Churn Prediction in Motor Vehicle Liability Insurance

Dóra Erdős, Attila Kiss,
Rita Madocsai, Jácint Szabó

Computer and Automation Research Institute, MTA SZTAKI
Lágymányosi út 11., Budapest, Hungary, H-1111
edori@informatika.ilab.sztaki.hu, kiss@inf.elte.hu,
mrita@informatika.ilab.sztaki.hu, szabo.jacint@sztaki.hu

Abstract. In this paper we describe an analysis of customer churn in automobile third party liability insurance and identify customers who are most likely to churn in the future. Identifying likely churners allows appropriate steps to be taken to prevent customers who are likely to churn from actually churning. We measure the Segmented Price Pressure Regression (SPPR) which is implemented as a supplementary module of our insurance data analysis toolkit and offer better classifications improving the ROC value by more than 10-15%. For this study we create a dedicated data mart, run a data mining tool and compare classification algorithms. The methodology includes adopting an appropriate definition of churn, analyzing historical data to identify significant features, preparing data for data mining, training different prediction models, verifying and comparing the results. The tests tell us that there is a significant statistical difference between Random Subspace and other classification models. The results show that Random Forest, J48, LogitBoost are also excellent alternatives to other methods in the prediction of customer churn in car insurance.

1 Introduction

Costumers typically purchase products from insurance companies who they perceive to be offering the best products at the lowest price. And while costumers are often loyal to the insurance company they are familiar with, they will surely shift allegiance if they believe they can obtain better products or a better price somewhere else. Losing customers to competitors can significantly cut into a company's revenue. Taking active steps to prevent customer "churn" is a high priority for many businesses. [1–7]

For managing this phenomenon, data mining is one of the major tools to insurance companies. Customer information, claim and policy histories, competitors' prices are the most important sources to setting strategies regarding customer relationship management. The data mining analysis methods related to customer churn include mostly classification algorithms.

Compulsory automobile third party liability insurance provides coverage for damages caused by the driver of the vehicle to others. Automobile third party liability insurance is not a voluntary insurance; it is a legally prescribed obligation of every owner of a motor vehicle. After the insurance has been concluded, during the year it is not allowed to choose another insurer, also, during the term of the insurance the contract may only be terminated in the case of lapse of interest (for example, sale of the vehicle). The contract may be terminated as of the end of the insurance year (December 31). Every contracting party who holds an individual vehicle third party liability insurance is subject to the bonus-malus system. "Bonus" means promotion to a higher bonus class after the period of observation, earned by no claims (discount off the premium), while "malus" means demotion to a lower class, depending on the number of at-fault loss claims (surcharge). When the premiums are determined, always the period of observation must be taken into account. As 30 days before the end of the year all insurers publish their rates and discounts in newspapers and on the Internet, every policyholder may compare them and make decision to renew the contract or change insurer.

Classification is a classic data mining task, with roots in machine learning. [8] The classification problem involves data which is divided into two or more groups, or classes. In our example the two classes are "switched insurer" and "didn't switch". Classification tasks can be divided into two sorts: supervised classification where some external mechanism (such as human feedback) provides information on the correct classification, and unsupervised classification, where the classification must be done entirely without reference to external information. A number of classifiers have been used to classify objects. An example of these classifiers are neural networks, support vector machines, decision trees, regression, Bayes classifier. In Weka 3.6.0. which is an open source collection of machine learning algorithms for data mining tasks, the classifiers are grouped into eight types (Bayes, functions, lazy, meta, mi, misc, trees, rules). Each type of classifiers has a number of modifications, so Weka 3.6.0 provides 117 classifiers altogether. [9]

SPPR is a special classifier implemented as a supplementary module of our insurance data analysis toolkit. It fits the user requests but after a year use we wanted to know its efficiency more exactly. Our goal is to update our complete software and in the second version we want to implement a better classification if we find one.

