Big data and its applications

Project

Professor G. Uzbelger – Spring Sem. 2017-2018

Thomas Bourany[1,2]

[1]UPMC-Sorbonne University – Math-Model (LJLL)

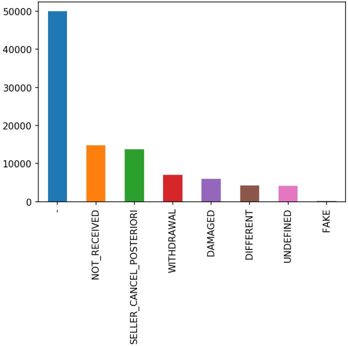
[2]Certificat Big Data

Defense, July, 2nd

Introduction – Challenges ENS

- Classification challenge
  - Predict if a transaction (good purchased) on PriceMinister platform is subject to a claim.
  - Multiclass classif for purpose of the claim
  - Using descriptive data on buyer, seller and transaction (good)
- Metrics : ROC AUC
  - Receiver operating characteristic, Area under curve.
  - True Pos. wrt. False Pos. :
    Power $(1-\beta)$ in terms of 1st class error (α)
  - Weighted (multiclasses)

Introduction – Data cleaning



- Dep. variable : Claims
  'OK' (≈ 50%), 'WITHDRAWAL', 'SELLER CANCEL POSTERIORI', 'NOT RECEIVED', 'DIFFERENT', 'UNDEFINED', 'DAMAGED', 'FAKE'

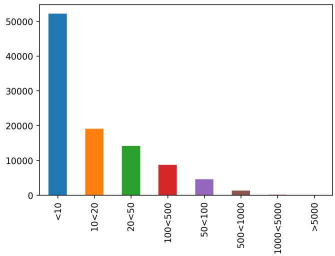
⇒ Unbalanced data.

- Indep. variables, different types :
  - Prices and potential 'numerical' variables (e.g. count & score)
  - Type and family of goods (strict categorical data)
  - Buyers and sellers locations
  - Dates, time & others (easy) variables

Data cleaning – 1^{st} : Pseudo numerical variables

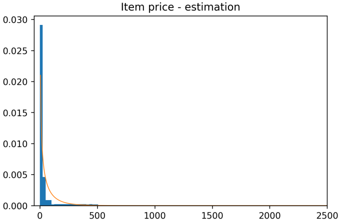
- Transform (string) cat. variable into numerical variable

Item price of the transaction



$\Rightarrow$

Estimation



Data cleaning – 1^{st} : Pseudo numerical variables

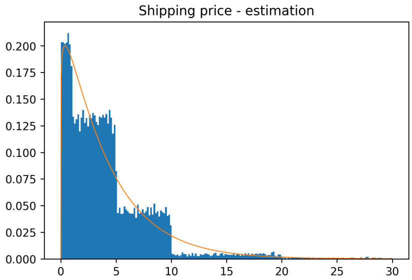
- Transformation steps :
  1. J categories of prices, consider histogram values : n_j observations for the category between x_j and x_{j+1} (with $j \in \{1, \ldots, J\}$)
  2. Simulate a subsample of n_j observations uniformly distributed in $[x_j ; x_{j+1}]$
  3. Concatenate a sample with these $n = \sum_j n_j$ observations
  4. Estimate the shape & scale parameters of a gamma distributions on this n-observations simulated sample (gamma.fit).
  5. Compute conditional expectation $\mu_j = \mathbb{E}(X | x_j < X < x_{j+1})$, for X a r.v. following a gamma distrib. with param. estimated in previous step [using IPP and numerical integration].
  6. Assign the num. value μ_j for data in category " $[x_j ; x_{j+1}]$".
- Example :

