Ecole Nationale Supérieure d'Ingénieurs de Tunis

الجمهورية التونسية
وزارة التعليم العالي والبحث العلمي
جامعة تونس
المدرسة الوطنية العليا للمهندسين بتونس

Réf : ING-GI-2016-15

Rapport de
Projet de Fin d'Études

Pour obtenir le

Diplôme d'Ingénieur en Génie Informatique

Option : Business Intelligence

Présentée et soutenue publiquement le 24 juin 2016

Par

Sameh HABBOUBI

Machine Intelligente d'analyse Financière

Composition du jury

Madame Ahlem BEN YOUNES.....**Président**

Madame Imen AOUINI .....**Rapporteur**

Madame Wahiba BEN FREDJ.....**Encadrant Entreprise**

Monsieur Wajdi GARALI .....**Encadrant ENSIT**

Année universitaire : 2015-2016

Ecole Nationale
Supérieure
d'Ingénieurs
de Tunis

الجمهورية التونسية
وزارة التعليم العالي والبحث العلمي
جامعة تونس
المدرسة الوطنية العليا للمهندسين بتونس

Réf : ING-GI-2016-15

Graduation Project

Report

To obtain

Engineering Degree in Computer Science

Option : Business Intelligence

Presented and supported publicly June 24, 2016

by

Sameh HABBOUBI

Intelligent Machine for Financial Analysis

Jury Composition

Mrs Ahlem BEN YOUNES ……**President**

Mrs Imen AOUINI ……**Rapportor**

Mrs Wahiba BEN FREDJ …… **Entreprise Supervisor**

Mr Wajdi GARALI …… **ENSIT Supervisor**

Academic year : 2015-2016

Acknowledgement

It is with a great pleasure that I reserve these few lines of gratitude and deep appreciation to all those who directly or indirectly contributed to the completion of this work:

I express my greatest thanks to Mrs Wahiba BEN FREDJ, despite her many duties, has agreed to collaborate and to provide leadership for this work. Her advices, her dedication, her availability, her relevant comments and corrections led to the success of this work.

I want to thank also Mr. Salah MEDDEB, General Manager at Computer Centre of the Ministry of Finance, to have lavished us the honor of working in the team.

Also I would like to thank Mr. Haithem JARRAYA Big Data Senior Developer, the machine learning expert from UK to whom we wish to express our gratitude for the help he has given to us during all phases of the course. His availability, coaching, and advice have been invaluable to achieve the objectives of this project.

I also express my greatest thanks to Mr Wajdi GARALI, who supported me with valuable feedback and always kindly encouraged me to succeed this project.

I thank all the teachers who have formed me during this cycle. The success of this project is also due to the knowledge that we have been taught in previous years at the High National School of Engineers of Tunis (ENSIT)

Finally, I wish to express my deep gratitude and thank my family who has consistently expressed its unconditional support and encouragement.

All those who contributed in one way or another, to make this work, can be found here, the crowning of their efforts.

Table of Content

List of Illustrations

List of Tables

General Introduction

Enjoy living in a modern society, is first of all having enough of financial resources that can meet more than one's living., the State budget is patently financed from public expenditure and public resources.

Since the tax is the main source of income in a country, anyone who takes advantage of public goods, should contribute to their funding. However, some companies often attempt to evade taxes' payment. Therefore, tax fraud creates a critical impact on the budget and government revenue. These companies often spend large sums of money to find ways to evade taxes, rather than focusing on their activities. This is why the State has to establish efficient means to detect tax fraud, but if the government cannot effectively detect this fraud, public investment would be worthy affected due to budget shortages resulting from tax revenue loss.

Currently, Tunisia is in a difficult and critical economic situation, the problem today is not to maximize or raise tax rates but to flatten them, to better allocate them and make them simpler, fairer and more legible.

Indeed the fight against fraud can contribute not only to lecture citizenship deficit in society, but also to recover funds which could be re-injected into the economy. Some taxes are more easily defrauded than others, control and prevention means are very unevenly successful. As a result, different social groups are not in the same situation with regard to tax fraud. Employees and retirees are easily controlled, from the statements made by their employers and pension funds. However, many liberal, commercial and industrial professions have many opportunities to take part of their activities to tax (fees recipe gross, concealment …) and the necessary measures for their control are not yet available.

Given the enlargement of data masses and high level transparency requirement in data processing and analyzing, an IT solution is the best solution to detect complex fraud cases. This practice is already multiplied in our European neighbors offering a high degree of transparency on new tools (machine learning) commissioned in recent years.

Faced with all this background which combines Big Data, Data Analysis and Data Warehouse, Tunisia is taking its first step on behalf of the Ministry of Finance which seeks a solution surrounding the use of large amounts of data in tax matters to detect fraud cases and to decrease

also human intervention and streamline tasks by automating control spot. This may improve the performance of screeners and help them select the real scammers and do not submit the same taxpayers

This final project study is most suitable to this scope, it lies indeed on the automation of audit operations extended to whole population and taxpayers domiciled within Tunisian territory. We will explain during this report the different undertaken steps we have performed to provide the ministry with a solution allowing to detect fraudsters and their tax adjustment's amount.

In the first chapter, we will present the project in its global context, we will present the host organization and the most relevant solution for the ministry requirement. The second chapter will highlight the machine learning basics which is the adapted science to analyze data masses and to predict fraudsters and their tax adjustment amount, we will also describe the mathematic aspects behind different used learning algorithms. The third chapter will be the most important part which focus on data extraction, processing, and preparing as input for machine learning algorithms. The fourth chapter will cover the test results of the different machine learning algorithms and their interpretations.

Chapter 1: Project Presentation

Introduction

This chapter will handle project's scope of work. We begin with project description and objectives. Next, we will briefly describe the host organization, we will also make a study of existing international solutions throughout a small description of their concepts. We finally end up by some financial general concepts and the adopted working methodology.

1.1. Host organization presentation

We present in this section the host organization and the fields in where it operates. This graduation project internship is spent into the Computer Center of the Ministry of Finances "CIMF" which is a public non-administrative organization stated with legal personality and financial autonomy. It was established in 1981 in Tunis.

According to the Finance Act No. 81/100 and Decree No. 82-799 of 17 May 1982 on the administrative and financial organization of CIMF, the center "is responsible for paying contest for study consulting, implementation and operation of computer applications to various departments of the Ministry of Finance and the institutions under guardianship of the Department."

1.1.1. Areas of activity

The Computer Centre of the Ministry of Finance is responsible for the following:

• Implement the Information Plan the Department of Finance.

• Develop information banks for the needs of departments of the Ministry and ensure their updates.

• Manage the IT equipment of the park's Department ensuring profitability.

• Ensure the operational work by ensuring the reliability of information, respect for deadlines carry computer refunds and security of all files.

• Develop the standards and methods in the design, construction, operation and maintenance of computer applications.

1.1.2. Organizational Structure

The center has nearly 200 employees who are all under the General Director leadership. The CIMF is organized around a General Manager in charge of more than fourteen directions, reflecting the organizational architecture of CIMF in its entirety. The figure n°1 presents the CIMF organizational chart:

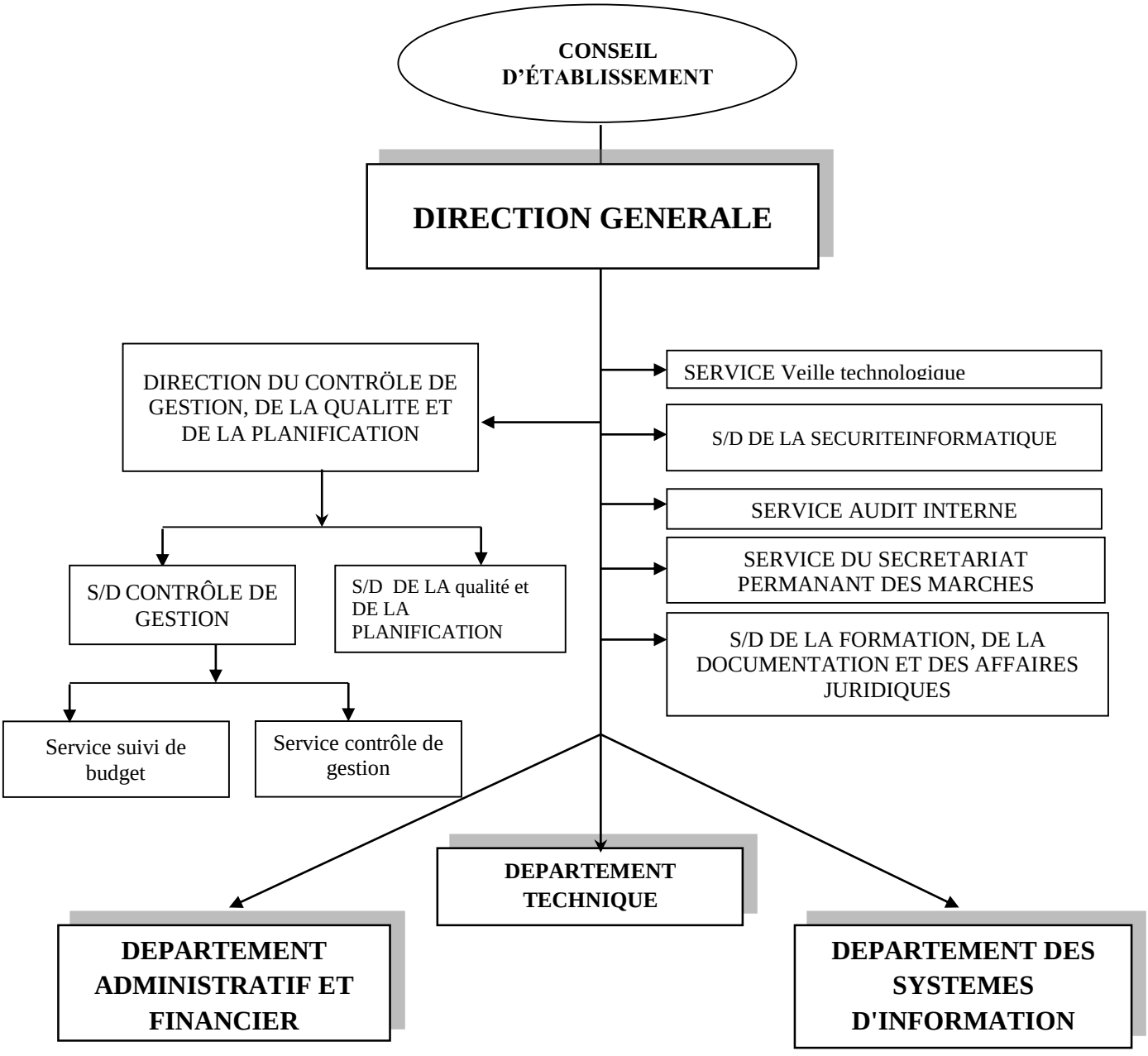


Figure n° 1 : CIMF Organizational chart [1]

This project was realized within the Information System Department specifically in SADEC project, we will present Information System Department's organizational chart in the figure n° 2:

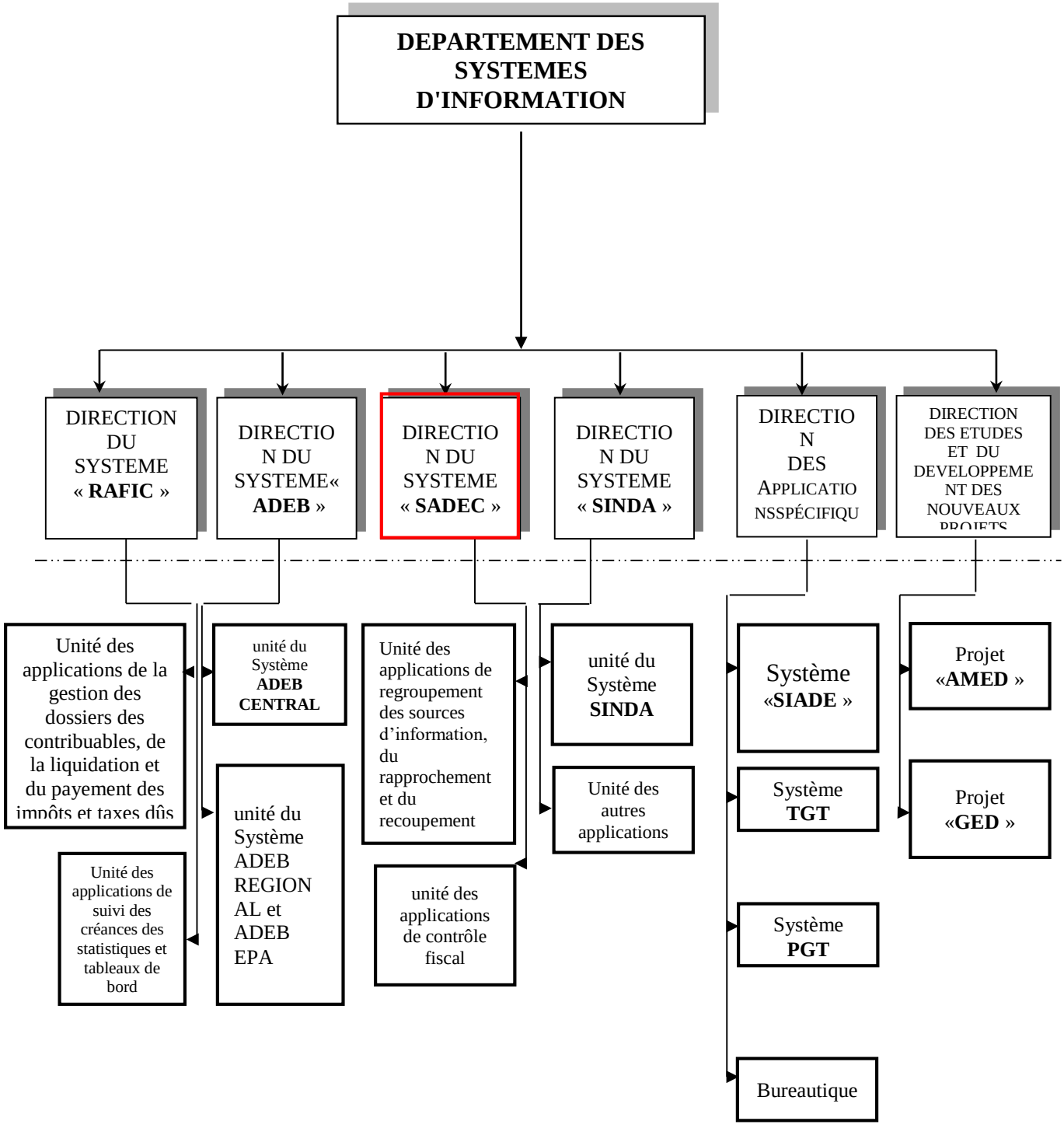


Figure n° 2 : Information System organizational chart [1]

1.1.3. Realized projects

The Computer Centre of the Ministry of Finance has made several applications. Among these applications we find:

1.1.3.1. ADEB

ADEB: Question-Budget Decision System (Système d'Aide à la Décision Budgétaire) is a transparent and fluid budget management system in a real-time status. It allows control and budget state operations monitoring (Commitment, schedules, payments ...) it provides reliable situations regarding actions taken by suppliers with the government and public authorities.

Users: This application is used by about 4,000 workplaces from 1200 to 1300 sites whose 264 are common, 24 councils and 400 for the central administration entirety.

1.1.3.2. RAFIC

RAFIC (Tax and Accounting Shares Streamlined System) manipulates streamlining accounting and tax actions. This application manages tax audit and recovery (indebted non-employees).

It maintains in real time Accounting Revenue Finance. It provides a national vision and professionals shared folders and their tax obligations. RAFIC is a set of sub-Systems organized concentrically.

- It integrates process control and revenue structures.
- It automates the recipe accounts.
- It provides assistance with tax audits.
- It allows the management of taxpayer's file (declaration, liquidation, tax monitoring...)
- It allows actions recording, debts management, VAT refunds, tax compensation on transportation, car sticker and road offenses…

It allows to collect 95% of tax per type, the distribution of recoveries and the number of transactions by mail, recipe and region.

Users: 6,000 users across all systems at over 500 sites:

- Taxation: Offices and control centers, DGE, counters, DGI (196 posts), Tax Centre service.

- Public Management: Finance Revenue, Regional Councils, Regional Treasurers.

1.1.3.3. SINDA

SINDA: Computer Automated Customs System (Système d'Informatique Douanier Automatisé)

SINDA provides the General Directorate of Customs an effective way to control the exported and imported goods.

It integrates the management and liquidation of customs transactions. It covers all customs procedures since goods arrival until their removal. It is used by Customs structures and its partners.

SINDA manages customs operations, economic operators, customs agents, industrial warehouses, customs disputes, deposits and goods seizures, data exchange with external partners, etc.

Users: Today, more than 2,000 users, whose partners are:

- Commissioners Customs
- Registrants Customs
- Customs offices

1.1.3.4. SADEC

SADEC: Decision Support and Fiscal Control System (Système d'Aide à la Décision Et au Contrôle Fiscal) is used to equip the tax administration with tools and efficient means for effective taxpayers' tax audit aid and rationalized working methods for permanent files monitoring. The new computer system SADEC was introduced to:

- Have useful information to get as close to the taxpayer real income as possible by exploiting different information sources: the employer's statement, SINDA and ADEB systems, as well as the CNSS file.
- Bring together various information cross-checked with those reported spontaneously by taxpayers.

- Undertake a more effective internal control performance of all control units with automatic management of the various actions undertaken by control services.

Users: Users of SADEC are control offices (167 identified so far). These offices are located throughout the Tunisian territory and are organized by a regional structure (3 in Tunis, Sfax and 2 on each governorate (except Tunis and Sfax)).

1.1.3.5. TGI

TGT: Tunisian General Treasury System (Système de la Trésorerie Générale de Tunisie)

This project's mission is the management and monitoring of internal Treasury operations and the centralization of finance recipients operations. The General Treasury of Tunisia acts as:

- Centralizing state accountant.
- Finance recipients' accountant.
- Public administrative institutions accountant.

Users: 450 accounting items, including 250 that use RAFIC directly to validate their accounts.

1.1.3.6. PGI

PGT: System Paymaster General of Tunisia (Système de la Paierie Générale de Tunisie)

The computer system is used as an accounting tool in the General Paymaster and departmental payment offices, this system complements the 'ADEB' system with monitoring spending state and the INSAF system with official payroll systems state whose paymaster is the main partner.

1.1.3.7. BI System

BI System: (Business Intelligence System) this system comes to complement existing systems **SADEC** and **RAFIC.** The BI system objectives are:

- Make available to relevant structures, means and tools for query, reporting and dashboards.
- Sectorial and geographical analysis of tax return.
- Monitoring taxpayers status.

- Comparative analysis between statements' elements in the same industry which can extract indicators and ratios to guide the cases selection that require monitoring or verification.
- Monitoring and analysis of changes in government revenue (tax, customs).

This system uses Business Object.

1.2. Project presentation

This project is realized within the preparation of my graduation project presented to obtain the engineering degree in computer science at the ENSIT during the academic year 2015/2016. I was performing my project into the CIMF.

This is a research based project. It aims to study, design, and develop a self-learning and smart machine to analysis the tax and accounting data masses to detect tax fraud and build predictive models in order to help detect in advance fraudsters.

Indeed, the CIMF is holding a mighty data set (e.g. VAT, Turnover, Income tax, perceived market amount, paid market amount, fiscal results, income at source...). We want to correctly predict the fraudsters' classes and their tax adjustment.

There are different models and algorithms for leading the study and performing as much tests to find a better representation that reaches a decisive result on fraudster state and on its tax adjustment.

1.3. International Examples

1.3.1. The Belgian Example

The Belgian example is developed by SAS firm on VAT carousel using software such as SPSS / SAS undermine, or I2, the "Social Network analysis" helps in the connections between the various stakeholders of fraud in the form of graphs. Today we can detect suspicious links and activities like never before. Establishing such transaction took place at the same time that such a change of address, even a protagonist is present in two countries at once and reiterates fraud schemes tested set the size of a network and identify [N1] leader "explains Cécile de Barsy consultant Business & Decision for the FPS Finance (federal public Service Finance) Belgian.

The losses to the Belgian Treasury on carousel fraud fell by € 1.1 billion in 2001 to € 26.04 million in 2006 and 93.6 million of euros in 2009, down from 85% in 8 years, and even 95% in 2011 with 18 million euros of VAT defrauded. Furthermore, the detection operations processing times have been shortened tremendously, from about 3 weeks to 5 minutes [N2], 99.9% of the detection process is triggered from the first false declaration.

An extrapolation for France recently conducted [N1] where the reporting inconsistencies would cost € 32 billion each year in public funds on VAT which 13 billion related to carousel fraud would recover nearly 9.8 billion euros via target Testing within 15 days, while the random tests currently performed between 12 and 15 months after receipt of VAT to control more than 48,000 companies / year for 1.1 billion euros recovered amounts.

1.3.2. The Italian Example

On 4 January 2013, the Italian tax authority has adopted a computerized robot, the "Redditometro", using hundreds of indicators to virtually reconstruct the amounts spent as treatment approaches the amounts reported on line (the paperless tax return is now mandatory in Italy). In case of difference of over 20% between the amounts declared and the amounts spent, fiscal control is immediately commissioned. Using the banking and financial data of individuals as well as data back traders, tax services can thus intersect the cost of purchasing a vehicle, small clothing expenses, financial or real estate investments, etc.

Its first use was initiated in March in relation to reported income since 2009. Audits are greatly enhanced by the fact that in Italy most operations require the use of a unique tax number possessed by each. For all other transactions, the Ministry of Finance uses national statistical data, on a geographical basis: Italy is thus divided into five geographical areas, the expenditures reported in 11 different family types ranging from Singles under 35 years to the couples over 65 years.

[N3] The software then in 2013 was applied to 35,000 Italian households during the last four years. Taxpayers however enjoying the right to oppose the request for clarification sent by administering evidence (documents...) to clear their selves. Moreover, indirect evidence such as the total income of each family is right-and-already taken into account in order to smooth possibly aberrations. In case of inability to prove good faith spontaneously, personal situation can then be initiated.

1.3.3. Britsh and American tax refund system

Developed in 2009 (launched in 2010) by the firm BAE systems [2] for a cost of 45 million pounds, the device 'Connect' has identified 2 billion of additional revenue for fiscal year 2011-2012 and is expected to generate 22 billion of additional books from 2014-2015.

