Predicting Probability of the Applicant Filing a Claim | Real Insurance Data Melike Savas

1. Work Sample Goal

The main goal of this task is by using the given data sets; after data aggregation, and manipulation create a model to predict the probability of the applicant filing a claim within a one-year timeframe.

2. Data Description and Manipulation

I started with the 'Claim Data' dataset which has all claims that were found from a Loss History Report at the driver level. It has 4 features with 5000 variables. I am dealing with the only amount paid for the claim being greater than 0, that's why I selected the units with the 'pd_amt' variable greater than 0. I also need to focus on the units which have a claim year between 2012-2017 since we wanted to include each of the last 5 years of claim history differentiation whether the claimant is at fault or not. Combining both dimensions, I created 10 new variables which tell me the number of claims in the last 5, 4,3,2,1 years with at fault 0 or 1.

I also have the 'Predictor Dataset' and 'Subsequent Loss Experience Dataset' datasets which are respectively giving me information about the household level collected at the time of the application and one year of subsequent loss-experience of these applicants. Both have 20000 household units, and 41 and 5 variables respectively.

I merged these two datasets by using the 'inner join' method on the 'hhld_id' column. Then I used the "left join" method to merge the new dataset with the first dataset based on the "hhld_id" column. I understand that not every applicant in the predictor dataset has a claim history. I assumed that if the applicant id is not in the predictor dataset then it would mean the applicant has no claim history. Therefore, I used an indicator to see who has no_claim then I dropped the _merge column which is created when using the pandas merge function. I also dropped all columns that came from the 'Subsequent Loss Experience' dataset except the 'future_clm_ind' variable to have only information known on or before the application date.

2.1. Handling Missing Values

I checked a number of missing values for each variable, then I decided to keep all variables to avoid information loss and handled missing values with imputation. I had 99.9% of the 'veh_lien_cnt' variable missing, however, I decided to keep it since the variable is important and it is expected to generally vehicles don't have liens, and I filled missing values with 0. From the Metadata definition, I understand that the 'prior_bi' and 'time_w_carr' variables could be missing if there is no current insurer. For that reason, I imputed both of them to zero. I imputed the remaining numerical variables by using mean or median. And for the new variables that I created in the "Claim Data" dataset, I filled missing values with 0.

I checked univariate plots to see the relationship of each variable with 'future_clm_id' and looked at their distribution of them.

2.2. Outliers

I am planning to use Random Forest Model. Theoretically, outliers have a negligible effect when dealing with tree-based models. However, the modeling dataset is small compared to the number of features after the creation of zip-level indicators. (see section 2.4 below). It still may have an impact therefore, it is prudent to do capping and flooring.

2.3 Correlations

I am planning to use the RandomForest model. Multicollinearity is not a problem for the accuracy of tree models generally as long as the correlation structure between the variables does not change over time. However, it will impact the variable importances which I would like to check as part of model validation. Variable importance will be impacted by highly correlated variables.

As I see from the correlation matrix and plot, some variables are highly correlated such as 'max_age', 'min_mon_loc', and 'max_mon_lic'. I will consider them as important variables if any variable correlated to them turns out important.

Secondly, I understand that most of the variables with high correlations are derived from one root variable, i.e all the age-related variables. I think the risk of change in correlation for these variables is minimal, therefore I am comfortable with the correlation structure in the dataset.

2.4 Feature Engineering

Before modeling I checked datatype for all variables. The 'Curnt_insurer' variable is a categorical variable. I created dummy variables for each insurance company.

I also created dummy variables for each 'zipcode'; Ideally, if I can get vehicle theft data or other important demographic data by zipcode, I would use them as features. That way, I would be able to use the model for any zip that is not in the modeling dataset but have similar zip-level properties with the ones in the sample I attempted to collect data from public FBI crime reports by MSA level however 50% of the zip codes were non-US. Therefore I have used zipcode indicators for this model, and with that, I added 112 new columns.

3. Model Selection

Future Claim indicator is a binary variable therefore I use classifier models. However, this dataset has an imbalanced class problem that needs to be addressed. One approach to making the random forest more suitable for learning from imbalanced data can be done by placing a heavier penalty on misclassifying the minority class. To achieve that I set the class_weight argument on the RandomForestClassifier to "balanced".

