Shortfall in Tax Revenue: Evaluating the Social Security Contribution Fraud

Denisa BANULESCU-RADU[a]
Sylvain BENOIT[b] Christophe HURLIN[a]

[a] Université d'Orléans, Laboratoire d'Economie d'Orléans

[b] Université Paris Dauphine - PSL

PSL Intensive Week AI for Economics and Finance

Public policy issue

Controlling the risks of social and fiscal fraud and combating illegal work are **important problems for social justice and economic efficiency.**

- **Government expectations:**
  1. Social cohesion; Impact on social security resources by reporting estimates to the *Haut Conseil pour le Financement de la Protection Sociale (HCFiPS)*.
  2. Creation of the *Conseil d'évaluation des fraudes* by Thomas Cazenave (Minister for Public Accounts), October 10, 2023.
- **Media coverage:**
  1. Le Monde, 2022/12/27: Quel est le coût de la fraude sociale ?; 2023/04/19 Contre la fraude fiscale et sociale, le gouvernement promet de durcir la lutte.
  2. Fondation iFRAP, 2022/12/20: Fraude aux cotisations sociales : une estimation à 10 milliards d'euros.
  3. Alternatives Economiques, 2021/12/30: La fraude fiscale écrase la fraude sociale.

Both a definition and an estimation issue

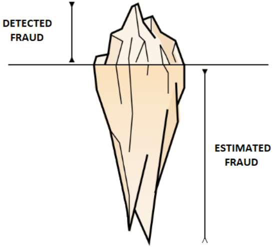
Lack of reliable assessment of the extent of the phenomenon.

Difficulty in evaluating a phenomenon that is by nature occult.

Both a definition and an estimation issue

Lack of reliable assessment of the extent of the phenomenon.

Difficulty in evaluating a phenomenon that is by nature occult.



- **A plurality of estimation methods:**
  1. indirect (macroeconomics).
  2. direct (microeconomics).

Empirical evaluation by the MSA

Communication to the HCFiPS

Estimation in million euros	2014	2015	2016
Detected Fraud	11	15	9
Estimated Fraud	180	170	170
STax / contributions	1,5%	1,5%	1,4%

- Scope based on the accounting control.
- Results similar to the others regimes.

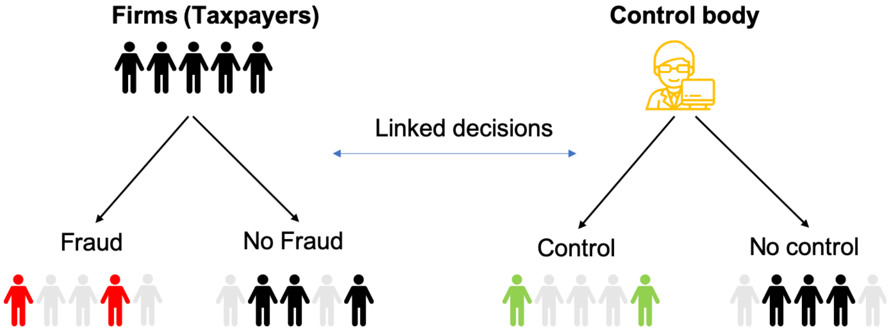
Our paper

1. We proposed a **model on the conditional distribution** of STax.
2. From this DGP, we derived explicit formulas for the two **first moments** of the aggregated STax, validated by simulations.
3. We proposed a **maximum likelihood estimation method** which makes it possible to estimate in a convergent way all the parameters of the model, including the correlations.
4. We showed that the Heckman approach (Mills ratio) is not valid in this context (see Discussion in Appendix), because it does not allow correlations to be estimated, and it is not suitable in the case of a non-linear model.
5. **STax estimation on real data**.

