Hybrid Classification Algorithm for Customer Churn Prediction

CO421: Data Warehousing and Data Mining

About the Paper

Topic:

A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision trees

Authors:

Arno De Caigny, Kristof Coussement, Koen W. De Bock

Link:

https://www.sciencedirect.com/science/article/pii/S0377221718301243

Problem Statement

- Predict the number of customers leaving
- Estimation of a future churn probability for every customer based on the historical knowledge of the customer

Dataset Description

- The dataset chosen is from: Telecom Customer Churn
- Each row represents a customer, each column contains customer's attributes described on the column Metadata.

Raw Data

- 7043 rows (customers)
- 21 columns (features)

Fields	**Fields**	**Fields**
customerID	MultipleLines	StreamingMovies
Gender	InternetService	Contract
SeniorCitizen	OnlineSecurity	PaperlessBilling
Partner	OnlineBackup	PaymentMethod
Dependents	DeviceProtection	MonthlyCharges
Tenure	TechSupport	TotalCharges
PhoneService	StreamingTV	**Churn**

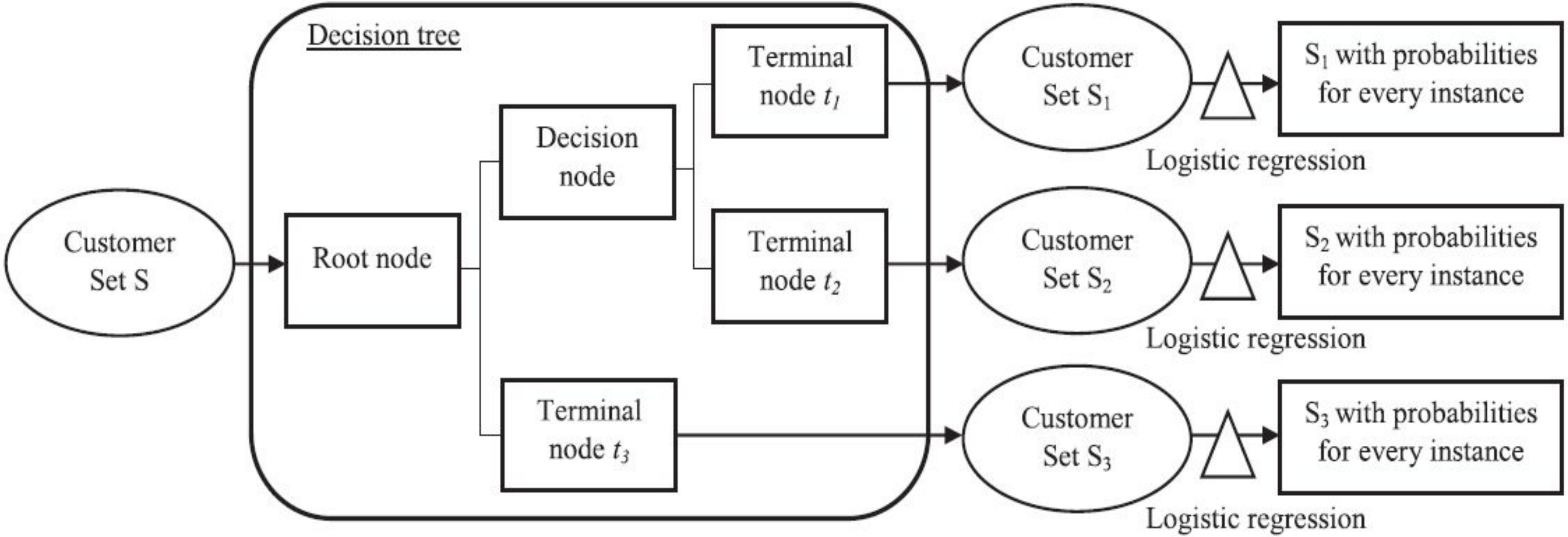
Existing Solutions

- Decision tree
- Logistic regression
- Random forest
- Logistic model trees

Proposed Approach (LLM)