For the random forest model, my target variable is the 'future_clm_ind', and the rest of the variables will be used for predictors. I split the dataset as train and test sets within the ratio of 3. For hyperparameter tuning, I have used the the hyperparameters(Table 3.1) for grid search with 3 cross-validation folds.

'bootstrap' :	True, False
'max_depth' :	10, 20, 30, 40, 50, 60, 70, 80, 90, 100, None

'max_features' :	'auto', 'sqrt'
'min_samples_leaf' :	1,2,4
'min_samples_split' :	2,5,10
'n_estimators' :	200, 400, 600, 800, 1000, 1200, 1400, 1600, 1800, 2000

Table 3.1 Hyperparameters for Grid Search

Note that I have expanded the search space for max depth and number of estimators since they are very important parameter settings for Random Forest.

The parameter space for this grid search is (2 x 11 x 2 x 3 x 3 x 10 = 3960), which take significant computational resources. Therefore, I used a Randomized Grid search where the algorithm randomly selects 100 elements from the grid space and finds the best parameters.

My final model is (best_random = rf_random.best_estimator_) with the following parameters
('n_estimators': 600,
'min_samples_split': 10,
'min_samples_leaf': 4,
'max_features': 'sqrt',
'max_depth': 90,
'bootstrap': False) .

4. Model Validation

In order to evaluate the model performance with the best parameters using the test dataset, I compared this final model with my base model which uses default RF settings. For comparison purposes, I have used the F1 score and ROC Curve (Table 4.1) keeping in mind the imbalanced data problem I have with this project. My final model accuracy score is 0.9944, The model has 28 misclassified units out of 5000 units. F-1 score is 93.10. ROC Curve can be found in Table 4.1. Only looking at accuracy can result in misleading results. And I also checked the confusion matrix(Table 4.2), and model performance(Table 4.3) for validation.

