What use of AI in finance?

ISCF Final Report

Denis DEMKO Thomas DEROO
Doris FEJZA Elona KARAJ
Hussein LEZZAIK Estia MALIQARI
Yijue XIE

Supervisor: Prof. Dr. Dritan NACE

University of Technology of Compiègne
12th January 2021

Contents

1 Introduction

Machine Learning is becoming more and more popular in finance. Thanks to its ability to handle large and complex amounts of data, Machine Learning is now transforming the financial industry like never before. This technology plays a very important role in many aspects of the finance world, from managing assets to carrying out loans. Because of the high volume of historical data available in the finance industry, Machine Learning has found various application in this field.

1.1 Objectives

There is a triple objective to this project.

Firstly, a study on the various applications of Machine Learning in finance should be provided. This would help to create a more general picture of how Machine learning and finance are connected.

Secondly, the work includes a study of the portfolio management problem as well as applying the SVM and neural network methods in the French stock market.

Thirdly, a study should be provided on the credit risk evaluation problem as well as "overdraft" data. An application should be made that takes into account all types of standard customer data.

For both applications, recent previous studies have been made from some UTC students as part of their Master thesis. These studies have been taken as a starting point for our project.

1.2 Project organization

1.2.1 Team organization

In order to attain all the objectives of this project, our team is divided in two sub-groups: one that works on the portfolio management problem and the other on credit risk evaluation problem. There is also one head of project who works on the detailed study of the different applications of machine learning in finance as well as help organize the work and make the connection between the two groups.

1.2.2 Tasks realised

In Figure 1 it is shown a schema of the main tasks realized by each subgroup. In blue there are the tasks done by Portfolio optimization group and in orange the ones by Credit risk group. Both of the problems treated in this project

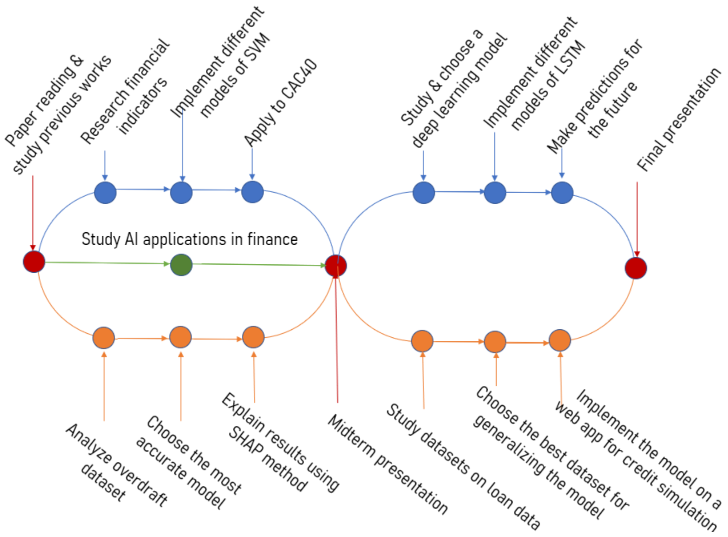


Figure 1: Main tasks realized

represent two different financial applications of Machine Learning, although the algorithms that can be used for predictions may be similar. However, in this project we have used different methods for each application. This is why the moments where both subgroups have worked together were mainly in the very beginning where we studied the previous models as well as during the preparation for the presentations and the report.

In Appendix A, a more detailed set of tasks is presented for each subgroup using Gantt charts. It can also be found the chart done at the very beginning of the project.

1.2.3 Tools used

During the course of the project, several tools were used that helped us better organize the work. We mention some of the most relevant below:

- **Slack** - inter and intra-group communication
- **Zoom** - weekly meetings with our supervisor
- **Overleaf/LaTex** - work on the report and presentations
- **AngularJS, Python** - building the programs
- **Gitlab UTC** - sharing and accessing project information

2 Applications of Artificial Intelligence in finance

In this section we study some of the various Artificial Intelligence applications in finance. Although in our project we are focusing on two applications, it is important to create an idea of the general picture and see more in details how Artificial Intelligence and finance can be connected.

