FOUNDATION FOR ORGANISATIONAL

RESEARCH AND EDUCATION

NEW DELHI

Academic Session 2023-2025

Classification and Prediction (Risk Assessment Data) based on Cluster data

Machine Learning for Managers

(FMG 32 Section B)

Submitted to

Prof. Amarnath Mitra

Submitted by:

321105 – Samriddhi Pandey

Table of Contents

Objectives of the Project

1.1. Classification of Consumer Data into {Segments | Clusters | Classes} using Cross-Validation

1.2. Classification of Consumer Data into {Segments | Clusters | Classes} using Ensemble Methods

1.3. Determination of an Appropriate Classification Model (Default vs Cross-Validation or Ensemble)

1.4. Identification of Important | Contributing | Significant Variables or Features and their Thresholds for Classification

Analysis

2.1 Data Analysis

2.1.1. Classification of Consumer Data into {Segments | Clusters | Classes} using Cross-Validation

Cross-Validation using Decision Tree

Cross-validation using a decision tree involves splitting the dataset into k subsets, training the decision tree on k-1 subsets and validating on the remaining subset by repeating this process k times and averaging the results to assess the model's performance and generalization ability.

Cross-Validation using Other Methods

Logistic Regression

Cross-validation with logistic regression involves partitioning the dataset into training and validation sets, fitting the logistic regression model on the training data and evaluating its performance on the validation set. This process is repeated multiple times with different partitions to estimate the model's generalization performance and minimize overfitting.

K-Nearest Neighbours

Cross-validation with KNN entails splitting the dataset into training and validation sets, then iterating through different values of k (number of nearest neighbours) to find the optimal k value that minimizes error on the validation set. This process helps assess the KNN model's performance and its ability to generalize to new data.

<u>Different classification model results before cross validation</u>

Decision Tree

Row ID	TruePo…	FalsePo…	TrueNe…	FalseN…	Recall	Precision	Sensitivity	Specificity	F-meas…	Accuracy	Cohen'…
cluster_0	2200	8422	11578	7800	0.22	0.207	0.22	0.579	0.213	?	?
cluster_2	2113	7963	12037	7887	0.211	0.21	0.211	0.602	0.211	?	?
cluster_1	1796	7506	12494	8204	0.18	0.193	0.18	0.625	0.186	?	?
Overall	?	?	?	?	?	?	?	?	?	0.204	-0.195

Logistic Regression

Row ID	TruePo…	FalsePo…	TrueNe…	FalseN…	Recall	Precision	Sensitivity	Specificity	F-meas…	Accuracy	Cohen'…
cluster_0	10000	0	20000	0	1	1	1	1	1	?	?
cluster_2	10000	0	20000	0	1	1	1	1	1	?	?
cluster_1	10000	0	20000	0	1	1	1	1	1	?	?
Overall	?	?	?	?	?	?	?	?	?	1	1

K Nearest Neighbour

Row ID	TruePo…	FalsePo…	TrueNe…	FalseN…	Recall	Precision	Sensitivity	Specificity	F-meas…	Accuracy	Cohen'…
cluster_0	9945	65	19935	55	0.995	0.994	0.995	0.997	0.994	?	?
cluster_2	9642	393	19607	358	0.964	0.961	0.964	0.98	0.963	?	?
cluster_1	9620	335	19665	380	0.962	0.966	0.962	0.983	0.964	?	?
Overall	?	?	?	?	?	?	?	?	?	0.974	0.96

Different classification model results after cross validation and Ensemble Method

Decision Tree

Row ID	TruePo...	FalsePo...	TrueNe...	FalseN...	Recall	Precision	Sensitivity	Specificity	F-meas...	Accuracy	Cohen'...
cluster_0	5380	29908	36758	27954	0.161	0.152	0.161	0.551	0.157	?	?
cluster_2	5544	28559	38108	27789	0.166	0.163	0.166	0.572	0.164	?	?
cluster_1	4364	26245	40422	28969	0.131	0.143	0.131	0.606	0.136	?	?
Overall	?	?	?	?	?	?	?	?	?	0.153	-0.271

Logistic Regression

Row ID	TruePo...	FalsePo...	TrueNe...	FalseN...	Recall	Precision	Sensitivity	Specificity	F-meas...	Accuracy	Cohen'...
cluster_0	33334	0	66666	0	1	1	1	1	1	?	?
cluster_2	33333	0	66667	0	1	1	1	1	1	?	?
cluster_1	33333	0	66667	0	1	1	1	1	1	?	?
Overall	?	?	?	?	?	?	?	?	?	1	1

K Nearest Neighbour

Row ID	TruePo...	FalsePo...	TrueNe...	FalseN...	Recall	Precision	Sensitivity	Specificity	F-meas...	Accuracy	Cohen'...
cluster_0	33332	2	66664	2	1	1	1	1	1	?	?
cluster_2	33332	1	66666	1	1	1	1	1	1	?	?
cluster_1	33330	3	66664	3	1	1	1	1	1	?	?
Overall	?	?	?	?	?	?	?	?	?	1	1

2.1.2. Classification of Consumer Data into {Segments | Clusters | Classes} using Ensemble Methods

Ensemble Method using Random Forest

Random forest is an ensemble learning method where multiple decision trees are trained on random subsets of the data and features. During prediction, each tree votes on the outcome and the final prediction is determined by the majority vote.