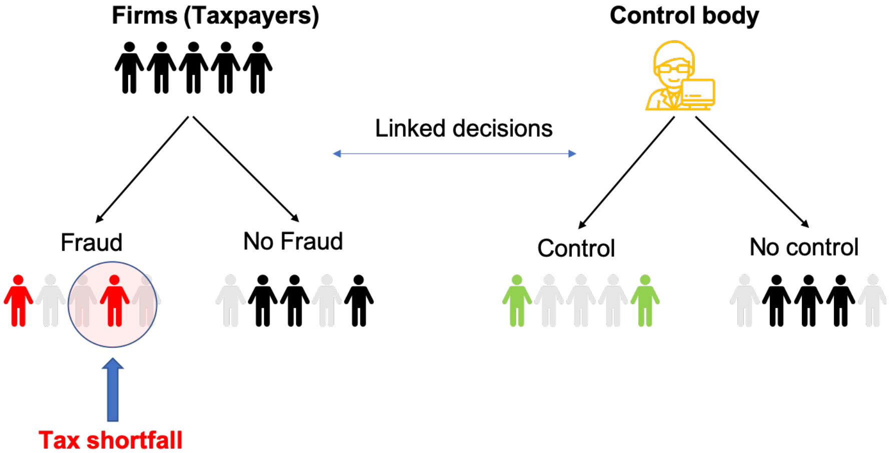
Outline

1. Introduction
2. Methodology
   - Definitions
   - DGP
3. Ensemble Methods
   - Setting
   - General principles
4. Bagging and Random Forests
   - Bagging
   - Random Forests
5. Semi-supervised learning

What is the shortfall in tax revenue?



What is the shortfall in tax revenue?



Firms (Taxpayers)

Control body

Linked decisions

Fraud

No Fraud

Control

No control

Tax shortfall

How to define it?

Statistical definition of the tax shortfall

The variable $STax_i$ is a **random variable**, and the aggregate $STax$ as well.

Our purpose is to determine the first two **theoretical moments** of its distribution, i.e. its expectation $\mathbb{E}\left(STax_i\right)$ and its variance $\mathbb{V}\left(STax_i\right)$.

How to fit it?

Since $STax_i$ is latent, we assume a parametric distribution:

1. We set a model, aka a **data generator process (DGP)**, on the **conditionnal distribution** of the $STax_i$ variable.
2. Based on this DGP, we get **theoretical formulas** for the **first two moments**:

   $$\mathbb{E}\left(STax_i \middle| \mathbf{X}_i = \mathbf{x}_i \right) = f\left(\mathbf{x}_i;\theta\right) \qquad \mathbb{V}\left(STax_i \middle| \mathbf{X}_i = \mathbf{x}_i \right) = g\left(\mathbf{x}_i;\theta\right)$$

   where $\mathbf{X}_i$ is a set of explanatory variables.
3. We **estimate** the model parameters (θ) to obtain a consistent estimator for the first two moments, such that $f(\mathbf{x}_i;\widehat{\theta})$ and $g(\mathbf{x}_i;\widehat{\theta})$.

Causes of errors in STax measurement

1. The postulated model on **the conditional distribution of $STax$ is not well-specified** (variables choice, wrong hypothesis, non-linear effect...), thus the $STax$ theoretical formula is wrong.

2. The **theoretical formulas for the conditional moments is not well-specified**,
Example: we consider $STax = f(X; \theta)$*, but the formula* $f(.)$ *is wrong.*

3. The **method for estimating model parameters is not adequate and does not lead to consistent estimators**,
i.e. $\widehat{\theta}$ is biais and/or not convergent.

DGP Description

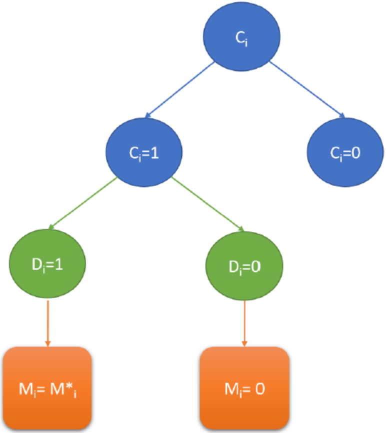


Figure: DGP schematic description

Control Equation

Control

Let C_i be the dummy control variable indicating whether the firm i is controlled $(C_i = 1)$ or not $(C_i = 0)$, such that:

$$C_i = \begin{cases} 1 & \text{if } C_i^* = \mathbf{X}_{c,i}\beta_c + \varepsilon_{c,i} > 0 \\ 0 & \text{otherwise} \end{cases} \qquad \forall i = 1, \ldots, n$$

where C_i^* is a latent variable, β_c a vector with k_c parameters and $\varepsilon_{c,i}$ an i.i.d. error term such that $\mathbb{E}\left(\varepsilon_{c,i}\right) = 0$ et $\mathbb{V}\left(\varepsilon_{c,i}\right) = \sigma_c^2$.