2.1 Fraud detection

Fraud is a deceptive action done with the intention to gain at the expense of another through illegal or unethical practices. In finance, fraud can take on many forms: credit card fraud, identity theft, tax evasion, insurance fraud, money laundering etc. It is a massive problem in finance resulting to the loss of billions of euros. That is why fraud detection is one of the most important Machine Learning applications.

With the help of Machine learning, money fraud detection actively learns and calibrates to new potential (or real) security threats. Systems can detect suspicious activities or behaviors and flag them for further verification. The biggest challenge for these systems is to avoid false positives: situations where a certain activity is flagged as a risk, but it was never one in the first place. They happen when the financial institution wrongly declines a legitimate transaction. These wrongful detections happen often and tend to make banks lose their credibility and customer loyalty, let alone the time lost for making the verifications.

2.2 Money Laundering

Money laundering is one special type of fraud, a major financial crime. It is the process of legalizing the money obtained illegally, by passing it through a complex sequence of banking transfers or commercial transactions. Placement, layering, and integration are the three phases in money laundering schemes. Proceeds from criminal activities enter the placement phase, where they are converted into monetary instruments or otherwise deposited in a financial institution (or both). Layering refers to the transfer of funds to other financial institutions or individuals via wire transfers, checks, money orders, or other methods. In the final phase of integration, funds are used to purchase legitimate assets or to continue financing criminalized enterprises. Here, illegally obtained money becomes part of the legitimate economy. According to a United Nations report [3], it is estimated that the total amount of the money laundered in a year represents $2-5\%$ of the global GDP. A lot of regulations have been put in place to fight this phenomenon.

Anti-Money Laundering is the term that describes the legal controls that require financial institutions to prevent, detect, and report money laundering activities. Today, all financial institutions are obliged to identify and report any suspicious transaction to the corresponding authorities. These tasks include

"Know-your-customer" (KYC) measures, which refers to knowing the identity of your customer as well as understanding the types of transactions he/she is likely to engage. These processes should take place at two times: before engaging with the potential customer to ensure that they are really who they claim to be and the business in which they are involved. They also should take place during the business relationship to establish that a customer's risk profile continues to match the firm's information on them. These steps are often manual and time-consuming, so it is important to automate them to simplify the process and detect the good and fraudulent clients.

Thanks to Machine learning and Big Data, AML mechanisms have become more efficient and more secure over the recent years. These mechanisms filter customer data, classify it according to level of suspicion, and inspect it for anomalies. Some typical anomalies that would trigger a flag on the account and its transactions include sudden and substantial increase in funds, a large withdrawal, or moving money to a bank secrecy jurisdiction.

The current Machine Learning methods that tend to detect money laundering are supervised and unsupervised learning.

For unsupervised methods, we try to identify a pattern in the data without any prior information whether the data corresponds to money laundering or not. For supervised methods, the data is labelled, and they learn the patterns by differentiating which correspond to money laundering and which to legal activity.

Undoubtedly, supervised methods are preferred when the data is labeled. However, when it comes to money laundering, it is complicated to have the necessary information whether an AML suspect was guilty or not, as the financial institution rarely finds out about it. The only information we can have is whether the activity is "suspicious" or not. But on the other hand, this information is enough. [4] According to General Data Protection Regulation (GRPD), every legal decision, including signaling the authorities or declaring whether the suspect is guilty or not, needs to be made by a human being. So, the machine learning methods should only detect suspicious activity, it is then the bank employees who decide to report the activity to the corresponding authorities or not, based on the available information and the level of suspicion.

Most used Machine Learning methods for Anti-Money Laundering and frauds that are often more preferred to evaluate suspicious activities are logistic methods, Decision Trees, Random Forest, SVM, Bayesian networks, neural networks etc.

2.3 Document analysis

Text analytic is the process of deriving valuable information from text. Although it is not connected directly to finance, it is still a very important application that helps banks and financial institutions save many hours of work. They must deal with tons of papers and legal documents every day in hard copy. Machine Learning systems are able to scan and analyze these documents at speed, with a very high accuracy. Employees can then instantly search and find the documents they need. They use Optical Character Recognition (OCR) and deep learning for image recognition. Using the Machine Learning classification methods, these systems can also recognize handwritten data and convert them to text.

