Fraud prediction: a implementation of a Observation Undersampling model

Implementation by José P. Barrantes
Model taken from Perols et al. (2017)[1]

[1] Perols, J. L., Bowen, R. M., Zimmermann, C., & Samba, B. (2017). Finding needles in a haystack: Using data analytics to improve fraud prediction. *The Accounting Review*, *92*, 221-245.

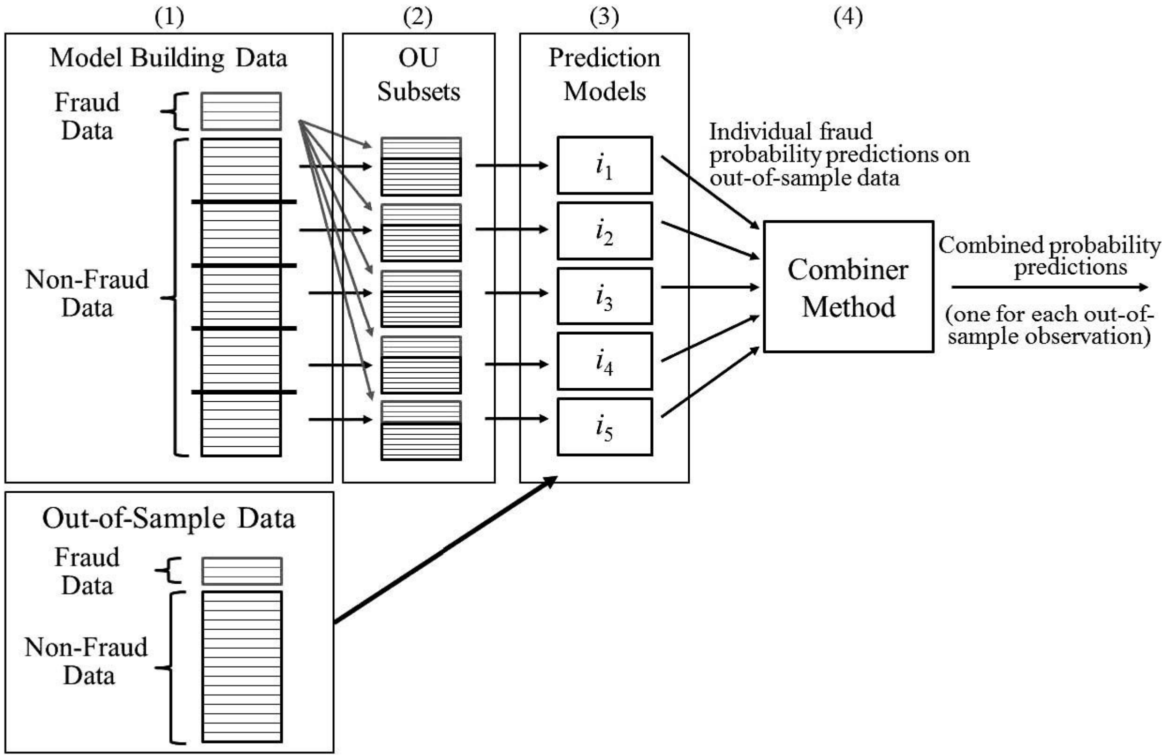
Overview

1. About the model.
2. Implementation.
3. Results, and commentaries.
4. Pros of the model.
5. Conclusions.

About the model: Multi-Subset Observation Undersampling

To address the rarity of fraud examples.