Detection equation

The fraud

Let $\widetilde{D}_i$ be the dummy detection variable indicating whether the firm i commits fraud ($\widetilde{D}_i = 1$) or not ($\widetilde{D}_i = 0$), such that:

$$\widetilde{D}_i = \begin{cases} 1 & \text{if } D_i^* = \mathbf{X}_{d,i}\beta_d + \varepsilon_{d,i} > 0 \\ 0 & \text{otherwise} \end{cases} \qquad \forall i = 1, \ldots, n$$

where D_i^* is a latent variable, β_d a vector with k_d parameters and $\varepsilon_{d,i}$ an i.i.d. error term such that $\mathbb{E}\left(\varepsilon_{d,i}\right) = 0$ and $\mathbb{V}\left(\varepsilon_{d,i}\right) = \sigma_d^2$.

Tax adjustment equation

Potential amount of fraud

Let M_i^* be the latent variable indicating the potential amount in euros of the tax adjustment for firm i such that:

$$M_i^* = \begin{cases} \mathbf{X}_{m,i}\beta_m + \varepsilon_{m,i} & \text{if } \widetilde{D}_i = 1 \\ 0 & \text{otherwise} \end{cases} \quad \forall i = 1, \ldots, n,$$

where β_m is a vector of k_m parameters, and the error term $\varepsilon_{m,i}$ satisfies $\mathbb{E}\left(\varepsilon_{m,i}\right) = 0$ and $\mathbb{V}\left(\varepsilon_{m,i}\right) = \sigma_m^2$.

Tax adjustment equation

The potential amount of fraud is only observable for firms which have been (i) audited by the inspection authority, and (ii) reassessed following the discovery of fraud.

Amount of fraud

We note M_i the amount of the tax adjustment actually observed such that:

$$M_i = \begin{cases} \mathbf{X}_{m,i}\beta_m + \varepsilon_{m,i} & \text{if } C_i = 1 \text{ et } \widetilde{D}_i = 1 \\ 0 & \text{otherwise} \end{cases} \qquad \forall i : C_i = 1,$$

Tax shortfall theoretical definition

Aggregate STax

The aggregate STax is defined by:

$$
\begin{aligned}
STax &= \sum_{i:(C_i=0)\cap(\widetilde{D}_i=1)} M_i^* \\
&= \sum_{i:C_i=0} \underbrace{M_i^*}_{\text{v.a.}} \times \underbrace{\mathbf{1}_{(\widetilde{D}_i=1)}}_{\text{v.a.}} = \sum_{i=1}^{n} \underbrace{M_i^*}_{\text{v.a.}} \times \mathbf{1}_{(C_i=0)} \times \underbrace{\mathbf{1}_{(\widetilde{D}_i=1)}}_{\text{v.a.}}
\end{aligned}
$$

where $\mathbf{1}_{(.)}$ is the dummy variable taking the value 1 when the condition is observed and 0 elsewhere.

How to fit shortfall in Tax Revenue with ML?

- How to deal with the **absence of Test data**?
- How to deal with **unbalanced data**?

1. Separate controlled and uncontrolled observations.
2. Fit a RF classifier to **explain who is a fraudster among the controlled observations**.
3. Fit a RF regression to **explain the amount of fraud among the fraudster of the controlled observations**.
4. Predict with the RF classifier **who is a fraudster among the uncontrolled observations**.
5. Predict the amount of fraud with the RF regression **of the predicted fraudsters among the uncontrolled observations**. Sum this predicted amount to get the STax.

