SCHOOL OF COMPUTING, ENGINEERING & DIGITAL TECHNOLOGIES

ICA Machine learning (CIS4035-N)

Churn prediction in telecom sector using supervised linear models

Ashish Kakran

B1110946@tees.ac.uk

School of Computing, Engineering and Digital Technologies (SCEDT)
Teesside University, England, United Kingdom.

Abstract

Churn prediction is challenging problem for companies in telecom sector (A. Gaur and R. Dubey, 2018) as they try to retain customers and not lose them to rival companies. Performing customer churn analysis precisely and rolling out measures for people likely to churn is a must for any company trying to maximise its revenue. Not leveraging churn analysis techniques on the other hand could lead to loss of revenue. In this paper, some linear supervised approaches have been explored and compared to develop a churn prediction model. While it doesn't specifically add something new to the field (B. P. and N. G.S., 2017) but highlights the challenges associated with high dimensionality data in the churn analysis problem context.

1 Introduction

Churn analysis is a classification problem (Géron, 2017, p. 9) where the aim is to predict whether or not an entity involved is going to churn away or not from the concerned entity. There are several datasets available in this domain to develop approaches to precisely predict (classify) churn status.

Results obtained from such model could be utilised to derive marketing campaigns targeting customers who are likely to churn in order to retain them and thus avoid loss of revenue.

A medium size dataset (*telecom customer dataset.*) in the CSV format has been downloaded from Kaggle website. The dataset contains 100000 rows and 100 columns. A full description of these columns is available in the source. Of the 100 columns, one of them is 'churn' which contains labels '1' if customer has churned and '0' if customer has not churned.

2 Data exploration and feature selection

2.1 Exploratory data analysis

The dataset has both numerical as well as categorical features. Of the 99 features, 78 features are numerical and 21 features are categorical. There are several features with missing values. Features with more 30% of the data missing were deleted as interpolating such values would have led to bias problem during training. Here is heat map of null values after removing sparse features:

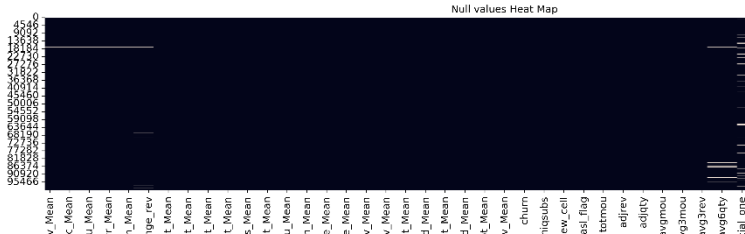

Fig 1: Heat map of features with null values

2.2 Feature selection

Most of the features are loosely correlated to the label column 'churn'. Features with high mutual correlation were dropped to avoid redundancy as many of the machine learning algorithm rely on independent and

identically distributed features for better accuracy.

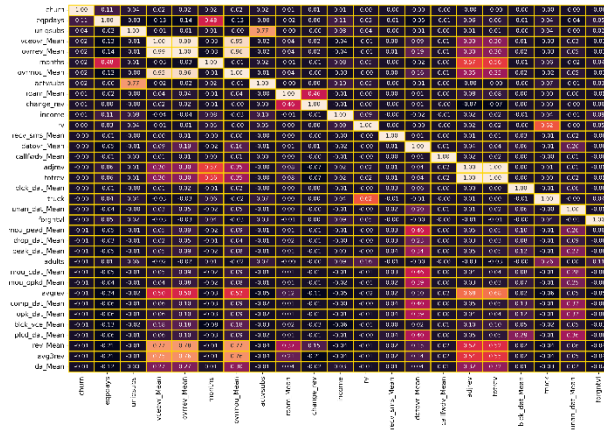

Fig 2: correlation matrix heatmap before removal of redundant features

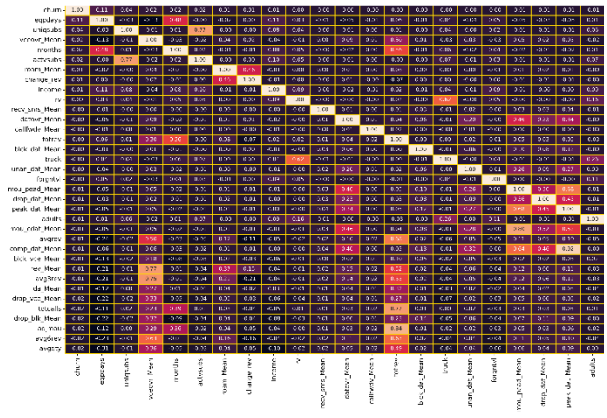

Fig 3: Correlation matrix heatmap after removal of redundant features

The dataset is highly imbalanced as concluded from reviewing distribution of various categorical features. Some of the numerical features were found to have tail-heavy distribution (Géron, 2017, p. 65), which suggests that data needs to be scaled. Separate processing pipelines (Géron, 2017, p. 40) were constructed to process categorical and numerical features.