In terms of word training, machine learning techniques are widely used in textual analysis. In finance literature, we find Naive Bayes, Support Vector Machine and Neural Network are among the most popular machine learning techniques.

2.4 Chatbots for customer support

Chatbots are computer programs that are used to conduct an online chat conversation without the help of a human agent. They are gaining more and more popularity in finance by improving the customer experience and reducing the gap in the human-machine interaction. Thanks to Machine Learning, finance-specific chatbots allow customers to manage requests in a faster and more efficient way. They can adapt to every customer and every behavior. Chatbots free up the time of customer support employees by solving basic queries, so they can concentrate on the more complex ones.

When a user asks a question, the chatbot analyzes it and identifies the intent and entities through machine learning algorithms and Natural Language Processing. It defines the type of request by capturing the key words. It then replies to the user.

In the next sections, the study on the two applications: portfolio optimization and credit risk evaluation will be provided. Both of them use supervised learning algorithms in order to make predictions. Different models have been studied and tested in both cases in order to choose the one with the best performance. The first problem predicts the stock market behaviour, whether the price will go up or down. In the second problem, we classify the clients whether they are good or bad.

3 Portfolio optimization

In this section we study the portfolio optimization problem. We implement two different algorithms: SVM and LSTM to make predictions about the stock market behaviour, specifically the CAC40 stock market.

3.1 Context of the project

When making stock market trend prediction, there are two main theories that are taken into consideration: the fundamental analysis and the technical analysis. They are concerned with different scopes, while the fundamental analysis attempts to assess the intrinsic value of stocks, technical analysis believes that past prices and previous trends affect future prices and trend changes. In this context, they are both valuable to be considered in this application of prediction of stock market trend.

3.1.1 The French Stock Market

The CAC 40 is the French stock market index that tracks the 40 largest French stocks based on the Euronext Paris market capitalization. The CAC 40 started with a base value of 1,000 in December 1987 and continued to operate on a total market capitalization system until 2003 when it was changed to a free float adjusted market capitalization methodology. The index is made up of the largest 40 companies listed in France screened by market capitalization, trading activity, size of balance sheet, and liquidity. The multinational reach of the companies listed on the CAC 40 makes it the most popular European index for foreign investors.

CAC 40 stands for ”Cotation Assistée en Continu”, which translates in English to ”continuous assisted trading”, and is used as a benchmark index for funds investing in the French stock market. The index also gives a general idea of the direction of the Euronext Paris, the largest stock exchange in France formerly known as the Paris Bourse. The CAC 40 represents a capitalization-weighted measure of the 40 most significant values among the 100 highest market caps on the exchange. The index is similar to the Dow Jones Industrial Average (DJIA) in that it is the most commonly used index that represents the overall level and direction of the market in France.

The CAC 40 index represents the 40 largest equities listed on the Euronext Paris in terms of liquidity, and includes such companies as L’Oreal, Renault, and Michelin. An independent steering committee reviews the CAC 40 index composition quarterly. At each review date, the committee ranks companies listed on Euronext Paris according to free float market capitalization and share turnover in the previous year. Forty companies from the top 100 are chosen to enter the CAC 40, and if a company has more than one class of shares traded on the exchange, only the most actively traded of these will be accepted into the index.

3.1.2 The CAC40 Dataset

It is important to present our dataset and its particularities. The dataset is taken from Yahoo! Finance. This is a very powerful website that provides both historical data and up to date information and fluctuations connected to the specific dataset. More specifically we can find information from the 25/02/1990 up until the current date. We have retrieved the data in a weekly fashion as it provides proper information about the weekly changes on the market.

