**Predicting Customer Response Usings Decision Trees, Random Forests, K-NN, and Logistic Regression Models**

Group 7: Gurjas Singh Chawla, Cat Kim, Khanh Huynh Bao Nguyen, Sriram Srinivasan

The Jindal School of Management, The University of Texas at Dallas

BUAN 6383/ MIS 6386: Modeling for Business Analytics

Professor Syam Menon, Ph.D.

October 6, 2024

Table of Contents

[1. Data Preprocessing: Identifying and Transforming Non- Binary Skewed Variables 3](#_Toc179118987)

[2. Generate and analyze the decision tree with no limitations on depth and entropy as the metric. 4](#_Toc179118988)

[3. Identify the best decision tree classifier by pruning the tree at different depths using 10-fold cross validation. 6](#_Toc179118989)

[4. Developing a random forest classifier with 100 trees and comparing performance 8](#_Toc179118990)

[5. Developing a random forest classifier with 50 trees and comparing performance 10](#_Toc179118991)

[6. Evaluating k-nearest neighbor models using 10-fold cross validation to identify the optimal value of k 10](#_Toc179118992)

[7. Developing a logistic regression model and evaluating its accuracy using 10-fold cross validation 12](#_Toc179118993)

[8. Developing a logistic regression model on the training dataset 12](#_Toc179118994)

[9. Evaluating the four best models using 10-fold cross validation to determine the recommended model 13](#_Toc179118995)

[10. Developing the final version of the recommended model using the dataset & evaluating its accuracy 14](#_Toc179118996)

[11. Making predictions on customer response using the final model and threshold for classification 15](#_Toc179118997)

[12. Analyzing model predictions for lapsing customers and comparing results with other customer segments 15](#_Toc179118998)

A screenshot of a computer

Description automatically generated

Figure 1: List of variables and descriptions for datasets

# Data Preprocessing: Identifying and Transforming Non- Binary Skewed Variables

1. **Explain what you found, what transformations you applied, and why.**

Table 1:Distribution of Non-Binary Values (Training Dataset)

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Skewness | Symmetric? | Transformation Applied |
| n24 | 1.935635 | Highly positively skewed | Log transformation – ln(x) |
| rev24 | 7.221135 | Highly positively skewed | Log transformation – ln(x) |
| revlast | 11.968288 | Highly positively skewed | Log transformation – ln(1+x) |
| elpsdm | -0.774676 | Somewhat skewed | None |
| ordfreq | 0.611939 | Somewhat skewed | None |
| ordcat | -0.338221 | Approximately symmetric | None |

* Variables n24, revlast, and rev24 are highly positively skewed and need to be transformed
* Used np.log (natural logarithm) to transform variables n24 and rev24
* Used np.log1p (NumPy function computes ln(1+x), where x is the input array) to transform variable revlast because there are eight values of 0.
* Reasons for different log-transformations:
* For large values of x, both transformations behave similarly. For instance, log(500) and log(501) are almost indistinguishable, because the difference between x and x+1 becomes negligible as x increases. So, for larger values, there is little difference between log(x) and log1p(x).
* When you are dealing with a dataset that has a significant number of zeros or small values, applying log1p(x) makes sense because it handles these problematic values effectively. It ensures that the transformation is smooth and prevents skewness from being exaggerated due to large negative values or undefined results from applying log(x) on small values.
* For example, if a column contains values ranging from 0 to small positive numbers (e.g., 0, 0.5, 1), applying log1p(x) will give reasonable results like log(1+0)=0, log(1+0.5)≈0.405, and log(1+1)=0.693, whereas applying log(x) would lead to issues for zero values and possibly skew the transformation.