It is impossible to construct a perfect classification model that would correctly classify all examples from any given test set. Therefore we have to choose a suboptimal classification model that best suits our needs and works best on our problem domain. There are different quality measures that can be used for such classifier selection. Measures are related to the accuracy, speed and comprehensibility of the classification. ROC curve performance analysis is the most appropriate and widely used method for a typical machine learning application setting where choosing a classifier with best classification accuracy for a selected problem domain is of crucial importance. Lift chart is also popular in data mining mostly in marketing and sales applications. [10]

The churn prediction has a large literature but only few of them are dealing with car insurance data. [11–13]

This study is structured as follows. Section 2 introduces basic concepts of classifiers, that are used in our experiments. As our goal was to measure the Segmented Price Pressure Regression (SPPR) classifier, we give its formal definition here. In section 3 we summarize our experimental results, which show that the Random Subspace classifier is the best one for the churn prediction, and it is much better than SPPR. Finally, in section 4 we give some ideas for further research.

2 Research Background

Statistical classification is a procedure in which individual items are placed into groups based on quantitative information on one or more characteristics inherent in the items and based on a training set of previously labeled items. Formally, the problem can be stated as follows: given training data $\{(\mathbf{x_1}, y_1), \ldots, (\mathbf{x_n}, y_n)\}$ produce a classifier $h : \mathcal{X} \to \mathcal{Y}$ which maps an object $\mathbf{x} \in \mathcal{X}$ to its classification label $y \in \mathcal{Y}$. The applications of classifiers are wide-ranging.

In this section we review the well-known classifiers which were used in our experiments. Then we give the formal details of Segmented Price Pressure Regression (SPPR) classifier. Finally we consider the definitions of ROC and Lift values that enable to compare the "goodness" of given classifiers.

2.1 Random Subspace

The essence of the Random Subspace method is that feature subsets are selected randomly from the entire range of features. Then the base classifiers are trained on these randomly selected subsets. The procedure takes advantage of the randomness, thus inducing diversity in the ensemble of experts. The parameters of the Random Subspace method are: the number of features in the subset, the number and type of base classifiers in the ensemble and the decision fusion rule. The default base classifier of Random Subspace in Weka is RepTree, which is a fast decision tree learner, that builds a decision/regression tree using information gain/variance and prunes it using reduced-error pruning with backfitting.

2.2 Random Forest, Random Trees

Random Forest is a classifier that consists of many decision trees and outputs the class that is the mode of the classes output by individual trees. The term came from random decision forests. The method combines the bagging idea and the Random Subspace method to construct a collection of decision trees with controlled variations. In Weka the Random Trees classifier is a Java Class for constructing a tree that considers K randomly chosen attributes at each node, while performs no pruning. The default base classifier of Random Forest in Weka is J48.

2.3 C4.5 (J48)

C4.5 is an algorithm used to generate a decision tree. C4.5 is an extension of ID3 algorithm. Weka classifier package has its own version of C4.5 known as J48.

C4.5 builds decision trees from a set of training data using the concept of information entropy. C4.5 uses the fact that each attribute of the data can be used to make a decision that splits the data into smaller subsets. C4.5 examines the normalized information gain (difference in entropy) that results from choosing an attribute for splitting the data. The attribute with the highest normalized information gain is the one used to make the decision. The algorithm then recurs on the smaller sublists.

2.4 Logistic Regression

Logistic Regression is a model used for prediction of the probability of occurrence of an event by fitting data to a logistic curve which is the most common sigmoid curve. It is a generalized linear model used for binomial regression. It makes use of several predictor variables that may be either numerical or categorical.

Other names for Logistic Regression used in various other application areas include logistic model, logit model, and maximum-entropy classifier.

2.5 AdaBoost

AdaBoost, short for Adaptive Boosting, is a machine learning algorithm. It is a meta-algorithm, and can be used in conjunction with many other learning algorithms to improve their performance. AdaBoost is adaptive in the sense that subsequent classifiers built are tweaked in favor of those instances misclassified by previous classifiers. AdaBoost is sensitive to noisy data and outliers. Otherwise, it is less susceptible to the overfitting problem than most learning algorithms.

