Model Report

January 23, 2023

1 Analytic Approach

The end goal of the final model is to increase the number of predictions laying within a chosen threshold from the true target with minimal MSE degradation.

- Boston housing target - Predict median value of owner-occupied homes in $1000's.
- French motor target - Predict the frequency variable (Claims/Exposure) of a customer.

The approach consists of a detection phase for under-performing data slices and a re-calibration phase for re-training and increasing the performance gaps between the slices. The underlying model used is a XGBoost Regressor.

2 Solution Description

Our solution is based on a generalized pipeline that is model-agnostic and data driven which re-trains the existing baseline model. The detection phase for finding weaknesses in the ML model is based on IBM's FreaAI method using Highest Prior Density (HPD) or Decision Trees. We chose to implement the Decision Tree since it allowed bivariate data slices with 2nd-order interactions. In addition, we implemented the FreaAI algorithm using regression decision trees which would be the natural choice when trying to optimize MSE metrics as these trees use MSE as the splitting criteria. In this implementation the FreaAI identified segments of data where the MSE, after splitting to leafs, was higher than a certain threshold, which in our case was the MSE of the baseline's result. Ultimately we used a classification decision trees in the pipeline due to deadline constraints. For the calibration phase we tried a number of approaches:

- Removing under-performing data from the training set. This could potentially remove noise, outliers and segments of data that decrease the overall model performance.
- Splitting to two Ad-hoc models: re-training two base models independently for each of the selected data slices. This solution will potentially increase the performance achieved on those data sets.
- Synthetic generation of additional data for the underperforming data slice. This was done using a generative adversarial network (CWGAN-GP). This could improve the model performance on the subgroup.

Ultimately Removing underperforming training data yielded the best test results and was implemented into the pipeline. Short description of the solution components:

- Run baseline.
- Binary thresholding for under performing data samples (prediction exceeds true target threshold).
- Perform data pre-processing.

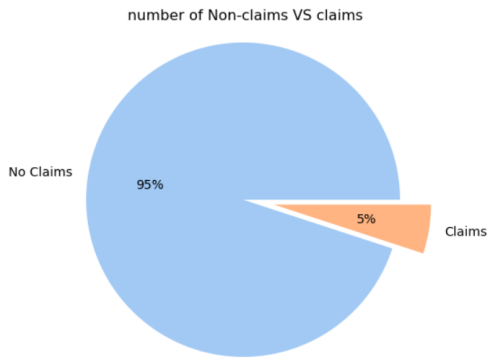
- Use FreaAI based model for partitioning data in feature space into 'low quality'/'high quality' data.
- Retrain base models for each subgroup.
- Final output:
  - New model equipped for handling new data. The new model is conditioned to partition the data and predict accordingly.
  - Interpretability of the model - if requested the customer can seek the exact subgroup in feature space that is partitioned as underperforming. This can be used as a business tool for focusing on specific market sections and avoid treading in dangerous waters.

3 Data

"Boston Housing" - The data was drawn from the Boston Standard Metropolitan Statistical Area (SMSA) in 1970. The dataset holds 506 records of house prices, each has 13 features.

"French Motor Claims" - In the dataset freMTPL2freq risk features and claim numbers were collected for 677,991 motor third-part liability policies (observed in a year), each has 9 explanatory features.

During development both datasets were used to create a baseline to test our proposed solutions model and measure the change in accuracy that was achieved. The solution supports tabular datasets with both categorical and continuous features. The Boston dataset mainly stresses the model in the domain of limited data (only 506 samples), it consists of only numerical data types. The French dataset consists of both numerical and categorical features. It stresses the solution in the domain of imbalanced data (from 677991 samples only 34060 filed claims, 5%).

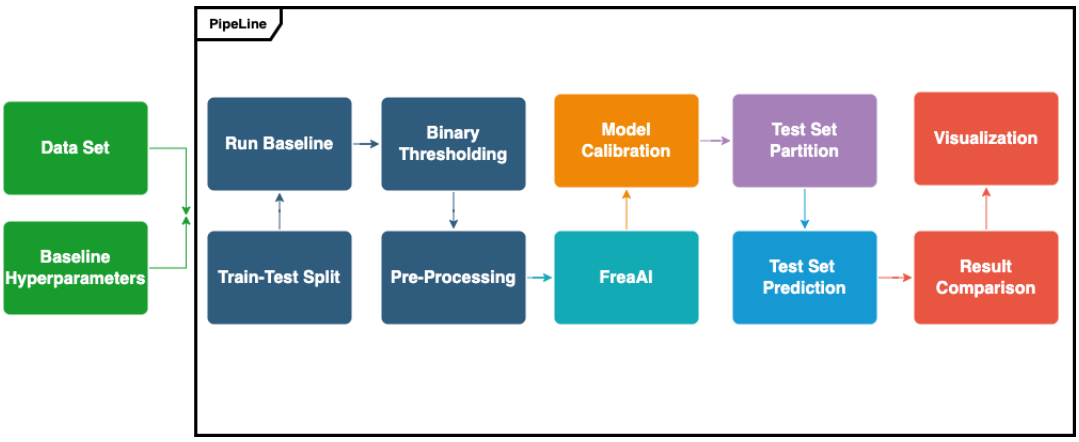


4 Features

In the training phase the solution applies FreaAI Decision Trees that take numerical data only. Both One-hot and ordinal encoding techniques were tested. The test results were similar while one-hot encoding runtime was significantly longer. One-hot encoding enlarges the feature space resulting in both longer runtime and interferes with interpretability. Therefore ordinal encoding was used on categorical features. Note that only the french dataset had categorical features where close categories had a relation, for example 'Area'. This could explain the similar results using both techniques. An additional binary feature is added after running the baseline for marking whether the prediction was within the required threshold. This feature is used as the target value in the FreaAI Decision Tree.

