James Taylor

Assignment 4

University of Maryland University College

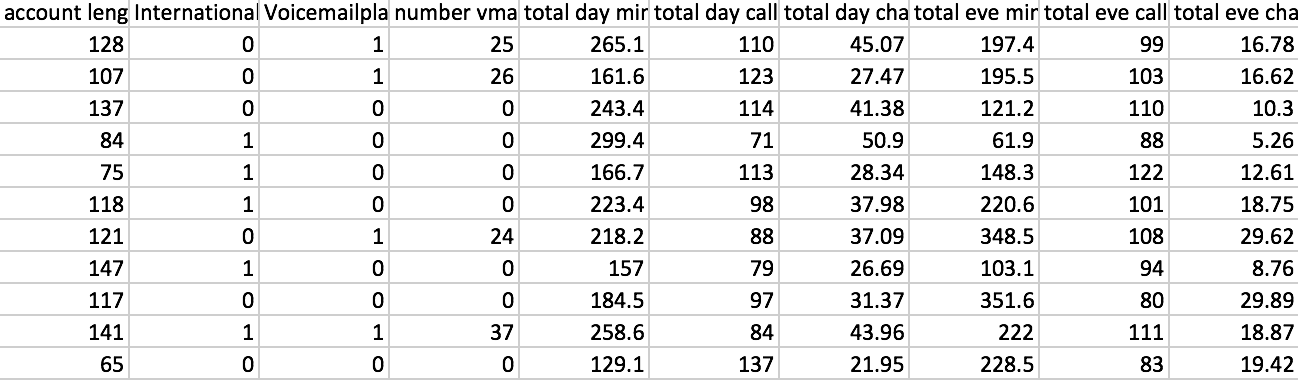
DATA640

Dr. Knode

james.taylor.ps3@gmail.com

**Introduction**

The purpose of this paper is to experiment with the effectiveness of the “Ensemble Node” in Enterprise Miner. This will allow for ensemble techniques to use more complex models to make predictions. The goal is to have the ensemble model perform better than the individual models built.

The data chosen for this paper was on churn from a phone company. The data is on the details of the account. Some of these variables are the number of calls, voicemails, minutes used, evening and night calls. The target is to predict which accounts will not renew or cancel their subscription. Below are some variables and observations in the dataset. There were 3,333 observation on 21 variables. Of all the observations in the set, 14.4% of them were customers who did not renew. 

**Data Preparation**

There were no missing values in the dataset. Three variables were binary but did not use numerical indicators. These were international plans, voicemail plans and customer churn. These variables were transformed to be a numerical one value if true. This was done so that models that require numerical values can be used. The data was partitioned so that 80% is used to train the model and the remaining 20% is tested on the model.

The state, area code and phone number were removed. The area code and phone numbers were not to be treated as numerical value because the value of the number is meaningless, besides categorizing the customer’s area. The phone number is more of an ID randomly given to the customer. The state variable should not be very relevant to the prediction and instead my over-fit the model.

For the Neural network the input variables were standardized. This was done so the model does not treat variables as more significant than others.

**Predictive Models Developed**

The target variable was to classify if an observation would not renew their subscription, or churn. Therefore, only models meant to classify were used. These were a neural network, logistic regression and gradient boosting.

These models represent different approaches in supervised learning. Neural networks rely on nodes and weights. These weights are adjusted by backpropagation until a threshold is satisfied. This optimization is what drives the model’s accuracy. It will use a sigmoidal activation function to get the predictions between zero and one.

Logistic regression fits a logistical equation, or line, to binary data. It creates on that best fits the data by relying on math. The end result is an equation that assigns coefficients to variables that sum to a probability. That probability is what the model outputs as an observations likelihood for being a zero or one.

Gradient descent use decision trees as the predicting mechanism. It is an ensemble model of their own. Gradient descent uses the residuals from a previous tree to build another model, learning from the errors of the old model to make better predictions. The most common prediction from all the decision trees is the final prediction for each observation.

