Proposal for Auto Insurance Company

Estimating an optimum annual premium for a policy such that the company never incurs loss is crucial. Loss happens when the claim amount exceeds the total premium. Thus, by studying the loss incurred by different policies in past, the premium can be adjusted ie. either can be increased or decreased for similar such policies to reduce this loss. The main business goal is to predict the logarithm of the loss ratio for given different portfolios, for pricing the policies appropriately. This is a supervised- Regression problem. Here, the target variable ie. The log of loss ratio isn't defined directly. It needs to be calculated for each portfolio by taking natural log of portfolio loss ratio. Modeling needs to done for claim count and severity as, this two attributes will give us the loss amount and the ratio of it with the annual premium will yield the loss ratio.

Data Preparation such that the train and test data format matches is very important. For this, we will merge the individual policies in the train set into portfolios by taking combination of no. of policies (1000, 3000, or 5000) in a portfolio and % (1%, 2%, 3%, 4%, 5%, 6%, 7%, 8%, 9%, 10%, 20%) of policies having losses, the way the test data was formed. This procedure will be done for 10 portfolios thus total portfolios formed would be 330. After this, the train and test data will be preprocessed using one hot encoding and imputing the missing values for individual portfolios. Visualization will be performed to get a better understanding of dependency of target variable with other variables. Also, some variables can be aggregated to reduce dimensionality. Then we will summarize these selected attributes for a portfolio in train and test set. Summarization for Loss Amount and Premium will be summation and for claim count will be mean/mode and for severity will be mean. For other attributes this could be sum, mean, median, etc. (that would be decided by investigating data more). Thus, we perform data reduction and aim for faster computation and high accuracy. The summarization would result in a feature vector representing one portfolio. This will be stored in separate train and test csv files having 330 rows each. This new test dataset will be used for modeling. Here the only the computation cost is involved as the data is pre-acquired and as no other costs are explicitly stated. We will perform attribute selection using Lasso or PCA to compute the attributes required to calculate the loss amount. The train and test data are being drawn from the same population.

The model we are planning on using would be a Regression model with Regularization-Ridge. This model, would thus have a good generalization performance. The amount of data required would be considerably low compared to the actual dataset given since we are summarizing the data. Cross-validation will be used for yielding better results and even to select the best regularization parameter. Type of regression to use ie. Linear or Polynomial needs to be tested as no prior knowledge of data is given. Thus, various models can be tried and compared using appropriate evaluation metrics to find the right model. For predicting the loss ratio, loss amount needs to be computed first. Loss amount is nothing but claim count multiplied by severity. Thus, Loss amount is highly dependent on these two and as they are missing in the test dataset, they need to be computed first. This means we need to build separate models for estimating claim count and severity. Attribute selection can be applied for each to build this model. Alternatively, if both depend on same attributes, scikit learn's MultiOutputRegressor method can be used. It is a Multi target regression model which consists of fitting one regressor per target thus, It feels as we are using a single regressor though in reality we are using an ensemble. There is cause and correlation between them and Loss Amount, this would lead in another model that utilizes these sub-target variables to predict the final ones more accurately.

To use the model: for each portfolio in the test data we would first predict it's claim count and severity using the models built for them. After these values are computed, Loss Amount will be predicted using the final model which utilizes these two attributes too. From the Loss Amount and Annual premium, Loss ratio is calculated. Then, natural log of it will give the final target value ie. the ln(loss ratio)

The final outcome of the model is prediction of ln(loss ratio) for each portfolio, thus, the model will be in the form that the domain experts or stakeholders can understand. They can easily evaluate which portfolio exceeded the threshold and what action in terms of premium needs to be taken for it. For example, if the permissible loss ratio is 0.7 and the predicted loss ratio for a portfolio is 0.66, then Rate Change Factor = 0.666/0.7 -1 = -0.04762 thus, there will be rate reduction of 4.762% in the premium amount for all the policies belonging to that portfolio. This dataset consisting of auto policies for one year (2006) and 50% of policies are mispriced from ±10% up to 50%. Thus, this loss ratio will be used to set their premiums right. Deciding upon the right evaluation metrics is important to understand how well the model has performed. Mean Square Error or R2 Score can be used for evaluation. Also, the model needs to be evaluated upon a baseline model. For Regression and considering our project, the baseline model will be to give the highest ln(loss ratio) amongst all the portfolio's in the train dataset for every test instance. Highest ln(loss ratio) is just arbitrary and can be different for a baseline model, but when given the highest ln(loss ratio) in train for every portfolio instance in test ensures that the company doesn't suffers from loss. The model will be evaluated based on its Mean Square Error on these data; in particular, we want to ensure that the mean square error is lower or equal to 0.67658, given in the Kaggle submission.