant_ethnicity_name',
 'census_tract_number',
 'as_of_year',
 'application_date_indicator',
 'applicant_sex_name',
 'applicant_ethnicity_name',
 'agency_name',
 'agency_abbr',
 'action_taken_name',
 'Cluster_Label']
```

```
time: 4.4 ms (started: 2024-04-13 14:05:55 +00:00)
```

[6]:
```
# 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', 'tract_to_msamd_income', 'rate_spread',
'population', 'minority_population', 'number_of_owner_occupied_units',
'number_of_1_to_4_family_units', 'loan_amount_000s', 'hud_median_family_income',
'applicant_income_000s', 'sequence_number', 'census_tract_number', 'as_of_year',
'application_date_indicator', 'Cluster_Label']
Non-Continuous Columns: ['state_name', 'state_abbr', 'respondent_id',
'purchaser_type_name', 'property_type_name', 'preapproval_name',
'owner_occupancy_name', 'msamd_name', 'loan_type_name', 'loan_purpose_name',
'lien_status_name', 'hoepa_status_name', 'edit_status_name',
'denial_reason_name_3', 'denial_reason_name_2', 'denial_reason_name_1',
'county_name', 'co_applicant_sex_name', 'co_applicant_ethnicity_name',
'applicant_sex_name', 'applicant_ethnicity_name', 'agency_name', 'agency_abbr',
'action_taken_name']
Categorical Columns: ['state_name', 'state_abbr', 'respondent_id',
'purchaser_type_name', 'property_type_name', 'preapproval_name',
'owner_occupancy_name', 'msamd_name', 'loan_type_name', 'loan_purpose_name',
'lien_status_name', 'hoepa_status_name', 'edit_status_name',
'denial_reason_name_3', 'denial_reason_name_2', 'denial_reason_name_1',
'county_name', 'co_applicant_sex_name', 'co_applicant_ethnicity_name',
'applicant_sex_name', 'applicant_ethnicity_name', 'agency_name', 'agency_abbr',
'action_taken_name']
Non-Categorical Columns: ['S.no', 'tract_to_msamd_income', 'rate_spread',
'population', 'minority_population', 'number_of_owner_occupied_units',
'number_of_1_to_4_family_units', 'loan_amount_000s', 'hud_median_family_income',
```

```
'applicant_income_000s', 'sequence_number', 'census_tract_number', 'as_of_year',
'application_date_indicator', 'Cluster_Label']
time: 34.9 ms (started: 2024-04-13 14:05:55 +00:00)
```

[7]:
```python
###  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  \
denial_reason_name_3                   denial_reason_name_3
denial_reason_name_2                   denial_reason_name_2
denial_reason_name_1                   denial_reason_name_1
rate_spread                                     rate_spread
edit_status_name                           edit_status_name
msamd_name                                       msamd_name
applicant_income_000s                 applicant_income_000s
number_of_owner_occupied_units  number_of_owner_occupied_units
tract_to_msamd_income                 tract_to_msamd_income
hud_median_family_income           hud_median_family_income
number_of_1_to_4_family_units    number_of_1_to_4_family_units
minority_population                     minority_population
population                                       population
census_tract_number                     census_tract_number
county_name                                     county_name
co_applicant_ethnicity_name         co_applicant_ethnicity_name
co_applicant_sex_name                 co_applicant_sex_name
S.no                                                   S.no
as_of_year                                       as_of_year
application_date_indicator       application_date_indicator
applicant_sex_name                       applicant_sex_name
```

agency_name	agency_name
agency_abbr	agency_abbr
action_taken_name	action_taken_name
applicant_ethnicity_name	applicant_ethnicity_name
loan_type_name	loan_type_name
hoepa_status_name	hoepa_status_name
lien_status_name	lien_status_name
loan_purpose_name	loan_purpose_name
owner_occupancy_name	owner_occupancy_name
preapproval_name	preapproval_name
property_type_name	property_type_name
purchaser_type_name	purchaser_type_name
respondent_id	respondent_id
sequence_number	sequence_number
state_abbr	state_abbr
state_name	state_name
loan_amount_000s	loan_amount_000s
Cluster_Label	Cluster_Label

	Missing Records	Percentage Missing
denial_reason_name_3	59999	99.998333
denial_reason_name_2	59957	99.928333
denial_reason_name_1	59882	99.803333
rate_spread	58134	96.890000
edit_status_name	47549	79.248333
msamd_name	8381	13.968333
applicant_income_000s	6370	10.616667
number_of_owner_occupied_units	124	0.206667
tract_to_msamd_income	122	0.203333
hud_median_family_income	122	0.203333
number_of_1_to_4_family_units	122	0.203333
minority_population	122	0.203333
population	122	0.203333
census_tract_number	122	0.203333
county_name	92	0.153333
co_applicant_ethnicity_name	0	0.000000
co_applicant_sex_name	0	0.000000
S.no	0	0.000000
as_of_year	0	0.000000
application_date_indicator	0	0.000000
applicant_sex_name	0	0.000000
agency_name	0	0.000000
agency_abbr	0	0.000000
action_taken_name	0	0.000000
applicant_ethnicity_name	0	0.000000
loan_type_name	0	0.000000
hoepa_status_name	0	0.000000
lien_status_name	0	0.000000

```
loan_purpose_name                          0            0.000000
owner_occupancy_name                       0            0.000000
preapproval_name                           0            0.000000
property_type_name                         0            0.000000
purchaser_type_name                        0            0.000000
respondent_id                              0            0.000000
sequence_number                            0            0.000000
state_abbr                                 0            0.000000
state_name                                 0            0.000000
loan_amount_000s                           0            0.000000
Cluster_Label                              0            0.000000
time: 79.6 ms (started: 2024-04-13 14:05:55 +00:00)
```

[8]:

```python
# List of columns to drop
columns_to_drop = ['state_name', 'state_abbr','denial_reason_name_3',␣
 ↪'denial_reason_name_2', 'denial_reason_name_1', 'rate_spread',␣
 ↪'edit_status_name', 'as_of_year']

# 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  \
msamd_name                                         msamd_name
applicant_income_000s                   applicant_income_000s
number_of_owner_occupied_units  number_of_owner_occupied_units
census_tract_number                       census_tract_number
population                                         population
minority_population                       minority_population
number_of_1_to_4_family_units    number_of_1_to_4_family_units
```

tract_to_msamd_income	tract_to_msamd_income
hud_median_family_income	hud_median_family_income
county_name	county_name
agency_name	agency_name
applicant_ethnicity_name	applicant_ethnicity_name
applicant_sex_name	applicant_sex_name
application_date_indicator	application_date_indicator
hoepa_status_name	hoepa_status_name
action_taken_name	action_taken_name
co_applicant_ethnicity_name	co_applicant_ethnicity_name
co_applicant_sex_name	co_applicant_sex_name
agency_abbr	agency_abbr
S.no	S.no
lien_status_name	lien_status_name
loan_purpose_name	loan_purpose_name
loan_type_name	loan_type_name
owner_occupancy_name	owner_occupancy_name
preapproval_name	preapproval_name
property_type_name	property_type_name
purchaser_type_name	purchaser_type_name
respondent_id	respondent_id
sequence_number	sequence_number
loan_amount_000s	loan_amount_000s
Cluster_Label	Cluster_Label

	Missing Records	Percentage Missing
msamd_name	8381	13.968333
applicant_income_000s	6370	10.616667
number_of_owner_occupied_units	124	0.206667
census_tract_number	122	0.203333
population	122	0.203333
minority_population	122	0.203333
number_of_1_to_4_family_units	122	0.203333
tract_to_msamd_income	122	0.203333
hud_median_family_income	122	0.203333
county_name	92	0.153333
agency_name	0	0.000000
applicant_ethnicity_name	0	0.000000
applicant_sex_name	0	0.000000
application_date_indicator	0	0.000000
hoepa_status_name	0	0.000000
action_taken_name	0	0.000000
co_applicant_ethnicity_name	0	0.000000
co_applicant_sex_name	0	0.000000
agency_abbr	0	0.000000
S.no	0	0.000000
lien_status_name	0	0.000000
loan_purpose_name	0	0.000000

```
loan_type_name                              0            0.000000
owner_occupancy_name                        0            0.000000
preapproval_name                            0            0.000000
property_type_name                          0            0.000000
purchaser_type_name                         0            0.000000
respondent_id                               0            0.000000
sequence_number                             0            0.000000
loan_amount_000s                            0            0.000000
Cluster_Label                               0            0.000000
time: 68.4 ms (started: 2024-04-13 14:05:55 +00:00)
```

[9]:
```
### 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: 13629
Number of rows with more than 50% missing data: 0
time: 126 ms (started: 2024-04-13 14:05:55 +00:00)
```

[10]:
```
# 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)
```

```
['respondent_id', 'purchaser_type_name', 'property_type_name',
'preapproval_name', 'owner_occupancy_name', 'msamd_name', 'loan_type_name',
'loan_purpose_name', 'lien_status_name', 'hoepa_status_name', 'county_name',
'co_applicant_sex_name', 'co_applicant_ethnicity_name', 'applicant_sex_name',
'applicant_ethnicity_name', 'agency_name', 'agency_abbr', 'action_taken_name']
['S.no', 'tract_to_msamd_income', 'population', 'minority_population',
'number_of_owner_occupied_units', 'number_of_1_to_4_family_units',
'loan_amount_000s', 'hud_median_family_income', 'applicant_income_000s',
'sequence_number', 'census_tract_number', 'application_date_indicator',
'Cluster_Label']
time: 27.4 ms (started: 2024-04-13 14:05:56 +00:00)
```

5 PreProcessing of Data

[11]:
```
# Data Bifurcation
catdf = df[['S.no', 'respondent_id', 'purchaser_type_name',␣
↪'property_type_name', 'preapproval_name', 'owner_occupancy_name',␣
↪'msamd_name', 'loan_type_name', 'loan_purpose_name',
           'lien_status_name', 'hoepa_status_name', 'county_name',␣
↪'co_applicant_sex_name', 'co_applicant_ethnicity_name',␣
↪'applicant_sex_name', 'applicant_ethnicity_name', 'agency_name',
           'agency_abbr', 'action_taken_name']] # Categorical Data [Nominal |␣
↪Ordinal]

noncatdf = df[['S.no', 'tract_to_msamd_income', 'population',␣
↪'minority_population', 'number_of_owner_occupied_units',␣
↪'number_of_1_to_4_family_units', 'loan_amount_000s',
              'hud_median_family_income', 'applicant_income_000s',␣
↪'sequence_number', 'census_tract_number', 'application_date_indicator',␣
↪'Cluster_Label']] # Non-Categorical Data
```

```
time: 23.8 ms (started: 2024-04-13 14:05:56 +00:00)
```

[12]:
```
####  STATISTICS OF CAT DATASET

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

```python
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  \
S.no                        60000.0    NaN
respondent_id                 60000    593
purchaser_type_name           60000     10
property_type_name            60000      3
preapproval_name              60000      3
owner_occupancy_name          60000      3
msamd_name                    51619     14
loan_type_name                60000      4
loan_purpose_name             60000      3
lien_status_name              60000      4
hoepa_status_name             60000      2
county_name                   59908     39
co_applicant_sex_name         60000      5
co_applicant_ethnicity_name   60000      5
applicant_sex_name            60000      4
applicant_ethnicity_name      60000      4
agency_name                   60000      6
agency_abbr                   60000      6
action_taken_name             60000      8

                                                                           top
\
S.no                                                                       NaN
```

respondent_id	32489
purchaser_type_name	Loan was not originated or was not sold in cal…
property_type_name	One-to-four family dwelling (other than manufa…
preapproval_name	Not applicable
owner_occupancy_name	Owner-occupied as a principal dwelling
msamd_name	Seattle, Bellevue, Everett - WA
loan_type_name	Conventional
loan_purpose_name	Refinancing
lien_status_name	Secured by a first lien
hoepa_status_name	Not a HOEPA loan
county_name	King County
co_applicant_sex_name	No co-applicant
co_applicant_ethnicity_name	No co-applicant
applicant_sex_name	Male
applicant_ethnicity_name	Not Hispanic or Latino
agency_name	Department of Housing and Urban Development
agency_abbr	HUD
action_taken_name	Loan originated

	freq	mean	std	min	25%	\
S.no	NaN	30000.5	17320.652413	1.0	15000.75	
respondent_id	5006	NaN	NaN	NaN	NaN	
purchaser_type_name	16112	NaN	NaN	NaN	NaN	
property_type_name	57630	NaN	NaN	NaN	NaN	
preapproval_name	47832	NaN	NaN	NaN	NaN	
owner_occupancy_name	53940	NaN	NaN	NaN	NaN	
msamd_name	17965	NaN	NaN	NaN	NaN	
loan_type_name	42917	NaN	NaN	NaN	NaN	
loan_purpose_name	28576	NaN	NaN	NaN	NaN	
lien_status_name	57046	NaN	NaN	NaN	NaN	
hoepa_status_name	59996	NaN	NaN	NaN	NaN	
county_name	12915	NaN	NaN	NaN	NaN	
co_applicant_sex_name	26987	NaN	NaN	NaN	NaN	
co_applicant_ethnicity_name	26987	NaN	NaN	NaN	NaN	
applicant_sex_name	37070	NaN	NaN	NaN	NaN	
applicant_ethnicity_name	44014	NaN	NaN	NaN	NaN	
agency_name	27514	NaN	NaN	NaN	NaN	
agency_abbr	27514	NaN	NaN	NaN	NaN	
action_taken_name	55815	NaN	NaN	NaN	NaN	

	50%	75%	max
S.no	30000.5	45000.25	60000.0
respondent_id	NaN	NaN	NaN
purchaser_type_name	NaN	NaN	NaN
property_type_name	NaN	NaN	NaN
preapproval_name	NaN	NaN	NaN
owner_occupancy_name	NaN	NaN	NaN
msamd_name	NaN	NaN	NaN

```
loan_type_name                  NaN       NaN       NaN
loan_purpose_name               NaN       NaN       NaN
lien_status_name                NaN       NaN       NaN
hoepa_status_name               NaN       NaN       NaN
county_name                     NaN       NaN       NaN
co_applicant_sex_name           NaN       NaN       NaN
co_applicant_ethnicity_name     NaN       NaN       NaN
applicant_sex_name              NaN       NaN       NaN
applicant_ethnicity_name        NaN       NaN       NaN
agency_name                     NaN       NaN       NaN
agency_abbr                     NaN       NaN       NaN
action_taken_name               NaN       NaN       NaN
time: 420 ms (started: 2024-04-13 14:05:56 +00:00)
```

