Model Card

Model Details

* Binary classification model – XGBoost
* Model version v8
* A classifier model focusing on classifying if a pensioner would get CETV transfer or not (A pensioner can take CETV transfer/Triv Comm/retire at retirement age)

Intended Use

* Individual predictions are aggregated by scheme to see if a scheme is either HIGH, MEDIUM, or LOW
* Intended to be used as assumption by other calculations, such as liability calculation
* Not intended to solve every binary classification problem

Special Considerations

* Interpretability is crucial:
  + Use logical models to enhance explainability
  + SHAP
  + PCA is not used
* Because of privacy concerns, the model does not train on personalized information: e.g. financial & location data

Data

* Sources: ACORN, in-house member data
* High dimensional (>1000 features)
* Imbalance (imbalance ratio: 9)
* Features and records removed because of null:

|  |  |
| --- | --- |
| Removed features | * Exit\_date * Exit\_type\_id * Protected\_pension\_age * Transition\_reason |
| Size of features after removed | * 99.33% of original size |
| Size of records after rows removed | * 90.25% of original size |

* Please refer to document xxx for further details

Training

|  |  |
| --- | --- |
| Data split  (train/validation/test) | * Split ratio: 0.8/ 0.0 /0.2 |
| Imbalance handling | * Algo adj: class weighted * Scale\_pos\_weight: 9 |
| Model family | * Logical models:   (e.g. Decision Tree, Random Forest, XGBoost, etc.) |
| Hyperparameter tuning | * Algo: GridSearch * Cross-validation: StratifiedKFold * Evaluation metric: F2 score |

* Params graph

Performance Metrics

* Baseline model: dummy classifier
* F2 score (assume the cost of a false negative outweights a false positive)
* Recall
* PR AUC

Limitations

* The model predicts the likelihood that the member would like to transfer, but this may not always be possible due to the regulatory or other restrictions
* Only focus on age >55, without considering people between 50 – 55
* Pension regulation changes frequently, model drift or data drift are expected in the future

Recommendations

* Acquire more data points to capture not only pensioner’s willingness to transfer, but also capability to do so
* Ensemble final model with neural network
* AutoML & MLflow to facilitate model logging
* Setup procedure and pipeline for model retraining (q.g. sequential learning model/process)

Evaluation results

* Features picked

|  |  |
| --- | --- |
| Feature name | What it represents |
| A |  |
| B |  |
| Sex\_id | Male gender |
| Age\_exact\_0.0 | People with age between x to y |
| Age\_exact\_1.0 | People with age between y to z |

* F2 score of XGBoost ~0.49 vs dummy model of ~0.3
  + F2 score is essentially the F1 score’s variant, merging precision and recall into a single score. However, in F2 score, there’s a more pronounced emphasis on recall compared to the conventional F1 score
* Feature importances

Feature importance of the model, based on 1) how frequent does the feature appears in a tree, and 2) average gain of splits by using a feature

Graph

Average gain of splits shows that exact age of people (55 to 62.9) has a dominant effect, while sex has the second strongest effect, on predictions. The rest has similar effect. This outcome matches with the SHAP values shown in the following graph. Perhaps a better approach is to split the dataset into two: one with age 55 to 62.9 and the other with age 62.9 to 100. Then see how well other features predict the outcome (although the dataset may become too small, and special cares may need to carry out for it)

* SHAP

Shapley values capture the marginal contribution of each feature to the predictions.

Below can be interpreted as: the lower the feature value, the higher likelihood the model will predict 1, and vice versa. And the influence of this feature to prediction is slightly stronger on prediction “0” than predicting “1”

Graph

* PR AUC graph

A graph showing performance of trained model vs benchmark model. A model needs can have a high precision at the cost of lower recall, or vice versa. Therefore, the further a model line towards top right corner, the better the performance of the model it represents as it can have a higher recall at a lower sacrifice of precision

Graph

The XGBoost model clearly is performing better than the dummy classifier, although there is still room to improve as it is nowhere close to the top right corner