A close-up of a graph

Description automatically generated

Figure 2: Histograms of non-binary variables with continuous features and bar charts of discrete non-binary variables from the training dataset

# Generate and analyze the decision tree with no limitations on depth and entropy as the metric.

1. **What is the depth of the tree that is generated?** 26
2. **Provide a plot of the tree**

A line of lines and dots

Description automatically generated with medium confidence

Figure 3: Decision tree of the training dataset with no limitations and using entropy as the metric

# Identify the best decision tree classifier by pruning the tree at different depths using 10-fold cross validation.

A graph with a line going up

Description automatically generated

Figure 4: Line plot of the decision tree accuracy versus the depths in the training dataset

* 1. **Provide your reasoning for using the values of tree-depth that you tried.**
* First, train decision trees with depths from 1-26 (depth of the largest tree generated in Question 3), then select top 6 trees with the highest accuracies for further analysis
* Highest accuracy levels
* At depths 1-3 cross-validation accuracy is at its highest
* Peaks at depth 6 at 69.95% then begins to gradually decline and show signs of overfitting
* Low complexity and easy to interpret
* Using fewer splits, the model can generalize well while capture complexity
  1. **Based on your results, what depth do you recommend?**
* Max depth of 3 because it is the first depth to isolate the `customer responded` class (i.e., class of interest) in a node despite the unbalanced dataset
* 1416 `no response` (70.8%)
* 584 `response` (29.2%)