AdaBoost calls a weak classifier repeatedly in a series of rounds $t = 1, \ldots, T$. For each call a distribution of weights D_t is updated that indicates the importance of examples in the data set for the classification. On each round, the weights of each incorrectly classified example are increased (or alternatively, the weights of each correctly classified example are decreased), so that the new classifier focuses more on those examples. The default weak classifier of AdaBoost in Weka is the Decision Stump, which is a machine learning model consisting of a one-level binary decision tree with categorical or numerical class labels.

2.6 LogitBoost

LogitBoost is a boosting algorithm. If one considers AdaBoost as a generalized additive model and then applies the functional cost of logistic regression, one can derive the LogitBoost algorithm. LogitBoost can be seen as a convex optimization. The default weak classifier of LogitBoost in Weka is the Decision Stump.

2.7 SVM (SMO)

Support vector machines (SVMs) are a set of related supervised learning methods used for classification and regression. Viewing input data as two sets of vectors in an n-dimensional space, an SVM will construct a separating hyperplane in that space, one which maximizes the margin between the two data sets. To calculate the margin, two parallel hyperplanes are constructed, one on each side of the separating hyperplane, which are "pushed up against" the two data sets. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the neighboring datapoints of both classes, since in general the larger the margin the better the generalization error of the classifier. The SVM implementation is called "SMO" in Weka.

2.8 Naive Bayes

A naive Bayes classifier is a simple probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions. A naive Bayes classifier assumes that the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature. Naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for naive Bayes models uses the method of maximum likelihood.

2.9 Non-Nested Generalized Exemplars (NNGE)

NNGE performs generalization by merging exemplars, forming hyperrectangles in the feature space that represent conjunctive rules with internal disjunction. NNGE forms a generalization each time a new example is added to the database, by joining it to its nearest neighbour of the same class. It does not allow hyperrectangles to nest or overlap. This is prevented by testing each prospective new generalization to ensure that it does not cover any negative examples, and by modifying any generalizations that are later found to do so.

2.10 Segmented Price Pressure Regression (SPPR)

SPPR is a special method for insurance data to predict the probability of churn when the client knows what is the price of his policy at the other insurers. The output of SPPR is a set of models where a model is a pair of price pressure and a regression function. Each model is tested then the best is chosen to predict the probabilities.

This method uses the heuristics that a client will switch the insurer if the price of his contract is relatively to high compared to the other proposals. In general domain experts tell what means "relatively to high". For this purpose usually different measures are taken called price pressures. The higher the price pressure the higher the churn probability. Here we give some examples for price pressure measures.

Suppose a client has a policy at an insurer with the price A and there are $n-1$ other insurers offering this kind of policies. Then take the increasing ordered list of the n prices. Denote $Index(A)$ the position where A appears first in list. Min, $Min2$, $AvgMin3$, Avg denote the smallest, the second smallest, the average of the three smallest price, and the average of the prices, respectively. Then we define the following price pressures:

- pp1: $\frac{Index(A)}{n}$
- pp2: $\frac{A}{Min}$
- pp3: $\frac{A}{Min2}$
- pp4: $\frac{A}{AvgMin3}$
- pp5: $\frac{A}{Avg}$

The insurer gives the rectangular segmentation of the policy space by making partitions in a few dimensions. For example the age can be categorized as it is less then or equal to 24, is between 25 and 30, or is between 31 and 50, or more. The dimensions for segmentation in our study were the engine size of the vehicle, the location, the age of the owner, the number of children, the policy age.

The policy space can be represented in 2 dimensions for a given price pressure type by (x, y) points where the first component is the price pressure value and the second is the indicator value of churning that is 1 if the policy is terminated and 0 otherwise. In order to make a model we choose a price pressure and a function to perform the regression.

For example we fit linear curves and a family of logistic curves.

- linear regression: fit $y = ax + b$ to the (x, y) points in each segment. If a,b are the computed parameters for a segment then $ax + b$ is the estimation of the churn probability for the x price pressure value in this segment.
- log_const regression: first transform y values from $[0, 1]$ to [const, 1-const] with $T(y) = (1 - 2 \cdot const)y + const$ then fit $\frac{1}{1+exp(ax+b)}$ to the $(x, T(y))$ points in each segment. If a,b are the computed parameters for a segment then $T^{-1}(\frac{1}{1+exp(ax+b)})$ is the estimation of the churn probability for the x price pressure value in this segment.