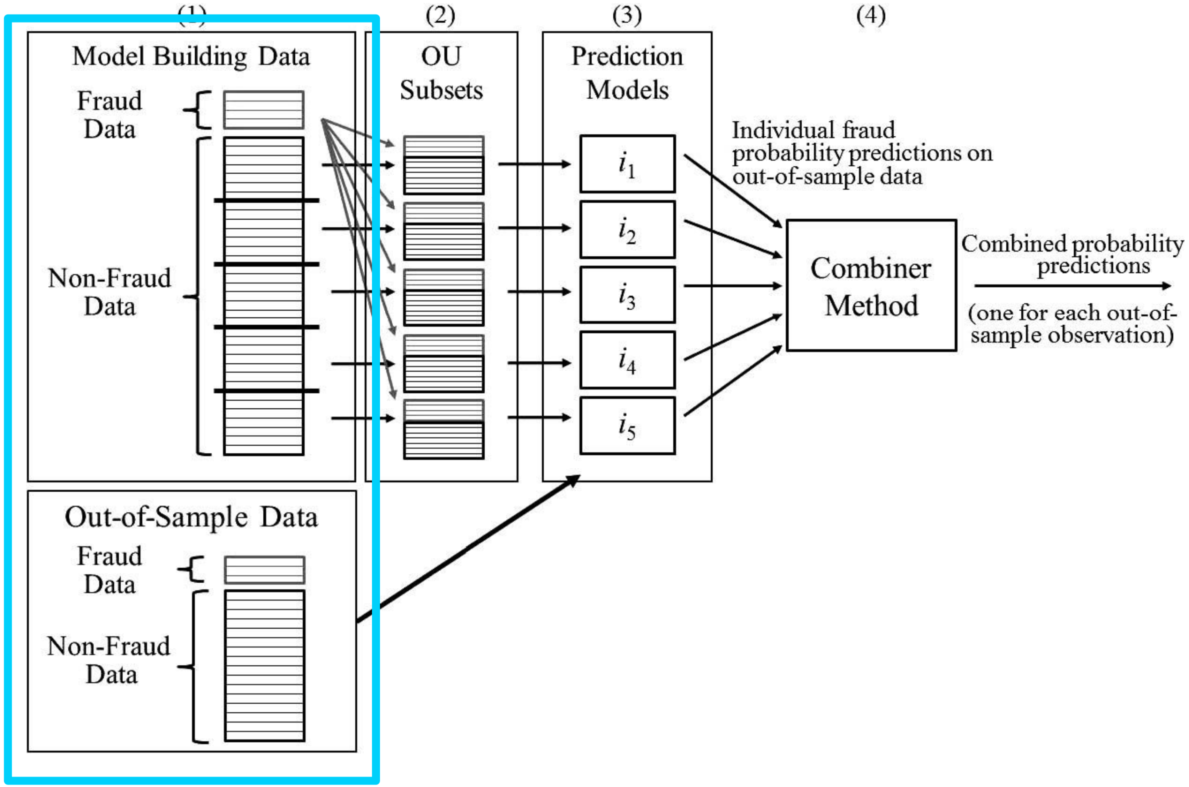
So the model can learn to recognise the fraud cases with limited fraud examples to learn.



About the model: first step

First we partition our data in two sets.