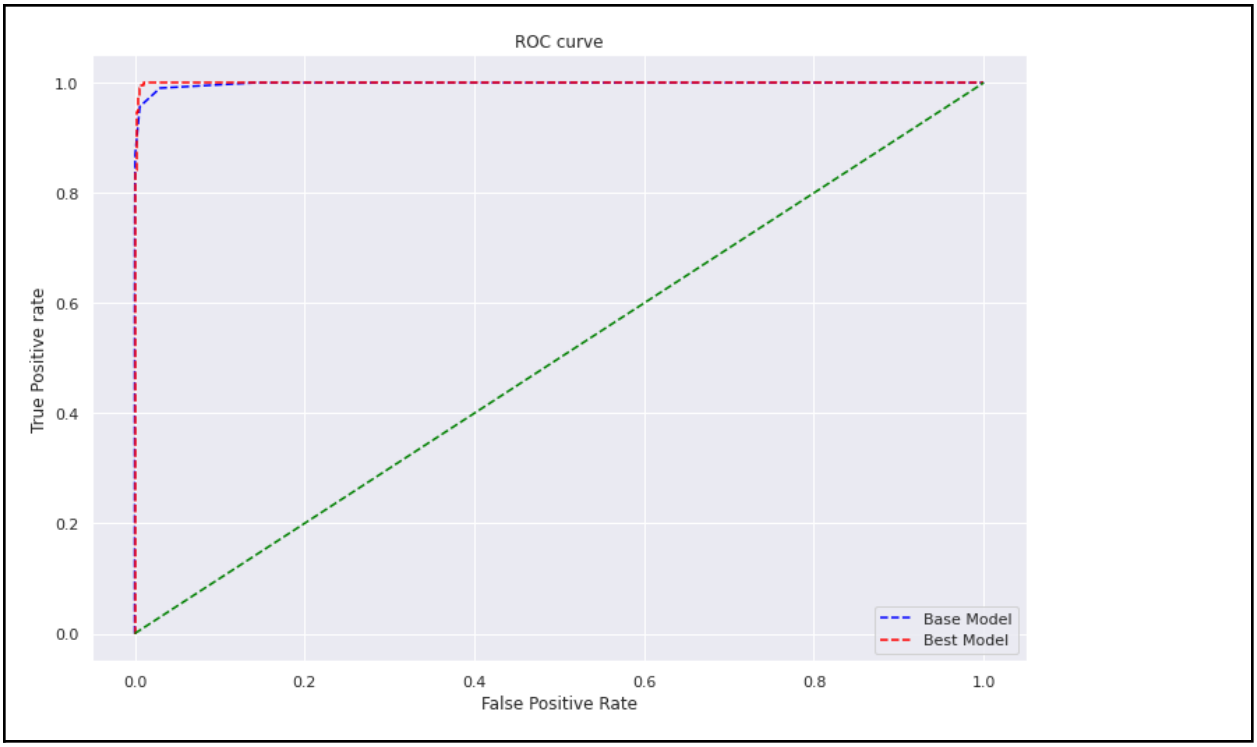

Table 4.1. ROC Curve for Base and Final Model

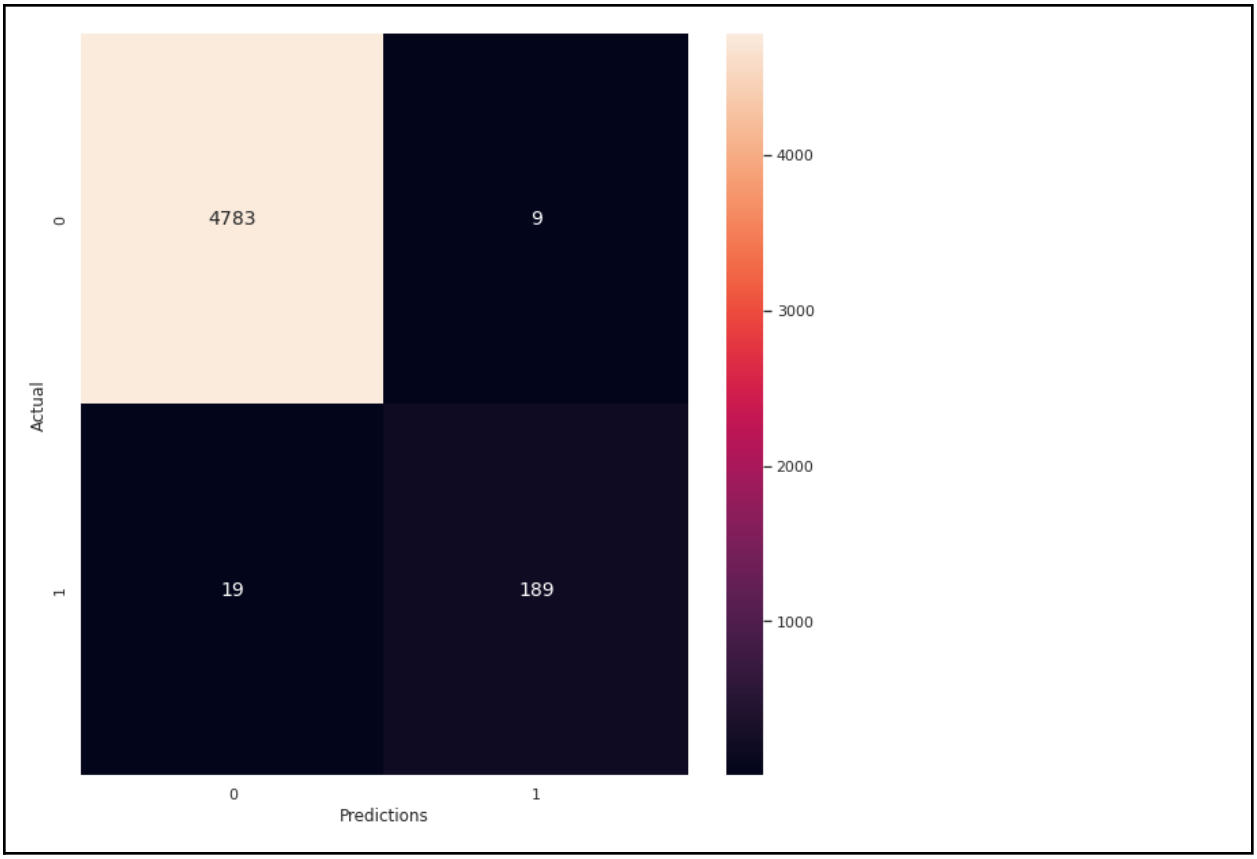

Table 3.2 Confusion Matrix for Final Model

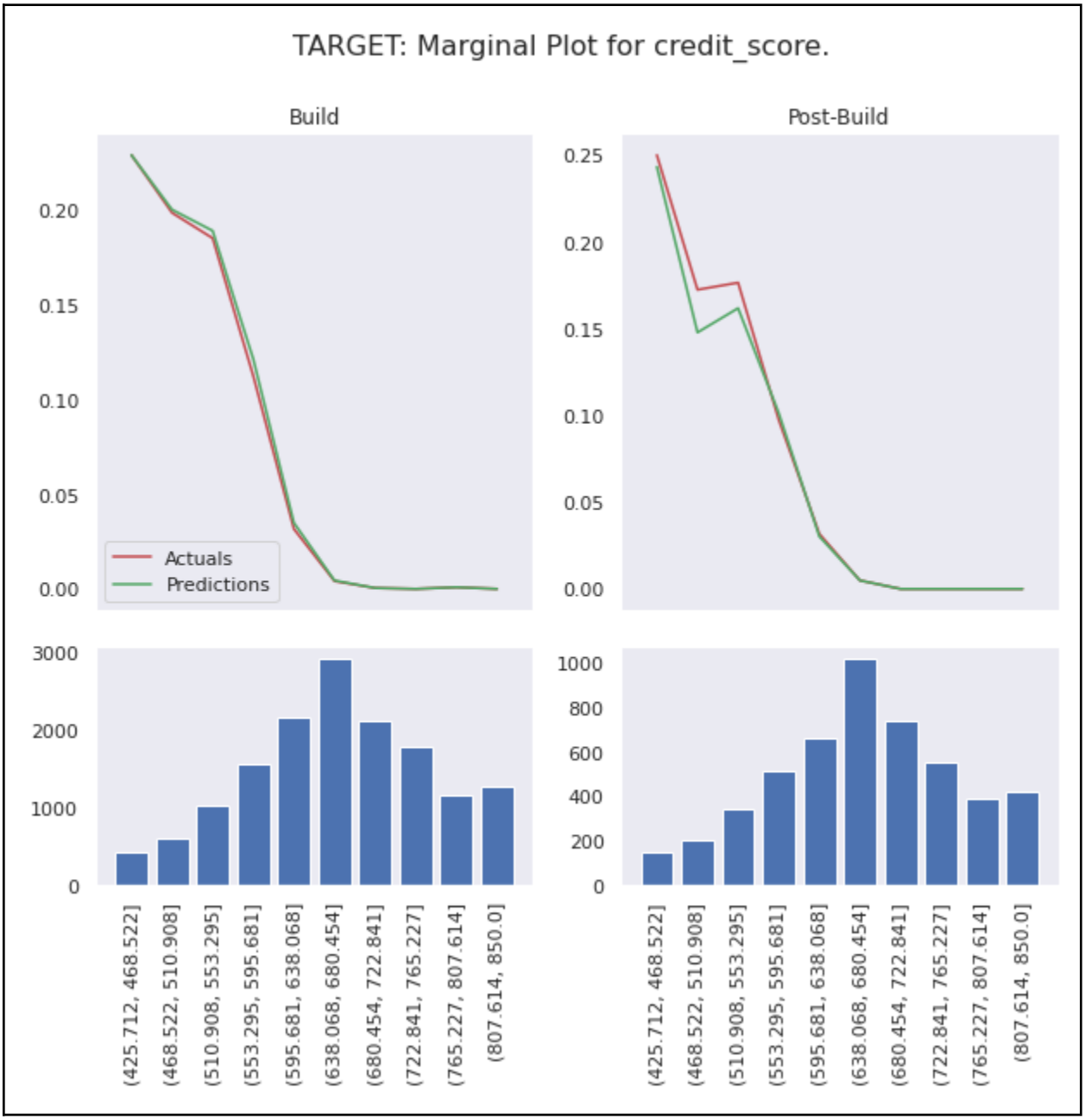

Table 4.3 Model Performance Chart that focuses on the most important variable 'credit_score'

Random Forest Method gives me also variable importance. I check which variables are more important. I only included the top 20 important variables in my chart (Table 4.6). According to the table below 'credit_score', ' prior_bi' and 'inforce_ind' are the top three most important variables.