In our case "const" is set to 0.2, 0.25, 0.3 and 0.35. It means we have 5 different curves, one linear and four logistic curves.

The SPPR method defines a model as a (price pressure, curve) pair, so we get 25 models. On the training set all the 25 models are fitted on each segment to compute the a,b parameters, then the globally best model on the test set is chosen, which is called the best SPPR classifier.

2.11 ROC and Lift Chart

ROC curve and Lift chart are well known graphical techniques that are useful for evaluating the quality of classification models used in data mining. Each technique defines its own measure of classification quality and its visualization. When dealing with two class classification problems we can always label one class as a positive and the other one as a negative class. The test set consists of P positive and N negative examples. A classifier assigns a class to each of them, but some of the assignments are wrong. To assess the classification results we count the number of true positive (TP), true negative (TN), false positive (FP) (actually negative, but classified as positive) and false negative (FN) (actually positive, but classified as negative) examples.

It holds $TP + FN = P$ and $TN + FP = N$.

Let us define a few well-known and widely used measures:

$$FPrate = \frac{FP}{N}, \qquad TPrate = \frac{TP}{P} = Recall,$$

$$Yrate = \frac{TP + FP}{P + N}, \quad Precision = \frac{TP}{TP + FP}, \quad Accuracy = \frac{TP + TN}{P + N}.$$

Precision and Accuracy are often used to measure the classification quality of binary classifiers. Several other measures used for special purposes can also be defined.

A probabilistic classifier $f : X \rightarrow [0, 1]$ assigns a probability to each example. Normally, a threshold t is selected for which the examples where $f(x) \geq t$ are considered positive and the others are considered negative.

Each pair of a probabilistic classifier and threshold t defines a binary classifier. Measures defined above can therefore also be used for probabilistic classifiers, but they are always a function of the threshold t. Note that $TP(t)$ and $FP(t)$ are always monotonic descending functions. For a finite example set they are stepwise, not continuous.

ROC graph is defined by a parametric definition

$$x = FPrate(t), \qquad y = TPrate(t).$$

Area under ROC curve A_{ROC}, called ROC value of the method, is often used as a measure of quality of a probabilistic classifier. Lift chart is quite similar to the ROC curve.

Lift chart is a graph with a parametric definition

$$x = Yrate(t) = \frac{TP(t) + FP(t)}{P + N}, \qquad y = TP(t).$$

Area under lift chart A_{lift} can be used as a measure of classification quality of a probabilistic classifier. A_{lift} and A_{ROC} are in strong relation.

$$A_{lift} = \frac{1}{P + N}(\frac{P^2}{2} + P \cdot N \cdot A_{ROC})$$

ROC curve performance analysis is the most widely used method for a typical machine learning application where choosing a classifier with best classification accuracy is of crucial importance.

3 Experiments

In this section we present our experiments on a given insurance data set. Our goal was to test the efficiency of the SPPR method for churn prediction, and offer a better classifier if we find such one.

In the insurance industry premium prices play a critical role in enabling insurance companies to find a balance between producing high profit and retaining a certain level of the market share. The challenge is to set premium prices so that expected claims are covered and a certain level of profitability is achieved, yet not to set premium prices so high that market share is jeopardized as consumers exercise their rights to choose their insurers. Compulsory third-party liability motor insurance are sold by about 15 Hungarian insurance companies every year. Insurance companies determine premium prices by assigning policy holders to pre-defined groups. The groups are formed based on industry experience about the perceived risk of different groups of policy holders. By using data mining techniques, the aim is to predict the churning rate. We measured the SPPR method and offered better classifications improving the ROC value significantly, by more than 10-15%.