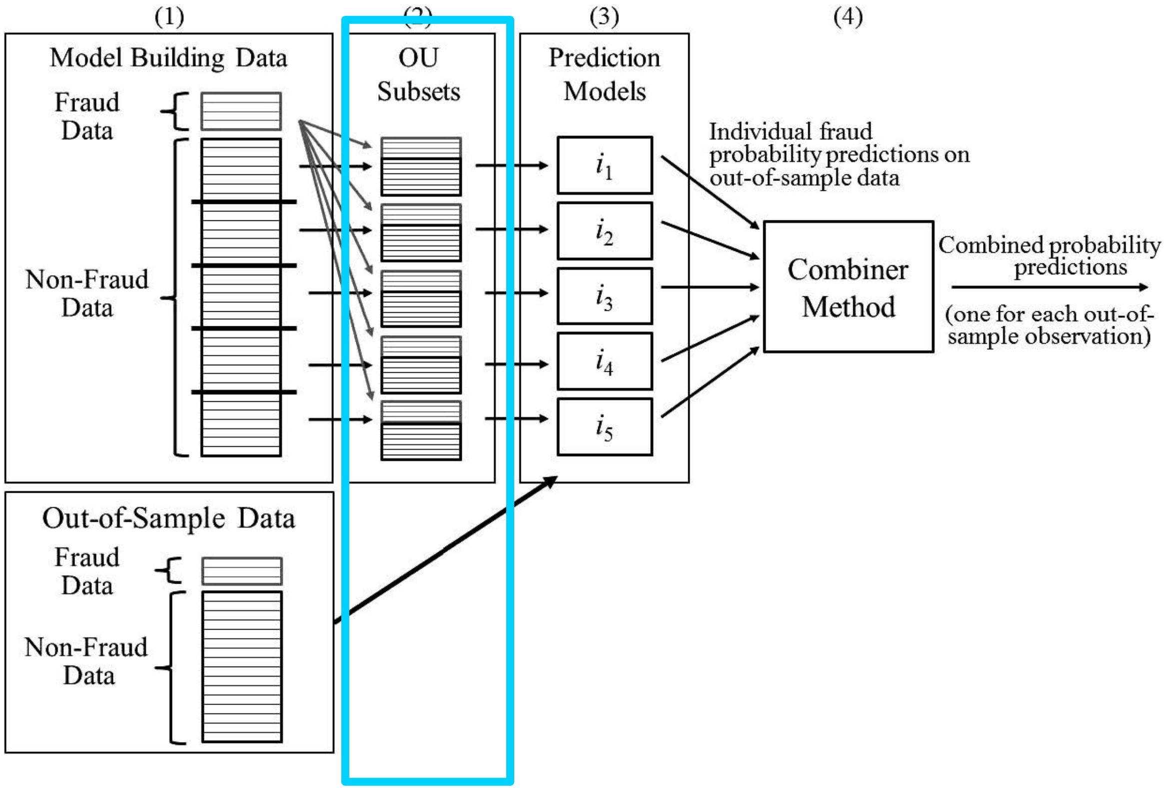
- One with the data to to be used to train the models.
- The other to fine-tune the cutoff (decision boundary) of the ensembled model.



About the model: OU Subsets

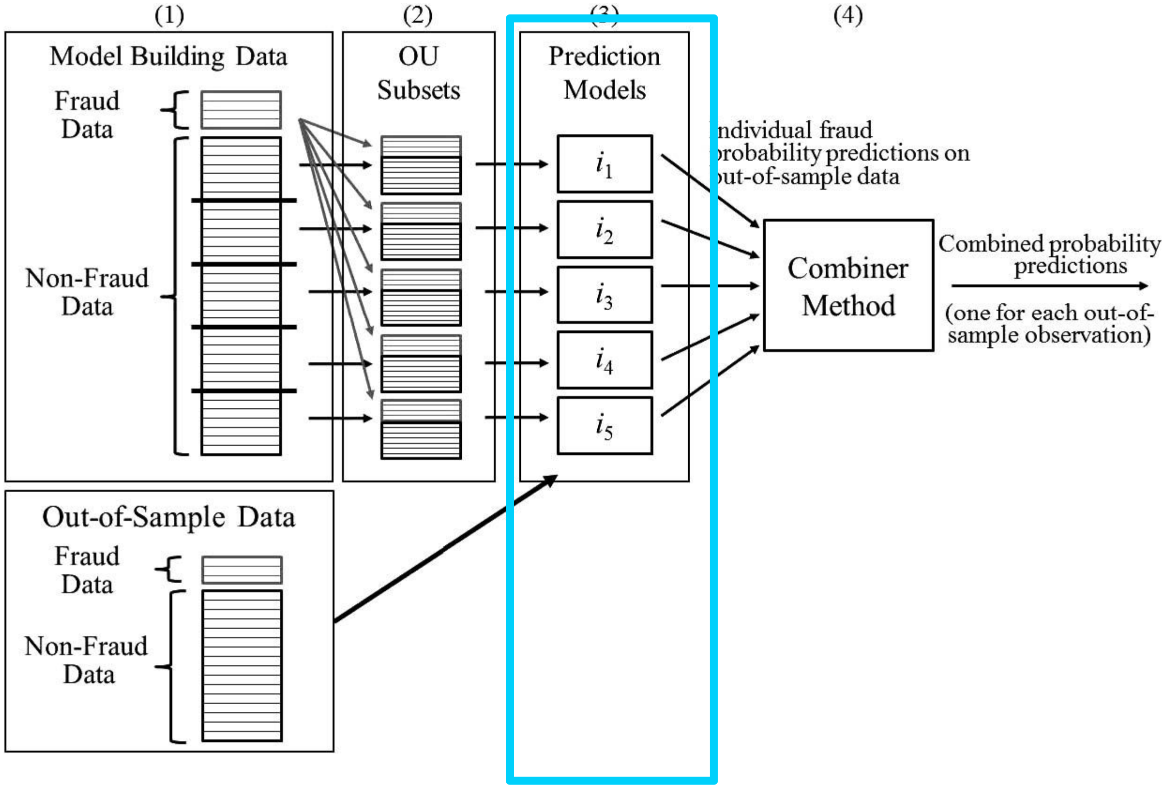
Then, we split the training data set into several OU subsets.