This approach improves prediction accuracy and reduces overfitting compared to individual decision trees.

Random Forest

Row ID	TruePo...	FalsePo...	TrueNe...	FalseN...	Recall	Precision	Sensitivity	Specificity	F-meas...	Accuracy	Cohen'...
cluster_0	6665	0	13334	1	1	1	1	1	1	?	?
cluster_2	6667	0	13333	0	1	1	1	1	1	?	?
cluster_1	6667	1	13332	0	1	1	1	1	1	?	?
Overall	?	?	?	?	?	?	?	?	?	1	1

Observations

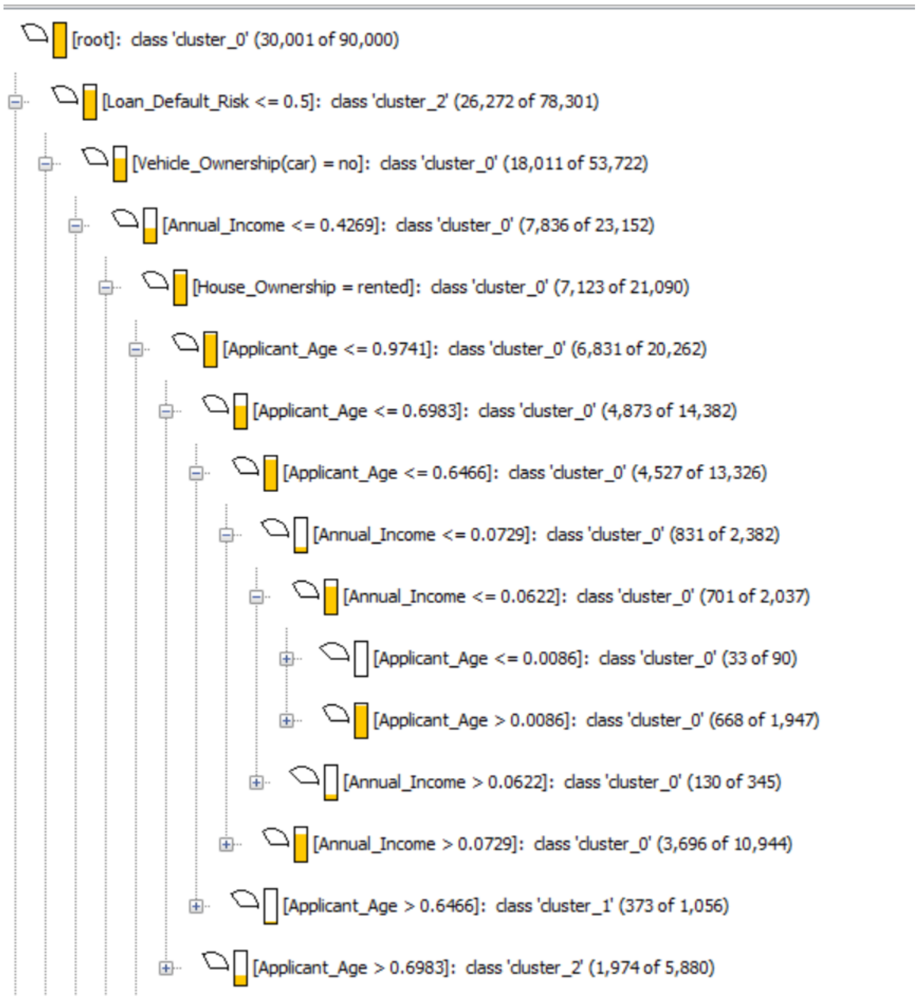
3.1 Determination of an Appropriate Classification Model (Default vs Cross-Validation or Ensemble)

	Cross Validation			Ensemble Learning
Metrics	**Decision Tree**	**Logistic Regression**	**KNN**	**Random Forest**
Accuracy (in %)	15.288%	100%	99.994%	99.995%
Error (in %)	84.712%	0%	0.006%	0.005%
Cohen's Kappa (in %)	-0.271%	1%	1%	1%
Correctly classified	15288	100000	99994	19999
Wrongly Classified	84712	0	6	1

- **Cross validation using Decision Trees**: It shows 15.288% accuracy and Cohen's Kappa scores indicating poor performance.
- **Cross validation using Logistic Regression:** This algorithm Shows higher accuracy and Cohen's Kappa score than decision tree, indicating robustness and effectiveness for the given dataset.
- **Cross validation using KNN:** Performs significantly higher compared to other models, with the highest accuracy and Cohen's Kappa score. This suggests that KNN might be suitable for this dataset or may require further tuning of hyperparameters.
- **Random Forest (Ensemble learning):** Performs exceptionally well with high accuracy and Cohen's Kappa scores.