[3] In the year of its launch, the tool has identified £ 600 million of additional revenue 330 million of gains on the TVA segment (amounts collected non-cash) and 151 million non-requests for additional fraudulent refund 118 million books being released on direct taxation. The VAT fraud have thus increased from 15.7% in 2002 to 9.5% in 2012, reimbursement of fraudulent tax lowered from 9.2% to 4.4% between 1997 and 2011 and scams to unemployment benefits from 13.2% to 4.6% over the same period. Even movement in the United States with the establishment by the Treasury Department (which depends the IRS) of the MeF platform (Modernized electronic Filing) for checking inconsistencies in tax returns corrective in 2006. At stake, a decrease of 20% of the amount of fraud detected, an annual average decrease of 6.7% / year to recover about 70 billion euros of additional revenue for the year 2010.

1.4. General financial concepts

In the following we present some financial theoretical definitions of the terminology that will be mentioned throughout this report

1.4.1. Tax definition

Taxation refers to all the rules, laws and measures governing the taxation of a country. Otherwise defined, taxation boils practices used by a state or a community to collect taxes and other levies.

Taxation plays a key role in the economy of a country. It participates to finance the latter needs and is causing public spending (motorway works, construction of buildings, etc.). Included in this tax, many taxes are paid directly by households and companies (housing tax, property tax, income tax, business tax, etc.) or indirectly (transfer taxes, vehicle registration, etc.). The economic policy of a country finally has a great influence on taxation with the power to tax more certain economic agents or, alternatively, reduce taxation of certain operations.

1.4.2. Taxpayer

The taxpayer is the person to contribute to public expenditure in paying his taxes. There are two big categories of taxpayers, “Physical Persons” taxpayers which includes traders and the category of “Moral Persons” which includes companies.

1.4.3. Tax Fraud & Tax evasion

Legally, tax fraud is defined as the unlawful removal from the tax legislation of all or part of the tax base of a taxpayer. In other words, the fraudster pays little or no tax by using illegal means.

This concept should not be confused with tax evasion, which is a practice to circumvent or reduce the tax by taking advantage of the opportunities offered by the tax rules or their shortcomings (tax loopholes, acquisition of another nationality ...), in other words tax evasion uses legal means to escape tax payment. [N4]

1.4.4. The components of tax fraud

The tax fraud is established by the meeting of:

- The deliberate omission of declaration within the prescribed period;
- The deliberate concealment of amounts subject to tax;
- The deliberate award of fictitious or inaccurate entries;
- Encouraging the public to refuse or delay tax payment;
- The issuance of false invoices;
- The opposition to Tax Administration’s action.

1.4.5. Taxation in Tunisia

The Tunisian tax system is based on the principle of "voluntary consent to taxation". In other words, the acquittals are spontaneously paid by the taxpayer. In such system, the taxpayer seeks necessarily to maximize his contribution. Certain categories of income escape largely indeed to tax authorities. Employees have few options in this regard, since their income is reported to the source by employers. However, recipients of non-wage income professionals have more leeway, because they report themselves their agricultural benefits, industrial, commercial or non-commercial and property income. [4]

Tunisia's tax system includes taxes and following duties:

- Tariffs,
- Value added tax,
- Consumer Law,
- Tax on personal income,
- Corporation tax,
- Registration and stamp duties,
- Local taxation,
- Various taxes on certain products, transportation, insurance ...

During our study, we will focus on the Value Added Tax VAT, Tax on physical person's income IRPP and tax on companies or moral persons IS.

1.4.6. Tax verification

The tax verification may take the form of a preliminary verification of the statements, deeds and documents held by the tax authorities or a comprehensive audit of the taxpayer's tax situation.

Each form of tax audit has peculiar rules. However, both forms of verification are joined at common rules on the notification date results of the taxpayer audit.

In Tunisia, the Ministry of Finance uses a methodology to send verification missions to the enterprises. In fact, The DGI annually prepares a list of companies/traders to plan for further verification and it assigns each control office a list, on top of that the offices and control centers can check other companies/traders and will not be limited to the list provided by the DGI.

1.4.6.1. Preliminary verification

The preliminary verification covers all operations of the tax administration services relating to the correction of errors or obvious omissions. It may affect one or more taxes to the recovery time extent. It takes place in the tax administration's premises using documents and information they have. A preliminary recovery (or REDR) is the money due sum from a preliminary audit. [6]

1.4.6.2. Thorough verification

Thorough tax audit covers all part of taxpayer's tax situation. It may affect one or more taxes, and one or more periods within the limit of recovery time. Given its characteristic, it often requires tax official's intervention at the premises of the taxpayer to consult and analysis accounting documents. A thorough recovery (or REDR APP) is the sum of money due from a thorough audit. [7]

1.4.7. IRPP

IRPP is the tax on personal income for physical persons (or IRPP: L'impôt sur le revenu des personnes physiques) or companies having individual business's form [N5]

1.4.8. IS

IS is the corporate tax (or IS: L'impôt sur les sociétés) concerns limited liability companies, one-person limited liability companies and anonymous companies [N5]

1.4.9. VAT

A value-added tax (VAT) is a type of consumption tax that is placed on a product whenever value is added at a stage of production and at final sale. The amount of VAT that the user pays is the cost of the product, less any of the costs of materials used in the product that have already been taxed. VAT rate depends on countries and on taxpayers' activities (medical activity, agriculture, goods transportation …

1.5. Working Methodology

During this internship, we adopted a working methodology for project analysis, data mining and data science. It is the "CRISP-DM" "Cross Industry Standard Process for Data Mining" which is the most popular method for this kind of project.

The CRISP method (initially known as CRISP-DM) was originally developed by IBM in the 60s to make data mining projects. It remains today the only effective method available for all Data Science projects. Figure n° 3 illustrates a voting result of using certain methodologies for data mining projects: [N6]

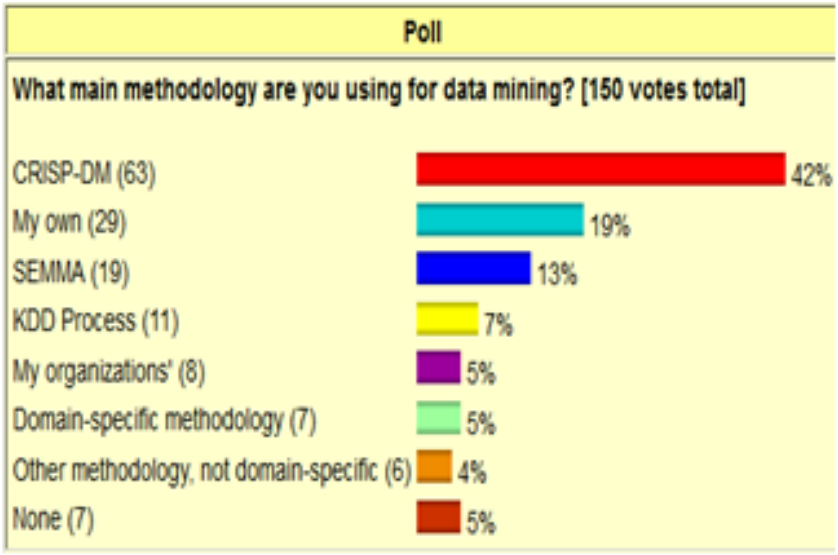


Figure n° 3: Voting result: most adopted data mining methodologies by companies

In Data mining work, we opted for the CRISP-DM methodology which is a methodology tested on land, it was officially adopted by Business & Decision and its use is a key factor to the success of Data Science projects [N7]:

As a methodology, CRISP-DM includes descriptions of typical project phases and tasks included in each phase, and an explanation of the relationships between these tasks.

As a process model, CRISP-DM provides an overview of data mining life cycle

The lifecycle model includes six phases with arrows indicating the dependencies the most important and most frequent. The phase sequence is not strictly established. Therefore, projects, for the most part, pass from one phase to the other as needed. Figure n°4 illustrates the life cycle of a data project mining [N8]:

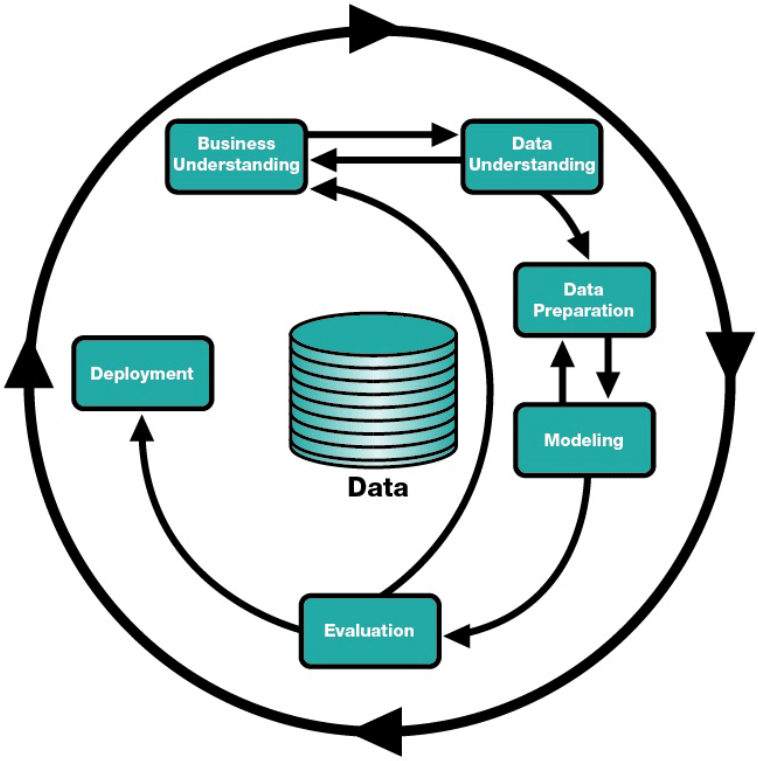


Figure n° 4: CRISP-DM life cycle

This method is iterative and agile, which means that each iteration brings additional business knowledge to better deal with the next iteration. [N9]:

The CRISP-DM phases are detailed in Table n° 1:

Phases	Tasks
Business Understanding	• Determine Business Objectives • Assess Situation • Determine Data Mining Goals • Produce Project Plan
Data Understanding	• Collect Initial Data • Describe Data • Explore Data • Verify Data Quality

Data Preparation	• Select Data • Clean Data • Construct Data • Integrate Data • Format Data
Modeling	• Select Modeling Technique • Generate Test Design • Build Model • Assess Model
Evaluation	• Evaluate Results • Review Process • Determine Next Steps
Deployment	• Plan Deployment • Plan Monitoring & Maintenance • Produce Final Report • Review Project

Table 1: CRISP-DM Phases and Tasks

Conclusion

In this introductory chapter, we presented the host organization, the project to achieve and some theoretical terminologies. We have also listed some existing solutions and the adopted methodology. We will now begin the project preparation phase of defining the different theoretical concepts - related to our field of study "Machine Learning", affecting our project. This will be detailed in the next chapter.

Chapter 2: Machine Learning

Introduction

During the second chapter, we will try to explain work baseline: What is machine learning and what difference there between data mining and machine learning. We will further elucidate how to learn from raw data new knowledge and how to predict.

2.1. Giving computers the ability to learn from data

Machine learning is a data analysis method which automates analytical model building. By using algorithms that iteratively learn from the input data, machine learning allows computers to find hidden insights without being explicitly programmed. [N10]

We are living in an age where data comes in abundance, throughout using self-learning algorithms from machine learning field we can turn this data into knowledge. Machine learning goal is to teach machines to carry out tasks by providing them a couple of examples (how to do or not do the task). This involves the development of algorithms to obtain a predictive analysis based on data for a specific purpose. This is sort of learning by example. In fact, machine learning will create a program, rather than attempting to define rules that ensue to definitely events. Let's imagine how with a large mass of data, define rules would be tedious!

2.2. A paradigm change

With the Machine learning, we further seek to establish correlations between two events rather than causality.

⇒ **Example**: we can detect a correlation between sugar consumption and heart disease without saying that one is causing the other. However, the correlation is useful if for example you wish to identify persons likely to heart disease.

Some people usually confuse data mining and machine learning. In fact, data mining in short is the science of finding patterns in data: Finding patterns means when we have a massive database, we will extract all the knowledge from this data to develop a consistent set of output

and report based on that data. To do so we need to know the structure of the data. We would summarize this: what knowledge can we gain from the data we have?

In the other hand machine learning involves the algorithm that improves automatically throw experience based on data. This learning involves two types of data: training data and test data, where training data is the data given to the algorithm to train itself how to pair the input with expected output to finally build up a model, while test data is used in order to estimate how well the model has been trained and to estimate model properties (mean error for numeric predictors, classification errors for classifiers, recall and precision for models etc…)

2.3. Classes of Machine Learning problems

In machine learning we have two classes, **supervised learning** and **unsupervised learning**.

2.3.1. Supervised Learning

Supervised learning is the most known type of machine learning problems. In supervised learning, we are given a data set and already know what our correct output should look like, having the idea that there is a relationship between the input and the output.

Supervised learning problems are categorized into "regression" and "classification" problems. In a regression problem, we are trying to predict results within a continuous output, meaning that we are trying to map input variables to some continuous function. In a classification problem, we are instead trying to predict results in a discrete output. In other words, we are trying to map input variables into discrete categories.

2.3.1.1. Classification

Classification is probably the most common type of task; this is due in part to the fact that it is relatively easy, well understood, and solves a lot of common problems. Classification is about assigning classes to a set of instances, based on their features. This is a supervised learning method because it relies on a labeled training set to learn a set of model parameters. This model can then be applied to unlabeled data to make a prediction on what class each instance belongs to. There are broadly two types of classification tasks: **binary classification** and **multiclass classification**.

⇒ **Example** A typical binary classification task is e-mail spam detection. Here we use the contents of an e-mail to determine if it belongs to one of the two classes: spam or not spam, for example 1 for spam and 0 for non-spam.

An example of multiclass classification is handwriting recognition, where we try to predict a class, for example, the letter name. In this case, we have one class for each of the alpha numeric character, for example 0: A, 1: B, 2:C … Multiclass classification can sometimes be achieved by chaining binary classification tasks together, however, we lose information this way, and we are unable to define a single decision boundary. For this reason, multiclass classification is often treated separately from binary classification.

Example modeling:

We will model another example similar to spam email classification. We will take the sample of cancer classification to know if a tumor is malignant or benign based on the tumor size. To model this we will take , x our feature, and y the predicted value:

To model this we will have:

$$\begin{cases} X: \text{Tumor size}, X \in \mathbb{R}^{12*1}, 12 \text{ training examples}, 1 \text{ feature} \\ y: \in \{0,1\} \begin{cases} 0: \text{Negative Class (e.g., benign tumor)} \\ 1: \text{Positive Class (e.g., malignant tumor)} \end{cases} \end{cases}$$

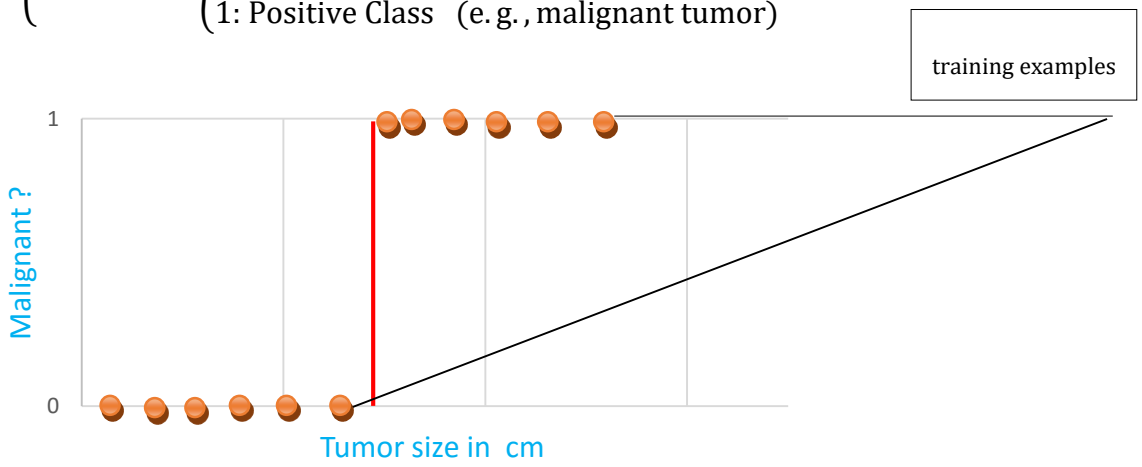


Figure n° 5: Breast Cancer (Malignant, Benign)

According to the figure n°5, the machine learning algorithm set up a threshold line which separate well our data set. The examples on the right of the red line are malignant tumors while examples on the left are benign tumors. To better the prediction of this learning algorithm we can add more features: Age, Clump thickness, Uniformity of cell size, Uniformity of cell shape…So will end up by:

$$\begin{cases} x_1: \text{Tumor size}, \\ x_2: Age, \\ x_3: \text{Clump thickness}, \\ x_4: \text{ Uniformity of cell size}, \\ x_5: \text{Uniformity of cell shape} \end{cases}$$

Thus, each row in this feature matrix represents one instance and can be written as five-dimensional column vector, to model this, we can have:

$$\begin{cases} \boldsymbol{X} = [x_1, \quad x_2, \quad x_3, \quad x_4, \quad x_5], \qquad \boldsymbol{X} \in \mathbb{R}^{12*5}, \qquad 12 \text{ training examples}, \qquad 5 \text{ features} \\ \boldsymbol{y}: tumor\ size\ type, y \in [0,1] \end{cases}$$

2.3.1.2. Regression

There are cases where what we are interested in are not discrete classes, but a continuous variable, for instance, a probability. These types of problems are regression problems. The aim of regression analysis is to understand how changes to the input, independent variables $\boldsymbol{X}$, effect changes to the dependent variable $\boldsymbol{y}$. The simplest regression problems are linear and involve fitting a straight line to a set of data in order to make a prediction. Typical regression problems include estimating the likelihood of a disease given a range and severity of symptoms, or predicting test scores given past performance.

Example modeling:

Let's say we want to predict house pricing, we are given this data set of 14 training examples, and we want to find the price of a new house based on its size in m², here we have one feature which is house size in m². The goal here is to predict housing price in TND which is y giving its size x in m², to summarize this we will have:

$$\begin{cases} X: House\ size\ in\ m^2, \qquad \boldsymbol{X} \in \mathbb{R}^{14*1}, \qquad 14 \text{ training examples}, \qquad 1 \text{ feature} \\ y: Price\ of\ the\ house, predicted\ value, y \in \mathbb{R} \end{cases}$$

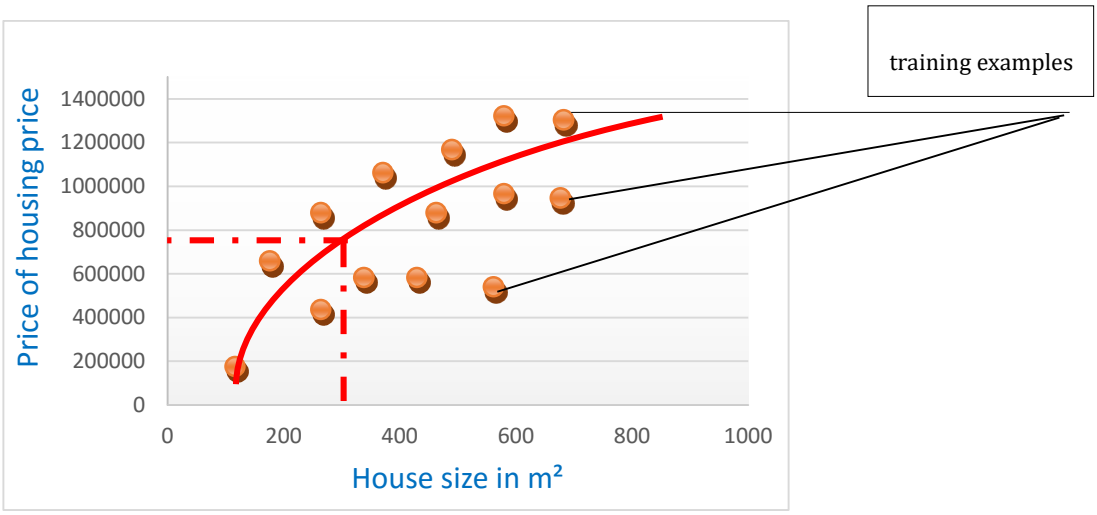


Figure n° 6: Housing price prediction

: Learning algorithm curve

The learning algorithm found that this curve best suits the data set. Into the figure n°6, we have tried to predict the price of 300 m² size house. The learning algorithm has predicted 790 000 TND. So this is an example of a regression problem where we want to predict continuous values. We can also add more features like: House emplacement, House condition:

$$\begin{cases} x_1: \text{House size in m}^2, \\ x_2: \text{House emplacement}, \\ x_3: \text{House condition} \end{cases}$$

We will end up by this mathematical representation:

$$\begin{cases} X = [x_1, \quad x_2, \quad x_3], \qquad \boldsymbol{X} \in \mathbb{R}^{14*3}, \qquad 14 \text{ training examples}, \qquad 3 \text{ features} \\ \qquad\qquad\qquad\qquad y: Price\ of\ the\ house, predicted\ value, y \in \mathbb{R} \end{cases}$$

2.3.2. Unsupervised Learning

In unsupervised learning, we deal with unlabeled data (data that has no sort of meaningful "tag," "label," or "class" that is somehow informative or desirable to know) and our goal is to find hidden patterns in this data to extract meaningful information. It allows us to approach problems with little or no idea what our results should look like. We can derive structure from data where we don't necessarily know the effect of the variables. With unsupervised learning there is no feedback based on the prediction results, i.e., there is no teacher to correct you.

2.3.2.1. Clustering

Clustering is the most well-known unsupervised method. Here, we are concerned with making a measurement of similarity between instances in an unlabeled dataset. We often use geometric models to determine the distance between instances, based on their feature values. We can use an arbitrary measurement of closeness to determine what cluster each instance belongs to. Clustering is often used in data mining and exploratory data analysis. There are a large variety of methods and algorithms that perform this task, and some of the approaches include the distance-based method, as well as finding a center point for each cluster, or using statistical techniques based on distributions.

2.3.3. Learning algorithm intuition

For the rest of this report, we will use the superscript i to refer to the ith training sample, and the subscript j to refer to the jth dimension or feature of the training dataset.

We use lower-case, bold-face letters to refer to vectors ($x \in \mathbb{R}^{n*1}$) and upper-case, bold-face letters to refer to matrices, respectively $X \in \mathbb{R}^{n*m}$ where n is the number of training examples and m the number of features. To refer to single elements in a vector or matrix, we write the letters in italics, x^n or x_m^n respectively).