Each one will contain a copy of **all** the instances of the minority class (frauds) available in the training set, and several random instances of the majority class.



About the model: train the models

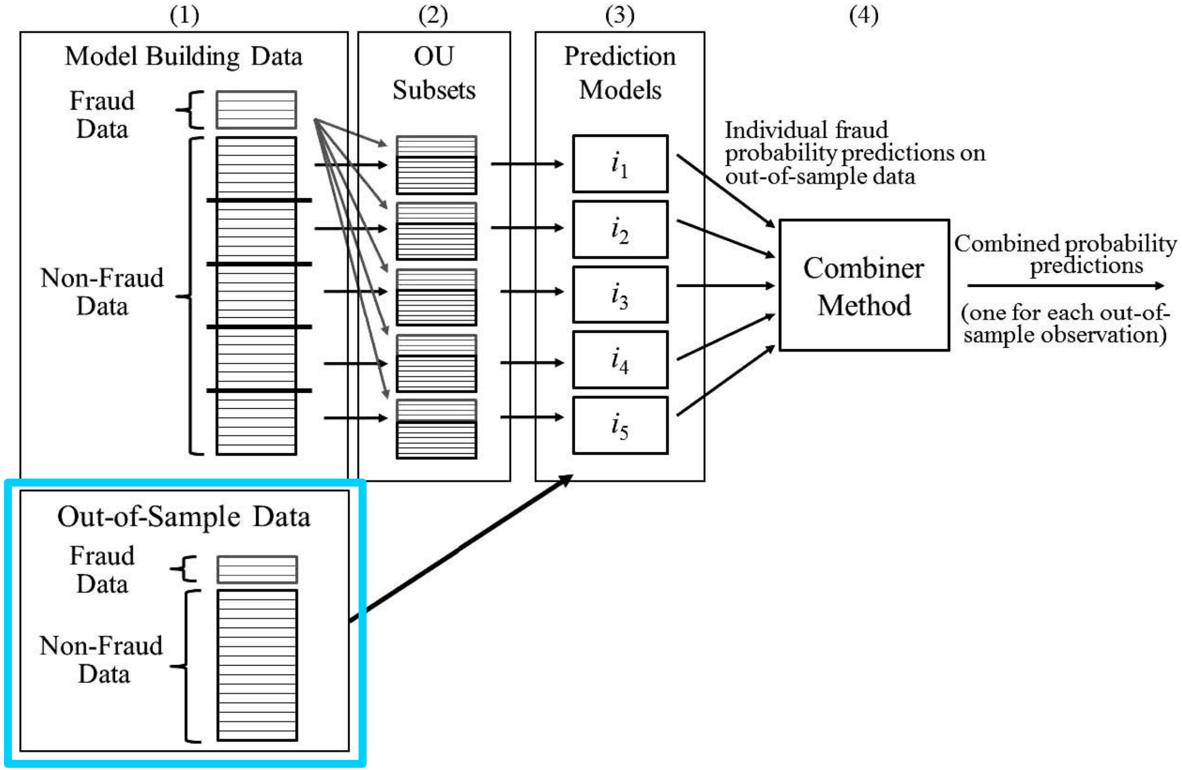
With each OU subset, we train a probability prediction model.



About the model: run models

We use the dataset we want to classify into frauds and true transactions.