Outline

1. Introduction

2. Methodology
   - Definitions
   - DGP

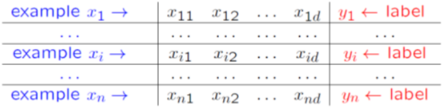
3. Ensemble Methods
   - Setting
   - General principles

4. Bagging and Random Forests
   - Bagging
   - Random Forests

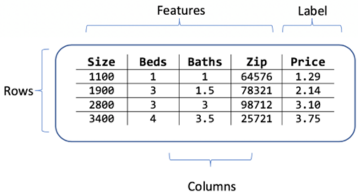
5. Semi-supervised learning

Vocabulary

- **Experiment:** data set $z_i = (x_i, y_i) \in \mathbb{R}^{d+1}$ extracted from the observation of a phenomenon.
- Let a **training dataset** be denoted by $(x_1, y_1), \ldots, (x_n, y_n)$
- Each element of the sample corresponds to an example (example) or an instance (**instance**).
- $x_i \in \mathbb{R}^d$ represents the **features** vector of an instance.
- y_i is the target variable (target) or **label**.



example x_1 →	x_{11}	x_{12}	$\ldots$	x_{1d}	y_1 ← label
$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$
example x_i →	x_{i1}	x_{i2}	$\ldots$	x_{id}	y_i ← label
$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$
example x_n →	x_{n1}	x_{n2}	$\ldots$	x_{nd}	y_n ← label


Features
Label
Rows
Columns

Size	Beds	Baths	Zip	Price
1100	1	1	64576	1.29
1900	3	1.5	78321	2.14
2800	3	3	98712	3.10
3400	4	3.5	25721	3.75

Setting

Standard supervised learning

- $\mathcal{X}$ the "input" space and $\mathcal{Y}$ the "output" space
- D a fixed and unknown distribution on $\mathcal{X} \times \mathcal{Y}$
- a loss function ℓ
- a data set $\mathcal{D} = ((\mathbf{X}_i, \mathbf{Y}_i))_{1 \leq i \leq N}$ with $\mathcal{D} \sim D^N$

Combining models

- several models: $g_1, \ldots, g_K$, K functions from $\mathcal{X}$ to $\mathcal{Y}$
- can one define g from $g_1, \ldots, g_K$ in order to get better performances than with using only one of the g_k?

Majority voting

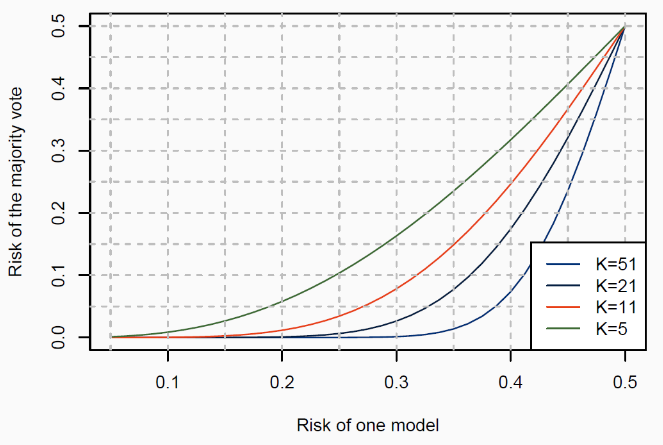
A simple example

- $\mathcal{Y} = \{-1, 1\}$ and $\ell(p, t) = \mathbb{1} p \neq t$
- $g(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} |\{1 \leq k \leq K \mid g_k(\mathbf{x}) = y\}|$ (majority voting)
- is g better than the g_k?

Analysis

- (partially unrealistic) assumptions:
  - $\forall k, R_\ell(g_k) = p < \frac{1}{2}$
  - the $g_k(\mathbf{x})$ are independent random variables (!)
- $\mathbb{P}(g(\mathbf{X}) \neq \mathbf{Y}) = \mathbb{P}\left(|\{1 \leq k \leq K \mid g_k(\mathbf{X}) \neq \mathbf{Y}\}| > \frac{K}{2}\right)$
- Binomial random variable with parameter p and K
  - $p = 0.45$ and $K = 21$: $R_\ell(g) = 0.321$
  - $p = 0.3$ and $K = 21$: $R_\ell(g) = 0.0264$

General behavior



Main ideas

To combine models

we need

- models with at least some predictive capabilities
- with independent errors

Bias variance decomposition

- another argument for model combination
- data point of view rather than model point of view:
  - a unique learning algorithm
  - several data sets
- but a similar conclusion

Ensemble methods

Techniques to combine models

Two main questions

Building an ensemble method

1. given K models, how to combine them efficiently?
2. given a data set and some learning strategies, how to produce K models that have “independent” errors?