Example modeling:

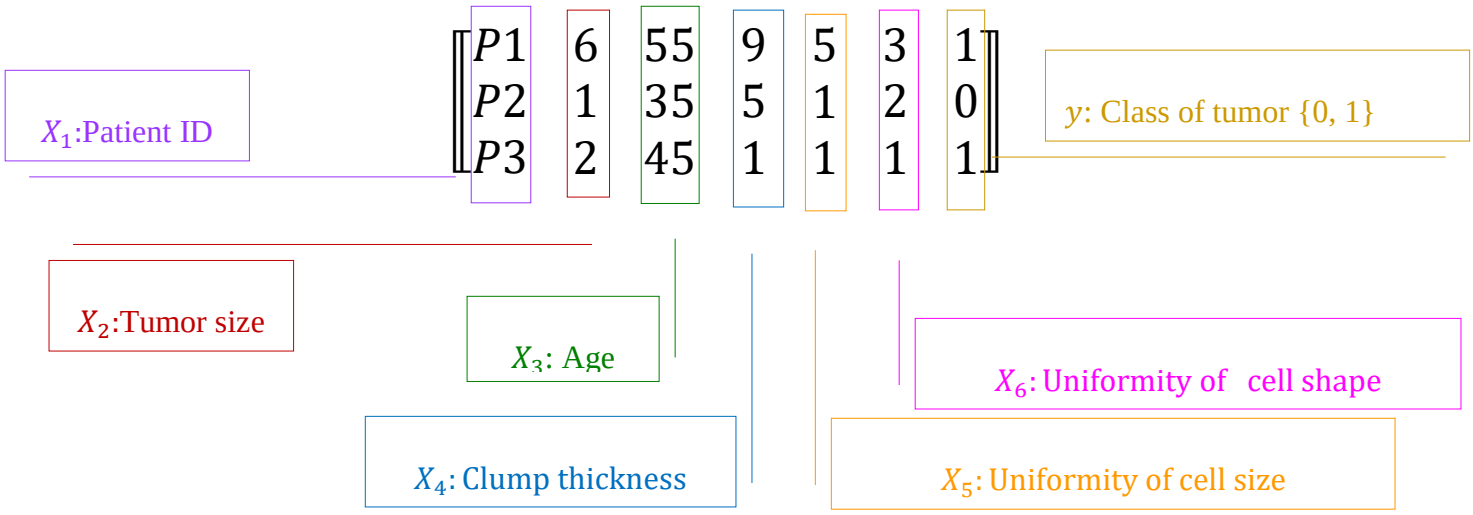
In the enriched Breast Cancer example, we use x_1^9 refers to the first dimension of tumor case sample n°9, the Age. Thus, each row in this feature matrix represents one instance and can be written as five-dimensional column vector $x \in \mathbb{R}^{1*5}$, $X = [x_1, \quad x_2, \quad x_3, \quad x_4, \quad x_5]$.

Each feature dimension is a 12-dimensional row vector, $\boldsymbol{X}^i \in \mathbb{R}^{12*1}$, for example:

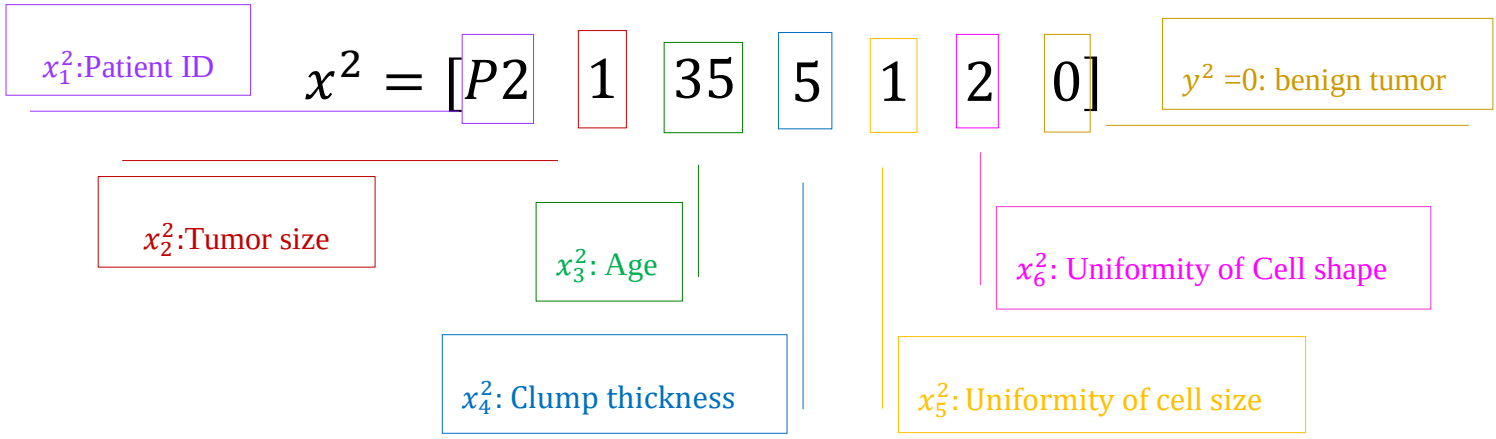
$$X_j = \begin{bmatrix} x_j^1 \\ x_j^2 \\ \dots \\ x_j^{11} \\ x_j^{12} \end{bmatrix}$$

Similarly, we will store the target variables (here: class labels) as a 12-dimensional column vector $y = \begin{bmatrix} y^1 \\ y^2 \\ \dots \\ y^{11} \\ y^{12} \end{bmatrix}, y \in \{0,1\}$

We will represent the final matrix as follows, the matrix contains 12 examples, and we will only take the three first examples:



For example for **the training sample n°2** we will have:



$\boldsymbol{x_1^2}$: Patient ID

$\boldsymbol{x_2^2}$: Tumor size of Patient ID=P2

$\boldsymbol{x_3^2}$: Age of Patient ID=P2...

$\boldsymbol{y^2}$: Tumor of Patient ID=P2 is benign

2.4. Machine Learning algorithms and processes

During this section, we will present the different learning algorithm we used, we will first introduce linear models then the nonlinear ones:

2.4.1. Linear regression

The core idea of linear models (or generalized linear models) is that we model the predicted outcome of interest as a function of a simple linear predictor applied to the input variables.

2.4.1.1. Hypothesis Function

Machine learning algorithms try to map the input X to the output y i.e. they try to find a function f that gives: $y = f(X)$, however this function $f()$ is unknown so the learning algorithm will try to guess this function $h()$ that approximates the unknown $f()$, the goal is to find the final hypothesis that best approximates the unknown target function.

The figure n°7 illustrates the hypothesis function creation process:

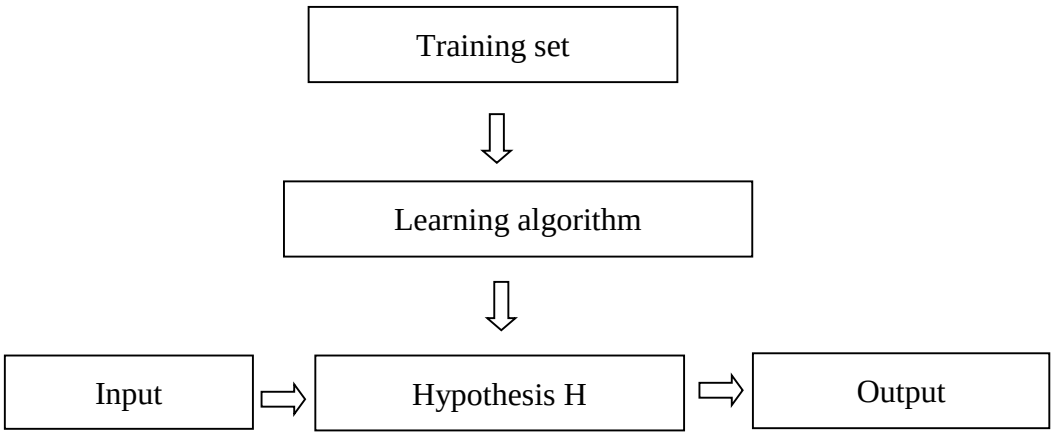
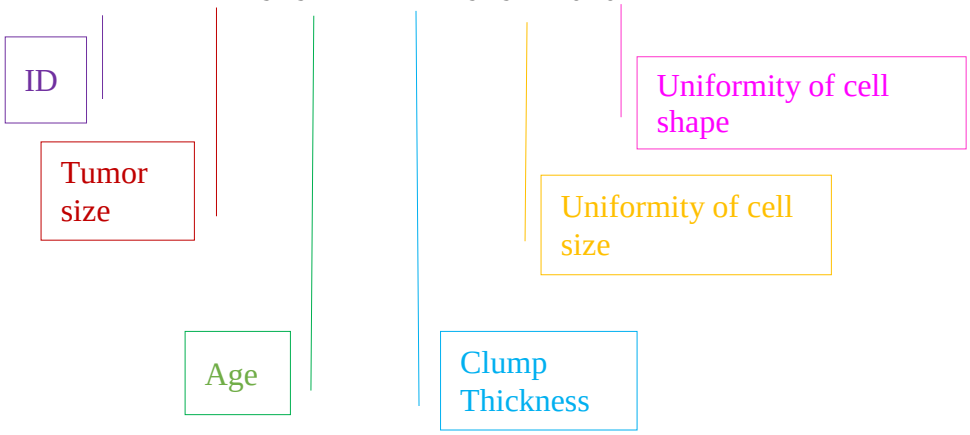


Figure n° 7: Hypothesis function creation process

If the input X is the tumor example features and y is the predicted tumor type, the hypothesis function will look like:

$$h_\theta(X) = \theta_0 + \theta_1 X_1 + \theta_2 X_2 + \theta_3 X_3 + \theta_4 X_4 + \theta_5 X_5 + \theta_6 X_6$$



To generalize the hypothesis function: when we have n features and m examples, we will have:

$$h_\theta(X) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n, \qquad n\ features$$

As a convention we note $x_0 = 0$

$$X = \begin{bmatrix} x_0 \\ x_1 \\ x^2 \\ \dots \\ x_{n-1} \\ x_n \end{bmatrix} \in \mathbb{R}^{n+1}, \theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta^2 \\ \dots \\ \theta_{n-1} \\ \theta_n \end{bmatrix} \in \mathbb{R}^{n+1}, \qquad n\ features$$

So we will end up by:

$$h_\theta(X) = \theta_0 + \theta_1 x_1 + \cdots + \theta_n x_n = \theta^T X, \qquad where\ \theta^T is\ the\ transpose\ of\ \theta \qquad (1)$$

2.4.1.2. Cost Function

The learning algorithm intuition is to find the learning curve that best suits the training data set i.e. minimizing the sum of squared errors over our training set and the learning curve. The sum of squared errors will be the cost function as presented in the equation (2) :

Cost Function

m training examples

$$J(\theta_0, \theta_1, .., \theta_n) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i)^2 \qquad (2)$$

2.4.1.3. Gradient Descent

After defining the cost function, the point here is to minimize this multivariable function i.e.: $\min_{\theta_0,\theta_1,..,\theta_n} J(\theta_0,\theta_1,..,\theta_n)$. So we need to choose $\theta_0,\theta_1,..,\theta_n$ that minimize the colored distances in the figure n° 8:

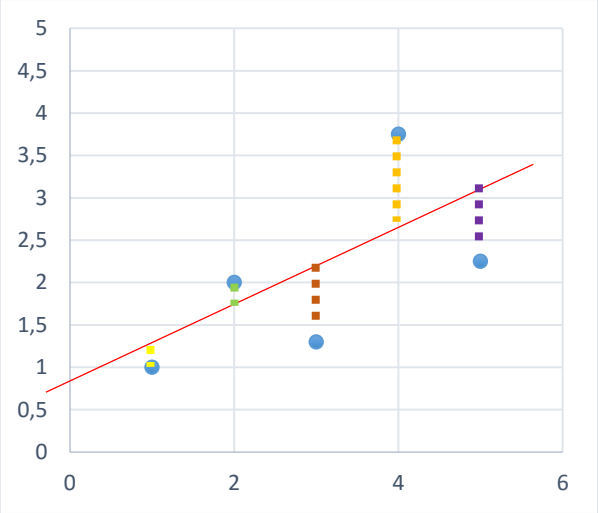


Figure n° 8: Gradient descent intuition

—— : Learning curve

The gradient descent is a method for finding the minimum of a function of multiple variables, its outline will be to:

- Start with some $(\theta_0,\theta_1,..,\theta_n)$
- Keep changing $(\theta_0,\theta_1,..,\theta_n)$ to reduce $J(\theta_0,\theta_1,..,\theta_n)$ until ending up at a minimum

A mathematical representation will be:

- Repeat until convergence:

$$\theta_j = \theta_j - \alpha \frac{\partial}{\partial\, \theta_j} J(\theta_0,\theta_1,..,\theta_n), \text{simultaneously update for every j} = 0,\dots,\text{n}$$

Where the learning rate α will define how fast the gradient will converge, α should not be taken too small neither too large: If α is too small, gradient descent will be slow, if α is too large, gradient descent can overshoot the minimum, it may fail to converge, it even diverges.

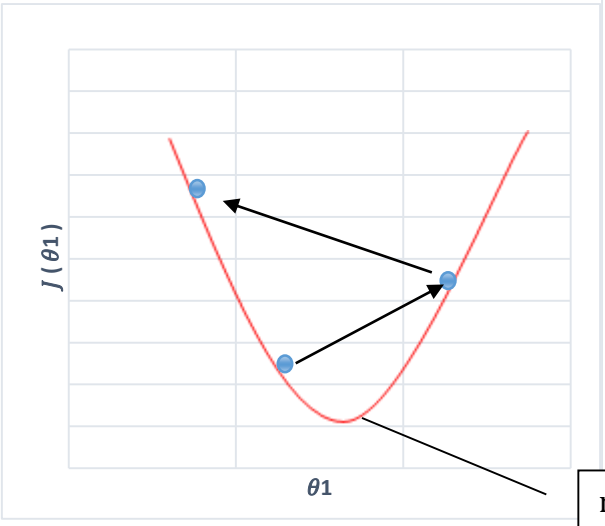


Figure n° 9: α too large

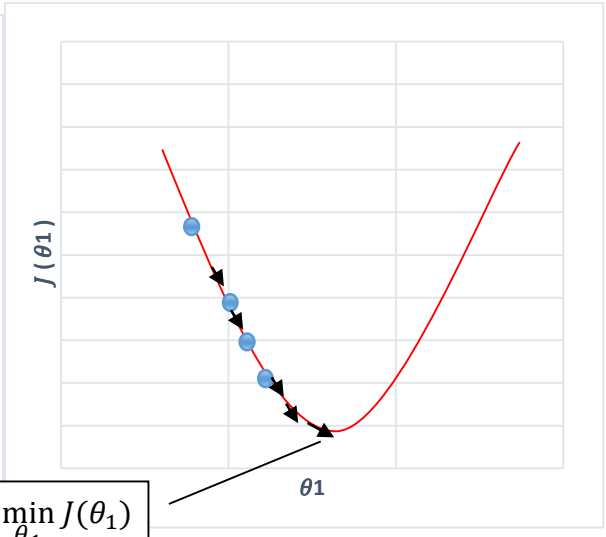


Figure n° 10: α too small

2.4.2. Ridge Regression

Linear regression works well for many problems, but when we apply it to certain machine learning applications, it can run into a problem called "over fitting" and give a poor performance. This is due to the high values of $\theta_0, \theta_1, .., \theta_n$ of the nonlinear model i.e. use of ($\theta_1^2, \theta_2^5, \theta_1^3 * \theta_3^4$, ...).

To shrink the coefficients and have a simpler hypothesis function, we add a regularization parameter which is λ as shown in the equation (3):

$$J(\theta_0, \theta_1, .., \theta_n) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i)^2 + \lambda \sum_{i=1}^{n} \theta_i^2 \qquad (3)$$

The regularization parameter λ controls a **tradeoff** between two different goals:

- Fitting the training set well.
- Keeping the parameter plan small and therefore keeping the hypothesis relatively simple to avoid over fitting.

➡We need to be very careful when setting the parameter λ, because if it is set to an extremely large value, the algorithm result would be under fitting (fail to fit even training data well)

2.4.2.1. Overfitting and underfitting

Both overfitting (high bias) and underfitting (high variance) are negative properties for types of models. The problem of overfitting occurs when a model is perfectly adapted to the input data, rather than generalizing and it has poor predictive performance while underfitting is when the algorithm could not learn all the information contained in the available data set, consequently it is not fit to the data already observed and leads to unuseful generalization.

For example when we have two features x_1 and x_2, we can model these problems by:

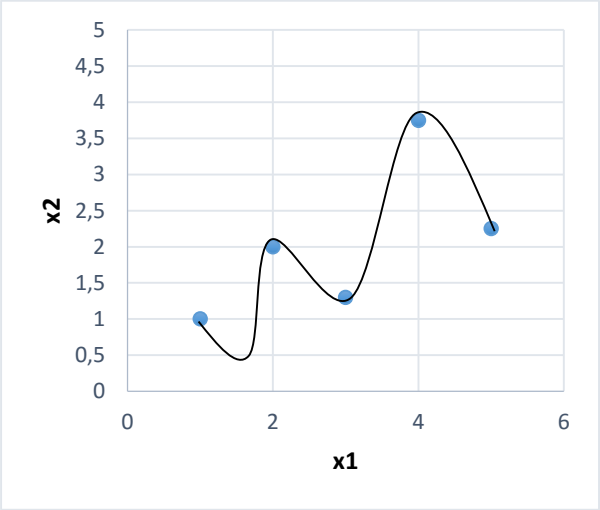


Figure n° 11: Overfitting problem

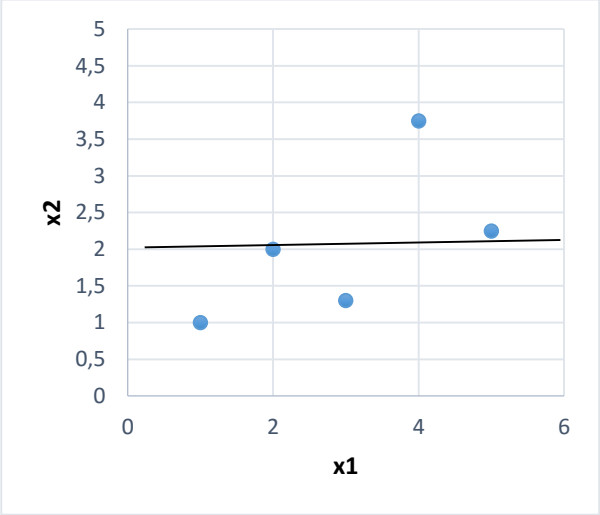


Figure n° 12: Underfitting problem

2.4.3. Multilayer perceptron

Working with non-linear features is a good option, but when we have a lot of features, the number of feature combination will increase greatly.

⇒ **Example**: number of features n=100, using Quadratic Features, we will end up with 3 millions features! O (n^4)

Having this huge number of features, we need to switch to a more sophisticated learning algorithm to deal with more complex problems, above these algorithms we have multilayer perceptron (MLP): MLP is a feedforward artificial neural network model that maps sets of input data onto a set of appropriate outputs. An MLP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. Except for the input nodes, each node is

a neuron (or processing element) with a nonlinear activation function. MLP utilizes a supervised learning technique called backpropagation for training the network. MLP is a modification of the standard linear perceptron and can distinguish data that are not linearly separable

The MLP origins are inspired by the way biological nervous systems work, it tries to mimic the brain which leads to learn how:

- to see,
- to walk,

➡Learns different skills but with only <u>one algorithm</u>!!

2.4.3.1. MLP representation

From biological neuron to artificial neuron

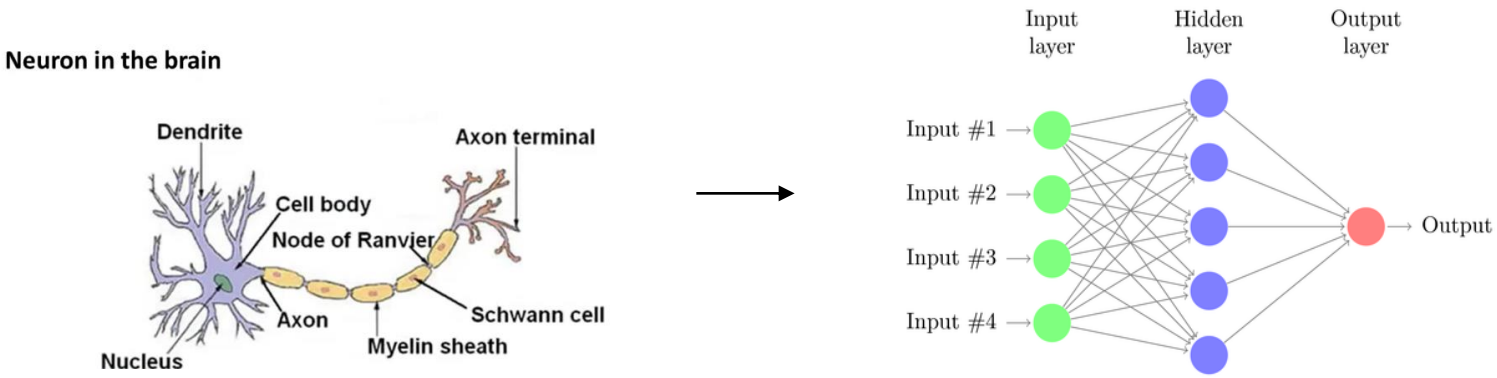


Figure n° 13 MLP representation

Input layer: The MLP first layer, where we put in our features $X = [x_1, \quad x_2, \quad \ldots, \quad x_n]$

Hidden layer: in MLP hidden layer is a layer between input and output layers, we can have zero or more than one hidden layer.