[13]:

```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  tract_to_msamd_income    population  minority_population  \
count  60000.000000           59878.000000  59878.000000         59878.000000
mean   30000.500000             107.617351   5278.782157            23.244442
std    17320.652413              28.233471   1716.101490            14.416209
min        1.000000              14.050000     98.000000             2.040000
25%    15000.750000              88.970001   4070.000000            12.950000
50%    30000.500000             105.550003   5145.000000            19.420000
75%    45000.250000             123.330002   6382.000000            29.680000
max    60000.000000             257.140015  13025.000000            94.790001

       number_of_owner_occupied_units  number_of_1_to_4_family_units  \
count                    59876.000000                   59878.000000
mean                      1399.044375                    1873.281456
std                        518.330561                     738.505184
min                         15.000000                      27.000000
25%                       1034.000000                    1414.000000
50%                       1359.000000                    1770.000000
75%                       1722.000000                    2249.000000
max                       2997.000000                    5893.000000

       loan_amount_000s  hud_median_family_income  applicant_income_000s  \
count      60000.000000              59878.000000           53630.000000
mean         291.358717              73869.411136             112.822301
std          604.958183              12811.243390             122.862496
min            1.000000              48700.000000               1.000000
25%          170.000000              63100.000000              61.000000
```

```
50%         242.000000            73300.000000             89.000000
75%         337.000000            90300.000000            132.000000
max       55000.000000            90300.000000           6161.000000

       sequence_number  census_tract_number  application_date_indicator  \
count     6.000000e+04         59878.000000                60000.000000
mean      7.752647e+04          1750.597252                    0.025867
std       1.505157e+05          3359.676740                    0.225976
min       1.000000e+00             1.000000                    0.000000
25%       3.231500e+03           114.020000                    0.000000
50%       1.648100e+04           403.020000                    0.000000
75%       7.276225e+04           713.100000                    0.000000
max       1.241590e+06          9757.000000                    2.000000

       Cluster_Label
count   60000.000000
mean        1.978817
std         1.395815
min         0.000000
25%         1.000000
50%         2.000000
75%         3.000000
max         4.000000
time: 72.8 ms (started: 2024-04-13 14:05:56 +00:00)
```

[14]:
```python
# 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            tract_to_msamd_income              122
2                      population              122
3              minority_population              122
4    number_of_owner_occupied_units              124
5     number_of_1_to_4_family_units              122
6                 loan_amount_000s                0
7           hud_median_family_income              122
8             applicant_income_000s             6370
9                  sequence_number                0
10             census_tract_number              122
11      application_date_indicator                0
12                   Cluster_Label                0
```

```
time: 8.21 ms (started: 2024-04-13 14:05:56 +00:00)
```

[15]:
```
#  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: 60000 entries, 0 to 59999
Data columns (total 13 columns):
 #   Column                         Non-Null Count  Dtype
---  ------                         --------------  -----
 0   S.no                           60000 non-null  float64
 1   tract_to_msamd_income          60000 non-null  float64
 2   population                     60000 non-null  float64
 3   minority_population            60000 non-null  float64
 4   number_of_owner_occupied_units 60000 non-null  float64
 5   number_of_1_to_4_family_units  60000 non-null  float64
 6   loan_amount_000s               60000 non-null  float64
 7   hud_median_family_income       60000 non-null  float64
 8   applicant_income_000s          60000 non-null  float64
 9   sequence_number                60000 non-null  float64
 10  census_tract_number            60000 non-null  float64
 11  application_date_indicator     60000 non-null  float64
 12  Cluster_Label                  60000 non-null  float64
dtypes: float64(13)
memory usage: 6.0 MB
time: 50.9 ms (started: 2024-04-13 14:05:56 +00:00)
```

[16]:
```
# 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        17320.652413
```

```
1           tract_to_msamd_income          28.204752
2                      population        1714.355870
3              minority_population          14.401545
4    number_of_owner_occupied_units        517.794667
5     number_of_1_to_4_family_units        737.753976
6                 loan_amount_000s        604.958183
7           hud_median_family_income      12798.211781
8             applicant_income_000s        116.157479
9                  sequence_number     150515.678152
10              census_tract_number       3356.259273
11        application_date_indicator          0.225976
12                    Cluster_Label          1.395815
time: 20 ms (started: 2024-04-13 14:05:56 +00:00)
```

[17]:
```python
# 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: 60000 entries, 0 to 59999
Data columns (total 19 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   S.no                        60000 non-null  object
 1   respondent_id               60000 non-null  object
 2   purchaser_type_name         60000 non-null  object
 3   property_type_name          60000 non-null  object
 4   preapproval_name            60000 non-null  object
 5   owner_occupancy_name        60000 non-null  object
 6   msamd_name                  60000 non-null  object
 7   loan_type_name              60000 non-null  object
 8   loan_purpose_name           60000 non-null  object
 9   lien_status_name            60000 non-null  object
 10  hoepa_status_name           60000 non-null  object
 11  county_name                 60000 non-null  object
 12  co_applicant_sex_name       60000 non-null  object
 13  co_applicant_ethnicity_name 60000 non-null  object
 14  applicant_sex_name          60000 non-null  object
 15  applicant_ethnicity_name    60000 non-null  object
 16  agency_name                 60000 non-null  object
 17  agency_abbr                 60000 non-null  object
 18  action_taken_name           60000 non-null  object
```

```
dtypes: object(19)
memory usage: 8.7+ MB
```

[17]:
```
   S.no respondent_id                                    purchaser_type_name  \
0     1        480228                                    Freddie Mac (FHLMC)
1     2    7257500009  Life insurance company, credit union, mortgage…
2     3    72-1545376  Loan was not originated or was not sold in cal…
3     4          4878  Loan was not originated or was not sold in cal…
4     5         32489                                    Freddie Mac (FHLMC)

                                 property_type_name preapproval_name  \
0  One-to-four family dwelling (other than manufa…   Not applicable
1  One-to-four family dwelling (other than manufa…   Not applicable
2  One-to-four family dwelling (other than manufa…   Not applicable
3  One-to-four family dwelling (other than manufa…   Not applicable
4  One-to-four family dwelling (other than manufa…   Not applicable

                       owner_occupancy_name  \
0  Owner-occupied as a principal dwelling
1  Owner-occupied as a principal dwelling
2  Owner-occupied as a principal dwelling
3  Owner-occupied as a principal dwelling
4  Owner-occupied as a principal dwelling

                                 msamd_name loan_type_name loan_purpose_name  \
0  Portland, Vancouver, Hillsboro - OR, WA   Conventional       Refinancing
1                        Walla Walla - WA    FHA-insured     Home purchase
2  Portland, Vancouver, Hillsboro - OR, WA   Conventional       Refinancing
3  Portland, Vancouver, Hillsboro - OR, WA   Conventional       Refinancing
4               Bremerton, Silverdale - WA   Conventional  Home improvement

         lien_status_name hoepa_status_name         county_name  \
0  Secured by a first lien  Not a HOEPA loan        Clark County
1  Secured by a first lien  Not a HOEPA loan  Walla Walla County
2  Secured by a first lien  Not a HOEPA loan        Clark County
3  Secured by a first lien  Not a HOEPA loan        Clark County
4  Secured by a first lien  Not a HOEPA loan       Kitsap County

  co_applicant_sex_name                           co_applicant_ethnicity_name  \
0                  Male                                Not Hispanic or Latino
1       No co-applicant                                       No co-applicant
2                Female                                Not Hispanic or Latino
3                Female  Information not provided by applicant in mail,…
4                  Male                                Not Hispanic or Latino

  applicant_sex_name                              applicant_ethnicity_name  \
0             Female                                Not Hispanic or Latino
```

```
1              Male                                 Hispanic or Latino
2              Male                             Not Hispanic or Latino
3              Male  Information not provided by applicant in mail,…
4            Female                             Not Hispanic or Latino

                                     agency_name agency_abbr action_taken_name
0         Consumer Financial Protection Bureau        CFPB   Loan originated
1   Department of Housing and Urban Development        HUD   Loan originated
2   Department of Housing and Urban Development        HUD   Loan originated
3         National Credit Union Administration       NCUA   Loan originated
4        Federal Deposit Insurance Corporation       FDIC   Loan originated
```

```
time: 818 ms (started: 2024-04-13 14:05:56 +00:00)
```

[18]:

```python
# ENCODING
# Converting Categorical Variable into Numeric

# 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", "Cluster_Label"]

# 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_)))
```

```python
# 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                    60000
1                 respondent_id                      593
2            purchaser_type_name                      10
3             property_type_name                       3
4               preapproval_name                       3
5           owner_occupancy_name                       3
6                     msamd_name                      14
7                 loan_type_name                       4
8              loan_purpose_name                       3
9               lien_status_name                       4
10             hoepa_status_name                       2
11                   county_name                      39
12          co_applicant_sex_name                      5
13    co_applicant_ethnicity_name                      5
14            applicant_sex_name                       4
15      applicant_ethnicity_name                       4
16                   agency_name                       6
17                   agency_abbr                       6
18             action_taken_name                       8

Mapping for respondent_id:
{'01-0681100': 0, '01-0726495': 1, '02-0793125': 2, '03-0488052': 3,
'04-3212636': 4, '04-3568208': 5, '04-3660901': 6, '04-7534967': 7,
'05-0402708': 8, '06-1016329': 9, '1015560': 10, '10257': 11, '1047': 12,
'1097500000': 13, '1099800006': 14, '11-3399725': 15, '11-3412303': 16,
'11-3714032': 17, '11162': 18, '112837': 19, '11443': 20, '1146500007': 21,
'11729': 22, '11734': 23, '12135': 24, '1216826': 25, '12219': 26, '1227300009':
27, '12311': 28, '1265': 29, '1281': 30, '13-3222578': 31, '13-3602661': 32,
'13-3753941': 33, '13-4225190': 34, '13-4362989': 35, '13-6131491': 36, '13232':
37, '13303': 38, '13392': 39, '13778': 40, '14-1841762': 41, '14206': 42,
'14252': 43, '143662': 44, '1461700004': 45, '14662': 46, '146672': 47, '14740':
48, '14843': 49, '151': 50, '15219': 51, '15732': 52, '16-1686740': 53, '16243':
54, '1635900004': 55, '16450': 56, '16814': 57, '169653': 58, '17587': 59,
'17672': 60, '177957': 61, '17874': 62, '17884': 63, '1842065': 64, '19307': 65,
'19628': 66, '197478': 67, '19899': 68, '19976': 69, '20-0142846': 70,
'20-0192872': 71, '20-0304793': 72, '20-0640473': 73, '20-0740151': 74,
'20-1255434': 75, '20-1832276': 76, '20-2053401': 77, '20-2355296': 78,
'20-2470783': 79, '20-2471369': 80, '20-2485875': 81, '20-2693054': 82,
```