Entering into more precise information, the shape of our data is made of 1600 samples and 7 features. Out of these 7 features, one describes the date of the sample so we have only 6 features that describe the situation of the stock market. The features consist of :

- open - stands for the open price
- high - stands for the high price
- low - stands for the low price
- close - stands for the closing price
- adjclose - stands for the adjusted closing price
- volume - stands for the trading volume

These features will be taken as input and then used to compute the technical indicators that we will in turn use in our program to train our model and produce the output and indicate to the user the correct approach to handle the stock. In other words, it will help with the prediction whether the price will increase or fall at the next moment. In the SVM implementation, 52 variables produced will be used to predict the current trend. All prices are initially converted to relative prices (a process commonly known as scaling) in order to avoid the influence of continuous rise and fall of the prices. In the deep learning implementation (LSTM), the close price will be used in order to train and test our model as well as the primal indication for future predictions.

3.1.3 Financial Indicators

Financial indicators are statistics extensively used to monitor the soundness, stability and performance of various sectors of the economy. They are key features in stock market prediction. Some of the most popular financial indicators are:

Price The Typical Price indicator is simply an average of each day's price. The Median Price and Weighted Close are similar indicators.

Moving Average (MA) Stock indicator that is commonly used in technical analysis. We calculate the MA of a stock to help smooth out the price data over a specified period of time by creating a constantly updated average price.

Adaptive Moving Average (AMA) It is used for constructing a moving average with low sensitivity to price series noises and is characterized by the minimal lag for trend detection.

Bollinger Bands Price envelopes plotted at a standard deviation level above and below a simple moving average of the price. They help determine whether prices are high or low on a relative basis.

Average True Rate (ATR) Technical indicator measuring market volatility. It is typically derived from the 14-day moving average of a series of true range indicators.

Moving Average Convergence Divergence (MACD) Trend-following momentum indicator that shows the relationship between two moving averages of a security's price. It helps investors understand whether the bullish or bearish movement in the price is strengthening or weakening.

Relative Strength Indicator (RSI) It measures the magnitude of recent price changes to evaluate overbought or oversold conditions in the price of a stock or other asset.

Stochastic Indicator A popular technical indicator for generating overbought and oversold signals.

3.2 Support Vector Machine

Support Vector Machine is a supervised learning method, frequently used in classification problems. The first part of our project was closely related to understanding how SVM was implemented in the context of stock market trend prediction and how we could adapt the existing program to our own necessities.

In the existing implementation, in order to apply SVM it was important that the dataset was changed in order to fit the type of data an SVM model typically expects. So a data transformation step is necessary to turn what is fundamentally a time series (stock dataset) into a new dataset made out of 52 variables that can then be used by the model. This is done through a series of transformations and actions between the original 6 features. The transformed dataset is then saved, preprocessed and scaled before it can be used by the SVM model. In the Trading Signal portion of the program, the trend prediction itself is performed by firstly building both the SVM and the AMA models and then performing a weighted average conclusion.

The number produced will reflect the trend of the data meaning that the trend is:

- uptrend if the number printed is higher than 0.4
- downtrend if the number printed is smaller than -0.4
- sideways if the number printed is between -0.4 and 0.4

It is important to mention that SVM is implemented only to detect the overall trend of the data and not to perform any specific future price detection. Furthermore, it has been noted by the creator of the program that trend prediction accuracy is higher on long term predictions rather than short term ones. This is not difficult to see when changing the period input for both Trading Signal and Backtesting sections.

Finally, it is important to mention that our contribution during this part was mostly related to understanding the financial background and terminology in relation to stock market trend prediction, understanding how the implementation of SVM was done, finding a dataset that was appropriate for the purpose of our subject and then change the program when necessary to better adapt it to the CAC40 dataset.

3.3 Adding a deep learning method

During the second part of our project, we focused on finding an appropriate deep learning method we could apply in the context of our subject. While there are a number of methods that would allow us to perform stock market trend prediction, a new and interesting method that is being applied in the field of stock market predictions was LSTM model. The current popularity of LSTM as well as its interesting capability to distinguish between and weigh differently recent and early examples in order to predict the next input was surely something we wanted to experiment with [9].

3.3.1 LSTM Model

Long Short Term Memory (LSTM) is an artificial recurrent neural network (RNN) architecture. They are very powerful in sequence prediction problems because they are able to learn selectively and remember or forget historical data as required. This is important in our case because the closing and historical price of a stock is highly volatile and LSTM is adaptable in predicting its future price.