The purpose of this paper is to use all of the above models as individual models in an ensemble model. Each of the model’s prediction on unseen observations will be taken into account and a final output will be drawn from them. Ideally, the different model’s techniques allowed some models to fit the data better in some circumstances. Using all the models in an ensemble technique allows all the benefits of the models to be used to make a final prediction.

A random forest model was included as another point of comparison. This is another ensemble model of its own randomizing the features it includes in its own models.

**Results**

The results are displayed in *Figure 1* in the appendix. All the models individually struggled with sensitivity. This is likely due in part to the imbalanced target variable. It also may be the case that customers that churn are sometimes similar to customers that don’t. Imagine a distribution for churn customers. The distribution is similarity to customers that do not churn. The models have an easier time distinguishing churn customers that standout from the non-churn (right-hand side of the distribution). The models struggle on distinguishing churn customers that look similar to non-churn (left-hand side of the distribution). If the distribution is skewed to the right, or has lots of observations on the left-hand side, sensitivity will struggle.

All models performed highly in specificity, which were all above .984. They varied a lot in sensitivity however. The neural network had the best performance overall, with a sensitivity of .745 and 73 true positive predictions. It had nearly a 95% accuracy on unseen data. The next best performer was random forest, which had .633 sensitivity and accuracy of 94%. The remaining models did not perform as good, with the gradient boosting model detecting less than half of the true churn customers as such and the logistic did so about 21%.

The ensemble node used the neural net, logit and gradient boosting as the individual models. There were three functions that combined the outputs of the models.

The average ensemble model performed very well on the specificity of unseen data, achieving a .998 level. That is better than all the individual models. The sensitivity of the average ensemble model was nearly the average of the individual models, with the average being .463 versus the ensemble’s sensitivity of .48 level. The overall accuracy is close to the average of the three models as well. The ensemble model did not prove itself to be a better predictor generally than the individual models.

The vote ensemble model performed similarly to the average ensemble. One of the benefits of having an uneven number of models in an ensemble technique is not needing to handle situations with a tie in the votes.

The maximum ensemble uses the predication with the largest probability. It did achieve the highest sensitivity of all the models and ensembles. Its overall accuracy also rivaled the neural net.

**Conclusions and Takeaways**

The ensemble models did not perform significantly better than the models it used. The maximum ensemble technique did achieve the highest sensitivity of all models and techniques. This measure is important for the purpose of the model because we want to detect who will churn and decisions might be made about customers using it.

The downside of the max ensemble however was the specificity. It made the most false-positive predictions of all the models/ensembles. Compare that to the average ensemble model. One way the average ensemble was better was the low number of false-positives. For the uses of this model on this data, having a false positive may be a costly mistake because a reduced rate may be extended to this person to incentivize them to stay. However, if this person is a false positive, the company is giving a reduced rate to someone who would have paid the full amount anyway, which isn’t desirable from a revenue perspective.

The user of these perspective models will need to prioritize which type of error (Type I and II) is costlier. This will lead them to finding the right model for them. The ensemble models are still potential options from that perspective, but the neural network performed well enough to be used on its own.

Appendix A

Figure 1.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Model** | **FN** | **TN** | **FP** | **TP** | **Sensitivity** | **Specificity** | **Accuracy** |
| **NN** | 25 | 562 | 9 | 73 | 0.745 | 0.984 | 0.949 |
| **Logit** | 77 | 565 | 6 | 21 | 0.214 | 0.989 | 0.876 |
| **Gradient Boosting** | 56 | 569 | 2 | 42 | 0.429 | 0.996 | 0.913 |
| **Ensemble (AVG)** | 51 | 570 | 1 | 47 | 0.480 | 0.998 | 0.922 |
| **Ensemble (VOTE)** | 53 | 569 | 2 | 45 | 0.459 | 0.996 | 0.918 |
| **Ensemble (MAX)** | 22 | 556 | 15 | 76 | 0.776 | 0.974 | 0.945 |
| **Random Forest** | 36 | 567 | 4 | 62 | 0.633 | 0.993 | 0.940 |