Output layer: The last layer of the MLP that outputs the final value computed by the hypothesis function $h_\theta(X)$

<u>Example modeling</u>:

This MLP example is a computational unit that takes as input $[x_1, \quad x_2, \quad x_3\ ]$, the circles labeled "+1" are called bias units, and correspond to the intercept term and outputs

$h_\theta(X) = f(\theta^T X)$, f is called the activation function. In these notes, we will choose f to be the sigmoid function: $f(z) = \frac{1}{1+e^{-z}}$

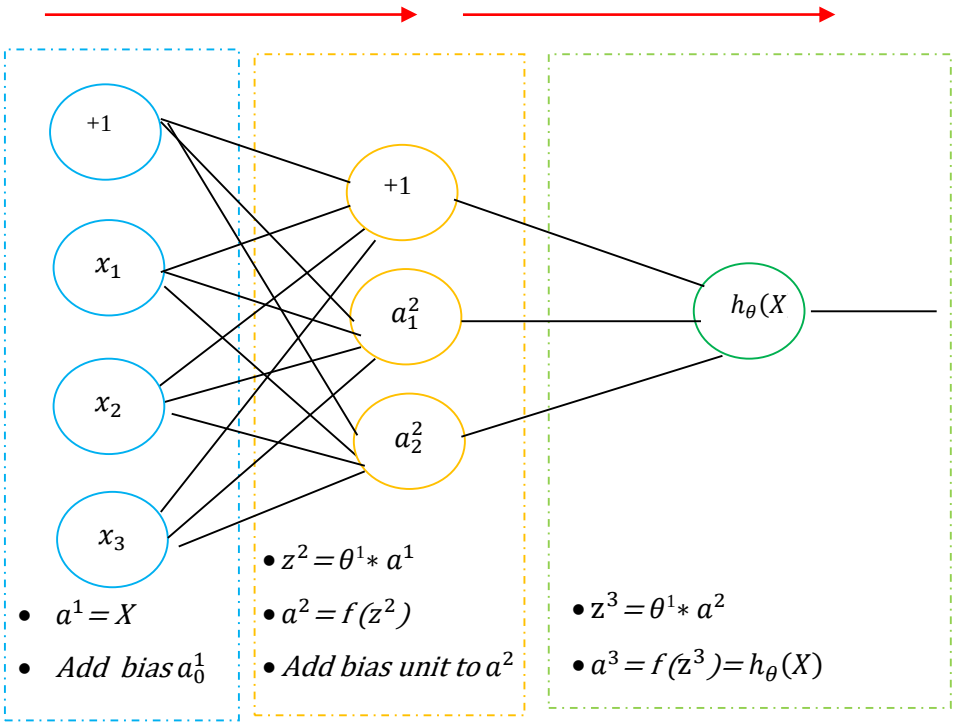


Figure n° 14: Forward propagation

We have 3 **input units** (not counting the bias unit), 2 **hidden units**, and 1 **output unit**. We will keep n_l denote the number of layers in the network; thus n_l = 3 in our example. We label layer l as L_1, so layer $L_1$is the input layer, and layer L_{n_l} i.e. L_3 is the output layer.

We will note

- θ^j: matrix of weights controlling function mapping from layer j to layer j+1
- a_i^j: Activation of unit i in layer j : (meaning output value) of unit i in layer l: j = 1, we also use $a_i^1 = x_i$

2.4.3.2. Forward propagation

In Forward propagation as seen in figure n°14, MLP defines a hypothesis function $h_\theta(X)$ sorted as follows:

- Add bias unit to X
- $a^1 = X$
- $z^2 = \theta^1 * a^1$
- $a^2 = f(z^2)$
- Add bias unit to a^2
- $z^3 = \theta^2 * a^2$
- $a^3 = f(z^3) = h_\theta(X)$, We call these steps **Forward propagation**

2.4.3.3. Cost function

The forward propagation intuition is to finally get the MLP cost function:

$$J(\theta_0, \dots, \theta_n) = -\frac{1}{m}\left[\sum_{i=1}^{m} y_k^i \log(h_\theta(x^i)_k) + (1 - y^i{}_k)\log(1 - h_\theta(x^i)_k)\right] + \frac{\lambda}{2m}\sum_{l=1}^{L}\sum_{i=1}^{s_L}\sum_{j=1}^{s_L} {\theta_{ji}^l}^2 \quad (4)$$

2.4.3.4. Back propagation

MLP is a self-correction learning algorithm based on real output Y, this technique is called backpropagation. It is a method which calculates the error gradient δ for each neuron of a neural network from the last layer to the second one and finally update $\theta_1, \dots, \theta_n$, δ^k *refers to the error in k layer*:

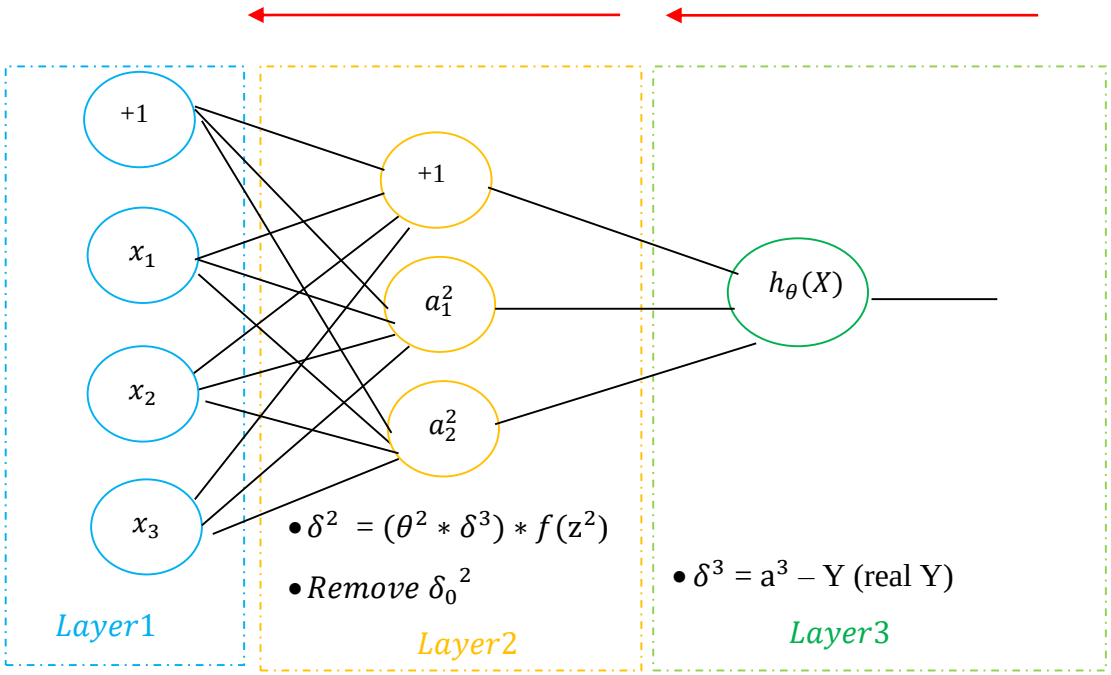


Figure n° 15: Backpropagation

The algorithm for **Backpropagation** will be

$\delta^3 = a^3 - Y$ (real Y)

$\delta^2 = (\theta^2 * \delta^3) * f(z^2)$

$Remove\ \delta_0{}^2$

$\theta^2\_updated = \theta^2\_updated + \delta^3 * a^2$

$\theta^1\_updated = \theta^1\_updated + \delta^2 * a^1$

NB: MLP can learn a non-linear function approximate for either classification or regression.

2.4.4. Support Vector machine

Support vector machines (SVMs) are a set of supervised learning methods used for classification, regression. It is effective in high dimensional spaces i.e. when n: number of features is so high.

Let's take a plan (two-dimensional space) in which are distributed two groups of points. These points are associated with a set of points (+) for y> x and points (-) for y <x. We can find an

obvious linear separator in this example, the line y = x. The problem would be linearly separable.

For more complicated problems, there is generally no linear separation. For example, let's take a plan in which the points (-) are grouped in a circle with points (+) all around: no linear separator cannot properly separate groups: the problem is not linearly separable. There is no separating hyperplan:

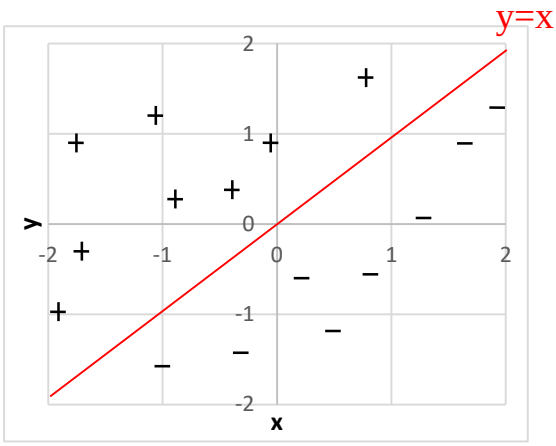


Figure n° 16: Linearly separable problem

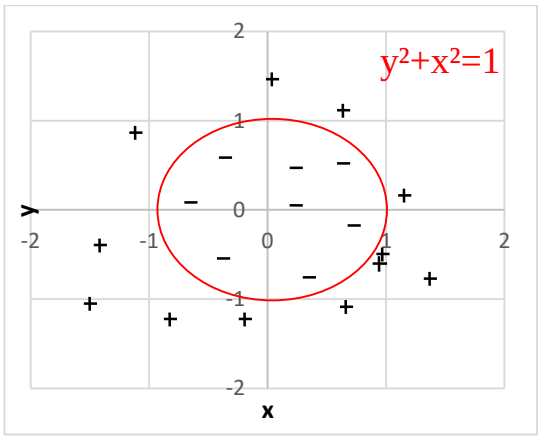


Figure n° 17: Nonlinearly separable problem

Let's consider the case where the problem is linearly separable. Even in this simple case, the choice of separating hyperplan is not obvious. Indeed, there is an infinity of hyperplans separators as shown in the figure n° 18, the learning performance is identical (the empirical risk is the same), but the generalization performance can be very different. To solve this problem, it was proved that there is a unique optimal hyperplan, defined as the hyperplan that maximizes the margin between the samples and the separating hyperplan as seen in the figure n°19

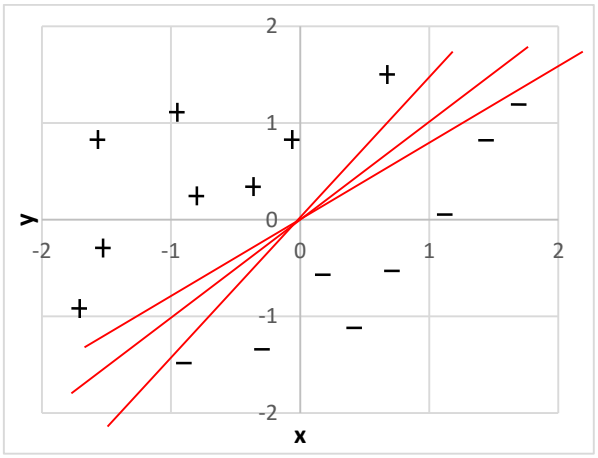


Figure n° 18: Different possible hyperplans

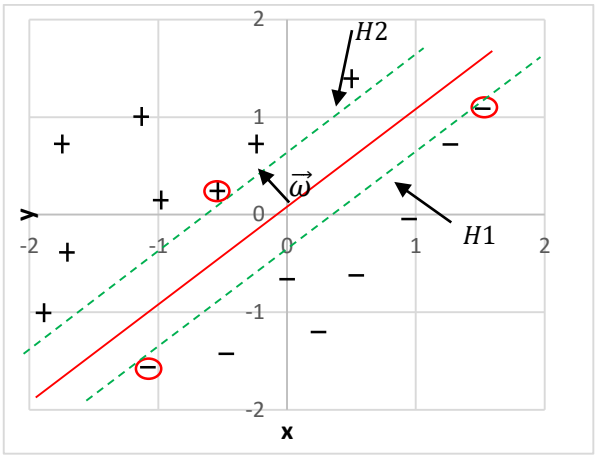


Figure n° 19: SVM intuition

Referring to Figure n°19, implementing SVM boils down to selecting the variables ω and b so that our training data can be described by:

$$x_i * \omega + b \geq +1 \text{ for } y_i = +1$$

$$x_i * \omega + b \geq -1 \text{ for } y_i = -1, \omega\text{: normal to the hyperplane.}$$

These formulas can be combined into:

$$y_i(x_i * \omega + b) - 1 \geq 0 \; \forall i$$

If we just now consider only the points that lie closest to the separating hyperplan, i.e. the Support Vectors (shown in circles in the diagram), then the two planes H1 and H2 (Figure n°19) that these points lie on can be described by:

$$x_i * \omega + b = +1 \text{ for } H_2$$

$$x_i * \omega + b = -1 \text{ for } H_1$$

Geometrically, the distance between these two hyperplans $H_1$and H_2 is $\frac{2}{\|\omega\|}$, so to maximize the distance between the planes we want to minimize $\|\omega\|$.

In this proceeding, gradient descent algorithms for the SVM work directly with the expression (5):

$$min\, f(\vec{w}, b) = \left[\frac{1}{m}\sum_{i=1}^{n} \max(0, 1 - y_i(\omega * x_i + b))\right] \lambda \sum_{i=1}^{n} \|\omega\|^2, m\ training\ examples$$

2.4.5. Random Forest

Random forest (RF) is a new algorithm, it was formally proposed in 2001 by Leo Breiman and Adèe Cutler. This algorithm is part of machine learning techniques. It combines random subspace concepts and bagging. The algorithm of Random forest performs based on multiple decision trees trained on subsets slightly different data, it uses adaptive strategies (boosting) or random strategies (bagging). The idea is to combine or aggregate a large number of models while avoiding overfitting.

The RF algorithms form a family of classification methods that rely on the combination of several decision trees as shown in the figure n°20:

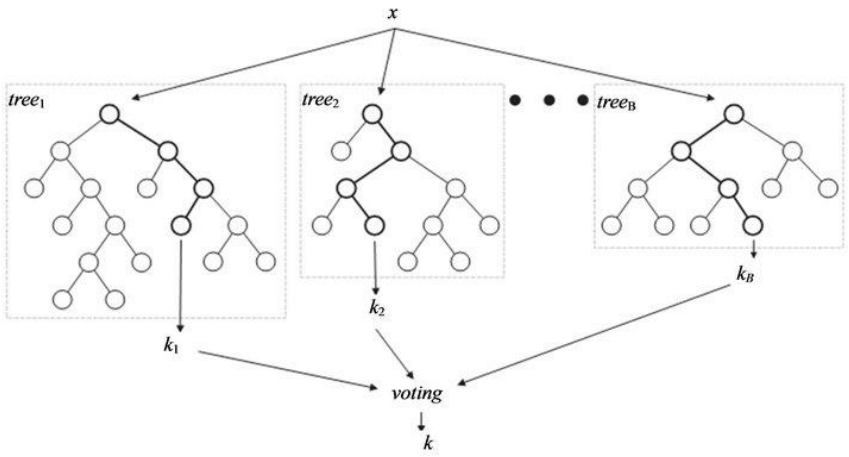


Figure n° 20: General Architect of random forest.

Each decision tree is composed from a random combination of explanatory variables (features), the final decision is the result of voting decision trees previously built. RF is improved Bagging algorithm which uses the chance to improve performance more unstable nonlinear algorithms. It is generally less effective for stable linear methods.

Bagging algorithm follows the next steps:

- y: target vector, $X_0 \dots X_p$: p *training examples*
- Φ: model learned on sample training set : $z = \{(y_1, x_1), \dots, (y_n, x_n)\}$
- We consider a B training set "bootstrap", $z_1, \dots, z_B$ labeled individuals by z random sampling with replacement.
- On each sample, we learn a model Φ_{z_i}

- y is predicted by aggregating the various decisions on each z_i by:

$\Phi(x) = \frac{1}{B}\sum_{i=1}^{B} \Phi_{z_i}(x)$, for quantitative variables

- We predict aggregating different decisions on each z_i by: Φ (x) = Majority decision among $\Phi_{z_i}(x)$ for a qualitative variable.

The purpose of RF algorithm is to make more independent models. This independence will allow to make experts' voting more efficient. This is a very successful approach to large dimensional scale used within the bio-chip, signals, images or curves domains. The major advantage of the algorithm is that it is very simple to implement and has a small digital cost with regard to the obtained performance.

The RF algorithm is as follows:

- y: target vector, $X_0 \dots X_p$: $p\ training\ example$
- Model learned on sample training set : $z = \{(y_1, x_1), \dots, (y_n, x_n)\}$
- For b = 1. . .B (B represents the number of trained trees in the forest)
- Take a random sample z_B with replacement from z
- Estimate a tree z_B with randomized variables
- For the construction of each node of each tree, we consistently derive q from p variables to form the decision associated with the node
- Last in the algorithm, we have B trees that are average or we do vote for regression or classification

➡In general, an optimal choice for q is approximately $q = \sqrt{p}$

Conclusion

Along this chapter we explained the different used algorithm for later classification and regression problems. These algorithms were used on the training data to predict fraudsters' class and their tax adjustment amount. Forward to the next chapter, we will present the used training data and its different features.

Chapter 3: Conceptual Study

Introduction

We dedicate this chapter for the two major phases of the CRISP-DM as introduced in the first chapter: understanding phase and data preparation phase. During data understanding phase, we will start by collection, exploration and verification of the available data, we describe as the database of our project. Next we go through data preparation phase which concerns selecting and cleaning data to build population study and undertaken transformations to fit learning algorithms.

3.1. Technological choice

When launching any project, it is always necessary to identify the technology and the most appropriate tools to fulfill the requirement and meet the need. Therefore, we conducted discussion on the steps to follow and the most used technologies for problems related to machine learning.

3.1.1. IBM SPSS Modeler

Belgium has opted to the SAS undermine and IBM SPSS Modeler solution. So we tried to download these frameworks and see what they offer. We will first talk about: IBM SPSS Modeler [N13] which is a predictive analytics platform that is designed to bring predictive intelligence to decisions made by individuals, groups, systems and the enterprise. It provides a range of advanced algorithms and techniques, including text analytics, entity analytics, decision management and optimization, to help select actions that result in better outcomes. Available in several editions, including a cloud-based version, SPSS Modeler can scale from desktop deployments to integration within operational systems as it offers 4 different packages: SPSS Modeler Personal, SPSS Modeler Professional, SPSS Modeler Premium and SPSS Modeler Gold. IBM SPSS Modeler offers a 30 day trial which is a really good option. The IBM SPSS interface is really simple to use, it only uses drag and drop object which is awesome.

SPSS Modeler is a good option and it also offers Tax fraud detection packages that were unfortunately unavailable in the trial version.

3.1.2. SAS

The second software used by Belgium was SAS. SAS Enterprise Miner data mining industrializes the process to define predictive models and segmentations with unmatched productivity. It enables to integrate absolutely all data sources in your organization and studies, even larger, to get more relevant and more accurate models.

Companies use SAS Enterprise Miner today for strategic issues:

- Minimize customer attrition,
- Reduce risk in terms of credit,
- Anticipate demand,
- Detect fraud

Inconvenient:

- SAS software is expensive and involves high and unpredictable annual costs.
- SAS software is not simple to use, requires specific expertise in SAS programming.
- Need practically to "redeem" the software each year.

3.1.3. Weka

The next tried software was Weka as it was seen in our academic courses at ENSIT. In fact, it is Waikato Environment for Knowledge Analysis is a data mining tool developed at the University of Waikato. Weka is a collection of algorithms and machine learning for data mining. It is a free software released under the General Public License (GNU).

Weka contains:

- 49 data preprocessing tools
- 76 classification and regression algorithms
- 8 classification algorithms (clustering)
- mining algorithms of association rules

It contains data analysis algorithms and forecast modeling with GUI. Through the API Java Data Base Connectivity (JDBC) software can make the connection between the JAVA interface and database.

It allows to train different learning algorithms at the same time and compare results. It is really easy to use thanks to its graphic interface.

Inconvenient: After testing several Weka algorithms, we noticed that the performances are low with large data amounts, the software crashes and does not support all the density we have.

3.1.4. R

R is a language that was designed specifically for statistical computing and graphics. As a result, it has a wide following in the statistics community, which has in turn created a very extensive set of packages for Machine Learning & statistics. The R command line is also very useful when doing interactive plotting & exploration of a dataset. Nevertheless R syntax is a bit uncomfortable and quirky in spots, and thus it's somewhat less pleasurable to code in than Python. R is software in which many modern as well as classics statistical techniques have been implemented. The most common methods to perform a statistical analysis are:

- Descriptive statistics
- Hypothesis testing
- Analysis of variance
- Linear regression methods (single and multiple)

Inconvenient: After conducting tests with R, we noticed that the execution time is really slow.

3.1.5. The Adopted Solution

We will start by a brief conclusion about the previous solutions then we will present the approved solution by the Ministry of Finance

	IBM SPSS Modeler	Weka	R	SAS
Simplicity	☺	☺	☹	☹
Open Source	☹: 11,300 $ per user per year	☺	☺	☹
Machine Learning Algorithm	☺	☺	☹	☺
Support large Datasets	☺	☹	☺	☺
Menu driven	☺	☺	☹	☹
Speed	☺	☹	☹	☺

Table 2: Final Comparison between managing TAX software solutions

Over all IBM SPSS Modeler seems to be the best solution to adopt. It is also ranked among the best dealing with clients' satisfaction. As shown in the figure n°21:

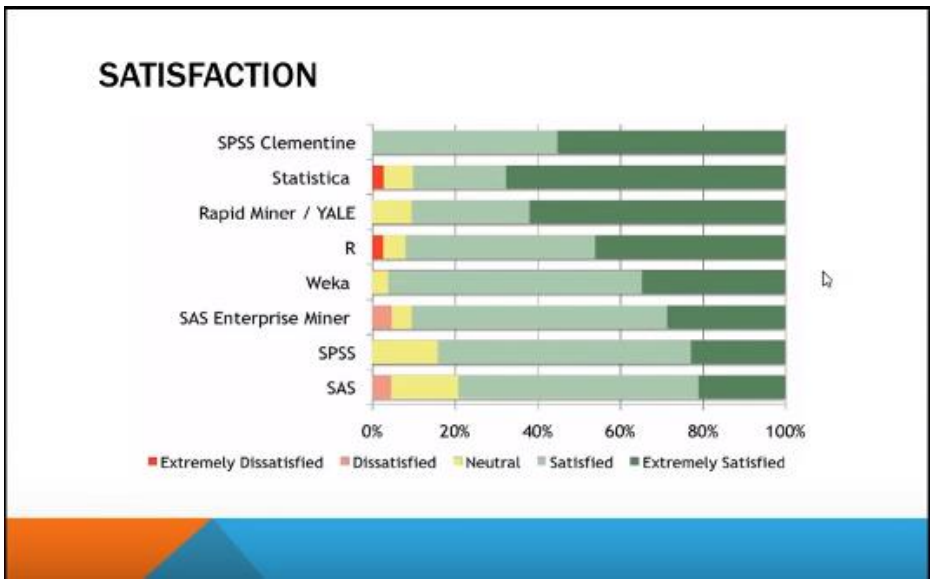


Figure n° 21: Client's Satisfaction on Machine Learning Software

With regard to the budget allocated, the Ministry of Finance could not hold to pay 11,300 $ per year for the IBM SPSS Modeler. So we decided to search for an open source and powerful tool. R looked a good alternative but while it is popular for applied machine learning, it lacks the large development community. R is a specialized tool for machine learning and statistical analysis. With recommendation from the machine learning expert from the UK, Python seemed the most suitable solution.