3.1 The Data

In this section we describe some of the characteristics of the data set. For the analysis we created a "star-like" data mart consisting of a master table of the policies and a set of smaller tables called dimension tables. Our goal was to apply only open source tools, therefore we implemented our relational model with PostgreSQL 8.1.15. The Policy table has 110 different features with a primary key "pid". A policy record includes aggregate statistics about its yearly price, payment and claim history. Essential features for churn analysis are the yearly prices of the other 14 insurers. Every record has an indicator attribute for each year whose value is 1 if the policy holder changes the insurer in that year and 0 otherwise. Time series go back 5 years. The table size is 853652 tuples. The insurer makes segmentation of the policy space by some given features of the vehicle and the owner such as the type of the vehicle, location, age of the owner and so on. Each segment has a hash key and each policy is joined to the segment to which it belongs by this hash key. There are 1296 segments and each segment has a basic constant price for each insurer which is modified by a bonus-malus and other discount factors for each policy.

3.2 Preprocessing the Data

In order to provide an optimal representation for the chosen data-mining technique we made the necessary preprocessing including simple transformations and cleaning. The competitors' prices are the most important features for us but they are not always available in the previous years. Fortunately the last 2 years, 2007 and 2008 were complete from this point of view. Therefore we reduced our dataset to these 2 years. We selected the policies living in 2007 and having the label that they changed the insurer in 2008 or not. Finally we left out some records which had too many missing values in the important features. So we get 208303 records. Then we added a new attribute for each price pressure measure and filled them for each record.

3.3 Data Mining with Weka

For data mining we chose another open source tool. Weka (Waikato Environment for Knowledge Analysis) is a popular suite of machine learning software written in Java, developed at the University of Waikato. Weka supports several standard data mining tasks, for example data preprocessing, clustering, classification, regression, visualization, and feature selection. Weka contains a collection of visualization tools and algorithms for data analysis and predictive modelling, together with graphical user interfaces for easy access to this functionality. Weka provides access to SQL databases using Java Database Connectivity and can process the result returned by a database query, so Weka works well with PostgreSQL. In Weka Explorer interface the Classify panel enables the user to apply classification and regression algorithms, to estimate the accuracy of the resulting predictive model, and to visualize erroneous predictions, ROC curves. Weka has a large number of classifiers, including all mentioned ones in the previous section except SPPR, therefore we had to implement only SPRR in Java. For each method we applied 10-folds cross validations. There are a lot of parameters for the classifiers in Weka which can be modified to get better performance, so we made some trials, but in our case the default values produces the best result or only a slightly improvement could be achieved.

3.4 Results

The experiments were executed on a SUN machine, with 60 gigabyte RAM, on Linux 2.6.27 operating system. First we ran basic statistics, histograms then we ran the classifiers described previously, the 9 general methods and the 25 SPPR models. The processing time of classification for one given classifier varied between 0.1 and 6 hours.

Classifier	A_{lift}	A_{ROC}
Random Subspace	60068.1	0.811973
Random Forest	57839.9	0.768183
J48	57290	0.757383
LogitBoost	57164.4	0.754908
AdaBoost	56249.9	0.736935
Logistic	56243.1	0.736809
(pp1, log_0.3)	50277.9	0.623892

Table 1. The best classifiers

In the literature SVM is thought to be one of the best methods for churn prediction but our tests did not justify this preconception. SVM produced only 0.568 A_{ROC} value while the best Random Subspace had 0.811973. From the 25 SPPR classifiers (pp1, log_0.3) had the highest A_{ROC} value, which was 0.623892. The other SPPR models were behind this a little bit, but all had A_{ROC} value at least 0.61. The rest of the classifiers were worse, their A_{ROC} values did not reach 0.6. The table shows the top 7 classifiers.

