Predicting Red Hat Business Value

Gautam Verma CS, NCSU *gverma@ncsu.edu*

Udit Deshmukh CS, NCSU *udeshmu@ncsu.edu*

Harshal Gala CS, NCSU *hgala2@ncsu.edu*

1 Background

1.1 Introduction

Customer Relationship Management (CRM) focuses on finding optimal techniques and practices that helps improve the association of every customer with the company by analyzing the information collected from the customers themselves [1]. Moreover, with the increasing customer-centric approach to business [2], this is steadily gaining more recognition. As per Parvatiyar and Sheth [3] and Krackleur et al. [4] CRM comprises of customer identification, attraction, retention, and development. Thus, several data mining techniques like classification, regression, clustering, forecasting, etc. [5] are applicable in this domain.

In this project, we undertake a classification type of data mining problem in which our goal is to identify those customers that have high business potential for Red Hat, Inc. The dataset has been provided by Red Hat and is available for use at Kaggle. We use R to implement the different classification models. For each type of model, we calculate the accuracy, precision, recall, and F1 scores which are presented in the Results section.

The specific refinements that we came up for this project was using Chi squared test and forward subset selection as tools for feature engineering

The special cases that we encountered in this dataset were that the variable names were masked by the organization, thus giving no clue of the physical significance of these variables and that certain categorical variables had a very large cardinality. We solved the problem of high cardinality using a technique called supervised ratio.

1.2 Dataset

The dataset comprises of three large files: people.csv, act_test.csv, and act_train.csv. The people file contains unique entries of all the people who have performed some activities in the past and the training file contains the information about all these activities. The dimensions and size of the data are as follows:

a. people.csv: (189118 rows, 41 columns); 47.1 MB.
b. act_train.csv = (2197291 rows,15 columns); 64.1 MB.
c. act_test = (498688 rows, 14 columns); 29.5 MB.

The people.csv is merged with act_train.csv and act_test.csv based on the people_ID for training and testing the different models. Thus, the total number of raw features available become 54 (excluding the label).

1.3 Related Work

We find that Ha et al. [7] and Kim et al. [7] have done some similar work to this study in which they use decision tree and self-organizing map to analyze customer behavioral patterns to identify loyal and valued customers. Also, Verfoef et al. [1] has showed how to predict customer potential in the insurance industry. The work by Reinartz et al. has also been a source of motivation for this project.

2 Methods

2.1 Data Preprocessing

2.1.1 Removing redundant features

We first removed the redundant features activity_id, people_id and date because we believe that these are have no significance as far as the classification task is concerned. We also removed the group_1 attribute since it has a very large number of possible levels.

2.1.2 Chi Square

While building linear models, it is essential to remove features which are correlated to each other. Since almost all our features are categorical, we used Chi squared test to determine the amount of correlation between each pair of features. We used the chisq.test() function in R to implement this. However, Chi squared test is sensitive to the size of the dataset. Hence, in order to normalize the correlation coefficient, we used Cramer's V coefficient. The Cramer's V coefficient is given by the formula

$$\boldsymbol{V} = \sqrt{\frac{\boldsymbol{\chi 2}}{\boldsymbol{\chi 2}_{max}}}$$

where $\boldsymbol{\chi 2}_{max} = N * (\min(N, P) - 1)$;
in which N is the number of records;
and P is the number of features.

The value of V is between 0 and 1, with 1 denoting maximum correlation. We set the threshold coefficient to be 0.7. After performing this step, the number of features were reduced to 40.

2.1.3 Forward Subset Selection

After removing linearly correlated features, the number of available features was still large (around 40). We then looked to select a subset of these features. Since an exhaustive search of all the possible subsets of these features is computationally expensive, we decided to use the forward subset selection algorithm. Forward subset selection [12] is a wrapper based approach, which searches through the feature space for an optimum subset by fitting models on these subsets. The best predictor set is determined by some quantitative measure, which in our case is the BIC score. The subset with the highest BIC score is considered to be optimum. We implemented the forward subset selection algorithm using the regsubsets() function in the 'leaps' package in R. After implementing the forward subset selection algorithm, we were left with around 26 features.