Numerical features were first imputed using median strategy as these features are mix of int64 type and float64 type values. Using mean as strategy would have converted integral columns to floating type, which is undesirable. These imputed numerical features were standardised.

Categorical features were first imputed using "most frequent" strategy. These imputed features were one-hot encoded as machine learning algorithms rely on numerical features.

A full pipeline was developed combining separate pipelines for numerical features and categorical features. Once processed through pipeline the dataset was split into training set and test set in 80:20 proportion.

3 Experiments

In the training process, a training set of 80000 x 195 dimensions was trained against 80000 x 1 label vector using three supervised learning linear models, lasso logistic regression, ensembled random forests and support vector machine.

These models' performance was evaluated and compared to select the final model.

3.1 Evaluation metrics

Since dataset is evenly balanced when it comes to label distribution (almost 51% '1' labels and 49% '1' label) accuracy was not used as evaluation metric for the comparison. A naïve classifier which

randomly classify an instance into one of the categories could achieve reasonably high accuracy in this case but less effective on new instances. For proper evaluation of performance of these models following evaluation metrics were used:

3.1.1 Precision

Precision or specificity (Géron, 2017, p. 93) measures ratio of true positive (TP) against total positive which is sum of true positive and false positive (FP). In context of churn prediction, high precision means model is able to classify people likely to churn correctly where as low precision means model is too rigid and classify some people who are not likely to churn as positive. Thus, high precision is desirable in this case.

$$\text{precision} = \frac{TP}{TP + FP}$$

3.1.2 Recall

Recall or sensitivity (Géron, 2017, p. 94) measures true positive rate. It measures true positives against sum of true positive and false negative (FN). Instances falsely classified as negative are actually positive, thus it measures true positive rate. In context of churn prediction, high recall means model is able to positively classify most of the people likely to churn whereas low recall means model is underperforming and doesn't classify many people who are likely to churn positively. High recall is desirable in this problem.

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

3.1.3 F1-Score

F1 score is simply harmonic mean of precision and recall (Géron, 2017, p. 95) and thus it favours precision and recall scores which are closer. If precision is high and recall is low then f1 score is low and converse is true in this case.

$$\text{f1} = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

3.2 Procedure

3.2.1 Cross validation

A 3-fold cross validation set was used to calculate evaluation metrics during the training. A confusion matrix was plotted for each model to observe absolute numbers of predictions.

3.2.2 Hyper-parameter tuning

A default implementation of models from scikit-learn (Pedregosa *et al.*, 2011) was used. Although attempts were made to find optimal values of hyperparameters (Géron, 2017, p. 32), due to lack of computation power required to process such high dimensional data these models didn't make it final list of models.

3.2.3 Dimensionality reduction

In order to deal with dimensionality in data (Géron, 2017, p. 215), principal component analysis was performed while retaining 95% of variance in data. Although

it lesser dimensions made it easy to train certain models, these models weren't desirable in problem context. Only SVM performed better after dimensionality reduction. This linear SVM model achieved high precision and better recall than any other model.

3.2.4 Models

Three models, namely, Lasso logistic regression, Random Forest classifier and linear Support Vector Machine (SVM) were trained in two step process. First these models were trained without reducing dimensions in the training data. After observing long training time, principal component analysis was used for dimensionality reduction and models were trained again on this reduced training set. Model with highest f1 score and high precision and high recall was selected.

3.2.5 Test set prediction

Random forest turned out to be model with best precision-recall trade-off and it was used to predict labels from test set.

4 Results

Here is summary of evaluation metrics of different models:

Table 1: Evaluation metric results

Classifier	Precision	Recall	F1 score
Logistic regression	0.58	0.62	0.60
Random forest	0.57	0.63	0.60
Linear SVM	0.57	0.65	0.61
Logistic regression with PCA	0.57	0.58	0.58
Random forest with PCA	0.57	0.59	0.58
Linear SVM with PCA	0.57	0.66	0.61

A look at precision order graph suggests that these models perform roughly the same

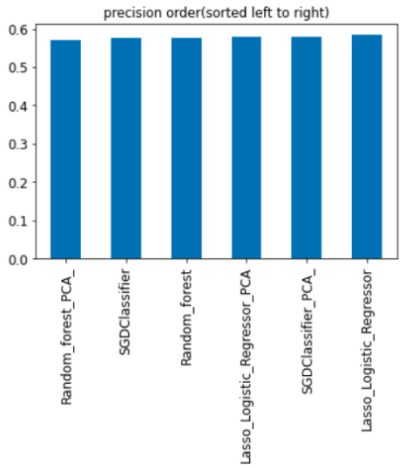

Fig 4: Precision order graph

Unlike precision graph there is significant variation in recall graph. Linear SVM with PCA is highest in recall while lasso logistic regression is lowest. A model with moderate recall is required.