Python is a general-purpose, widely-used programming language that also has excellent libraries for machine learning applications. It offers the library Sickit Learn library which is the most popular free software for machine learning. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

Scikit-learn is described as "well-maintained and popular" in November 2012. As of 2015, scikit-learn is under active development and is sponsored by INRIA, Telecom ParisTech [N11] and occasionally Google (through the Google Summer of Code). [N12]

3.2. Understanding phase

The data understanding phase concerns the study of available data. This step is vital because it avoids unexpected problems during the next phase (data preparation phase). Data understanding involves data access and exploration through tables and graphs.

3.2.1. Collecting data

To implement this project, we intend to study all available data and extract the most relevant in order to build basis for analysis. To do this we collected a wide range of data that is related to our field of study, including information regarding tax value added tax, corporate income tax, tax on personal income and other data that will be detailed in the next section.

After conducting a study on CIMF decision database, we found that the data concerning our study are the tables belonging to the scheme Risk Analysis. This scheme contains 21 tables with the necessary information for monitoring and control.

Indeed, risk analysis tables help determine the files that are at risk of tax fraud by using some calculated ratios based on the amounts reported by taxpayers and overlap amounts (rent paid, CNSS declaration...). Analysis are always made on the basis of the three years preceding the year of the audit.

In our study, we decided to generate the data that seem most promising for a first analysis. Therefore we have chosen to retrieve 14 of 21 tables as CSV files. The generation of these files was conducted via a PL / SQL script that involves extracting data from the CIMF decisional database, this data include the statements that come from RAFIC and ceiling that come from SADEC

Table n° 3 describes all generated tables:

Table	Description	Records number
Decsoc	Declaration on companies (Income tax), which contains: • Corporations Information: Company Type: P: physical, C: commercial, M: moral...	138 928

	• Different revenue types, • Different deduction types, • Tax benefit rate and taxable profit...	
Agrpay	This table contains Aggregation of taxes' payments.	573 057
Sitfis	This table contains Fiscal Situation Description which contains the year of the fiscal situation and its code for each taxpayer.	1 048 576
Agcnss	Aggregation on CNSS statements (annual) declared payroll and contributions.	530 100
Cntrib (big entreprise)	Contains tax obligations list for each company, it contains also information on the company: its activity and secondary activity, Enterprise type P: Physical C: commercial, M: moral.	1 048 576
Resvap (RsltVerifApp)	Thorough Verification Result table which contains end year of the audit, Tax Code on which the verification was made and its resulting amount.	194 975
Agranx (employeur)	Aggregation on Employer declaration which contains 5 different appendices: • Annex1: total of gross paid wages • Annex2: ax2_honoraires: Fees paid amount • ax2_loyers: Rent paid total • Annex3: Paid movable capital Total (interest on bank accounts) • Annex5: Market Total • Annex6: Recovered cash amount	1 048 576
Actagr (immobilier)	It is an aggregation of registered actions, it contains the actual purchase or sale year, taxpayer's quality (purchaser, vendor, partner, under missionary project owner, lessee, lessor ...) and duty well amount.	1 048 576
Agrtva	This table contains different turnover types (turnover subject to 6%, revenue subject to 18% ...), different	412 575

	purchase bases, different deducted VAT, refunded VAT, VAT on transport ...).	
Dectva	This table contains all information regarding VAT, including the identifier of each taxpayer, the year, amount of VAT on sales, total amount of VAT (the amount that the taxpayer must pay), amount of paid VAT, VAT initial credit, VAT final credit, amount of preliminary recovery and depth adjustment amount.	752 580
Asinda	This table contains customs transactions, including the Import, Export, and the amounts approved.	263 166
Nomimp	This table contains tax code, tax wording, tax short text, tax category (A: annual tax, M: monthly tax, T: quarterly tax).	34
Decird	This table shows Income tax for Physical Persons, P (individual) or C (commercial), it contains income type, local business and export turnover, Turnover sum TTC, accounting income amount, taxable income amount, deduction for export and other deduction.	171 700
Decirp	This table shows personal income tax - it is the detail of "Decird" table, it contains different income types (taxable net income, revenues from securities, exempt revenues ...), different result tax types, salaries wages and pensions, allowances, benefits, withholding, different turnover types and associates shares...	171 456

Table 3: Tables' Description

After collecting the data, we passed all generated files to a Scala script to clean unuseful spaces. This operation helped us a lot because we gained a lot of storage space, For example a file with 17 GO size has become a 0.61 GO size file and the difference was obviously noticed when reading files and when running algorithms.

```
C:\Program Files (x86)\scala\bin>scala C:\Users\sameh\ifamgroup\scala\data-cleaner.sc

C:\Program Files (x86)\scala\bin>scala C:\Users\sameh\ifam-data-exploration\scala\data-cleaner.sc
cleaning file C:\Users\sameh\Desktop\Final Data\big_agrpay.csv TO  C:/Users/sameh/Desktop/Clean Data/big_agrpay.csv
cleaning file C:\Users\sameh\Desktop\Final Data\big_agrtva.csv TO  C:/Users/sameh/Desktop/Clean Data/big_agrtva.csv
cleaning file C:\Users\sameh\Desktop\Final Data\big_asinda.csv TO  C:/Users/sameh/Desktop/Clean Data/big_asinda.csv
cleaning file C:\Users\sameh\Desktop\Final Data\big_decird.csv TO  C:/Users/sameh/Desktop/Clean Data/big_decird.csv
cleaning file C:\Users\sameh\Desktop\Final Data\big_decirp.csv TO  C:/Users/sameh/Desktop/Clean Data/big_decirp.csv
cleaning file C:\Users\sameh\Desktop\Final Data\big_decsoc.csv TO  C:/Users/sameh/Desktop/Clean Data/big_decsoc.csv
cleaning file C:\Users\sameh\Desktop\Final Data\cnss_test.csv TO  C:/Users/sameh/Desktop/Clean Data/cnss_test.csv
```

Figure n° 22: Cleaning Tables

3.2.2. Data Exploration

In this section, we begin the mining stage of seeking usable information in a large data set. We also seek to identify missing data and noise.

Given that VAT is the most important tax in our study, we chose in the first place, to explore the data file "dectva". We plotted the histogram of VAT revenue to get an idea on the distribution of sales subject to VAT for each year

To explore the available data we worked on the 2011's data. In fact in 2011 as seen in the figure n° 23, tax adjustments amounts were so high, this can help us a lot study fiscal fraud cases. We took only taxpayers with positive tax adjustment

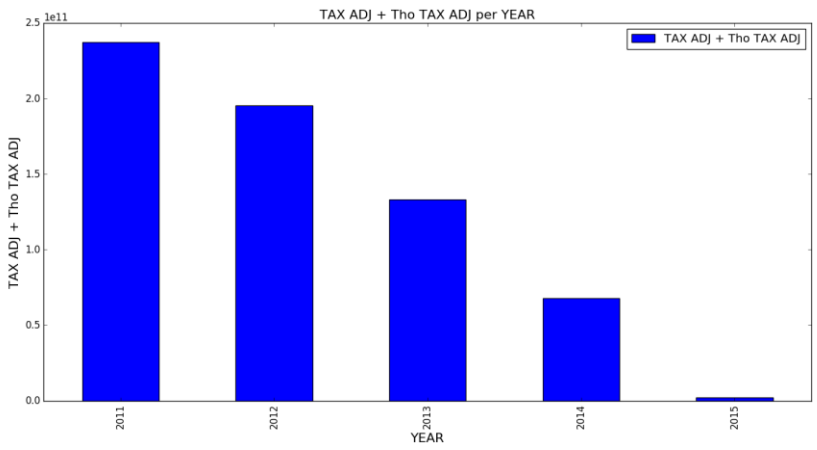


Figure n° 23: Amount of (tax adjustment+ thorough tax adjustment) per year

Given that fraud rate is higher in 2011, we chose to limit our study, first place to this year. We selected taxpayers with preliminary or thorough adjustments greater than zero which means we have selected carrier information data. Indeed the taxpayer population with preliminary or thorough adjustments equal to zero do not interest us because it adds nothing to our learning

algorithms. Our objective is to teach our algorithm fraudulent conduct of taxpayers, so we need to study the population with fraud.

After selecting our dataset that represents observation history, we passed to the verification phase of data quality. We met many problems regarding the form of data, including the types of incorrect data that suffered from transformations during the export process from decision database, but we remedied these problems by python scripts and / or PL / SQL. We have normalized all quantitative variables and we represented scatter plots also known as correlation diagrams between all variables. These diagrams show how a variable is assigned with another. The relationship between the two variables is called their correlation.

The correlation relationship is as follows: Given two quantitative variables x and y, describing the same set of units. We say there is a relationship between x and y if the assignment terms of x and y is not done randomly, i.e. whether x values depend on the values of y or values y depend on x values. To say that y depends on x means that knowing x values, we can predict y values. In other words, if y depend on x, we can find a function f such that

$$y = f(x) \qquad (6)$$

Figure n°24 shows correlation between two variables x and y:

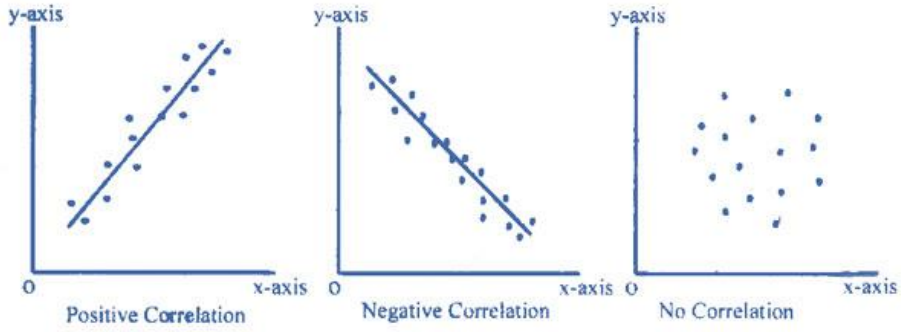


Figure n° 24: Correlation between variables descriptive diagram

When looking at the “Pairplot” of normalized data of 2011 we clearly see outlier making difficult to see or interpret the relation between variables. We barely see the data dots as shown in the figure n°25:

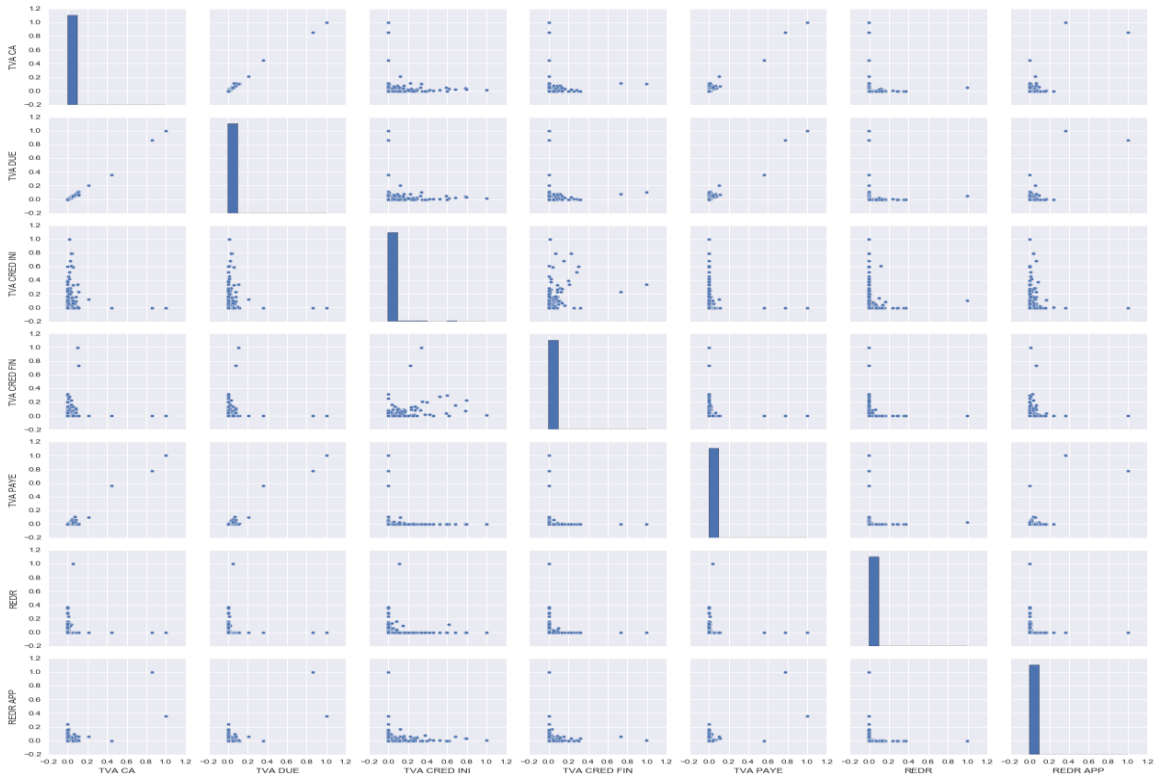


Figure n° 25: Pairplot of normalized data 2011

To resolve this issue, grouping each company by turnover might give us better insight. We will follow similar grouping in 5 classes:

TYPE	TURNOVER
XLARGE	[25 000 TND, Infinity [
LARGE	[10 000 TND, 25 000 TND[
MEDUIM	[1 000 TND, 10 000 TND [
SMALL	[100 TND, 1 000 TND [
MINI	[0, 100 TND [

Table 4: Companies' Classification

The figure n° 26 shows the quality of data plot for Class: Large has improved and it is more legible as we can clearly see data dots. This grouping will help a lot to sort out a better prediction quality

Figure n° 26: “Pairplot” of normalized 2011 data for Class: Large

We used the Describe command that generates various summary statistics for each class, the figures from 27 to 31 show the output for each class:

```
xlargeTaxAdjusted2011.describe()
```

	TVA CA	TVA DUE	TVA CRED INI	TVA CRED FIN	TVA PAYE	REDR	REDR APP
count	5.800000e+01	5.800000e+01	5.800000e+01	5.800000e+01	5.800000e+01	5.800000e+01	5.800000e+01
mean	1.041696e+08	1.768478e+07	4.398206e+05	4.334657e+05	6.613469e+06	4.442811e+04	4.727279e+05
std	2.037395e+08	3.617174e+07	7.642886e+05	1.243903e+06	2.124980e+07	2.788725e+05	1.737304e+06
min	2.583350e+07	2.456134e+06	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00
25%	3.404169e+07	5.375389e+06	0.000000e+00	0.000000e+00	1.146880e+05	0.000000e+00	8.721214e+03
50%	5.266577e+07	9.276716e+06	6.864756e+04	0.000000e+00	8.649934e+05	0.000000e+00	8.179966e+04
75%	7.891731e+07	1.343245e+07	4.542531e+05	3.346124e+05	2.656079e+06	0.000000e+00	2.363194e+05
max	1.182042e+09	2.127676e+08	3.020598e+06	7.562758e+06	1.199378e+08	2.107304e+06	1.257774e+07

Figure n° 27: Xlarge Class Description

```
largeTaxAdjusted2011.describe()
```

	TVA CA	TVA DUE	TVA CRED INI	TVA CRED FIN	TVA PAYE	REDR	REDR APP
count	81.000000	81.000000	81.000000	81.000000	81.000000	81.000000	81.00000
mean	15337759.324296	2571398.489815	218086.562074	103127.478543	383440.721173	6899.044284	81052.50321
std	4195254.972041	806133.468267	559078.722976	289327.517001	491022.341886	55719.711242	133270.10178
min	10023270.978000	862983.412000	0.000000	0.000000	0.000000	0.000000	0.00000
25%	11972251.998000	1924683.252000	0.000000	0.000000	12658.944000	0.000000	5623.96600
50%	14766124.471000	2425797.515000	411.915000	0.000000	186962.021000	0.000000	19160.46500
75%	17893670.689000	3096775.533000	144578.017000	79098.620000	511172.859000	0.000000	97473.27600
max	24310700.613000	4397409.873000	3803549.765000	2176032.630000	2014427.710000	499877.064000	630599.79000

Figure n° 28: Large Class Description

```
meduimTaxAdjusted2011.describe()
```

	TVA CA	TVA DUE	TVA CRED INI	TVA CRED FIN	TVA PAYE	REDR	REDR APP
count	337.000000	337.000000	337.000000	337.000000	337.000000	337.000000	337.000000
mean	1424341.766976	240970.689000	33219.644042	20614.875742	34116.486932	9173.175558	55959.872318
std	279667.355146	62424.698055	94880.080292	53169.659000	60661.101829	33403.308037	118535.777432
min	1003795.263000	70568.768000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	1198488.530000	200397.973000	0.000000	0.000000	0.000000	0.000000	2196.161000
50%	1360702.668000	235435.443000	1690.587000	784.660000	5288.293000	0.000000	12579.152000
75%	1668449.787000	289071.313000	29718.520000	21012.206000	40179.912000	0.000000	43850.630000
max	1991649.167000	529458.853000	967276.396000	691736.903000	344844.665000	345153.346000	815819.839000

Figure n° 29: Medium Class Description

```
smallTaxAdjusted2011.describe()
```

	TVA CA	TVA DUE	TVA CRED INI	TVA CRED FIN	TVA PAYE	REDR	REDR APP
count	1864.000000	1864.000000	1864.000000	1864.000000	1864.000000	1864.000000	1864.000000
mean	362223.666788	60609.088998	14220.791923	13602.179675	8145.803986	5622.055162	27631.734295
std	239968.205100	44221.427115	42918.424278	42791.109284	17746.210961	29947.220524	88528.569951
min	100065.143000	6121.080000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	165082.764750	26286.164000	0.000000	0.000000	0.000000	0.000000	0.000000
50%	274409.200500	44728.466500	799.985500	698.395000	839.627500	0.000000	6951.933000
75%	520755.565750	86006.506250	9267.883250	9102.481750	8047.524500	783.147750	24748.828250
max	994833.267000	536096.544000	644239.504000	517056.998000	171301.579000	785823.198000	1789310.921000

Figure n° 30: Small Class Description

```
miniTaxAdjusted2011.describe()
```

	TVA CA	TVA DUE	TVA CRED INI	TVA CRED FIN	TVA PAYE	REDR	REDR APP
count	4691.000000	4691.000000	4.691000e+03	4.691000e+03	4691.000000	4691.000000	4.691000e+03
mean	21608.050507	3594.544137	5.186776e+03	5.406971e+03	802.349326	5062.250084	1.070479e+04
std	26091.252500	14387.707699	5.621003e+04	6.410600e+04	2036.150787	26320.940826	7.756077e+04
min	0.000000	0.000000	0.000000e+00	0.000000e+00	0.000000	0.000000	0.000000e+00
25%	0.000000	0.000000	0.000000e+00	0.000000e+00	0.000000	0.000000	0.000000e+00
50%	10838.800000	1303.105000	0.000000e+00	1.800000e+00	0.000000	146.441000	0.000000e+00
75%	34503.391500	4396.226000	4.693810e+02	5.959215e+02	890.775000	2601.547000	3.658653e+03
max	99807.712000	607164.402000	2.291172e+06	2.426562e+06	65600.891000	739854.225000	3.043912e+06

Figure n° 31: Mini Class Description

In the above, we can see the mean of each value, the minimum and maximum value, standard deviation and the mean value of 25, 50 and 75% of the population for each class.

3.3. Data preparation

Raw data rarely comes in the form and shape that is not necessary for the optimal performance of a learning algorithm. Thus, the preprocessing of the data i.e. getting data into shape is one of the most crucial steps in any machine learning application. Data preparation represents an estimated 50% to 70% of the time and effort of any project.

3.3.1. Data selection

In addition to the collected 14 tables from BI system, we added additional data which is the criteria. These criteria define risk analysis ratios that identify potential fraudsters using the 14 tables. They are the result of automated audit selection. It is a process by which each taxpayer is given a score on the likelihood of the taxpayer evading tax. It requires a good set of criteria and quality data. This method is an iterative process and often does not produce perfect results at the first time.

The criteria were made in collaboration with the CIMF and an expert from Vietnam where taxpayers are divided into groups and their national averages are calculated for each group of taxpayer. Each taxpayer is compared against the average for their group and a score is given based on how far it deviates from the average.

We calculated these criteria and added them to our dataset as columns i.e. features for example for criteria 1 we added a feature named C1. Below is the list of developed criteria:

Criteria	Description
Criteria 1	Gross Profit Margin:
Criteria 2	Net Profit Margin
Criteria 3	Wage Intensity
Criteria 4	VAT Turnover Compared to Domestic Turnover
Criteria 5	Consecutive years of VAT credit with no audit
Criteria 6	Consecutive years of Net Loss with no audit
Criteria 7	Several Years where Turnover < Imported Goods
Criteria 8	Income Tax Non-filer
Criteria 9	VAT Non-filer
Criteria 10	Has not been audited for at least 10 years
Criteria 11	Has export turnover, but not claiming export deduction
Criteria 12	3 years of zero turnover and registered > 5 years
Criteria 13	No rent reported, but rental agreement registered
Criteria 14	Did not file December VAT return in 3 years
Criteria 15	Gap between last audit result and what was paid in the last 3 years
Criteria 16	Claiming export benefits, but bought buildings
Criteria 17	Registered sale greater than Fiscal result
Criteria 18	Cash transactions higher than turnover
Criteria 19	Accounting Result is Positive and Fiscal Result is Negative
Criteria 20	Imputations' total compared to CA tot TTC
Criteria 21	TCL plate compared to au CA annual Total TTC

Table 5: Criteria Description

3.3.2. Data Cleaning

When cleaning the data, we need to deeply inspect the problems of chosen data with the aim of consideration in the analysis.

3.3.2.1. Missing data

Missing data involves operations on data where we exclude rows or characteristics, or insert an estimated value. Missing data can be due to Taxpayers who have not completed the online questionnaire or mistakes done by cashiers. The Ministry of Finance could ask again those taxpayers to complete the questionnaire, but this cost more time and money than have been spent. On the other hand, we can model the differences between taxpayers who responded and

those who have not responded. If these two sets have similar habits, questionnaires missing parts are less problematic.

In our case, we tried to replace missing data which have been determined in the previous data understanding phase, with significant value. This requires the intervention of a tax expert. For this reason, we followed the recommendations of the trade:

- When calclating Criteria 1 (Moral Persons): If exportation turnover in "Decsoc" table is missing we take its value from exportation turnover in "Agrtva" table for the same taxpayer in the same year. If it is also missing in "Agrtva" we set 0.
- When calculating the criteria, Nan or infinite value as a result of a mathematical division is replaced by 0.
- In some missing data cases that lead us do some data transformation that we will detail in transformation section
- In criteria 21 if TCL amount is missing in "Agrpay" table the value for the column C21 for that taxpayer in that year is filled with 100.