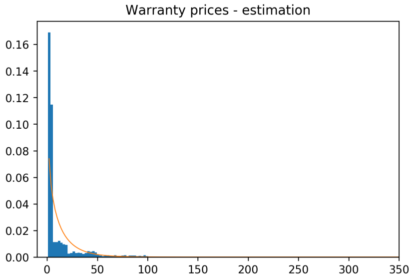
$$\{[0, 10]; [10, 20]; [20, 50]; [50, 100], [100, 500]; [500, 1000]; [1000, 5000]; [5000; 6000]\}$$

$$\Rightarrow \quad \{3; 15; 33; 72; 188; 607; 1213; 5026\}$$

Data cleaning – 1^{st} : Pseudo numerical variables

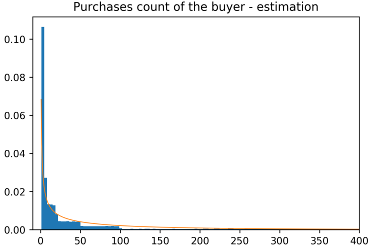
- Same method applied to Shipping price and Warranty price.

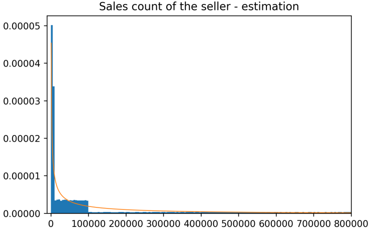




Data cleaning – 1^{st} : Pseudo numerical variables

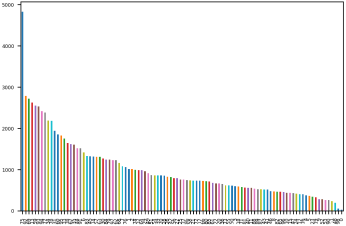
- Same method applied to count variables : Purchase count (buyer) and sales count (seller).

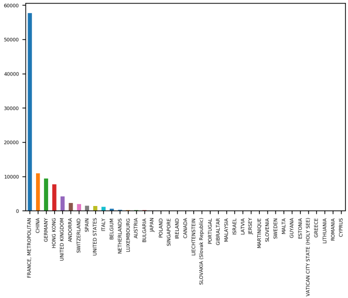




Data cleaning – 2^{nd} : spatial variables

- Three spatial variables :
  - Departement (buyer)
  - Departement (seller)
  - Country (seller)

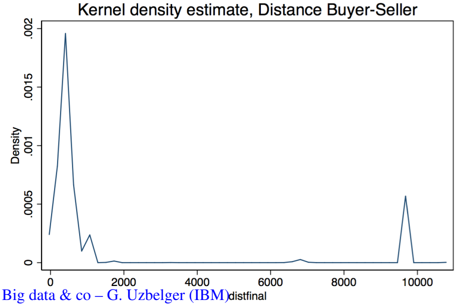




- Distance buyer-seller may matter for the transaction
  - Similar mechanism to distance/transaction cost in gravity models in international economics.

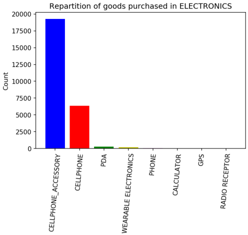
Data cleaning – 2^{nd} : spatial variables

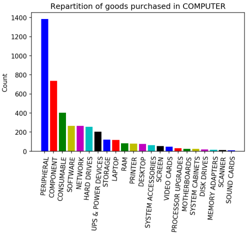
- How to compute the (geodesic) distance :
  - Cross country distance : constructed by Head & Mayer (2002) and Mayer & Zignago (2011)
  - Dep. location : Prefecture latitude and longitude data from INSEE.
  - Matching (dep. code & cities) and cleaning (depts. that don't exist).
  - Computation of distance via ad-hoc formula $dist = \arccos(\sin(\text{lat}_a)\sin(\text{lat}_b) + \cos(\text{lat}_a)\cos(\text{lat}_b)\cos(\text{long}_b - \text{long}_a))R_{earth}$
- Results :

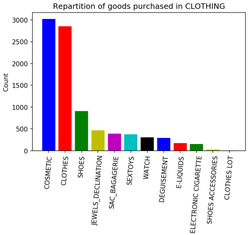


Data cleaning – 3^{rd} : Categorical variables

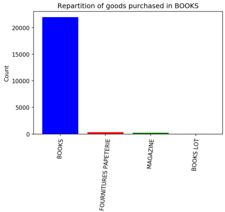
- Usual treatment for categorical variables :
- Binarization :
    - If a variable has K potential categories
    - Create $K - 1$ new dummy variables : 1 for a type, 0 if it is the 'standard' (most frequent) category.
    - Choice of the reference often non-ambiguous (one type is often very frequent).

