Insurance Classification

Project Team: Weiwen Yu, wyu14
Tianqi Wu, twu38
Chao Yu, chaoyu2

Kaggle Team Name: Wudi

I. Introduction

In this supervised learning project, we are given different attributes of the life insurance applicants with the level of risk in providing the insurance. Our goal is to predict to which level the risk of the new insurance application belongs. The prediction can be crucial since it provides the company evaluation of the potential customers. Risky application can be rejected to increase the revenue. Also, the company may focus more on the clients with low risk and find out the characteristics of potential safe customers. The problem is challenging since there are 128 features with lots of missing values and the importance of the attributes has to be understood in order to make promising predictions. Moreover, prediction algorithm has to be chosen from various models to fit the data.

Among all the features of the clients, there are three categories of independent variables, which are nominal, continuous, and discrete features. After fitting the data with certain model, response is then predicted based on the attributes of the clients.

For the data preprocessing, pilot treatment includes missing value imputation, numerical value scaling along each attribute and categorical variables encoding. In the imputation process, the median of that numerical attributes is chosen to fill the empty values and missing entries of the categorical attributes is classified as a new value. Numerical attributes are scaled by removing the mean and scaling to the unit variance. Values combined of character and digit like ‘D4’ appear in one attribute. Thus, this attribute is transformed using label encoder. Moreover, we only keep those features which have correlation value larger than 0.15 to response. Finally, the preprocessed data is then split into 80% of training data and 20% of testing data. Random forest classifier from the sklearn library is chosen to be the model to fit the data.

II. Related Work

The project team reviewed “An Ensemble Random Forest Algorithm for Insurance Big Data Analysis” [1] as the related technical support. Ensemble random forest algorithm is the main contribution from this paper. Random forest is consisted of many decision trees. In the project,

the main idea of model construction is to build decision trees and let the instances classified by each tree give a vote at each class. Each decision tree will be fully grown to its largest extent. To increase the prediction accuracy, a bootstrap sampling is used to deal with minor classes. The main advantage of this algorithm is that it will select features automatically and can handle a large number of input attributes. However, the weakness of random forest is that the algorithm is biased to the variables with high number of levels. Moreover, it does not deal with irrelevant features. The inspiration we can obtain from this paper is that ensemble thinking can be incorporated into our later revised model. Weak learners comprising the final random forest can be built on best feature selected from the bootstrap subset of all features.

III. Result Interpretation

As a result, our prediction scored 0.427 after submitting the generated solution to Kaggle. Several baseline models including decision tree classifier, random forest classifier and logistic regression with default parameters have been tested. We chose random forest classifier to be our present model since it gives higher prediction score than the other two algorithms. To improve the result, different parameters of the model can be learned and more prediction algorithms can be studied.

IV. Work distribution

Weiwen Yu: Related work study and algorithm recommendation

Tianqi Wu: Implementation of the algorithm

Chao Yu: Introduction written and data preprocess

V. Reference

[1] Lin, Weiwei et al. “An Ensemble Random Forest Algorithm for Insurance Big Data Analysis.” *22017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)* 01 (2017): 531-536.