2.1.4 Supervised Ratio

There were two categorical variables in the dataset, namely char_1 and char_10 which had a very large cardinality. The variable char_1 had around 50 levels whereas the variable char_10 had around 6000 levels. If these features are used to build a model, the computing system requires a very large amount of memory. Given the hardware restrictions that we had, we could not afford to use these raw features to build any model. The approach we came up with to deal these variables is to use supervised ratio [13]. Supervised ratio transforms categorical variables into continuous variables having a range from 0 to 1. Supervised ratio is defined as:

$$SR(i) = \frac{P(i)}{P(i) + N(i)}$$

where P(i) is the number of records having value i of the given categorical variable and has outcome 1, whereas N(i) is the number of records having value i of the given categorical variable and has outcome 0. The other approach we looked at for converting these variables into a continuous range was Weight of Evidence [13]. However, while computing WOE, there were certain cases where the denominator of the expression turned out to be 0, thus making WOE undefined for these cases. In order to avoid the shortcomings of WOE, we decided to go forward with supervised ratio

2.1.5 Random sampling

The original dataset contains a very large number of records. Due to restrictions on the hardware, we perform random sampling without replacement on the dataset to construct a smaller training set. The records which are not part of the training set form the testing set. We chose a sample size of 50000 for this problem.

2.1.6 Partitioning the dataset based on activity category

There are two major categories of activities: Type 1 activity and Type 2-7 activities. The distinction between these two categories is that Type 1 activity has nine features associated with it and Type 2-7 activities have only one associated feature. The features that are not associated with an activity thus have blank values. Hence, we partition the dataset into two parts. The first part contains only Type 1 activities and the second part contains the rest of the activities. We apply classification algorithms independently on these two partitioned datasets.

2.2 Models

As a baseline, we implemented the Logistic Regression and Decision Tree models to identify the high potential customers. We then implemented two more models viz. Support Vector Machine, and Random Forests to improve the performance metrics.

2.2.1 Support Vector Machine

SVMs try to separate the classes such that the line or curve separating them has the maximum margin from all the data points. These margins are also called support vectors. SVM uses a cost parameter to reduce overfitting.

We implemented the SVM using the e1071 package in R. We used different type of Kernels to build the SVMs. They are enlisted as follow:

2.2.1.1 Polynomial

This type of kernel produces a separating curve like that of a polynomial equation.

Formula: $(\Gamma * u' * v + coef0)^{degree}$

Here, the parameters Γ, coef0, and degree can be varied to tune the performance of the model.

2.1.1.2 Linear

This type of kernel generates a straight line that separates the two classes with the maximum margin.

Formula: $u'*v$

2.1.1.3 Sigmoid

This type of kernel produces an S – shaped separating curve (generally).

Formula: $\tanh(\Gamma * u' * v + coef0)$

Only the parameter – Γ can be varied to tweak the performance of the kernel.

2.1.1.4 Radial Basis

The radial kernel is given by the formula: $e^{-\Gamma * |u-v|^2}$

In this, only gamma parameter can be adjusted to improve the performance of the model.

2.2.2 Decision Tree

The classification technique is a systematic approach to build classification models from an input data set and one of the technique is using decision tree. [6] Decision tree is a graph to represent choices and their results in the form of a tree. The nodes in the graph represent an event or choices (i.e. outcome in our scenario) and the edges represent the decision rules or conditions based on the other features. We use the rpart package in R to build the decision tree model. It builds a binary tree by splitting the data into two subgroups at each stage using an attribute that provides the best split based on splitting criteria.

2.2.3 Logistic Regression

[7] Logistic regression is a regression model where the dependent variables are categorical. Here we measure the relationship between the categorical variables and the different independent variables estimating probabilities using a logistic function which is a cumulative logistic distribution. We use the glm() function from stats package in R to fit the model using features obtained from the preprocessing step.

2.2.4 Random Forest

Random forests uses multiple decision trees to generate the classification model in this problem. It generates multiple trees that use different features to train themselves. These features are only a subset of features of the complete dataset, which can also be specified during training the complete model. Whenever an input is received, the decision from all the tree in the forest is considered and the prediction with the highest mode is classified as the potential outcome. We use the "randomForest" package in R to use this classifier.

The project flow illustrating all the pre-processing and model implementations is shown in Figure 1.