- Logit leaf model
- A combination of decision trees and logistic regression
- The customer set S is split by the tree structure into customer subsets
- In the LLM algorithm, in a second step, a logistic regression with forward variable selection is fitted at every terminal node

Proposed Model Structure

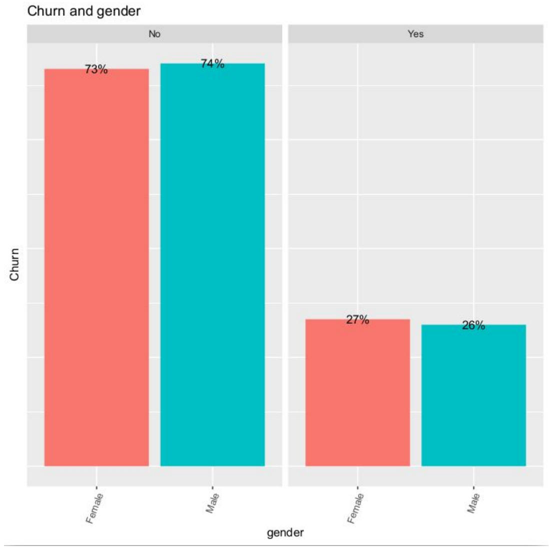


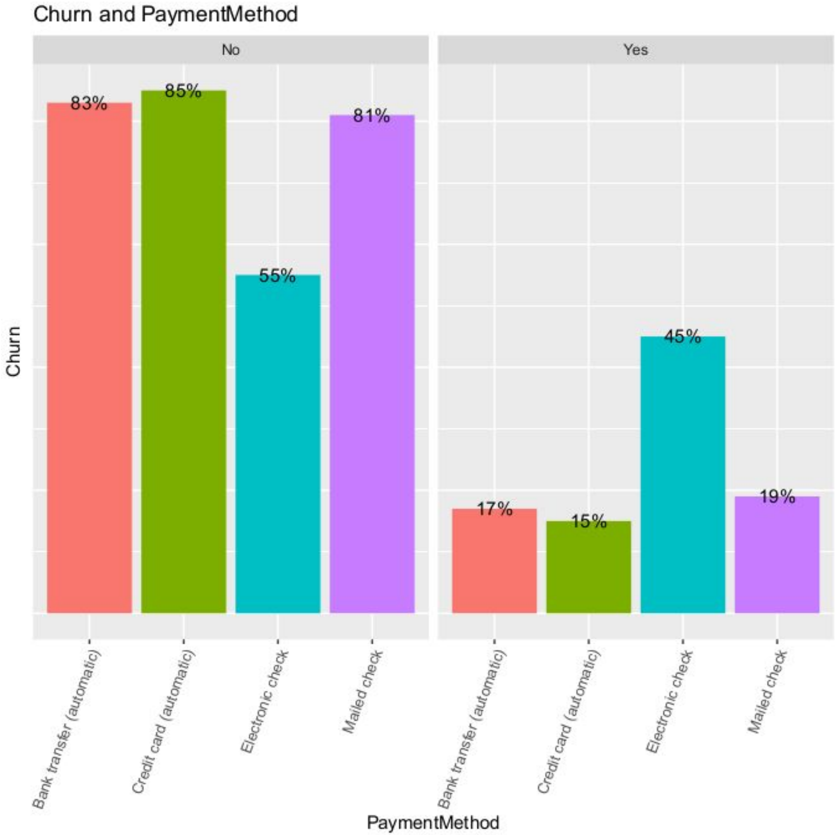
Progress

- Programming environment - R
- Dataset
- Data analysis by plots
- Data preprocessing
- Training
- Cross-validation
- Prediction
- Improvement