Combining models

Combination

- $g_1, \ldots, g_K$, K functions from $\mathcal{X}$ to $\mathcal{Y}$
- how to build g from $g_1, \ldots, g_K$?
- numerous solutions related to the nature of $\mathcal{Y}$ and to ℓ

Linear combination

- for $\mathcal{Y} = \mathbb{R}$ (or $\mathbb{R}^Q$)
- define

$$g(\mathbf{x}) = \sum_{k=1}^{K} w_k g_k(\mathbf{x})$$

- one *could* learn the $(w_k)_{1 \leq k \leq K}$ from the data but:
  - risk of overfitting
  - when there is no overfitting, the gain is small
- frequent solution $w_k = \frac{1}{K}$

Discrete predictions

Discrete case

- when $|\mathcal{Y}| < \infty$
- majority voting

$$g(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} |\{1 \leq k \leq K \mid g_k(\mathbf{x}) = y\}|$$

Binary case with score

- when $\mathcal{Y} = \{-1, 1\}$ and $g_k(\mathbf{x}) = (f_k(\mathbf{x}))$
- linear combination

$$g(\mathbf{x}) = \left(\sum_{k=1}^{K} w_k f_k(\mathbf{x}) \right)$$

Probabilities and more

Probabilities

- when $|\mathcal{Y}| < \infty$
- $g_k(\mathbf{y}, \mathbf{x})$ estimates $\mathbb{P}(\mathbf{Y} = \mathbf{y} | \mathbf{X} = \mathbf{x})$
- weighted product (Z is a normalization constant)

$$g(\mathbf{y}, \mathbf{x}) = \frac{1}{Z} \prod_{k=1}^{K} g_k(\mathbf{y}, \mathbf{x})^{w_k}$$

Numerous other solutions

- mostly based on the concept of generalized mean
- adapted to special cases such as ranking (or even unsupervised settings)
- but in general, the mean, the uniform product or the majority voting are sufficient

Building models

Diversity

- is mandatory!
- most learning algorithms are deterministic: same data $\Rightarrow$ same model
- two general solutions:
  1. introduce some randomness in the algorithm (or leverage the natural randomness of the algorithm)
  2. run the same algorithm on modified data (randomly or deterministically)

Data level diversity

Instance subsets

- each classifier is obtained on a “subset” of the data set
- *Bagging*: bootstrap sample
- *Cross-validated committee*: block decomposition

Feature subsets

- each classifier is obtained using only a subset of the features
- *Random Subspace Method*: as is
- *Random Forests*: combined with *Bagging*

Data level diversity

Weighted instances

- each classifier is obtained using a weighted version of the data set
- *Boosting*: sequential training where weights are increased for badly predicted instances so far

Algorithmic level

Feature subsets

- when features are randomly selected repeatedly at different stages of the algorithm
- *Random Forests*

Natural instability

- complex models with strong dependence to the random initialization
- typical example: *Neural networks*
- could be forced further:
  - ensemble training
  - penalty for correlated prediction
  - *Negative Correlation Learning*

Outline

1 Introduction

2 Methodology
   - Definitions
   - DGP

3 Ensemble Methods
   - Setting
   - General principles

4 Bagging and Random Forests
   - Bagging
   - Random Forests

5 Semi-supervised learning

Bagging

Boostrap Aggregating

- Breiman (1996)
- simple and efficient
- consists simply in training models on bootstrap samples!
- adapted to models with high variance such as decision trees

Bagging and imbalanced data

- a bootstrap sample contains roughly 63.2% of the original data: could induce serious balance issues
- stratified bootstrap must be used
- avoid balancing the data *before* the bagging

Out-of-bag estimate

Principle

- leverage the fact that a bootstrap sample contains only 63.2% of the original data
- compute the prediction for a data point x using only the models for which x was not in the bootstrap sample