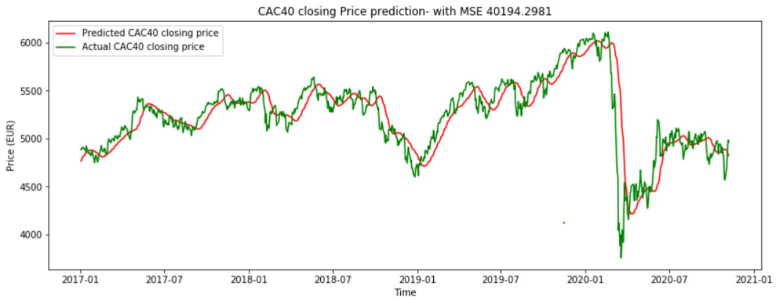


Figure 2: *LSTM Vanilla (FCHI) predictions for CAC40*

In our project, basing on [11], we have implemented two different versions of the LSTM architecture: the base version (referred as LSTM Vanilla) and an advanced one by combining several datasets that we believe may have a correlation by proxy of similar European economy (Eurozone). With each model, we try to predict future values for the closing price of the market. By looking at 7-day predictions, we try to draw conclusions on the trend of the market: whether it will go up (uptrend), down (downtrend) or sideways (change is insignificant to prompt trend).

Although in this project we have made predictions based on the whole CAC40 stock market, the same procedure can be applied to get results about a particular company that is part of 40 companies that form the CAC40 itself. Additional model tuning may be required, relating to the particularities of the dataset.

LSTM Vanilla

In the LSTM Vanilla version, we train the base model of LSTM with the CAC40 data. We start by dividing it into a training and test set. Next, we scale the sets separately before starting the training process by LSTM. After we get the training results, we make the predictions and then we return the data to their original scale. Finally, we plot the results and calculate the error.

LSTM Advanced

For the LSTM advanced version[11], we have tried to train our model on different datasets as well as the CAC40 dataset to check if the addition of extra datasets would help with improvement of prediction on LSTM side. In this context, we have tried implementing of LSTM Advanced with French (FCHI), Belgian (BFX) and Amsterdam (AEX) datasets as well as LSTM advanced with German (GDAXI), Spanish (IBEX), Oil and French dataset.

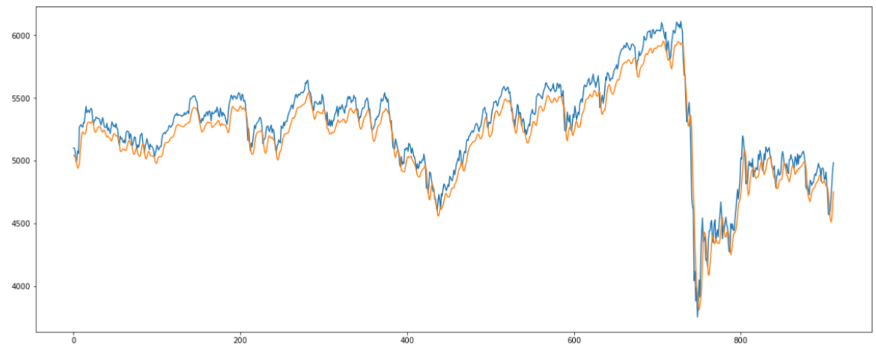


Figure 3: *LSTM Advanced (FCHI,GDAXI,IBEX,Oil) predictions for CAC40 with RMSE=115*