Data Analysis Plots

- Programming environment - R
- Dataset
- Data analysis by plots





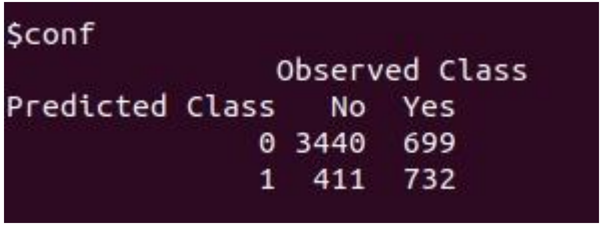
Data Pre Processing

Data cleaning and preparation

- Checking for “null” values.
- Removing the columns we won’t analyze.
  - eg: customer ID, gender, phone service, multiple lines, monthly charges, total charges
- Converted output labels from textual to numerical binary values

Model Implementation

- The dataset is shuffled to remove bias
- The dataset is split into training and test
- 7043 rows - 5282 for training and 1761 for testing
- Cross validation is performed on the training dataset



```
$conf
               Observed Class
Predicted Class    No  Yes
              0 3440  699
              1  411  732
```

Confusion Matrix

Cross Validation

The following csv files are generated after cross validation

- foldpred: a data frame with, per fold, predicted class membership probabilities for the left-out observations
- pred: a data frame with predicted class membership probabilities.
- foldclass: a data frame with, per fold, predicted classes for the left-out observations.
- class: a data frame with the predicted classes.
- conf: the confusion matrix which compares the real versus the predicted class memberships based on the class object

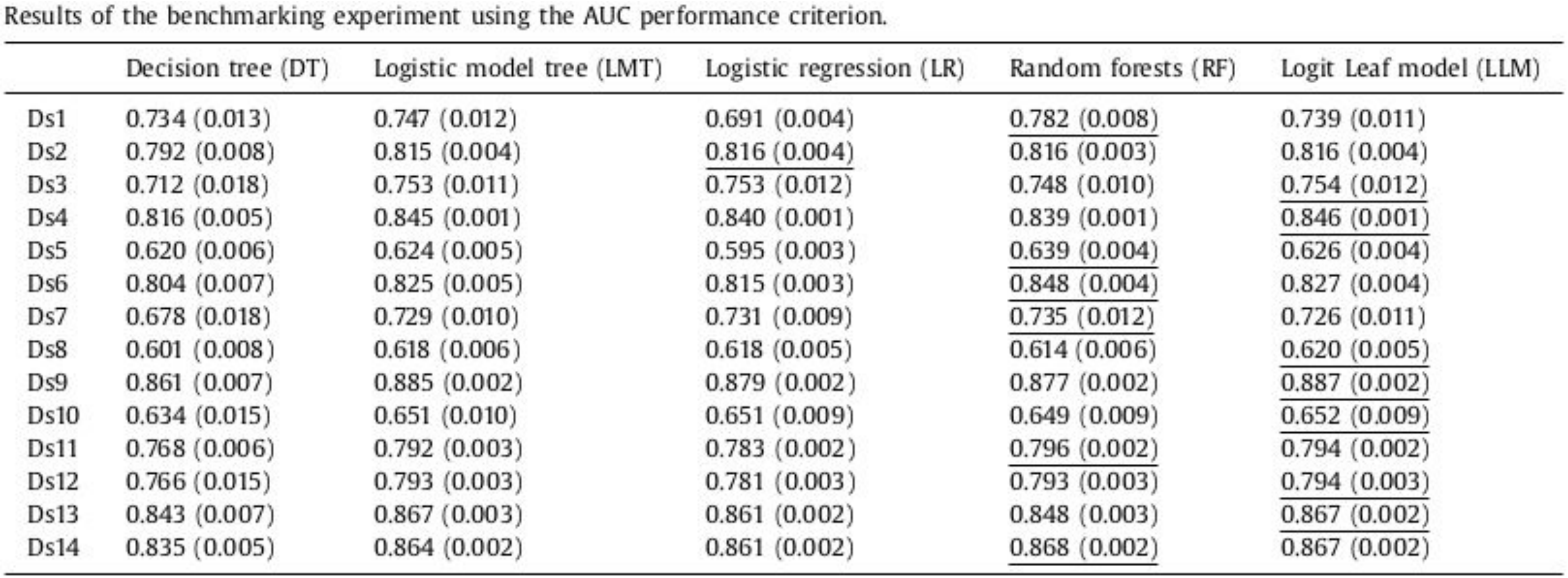
Churn Prediction

- After cross validation, the LLM model is used to make predictions for the churn probability
- The performance of the model is measured by using AUC (Area under the Curve, as is done in the paper)