3.3.2.2. Errors in data

Errors found during the exploration process can be corrected in this section. Poor wording of questionnaire items can greatly affect data quality. As for missing questionnaires, this is a delicate problem, because either money or time are needed to collect answers to a substitution question. In problem areas, sometimes it is better to make step backward to the selection process and filter these elements further analysis, for example:

- When having a line that only contains taxpayer id and has no values for other columns, this example should be deleted
- We also corrected the inconsistencies: For example in taxpayer category in "Agrpay" table is value should be "M "or "P" or "C" or "E" or "N", but we found values like "2", so we deleted this line
- When we have a numeric value but instead we find a random string or character, we replace these inconsistencies by a 0.
- In "Income tax credit" in "Decsoc" table, values should "P" or "B" but we found values in lower case "p" and "b" so we had to transform them to upper case characters, because

they are considered as two different values by the learning algorithm and when calculating the criteria.

3.3.3. Data Integration

Having multiple data sources for the same set of business questions is a becoming usual. For example, for the same set of taxpayers, we can access data on their categories, as well as data on their purchases, sales ... If these datasets use the same unique identifier, we can merge them using this key field. The main method for data integration is:

Adding data, which involves the integration of multiple data sets with similar attributes but different records. These data are integrated according to a same field (in our case Tax payer ID and year).

After joining the 14 tables and the generated criteria, we noticed that some explanatory variables affect only a specific category of taxpayers, resulting in the appearance of empty fields. We found more certain criteria concern “Moral Persons” category and others only concern “Physical Persons” category of taxpayers. This observation lead to divide our data set into two category, the category of “Moral Persons”: companies and the category of “Physical Persons”: traders, this partition was proven to be the most effective solution to solve this problem. So in the end, we got two different files: A file that contains all tables and criteria for “Physical Persons” and another file for “Moral Persons”.

3.3.4. Transformations

Before creating a model, it is useful to check whether certain tables require the application of particular transformations to get the data in the right format. In our case, our goal was to have a unique training example per taxpayer per year. In order to perform this condition, we did several transformations:

3.3.4.1. Conversion to dinars

The amounts in the 14 extracted tables were in millimes, so we transformed them into dinars to have a better legibility and reduce the number of zeros when performing calculations.

3.3.4.2. Feature scaling

In order to have a better performance in learning machine algorithms and better results, feature scaling or data normalization is the key. It is a method used to standardize the range of independent variables or data features, we carry it out real and integer values, we applied the following formula:

$$A = (A - A.mean)/(A.max - A.\min) \quad (7)$$

3.3.4.3. Tables' transformations

During this section, we will present all performed transformations for each table:

- **Sitfis Table**

The problem faced with Sitfis table was for each obligation code, the taxpayer id and year are repeated which oppose our goal to have a unique training example per taxpayer per year

ID	ANNEE	OBL_CODOBL	SIT_DEFDEP
9	2015	08	RRRRRRRRRRR
9	2015	12	NNNNNNNNNNN
9	2015	10	RRRRRRRRRRR
9	2015	02	D
9	2015	04	RRRRRRRRRRR
9	2015	05	NNNNNNNNNNN
9	2015	13	NNNNNNNNNNN
9	2015	14	NNNNNNNNNNN

Figure n° 32: Original "Sitfis" Table

Pivot Transformation

To resolve this problem, we had to do a pivot command to transform each "*OBL_CODOBL*" code in columns and have in index taxpayer id and year.

The resulting table is shown in the figure n° 33:

02	03	04	05	06	07	08	10
R	NNN	NNNNNNNNNNNN	NNNNNNNNNNNN	RRRRRRRRRRRR	RRRRRRRRRRRR	RRRRRRRRRRRR	NNNNNNNNNNNN
R	NNN	NNNNNNNNNNNN	NNNNNNNNNNNN	RRRRRRRRRRRR	RRRRRRRRRRRR	RRRRRRRRRRRR	NNNNNNNNNNNN
R	RRR	RRRRRRRRRRRR	NNNNNNNNNNNN	RRRRRRRRRRRR	RRRRRRRRRRRR	RRRRRRRRRRRR	RRRRRRRRRRRR

Figure n° 33: Pivot: Transforming "*OBL_CODOBL*" code in columns

Encoding Transformation

Categorical values are not accepted in learning algorithms so we need to encode each categorical feature, "*OBL_CODOBL*" code is a categorical feature. Before reaching the encoding step we need to understand what it refers to. The "*OBL_CODOBL*" code can be annual or quarterly or monthly referring to the table "Nomimp", where R N and D describes the fiscal situation:

⇒ **Example:** 02 Code is annual, we will have only one letter

03 code is quarterly: we will have 3 values: RND, where R refers to the first quarter, N: refers to the second quarter and D refers to the third quarter

04 code is monthly NRNNNNNNNNNN: The first letter refers to the first month: January, second letter refers to February …

So we decided to have 1 column for annual codes, 3 columns for each quarter in quarterly codes, and 12 columns for each month in monthly codes as shown in the figure n° 34:

02-IR-A-Annuel	03-AP-T-Tri1	03-AP-T-Tri2	03-AP-T-Tri3	04-TCL-M-01	04-TCL-M-02	04-TCL-M-03	04-TCL-M-04	04-TCL-M-05	04-TCL-M-06	04-TCL-M-07	04-TCL-M-08	04-TCL-M-09	04-TCL-M-10	04-TCL-M-11	04-TCL-M-12
R	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N

Figure n° 34: Division of each code by type: annual, quarterly, monthly

After creating these columns we encoded each letter as follows:"R":"2","N":"1","D":"3" and missing values with "0", to end up with a final "Sitfis" table that looks like:

02-IR-A-Annuel	03-AP-T-Tri1	03-AP-T-Tri2	03-AP-T-Tri3	04-TCL-M-01	04-TCL-M-02	04-TCL-M-03	04-TCL-M-04	04-TCL-M-05	04-TCL-M-06	04-TCL-M-07	04-TCL-M-08	04-TCL-M-09	04-TCL-M-10	04-TCL-M-11	04-TCL-M-12
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Figure n° 35: Final "Sitfis" Table

- **Agranx Table (big employeur)**

In "Agranx" table the column «*AGX_TYPOCC* » can be M: minus: declarer or P: Plus, beneficiary. This code enables to know if the tax payer has paid or has been paid different amount types.

AGX_TYPOCC	ID	ANNEE	AX1_REVIMP	AX1_AVGNAT	AX1_BRTIMP	AX1_REVINV	AX1_MNTRET	AX1_NETSRV	AX2_HONORA
P	12765	2010	0.0	0.0	0.0	0.0	0.0	0.0	13151600.0
P	12765	2011	0.0	0.0	0.0	0.0	0.0	0.0	9707333.0
P	12765	2012	0.0	0.0	0.0	0.0	0.0	0.0	14493.0
P	12765	2009	0.0	0.0	0.0	0.0	0.0	0.0	0.0
P	12765	2007	0.0	0.0	0.0	0.0	0.0	0.0	0.0
M	12765	2008	183080710.0	0.0	183080710.0	0.0	22485817.0	160594893.0	37746786.0
M	12765	2009	186516540.0	0.0	186516540.0	0.0	23288738.0	163227802.0	32117220.0
M	12765	2010	210600320.0	0.0	210600320.0	0.0	28593575.0	182006744.0	41652256.0
M	12765	2011	203153898.0	0.0	203153898.0	0.0	26289754.0	176864144.0	31674860.0
M	12765	2012	204521310.0	0.0	204521310.0	0.0	26281880.0	178239430.0	38116440.0

Figure n° 36: Original "Agranx" Table

As seen in the figure n° 36, for the same taxpayer, in the same year we have two training examples. To resolve this problem, we need to transform M and P to columns. But the different amounts *AX1_REVIMP, AX1_AVGNAT* … can be M and P, so we decided to divide each amount in tow types

⇒ **Example** *AX1_REVIMP-M*: Wages taxable revenue paid by the taxpayer

AX1_REVIMP-P: Wages taxable revenue gained by the taxpayer

After this transformation, missing values were replaced by 0

ID	ANNEE	AX1_REVIMP-M	AX1_REVIMP-P	AX1_AVGNAT-M	AX1_AVGNAT-P	AX1_BRTIMP-M	AX1_BRTIMP-P	AX1_REVINV-M	AX1_REVINV-P
12765	2006	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
12765	2007	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
12765	2008	183080710.0	0.0	0.0	0.0	183080710.0	0.0	0.0	0.0
12765	2009	186516540.0	0.0	0.0	0.0	186516540.0	0.0	0.0	0.0
12765	2010	210600320.0	0.0	0.0	0.0	210600320.0	0.0	0.0	0.0
12765	2011	203153898.0	0.0	0.0	0.0	203153898.0	0.0	0.0	0.0
12765	2012	204521310.0	0.0	0.0	0.0	204521310.0	0.0	0.0	0.0
12765	2013	269419182.0	0.0	0.0	0.0	269419182.0	0.0	0.0	0.0
12765	2014	194292160.0	0.0	0.0	0.0	194292160.0	0.0	0.0	0.0
12765	2015	211983400.0	0.0	0.0	0.0	211983400.0	0.0	0.0	0.0

Figure n° 37: Final "Agranx" Table

Figure n° 37 shows that we ended up by a unique training example per year and id.

- **Decirp Table & Decird Table**

Encode the categorical feature "*IRP_TYPREV*" into columns:

	ID	ANNEE	IRP_TYPREV
0	37	2013	A1
1	37	2014	A1
2	44	2014	A1
3	50	2013	A1
4	50	2014	A1
5	65	2013	B1

Figure n° 38: Original "Decirp" Table

	ID	ANNEE	A1	B1	B2
0	37	2013	1.0	0.0	0.0
1	37	2014	1.0	0.0	0.0
2	44	2014	1.0	0.0	0.0
3	50	2013	1.0	0.0	0.0
4	50	2014	1.0	0.0	0.0
5	65	2013	0.0	1.0	0.0

Figure n° 39: Transformed "Decirp" Table

- **Cntrib Table (Big_entreprise)**

Transform "*CTR_CATEGO*" and "*CTR_CODTVA*" to columns (same transformation for "Decirp" Table)

ID	CTR_CATEGO	CTR_CODTVA
1	C	N
2	C	N
3	C	N

Figure n° 40: "Cntrib" Original Table

CTR_CATEGO: The possible values for this feature are:

P: Physical Persons,

M: Moral Persons,

C: Commercial.

E: Secondary establishment

N: Public establishment

To know each letter comes from which feature, we added "_original feature name" at the end as shown in figure n° 41:

ID	C_CTR_CATEGO	M_CTR_CATEGO	N_CTR_CATEGO	P_CTR_CATEGO
1	1.0	0.0	0.0	0.0
2	1.0	0.0	0.0	0.0
3	1.0	0.0	0.0	0.0

Figure n° 41: Transforming "*CTR_CATEGO"* into columns

CTR_CODTVA: The possible values for this feature are : A,B,D,M,F,N,P. We added "_original feature name" at the end as shown in figure n° 42:

ID	A_CTR_CODTVA	B_CTR_CODTVA	D_CTR_CODTVA	M_CTR_CODTVA	F_CTR_CODTVA	N_CTR_CODTVA	P_CTR_CODTVA
1	0.0	0.0	0.0	0.0	0.0	1.0	0.0
2	0.0	0.0	0.0	0.0	0.0	1.0	0.0

Figure n° 42: Transforming "*CTR_CODTVA*" feature into columns

- **Decsoc Table**

Transform "*IMS_CRESCP*", "*IMS_CRESFI*", "*IMS_TYPDEC*" to columns (same transformation for "Decirp" Table)

ID	ANNEE	IMS_CRESF	IMS_TYPDEC	IMS_CRESCP
54	2013	P	IS	B
54	2014	P	IS	B
87	2013	B	IS	B

IMS_CRESFI: Possible values are P or B:

ID	ANNEE	P_IMS_CRESFI	B_IMS_CRESFI
54	2013	1.0	0.0
54	2014	1.0	0.0
87	2013	0.0	1.0

Figure n° 44: Transforming "*IMS_CRESFI*" feature into columns

IMS_TYPDEC: Tow possible values are AS and IS

ID	ANNEE	AS_IMS_TYPDEC	IS_IMS_TYPDEC
54	2013	0.0	1.0
54	2014	0.0	1.0
87	2013	0.0	1.0
87	2014	0.0	1.0

Figure n° 45: Transforming "*IMS_TYPDEC*" feature into columns

IMS_CRESP: The tow possible values are P and B

ID	ANNEE	P_IMS_CRESCP	B_IMS_CRESCP
54	2013	0.0	1.0
54	2014	0.0	1.0
87	2013	0.0	1.0
87	2014	1.0	0.0

Figure n° 46: Transforming "*IMS_CRESP*" feature into columns

- **Actagr Table (immobilier)**

Transform "*IMS_CRESCP*", "*IMS_CRESFI*", "*IMS_TYPDEC*" to columns (same transformation for "Decirp" Table)

ID	ANNEE	ACT_QALLIB	AGR_MNTBIN
5	2006	Locataire	4800000
8	2008	Vendeur	25000000
8	2009	Acheteur	50000000
8	2014	Vendeur	80000000

Figure n° 47: Original "Decirp" Table

ID	ANNEE	ACHETEUR	VENDEUR	BAILLEUR	LOCATAIRE	MAITRE
4	2009	0.0	275000000.0	0.0	0.0	0.0
5	2006	0.0	0.0	0.0	4800000.0	0.0
8	2008	0.0	25000000.0	0.0	0.0	0.0
8	2009	50000000.0	0.0	0.0	0.0	0.0

Figure n° 48: Transforming "*ACT_Quallib*" to columns

The "Actagr" table contains yearly amounts of purchase, sale and so on, we noticed that it is not common for a taxpayer to perform these commercial acts as we found lots of zeros. To resolve this problem we decided to add all of the three (03) consecutive years of amounts by act (Bailleur, Acheteur, Vendeur…)

ID	ANNEE	ACHETEUR	VENDEUR	BAILLEUR	LOCATAIRE	MAITRE	SOUMISSION	ADJUCATAIRE	ASSOCIE
4	2011	0	275000000	0	0	0	0	0	0
8	2011	50000000	0	0	0	0	0	0	0
8	2014	0	80000000	0	0	0	0	0	0
8	2015	0	160000000	0	0	0	0	0	0

Figure n° 49:"Actagr" Table with 3 years sum

Conclusion

Now that data is shaped for learning algorithms and it respect the condition of having one unique training example per year and per tax-payer Id and giving that all categorical features are encoded, we can now start the most important part which is modeling, we will detail it in the next chapter.

Chapter 4: Realization

Introduction

During this chapter, we present the used technology, and learning algorithms we have chosen to resolve the problems of our project. For each algorithm, we will detail the modeling part that involves the calibration parameters to optimize the result, this part allows rollback to review possible data preparation and adapt them to the used techniques. We will also detail the evaluation part of determining the quality of prediction or classification algorithm.

4.1. Regression Algorithms

We present in this section algorithms used for regression. The used data is from "dectva" table. Before beginning the regression, we sorted our basic learning by ascending adjustement (preliminary and detailed), to eliminate 20% of outliers, observed during the data exploration phase) that can distort the analysis.

For linear regression, we used the method ***sklearn.linear_model.LinearRegression*** of sickit-learn API with parameters:

fit_intercept = True, it is the regression constant which is simply the value at which the regression line crosses the y-axis (also called, interception). If this value is not specified, the regression line is forced to pass through the origin. This means that all of the predictors and the response variable must be equal to zero at this moment. If set to false, no interceptions will be used.

normalize = True, the regressors (objects that model observations) will be standardized before the regression.

We started with the preliminary prediction of 2011 adjustment for all classes that have been previously defined, the figures from 50 to 54 present θ coefficients, variance and the score that gives an idea about the quality of prediction, when it is close to 1 we have a good prediction. The score variance is calculated as following:

$$explained\ variance = (y, \hat{y}) = 1 - \frac{Var(y-\hat{y})}{Var(y)} \quad (8)$$

With, $\hat{y}$, the predicted target output, y the actual value, Var: variance. These figures also have the curve shape and the mean square error value, the axis x-represented taxpayers belonging to the test database and the Y-axis represented the value of this error.

In figures n°50 and n°51, we can clearly remark that some outliers (the red rectangle) have high error values: about 10 exponent 7, this can prevent us from following the curve shape with small error values, to resolve this issue we can increase the eliminated outliers margin and change it to 40% instead of 20%.

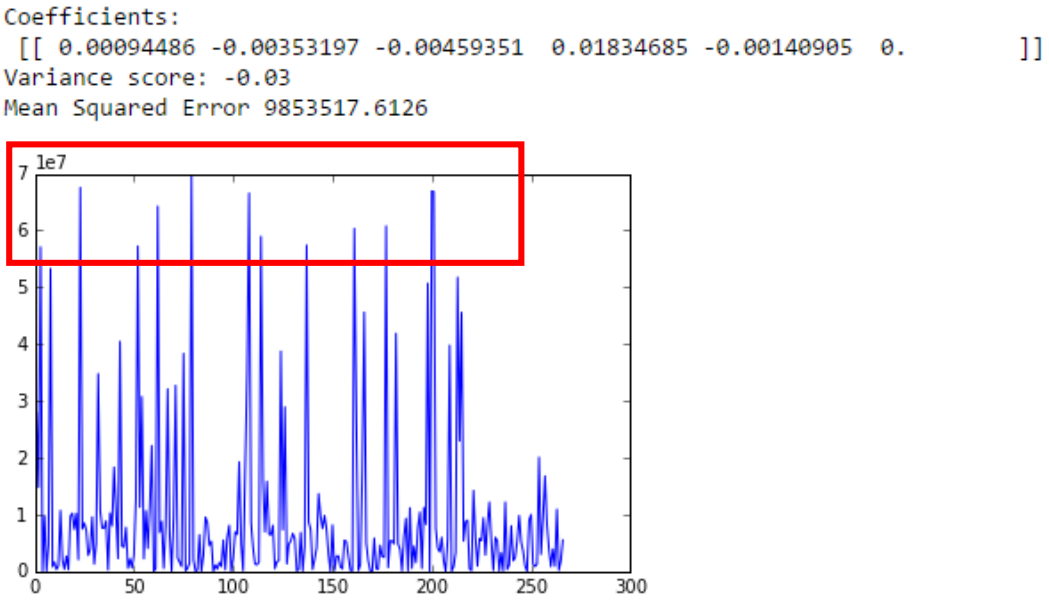


Figure n° 50: Linear regression result of 2011 for class XLARGE

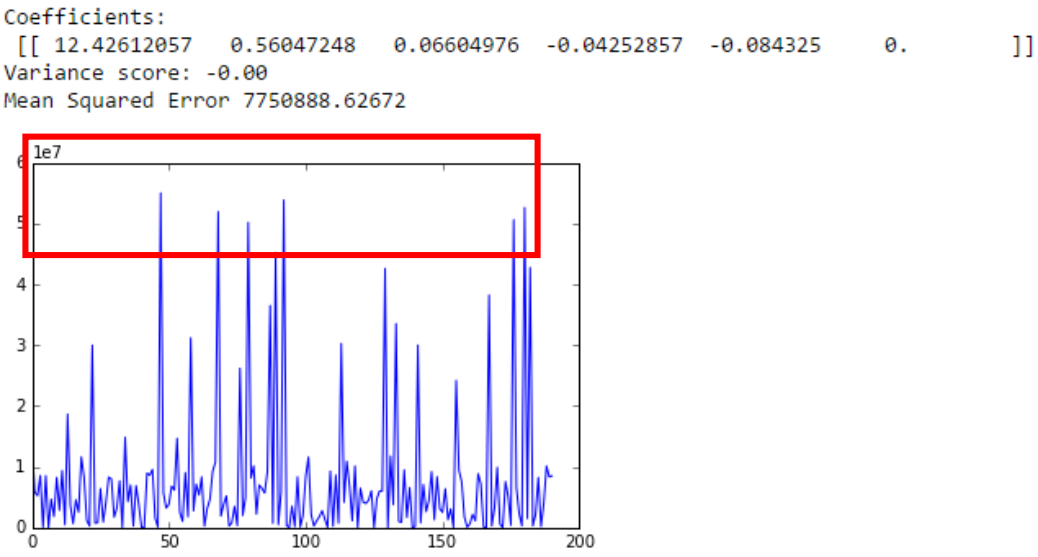


Figure n° 51: linear regression result of 2011 for class MINI

Figures n° 52, 53, 54 show that the error with LARGE, MEDIUM and SMALL class has improved, in fact the error values has decreased but it remains too high compared to normal error values. We can clearly see the curve shape with small error values as well as high error values.