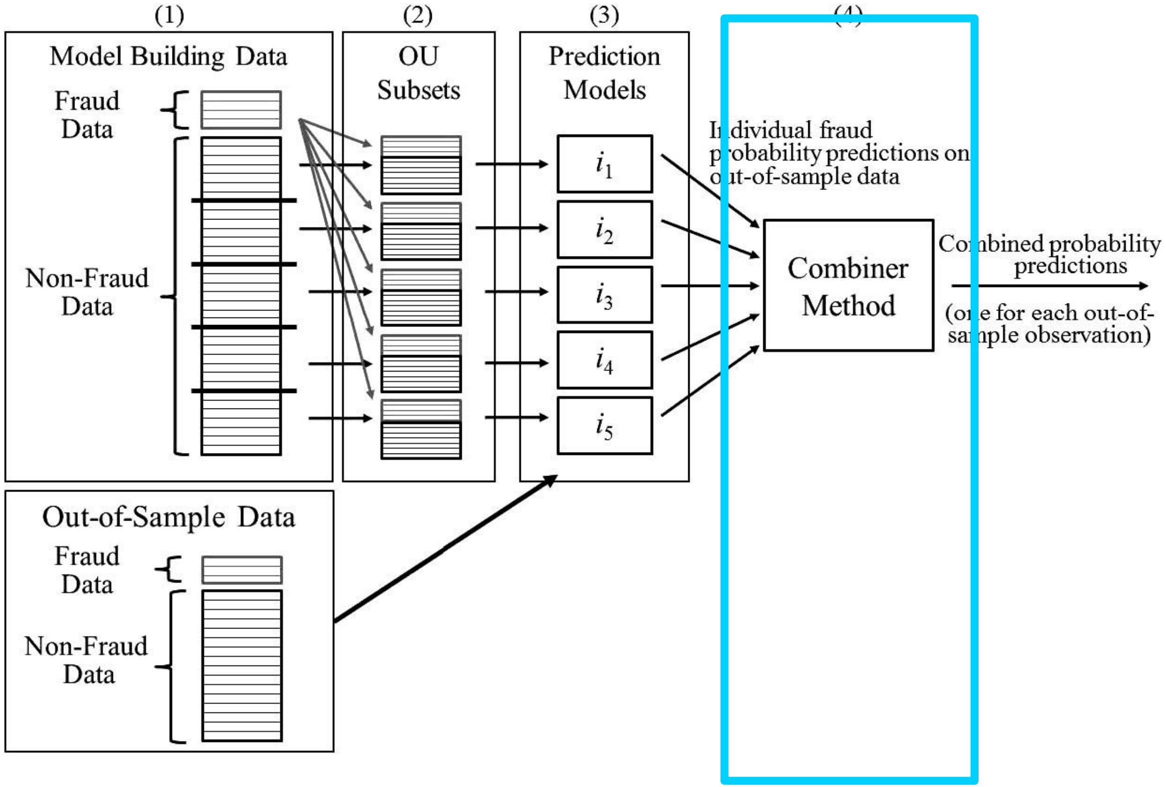
To predict a probability for each instance, in every model.