'20-2718340': 83, '20-2752826': 84, '20-2928975': 85, '20-3702275': 86, '20-3828708': 87, '20-4136310': 88, '20-4224234': 89, '20-4255880': 90, '20-4866754': 91, '20-5238443': 92, '20-5239910': 93, '20-5741925': 94, '20-8006279': 95, '20-8083209': 96, '20-8544905': 97, '20-8745846': 98, '20-8803449': 99, '20-8921389': 100, '2003500009': 101, '20061': 102, '20068': 103, '20214': 104, '20516': 105, '20624': 106, '20774': 107, '210434': 108, '21122': 109, '212465': 110, '2137100009': 111, '2149009991': 112, '21717': 113, '2191': 114, '2193616': 115, '22-3039688': 116, '22-3470404': 117, '22-3554558': 118, '22-3626426': 119, '22-3747694': 120, '22-3887207': 121, '22134': 122, '22157': 123, '22407': 124, '22444': 125, '22637': 126, '2285': 127, '22939': 128, '23-2470039': 129, '23-2769131': 130, '23041': 131, '2317700005': 132, '23216': 133, '23416': 134, '23521': 135, '23850': 136, '23922': 137, '23957': 138, '24077': 139, '24080': 140, '24107': 141, '24169': 142, '24224': 143, '24235': 144, '24326': 145, '24382': 146, '24671': 147, '24708': 148, '24713': 149, '24719': 150, '24753': 151, '24760': 152, '24831': 153, '24849': 154, '2489805': 155, '25080': 156, '25093': 157, '25103': 158, '2562164': 159, '2590037': 160, '26-0012825': 161, '26-0021318': 162, '26-0335190': 163, '26-0360466': 164, '26-0362771': 165, '26-0423240': 166, '26-0455770': 167, '26-0508430': 168, '26-0595342': 169, '26-0707492': 170, '26-1242154': 171, '26-1334020': 172, '26-1589507': 173, '26-1773722': 174, '26-2049351': 175, '26-2261031': 176, '26-2593704': 177, '26-2689428': 178, '26-2916887': 179, '26-3264687': 180, '26-3416474': 181, '26-3780954': 182, '26-4193875': 183, '26-4461592': 184, '26-4558390': 185, '26-4599244': 186, '2618780': 187, '26610': 188, '267100004': 189, '27-0222046': 190, '27-0267182': 191, '27-0684906': 192, '27-0812934': 193, '27-1190043': 194, '27-1438405': 195, '27-2389039': 196, '27-3459350': 197, '27-4023565': 198, '27-4626537': 199, '27280': 200, '2734': 201, '2735146': 202, '27601': 203, '276579': 204, '280110': 205, '28116': 206, '28151': 207, '28316': 208, '28405': 209, '28453': 210, '28454': 211, '28489': 212, '28599': 213, '2888798': 214, '29012': 215, '29058': 216, '29209': 217, '2945': 218, '3027509990': 219, '3076248': 220, '30788': 221, '30810': 222, '31-1197926': 223, '31-1690008': 224, '31-1712553': 225, '311845': 226, '3121': 227, '31286': 228, '3150447': 229, '319': 230, '32-0016270': 231, '3212149': 232, '32178': 233, '32489': 234, '3284070': 235, '33-0231744': 236, '33-0397503': 237, '33-0419992': 238, '33-0594693': 239, '33-0750812': 240, '33-0816610': 241, '33-0828099': 242, '33-0941669': 243, '33-0962918': 244, '33-0975529': 245, '33183': 246, '33508': 247, '3374412': 248, '33806': 249, '33826': 250, '339858': 251, '34-1194858': 252, '34-1633105': 253, '34-1716542': 254, '34-1719615': 255, '34-2000096': 256, '34106': 257, '34214': 258, '34585': 259, '34607': 260, '34627': 261, '34953': 262, '35-2486440': 263, '35013': 264, '35014': 265, '35139': 266, '35261': 267, '35355': 268, '3537897': 269, '35406': 270, '36-4327855': 271, '365325': 272, '37-1493496': 273, '37-1542226': 274, '38-2434249': 275, '38-2749215': 276, '38-2750395': 277, '38-2799035': 278, '38-3564305': 279, '39-2001010': 280, '3918898': 281, '3938186': 282, '3956': 283, '4004574': 284, '41-1795868': 285, '41-1914032': 286, '41-2277737': 287, '4114567': 288, '413208': 289, '4142': 290, '42-1554181': 291, '42-1739728': 292, '4239': 293, '43-0951349': 294, '43-1965151': 295, '435': 296, '45-3143795': 297, '45-3508295': 298, '451965': 299, '46-0530020': 300, '46-1728831': 301, '46-1836968': 302, '46-2477192': 303, '46-2888046': 304, '46-3435079': 305, '46-3528270': 306, '46-3676810': 307, '46-5671661': 308, '47-0873092': 309, '47-0912342': 310, '47-0933090': 311, '47-3632618': 312, '471809999': 313, '476810': 314, '48-1148159': 315, '48-1236121': 316, '480228': 317, '4878': 318, '491224': 319, '501105': 320, '504713': 321, '51-0488301': 322, '51-0517525': 323, '52-2091594': 324, '52-2276553': 325, '52-2321476': 326, '52-2323186': 327, '5380': 328, '54-0259290': 329, '54-1094297': 330, '54-1994393': 331, '54-2070914': 332, '542409990': 333, '542649': 334, '546571': 335, '55130': 336, '5556209999': 337, '5582': 338, '5588': 339, '56-2103469': 340, '56-2237729': 341, '56-2471041': 342, '566': 343, '57-1175755': 344, '57033': 345, '57071': 346, '57074': 347, '57167': 348, '57451': 349, '57542': 350, '57607': 351, '57614': 352, '57633': 353, '57776': 354, '57777': 355, '57949': 356, '57955': 357, '57978': 358, '58-1865166': 359, '58305': 360, '58322': 361, '58341': 362, '58380': 363, '58778': 364, '59-3378746': 365, '5912': 366, '5913': 367, '592448': 368, '595270': 369, '595869': 370, '60042': 371, '60059': 372, '60079': 373, '601050': 374, '60143': 375, '60438': 376, '606046': 377, '60613': 378, '6158': 379, '6161': 380, '617677': 381, '619877': 382, '62-0997810': 383, '62-1494087': 384, '62-1532940': 385, '624': 386, '6248': 387, '62659': 388, '62665': 389, '62745': 390, '6288': 391, '63-1052225': 392, '63069': 393, '63194': 394, '63196': 395, '6328': 396, '63315': 397, '63440': 398, '63799': 399, '64103': 400, '644': 401, '6443809990': 402, '64482': 403, '64546': 404, '64552': 405, '65595': 406, '656377': 407, '65644': 408, '656733': 409, '66157': 410, '66328': 411, '66331': 412, '66337': 413, '66349': 414, '66373': 415, '66399': 416, '665': 417, '66734': 418, '66751': 419, '66841': 420, '67201': 421, '67262': 422, '67264': 423, '67389': 424, '675332': 425, '67911': 426, '67955': 427, '68-0151632': 428, '68-0295876': 429, '68-0309242': 430, '68061': 431, '68095': 432, '68186': 433, '68187': 434, '68196': 435, '68203': 436, '68205': 437, '68222': 438, '68223': 439, '68237': 440, '68239': 441, '68253': 442, '68255': 443, '68271': 444, '68278': 445, '68284': 446, '68293': 447, '68298': 448, '68304': 449, '68315': 450, '68362': 451, '68375': 452, '68394': 453, '68423': 454, '68457': 455, '68465': 456, '68517': 457, '68530': 458, '68546': 459, '685676': 460, '68576': 461, '68598': 462, '68601': 463, '68613': 464, '68622': 465, '694904': 466, '697633': 467, '702825': 468, '702889': 469, '703927': 470, '704347': 471, '7056000000': 472, '706707': 473, '706809': 474, '707674': 475, '708146': 476, '708186': 477, '7101': 478, '712504': 479, '713570': 480, '713964': 481, '714970': 482, '715018': 483, '716195': 484, '7162800002': 485, '716456': 486, '717936': 487, '7197000003': 488, '72-1545376': 489, '7257500009': 490, '7272800006': 491, '7275700004': 492, '73-1374559': 493, '73-1545233': 494, '73-1577221': 495, '74-2508160': 496, '75-2585326': 497, '75-2695327': 498, '75-2921540': 499, '75-3170028': 500, '7505400005': 501, '7516800003': 502, '75633': 503, '76-0236067': 504, '76-0503625': 505, '76-0561995': 506, '76-0629353': 507, '7635500004': 508, '7638200000': 509, '7667200009': 510, '7674000006': 511, '77-0158990': 512, '77-0605392': 513, '77-0672274': 514, '77-0717225': 515, '7748': 516, '7756300009': 517, '7810600004': 518, '7811300008': 519, '7927200007': 520, '7983500003': 521, '7992700007': 522, '80-0233937': 523, '80-0312140': 524, '80-0860209': 525, '80122': 526, '804963': 527, '8100': 528, '817824': 529, '83-0171636': 530, '83-0368862': 531, '84-0927358': 532, '84-1040263': 533, '84-1412422': 534,

'84-1496821': 535, '84-1564935': 536, '84-1594306': 537, '85-0260899': 538, '852218': 539, '852320': 540, '857': 541, '86-0415227': 542, '86-0431588': 543, '86-0634557': 544, '86-0860478': 545, '8663': 546, '87-0623581': 547, '87-0675992': 548, '87-0682600': 549, '87-0691650': 550, '8796': 551, '8797': 552, '88-0209429': 553, '88-0508228': 554, '8854': 555, '90-0790926': 556, '91-1374387': 557, '91-1395192': 558, '91-1441009': 559, '91-1465333': 560, '91-1529683': 561, '91-1569077': 562, '91-1780488': 563, '91-1841798': 564, '91-1913382': 565, '91-2006136': 566, '915878': 567, '9289': 568, '93-1231049': 569, '93-1248952': 570, '93-1296762': 571, '93-1301081': 572, '934329': 573, '9366': 574, '936855': 575, '9373': 576, '94-3195577': 577, '9483': 578, '9486': 579, '95-3821253': 580, '95-3990375': 581, '95-4196389': 582, '95-4234730': 583, '95-4267987': 584, '95-4462959': 585, '95-4482547': 586, '95-4523866': 587, '95-4623407': 588, '95-4762204': 589, '95-4769926': 590, '95-4866828': 591, '972590': 592}

Mapping for purchaser_type_name:
{'Affiliate institution': 0, 'Commercial bank, savings bank or savings association': 1, 'Fannie Mae (FNMA)': 2, 'Farmer Mac (FAMC)': 3, 'Freddie Mac (FHLMC)': 4, 'Ginnie Mae (GNMA)': 5, 'Life insurance company, credit union, mortgage bank, or finance company': 6, 'Loan was not originated or was not sold in calendar year covered by register': 7, 'Other type of purchaser': 8, 'Private securitization': 9}

Mapping for property_type_name:
{'Manufactured housing': 0, 'Multifamily dwelling': 1, 'One-to-four family dwelling (other than manufactured housing)': 2}

Mapping for preapproval_name:
{'Not applicable': 0, 'Preapproval was not requested': 1, 'Preapproval was requested': 2}

Mapping for owner_occupancy_name:
{'Not applicable': 0, 'Not owner-occupied as a principal dwelling': 1, 'Owner-occupied as a principal dwelling': 2}

Mapping for msamd_name:
{'Bellingham - WA': 0, 'Bremerton, Silverdale - WA': 1, 'Kennewick, Richland - WA': 2, 'Lewiston - ID, WA': 3, 'Longview - WA': 4, 'Mount Vernon, Anacortes - WA': 5, 'Olympia, Tumwater - WA': 6, 'Portland, Vancouver, Hillsboro - OR, WA': 7, 'Seattle, Bellevue, Everett - WA': 8, 'Spokane, Spokane Valley - WA': 9, 'Tacoma, Lakewood - WA': 10, 'Walla Walla - WA': 11, 'Wenatchee - WA': 12, 'Yakima - WA': 13}

Mapping for loan_type_name:
{'Conventional': 0, 'FHA-insured': 1, 'FSA/RHS-guaranteed': 2, 'VA-guaranteed': 3}

Mapping for loan_purpose_name:

{'Home improvement': 0, 'Home purchase': 1, 'Refinancing': 2}

Mapping for lien_status_name:
{'Not applicable': 0, 'Not secured by a lien': 1, 'Secured by a first lien': 2, 'Secured by a subordinate lien': 3}

Mapping for hoepa_status_name:
{'HOEPA loan': 0, 'Not a HOEPA loan': 1}

Mapping for county_name:
{'Adams County': 0, 'Asotin County': 1, 'Benton County': 2, 'Chelan County': 3, 'Clallam County': 4, 'Clark County': 5, 'Columbia County': 6, 'Cowlitz County': 7, 'Douglas County': 8, 'Ferry County': 9, 'Franklin County': 10, 'Garfield County': 11, 'Grant County': 12, 'Grays Harbor County': 13, 'Island County': 14, 'Jefferson County': 15, 'King County': 16, 'Kitsap County': 17, 'Kittitas County': 18, 'Klickitat County': 19, 'Lewis County': 20, 'Lincoln County': 21, 'Mason County': 22, 'Okanogan County': 23, 'Pacific County': 24, 'Pend Oreille County': 25, 'Pierce County': 26, 'San Juan County': 27, 'Skagit County': 28, 'Skamania County': 29, 'Snohomish County': 30, 'Spokane County': 31, 'Stevens County': 32, 'Thurston County': 33, 'Wahkiakum County': 34, 'Walla Walla County': 35, 'Whatcom County': 36, 'Whitman County': 37, 'Yakima County': 38}

Mapping for co_applicant_sex_name:
{'Female': 0, 'Information not provided by applicant in mail, Internet, or telephone application': 1, 'Male': 2, 'No co-applicant': 3, 'Not applicable': 4}

Mapping for co_applicant_ethnicity_name:
{'Hispanic or Latino': 0, 'Information not provided by applicant in mail, Internet, or telephone application': 1, 'No co-applicant': 2, 'Not Hispanic or Latino': 3, 'Not applicable': 4}

Mapping for applicant_sex_name:
{'Female': 0, 'Information not provided by applicant in mail, Internet, or telephone application': 1, 'Male': 2, 'Not applicable': 3}

Mapping for applicant_ethnicity_name:
{'Hispanic or Latino': 0, 'Information not provided by applicant in mail, Internet, or telephone application': 1, 'Not Hispanic or Latino': 2, 'Not applicable': 3}

Mapping for agency_name:
{'Consumer Financial Protection Bureau': 0, 'Department of Housing and Urban Development': 1, 'Federal Deposit Insurance Corporation': 2, 'Federal Reserve System': 3, 'National Credit Union Administration': 4, 'Office of the Comptroller of the Currency': 5}

Mapping for agency_abbr:
{'CFPB': 0, 'FDIC': 1, 'FRS': 2, 'HUD': 3, 'NCUA': 4, 'OCC': 5}

```
Mapping for action_taken_name:
{'Application approved but not accepted': 0, 'Application denied by financial
institution': 1, 'Application withdrawn by applicant': 2, 'File closed for
incompleteness': 3, 'Loan originated': 4, 'Loan purchased by the institution':
5, 'Preapproval request approved but not accepted': 6, 'Preapproval request
denied by financial institution': 7}
        S.no  respondent_id  purchaser_type_name  property_type_name  \
0          1            317                    4                   2
1          2            490                    6                   2
2          3            489                    7                   2
3          4            318                    7                   2
4          5            234                    4                   2
…        …             …                   …                  …
59995  59996            472                    8                   2
59996  59997            488                    4                   2
59997  59998             55                    5                   2
59998  59999            116                    5                   2
59999  60000             47                    2                   2

       preapproval_name  owner_occupancy_name  msamd_name  loan_type_name  \
0                     0                     2           7               0
1                     0                     2          11               1
2                     0                     2           7               0
3                     0                     2           7               0
4                     0                     2           1               0
…                   …                   …         …             …
59995                 0                     2           7               0
59996                 0                     1           5               0
59997                 0                     2           0               1
59998                 0                     2           1               3
59999                 0                     2           7               0

       loan_purpose_name  lien_status_name  hoepa_status_name  county_name  \
0                      2                 2                  1            5
1                      1                 2                  1           35
2                      2                 2                  1            5
3                      2                 2                  1            5
4                      0                 2                  1           17
…                    …                …                 …           …
59995                  2                 2                  1            5
59996                  2                 2                  1           28
59997                  0                 2                  1           36
59998                  2                 2                  1           17
59999                  2                 2                  1            5

       co_applicant_sex_name  co_applicant_ethnicity_name  applicant_sex_name  \
0                          2                            3                   0
```

```
1                          3                            2                   2
2                          0                            3                   2
3                          0                            1                   2
4                          2                            3                   0
…                        …                          …                 …
59995                      0                            0                   2
59996                      3                            2                   0
59997                      0                            3                   2
59998                      0                            3                   2
59999                      0                            3                   2

       applicant_ethnicity_name  agency_name  agency_abbr  action_taken_name
0                             2            0            0                  4
1                             0            1            3                  4
2                             2            1            3                  4
3                             1            4            4                  4
4                             2            2            1                  4
…                           …          …          …                …
59995                         2            1            3                  4
59996                         1            1            3                  4
59997                         2            1            3                  4
59998                         2            0            0                  4
59999                         2            0            0                  4

[60000 rows x 19 columns]
time: 505 ms (started: 2024-04-13 14:05:57 +00:00)
```

[19]:

```python
print(imputed_data_non_categorical.columns)
```

```
Index(['S.no', 'tract_to_msamd_income', 'population', 'minority_population',
       'number_of_owner_occupied_units', 'number_of_1_to_4_family_units',
       'loan_amount_000s', 'hud_median_family_income', 'applicant_income_000s',
       'sequence_number', 'census_tract_number', 'application_date_indicator',
       'Cluster_Label'],
      dtype='object')
time: 5.03 ms (started: 2024-04-13 14:05:58 +00:00)
```

[20]:

```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 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
tract_to_msamd_income           1309
population                       553
minority_population             2641
number_of_owner_occupied_units   478
number_of_1_to_4_family_units   2150
loan_amount_000s                2467
hud_median_family_income           0
applicant_income_000s           3765
sequence_number                 7898
census_tract_number             9391
application_date_indicator       776
Cluster_Label                      0
dtype: int64
time: 90.2 ms (started: 2024-04-13 14:05:58 +00:00)
```