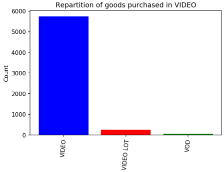


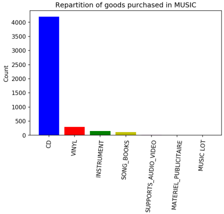


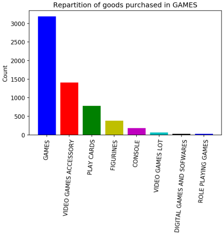


Data cleaning – 3rd : Categorical variables







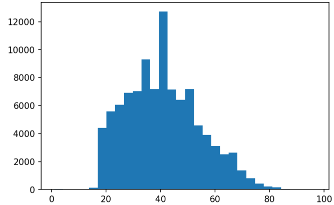


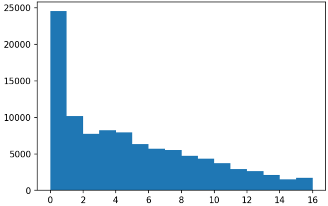
- ’Typical’ example of goods :
  - Cellphone accessories (typically a smartphone protection) purchased at a low cost (less than 10 euro, cf. estimation above), from China or from retailers in France.
  - Books, DVDs or CDs (again at a low price) from French editors or French retailers.

Data cleaning – 4^{th} : other client data

- Age and Time from registration :
  - After cleaning :

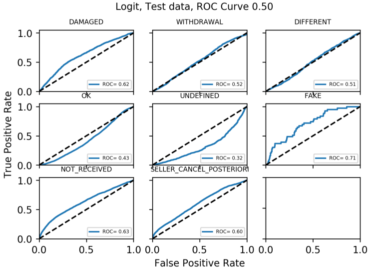
Buyer age (left) and Time from registration (right) in years





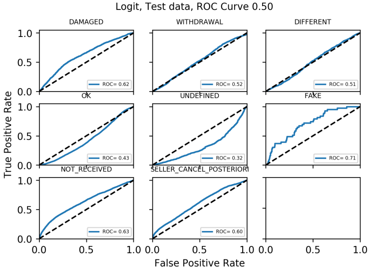
Classification algorithms – Regression and NN

- Multinomial Logit Regressions
  - Treatment heterogeneous depending on the class :
    - Fake/ Not received /Damaged $\Rightarrow$ good classif.
    - Not so good for others (OK !)

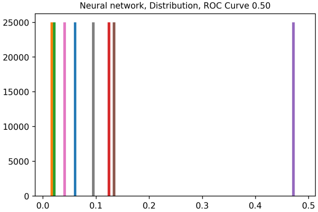


Classification algorithms – Regression and NN

- Multinomial Logit Regressions
  - Treatment heterogeneous depending on the class :
    - Fake/ Not received /Damaged $\Rightarrow$ good classif.
    - Not so good for others (OK !)



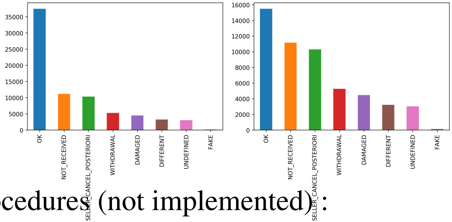
- Neural Networks :
  - Completely unable to manage the unbalanced data
  - Despite change in hyperparameters, assign the same proba value for all



Classification algorithms – Rebalancing

- Different methods to 'help' the algos to perform better on this unbalanced data set :
- Drop (randomly) 'OK' label data for the training dataset :
  - Increase performance (Roc Auc) by 2%.