For this dataset, ensemble learning methods like Random Forest along with Logistic Regression seem to be the most effective models in terms of accuracy and robustness. KNN also performs well and provides interpretable results which can be advantageous in certain scenarios. However, Decision Tree appear to be less suitable due to their less accuracy.

3.2 Identification of Important | Contributing | Significant Variables and their Thresholds for Classification


[root]: class 'cluster_0' (30,001 of 90,000)
[Loan_Default_Risk <= 0.5]: class 'cluster_2' (26,272 of 78,301)
[Vehicle_Ownership(car) = no]: class 'cluster_0' (18,011 of 53,722)
[Annual_Income <= 0.4269]: class 'cluster_0' (7,836 of 23,152)
[House_Ownership = rented]: class 'cluster_0' (7,123 of 21,090)
[Applicant_Age <= 0.9741]: class 'cluster_0' (6,831 of 20,262)
[Applicant_Age <= 0.6983]: class 'cluster_0' (4,873 of 14,382)
[Applicant_Age <= 0.6466]: class 'cluster_0' (4,527 of 13,326)
[Annual_Income <= 0.0729]: class 'cluster_0' (831 of 2,382)
[Annual_Income <= 0.0622]: class 'cluster_0' (701 of 2,037)
[Applicant_Age <= 0.0086]: class 'cluster_0' (33 of 90)
[Applicant_Age > 0.0086]: class 'cluster_0' (668 of 1,947)
[Annual_Income > 0.0622]: class 'cluster_0' (130 of 345)
[Annual_Income > 0.0729]: class 'cluster_0' (3,696 of 10,944)
[Applicant_Age > 0.6466]: class 'cluster_1' (373 of 1,056)
[Applicant_Age > 0.6983]: class 'cluster_2' (1,974 of 5,880)


[Applicant_Age <= 0.6466]: class 'cluster_0' (4,527 of 13,326)
[Annual_Income <= 0.0729]: class 'cluster_0' (831 of 2,382)
[Annual_Income <= 0.0622]: class 'cluster_0' (701 of 2,037)
[Applicant_Age <= 0.0086]: class 'cluster_0' (33 of 90)
[Applicant_Age > 0.0086]: class 'cluster_0' (668 of 1,947)
[Annual_Income > 0.0622]: class 'cluster_0' (130 of 345)
[Annual_Income > 0.0729]: class 'cluster_0' (3,696 of 10,944)
[Applicant_Age > 0.6466]: class 'cluster_1' (373 of 1,056)
[Applicant_Age > 0.6983]: class 'cluster_2' (1,974 of 5,880)
[Applicant_Age > 0.9741]: class 'cluster_0' (292 of 828)
[House_Ownership = owned]: class 'cluster_0' (423 of 1,204)
[House_Ownership = norent_noown]: class 'cluster_0' (290 of 858)
[Annual_Income > 0.4269]: class 'cluster_1' (10,251 of 30,570)
[Vehicle_Ownership(car) = yes]: class 'cluster_2' (8,554 of 24,579)
[Loan_Default_Risk > 0.5]: class 'cluster_0' (3,989 of 11,699)

Managerial Insights

Logistic Regression has the highest accuracy followed closely by Random Forest Ensemble Learning.
Decision tree has the lowest of accuracy when compared to all the models, thus it won't be preferred when classifying risk assessment on the basis of affluency and loan risk.

Managerial insights according to the appropriate model (Random Forest ensemble learning and Logistic Regression)

- **Credit Scoring:**
  Logistic regression can be used to predict the probability of a customer defaulting on a loan based on various features such as the customer's income, credit score, employment status, and loan amount.
  Random Forest can be used to predict the probability of a customer defaulting on a loan based on various features. It can handle missing values, outliers, and non-linear relationships.

- **Fraud Detection:**
  Logistic regression can help identify fraudulent transactions by classifying transactions based on features like transaction amount, location, and time of day.
  Random Forest can be used to detect fraudulent transactions. It can handle high-dimensional data better and is known for its ability to handle complex data, reduce overfitting, and provide reliable forecasts in different environments.

- **Customer Churn Prediction:**
  Logistic regression can be used to predict whether a customer will churn (i.e., stop doing business with the company), which is crucial for customer relationship management.
  Random Forest can be used to predict customer churn. It can handle imbalanced datasets better and can be used for feature importance analysis.