Analysis

Introduction

This analysis was conducted on customer data from a telecommunication company called *BitTel* to gain insight in customer churn. The initial dataset consisted of 3,333 rows of customer information with associated features:

State
Account length
Area Code
Phone Number
Intl Plan
Voicemail Plan
Voicemail Messages
Day Minutes
Day Calls
Day Charge
Eve Minutes
Eve Calls
Eve Charge
Night Calls
Night Charge
Intl Minutes
Intl Calls
Intl Charge
CustServ Calls
Churn

Program Flow

Using KNIME, the data was read in and analyzed using a decision tree model which provided metrics we discuss below. For the baseline model, we removed features including state, account length, area code, phone, intl plan, day calls, eve calls, and night calls. A statics and histogram node were used to evaluate overall dataset statistics. We then used a string manipulator node to change the churn feature into a string variable type that can be used by the decision tree nodes. The data was then partitioned into 80% for training and 20% for testing.

At this point we branched off to two separate flow paths; the bottom included training and testing for the baseline model, and the top flow path for testing with optimization. The top (optimized) path included a linear correlation node and filter to find and filter out correlated factors prior to the decision tree learning. The bottom path went directly into the learner and predictor nodes.

Both flow paths utilized a tree learner node (where the model and tree were built) and the predictor node (where the model was tested with known outcomes); a decision tree to ruleset and tree to image

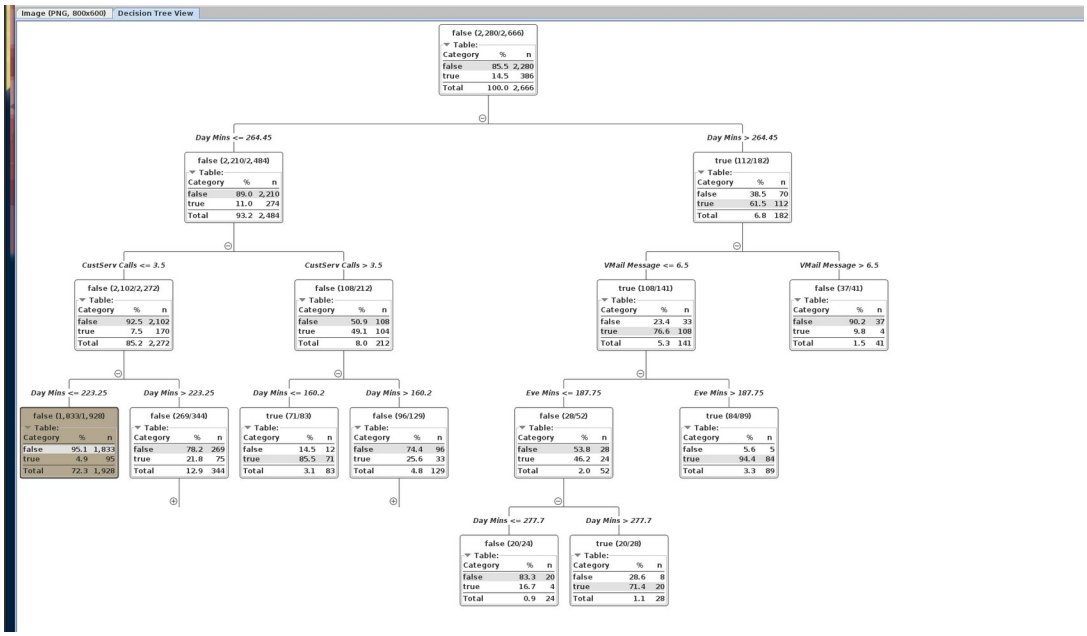
nodes to evaluate the rules used to build the tree and the tree image (which can be seen in the provided screenshot); and scorer node with views and CSV writer to evaluate the model performance.

Observation of Specific Ruleset

Row 1 of optimized decision tree to ruleset:

$Day Mins$ <= 223.25 AND $CustServ Calls$ <= 3.5 AND $Day Mins$ <= 264.45 => "false"

This patch flows down the left side of the tree with specified criteria above and ends at the selected (brown) leaf.