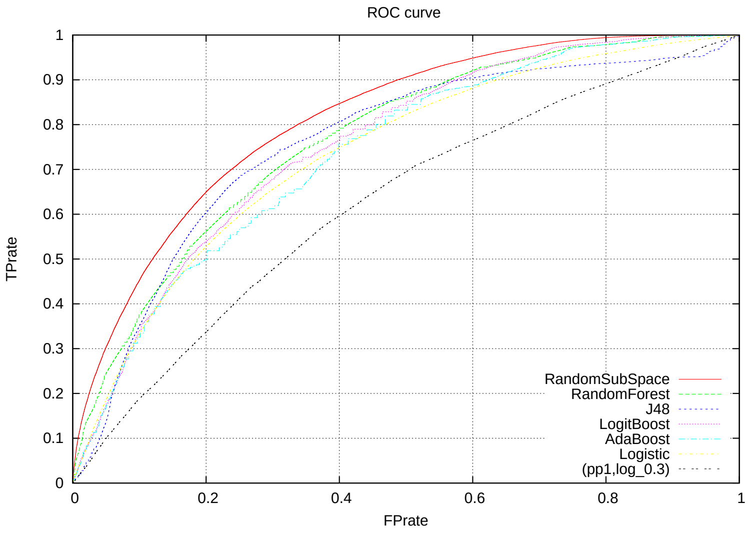

Fig. 1. The ROC curves of the best classifiers

The winners have very similar ROC curves as one can see in the figure, excepting SPPR, which is much below the others.

4 Conclusions and Future Work

We concluded that Random Subspace classifier is the most suitable one for churn prediction in car liability insurance. It is much better than the SPPR models. The problem with SPPR can be in the forced segmentation. We think if preclustering were performed in the policy space and price plans were fitted to these clusters then we would receive better results with SPPR.

In the car insurance business the movement rate from one insurer to another is usually very high, we got 42.43%, but this is only the one side of the coin. Because it is obligatory to buy liability insurance for each car, so there are always new incomers from other insurers. The two processes balance each other in general. We plan to make models to estimate the number of incomers and validate the models on our data set.

The policies generate premium income and losses to be paid. Joining the policy table to the loss table, we can estimate the positive and negative payments caused by churning and immigration, which we plan also in the near future.

References

1. Morik, K., Köpcke, H.: Analysing customer churn in insurance data - a case study. In: PKDD. (2004) 325–336
2. Hadden, J., Tiwari, A., Roy, R., Ruta, D.: Computer assisted customer churn management: State-of-the-art and future trends. Computers & OR **34**(10) (2007) 2902–2917
3. Xie, Y., Li, X., Ngai, E., Ying, W.: Customer churn prediction using improved balanced random forests. Expert Systems with Applications **36**(3, Part 1) (2009) 5445 – 5449
4. Lee, J.S., Lee, J.C.: Customer churn prediction by hybrid model. In: ADMA. (2006) 959–966
5. Yang, L.S., Chiu, C.: Knowledge discovery on customer churn prediction. In: MATH'06: Proceedings of the 10th WSEAS International Conference on APPLIED MATHEMATICS, Stevens Point, Wisconsin, USA, World Scientific and Engineering Academy and Society (WSEAS) (2006) 523–528
6. Au, W.H., Chan, K.C.C., Yao, X.: A novel evolutionary data mining algorithm with applications to churn prediction. IEEE Trans. Evolutionary Computation **7**(6) (2003) 532–545
7. Chiang, D.A., Wang, Y.F., Lee, S.L., Lin, C.J.: Goal-oriented sequential pattern for network banking churn analysis. Expert Systems with Applications **25**(3) (2003) 293 – 302
8. Han, J., Kamber, M.: Data Mining: Concepts and Techniques (The Morgan Kaufmann Series in Data Management Systems). Morgan Kaufmann (September 2000)
9. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques. Second edn. Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann (June 2005)
10. Vuk, M., Curk, T.: Roc curve, lift chart and calibration plot. Metodološki zvezki **3**(1) (2006) 89–108
11. Hur, Y., Lim, S.: Customer churning prediction using support vector machines in online auto insurance service. In: ISNN (2). (2005) 928–933

12. Christmann, A.: On a strategy to develop robust and simple tariffs from motor vehicle insurance data. Acta Mathematicae Applicatae Sinica **21** (May 2005) 193–208(16)
13. Yeo, A.C., Smith, K.A., Willis, R.J., Brooks, M.: Modelling the effect of premium changes on motor insurance customer retention rates using neural networks. In: International Conference on Computational Science (2). (2001) 390–399