[21]:
```
# 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: 60000
Count of unique values in tract_to_msamd_income column: 1328
Count of unique values in population column: 1284
Count of unique values in minority_population column: 1217
Count of unique values in number_of_owner_occupied_units column: 997
Count of unique values in number_of_1_to_4_family_units column: 1052
Count of unique values in loan_amount_000s column: 1344
Count of unique values in hud_median_family_income column: 16
Count of unique values in applicant_income_000s column: 808
Count of unique values in sequence_number column: 39340
Count of unique values in census_tract_number column: 1109
Count of unique values in application_date_indicator column: 2
Count of unique values in Cluster_Label column: 5
time: 47.2 ms (started: 2024-04-13 14:05:58 +00:00)
```

[22]:
```
# 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
tract_to_msamd_income           1309
population                       553
minority_population             2641
number_of_owner_occupied_units   478
number_of_1_to_4_family_units   2150
loan_amount_000s                2467
hud_median_family_income           0
applicant_income_000s           3765
sequence_number                 7898
census_tract_number             9391
application_date_indicator       776
Cluster_Label                      0
dtype: int64
time: 71 ms (started: 2024-04-13 14:05:58 +00:00)
```

[23]:
```
# 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
tract_to_msamd_income          1309
population                      553
minority_population            2641
number_of_owner_occupied_units   478
number_of_1_to_4_family_units  2150
loan_amount_000s               2467
hud_median_family_income          0
applicant_income_000s          3765
sequence_number                7898
census_tract_number            9391
application_date_indicator      776
Cluster_Label                     0
dtype: int64
time: 81.8 ms (started: 2024-04-13 14:05:58 +00:00)
```

[24]:
```python
# Define columns to exclude from normalization
columns_to_exclude = ['S.no','hud_median_family_income',␣
 ↪'application_date_indicator', 'Cluster_Label']

# 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)
```

```
       tract_to_msamd_income  population  minority_population  \
0                   0.442799    0.640752             0.234501
1                   0.285162    0.372631             0.236658
2                   0.317084    0.385008             0.105445
3                   0.543502    0.381682             0.070620
4                   0.610556    0.393363             0.091213
…                        …          …                   …
59995               0.394915    0.299373             0.149434
59996               0.445679    0.369923             0.060162
59997               0.418734    0.609964             0.105553
```

```
59998             0.336419    0.179392            0.322803
59999             0.442799    0.640752            0.234501

       number_of_owner_occupied_units  number_of_1_to_4_family_units  \
0                            0.724346                       0.448858
1                            0.420188                       0.298329
2                            0.375922                       0.308728
3                            0.506372                       0.305660
4                            0.566734                       0.354074
…                                 …                             …
59995                        0.370557                       0.240709
59996                        0.556673                       0.383396
59997                        0.783032                       0.558814
59998                        0.195171                       0.116263
59999                        0.724346                       0.448858

       loan_amount_000s  applicant_income_000s  sequence_number  \
0              0.004109               0.018669         0.096625
1              0.004346               0.006656         0.042368
2              0.004364               0.018831         0.005001
3              0.006364               0.050974         0.000158
4              0.007564               0.018344         0.026241
…                   …                     …               …
59995          0.002927               0.012013         0.024452
59996          0.002564               0.007955         0.268165
59997          0.004546               0.014123         0.020504
59998          0.004655               0.018153         0.289037
59999          0.005309               0.013149         0.032247

       census_tract_number     S.no  hud_median_family_income  \
0                 0.042258      1.0                   73300.0
1                 0.943728      2.0                   57900.0
2                 0.042333      3.0                   73300.0
3                 0.041421      4.0                   73300.0
4                 0.092866      5.0                   78100.0
…                      …        …                        …
59995             0.041926  59996.0                   73300.0
59996             0.963715  59997.0                   61400.0
59997             0.000724  59998.0                   69900.0
59998             0.082002  59999.0                   78100.0
59999             0.042258  60000.0                   73300.0

       application_date_indicator  Cluster_Label
0                             0.0            4.0
1                             0.0            3.0
2                             0.0            4.0
3                             0.0            4.0
4                             0.0            4.0
```

```
…                                    …             …
59995                                  0.0           1.0
59996                                  0.0           1.0
59997                                  0.0           1.0
59998                                  0.0           1.0
59999                                  0.0           1.0

[60000 rows x 13 columns]
time: 35.3 ms (started: 2024-04-13 14:05:58 +00:00)
```

[25]:
```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)
```

```
tract_to_msamd_income             1309
population                         553
minority_population               2641
number_of_owner_occupied_units     478
number_of_1_to_4_family_units     2150
loan_amount_000s                  2467
applicant_income_000s             3765
sequence_number                   7898
census_tract_number               9391
S.no                                 0
hud_median_family_income             0
application_date_indicator         776
Cluster_Label                        0
dtype: int64
time: 51.9 ms (started: 2024-04-13 14:05:58 +00:00)
```

[26]:
```python
# 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            tract_to_msamd_income            0.116026
1                       population            0.132618
2              minority_population            0.155273
3    number_of_owner_occupied_units            0.173640
4     number_of_1_to_4_family_units            0.125768
5                 loan_amount_000s            0.010999
6             applicant_income_000s            0.018857
7                  sequence_number            0.121228
8              census_tract_number            0.344020
9                             S.no        17320.652413
10           hud_median_family_income     12798.211781
11         application_date_indicator            0.225976
12                    Cluster_Label            1.395815
time: 20.3 ms (started: 2024-04-13 14:05:58 +00:00)
```

[27]:
```
# Pre-Processed Dataset
combined_data = pd.merge(encoded_data_categorical, scaled_df, on='S.no')

# Display the Pre-Processed Dataset
%memit
combined_data
```

peak memory: 400.98 MiB, increment: 0.27 MiB

[27]:

	S.no	respondent_id	purchaser_type_name	property_type_name	\
0	1	317	4	2	
1	2	490	6	2	
2	3	489	7	2	
3	4	318	7	2	
4	5	234	4	2	
…	…	…	…	…	
59995	59996	472	8	2	
59996	59997	488	4	2	
59997	59998	55	5	2	
59998	59999	116	5	2	
59999	60000	47	2	2	

	preapproval_name	owner_occupancy_name	msamd_name	loan_type_name	\
0	0	2	7	0	
1	0	2	11	1	

```
2                    0                    2          7                0
3                    0                    2          7                0
4                    0                    2          1                0
…                  …                  …         …              …
59995                0                    2          7                0
59996                0                    1          5                0
59997                0                    2          0                1
59998                0                    2          1                3
59999                0                    2          7                0

       loan_purpose_name  lien_status_name  …  minority_population  \
0                      2                 2  …             0.234501
1                      1                 2  …             0.236658
2                      2                 2  …             0.105445
3                      2                 2  …             0.070620
4                      0                 2  …             0.091213
…                    …               …  …                …
59995                  2                 2  …             0.149434
59996                  2                 2  …             0.060162
59997                  0                 2  …             0.105553
59998                  2                 2  …             0.322803
59999                  2                 2  …             0.234501

       number_of_owner_occupied_units  number_of_1_to_4_family_units  \
0                            0.724346                       0.448858
1                            0.420188                       0.298329
2                            0.375922                       0.308728
3                            0.506372                       0.305660
4                            0.566734                       0.354074
…                                 …                            …
59995                        0.370557                       0.240709
59996                        0.556673                       0.383396
59997                        0.783032                       0.558814
59998                        0.195171                       0.116263
59999                        0.724346                       0.448858

       loan_amount_000s  applicant_income_000s  sequence_number  \
0              0.004109               0.018669         0.096625
1              0.004346               0.006656         0.042368
2              0.004364               0.018831         0.005001
3              0.006364               0.050974         0.000158
4              0.007564               0.018344         0.026241
…                   …                    …              …
59995          0.002927               0.012013         0.024452
59996          0.002564               0.007955         0.268165
59997          0.004546               0.014123         0.020504
59998          0.004655               0.018153         0.289037
```

```
59999          0.005309              0.013149        0.032247

       census_tract_number  hud_median_family_income  \
0                 0.042258                   73300.0
1                 0.943728                   57900.0
2                 0.042333                   73300.0
3                 0.041421                   73300.0
4                 0.092866                   78100.0
…                      …                        …
59995             0.041926                   73300.0
59996             0.963715                   61400.0
59997             0.000724                   69900.0
59998             0.082002                   78100.0
59999             0.042258                   73300.0

       application_date_indicator  Cluster_Label
0                             0.0            4.0
1                             0.0            3.0
2                             0.0            4.0
3                             0.0            4.0
4                             0.0            4.0
…                          …             …
59995                         0.0            1.0
59996                         0.0            1.0
59997                         0.0            1.0
59998                         0.0            1.0
59999                         0.0            1.0

[60000 rows x 31 columns]

time: 411 ms (started: 2024-04-13 14:05:58 +00:00)
```

[28]:
```python
# Get the index of the 'Cluster_Label' column
cluster_label_index = combined_data.columns.get_loc('Cluster_Label')

# Reorder the columns to move 'Cluster_Label' to the extreme right
combined_data = combined_data[[col for col in combined_data if col !=
 ↪'Cluster_Label'] + ['Cluster_Label']]

# Display the updated dataset
print(combined_data.head())
```

```
  S.no  respondent_id  purchaser_type_name  property_type_name  \
0    1            317                    4                   2
1    2            490                    6                   2
2    3            489                    7                   2
3    4            318                    7                   2
```

```
4    5           234                     4                   2

   preapproval_name  owner_occupancy_name  msamd_name  loan_type_name  \
0                 0                     2           7               0
1                 0                     2          11               1
2                 0                     2           7               0
3                 0                     2           7               0
4                 0                     2           1               0

   loan_purpose_name  lien_status_name  …  minority_population  \
0                  2                 2  …             0.234501
1                  1                 2  …             0.236658
2                  2                 2  …             0.105445
3                  2                 2  …             0.070620
4                  0                 2  …             0.091213

   number_of_owner_occupied_units  number_of_1_to_4_family_units  \
0                        0.724346                       0.448858
1                        0.420188                       0.298329
2                        0.375922                       0.308728
3                        0.506372                       0.305660
4                        0.566734                       0.354074

   loan_amount_000s  applicant_income_000s  sequence_number  \
0          0.004109               0.018669         0.096625
1          0.004346               0.006656         0.042368
2          0.004364               0.018831         0.005001
3          0.006364               0.050974         0.000158
4          0.007564               0.018344         0.026241

   census_tract_number  hud_median_family_income  application_date_indicator  \
0             0.042258                   73300.0                         0.0
1             0.943728                   57900.0                         0.0
2             0.042333                   73300.0                         0.0
3             0.041421                   73300.0                         0.0
4             0.092866                   78100.0                         0.0

   Cluster_Label
0            4.0
1            3.0
2            4.0
3            4.0
4            4.0

[5 rows x 31 columns]
time: 22.5 ms (started: 2024-04-13 14:05:59 +00:00)
```

[29]:
```
df_ppd_subset = combined_data.copy()
```

time: 17.7 ms (started: 2024-04-13 14:05:59 +00:00)

[30]:
```
list(combined_data.columns)
```

[30]:
```
['S.no',
 'respondent_id',
 'purchaser_type_name',
 'property_type_name',
 'preapproval_name',
 'owner_occupancy_name',
 'msamd_name',
 'loan_type_name',
 'loan_purpose_name',
 'lien_status_name',
 'hoepa_status_name',
 'county_name',
 'co_applicant_sex_name',
 'co_applicant_ethnicity_name',
 'applicant_sex_name',
 'applicant_ethnicity_name',
 'agency_name',
 'agency_abbr',
 'action_taken_name',
 'tract_to_msamd_income',
 'population',
 'minority_population',
 'number_of_owner_occupied_units',
 'number_of_1_to_4_family_units',
 'loan_amount_000s',
 'applicant_income_000s',
 'sequence_number',
 'census_tract_number',
 'hud_median_family_income',
 'application_date_indicator',
 'Cluster_Label']
```

time: 4.05 ms (started: 2024-04-13 14:05:59 +00:00)

6 Decision Tree

[31]:
```
#####  DT
```

time: 407 µs (started: 2024-04-13 14:05:59 +00:00)

[32]:
```
df1 = df_ppd_subset.copy()
```

```
time: 9.46 ms (started: 2024-04-13 14:05:59 +00:00)
```

[33]:
```
df1.columns
```

[33]:
```
Index(['S.no', 'respondent_id', 'purchaser_type_name', 'property_type_name',
       'preapproval_name', 'owner_occupancy_name', 'msamd_name',
       'loan_type_name', 'loan_purpose_name', 'lien_status_name',
       'hoepa_status_name', 'county_name', 'co_applicant_sex_name',
       'co_applicant_ethnicity_name', 'applicant_sex_name',
       'applicant_ethnicity_name', 'agency_name', 'agency_abbr',
       'action_taken_name', 'tract_to_msamd_income', 'population',
       'minority_population', 'number_of_owner_occupied_units',
       'number_of_1_to_4_family_units', 'loan_amount_000s',
       'applicant_income_000s', 'sequence_number', 'census_tract_number',
       'hud_median_family_income', 'application_date_indicator',
       'Cluster_Label'],
      dtype='object')

time: 4.27 ms (started: 2024-04-13 14:05:59 +00:00)
```

[34]:
```
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 60000 entries, 0 to 59999
Data columns (total 31 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   S.no                            60000 non-null  object
 1   respondent_id                   60000 non-null  int64
 2   purchaser_type_name             60000 non-null  int64
 3   property_type_name              60000 non-null  int64
 4   preapproval_name                60000 non-null  int64
 5   owner_occupancy_name            60000 non-null  int64
 6   msamd_name                      60000 non-null  int64
 7   loan_type_name                  60000 non-null  int64
 8   loan_purpose_name               60000 non-null  int64
 9   lien_status_name                60000 non-null  int64
 10  hoepa_status_name               60000 non-null  int64
 11  county_name                     60000 non-null  int64
 12  co_applicant_sex_name           60000 non-null  int64
 13  co_applicant_ethnicity_name     60000 non-null  int64
 14  applicant_sex_name              60000 non-null  int64
 15  applicant_ethnicity_name        60000 non-null  int64
 16  agency_name                     60000 non-null  int64
 17  agency_abbr                     60000 non-null  int64
 18  action_taken_name               60000 non-null  int64
 19  tract_to_msamd_income           60000 non-null  float64
 20  population                      60000 non-null  float64
```

```
 21  minority_population             60000 non-null  float64
 22  number_of_owner_occupied_units  60000 non-null  float64
 23  number_of_1_to_4_family_units   60000 non-null  float64
 24  loan_amount_000s                60000 non-null  float64
 25  applicant_income_000s           60000 non-null  float64
 26  sequence_number                 60000 non-null  float64
 27  census_tract_number             60000 non-null  float64
 28  hud_median_family_income        60000 non-null  float64
 29  application_date_indicator      60000 non-null  float64
 30  Cluster_Label                   60000 non-null  float64
dtypes: float64(12), int64(18), object(1)
memory usage: 14.2+ MB
time: 26.7 ms (started: 2024-04-13 14:05:59 +00:00)
```