Model Testing and Analysis

The following tests were used and compared to the original model used (bottom flow in KNIME). The metrics can be seen in the scorer_metrics_optimized.csv

1. Baseline (bottom flow) with correlation filter; removed highly correlated features: Day Charge, Eve Charge, Night Charge, Int Charge → one more false negative on optimized, otherwise the same as the baseline model.
2. Changed quality measure to **Gain ratio** → *improved false negatives of no churn*
3. Quality measure Gain ratio; changed **pruning to MDL** → improved no churn metrics, *decreased churn metrics.*
4. Quality measure – Gain ratio; pruning – MDL; changed **min number splits to 20** → *improved churn metrics, decreased no churn metrics*.

5. Quality measure – Gain ratio; pruning – MDL; min number splits – 20; **changed binary nominal splits** (10) → *no change.*
6. Quality measure – Gini index; pruning – none; min number splits – 20; binary nominal splits - none →**best model!**

In order to evaluate the best model to use for prediction and insight to customer churn, I decided to create a data sheet that appended the changes noted below and compared them to the base model and subsequent changes to the optimized decision tree.

Using the linear correlation node we found perfect correlation value of 1 to the features: day mins | day charge; eve mins | eve charge; night mins | night charge; intl mins | int charge. This was presumably from the charges being directly correlated to a fixed rate for each. Those correlated features were removed and showed almost no negative impact on the model performance; the only decrease was one more false positive of the no churn metric.

Besides feature selection, model testing was evaluated changing the gain measure, pruning vs. no pruning, changing the minimum number of necessary positive results to 20, and binary nominal splits of 10. The best model out of these test included Gini index quality measure, no pruning and increasing the minimum number split requirement to 20. I would assume that the pruning was taking away details which predicted customer churn, and initial low minimum number required for splits was making the model too complex.

Overall Model Performance

The model was considerably better at predicting what customers will stay over what customer will turn. I would assume this is based off of class imbalance of no churn vs. churn data.

Overall, the model performed moderately well with a Cohen’s kappa score of 0.559 and accuracy of 0.91.

The metrics for customers that did not churn were excellent; 98.4% of all customers that did not churn were predicted correctly (recall), of all the predictions that customers did not churn, 91.7% were correct (precision), and F-measure of 0.949.

The metrics predicting customer churn were not as impressive. Recall of 47.4% (low), precision of 83.6% (moderate), and F-measure of 0.605.

With these metrics we can be confident at the model predicting who will stay. Insight can still be gained from the churn feature. Although it may not have very good predictive power on showing who is leaving, through the higher precision score, we can look to see what factors may be causing customers to churn.

Stakeholder Insight

Looking at the decision tree rule set we can sort the table based on associated record accounts that that specific group of criteria predicted a customer staying (false) or leaving (true). The top staying vs. leaving criteria is as follows:

Customers staying

```
$Day Mins$ <= 223.25 AND $CustServ Calls$ <= 3.5 AND $Day Mins$ <= 264.45 => "false"
```

- Day Mins <= 223.125
- CustServ Calls <= 3.5

Customers leaving

```
-$Night Mins$ > 186.2 AND $Eve Mins$ > 204.89999999999998 AND $Day Mins$ > 245.1 AND $Intl Mins$ <= 13.6 AND $Eve Mins$ <= 265.15 AND $Day Mins$ > 223.25 AND $CustServ Calls$ <= 3.5 AND $Day Mins$ <= 264.45 => "true"
```

- Night Mins > 186.2
- Eve Mins → 204.9 < Eve Mins <= 265.15
- Day Mins → 223.25 < Day Mins < 245.1
- Intl Mins < 13.6
- CustSev Calls <= 3.5

With this information we can conclude that most customers that stay have average call time during the day, and do not call customer service often. Conversely, customers that leave appear to have multiple factors and use the service more during the day, evening and night, make international calls and do not call customer service often. From this we can identify these customers before they leave and guarantee the product they are receiving is meeting their needs and if there are any problems. We can also look into whether there drop in effective services in the evening or night time, or if long distance calls are not being provided well.

Further thoughts:

This decision tree model did not give *excellent* metrics to predict customer churn. Using different machine learning models might produce better results and give better insights. Also, inspection of the subsequent rule sets may give further information to find correlations in customers staying and leaving.