About the model: combine probabilities

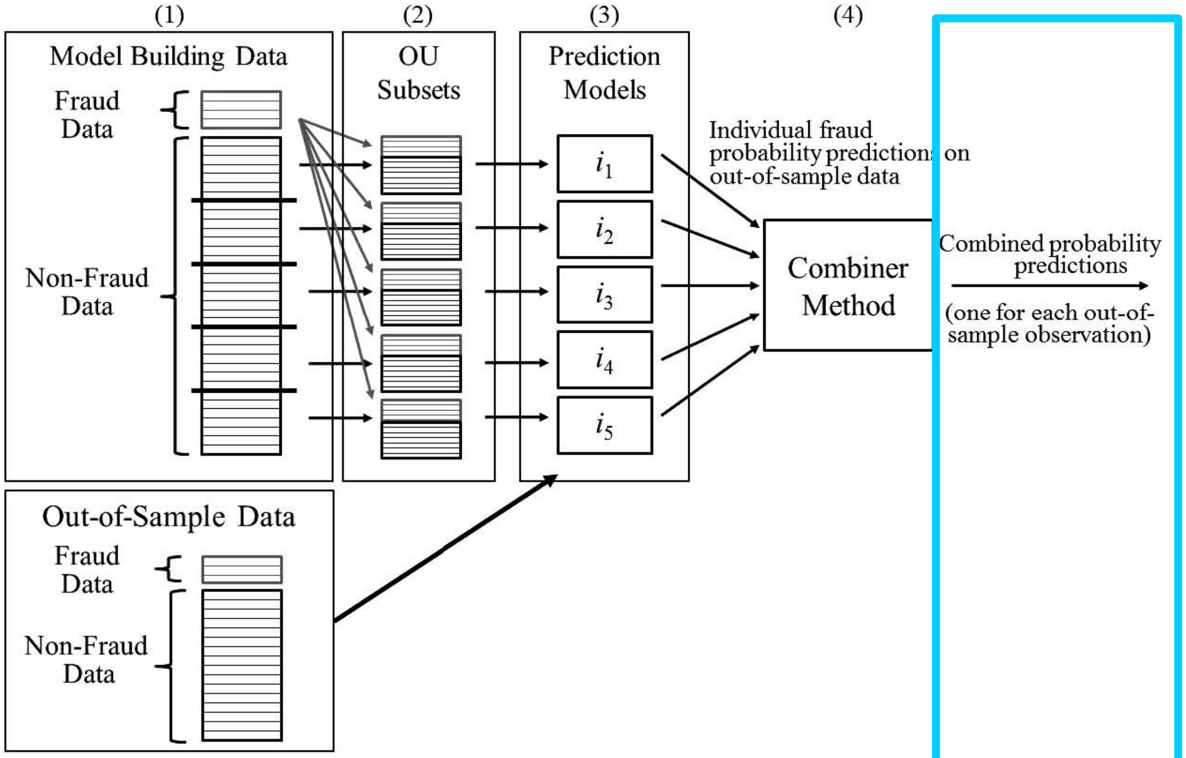
For each instance, we average the probability who renders every model.

This is the combiner method.



About the model: prediction

We make a decision with the ensembled probability.



About the model: cutoff

Decision threshold must be fine-tuned with validation unseen (by the models) data.

The threshold who renders the lesser error is selected.