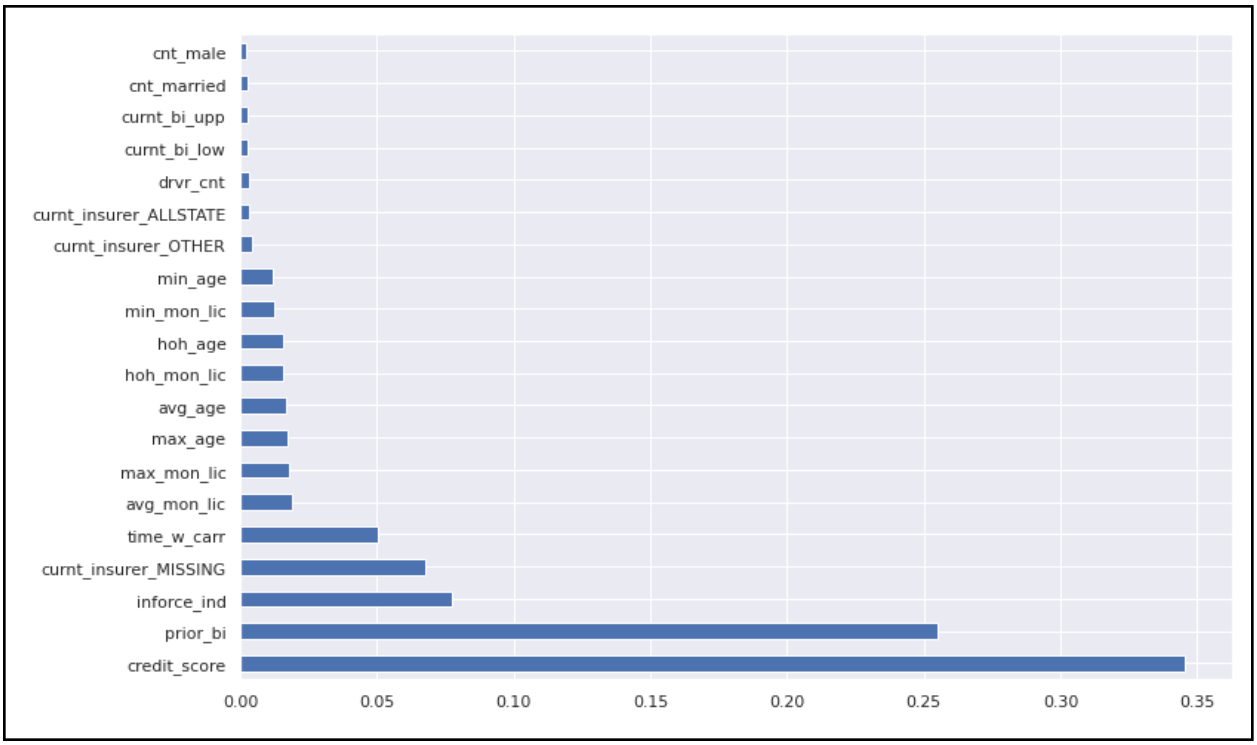

Table 4.6: Variable Importance Chart

5. Conclusion

The predictive model that I have developed shows good performance with a high F1 score and ROC Curve.