AUC-ROC Curve

- AUC - ROC curve is a performance measurement for classification problem at various thresholds settings
- Higher the AUC, better the model is at predicting 0s as 0s and 1s as 1s.
- AUC using Concordance and Tied Percent :
  - Divide the data into two datasets. One dataset contains observations having actual value of dependent variable with value 1 (i.e. event) and corresponding predicted probability values(x). And the other dataset contains observations having actual value of dependent variable 0 (non-event) against their predicted probability scores(y). Compare each predicted value in first dataset with each predicted value in second dataset.
  - Total Number of pairs to compare = x * y
  - Percent Concordant = 100*[(Number of concordant pairs)/Total number of pairs]
    Percent Tied = 100*[(Number of tied pairs)/Total number of pairs]
  - Area under curve (AUC) = (Percent Concordant + 0.5 * Percent Tied)/100

Results in the paper

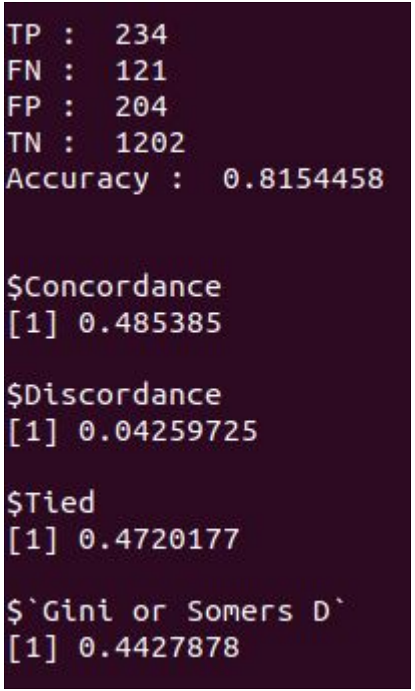


Results of the benchmarking experiment using the AUC performance criterion.

	Decision tree (DT)	Logistic model tree (LMT)	Logistic regression (LR)	Random forests (RF)	Logit Leaf model (LLM)
Ds1	0.734 (0.013)	0.747 (0.012)	0.691 (0.004)	0.782 (0.008)	0.739 (0.011)
Ds2	0.792 (0.008)	0.815 (0.004)	0.816 (0.004)	0.816 (0.003)	0.816 (0.004)
Ds3	0.712 (0.018)	0.753 (0.011)	0.753 (0.012)	0.748 (0.010)	0.754 (0.012)
Ds4	0.816 (0.005)	0.845 (0.001)	0.840 (0.001)	0.839 (0.001)	0.846 (0.001)
Ds5	0.620 (0.006)	0.624 (0.005)	0.595 (0.003)	0.639 (0.004)	0.626 (0.004)
Ds6	0.804 (0.007)	0.825 (0.005)	0.815 (0.003)	0.848 (0.004)	0.827 (0.004)
Ds7	0.678 (0.018)	0.729 (0.010)	0.731 (0.009)	0.735 (0.012)	0.726 (0.011)
Ds8	0.601 (0.008)	0.618 (0.006)	0.618 (0.005)	0.614 (0.006)	0.620 (0.005)
Ds9	0.861 (0.007)	0.885 (0.002)	0.879 (0.002)	0.877 (0.002)	0.887 (0.002)
Ds10	0.634 (0.015)	0.651 (0.010)	0.651 (0.009)	0.649 (0.009)	0.652 (0.009)
Ds11	0.768 (0.006)	0.792 (0.003)	0.783 (0.002)	0.796 (0.002)	0.794 (0.002)
Ds12	0.766 (0.015)	0.793 (0.003)	0.781 (0.003)	0.793 (0.003)	0.794 (0.003)
Ds13	0.843 (0.007)	0.867 (0.003)	0.861 (0.002)	0.848 (0.003)	0.867 (0.002)
Ds14	0.835 (0.005)	0.864 (0.002)	0.861 (0.002)	0.868 (0.002)	0.867 (0.002)