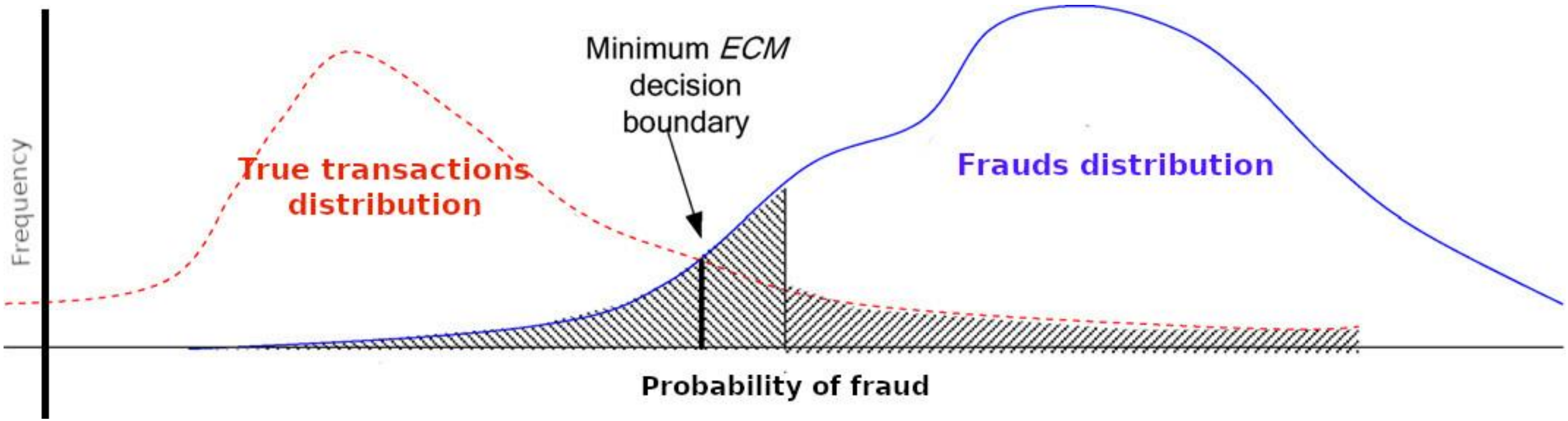


Figure modified from:
Huang, S. H., Mo, D., Meller, J., & Wagner, M. (2012). Identifying a small set of marker genes using minimum expected cost of misclassification. *Artificial intelligence in medicine*, *55*, 51-59.

About the implementation

- One Jupyter notebook, in which I detail all the steps.
- Two Python files containing the classes to manage the ensembled models.
- Python packages:
  - Pandas.
  - NumPy.
  - Scikit-Learn.
  - Matplotlib.
  - Pandarallel
- Dataset: Machine Learning Group - ULB. (2018, March). **Credit Card Fraud Detection: Anonymized credit card transactions labeled as fraudulent or genuine**, Versión 3. Rescatado el 15 de octubre del 2019 de https://www.kaggle.com/mlg-ulb/creditcardfraud/

Performance measure: Expected Cost of Misclassification (ECM)

$$\textbf{ECM} = C^{FN} \times P(\text{Fraud}) \times n^{FN} / n^{P} + C^{FP} \times P(\text{Non-Fraud}) \times n^{FP} / n^{N}$$

C^{FN}: cost of false positive.

C^{FP}: costo of false negative.

P(Fraud): prior probability of fraud.

n^{FN}: number of false negatives.

n^{FP}: number of false positives.

n^{P}: number of positives, in out-sample set.

n^{N}: number of negatives, in out-sample set.

P(Non-Fraud): prior probability of non-fraud.

Performance measure: Expected Cost of Misclassification (ECM)

$$\textbf{ECM} = C^{FN} \times P(\text{Fraud}) \times n^{FN} / n^{P} + C^{FP} \times P(\text{Non-Fraud}) \times n^{FP} / n^{N}$$

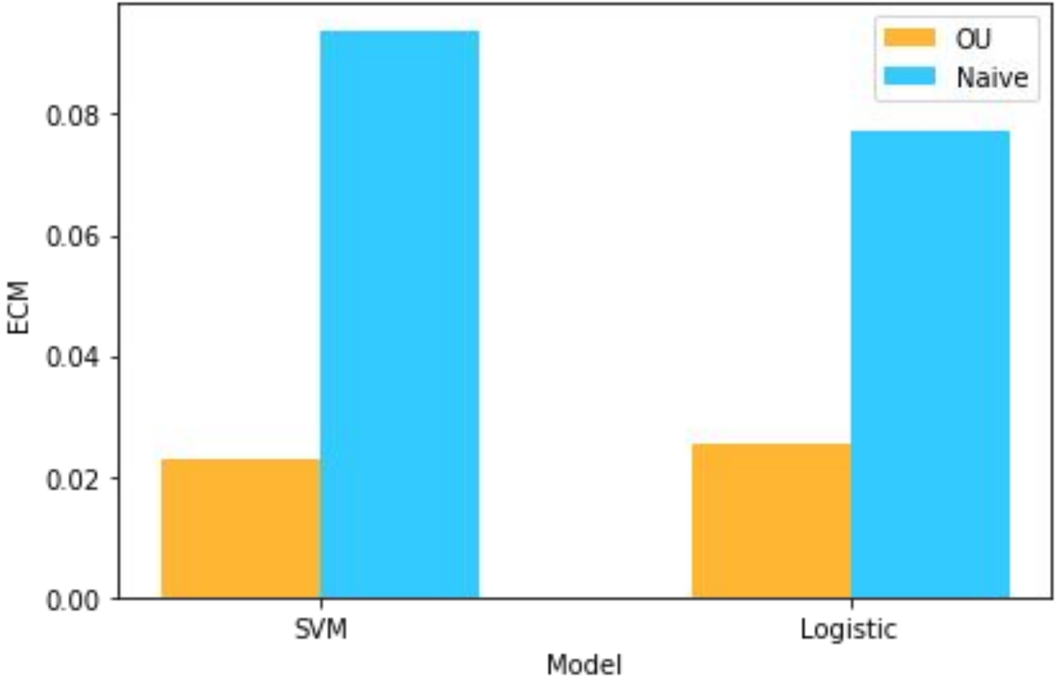
A Bayesian approach. The value of the costs, and priors are obtained through literature.