The major drivers for the probability of future claims in one year is Household's credit score: The lower the credit score the higher the probability of a future claim. (Table 4.3)

Bodily Injury Coverage Individual Limit with current insurer : When the limit is low the future claim probability is high (Table 5.1)

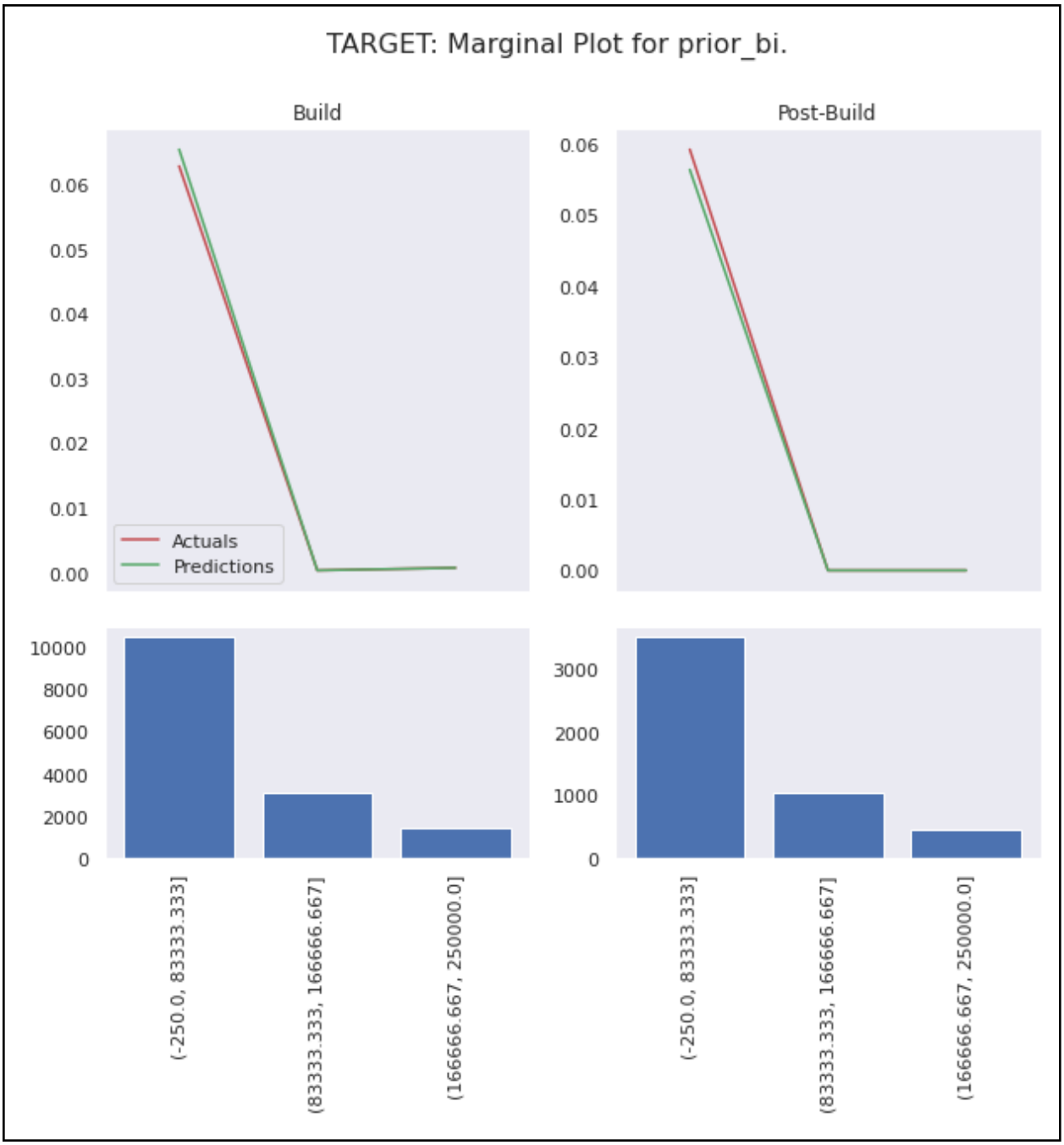

Table 5.1

Inexistence of a current insurer: If the applicant doesn't have a current insurer now the probability of future claim in one year is very low. This could be unintuitive, therefore I suggest checking with an expert. (Table 5.2)