Results

Accuracy: 81.54%
AUC: 0.721



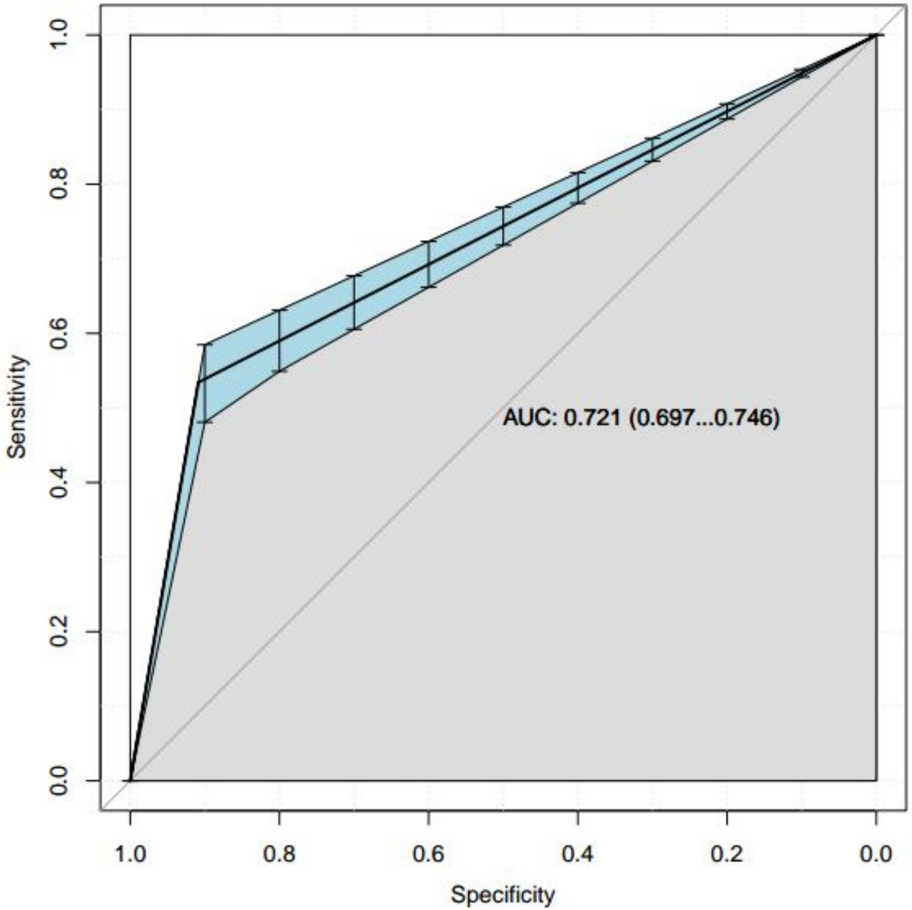
```
TP :  234
FN :  121
FP :  204
TN :  1202
Accuracy :  0.8154458


$Concordance
[1] 0.485385

$Discordance
[1] 0.04259725

$Tied
[1] 0.4720177

$`Gini or Somers D`
[1] 0.4427878
```



Improvement

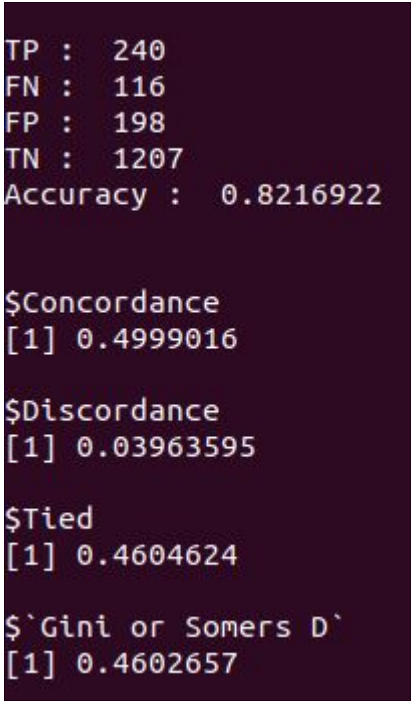
Using different models in the leaves such as Random Forest, we can potentially improve the predictive performance further.