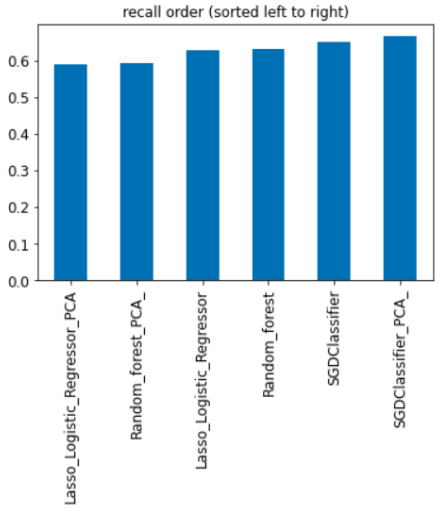

Fig 5: Recall order graph

A look at f1 score graph clearly suggest stochastic linear SVM performed best.

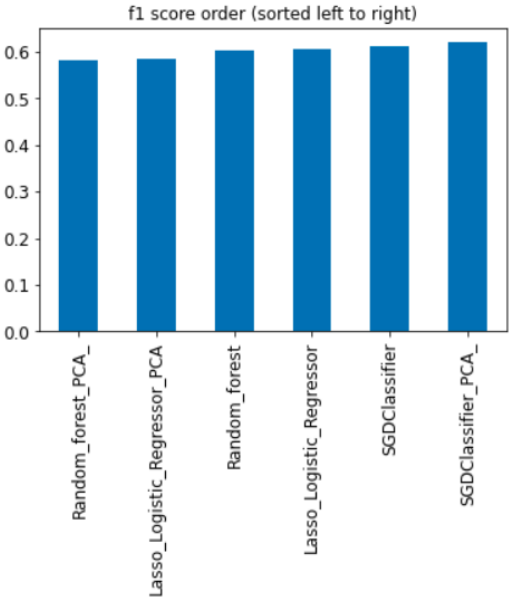

Fig 6: f1 score comparison graph

5 Discussion

As a requirement of ICA, at least three models were trained and evaluated to select final model. Review of these models is as follows:

5.1 Lasso Logistic Regression

The logistic regression (Géron, 2017, p. 44) is simplest linear model for binary classification and performs reasonably well compared to Random Forest and SVM when it comes to precision. However, the model is low in recall which means many people who are likely to churn will be classified as negative. This model is not suitable compared to other. After reducing dimension recall reduces further.

5.2 Random Forest Classifier

Random forest classifier is an ensemble learner (Géron, 2017, p. 199) combining configurable number of decision tree classifiers. While this model has higher precision than logistic regression, it is slower to train and doesn't compete well against support vector machine. Also, after reducing dimensions Random Forest classifier has lowest recall.

5.3 Linear Support Vector Machine

Support vector machine is capable of both linear and nonlinear classification (Géron, 2017, p. 155). Here stochastic gradient descent approach was used as training linear SVM took long time. After reducing dimension this model achieved high precision and highest recall.

6 Conclusion and future work

This exploratory study is fairly limited. High dimensionality of dataset poses problems to achieve high precision via

linear models. More domain expertise could be leveraged to reduce number of features and/or features with higher correlation could be collected as evident in data exploration stage. Besides data is imbalanced across certain categories which makes it harder to capture population distribution across training and testing sample.

In future, advanced data mining techniques could be used to balance the dataset and non-linear models like neural networks could be trained and compared to improve upon linear models.

References

A. Gaur and R. Dubey (2018) 'Predicting Customer Churn Prediction In Telecom Sector Using Various Machine Learning Techniques', - *2018 International Conference on Advanced Computation and Telecommunication (ICACAT)*. doi: 10.1109/ICACAT.2018.8933783.

B. P. and N. G.S. (2017) 'A Review on Churn Prediction Modeling in Telecom Environment', - *2017 2nd International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*. doi: 10.1109/CSITSS.2017.8447617.

Géron, A. (2017) *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. Sebastopol: O'Reilly Media, Incorporated.

Pedregosa, F. *et al.* (2011) 'Scikit-learn: Machine Learning in Python', *Journal of Machine Learning Research,* 12, pp. 2825-2830.

telecom customer dataset. Available at: https://www.kaggle.com/datasets/abhinav89/telecom-customer .