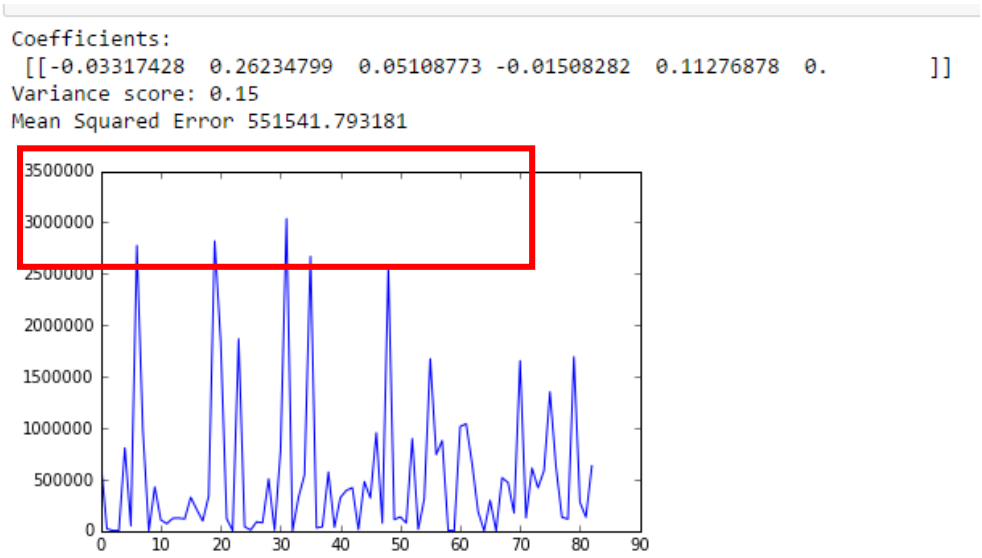


Figure n° 52: linear regression result of 2011 for class LARGE

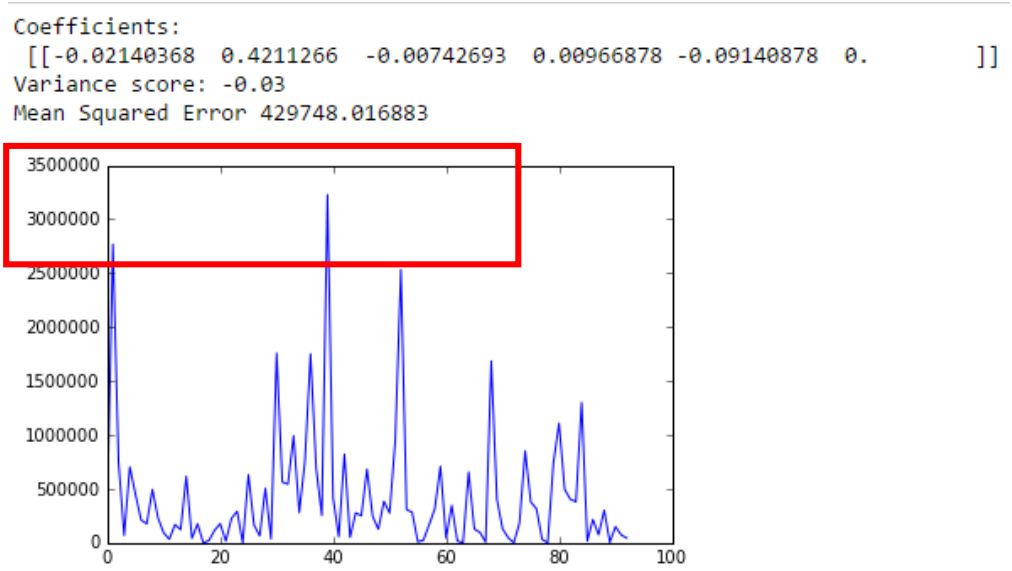


Figure n° 53: linear regression result of 2011 for class MEDIUM

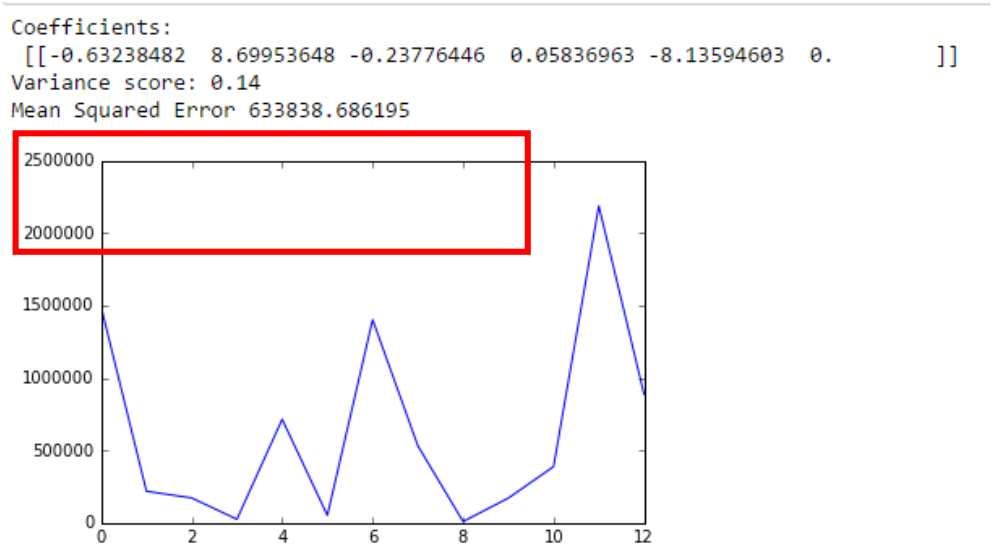


Figure n° 54: linear regression result of 2011 for class SMALL

We found that the error is very high and the score variance is very low, the prediction quality is very low. However, we tried to apply the model prediction of 2011 over other years, namely the years 2012, 2013, 2014 and 2015. The prediction results are shown in Table n°6:

YEAR	CLASS	VARIANCE	MSE
2012	MINI	0	7.295968e+14
2012	SMALL	-1.9	1.310519e+13
2012	MEDIUM	-0.20	2.091033e+13
2012	LARGE	0.13	8.197057e+13
2012	XLARGE	-0.31	3.038605e+16
2013	MINI	-15.37	3.183825e+15
2013	SMALL	0.39	5.205429e+13
2013	MEDIUM	0.03	1.149427e+14
2013	LARGE	-0.01	1.149449e+14
2013	XLARGE	0.02	2.966604e+13
2014	MINI	-25.34	2.230142e+15
2014	SMALL	-0.55	6.119121e+13

2014	MEDIUM	-0.06	1.149431e+14
2014	LARGE	0.05	2.966471e+13
2014	XLARGE	0.11	8.126772e+15
2015	MINI	-16.75	9.881133e+14
2015	SMALL	-81.40	1.640383e+12
2015	MEDIUM	-0.01	3.103025e+12
2015	LARGE	0	3.016631e+12
2015	XLARGE	0.21	9.118902e+15

Table 6: Descriptive table of linear regression prediction results

We found that the mean square error has improved but still high compared to a good prediction quality. We subsequently applied the same treatment for further adjustment but the results are always the same. In this situation, we decided to improve our algorithm by adding a regularization term α which is ridge regression. To do this we used the method ***sklearn.linear_model.Ridge*** from the library sickit_learn with such parameters:

alpha: a small positive value, it improves the conditioning of the problem and reduce the variance of estimates.

normalize = True

To choose the correct value of α, we used ***RandomizedSearchCV*** that implements a method of adjustment and scoring, the estimator parameters are optimized by ***cross_validation*** method. This method generates the estimator variables chosen based on research, namely, the estimator that gave the highest score (best estimator) and the score associated with it (best score).

We find that the MSE is always high, that is why we decided to abandon the track that has produced no relevant results. Indeed, these algorithms respond to linear problems, but ours is still a nonlinear problem.

These results endorse the correlation diagram - as we saw in the previous chapter - which showed that there is no linear relationship between the variables.

We will start now, the study of algorithms that respond to nonlinear problems including classification algorithms specific to these models.

4.2. Classification Algorithms

Before starting the implementation of each classification algorithm, we present some theoretical concepts related to their evaluation. Indeed, for classification problems, we handle discrete values, for this reason the MSE can generate false results because it calculates the difference between continuous values. The best solution is to use the confusion matrix shown in figure n° 55. Each matrix column represents the number of occurrence of an estimated class, while each row represents the number of a real class. This matrix is used to describe the accuracy of population classified with respect to a reference population, it gives an idea about the percentage of populations correctly and incorrectly predicted for each class. We also use precision performance metrics and recall. By adapting this ended model, several cases may fail, precision is the percentage of correct answers, and Recall is the percentage of correct answers are given.

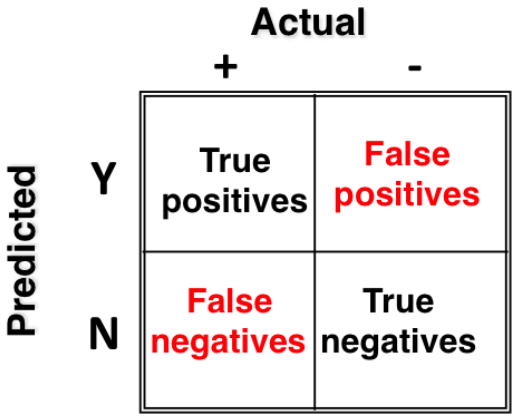


Figure n° 55: Confusion matrix

Precision is defined as the number of true positives (TP) divided by the sum of true positives and false positives (FP):

$$precision = \frac{TP}{TP+FP} \quad (9)$$

The recall is defined as the number of true positives divided by the sum of the true positives and true negatives:

$$recall = \frac{TP}{TP + TN} \quad (10)$$

The Table n°7 presents a description on how to calculate precision and recall:

Case name	Description
True positive (TP)	The learning model is right as the taxpayer belongs to the class
False positive (FP)	The model is wrong as the taxpayer does not belong to the class
True negative (TN)	The model is right as the taxpayer does not belong to the class
False negative (FN)	The model is wrong as the taxpayer does not belong to the class

Table 7: Recall and precision description

We will also use for the evaluation algorithms f1_score (also known as f-score or f-measure) is a measurement accuracy. It considers both precision and recall.

$$f1 = 2 * \frac{precision * recall}{precison + recall} \quad (11)$$

We start now the modeling phase, we have chosen to teach our algorithms data from two files, “Moral Persons” and “Physical Persons”. We also intend to manipulate the inputs of our models to determine the algorithm having the best compromise between quality and performance.

In addition, we have divided our learning bases in 6 classes as our dependent variable y which is equal to the sum of the preliminary and thorough recovery REDR + REDR APP =y. These classes are as follows in table n° 8:

Classes	Description
1	[0, 1 000 TND[
2	[1 000 TND, 3 500 TND [
3	[3 500 TND, 10 000 TND[
4	[10 000 TND, 30 000 TND[
5	[30 000 TND, 60 000 TND[
6	[60 000 TND, Inf[

Table 8: Tax payers' classes based on adjustment amounts

The division of classes is chosen at follows: we tried to make it as close as possible to the data distribution. Indeed, we have seen from the statistics presented by the figure n° 56 and figure n° 57 that 25% of taxpayers have a recovery average of about 1600 TND, 50% have an average of about 5 700 TND, 75% have an average of 20 000 TND, and when we observe taxpayers who have an average above 20 000 TND we find that it is between 30 000 TND, 50 000 TND. Therefore, we tried to group the distribution in classes with low margins in order to minimize and estimate the range of fraud.

```
count    5.416000e+03
mean     3.245845e+04
std      1.115977e+05
min      1.000000e-03
25%      1.648154e+03
50%      5.702985e+03
75%      2.015150e+04
max      2.296912e+06
```

Figure n° 56: Y distribution

```
count    1.357000e+03
mean     1.144290e+05
std      2.017051e+05
min      2.005833e+04
25%      3.015491e+04
50%      4.899921e+04
75%      1.094194e+05
max      2.296912e+06
```

Figure n° 57: Y distribution (Y>20Millions)

4.2.1. Multilayer perceptron

For the implementation of the neural network algorithm, we used the method ***sklearn. grid_search.GridSearchCV*** of scikit_leran library with the estimator "classifier" which has the same behavior as the MLP (Mutil Layer Perceptron) Furthermore, we used ***GridSearchCV*** for searching for the best estimator parameters. Indeed, the parameters which are not directly taught

in the estimators can be set by looking for wider parameters to get the best cross-validation performance score for the estimator evaluation. It allows to determine the best estimation parameters for our model.

During this modeling, we conducted several tests by the term calibration (learning spleen), the number of iterations and the number of layers (hidden layer). We noticed several models have a low classification quality, other models have a better quality. The running time of the algorithm depends on α, the number of iterations and number of layers, the algorithm converges sometimes after 30 minutes and sometimes it takes over time (3 to 4 hours and sometimes a whole night).

After developing several test cases, we found that the use of the "Sigmoid" function as the function of the hidden layer and the function as "Softmax" function gives better results. The "Softmax" function is a specialized activation function for the classification of networks, it performs a standard exponential (that is to say the sum of total output). In combination with the cross-entropy error function, it can change the multilayer perceptron networks for estimating probabilities classes. We will introduce now, the best models we generate and some low classification quality models.

Figures n° 58 and n° 59 presents "Moral Persons" with classification model with α = 0.3, the number of iterations = 150 and the number of units = 50

	precision	recall	f1-score	support
[0,1000dt[	0.34	0.30	0.32	54
[1000dt,3500dt[	0.16	0.18	0.17	73
[3500dt,10000dt[	0.57	0.04	0.08	98
[10000dt, 30000dt[	0.25	0.45	0.32	113
[30000dt, 60000dt[	0.18	0.24	0.20	59
[60000dt, Inf[	0.42	0.34	0.38	82
avg / total	0.33	0.26	0.24	479

Figure n° 58: PM NN evaluation

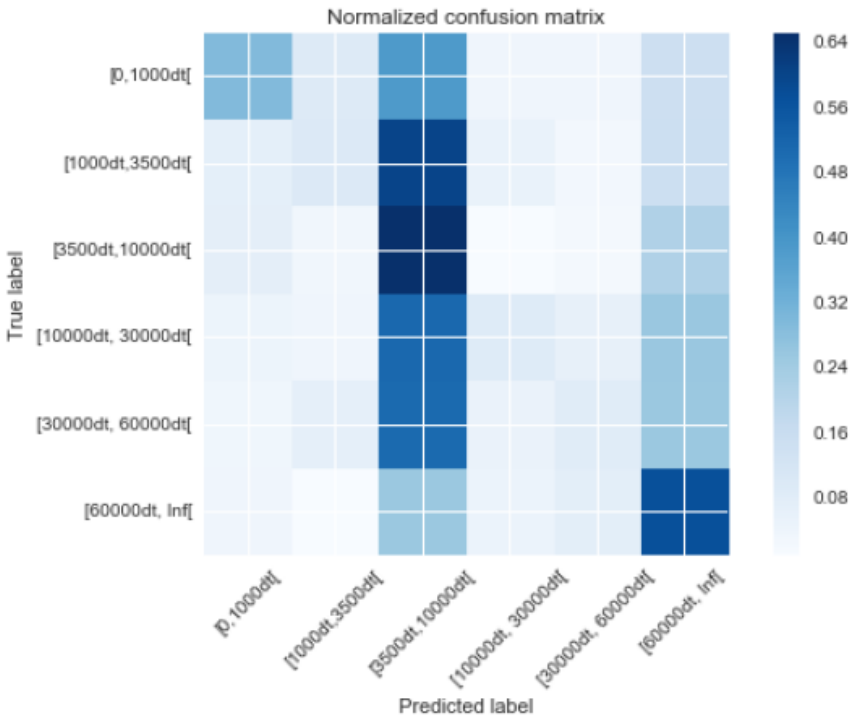


Figure n° 59: PM MLP Confusion matrix

This first model has a precision of average equal to 33% (67% error) for the class [60 000 TND, Inf [the algorithm correctly predicted 42% of taxpayers belonging to the test base, for the class [3 500 TND, 10 000 TND [the precision is equal to 57% (43% error) which is a good result, for the other classes we find that precision is low.

Figures n° 60 and n° 61 show “Physical Persons” classification model with = 0.01, the number of iterations = 10 and the number of unit = 100

	precision	recall	f1-score	support
[0,1000dt[	0.28	0.65	0.39	134
[1000dt,3500dt[	0.24	0.11	0.15	178
[3500dt,10000dt[	0.23	0.05	0.08	135
[10000dt, 30000dt[	0.30	0.47	0.36	100
[30000dt, 60000dt[	0.25	0.04	0.07	26
[60000dt, Inf[	0.57	0.25	0.35	32
avg / total	0.27	0.28	0.23	605

Figure n° 60: PP MLP evaluation

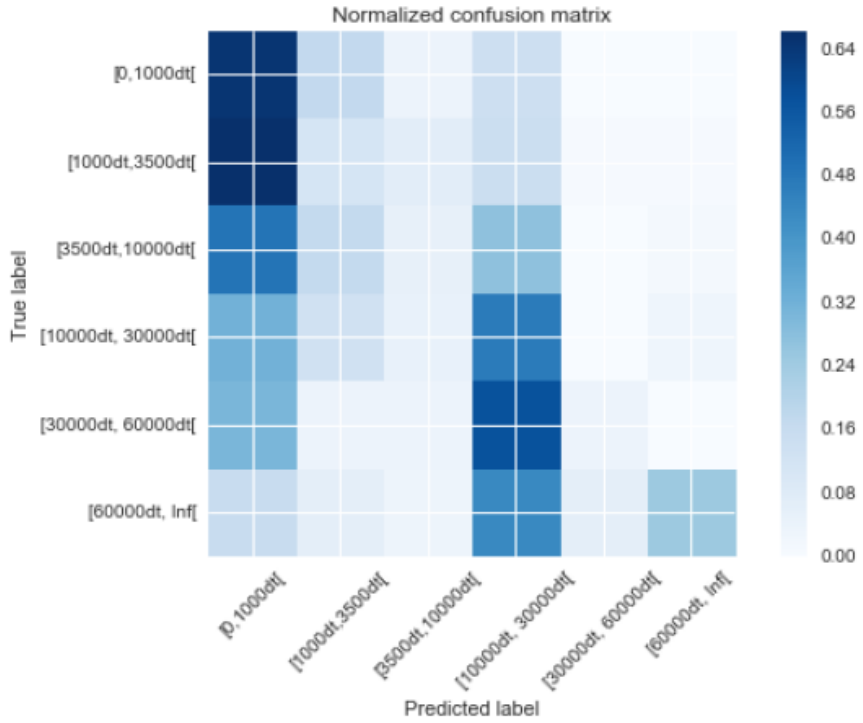


Figure n° 61: PP NN confusion matrix

Figures n° 62 and n° 63 show the Corporations "PM" classification model α = 0.01, the number of iterations = 10 and the number of unit = 150

[(479, 6)]

	precision	recall	f1-score	support
[0,1000dt[	0.56	0.17	0.26	54
[1000dt,3500dt[	0.14	0.01	0.03	73
[3500dt,10000dt[	0.22	0.11	0.15	98
[10000dt, 30000dt[	0.24	0.79	0.37	113
[30000dt, 60000dt[	0.22	0.03	0.06	59
[60000dt, Inf[	0.50	0.21	0.29	82
avg / total	0.30	0.27	0.21	479

Figure n° 62: PM NN evaluation

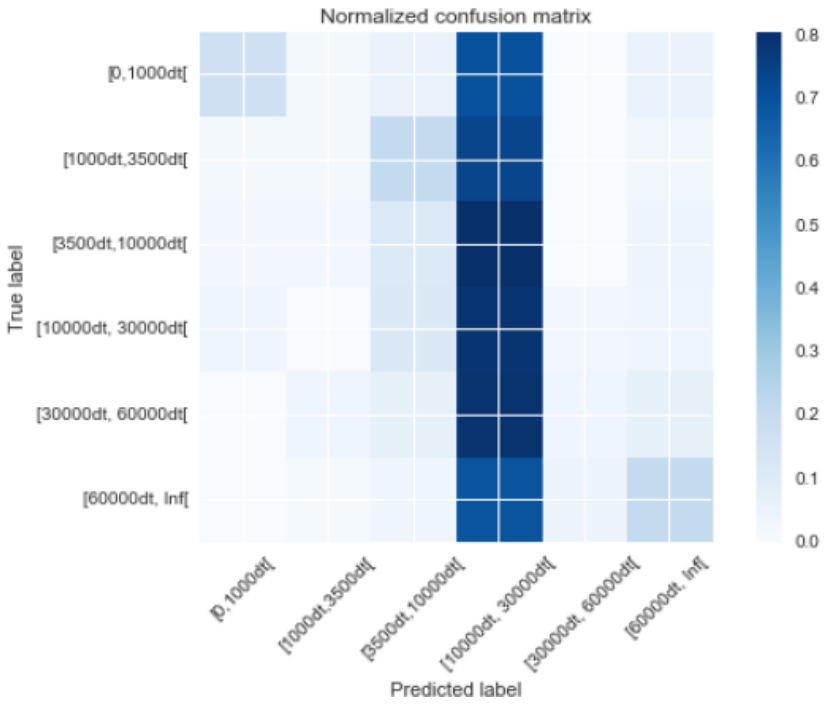


Figure n° 63: PM NN confusion matrix

We note that this lasdt model has improved for the first and last class. Indeed, the precision of latter class is equal to 50% (50% error) which is an interesting percentage in our case. The model has succeeded classified 50% of the population belonging to this class.

Now if a screening officer suspects treats 10 cases, There is a high probability that he gets 5 cases whose recovery is greater than 60 000 TND. We are now able to classify taxpayers, not just whether they are fraudulent or not, but we can adjust the recovery value.

We will try in next sections to improve the precision of our models.

4.2.2. Random Forest

For the implementation of the machine learning algorithm Random forest, we used the ***ExtraTreesClassifier*** class that implements an estimator adjusting to a large number of random decision trees and applying to different subset of dataset samples. It uses the average to improve the classification precision and control overfitting. We also used the ***GridSearchCV*** method which is used to optimize the grid search parameters.

We modeled the RF algorithm on the basics of learning on “Moral Persosn” and “Pysical Persons”, we manipulated the “*n_estimator*” parameter which is the tree number in the forest.

We noticed that the algorithm converges faster than the neural network algorithm, the execution time is estimated at few minutes (about 15 minutes).