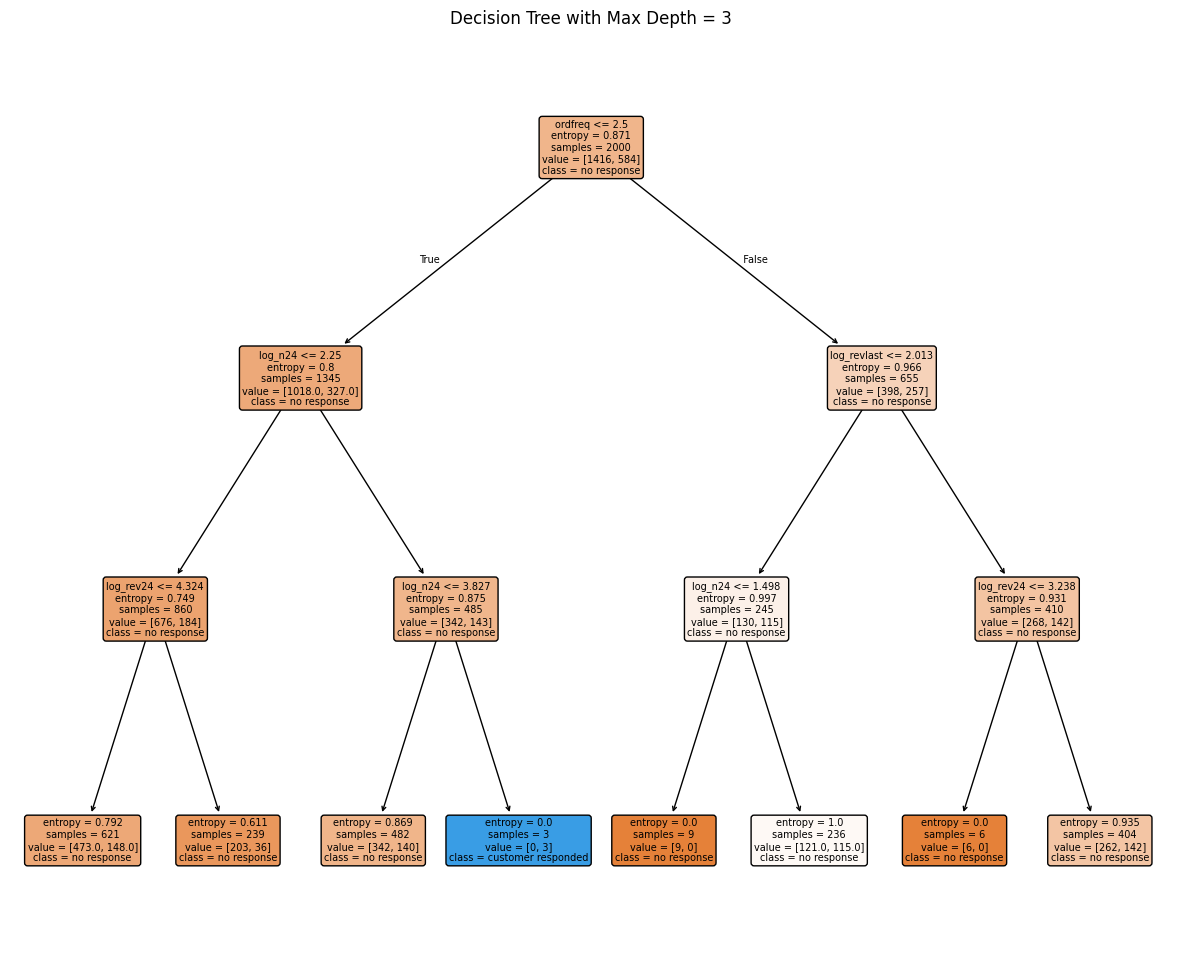


Figure 5: Decision tree visualization of max depth 3 using the training dataset

**What is the accuracy associated with this tree depth?** 70.40%

* 1. **If you had to select the three best values of tree-depth, what would they be?**
* Max depths 1 – 3 show consistent accuracy levels across the 10 folds with the narrowest IQR scores
* Highest accuracy levels across the 26 depths

A graph with black and orange squares

Description automatically generated

Figure 6: Box plots of accuracy for top 6 depths using 10-fold cross validation on the training dataset

# Developing a random forest classifier with 100 trees and comparing performance

Table 2: Random Forest classifier performance using max depths 1, 2, 3 using the training dataset

|  |  |
| --- | --- |
| Max Depth | Accuracy |
| 3 | 71.15% |
| 1 | 70.80% |
| 2 | 70.80% |
|  |  |

* 1. **Which tree-depth results in the best random forest classifier?** Max depth 3
  2. **How does it perform relative to the best decision tree?**
* Random forest using 100 trees with max depth of 3 outperforms the best decision tree (max depth of 3) in terms of accuracy (71.15% vs 70.40%).
* Decision trees tend to overfit when allowed to grow deeper and therefore will generalize poorly to new, unseen data.
* Random forests, which are ensembles of decision trees, take the average of the predictions of individual decision trees in the ensemble. This allows individual trees to be grown deeper and capture finer details in the dataset but not overfit due to the averaging effect.
* In further analysis, using 10-fold cross validation to compare, random forest using 100 trees with max depth of 3 also outperforms the best decision tree:
* Higher mean cross-validation accuracy on training data
* Lower IQR % showing more consistency across 10 folds

A graph with black squares and red lines

Description automatically generated with medium confidence*Figure 7: Box plots comparing best random forest tree vs. decision tree using the training dataset and 10-fold cross-validation*

# Developing a random forest classifier with 50 trees and comparing performance

Table 3: Random Forest classifier performance using max depths 1,2,3 on the training dataset

|  |  |
| --- | --- |
| Max Depth | Accuracy |
| 3 | 71.45% |
| 1 | 70.80% |
| 2 | 70.75% |

1. **Does your recommendation change?** No

A graph with squares and text

Description automatically generated with medium confidence

Figure 8: Comparison of performance of Random Forest classifier using 50 and 100 trees compared to best decision tree of the training dataset

* Using 10-fold cross-validation, random forest classifier using 50 trees shows the highest mean accuracy.

# Evaluating k-nearest neighbor models using 10-fold cross validation to identify the optimal value of k

* 1. **Provide all relevant results.**

Table 4: K-Nearest Neighbors classifier performance using k values of 10,11,12 on training dataset

|  |  |
| --- | --- |
| Number of Neighbors k | Accuracy |
| 12 | 69.95% |
| 10 | 69.90% |
| 11 | 69.55% |

* 1. **What value of k do you recommend?**

k = 12 because:

* KNN model with k = 12 has the highest mean accuracy and highest median accuracy.
* IQR is relatively small, indicating consistent performance across all 10 folds.
* Models with higher k values are smoother.