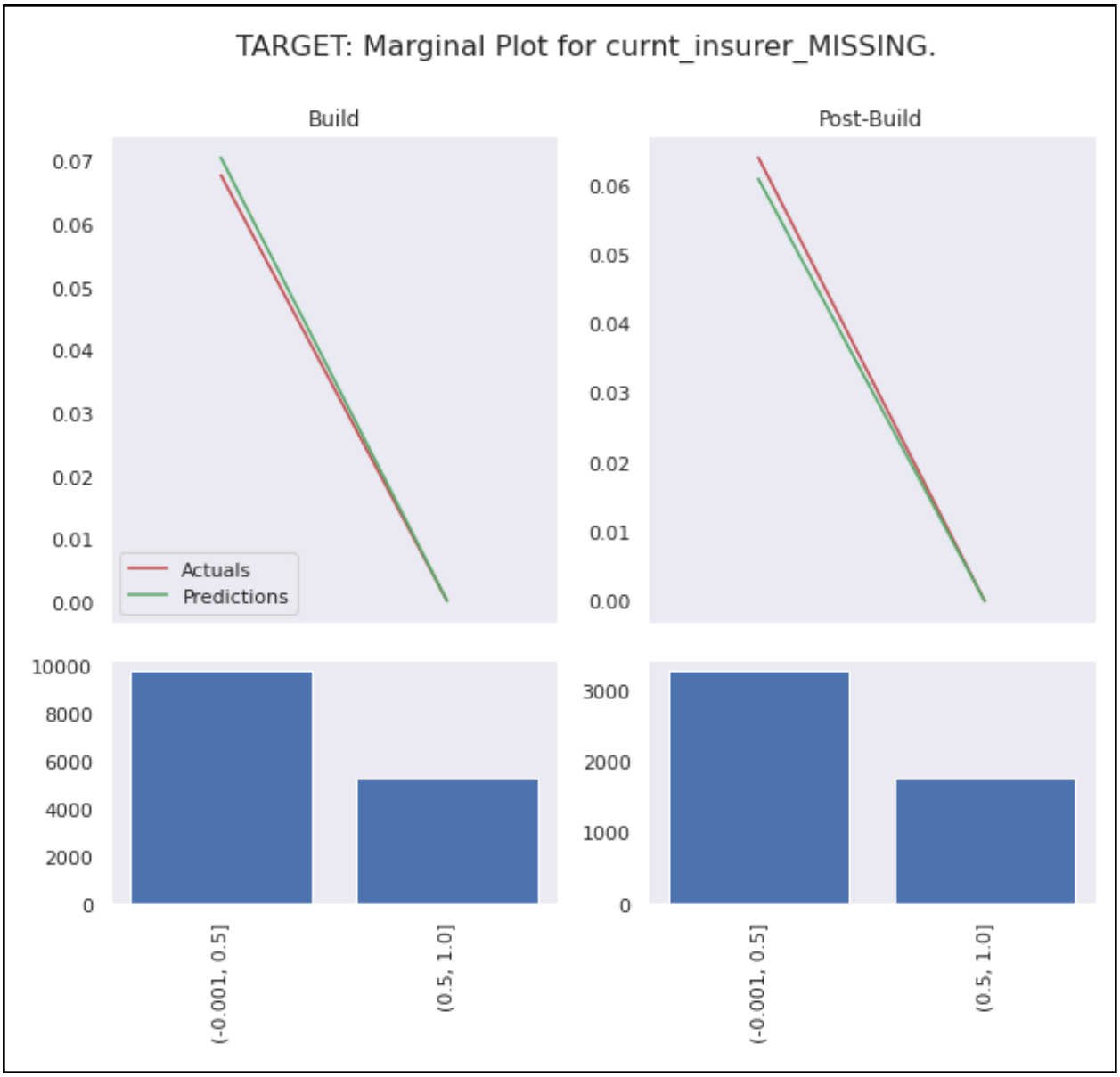

Table 5.2

Time with the current insurer has a decreasing trend. The future claim probability reduces for the applicants as they have stayed more with their current insurer. (Table 5.3)