Here are some modeling examples presented by the Figures 66, 67, 68 and 69:

Figures n° 64 and n° 65 show the evaluation of model 4 which is the application of random forest algorithm on individuals "Physical Persons". We see clearly improved precision percentages for all classes with a total precision 44% (56% error) which is an interesting result! In this case, we set a range for the estimator [500, 750, 900], the algorithm has determined that the best parameter *n_estimator* = 900. We got a remarkable result for class 5: [60 000 TND, Inf [71% (29% error) which is great!

	precision	recall	f1-score	support
[0,1000dt[	0.45	0.34	0.39	134
[1000dt,3500dt[	0.43	0.47	0.45	178
[3500dt,10000dt[	0.38	0.44	0.41	135
[100000dt, 300000dt[	0.46	0.54	0.50	100
[300000dt, 60000dt[	0.40	0.15	0.22	26
[600000dt, Inf[	0.71	0.47	0.57	32
avg / total	0.44	0.43	0.43	605

Figure n° 64: PP RF evaluation

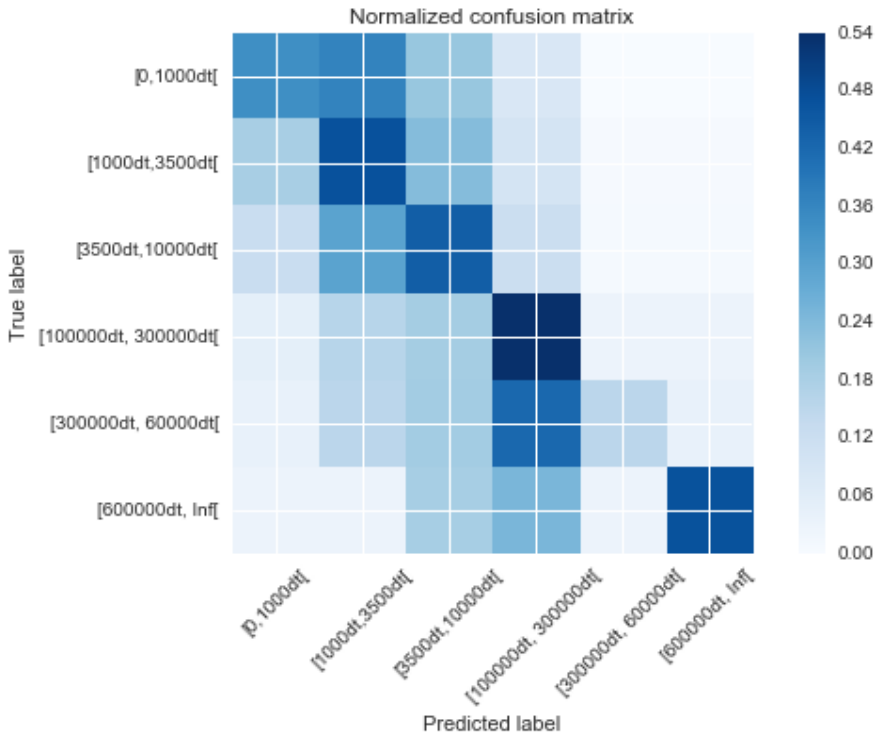


Figure n° 65: PP RF Confusion matrix

Figures n° 66 and n° 67 show model 5 evaluation which is the application of random forest algorithm for “Physical Persons”. In this case, the best parameter is equal to 900. For class 5, we have great results 44% (56% error)

	precision	recall	f1-score	support
[0,1000dt[	0.43	0.17	0.24	54
[1000dt,3500dt[	0.36	0.29	0.32	73
[3500dt,10000dt[	0.41	0.39	0.40	98
[10000dt, 30000dt[	0.32	0.47	0.38	113
[30000dt, 60000dt[	0.25	0.10	0.14	59
[60000dt, Inf[	0.44	0.63	0.52	82
avg / total	0.37	0.37	0.35	479

Figure n° 66: PM RF evaluation

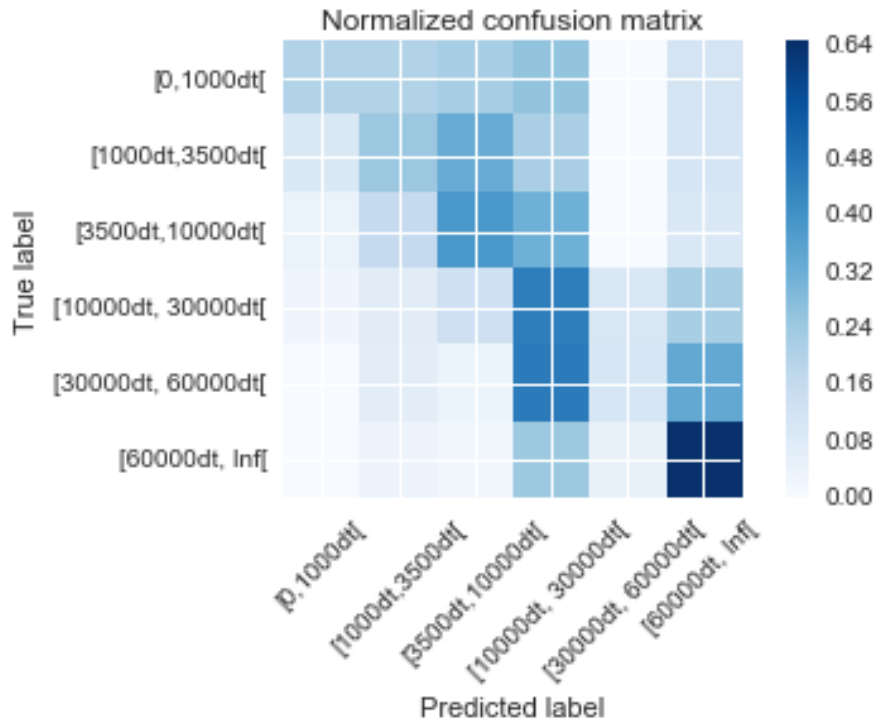


Figure n° 67: PM RF confusion matrix

Figures n° 68 and n° 69 present model 6 evaluation which is the application of random forest algorithm on corporations "Moral Persons" with criteria 13 and 19 and the base containing taxpayers' activity (The input data for this model is different from the others). The algorithm's precision is very interesting. The best *n_estimator* value for this model is 200, all classes have a precision above 50%, it is a model to remember. We see clearly that the highest percentage are on the matrix confusion diagonal.

	precision	recall	f1-score	support
[0,1000dt[	0.61	0.59	0.60	148
[1000dt,3500dt[	0.52	0.60	0.56	181
[3500dt,10000dt[	0.52	0.50	0.51	147
[100000dt, 300000dt[	0.50	0.49	0.49	98
[300000dt, 60000dt[	0.57	0.38	0.46	21
[60000dt, Inf[	0.81	0.57	0.67	30
avg / total	0.56	0.55	0.55	625

Figure n° 68: PM RF 2evaluation

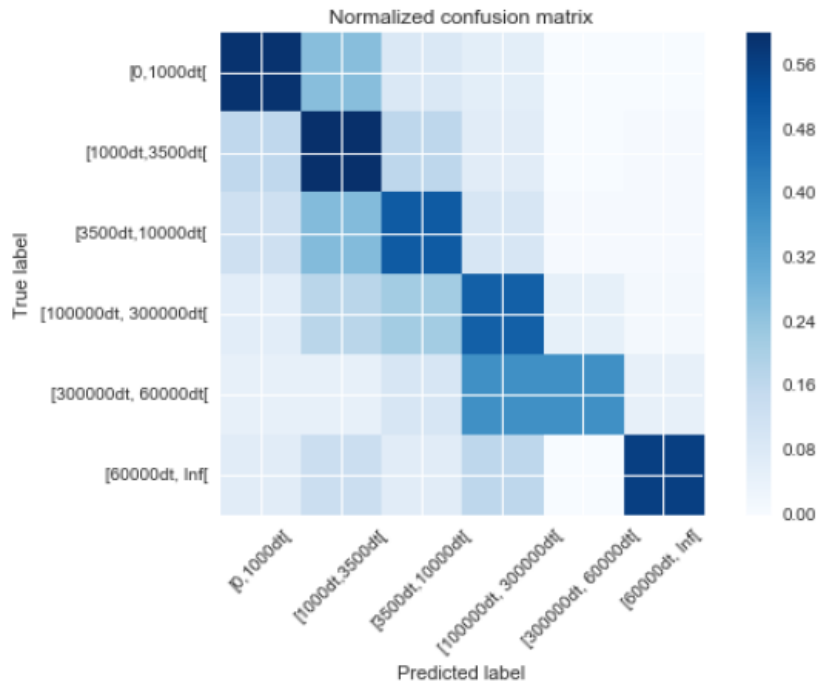


Figure n° 69: PM RF 2 Confusion Matrix

4.2.3. Support Vector Machine

Figures n °70 and n° 71 show model 7 evaluation which is the application of SVM algorithm on corporations “Moral Persons”. This model has given good prediction for class 1 and class 5 but remains small for other classes.

	precision	recall	f1-score	support
[0,1000dt[	0.57	0.07	0.13	54
[1000dt,3500dt[	0.21	0.08	0.12	73
[3500dt,10000dt[	0.24	0.10	0.14	98
[10000dt, 30000dt[	0.26	0.79	0.39	113
[30000dt, 60000dt[	0.20	0.02	0.03	59
[60000dt, Inf[	0.46	0.28	0.35	82
avg / total	0.31	0.28	0.22	479

Figure n° 70: PM SVM evaluation

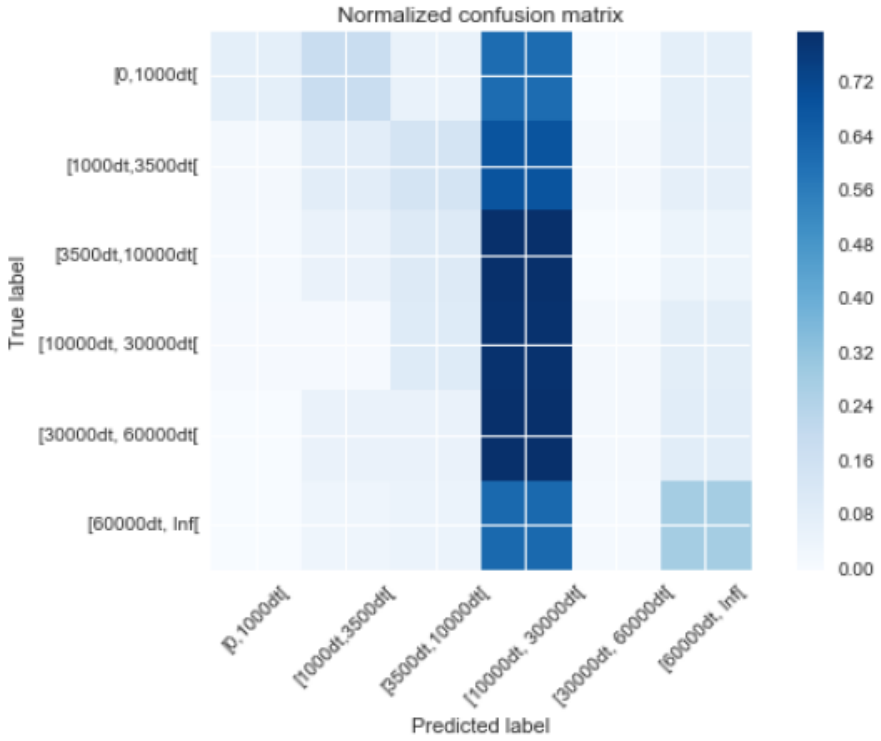


Figure n° 71: PM SVM Confusion matrix

Figures n °72 and n° 73 show model 8 evaluation which is the application of SVM algorithm on traders: “Physical Persons.

	precision	recall	f1-score	support
[0,1000dt[	0.36	0.29	0.32	134
[1000dt,3500dt[	0.35	0.43	0.38	178
[3500dt,10000dt[	0.27	0.33	0.30	135
[100000dt, 300000dt[	0.27	0.24	0.25	100
[300000dt, 60000dt[	0.25	0.08	0.12	26
[600000dt, Inf[	0.47	0.22	0.30	32
avg / total	0.32	0.32	0.31	605

Figure n° 72: PP SVM evaluation

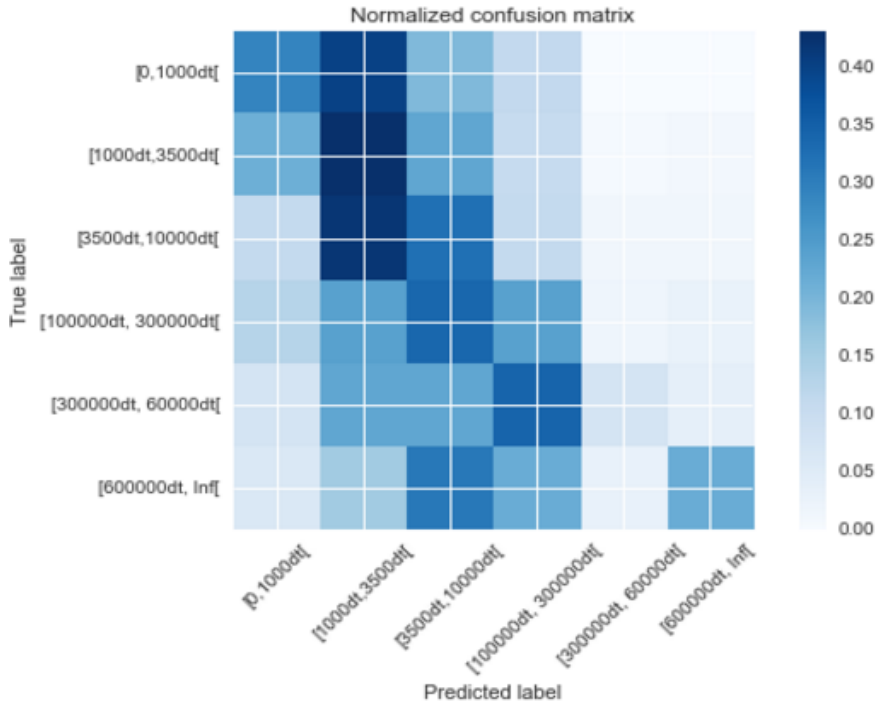


Figure n° 73: PP SVM Confusion matrix

For this model, SVM has yielded interesting results especially for class 5 where the recovery amount is above 60 000 TND with a percentage of 47% accuracy for “Moral Persons” and 46% for “Physical Persons”.

The found results were found following careful parameters selection through the ***GridSerachCV*** estimator. This estimator has found that for the “Moral Persons” category the best parameters are: 'C': 25, 'kernel' 'rbf', 'gamma': 0.0001 and for “Physical Persons”, we had: 'gamma': 0001, 'C': 50, 'kernel' 'rbf', with:

'Kernel' or the kernel function converts a scalar product in the high dimensional space which is complex to a simpler operation, the kernel for SVM 'Rbf' is selected by default

The parameter C indicates SVM optimization: how to avoid a bad classification. For large values of C, the optimization will choose a hyperplan with a small margin if such hyperplan does a better job of getting all the training points classified correctly. Conversely, a low value of C will cause the optimizer to find a larger margin hyperplan separation, even if it gets wrong hyperplan to classify more points. For very small values of C, we can obtain misfiled examples, even if our training data is linearly separable. To respect these characteristics 'C' we chose margin [25, 50, 75, 100] with values neither too large nor too small.

The gamma parameter defines how far the influence of a single training example is achieved with lower values meaning "far" and higher values mean "close". When gamma is very low, the “rbf” kernel is very broad. Suppose that the “rbf” kernel is sufficiently positive for every point of the data set. This will likely give the optimizer hard work since changing the value of one point will change the decision based on all points. The other extreme is when gamma is large, which means that the “rbf” kernel is very narrow. This means that, probably, all training vectors will end as a support vector. This is clearly undesirable. Therefore, we have chosen the values neither too large nor too small, the suggested range is [1-2, 1e-3, 1e-4]

Tables n° 9 and n°10 present a summary for classification prediction for both categories “Physical Persons” and “Moral Persons”. We only took in consideration precision. We can see that Random Forest has given the best total results.

First line Intervals are in TND.

	[0, 1 000[	[1 000 , 3 500[	[3 500, 10 000[	[10 000, 30 000[	[30 000, 60 000[	[60 000, Inf[	Total
SVM	0.57	0.21	0.24	0.26	0.20	0.46	0.31
RF	0.43	0.36	0.41	0.32	0.25	0.44	0.37
NN	0.50	0.23	0.29	0.26	0.17	0.46	0.31

Table 9: Classification prediction results for “Moral Persons”

	[0, 1 000[	[1 000, 3 500[	[3 500, 10 000[	[10 000, 30 000[	[30 000, 60 000[	[60 000, Inf[	Total
SVM	0.36	0.35	0.27	0.27	0.25	0.47	0.32
RF	0.45	0.43	0.38	0.46	0.40	0.71	0.44
NN	0.28	0.24	0.23	0.30	0.25	0.57	0.27

Table 10: Classification prediction results for “Physical Persons”

After classification models development, we found that random forest algorithm gave the best result and the performance regarding execution time, but this does not mean that other algorithms are ineffective, in fact they require more material resources, powerful and scalable software with a large storage capacity

Conclusion

In this final chapter, we presented the implementation process of machine learning models, namely, regression and classification algorithms. We evaluated each algorithm using MSE, for regression algorithms and confusion matrix, precision and recall for classification algorithm. RF algorithm has given so far the best results, but over all, this prediction can be improved in the future for all algorithms.

General Conclusion

Fraud detection is particularly a difficult and complex task. Fraudulent activities are constantly changing events and remain difficult to model. Large transactions' volume requires the presence of daily automatic analysis tools to allow human resources focus on most suspicious files.

In Tunisia economic experts consider that existing tax control cannot achieve effective participation of all taxpayers in state budget for various reasons (transparency lack for the revenues' statements and officers control diligence). In response to these challenges, the Ministry of Finance data center offers a solution to automate and target high income recovery missions in terms of money and time to select taxpayers who have the highest recovery amount.

This graduation project aims to address this need, the first inspiration was European examples then we opted for an open source solution which is the use of machine learning library Sickit Learn with python. We have carefully developed data preparation phase to succeed thereafter modeling phase where we tried many models. We have obtained very interesting prediction results that can be improved in the future.

To extend the Tunisian tool, conditions must be created and political decisions should be taken. The fight against tax fraud at international level requires the establishment of a platform for data exchange, widespread access and direct data, etc. things that have not yet been materialized. Similar efficient projects exist in other European countries but they are mainly focused on other types of fraud. Sharing experiences and knowledge between countries would be another success factor of these Big Data strategies against tax fraud in the European Union.

Regardless of any technical considerations, two major issues are countering these projects. The first is inherent to control principle: cross information from different and heterogeneous sources involves having administrative structures and procedures that can supply reliable and permanent platform for Big Data. The second difficulty comes from the respect for private life, such projects can counteract. As part of the project "redditometro" for example, monitoring of individuals' train lives has provoked strong protests. If these technologies seem full of promises, one question remains unanswered: "Will Big Data be the Big Brother of tomorrow?"

Bibliography

[1] CIMF, Organizational chart (CIMF Internal document)

[2] Lucy Warwick-Ching,"Ten Ways HMRC checks if you're cheating", 16 nov 2012 [seen on 04/02/2016]

[3] Capgemini consulting, "Our take on the impact of digital technologies on tax and welfare fraud" [seen on 04/02/2016]

[4] Mohamed Ben Naceur, Réalités Online. Fraude fiscale : La partie immergée de l'iceberg http://www.realites.com.tn/2015/02/raude-fiscale-la-partie-immergee-de-liceberg [seen on 10/02/2016]

[5] Code des Droits et Procédures Fiscaux, Article 36. [seen on 15/02/2016]

[6] Code des Droits et Procédures Fiscaux, Article 37 [seen on 15/02/2016]

[7] Code des Droits et Procédures Fiscaux, Articles 38 to 41 [seen on 15/02/2016]

Netography

[N1]http://www.leparisien.fr/magazine/grand-angle/escroquerie-a-la-tva-comment-l-etat-perd-10-milliards-d-euros-par-an-17-04-2014-3775281.php [consulted in 05/02/2016]

[N2]http://clubacteurspublics.com/2013/12/belgique%89-fraude-a-la-TVA/ [consulted in 08/02/ 2016]

[N3]http://www.economist.com/blogs/schumpeter/2013/01/tax-evasion-italy

[consulted in 07/02/2016]

[N4]http://droit-finances.commentcamarche.net/contents/1579-fraude-fiscale-definition-et-sanctions [consulted in 05/02/2016]

[N5] http://www.profiscal.com [consulted in 05/02/2016]

[N6] Kingsley Gaius. Data Analytics and Predictive Analytics, Inferential Statistics and CRISP-DM process model. http://kingsleygaiusblogondataanalytics.blogspot.com [consulted in 05/03/2016]

[N7] IBM Knowledge Center. Présentation de l'aide CRISP-DM. http://www.ibm.com/support/knowledgecenter/SS3RA7_15.0.0/com.ibm.spss.crispdm.help/crisp_overview.htm?lang=fr [consulted in 16/02/2016]

[N8] Olivier Decourt. Le DataMining, qu'est-ce que c'est et comment l'appréhender? http://www.od-datamining.com/dm/what_is.html [consulted in 20/02/2016]

[N9] DIDIER GAULTIER. Méthode CRISP : la clé de la réussite en Data Science. http://blog.businessdecision.com/bigdata/2016/02/methode-crisp-la-cle-de-la-reussite-en-data-science/ [consulted in 05/02/2016]

[N10] http://www.sas.com/en_id/insights/analytics/machine-learning.html consulted in 15/02/2016 [consulted in 05/02/2016]

[N11] https://en.wikipedia.org/wiki/Scikit-learn [consulted in 12/02/2016]

[N12] http://scikit-learn.org/stable/testimonials/testimonials.html [consulted in 12/02/2016]

[N13] http://www-03.ibm.com/software/products/en/spss-modeler[consulted in 07/02/2016]

Glossary

«BI»	Business Intelligence
« CIMF »	Centre Informatique du ministère des finances
«DGE»	Direction générale des grandes entreprises
« DGI »	Direction générale des impôts
« ENSIT »	High National School of Engineers of Tunis
«MLP»	Multilayer perceptron
«MSE»	Mean squared error
«PP»	Physical Persons
«PM»	Moral Persons
«RF»	Random Forest
« TND»	Tunisian Dinar

ملخّص: تشكّل هذه الوثيقة تقرير التّربّص النّهائي لمرحلة الهندسة في مجال هندسة الكمبيوتر. نظرا للوضع الاقتصادي الحالي في تونس، من الضروري وضع وسائل فعالة لمكافحة التهرب الضريبي ولاستجابة لهذه الحاجة يقترح مركز الإعلامية لوزارة المالية الأتمتة وتوجيه مهمات التعديل الضريبي. وفي هذا لإطار تمثل التربص لمشروع التخرج في برمجة آلة ذكية للتحليل المالي مع تقنيات التعلم الآلي التي تسمح بتصنيف المتهربين من الدفع الضريبي وتقدير قيمة الاحتيال.

المفاتيح دافع الضريبة، التهرب الجبائي، التعديل الضريبي، التعلم الآلي، تصنيف، التوقع

Mise en place d'une Machine Intelligente d'analyse financière

Le présent document forme un rapport de stage de fin d'études pour l'obtention du diplôme d'ingénieur en Génie Informatique. Suite à la situation économique actuelle de la Tunisie, il est nécessaire de mettre en place des moyens efficaces afin de lutter contre la fraude fiscale. En réponse à ce besoin, le centre informatique du ministère des finances propose d'automatiser et de cibler les missions du redressement. Notre stage de fin d'étude consiste à mettre en place une machine intelligente d'analyse utilisant des techniques d'apprentissage automatique qui permettent la classification des contribuables fraudeurs et l'estimation de la valeur de leur redressement.

Mots-clés : contribuable, fraude, redressement, apprentissage automatique, classification, régression.

Intelligent Financial Analysis Machine

This document is the report of Computer Science course's graduation project. Considering the current economic situation in Tunisia, it is necessary to establish effective ways to fight against tax fraud, in response to this need, the computer center of the Finance of Ministry proposes to automate and target recovery missions. Our final internship study is to develop an intelligent machine of financial analysis with machine learning techniques that allow the classification of fraudster's taxpayers and fraud estimation.

Key Word: taxpayers, fraud, recovery, machine learning, classification, regression

Entreprise Complete Address

Entreprise : Centre Informatique du Ministère des finances
Address : Centre Urbain Nord
Tél. : 99 740 922
Fax : 71 948 415