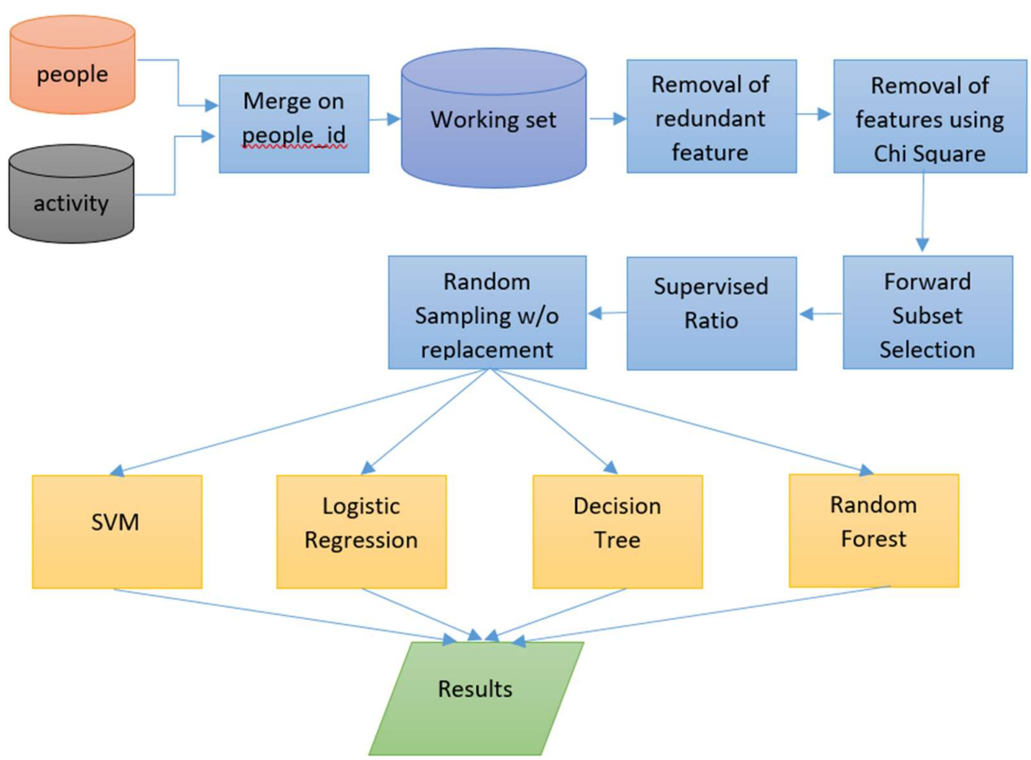

Figure 1: All the steps included in pre-processing and model implementation during the project flow. The "working set" consists of both the type_1 and type_2 to type_7 activities partitioned on the activity_id. All the pre-processing and models implemented on both the datasets are same and thus, we just call them "working set" together for illustration purpose.

3 Plan and Experiment

In this problem, we believe that the highest matric of importance for Red Hat should be Recall. It is because if customers with low business potentials are predicted incorrectly, it may result in monetary loss to the company. However, since we do not know what the high potential customers themselves mean, as no detail is given about it, other performance measures may also be applicable under different scenarios. Therefore, we use 4 performance metrics viz. accuracy, precision, recall, and f1 scores for all the models.

Impressed by the baseline results of decision trees, we hypothesize that random forests should give us even better results. The reason behind this hypothesis is that random forests inherently uses decision trees, which is observed to give better results than the LR model.

4 Results and Experiments

4.1 Support Vector Machine

We used 4 different types of SVM kernels. The following lists them and the configuration in which they were used.

4.1.1 Polynomial

The parameters used are:
cost = 1
degree = 1
Coef0 = 1000
Gamma = 1

4.1.2 Linear

The parameters used are:
cost = 100

4.1.3 Sigmoid

The parameters used are:
cost = 10
Coef0 = 0
Gamma = 100

4.1.4 Radial

The parameters used are:
cost = 100
Gamma = 1

4.2 Decision Tree

There are no tunable settings in the “rpart” package for this model. Therefore, the default settings are used to train on the refined dataset.

4.3 Logistic Regression

We test the model on the testing data set that we obtained from the preprocessing step. We set the threshold probability to 0.5, which means that if the probability of a record belonging to class 1, given its attribute values is more than 0.5, we classify that as a positive. Else, we classify it as a negative. Figure 2 shows the evaluation metrics obtained on the testing set.

4.4 Random Forest

The parameters used are:

ntree = 500

sampsize = 3000

The performance metrics for all the model configurations are shown in Figure 2. We have done 20 simulations of each model across each type of activity set partition viz. type_1 and type_2 to type_7 activities. The results shown in Figure 2 are average of all these simulations.