[35]:
```
# Subset df1 based on Inputs as {mpg, hp, cyl, vs} & Output as {am}
df1_inputs = df1[['msamd_name', 'loan_type_name', 'loan_purpose_name',
        'hud_median_family_income', 'loan_amount_000s']];
df1_inputs
df1_output = df1[['action_taken_name']]; df1_output

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

[35]:
```
array(['4', '0', '1', '2', '3', '5', '6', '7'], dtype='<U21')

time: 9.61 ms (started: 2024-04-13 14:05:59 +00:00)
```

[36]:
```
# Initialize StratifiedShuffleSplit with desired test size and random state
stratified_split = StratifiedShuffleSplit(n_splits=1, test_size=0.25,
 ↪random_state=45005)

# 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.iloc[train_index]
    df1_inputs_test = df1_inputs.iloc[test_index]
    df1_output_train = df1_output.iloc[train_index]
    df1_output_test = df1_output.iloc[test_index]
```

```
time: 321 ms (started: 2024-04-13 14:05:59 +00:00)
```

[37]:
```
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=45011)
```

```
# 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(['msamd_name', 'loan_type_name', 'loan_purpose_name',
       'loan_amount_000s'],
      dtype='object')
Threshold: 0.24582593448183424
time: 3.66 s (started: 2024-04-13 14:05:59 +00:00)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/svm/_base.py:1244:
ConvergenceWarning: Liblinear failed to converge, increase the number of
iterations.
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:432: UserWarning: X has
feature names, but SelectFromModel was fitted without feature names
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:432: UserWarning: X has
feature names, but SelectFromModel was fitted without feature names
  warnings.warn(
```

[38]:
```
# Decision Tree : Model (Training Subset)
dtc = DecisionTreeClassifier(criterion='gini', random_state=45005) # Other␣
↪Criteria : Entropy,  Log Loss
dtc_model = dtc.fit(df1_inputs_train, df1_output_train); dtc_model
```

[38]:
```
DecisionTreeClassifier(random_state=45005)
```

```
time: 94.3 ms (started: 2024-04-13 14:06:03 +00:00)
```

[39]:
```python
# Decision Tree : Model Rules
dtc_model_rules = export_text(dtc_model, feature_names =␣
↪list(df1_inputs_names)); print(dtc_model_rules)
```

```
|--- hud_median_family_income <= 58650.00
|   |--- loan_type_name <= 0.50
|   |   |--- loan_purpose_name <= 1.50
|   |   |   |--- msamd_name <= 9.50
|   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 12
```

```
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 12
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
```

```
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 19
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
```

```
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 10
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 11
|   |   |   |--- msamd_name >  9.50
|   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income <=
53300.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income >
53300.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- msamd_name <= 12.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- msamd_name >  12.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
```

```
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- hud_median_family_income <= 53300.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 11
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |   |--- hud_median_family_income >  53300.00
|   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- hud_median_family_income <= 53300.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- hud_median_family_income >  53300.00
|   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- hud_median_family_income <= 53300.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- hud_median_family_income >  53300.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
```

```
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- msamd_name <= 12.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- msamd_name >  12.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 13
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |--- loan_purpose_name >  1.50
|   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |--- class: 5
|   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |--- msamd_name <= 9.50
```

```
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 16
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 20
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 11
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
```

```
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |--- msamd_name >  9.50
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 13
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 9
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
```

```
|   |   |   |   |   |   |   |   |--- hud_median_family_income <= 53300.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- hud_median_family_income >  53300.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income <=
53300.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income >
53300.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income <=
53300.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income >
53300.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |--- loan_amount_000s <= 0.50
|   |   |   |   |   |   |--- hud_median_family_income <= 56750.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- hud_median_family_income <= 52150.00
```

```
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 9
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- hud_median_family_income >  52150.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 14
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 13
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- msamd_name <= 10.50
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 12
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 15
|   |   |   |   |   |   |   |   |   |--- msamd_name >  10.50
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |--- hud_median_family_income >  56750.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 4
```

```
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |--- loan_amount_000s >  0.50
|   |   |   |   |   |   |--- class: 5
|   |--- loan_type_name >  0.50
|   |   |--- loan_purpose_name <= 1.50
|   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |--- msamd_name <= 9.50
|   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_type_name <= 1.50
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_type_name >  1.50
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 13
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name <= 1.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name >  1.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
```

```
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_type_name <= 1.50
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_type_name >  1.50
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |--- msamd_name >  9.50
|   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- hud_median_family_income <= 53300.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |--- hud_median_family_income >  53300.00
|   |   |   |   |   |   |   |   |--- loan_type_name <= 1.50
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_type_name >  1.50
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_type_name <= 1.50
```

```
|   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_type_name >  1.50
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income <=
53300.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income >
53300.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |--- loan_type_name <= 1.50
|   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 19
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- msamd_name <= 12.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- msamd_name >  12.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
```

```
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income <=
56750.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 11
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income >
56750.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |--- loan_type_name >  1.50
|   |   |   |   |   |--- msamd_name <= 9.50
|   |   |   |   |   |   |--- loan_type_name <= 2.50
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 22
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |--- loan_type_name >  2.50
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 13
```

```
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 11
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 23
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |--- msamd_name >  9.50
|   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |--- hud_median_family_income <= 53300.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
```

```
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |--- hud_median_family_income >  53300.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |--- loan_purpose_name >  1.50
|   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |--- loan_type_name <= 2.50
|   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |--- msamd_name <= 9.50
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 9
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
```

```
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |--- msamd_name >  9.50
|   |   |   |   |   |   |   |--- msamd_name <= 12.00
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- msamd_name >  12.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 11
|   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |--- msamd_name <= 12.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_type_name <= 1.50
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
```

```
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 10
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |--- loan_type_name >  1.50
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- msamd_name >  12.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |--- loan_type_name >  2.50
|   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |--- msamd_name <= 9.50
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
```

```
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 11
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 10
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 13
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |--- msamd_name >  9.50
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 9
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- hud_median_family_income <= 56750.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
```

```
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- hud_median_family_income <= 52150.00
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- hud_median_family_income >  52150.00
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 10
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |--- hud_median_family_income >  56750.00
|   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |--- loan_amount_000s >  0.00
```

```
|   |   |   |   |   |   |--- msamd_name <= 12.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- msamd_name >  12.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_amount_000s >  0.01
```

```
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.02
|   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.02
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 13
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
```

```
|   |   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- class: 4
|--- hud_median_family_income >  58650.00
|   |--- msamd_name <= 9.50
|   |   |--- msamd_name <= 7.50
|   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |--- class: 4
|   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |--- class: 2
|   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- msamd_name <= 4.50
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- msamd_name >  4.50
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- hud_median_family_income <= 73450.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- hud_median_family_income >  73450.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- msamd_name <= 5.50
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- msamd_name >  5.50
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
```

```
|   |   |   |   |   |   |   |   |--- hud_median_family_income <= 73450.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- hud_median_family_income >  73450.00
|   |   |   |   |   |   |   |   |   |--- hud_median_family_income <= 75850.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- hud_median_family_income >  75850.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- hud_median_family_income <= 67850.00
|   |   |   |   |   |   |   |   |   |   |--- msamd_name <= 2.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- msamd_name >  2.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- hud_median_family_income >  67850.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |--- msamd_name <= 1.50
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 1.50
|   |   |   |   |   |   |   |   |   |   |--- msamd_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- msamd_name >  0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  1.50
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 11
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name >  0.50
```

```
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- msamd_name >  1.50
|   |   |   |   |   |   |   |--- loan_purpose_name <= 1.50
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- hud_median_family_income <= 73450.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- hud_median_family_income >  73450.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |--- loan_purpose_name >  1.50
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_type_name <= 2.50
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name >  0.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |--- loan_type_name >  2.50
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income <=
73450.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income >
73450.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income <=
61350.00
```

```
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income >
61350.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 10
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |--- loan_type_name <= 1.50
|   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_type_name >  1.50
|   |   |   |   |   |   |   |--- hud_median_family_income <= 75850.00
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- hud_median_family_income >  75850.00
|   |   |   |   |   |   |   |   |--- loan_purpose_name <= 1.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_purpose_name >  1.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |--- loan_type_name <= 2.50
|   |   |   |   |   |   |   |--- msamd_name <= 0.50
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- msamd_name >  0.50
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 1.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  1.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income <=
67350.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income >
67350.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_type_name >  2.50
```

```
|   |   |   |   |   |   |   |--- msamd_name <= 6.50
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- msamd_name >  6.50
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 1.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  1.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |--- msamd_name >  7.50
|   |   |   |--- hud_median_family_income <= 82084.71
|   |   |   |   |--- hud_median_family_income <= 67834.71
|   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 1.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  1.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_type_name <= 0.50
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_type_name >  0.50
|   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 1.50
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  1.50
|   |   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
```

```
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 1.50
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  1.50
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_type_name <= 2.50
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_type_name >  2.50
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |--- hud_median_family_income >  67834.71
|   |   |   |   |   |--- loan_type_name <= 0.50
|   |   |   |   |   |   |--- loan_amount_000s <= 0.02
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_purpose_name <= 1.50
|   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |--- class: 4
```

```
|   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |--- loan_purpose_name >  1.50
|   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_purpose_name <= 1.50
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 6
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |--- loan_purpose_name >  1.50
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |--- loan_amount_000s >  0.02
|   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |--- loan_type_name >  0.50
|   |   |   |   |   |   |--- loan_purpose_name <= 1.50
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_type_name <= 2.50
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |--- loan_type_name >  2.50
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
```

```
|   |   |   |   |   |   |   |   |   |   |--- class: 7
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |   |   |   |   |--- class: 7
|   |   |   |   |   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |   |   |   |   |--- class: 5
|   |   |   |   |   |   |--- loan_purpose_name >  1.50
|   |   |   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |--- hud_median_family_income >  82084.71
|   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |--- loan_purpose_name <= 1.00
|   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |--- loan_purpose_name >  1.00
|   |   |   |   |   |   |--- class: 5
|   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_type_name <= 0.50
|   |   |   |   |   |   |   |   |   |--- class: 4
```

```
|   |   |   |   |   |   |   |   |--- loan_type_name >  0.50
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_purpose_name <= 1.50
|   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |--- loan_purpose_name >  1.50
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |--- loan_purpose_name <= 1.50
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_type_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_type_name >  0.50
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 11
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 14
|   |   |   |   |   |   |--- loan_purpose_name >  1.50
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
```

```
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 21
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- loan_type_name <= 1.50
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_type_name >  1.50
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.02
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 12
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.02
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.02
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.02
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.02
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.02
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |--- msamd_name >  9.50
|   |   |--- loan_purpose_name <= 1.50
|   |   |   |--- loan_type_name <= 0.50
|   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
```

```
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- msamd_name <= 11.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- msamd_name >  11.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income <=
67700.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- hud_median_family_income >
67700.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
```

```
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 13
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 14
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- msamd_name <= 11.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- msamd_name >  11.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- hud_median_family_income <= 67700.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |--- hud_median_family_income >  67700.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
```

```
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 18
|   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- msamd_name <= 11.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- msamd_name >  11.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |--- class: 4
|   |   |   |--- loan_type_name >  0.50
|   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |--- msamd_name <= 11.00
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_type_name <= 1.50
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 12
```

```
|   |   |   |   |   |   |   |--- loan_type_name >  1.50
|   |   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |--- msamd_name >  11.00
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_type_name <= 2.50
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_type_name >  2.50
|   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |--- loan_purpose_name <= 0.50
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
```

```
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |--- loan_purpose_name >  0.50
|   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 15
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
```

```
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 19
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |--- loan_purpose_name >  1.50
|   |   |   |--- msamd_name <= 11.00
|   |   |   |   |--- loan_type_name <= 2.50
|   |   |   |   |   |--- loan_type_name <= 0.50
|   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
```

```
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 18
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 12
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 6
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 10
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
```

```
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |--- loan_type_name >  0.50
|   |   |   |   |   |   |--- loan_type_name <= 1.50
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 16
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 10
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_type_name >  1.50
|   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |--- loan_type_name >  2.50
|   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
```

```
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 13
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 8
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
```

```
|   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 12
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 5
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 7
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 12
|   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |--- class: 4
|   |   |   |--- msamd_name >  11.00
|   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |--- loan_type_name <= 0.50
|   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_type_name >  0.50
|   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- class: 2
```

```
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 10
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name <= 0.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 9
|   |   |   |   |   |   |   |   |   |   |--- loan_type_name >  0.50
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |--- loan_type_name <= 0.50
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
```

```
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_type_name >  0.50
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |--- loan_type_name <= 2.00
|   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_type_name >  2.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.00
|   |   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |--- loan_amount_000s >  0.00
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 14
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |   |   |--- truncated branch of depth 3
|   |   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |   |--- class: 2
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |--- loan_amount_000s <= 0.01
|   |   |   |   |   |   |   |--- class: 3
|   |   |   |   |   |   |--- loan_amount_000s >  0.01
|   |   |   |   |   |   |   |--- loan_amount_000s <= 0.25
|   |   |   |   |   |   |   |   |--- class: 4
|   |   |   |   |   |   |   |--- loan_amount_000s >  0.25
|   |   |   |   |   |   |   |   |--- class: 3

time: 33.9 ms (started: 2024-04-13 14:06:03 +00:00)
```

[40]:
```python
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import log_loss

# Assuming you have already trained the Decision Tree classifier
# dtc = DecisionTreeClassifier(criterion='gini', random_state=45005)
# dtc_model = dtc.fit(df1_inputs_train, df1_output_train)

# Predict probabilities for each class
y_pred_proba_dtc = dtc_model.predict_proba(df1_inputs_test)

# Calculate entropy
entropy_dtc = log_loss(df1_output_test, y_pred_proba_dtc)

# Calculate Gini impurity
gini_impurity_dtc = 1 - (y_pred_proba_dtc ** 2).sum(axis=1).mean()

print("Entropy for Decision Tree:", entropy_dtc)
print("Gini Impurity for Decision Tree:", gini_impurity_dtc)
```