By using RF instead of Logistic Regression in the leaves, we obtained the following results:

- Accuracy: 82.17%
- AUC: 0.73

Improvement

Accuracy: 82.17%
AUC: 0.73



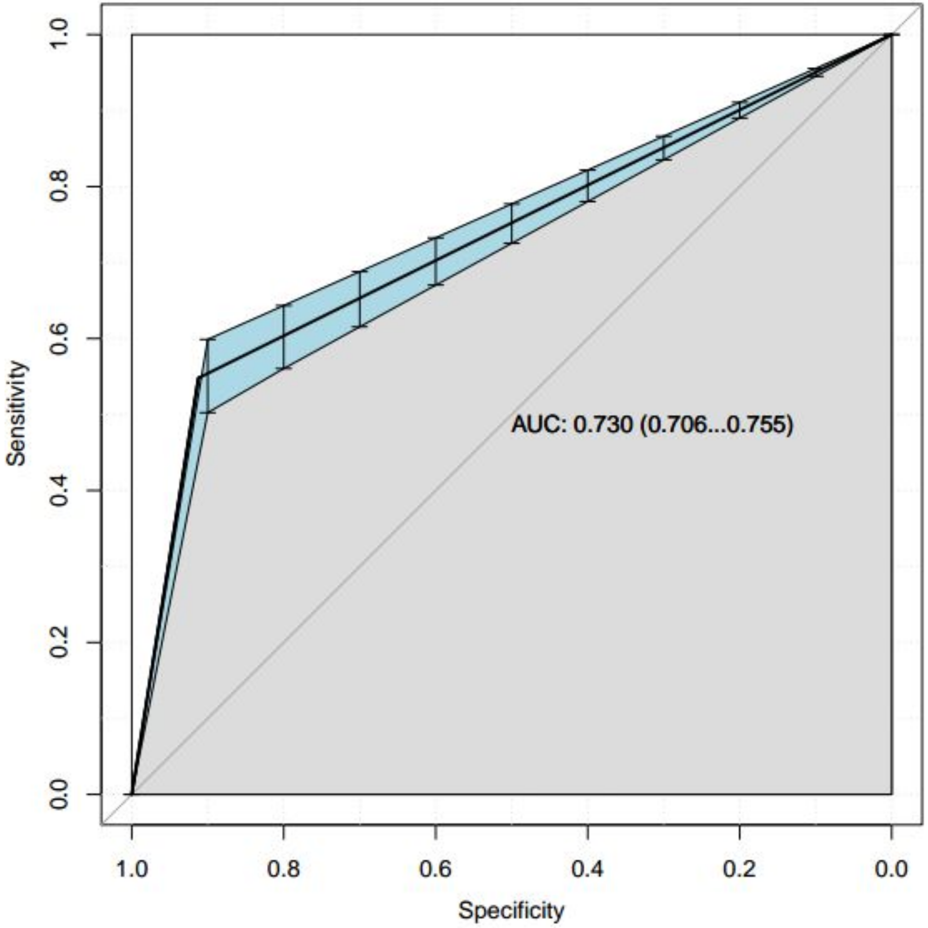
```
TP :  240
FN :  116
FP :  198
TN :  1207
Accuracy :  0.8216922


$Concordance
[1] 0.4999016

$Discordance
[1] 0.03963595

$Tied
[1] 0.4604624

$`Gini or Somers D`
[1] 0.4602657
```



Thank You!

Mehnaz Yunus - 16CO124

Mishal Shah - 16CO125

Sharanya Kamath - 16CO140