Formal definition

- K bootstrap samples $\mathcal{D}_1, \ldots, \mathcal{D}_K$ with associated models $g_1, \ldots, g_K$
- O_i: the set of indices k for each $\mathbf{X}_i \notin \mathcal{D}_k$
- risk estimate (averaging case):

$$\widehat{R}_{\ell}^{oob}(g_{bag}) = \frac{1}{N} \sum_{i=1}^{N} \ell \left(\frac{1}{|O_i|} \sum_{k \in O_i} g_k(\mathbf{X}_i), \mathbf{Y}_i \right)$$

Increasing the diversity

Feature subsets

- the bootstrap might not induce enough diversity
- adding random subspace might also be insufficient

Random Forest

- bagging of trees
- local random feature subsets
- tree growing:
  - standard CART approach: chose the best variable among all the variables
  - random forests: chose the best variable among a random subset of the variables
- no pruning

Random Forests

General outline (Breiman, 2001):

- Random Forests are an improvement on Bagging in the context of decision trees (CART algorithm).
- Bagging performance in terms of variance reduction is a decreasing function of the correlation between individual trees.
- The aim of random forests is to further reduce this correlation during the sampling process.
- **Sampling no longer only concerns instances**, but also **candidate predictors** for binary splitting.

Random Forests

General outline (Breiman, 2001):

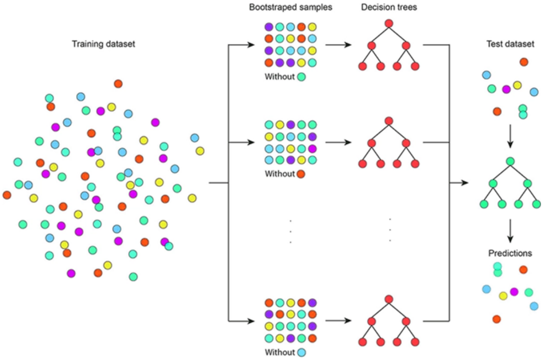


Figure: Source : https://learnetutorials.com

Random Forests

- A Random Forest is an ensemble of Decision Trees, generally trained via the bagging method.
- The Random Forest algorithm introduces extra randomness when growing trees:
    - Normal Decision Trees search for the very best feature when splitting a node.
    - Random Forests search for the best feature among a random subset of features.
        - This results in a greater tree diversity, which trades a higher bias for a lower variance, generally yielding an overall better model.

Extra-Trees

- It is possible to make trees even more random by also using random thresholds for each feature.
- A forest of such extremely random trees is simply called an Extremely Randomized Trees ensemble (Extra-Trees).
    - Extra-Trees are much faster to train than regular Random Forests since finding the best possible threshold for each feature at every node is one of the most time-consuming tasks of growing a tree.

Feature Importance

- Random Forests are very handy if you need to perform feature selection.
- Important features are likely to appear closer to the root of the tree. Hence, to get an estimate of a feature's importance average depth at which it appears across all trees in the forest.
    - In the Iris dataset, the most important features are the petal length (44%) and width (42%), while sepal length and width are at 11% and 2%.

Random Forests

In practice

- one of the best state-of-the-art solution for classical data
- leverage the out-of-bag estimate (all in one solution)
- parallel computation of the trees
- classical parameters:
  - at least 500 trees (a.k.a. bootstrap samples), depends on the complexity of the data
  - the “optimal” size of the subset of variables depends on several aspects:
    - highly correlated variables do not need a large subset (!)
    - Breiman recommended to use $P/3$ for regression problems and $\sqrt{P}$ for classification problems
  - the parameters can (should) be optimized

In summary

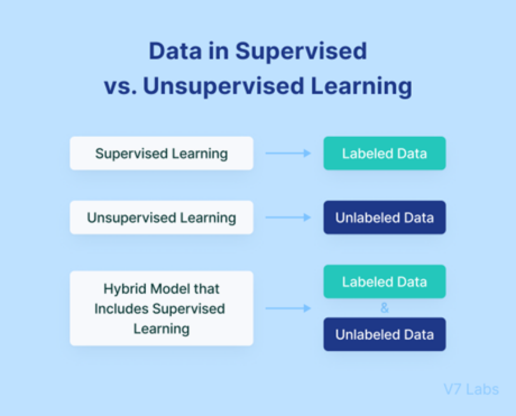
Ensemble methods

- Random Forests
- Boosting (to investigate yourself)
- pros and cons
  - + state of the art performances
  - + straightforward parallel implementation
  - + efficient large scale implementations
  - + adapted to mixed data
  - + handle missing data
  - - black box models
  - - high runtime compared to many other models
  - - high storage cost compared to many other models