```
Entropy for Decision Tree: 2.355324846665792
Gini Impurity for Decision Tree: 0.054342145064425695
time: 51.4 ms (started: 2024-04-13 14:06:03 +00:00)
```

[41]:
```python
# 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
```

[41]:

	feature	importance
4	loan_amount_000s	0.604
3	hud_median_family_income	0.162
0	msamd_name	0.148
1	loan_type_name	0.049
2	loan_purpose_name	0.038

```
time: 19.5 ms (started: 2024-04-13 14:06:03 +00:00)
```

[42]:
```python
# Decision Tree : Model Prediction (Training Subset)
dtc_model_predict = dtc_model.predict(df1_inputs_train); dtc_model_predict
```

[42]:
```
array([4, 4, 2, …, 4, 4, 4])
```

```
time: 18.3 ms (started: 2024-04-13 14:06:03 +00:00)
```

[43]:
```python
# Decision Tree : Prediction (Testing Subset)
dtc_predict = dtc_model.predict(df1_inputs_test); dtc_predict
```

[43]: array([4, 4, 4, …, 4, 2, 4])

time: 15.5 ms (started: 2024-04-13 14:06:03 +00:00)

[44]:
```
# 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       0.58      0.44      0.50        63
           1       0.66      0.34      0.45       121
           2       0.68      0.59      0.63      2000
           3       0.72      0.43      0.54       334
           4       0.97      0.99      0.98     41861
           5       0.92      0.26      0.41       582
           6       1.00      0.38      0.56        13
           7       1.00      0.88      0.94        26

    accuracy                           0.95     45000
   macro avg       0.82      0.54      0.63     45000
weighted avg       0.95      0.95      0.95     45000
```

time: 84.5 ms (started: 2024-04-13 14:06:03 +00:00)

[45]:
```
# 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       0.05      0.05      0.05        21
           1       0.00      0.00      0.00        40
           2       0.22      0.20      0.21       666
           3       0.12      0.08      0.10       112
           4       0.94      0.96      0.95     13954
           5       0.11      0.03      0.05       194
           6       0.00      0.00      0.00         4
           7       0.50      0.44      0.47         9

    accuracy                           0.90     15000
   macro avg       0.24      0.22      0.23     15000
weighted avg       0.89      0.90      0.90     15000
```

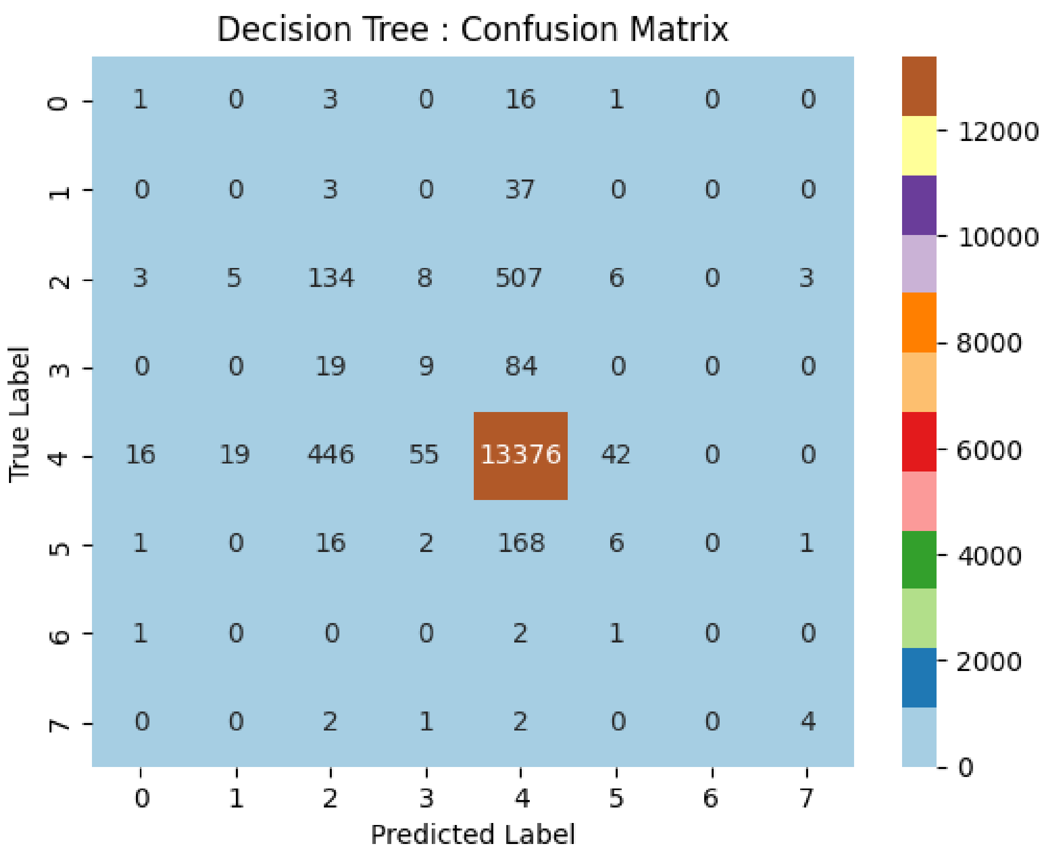
```
time: 39.4 ms (started: 2024-04-13 14:06:03 +00:00)

/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```

[46]:
```
#     SO MUCH TIME TO LOAD :(
'''
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()
'''
```

[46]:
```
'\nimport matplotlib.pyplot as plt\nfrom sklearn.tree import plot_tree\n\n# Set
a larger figure size for better clarity\nplt.figure(figsize=(10, 10))\n\n# Plot
the decision tree\ntrain_subset_dtc_plot = plot_tree(dtc_model,
feature_names=df1_inputs_names, class_names=df1_output_labels, rounded=True,
filled=True, fontsize=20)\n\n# Show the plot\nplt.show()\n'

time: 7.28 ms (started: 2024-04-13 14:06:03 +00:00)
```

[47]:
```
# 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')
```

```python
# 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: 483 ms (started: 2024-04-13 14:06:03 +00:00)

[48]:
```python
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split, StratifiedShuffleSplit,␣
↪cross_val_score
from sklearn.tree import DecisionTreeClassifier, export_text, plot_tree
from sklearn.linear_model import LogisticRegression
from sklearn.feature_selection import SelectFromModel
from sklearn.metrics import accuracy_score, classification_report,␣
↪confusion_matrix, f1_score
```

```python
import numpy as np
import time
import psutil

# Function to measure memory usage
def memory_usage():
    process = psutil.Process()
    return process.memory_info().rss / 1024 ** 2  # Memory usage in MB

# Start time
start_time = time.time()

# Your code for data preprocessing and model training goes here...

# End time
end_time = time.time()

# Time taken
execution_time = end_time - start_time

# Memory usage
memory_used = memory_usage()

# Print memory usage and execution time
print("Memory used (MB):", memory_used)
print("Time taken (seconds):", execution_time)

# Make predictions using the trained model and measure accuracy
# Assuming you have already trained your decision tree model (dtc_model) and␣
 ↪logistic regression model (logreg_l1)
# For example:

# Decision Tree predictions and evaluation
dtc_predict_train = dtc_model.predict(df1_inputs_train)
dtc_predict_test = dtc_model.predict(df1_inputs_test)

# Calculate accuracy for decision tree
accuracy_dtc_train = accuracy_score(df1_output_train, dtc_predict_train)
accuracy_dtc_test = accuracy_score(df1_output_test, dtc_predict_test)


# Print accuracy
print("Decision Tree Training Accuracy:", accuracy_dtc_train)
print("Decision Tree Testing Accuracy:", accuracy_dtc_test)
```

```
Memory used (MB): 453.47265625
Time taken (seconds): 3.647804260253906e-05
```

```
Decision Tree Training Accuracy: 0.9523555555555555
Decision Tree Testing Accuracy: 0.902
time: 35.6 ms (started: 2024-04-13 14:06:04 +00:00)
```

[49]:
```python
# 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=45005)

# 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))
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 17 members, which is less
than n_splits=20.
  warnings.warn(
```

```
Cross-Validation Scores: [0.88366667 0.85566667 0.83433333 0.845      0.892
0.88966667
 0.856      0.79133333 0.877      0.919      0.93133333 0.931
 0.93066667 0.93166667 0.93133333 0.92933333 0.92866667 0.92966667
 0.929      0.92433333]
Average Cross-Validation Score: 0.8970333333333332
time: 4.66 s (started: 2024-04-13 14:06:04 +00:00)
```

[50]:
```python
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: 0.2274677743367185
Weighted F1 Score: 0.8951112458040403
time: 74.2 ms (started: 2024-04-13 14:06:08 +00:00)
```

7 KNN

[51]:
```
#############    KKKKKKKKKK  NNNNNNNNN  NNNNNNNNN
```

time: 372 µs (started: 2024-04-13 14:06:09 +00:00)

[52]:
```
# Specify the number of neighbors (k)
k = 5

# 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)
```

```
/usr/local/lib/python3.10/dist-
packages/sklearn/neighbors/_classification.py:215: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  return self._fit(X, y)
```

[52]:
```
KNeighborsClassifier()
```

time: 92.3 ms (started: 2024-04-13 14:06:09 +00:00)

[53]:
```
# Make predictions using the testing data
y_pred = knn.predict(df1_inputs_test)
```

time: 2.71 s (started: 2024-04-13 14:06:09 +00:00)

[76]:
```
from sklearn.inspection import permutation_importance

# Perform permutation importance
perm_importance = permutation_importance(knn, df1_inputs_test, df1_output_test,␣
↪n_repeats=10, random_state=42)

# Get the feature importances
importances = perm_importance.importances_mean

# Create a DataFrame to store the feature importance
knn_imp_features = pd.DataFrame({'feature': df1_inputs_names, 'importance':␣
↪importances})
knn_imp_features.sort_values('importance', ascending=False, inplace=True)
print(knn_imp_features)
```

```
                   feature  importance
3  hud_median_family_income    0.003083
1            loan_type_name    0.000383
```

```
0                 msamd_name    0.000000
2          loan_purpose_name   -0.000375
4           loan_amount_000s   -0.000642
time: 1min 16s (started: 2024-04-13 14:20:40 +00:00)
```

[54]:

```python
# 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: 0.9174666666666667
Classification Report:
              precision    recall  f1-score   support

           0       0.00      0.00      0.00        21
           1       0.00      0.00      0.00        40
           2       0.23      0.12      0.16       666
           3       0.02      0.01      0.01       112
           4       0.94      0.98      0.96     13954
           5       0.09      0.01      0.02       194
           6       0.00      0.00      0.00         4
           7       0.67      0.67      0.67         9

    accuracy                           0.92     15000
   macro avg       0.24      0.22      0.23     15000
weighted avg       0.88      0.92      0.90     15000

Confusion Matrix:
[[    0     0     2     1    17     1     0     0]
 [    0     0     2     0    38     0     0     0]
 [    0     1    78     9   575     2     0     1]
 [    1     0    13     1    97     0     0     0]
 [    2     0   230    30 13675    17     0     0]
 [    0     0     9     1   181     2     0     1]
 [    0     0     1     0     2     0     0     1]
 [    0     0     2     0     1     0     0     6]]
time: 123 ms (started: 2024-04-13 14:06:11 +00:00)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
```

```
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```

[56]:

```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import log_loss
import numpy as np

# Assuming you have already trained the KNN classifier
# knn = KNeighborsClassifier(n_neighbors=k)
# knn.fit(df1_inputs_train, df1_output_train)

# Predict probabilities for each class
y_pred_proba_knn = np.zeros((len(df1_inputs_test), len(np.
 ↪unique(df1_output_train))))

for i, y in enumerate(knn.classes_):
    y_pred_proba_knn[:, i] = (y_pred == y).sum(axis=0)  # Sum along axis 0

# Normalize probabilities
y_pred_proba_knn /= y_pred_proba_knn.sum(axis=1).reshape(-1, 1)

# Calculate entropy
entropy_knn = log_loss(df1_output_test, y_pred_proba_knn)

# Calculate Gini impurity
gini_impurity_knn = 1 - (y_pred_proba_knn ** 2).sum(axis=1).mean()

print("Entropy for KNN:", entropy_knn)
print("Gini Impurity for KNN:", gini_impurity_knn)
```

```
Entropy for KNN: 0.37447244277425973
Gini Impurity for KNN: 0.05392309333333323
time: 28.4 ms (started: 2024-04-13 14:06:12 +00:00)
```

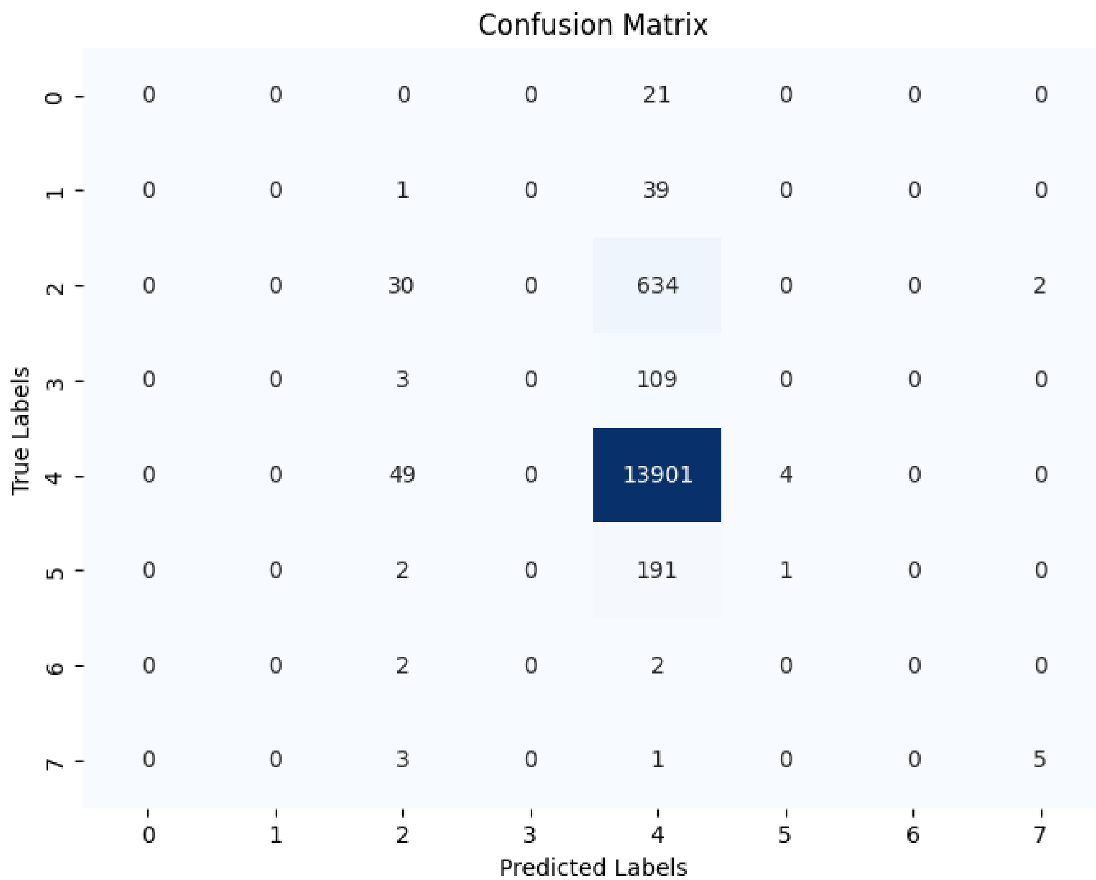
[57]:

```
k_values = [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}')
```

```
/usr/local/lib/python3.10/dist-
packages/sklearn/neighbors/_classification.py:215: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  return self._fit(X, y)

Accuracy for k=11: 0.9276

/usr/local/lib/python3.10/dist-
packages/sklearn/neighbors/_classification.py:215: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  return self._fit(X, y)

Accuracy for k=13: 0.9281333333333334

/usr/local/lib/python3.10/dist-
packages/sklearn/neighbors/_classification.py:215: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  return self._fit(X, y)

Accuracy for k=15: 0.9291333333333334
time: 8.84 s (started: 2024-04-13 14:06:12 +00:00)
```