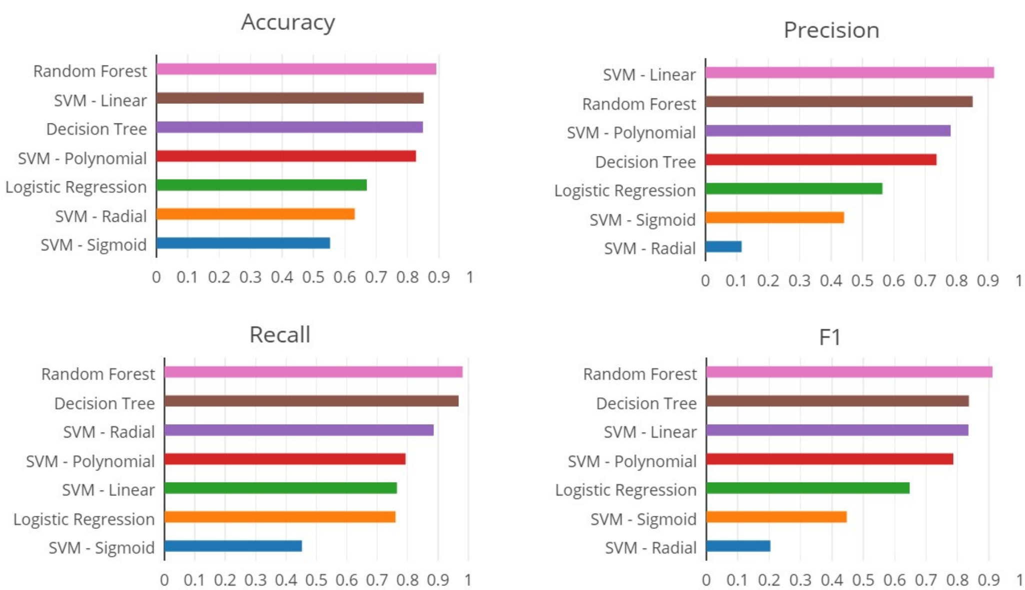

Figure 2: Average Performance metrics for all the model configurations for 20 simulations.

From Figure 2, we observe that Random Forest outperforms all in almost all of the cases, which makes the hypothesis stand correct for the type_1 activities. An interesting point of inference is that SVM – Radial has the best recall, which we assume is most important, amongst all SVMs. However, it performs poorly in other performance measurements, in which the SVM – linear is observed to perform better.

5 Conclusion

In this study, we used different classification techniques in order to predict potential customers for Red Hat, based on the data provided. We found that Random Forest gave the best accuracy and recall out of all the models. While considering potential customers for any organization, recall is an important measure since the organization should not waste their resources on false positives. Assuming this, we conclude that random forest is the best method of all the models applied. However, more than the models we implemented, the biggest takeaway from this was that we realized data preprocessing is the most critical step involved in the data mining pipeline. Around 80% of our efforts were concentrated on this step. Since the variable names were masked by the organization, there was no way to understand the physical significance of the attributes. The next challenge we faced was the high dimensionality and cardinality of the dataset and the techniques we used to tackle them gave reasonable results.

However, there were a few limitations which we could have overcome. Random sampling and feature subset selection may have led to a lot of information loss. We could have used better sampling techniques or advanced computing resources. Since we did not use k fold cross validation, chances are that the results might have been a little overfitted (for example decision trees). The forward subset selection algorithm we used may not result in the best subset of features and could further be optimized using various greedy approaches.

The code for this project can be found at the following GitHub repository: https://github.ncsu.edu/gverma/ALDA_RedHat_Predicting_Customer_Potential.git

5 References

[1] Verhoef, P. C., & Donkers, B. (2001). *Predicting customer potential value an application in the insurance industry. Decision support systems*, *32*(2), 189-199.

[2] Reinartz, W., Krafft, M., & Hoyer, W. D. (2004*). The customer relationship management process: Its measurement and impact on performance. Journal of marketing research*, *41*(3), 293-305.

[3] Parvatiyar, A., & Sheth, J. N. (2001). *Customer relationship management: Emerging practice, process, and discipline. Journal of Economic and Social research*, *3*(2), 1-34.

[4] Kracklauer, A. H., Mills, D. Q., & Seifert, D. (2004). *Customer management as the origin of collaborative customer relationship management.* In *Collaborative Customer Relationship Management* (pp. 3-6). Springer Berlin Heidelberg.

[5] Ngai, E. W., Xiu, L., & Chau, D. C. (2009). *Application of data mining techniques in customer relationship management: A literature review and classification. Expert systems with applications*, *36*(2), 2592-2602.

[6] Ha, S. H., Bae, S. M., & Park, S. C. (2002). Customer's time-variant purchase behavior and corresponding marketing strategies: an online retailer's case. *Computers & Industrial Engineering*, *43*(4), 801-820.

[7] Kim, S. Y., Jung, T. S., Suh, E. H., & Hwang, H. S. (2006). Customer segmentation and strategy development based on customer lifetime value: A case study. *Expert systems with applications*, *31*(1), 101-107.

[8] https://www.tutorialspoint.com/r/r_decision_tree.htm

[9] https://en.wikipedia.org/wiki/Logistic_regression

[10] https://en.wikipedia.org/wiki/Random_forest

[11] http://mlwiki.org/index.php/Cramer%27s_Coefficient

[12] https://www.r-bloggers.com/introduction-to-feature-selection-for-bioinformaticians-using-r-correlation-matrix-filters-pca-backward-selection/

[13] http://www.kdnuggets.com/2016/08/include-high-cardinality-attributes-predictive-model.html