5 Algorithm

- Data Flow



- Algorithm description

  The Pipeline generic approach enables the handling of both datasets. The code is written as a python script. The pipeline receives as inputs the following parameters:

  - Dataset
  - Target column name
  - Split ratio - The split ratio specific to the dataset according to the baseline
  - Baseline params - XGBoost parameters according to the baseline
  - Datapoint cutoff number - Minimum number of samples in a leaf to be considered as a candidate in the FreaAI module
  - Threshold - The threshold used for creating the classification

  1. When running the command to run the pipeline, the name of the desired database should be specified in the CLI, this will prompt the DB load service of the pipeline to download and prepare the relevant database. Then, the pipeline will be initiated with the corresponding database and baseline parameters. It is designed to be easily extended to other databases.
  2. In the first step the pipeline runs the baseline model with the 'baseline params' parameter in order to calculate its result metric for future comparison against the improved model. It uses the threshold parameter in order to classify the baseline predictions as good/bad predictions in the following manner:
     $$\begin{cases} (pred - target) < (1 + target) * threshold : & 1 \\ else : & 0 \end{cases}$$
     The thresholds chosen were 0.18 and 0.015 for the French and Boston dataset respectively.
  3. The pipeline splits the dataset to train and test datasets according to the split ratio parameter.
  4. The train dataset is preprocessed by the data processing unit of the pipeline which transforms categorical features into ordinal features, using scikit-learn's pipeline module.
  5. The FreaAI process, using datapoints cutoff number, is run over the processed train dataset to detect weak data segments.
  6. Based on the FreaAI analysis, both train and test datasets are partitioned to high/low-quality data segments. Two new models are calibrated, one is trained on the low-quality data and the other on the High-quality data.
  7. The high-quality model is used to predict the high-quality test data segment and the low-quality model is used to predict the low-quality test data segments. The combined results are compared to baseline's result.
  8. In addition, the high-quality model is used to predict the entire test dataset, as we wanted to explore the result of the model when the low-quality data segments were removed from the train set. The results are compared to the baseline's results.

9. Finally the pipeline produces visual graphs to illustrate the results.

6 Results

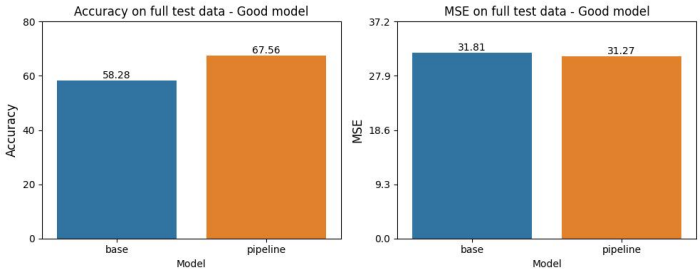
French Results

Model	MSE	MAE	Accuracy	Improvement
Baseline model	31.81	0.52	58%	
High/low models - combined results	31.24	0.51	60%	**MSE:** 1.79% reduction **Quality:** 3.45%
High model alone	31.27	0.49	68%	**MSE:** 1.70% reduction **Quality:** 13.33%

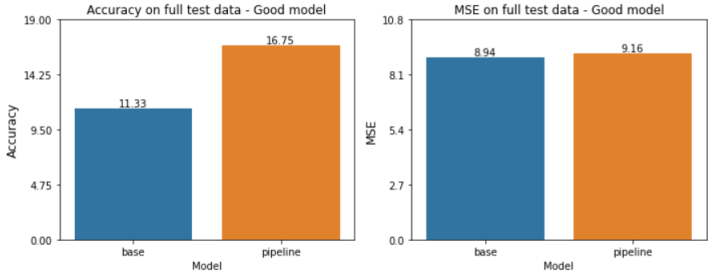
Boston Results

Model	MSE	MAE	Accuracy	Improvement
Baseline model	8.94	2.11	11%	
High/low models - combined results	11.14	2.29	16%	**MSE:** 24.6% increase **Quality:** 47.8%
High model alone	9.16	2.12	17%	**MSE:** 2.46% increase **Quality:** 52.2%

French motor:



Boston housing:



It can be seen that in both cases, the models trained on the high-quality data segments yielded the best results. In the French case, both the combined model and the High-quality data model were able to improve accuracy while reducing the MSE in respect to the baseline. It was expected that the results will be less ideal for the Boston dataset, as it is a very small dataset. The FreaAI is not an optimal approach for small datasets, as it produces very narrow low-quality data segments on which the new model is trained on. Thus it might generate unreliable results.