A graph of a number of neighbors

Description automatically generated

Figure 9: Box plot of accuracy using 10-fold cross-validation for K-Nearest Neighbors on the training dataset

* 1. **What is the accuracy associated with this value of k?** 69.95%
  2. **If you had to select the three best values of k, what would they be?**
* k = 10: Second highest mean accuracy
* k = 11: Third highest mean accuracy. Lower k values may be able to capture finer details in the imbalanced dataset. An odd value of k is often preferable to avoid ties in voting.

# Developing a logistic regression model and evaluating its accuracy using 10-fold cross validation

**a. What is the associated accuracy?** 71.15%

# Developing a logistic regression model on the training dataset

**a. What is the associated accuracy?** 71.45%

A screenshot of a computer

Description automatically generated

Figure 10: Output of the logistic regression model on the training dataset

A screenshot of a computer

Description automatically generated

Figure 11: Logistic Regression model developed using the entire training dataset

# Evaluating the four best models using 10-fold cross validation to determine the recommended model

A close-up of a screen

Description automatically generated

Figure 12: Overall accuracy comparison of the four models in each category using 10-fold cross validation accuracy

A screen shot of a graph

Description automatically generated

Figure 13: Box plot using 10-fold cross validation to visualize accuracy and performance of the top four models from each category

* 1. **Across the four models, which would you recommend and why?**
* Logistics Regression model using stratified 10-fold cross validation
* Highest accuracy using the training dataset
* Logistics regression model is ideal for predicting classes involving binary values (0 or 1)
* Easy to interpret due to the linearity in coefficients because the log-odds (logarithm of the odds of the outcome) are a linear function of the input features.
* Simple to implement with a few tuning parameters and greatest computation efficiency

# Developing the final version of the recommended model using the dataset & evaluating its accuracy

* 1. **Provide all details of the model (and the tree if the recommended model is the decision tree).**
* Model is using logistics regression using stratified 10-fold cross validation
  1. **What is that accuracy of this model on the training dataset?** 71.15%

# Making predictions on customer response using the final model and threshold for classification

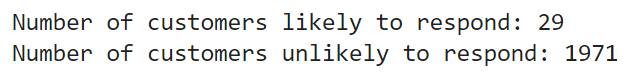


Figure 13: Number of predictions for each class in the test set

Note: Please refer to attached csv file for full predictions.

# Analyzing model predictions for lapsing customers and comparing results with other customer segments

1. **If you were to focus on the “lapsing customers” (customers who made their last purchase 13 to 24 months ago), do you expect your model to be different?**

Table 4: Accuracy for lapsing and non-lapsing customers

|  |  |
| --- | --- |
| Type of Customer | Accuracy |
| Lapsing | 72.41% |
| Non-Lapsing | 66.86% |

1. **For the selected model, compare the quality of predictions for these customers relative to predictions for the others on records in the training set. Discuss your findings.**

* Accuracy of the logistic regression model for the entire dataset (lapsing and non-lapsing customers combined) is 71.15%.
* Accuracies of the logistic regression model for the lapsing customers and non-lapsing customers are 72.41% and 66.86%, respectively.
* Accuracy improves for lapsing customers group and declines for non-lapsing customers group (suggests that 'elpsdm' is in fact an important predictor)
* **Possible reason:** Lapsing customers may exhibit more distinct behaviors for the model to classify while non-lapsing customers may not have stabilized yet, which makes it harde r to the model to accurately predict the response.
* **Potential issue:** Since the training dataset has nearly 5 times more lapsing customers than non-lapsing customers, the model may not generalize well with non-lapsing customers.
* A recommendation is to further segment the customers based on this information.
* A larger dataset with more balanced split for lapsing and non-lapsing customers should be used to develop a more robust model that can better capture the behaviors of the non-lapsing customers.