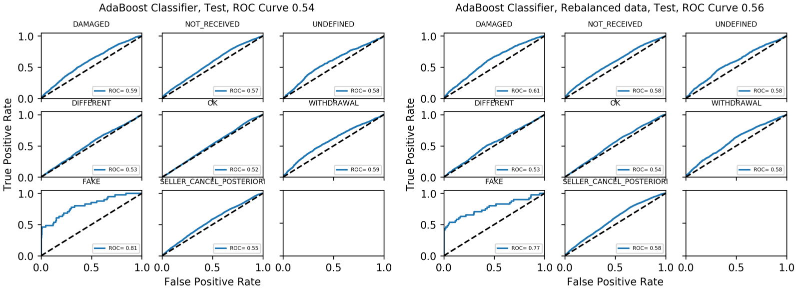
Class repartition - Unbalanced (LHS), Rebalanced (RHS)



- Two-steps procedures (not implemented) :
  - Binary classification : OK vs. Claim (50/50 : balanced data !)
  - Multiclass for type of claim issue :
    more balanced data for 7 other labels.

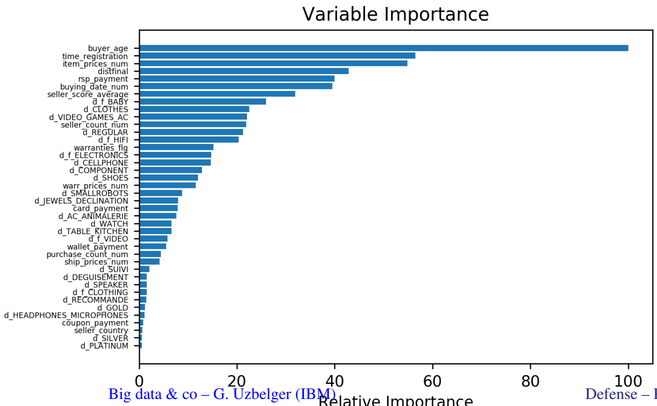
Classification algorithms – Adaboost

- Adaboost :
  - Classifier as a linear combinaison of weak learners (simple decision tree).
  - Recursive algo (description in report).



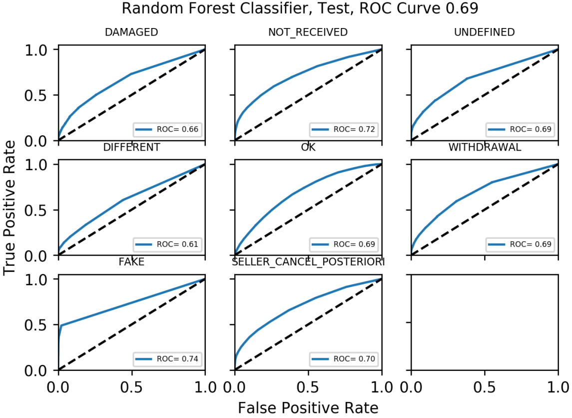
Classification algorithms – Adaboost

- Adaboost :
  - Classifier as a linear combinaison of weak learners (simple decision tree).
  - Recursive algo (description in report).



Classification algorithms – Random Forest

- Most efficient algorithm
  - Better on all classes



Assessment and conclusion

TABLE – Assessment of the different methods

Algo	Specification	Accuracy	*Roc Auc per class* Dam	Diff	Fake	Not Rec	OK	Cancel	Undef	Withdraw	*Weighted Roc Auc*
Logit	Train data	0.497	0.625	0.563	0.848	0.641	0.611	0.603	0.664	0.653	0.626
Logit	Test data	0.504	0.631	0.578	0.837	0.627	0.605	0.603	0.633	0.664	0.621
Logit	Rebalanced train	0.315	0.604	0.520	0.829	0.639	0.582	0.616	0.659	0.638	0.617
Logit	Rebalanced test	0.466	0.633	0.539	0.839	0.631	0.580	0.599	0.651	0.663	0.615
Neural network	4 layers, 20 epochs	0.146	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
Adaboost	200 estimators	0.325	0.590	0.527	0.808	0.566	0.519	0.554	0.580	0.586	0.543
Adaboost	200 est., rebal. train	0.263	0.605	0.531	0.774	0.577	0.537	0.577	0.580	0.585	0.557
Random forest	20 estimators	0.361	0.664	0.607	0.736	0.719	0.688	0.704	0.690	0.688	0.690