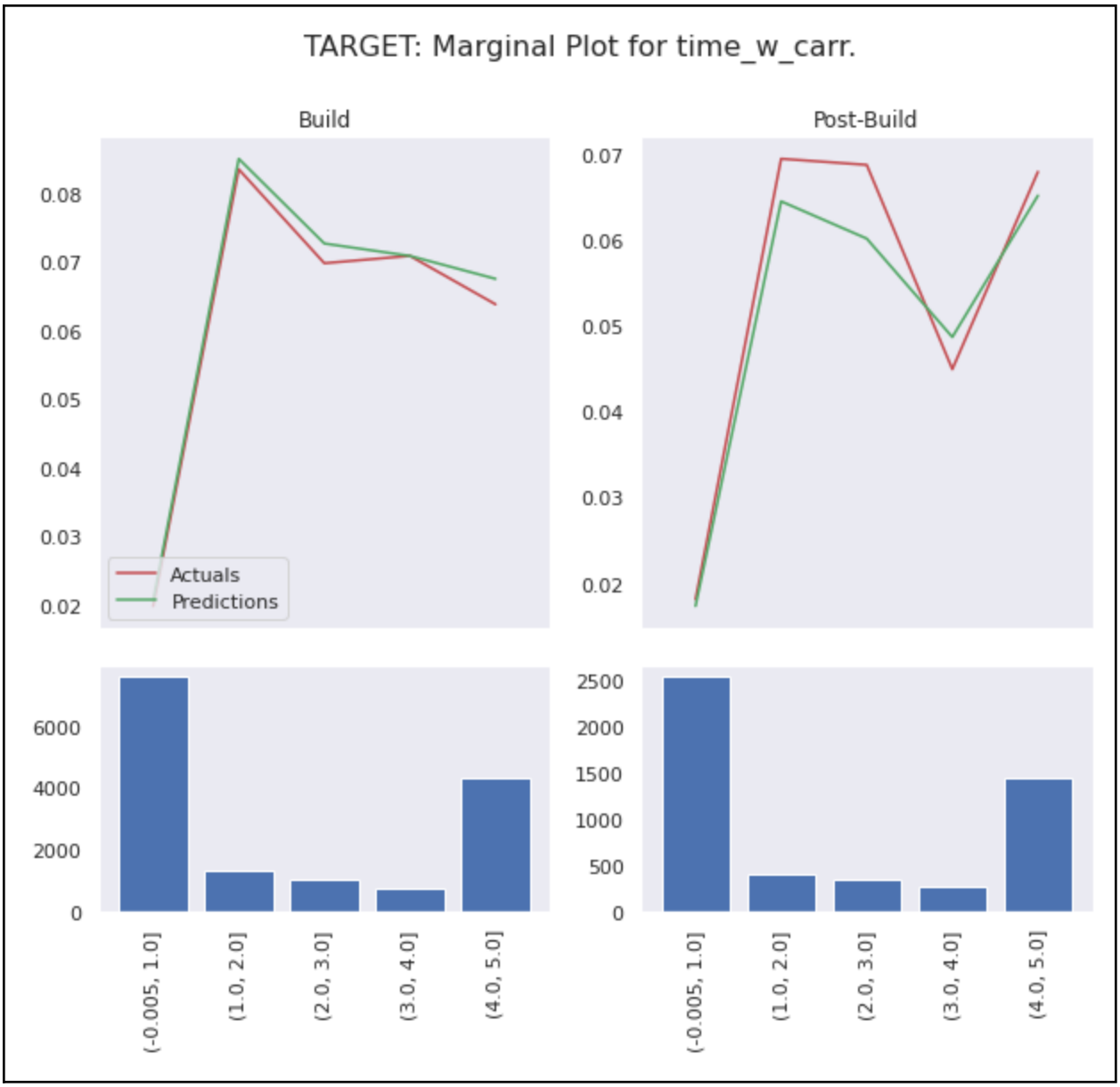

Table 5.3 Model Fit Plot focuses on 'time_w_carr'

Lastly, per my comment at section 2.4 since the dataset has only limited number of zipcodes it poses a limitation on the applicability of the model for out of sample zipcodes. One way to address that problem is mapping out of sample zipcodes to one of the in sample zipcodes based on some similarity logic and use the model accordingly.