Outline

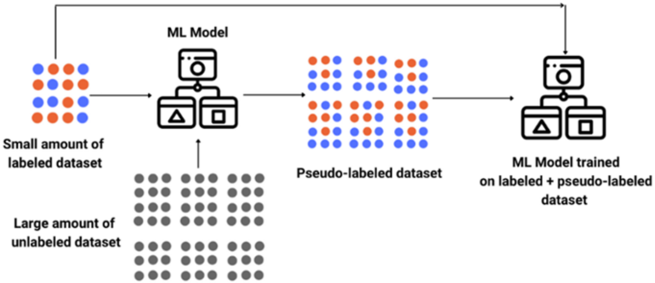
1. Introduction
2. Methodology
   - Definitions
   - DGP
3. Ensemble Methods
   - Setting
   - General principles
4. Bagging and Random Forests
   - Bagging
   - Random Forests
5. Semi-supervised learning

How to use uncontrolled data in our model construction?

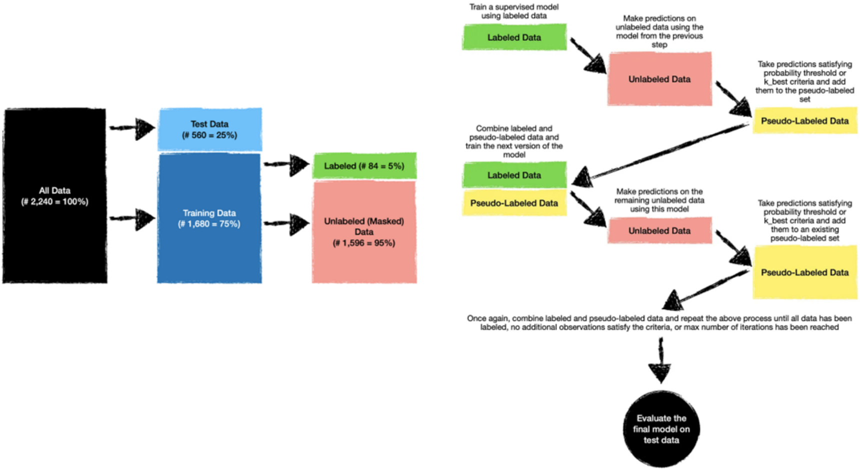


Definition

Semi-supervised learning is a learning process **based on both labeled (usually in the minority) and unlabeled (usually in the majority) data.**


ML Model
Small amount of labeled dataset
Large amount of unlabeled dataset
Pseudo-labeled dataset
ML Model trained on labeled + pseudo-labeled dataset

Definition



All Data
(# 2,240 = 100%)

Test Data
(# 560 = 25%)

Training Data
(# 1,680 = 75%)

Labeled (# 84 = 5%)

Unlabeled (Masked) Data
(# 1,596 = 95%)

Train a supervised model using labeled data

Labeled Data

Make predictions on unlabeled data using the model from the previous step

Unlabeled Data

Take predictions satisfying probability threshold or k_best criteria and add them to the pseudo-labeled set

Pseudo-Labeled Data

Combine labeled and pseudo-labeled data and train the next version of the model

Labeled Data

Pseudo-Labeled Data

Make predictions on the remaining unlabeled data using this model

Unlabeled Data

Take predictions satisfying probability threshold or k_best criteria and add them to an existing pseudo-labeled set

Pseudo-Labeled Data

Once again, combine labeled and pseudo-labeled data and repeat the above process until all data has been labeled, no additional observations satisfy the criteria, or max number of iterations has been reached

Evaluate the final model on test data

Application example for credit risk

Credit scoring with reintegration of rejects, Shen et al. (2022).

- When assessing credit risk, reintegrating rejected customers into the learning base is a technique that **resolves sample selection bias**.
- In credit evaluations, the accepted sample is labeled and the rejected sample is unlabeled.
- This article proposes a new approach based on:
  - A method for correcting feature distributions between accepted and rejected.
  - A **semi-supervised classification method** of the semi-supervised support vector machine (S4VM) type.