[58]:
```
# 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: 659 ms (started: 2024-04-13 14:06:21 +00:00)

[59]:
```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score, classification_report,␣
↪confusion_matrix
import time
import psutil

# Function to measure memory usage
def memory_usage():
    process = psutil.Process()
    return process.memory_info().rss / 1024 ** 2  # Memory usage in MB

# Specify the number of neighbors (k)
k_values = [9, 11, 13, 15,17]

for k in k_values:
    # Initialize KNN classifier with k neighbors
    knn = KNeighborsClassifier(n_neighbors=k)
```

```
    # Start time
    start_time = time.time()

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

    # End time
    end_time = time.time()

    # Time taken for training
    training_time = end_time - start_time

    # Make predictions using the testing data
    y_pred = knn.predict(df1_inputs_test)

    # Calculate accuracy
    accuracy = accuracy_score(df1_output_test, y_pred)

    # Generate classification report
    classification_rep = classification_report(df1_output_test, y_pred)

    # Generate confusion matrix
    conf_mat = confusion_matrix(df1_output_test, y_pred)

    # Memory usage
    memory_used = memory_usage()

    print(f'\nResults for k={k}:')
    print(f'Training Time (seconds): {training_time}')
    print(f'Accuracy: {accuracy}')
    print(f'Classification Report:\n{classification_rep}')
    print(f'Confusion Matrix:\n{conf_mat}')
    print(f'Memory Used (MB): {memory_used}')

# 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
```

```python
plot_confusion_matrix(df1_output_test, y_pred)

from sklearn.model_selection import cross_val_score

# 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 20 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)
```

```
/usr/local/lib/python3.10/dist-
packages/sklearn/neighbors/_classification.py:215: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  return self._fit(X, y)
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-
packages/sklearn/neighbors/_classification.py:215: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  return self._fit(X, y)
```

```
Results for k=9:
Training Time (seconds): 0.1091165542602539
Accuracy: 0.9258
Classification Report:
```

```
              precision    recall  f1-score   support

           0       0.00      0.00      0.00        21
           1       0.00      0.00      0.00        40
           2       0.29      0.08      0.12       666
           3       0.00      0.00      0.00       112
           4       0.93      0.99      0.96     13954
           5       0.08      0.01      0.01       194
           6       0.00      0.00      0.00         4
           7       0.67      0.67      0.67         9

    accuracy                           0.93     15000
   macro avg       0.25      0.22      0.22     15000
weighted avg       0.88      0.93      0.90     15000

Confusion Matrix:
[[    0     0     1     1    19     0     0     0]
 [    0     0     1     0    39     0     0     0]
 [    0     1    53     0   611     0     0     1]
 [    0     0    11     0   101     0     0     0]
 [    0     0   113     2 13827    12     0     0]
 [    0     0     1     0   191     1     0     1]
 [    0     0     1     0     2     0     0     1]
 [    0     0     2     0     1     0     0     6]]
Memory Used (MB): 459.1171875
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-
packages/sklearn/neighbors/_classification.py:215: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  return self._fit(X, y)


Results for k=11:
```

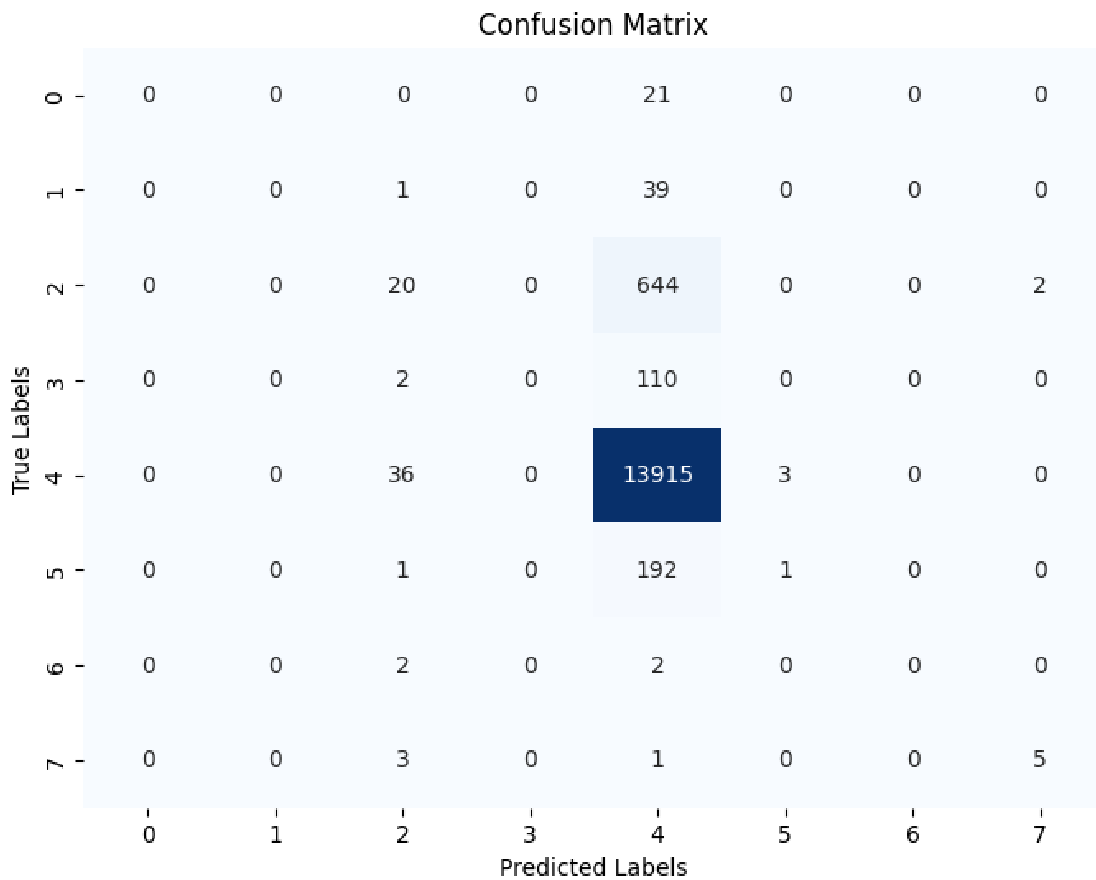
```
Training Time (seconds): 0.24759531021118164
Accuracy: 0.9276
Classification Report:
              precision    recall  f1-score   support

           0       0.00      0.00      0.00        21
           1       0.00      0.00      0.00        40
           2       0.31      0.06      0.10       666
           3       0.00      0.00      0.00       112
           4       0.93      0.99      0.96     13954
           5       0.00      0.00      0.00       194
           6       0.00      0.00      0.00         4
           7       0.60      0.67      0.63         9

    accuracy                           0.93     15000
   macro avg       0.23      0.21      0.21     15000
weighted avg       0.88      0.93      0.90     15000

Confusion Matrix:
[[    0     0     0     0    21     0     0     0]
 [    0     0     1     0    39     0     0     0]
 [    0     0    39     1   624     0     0     2]
 [    0     0     4     0   108     0     0     0]
 [    0     0    78     1 13869     6     0     0]
 [    0     0     2     0   191     0     0     1]
 [    0     0     1     0     2     0     0     1]
 [    0     0     2     0     1     0     0     6]]
Memory Used (MB): 459.1171875
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-
packages/sklearn/neighbors/_classification.py:215: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
```

```
  return self._fit(X, y)

Results for k=13:
Training Time (seconds): 0.16437482833862305
Accuracy: 0.9281333333333334
Classification Report:
              precision    recall  f1-score   support

           0       0.00      0.00      0.00        21
           1       0.00      0.00      0.00        40
           2       0.31      0.05      0.08       666
           3       0.00      0.00      0.00       112
           4       0.93      0.99      0.96     13954
           5       0.00      0.00      0.00       194
           6       0.00      0.00      0.00         4
           7       0.60      0.67      0.63         9

    accuracy                           0.93     15000
   macro avg       0.23      0.21      0.21     15000
weighted avg       0.88      0.93      0.90     15000

Confusion Matrix:
[[    0     0     0     0    21     0     0     0]
 [    0     0     1     0    39     0     0     0]
 [    0     0    32     0   632     0     0     2]
 [    0     0     4     0   108     0     0     0]
 [    0     0    62     0 13884     8     0     0]
 [    0     0     1     0   192     0     0     1]
 [    0     0     1     0     2     0     0     1]
 [    0     0     2     0     1     0     0     6]]
Memory Used (MB): 460.14453125

/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-
```

```
packages/sklearn/neighbors/_classification.py:215: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  return self._fit(X, y)


Results for k=15:
Training Time (seconds): 0.0906221866607666
Accuracy: 0.9291333333333334
Classification Report:
              precision    recall  f1-score   support

           0       0.00      0.00      0.00        21
           1       0.00      0.00      0.00        40
           2       0.33      0.05      0.08       666
           3       0.00      0.00      0.00       112
           4       0.93      1.00      0.96     13954
           5       0.20      0.01      0.01       194
           6       0.00      0.00      0.00         4
           7       0.71      0.56      0.63         9

    accuracy                           0.93     15000
   macro avg       0.27      0.20      0.21     15000
weighted avg       0.89      0.93      0.90     15000

Confusion Matrix:
[[    0     0     0     0    21     0     0     0]
 [    0     0     1     0    39     0     0     0]
 [    0     0    30     0   634     0     0     2]
 [    0     0     3     0   109     0     0     0]
 [    0     0    49     0 13901     4     0     0]
 [    0     0     2     0   191     1     0     0]
 [    0     0     2     0     2     0     0     0]
 [    0     0     3     0     1     0     0     5]]
Memory Used (MB): 460.6328125

/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
```

```
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))


Results for k=17:
Training Time (seconds): 0.12061357498168945
Accuracy: 0.9294
Classification Report:
              precision    recall  f1-score   support

           0       0.00      0.00      0.00        21
           1       0.00      0.00      0.00        40
           2       0.31      0.03      0.05       666
           3       0.00      0.00      0.00       112
           4       0.93      1.00      0.96     13954
           5       0.25      0.01      0.01       194
           6       0.00      0.00      0.00         4
           7       0.71      0.56      0.63         9

    accuracy                           0.93     15000
   macro avg       0.28      0.20      0.21     15000
weighted avg       0.88      0.93      0.90     15000

Confusion Matrix:
[[    0     0     0     0    21     0     0     0]
 [    0     0     1     0    39     0     0     0]
 [    0     0    20     0   644     0     0     2]
 [    0     0     2     0   110     0     0     0]
 [    0     0    36     0 13915     3     0     0]
 [    0     0     1     0   192     1     0     0]
 [    0     0     2     0     2     0     0     0]
 [    0     0     3     0     1     0     0     5]]
Memory Used (MB): 461.08984375
```




```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 17 members, which is less
than n_splits=20.
  warnings.warn(
```

```
Cross-Validation Scores: [0.92866667 0.918      0.91233333 0.91866667 0.931
0.927
 0.92733333 0.91033333 0.922      0.92666667 0.932      0.93133333
 0.92933333 0.93266667 0.931      0.93166667 0.93266667 0.93133333
 0.93133333 0.929     ]
Average Accuracy: 0.9267166666666666
time: 34.9 s (started: 2024-04-13 14:06:21 +00:00)
```

[60]:

```
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=15)  # 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)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 17 members, which is less
than n_splits=20.
  warnings.warn(
```

```
Cross-Validation Scores: [0.928      0.92       0.91666667 0.925      0.93066667
0.927
 0.92566667 0.91566667 0.924      0.92766667 0.932      0.931
 0.92966667 0.93166667 0.93133333 0.931      0.931      0.931
 0.93133333 0.92966667]
Average Accuracy: 0.9275
time: 8.01 s (started: 2024-04-13 14:06:56 +00:00)
```

8 LR

[61]:
```
#########   LR
```

```
time: 403 µs (started: 2024-04-13 14:07:04 +00:00)
```

[62]:
```
# Create and fit a Logistic Regression model
logreg = LogisticRegression(random_state=45011, solver='liblinear')
logreg.fit(df1_inputs_train, df1_output_train)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/utils/validation.py:1143:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
```

[62]:
```
LogisticRegression(random_state=45011, solver='liblinear')
```

```
time: 282 ms (started: 2024-04-13 14:07:04 +00:00)
```

[63]:
```
# Make predictions using the trained model
y_pred = logreg.predict(df1_inputs_test)

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

```python
# 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: 0.9302666666666667
Classification Report:
              precision    recall  f1-score   support

           0       0.00      0.00      0.00        21
           1       0.00      0.00      0.00        40
           2       0.00      0.00      0.00       666
           3       0.00      0.00      0.00       112
           4       0.93      1.00      0.96     13954
           5       0.00      0.00      0.00       194
           6       0.00      0.00      0.00         4
           7       0.00      0.00      0.00         9

    accuracy                           0.93     15000
   macro avg       0.12      0.12      0.12     15000
weighted avg       0.87      0.93      0.90     15000

Confusion Matrix:
[[    0     0     0     0    21     0     0     0]
 [    0     0     0     0    40     0     0     0]
 [    0     0     0     0   666     0     0     0]
 [    0     0     0     0   112     0     0     0]
 [    0     0     0     0 13954     0     0     0]
 [    0     0     0     0   194     0     0     0]
 [    0     0     0     0     4     0     0     0]
 [    0     0     0     0     9     0     0     0]]
time: 91.1 ms (started: 2024-04-13 14:07:05 +00:00)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
```

```
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```

[77]:

```python
# Get the coefficients from the logistic regression model
coefficients = logreg.coef_[0]

# Create a DataFrame to store the feature coefficients
lr_imp_features = pd.DataFrame({'feature': df1_inputs_names, 'importance':␣
 ↪coefficients})
lr_imp_features['importance'] = lr_imp_features['importance'].abs()  # Take␣
 ↪absolute values
lr_imp_features.sort_values('importance', ascending=False, inplace=True)
print(lr_imp_features)
```

```
                    feature    importance
3  hud_median_family_income  9.955654e-05
0                msamd_name  1.011932e-08
2         loan_purpose_name  1.903765e-09
1            loan_type_name  9.476882e-10
4          loan_amount_000s  6.487305e-12
time: 19.7 ms (started: 2024-04-13 14:22:21 +00:00)
```

[64]:

```python
from sklearn.metrics import log_loss

# Predict probabilities for each class
y_pred_proba_lr = logreg.predict_proba(df1_inputs_test)

# Calculate log loss (which is similar to entropy)
log_loss_lr = log_loss(df1_output_test, y_pred_proba_lr)

print("Log Loss (Entropy) for Logistic Regression:", log_loss_lr)
```

```
Log Loss (Entropy) for Logistic Regression: 0.29761744182376243
time: 37.2 ms (started: 2024-04-13 14:07:05 +00:00)
```

[65]:

```python
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, classification_report,␣
 ↪confusion_matrix
import time
import psutil

# Function to measure memory usage
def memory_usage():
    process = psutil.Process()
    return process.memory_info().rss / 1024 ** 2  # Memory usage in MB
```

```python
# Start time
start_time = time.time()

# Create and fit a Logistic Regression model
logreg = LogisticRegression(random_state=45007, solver='liblinear')
logreg.fit(df1_inputs_train, df1_output_train)

# End time
end_time = time.time()

# Time taken for training
training_time = end_time - start_time

# Make predictions using the trained model
start_time = time.time()  # Start time for prediction
y_pred = logreg.predict(df1_inputs_test)
end_time = time.time()  # End time for prediction

# Time taken for prediction
prediction_time = end_time - start_time

# Calculate accuracy
accuracy = accuracy_score(df1_output_test, y_pred)

# Memory usage
memory_used = memory_usage()

# Print results
print("Time taken for training (seconds):", training_time)
print("Time taken for prediction (seconds):", prediction_time)
print("Accuracy:", accuracy)
print("Memory used (MB):", memory_used)

# 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}')
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/utils/validation.py:1143:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
```

```
Time taken for training (seconds): 0.3720817565917969
Time taken for prediction (seconds): 0.003966331481933594
Accuracy: 0.9302666666666667
Memory used (MB): 466.4453125
Classification Report:
              precision    recall  f1-score   support

           0       0.00      0.00      0.00        21
           1       0.00      0.00      0.00        40
           2       0.00      0.00      0.00       666
           3       0.00      0.00      0.00       112
           4       0.93      1.00      0.96     13954
           5       0.00      0.00      0.00       194
           6       0.00      0.00      0.00         4
           7       0.00      0.00      0.00         9

    accuracy                           0.93     15000
   macro avg       0.12      0.12      0.12     15000
weighted avg       0.87      0.93      0.90     15000

Confusion Matrix:
[[    0     0     0     0    21     0     0     0]
 [    0     0     0     0    40     0     0     0]
 [    0     0     0     0   666     0     0     0]
 [    0     0     0     0   112     0     0     0]
 [    0     0     0     0 13954     0     0     0]
 [    0     0     0     0   194     0     0     0]
 [    0     0     0     0     4     0     0     0]
 [    0     0     0     0     9     0     0     0]]
time: 444 ms (started: 2024-04-13 14:07:05 +00:00)

/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```

9 SVM

[66]:
```
###########        SSSSSSSSSSS      VVVVVVVVVVV       MMMMMMMMMMM
```

time: 447 µs (started: 2024-04-13 14:07:05 +00:00)

[67]:
```
# 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: 475 µs (started: 2024-04-13 14:07:05 +00:00)

[68]:
```
classifier = SVC(kernel = 'rbf', random_state = 45005)
classifier.fit(df1_inputs_train, df1_output_train)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/utils/validation.py:1143:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
```

[68]: SVC(random_state=45005)

time: 12.2 s (started: 2024-04-13 14:07:05 +00:00)

[69]:
```
y_pred = classifier.predict(df1_inputs_test)
```

time: 5.7 s (started: 2024-04-13 14:07:18 +00:00)

[78]:
```
# Get the indices of support vectors
support_vector_indices = classifier.support_

# Extract the support vectors from the training data
support_vectors = df1_inputs_train.iloc[support_vector_indices]

# Calculate the mean value of each feature across support vectors
svm_feature_importance = support_vectors.mean()

# Create a DataFrame to store the feature importance
svm_imp_features = pd.DataFrame({'feature': df1_inputs_names, 'importance':␣
↪svm_feature_importance})
```

```
svm_imp_features.sort_values('importance', ascending=False, inplace=True)
print(svm_imp_features)
```

```
                                              feature    importance
hud_median_family_income  hud_median_family_income  73931.750494
msamd_name                                  msamd_name      6.869248
loan_purpose_name                    loan_purpose_name      1.421012
loan_type_name                          loan_type_name      0.592216
loan_amount_000s                      loan_amount_000s      0.005059
time: 31.1 ms (started: 2024-04-13 14:23:27 +00:00)
```

[70]:
```
cm = confusion_matrix(df1_output_test, y_pred)
print(cm)
accuracy_score(df1_output_test,y_pred)
```

```
[[    0     0     0     0    21     0     0     0]
 [    0     0     0     0    40     0     0     0]
 [    0     0     0     0   666     0     0     0]
 [    0     0     0     0   112     0     0     0]
 [    0     0     0     0 13954     0     0     0]
 [    0     0     0     0   194     0     0     0]
 [    0     0     0     0     4     0     0     0]
 [    0     0     0     0     9     0     0     0]]
```

[70]: 0.9302666666666667

```
time: 31.2 ms (started: 2024-04-13 14:07:23 +00:00)
```

[71]:
```
# Get decision function values for each sample
decision_values = classifier.decision_function(df1_inputs_test)

# Calculate margin distances
margin_distances = 2 * (1 - decision_values)

# Calculate Gini impurity
gini_impurity_svm = 1 - (margin_distances ** 2).mean()

print("Gini Impurity for SVM:", gini_impurity_svm)
```

```
Gini Impurity for SVM: -49.57681961141723
time: 4.82 s (started: 2024-04-13 14:07:23 +00:00)
```

[72]:
```
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score, classification_report,␣
↪confusion_matrix
import time
import psutil
```

```python
# Function to measure memory usage
def memory_usage():
    process = psutil.Process()
    return process.memory_info().rss / 1024 ** 2  # Memory usage in MB

# Start time
start_time = time.time()

# Create and fit an SVM classifier with RBF kernel
svm_classifier = SVC(kernel='rbf', random_state=45011)
svm_classifier.fit(df1_inputs_train, df1_output_train)

# End time
end_time = time.time()

# Time taken for training
training_time = end_time - start_time

# Make predictions using the trained model
start_time = time.time()  # Start time for prediction
y_pred = svm_classifier.predict(df1_inputs_test)
end_time = time.time()  # End time for prediction

# Time taken for prediction
prediction_time = end_time - start_time

# Calculate accuracy
accuracy = accuracy_score(df1_output_test, y_pred)

# Memory usage
memory_used = memory_usage()

# Print results
print("Time taken for training (seconds):", training_time)
print("Time taken for prediction (seconds):", prediction_time)
print("Accuracy:", accuracy)
print("Memory used (MB):", memory_used)

# 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}')
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/utils/validation.py:1143:
```

```
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
```

```
Time taken for training (seconds): 11.703947305679321
Time taken for prediction (seconds): 4.541745185852051
Accuracy: 0.9302666666666667
Memory used (MB): 482.22265625
Classification Report:
              precision    recall  f1-score   support

           0       0.00      0.00      0.00        21
           1       0.00      0.00      0.00        40
           2       0.00      0.00      0.00       666
           3       0.00      0.00      0.00       112
           4       0.93      1.00      0.96     13954
           5       0.00      0.00      0.00       194
           6       0.00      0.00      0.00         4
           7       0.00      0.00      0.00         9

    accuracy                           0.93     15000
   macro avg       0.12      0.12      0.12     15000
weighted avg       0.87      0.93      0.90     15000

Confusion Matrix:
[[    0     0     0     0    21     0     0     0]
 [    0     0     0     0    40     0     0     0]
 [    0     0     0     0   666     0     0     0]
 [    0     0     0     0   112     0     0     0]
 [    0     0     0     0 13954     0     0     0]
 [    0     0     0     0   194     0     0     0]
 [    0     0     0     0     4     0     0     0]
 [    0     0     0     0     9     0     0     0]]
time: 16.3 s (started: 2024-04-13 14:07:28 +00:00)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
```

```
control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```

10 COMPARISON

[73]:
```python
import pandas as pd
import time
import psutil
from sklearn.metrics import accuracy_score, classification_report,␣
↪confusion_matrix
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
from sklearn.model_selection import train_test_split

# Function to measure memory usage
def memory_usage():
    process = psutil.Process()
    return process.memory_info().rss / 1024 ** 2  # Memory usage in MB

# Data preprocessing and splitting
# Assuming you have your data loaded into cars_inputs and cars_output
df1_inputs_train, df1_inputs_test, df1_output_train, df1_output_test =␣
↪train_test_split(df1_inputs, df1_output, test_size=0.2, random_state=42)

# Results storage
results = []

# Decision Tree
dt_start_time = time.time()
dt_model = DecisionTreeClassifier(random_state=42)
dt_model.fit(df1_inputs_train, df1_output_train)
dt_training_time = time.time() - dt_start_time
dt_memory_used = memory_usage()
dt_pred = dt_model.predict(df1_inputs_test)
dt_accuracy = accuracy_score(df1_output_test, dt_pred)
results.append({'Model': 'Decision Tree', 'Time Taken (s)': dt_training_time,␣
↪'Memory Used (MB)': dt_memory_used, 'Accuracy': dt_accuracy})

# KNN
k_values = [5, 7, 9, 11, 13, 15]
for k in k_values:
    knn_start_time = time.time()
    knn = KNeighborsClassifier(n_neighbors=k)
    knn.fit(df1_inputs_train, df1_output_train)
    knn_training_time = time.time() - knn_start_time
```

```
    knn_memory_used = memory_usage()
    knn_pred = knn.predict(df1_inputs_test)
    knn_accuracy = accuracy_score(df1_output_test, knn_pred)
    results.append({'Model': f'KNN (k={k})', 'Time Taken (s)':␣
↪knn_training_time, 'Memory Used (MB)': knn_memory_used, 'Accuracy':␣
↪knn_accuracy})

# Logistic Regression
lr_start_time = time.time()
logreg = LogisticRegression(random_state=45011, solver='liblinear')
logreg.fit(df1_inputs_train, df1_output_train)
lr_training_time = time.time() - lr_start_time
lr_memory_used = memory_usage()
lr_pred = logreg.predict(df1_inputs_test)
lr_accuracy = accuracy_score(df1_output_test, lr_pred)
results.append({'Model': 'Logistic Regression', 'Time Taken (s)':␣
↪lr_training_time, 'Memory Used (MB)': lr_memory_used, 'Accuracy':␣
↪lr_accuracy})

# SVM
svm_start_time = time.time()
svm_classifier = SVC(kernel='rbf', random_state=45011)
svm_classifier.fit(df1_inputs_train, df1_output_train)
svm_training_time = time.time() - svm_start_time
svm_memory_used = memory_usage()
svm_pred = svm_classifier.predict(df1_inputs_test)
svm_accuracy = accuracy_score(df1_output_test, svm_pred)
results.append({'Model': 'SVM', 'Time Taken (s)': svm_training_time, 'Memory␣
↪Used (MB)': svm_memory_used, 'Accuracy': svm_accuracy})

# Create DataFrame
results_df = pd.DataFrame(results)

# Display the results
print(results_df)
```

```
/usr/local/lib/python3.10/dist-
packages/sklearn/neighbors/_classification.py:215: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  return self._fit(X, y)
/usr/local/lib/python3.10/dist-
packages/sklearn/neighbors/_classification.py:215: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  return self._fit(X, y)
/usr/local/lib/python3.10/dist-
```

```
packages/sklearn/neighbors/_classification.py:215: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  return self._fit(X, y)
/usr/local/lib/python3.10/dist-
packages/sklearn/neighbors/_classification.py:215: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  return self._fit(X, y)
/usr/local/lib/python3.10/dist-
packages/sklearn/neighbors/_classification.py:215: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  return self._fit(X, y)
/usr/local/lib/python3.10/dist-
packages/sklearn/neighbors/_classification.py:215: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  return self._fit(X, y)
/usr/local/lib/python3.10/dist-packages/sklearn/utils/validation.py:1143:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.10/dist-packages/sklearn/utils/validation.py:1143:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
```

	Model	Time Taken (s)	Memory Used (MB)	Accuracy
0	Decision Tree	0.085383	482.472656	0.899917
1	KNN (k=5)	0.045039	482.472656	0.919667
2	KNN (k=7)	0.044112	482.472656	0.925583
3	KNN (k=9)	0.047607	482.472656	0.927500
4	KNN (k=11)	0.075459	482.472656	0.928583
5	KNN (k=13)	0.064560	482.472656	0.931000
6	KNN (k=15)	0.047166	482.472656	0.931333
7	Logistic Regression	0.334362	482.472656	0.934083
8	SVM	13.308904	488.410156	0.934083

```
time: 24.9 s (started: 2024-04-13 14:07:44 +00:00)
```

[74]:

```python
# 4 minutes to load

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: [0.93025 0.93025 0.93025 0.93025 0.93025]
Average Cross-validation score: 0.93025
time: 1min 25s (started: 2024-04-13 14:08:09 +00:00)
```

[75]:

```
#  4 minutes to load

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): [0.89663522 0.89663522 0.89663522
0.89663522 0.89663522]
Average Cross-validation score (Weighted F1): 0.8966352156456419
time: 1min 30s (started: 2024-04-13 14:09:34 +00:00)
```