Results and commentaries: ECM

Lower is better.

The ECM is almost **three times lower** in the OU models.

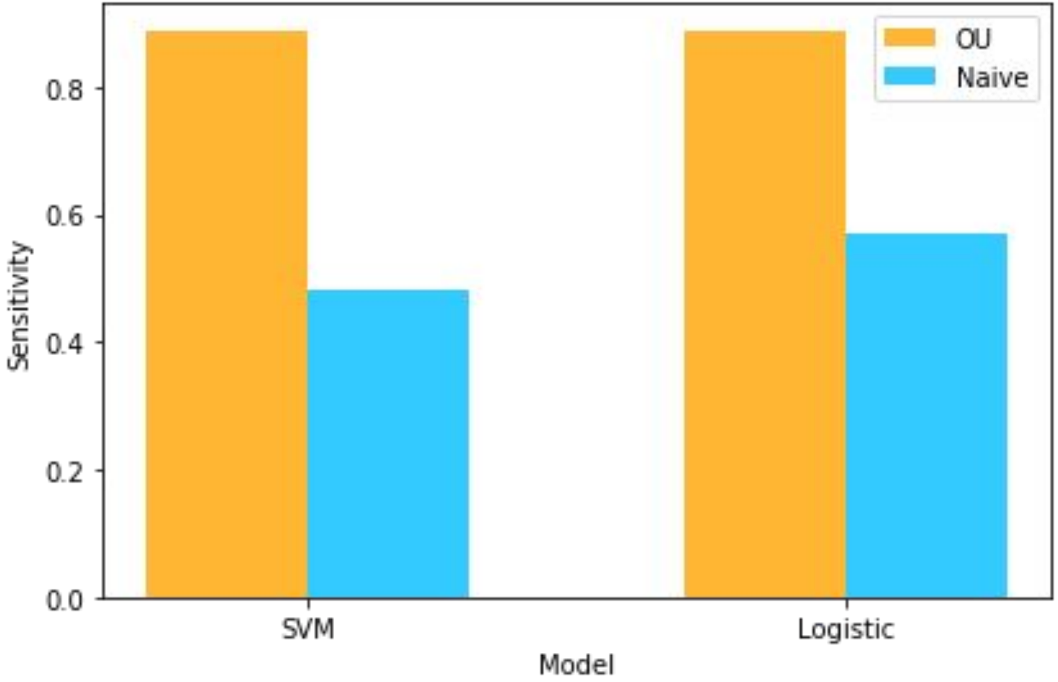
The superior performance in the OU models is evident.



Results and commentaries: Sensitivity

Capacity of detecting fraud.

With the **sensitivity**, we can see that the OU models are more capable of correctly classify the fraud cases.



Results and commentaries: general overview

Model	Type	False Negatives	False Positives	True Positives	Sensitivity	ECM
SVMs	OU	11.2	119	86.8	0.886	0.023
Logistics	OU	11	285.7	87	0.888	0.025
SVMs	Naive	51	7	47	0.48	0.094
Logistics	Naive	42	12	56	0.571	0.077

Pros of the Observation Undersampling model

- Renders a good performances under class-unbalance conditions.
- Models like SVM are very computing expensive, the smaller subsets make possible use big data to train several models.
  - One big model could be impossible to train, several small ones is very feasible.
- It addresses the rarity of fraud instances problem.
- Results could be improved with causal thinking and feature engineering.

Conclusions

- The Observation Undersampling Model is a reliable approach to solve problems with unbalanced classes.
- This methodology can be also an alternative to process big data when the computing power is limited.