Semi-supervised learning with Python

Semi-supervised algorithms need to make assumptions about the distribution of the dataset in order to achieve performance gains.

- The continuity assumption: *Points that are close to each other are more likely to share a label.*
- The cluster assumption: *The data tend to form discrete clusters, and points in the same cluster are more likely to share a label.*
- The manifold assumption: *The data lie approximately on a manifold of much lower dimension than the input space.*

Semi-supervised learning with Python

The semi-supervised estimators in sklearn.semi_supervised are able to **make use of this additional unlabeled data to better capture the shape of the underlying data distribution** and generalize better to new samples.

These algorithms can perform well when we have a very small amount of labeled points and a large amount of unlabeled points.

There are two main learning approaches:

- Self Training is based on a recursive algorithm which uses a given supervised classifier to function as a semi-supervised classifier.
- Label Propagation (and Label Spreading) algorithm construct a similarity graph over all items in the input dataset.

Outline

6 Discussion

Mills Ratio

When control and fraud decisions are not linked, more precisely if $\rho_{cd} = 0$, the mean and variance formulas include the **inverse of the Mills ratio**:

$$\begin{aligned}
\mathbb{E}_X\left(STax\right) &= \sum_{i:C_i=0} \mathbf{X}_{m,i}\beta_m P_{\widetilde{D}_i=1} + \delta_c \sum_{i:C_i=0} \mathbb{E}_X\left(\varepsilon_{c,i} \middle| \varepsilon_{c,i} < b_{c,i}\right) \times P_{\widetilde{D}_i=1} \\
&\quad + \delta_d \sum_{i:C_i=0} \mathbb{E}_X\left(\varepsilon_{d,i} \middle| \varepsilon_{d,i} > a_{d,i}\right) \times P_{\widetilde{D}_i=1} \\
&= \sum_{i:C_i=0} \mathbf{X}_{m,i}\beta_m P_{\widetilde{D}_i=1} - \delta_c \sum_{i:C_i=0} \sigma_c \frac{\phi\left(b_{c,i}/\sigma_c\right)}{\Phi\left(b_{c,i}/\sigma_c\right)} \times P_{\widetilde{D}_i=1} \\
&\quad + \delta_d \sum_{i:C_i=0} \sigma_d \frac{\phi\left(a_{d,i}/\sigma_d\right)}{1 - \Phi\left(a_{d,i}/\sigma_d\right)} \times P_{\widetilde{D}_i=1}
\end{aligned}$$

where $P_{\widetilde{D}_i=1} = \Pr(\widetilde{D}_i = 1 | C_i = 0) = 1 - \Phi\left(a_{d,i}/\sigma_d\right)$.

Mills Ratio

- In the general case $\rho_{cd} \neq 0$, the expression of the conditional moments of the rectification M_i^* involves the moments of a **bivariate normal law with double truncation**.
- Manjunath and Wilhelm (2010) extend these results to the case of a multivariate distribution with arbitrary double truncation
- Kan and Robotti (2018) propose an alternative approach based on recurrence relations between integrals that involve the density of the multivariate normal to calculate moments for multivariate laws with double truncations arbitrary.
- The important point is that these double truncation moments can be very different from the **inverse Mills ratios usually used to deal with selection problems.**

Heckman estimation

Following Greene (2006):

“*Based on the wisdom in Heckman’s (1979) treatment of the linear model, there seems to be a widespread tendency (temptation) to extend his approach to other frameworks by mimicking his two step approach. Thus, for example, Wynand and van Praag (1981), in an early application, proposed to fit a probit model with sample selection with the following two steps:*

Step 1. *Fit the probit model for the sample selection equation.*

Step 2. *Using the selected sample, fit the second step probit model merely by adding the inverse Mills ratio from the first step to the main probit equation as an additional independent variable.*

This approach is inappropriate for several reason”, Greene (2006), page 1. **Indeed, (i) it does not allow correlations to be estimated, and (ii) it is not suitable in the case of a non-linear model.**

,:,:! nn:oooooo