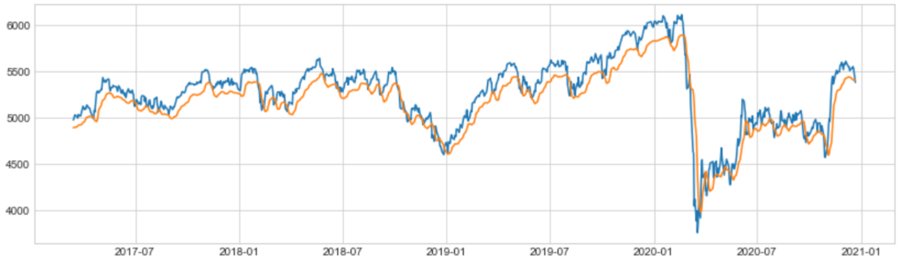


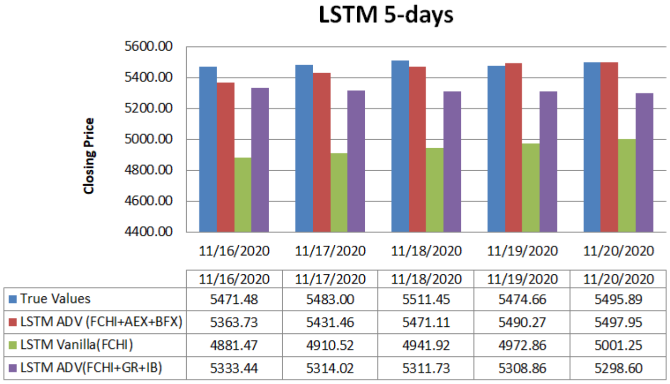
Figure 4: *LSTM Advanced (FCHI,BFX,AEX) predictions for CAC40 with RMSE=166*

LSTM Advanced with German (GDAXI), Spanish (IBEX), Oil and French dataset

In this part, we have tried to train, test and predict with our model on different combinations of the German, Spanish, Oil and French datasets. It is important to mention that we have kept the datasets separated, have normalized them independently and split them independently. Furthermore, we have trained our model through iteration on each of these datasets but we have only tested and done predictions for the CAC40 dataset as it is our focus for this project.

LSTM Advanced with French (FCHI), Belgian (BFX) and Amsterdam (AEX) dataset

Similarly with the version explained above, we have also tested our model with the French (FCHI), Belgian (BFX) and Amsterdam (AEX) dataset. All of them are the three main indices of the pan-European stock exchange group *Euronext*, so we expect that they somehow influence each other and improve our predictions.



	11/16/2020	11/17/2020	11/18/2020	11/19/2020	11/20/2020
🟦 True Values	5471.48	5483.00	5511.45	5474.66	5495.89
🟥 LSTM ADV (FCHI+AEX+BFX)	5363.73	5431.46	5471.11	5490.27	5497.95
🟩 LSTM Vanilla(FCHI)	4881.47	4910.52	4941.92	4972.86	5001.25
🟪 LSTM ADV(FCHI+GR+IB)	5333.44	5314.02	5311.73	5308.86	5298.60

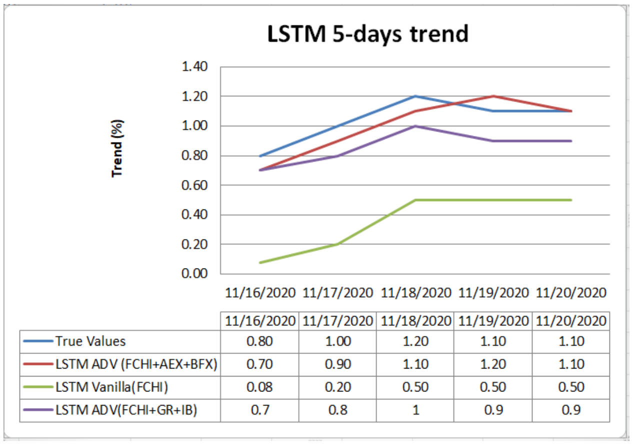
Figure 5: *LSTM 5 days closing price predictions according to different models*

Regarding the focus of our optimization, we have checked and compared results between different implementation by mainly taking into consideration the root mean square deviation (RMSE) as it was quite efficient in understanding whether the tested and predicted values were close to the real values. Further on, upon choosing the best dataset combination to train our model with, we focused on the signal trend of the predicted values and compared them to the trend of the actual dataset. This is something important to do considering that we are not only interested on the values of the prediction but also on the signals that this prediction will generate and how do they reflect reality.

3.3.2 Making forecasts for the future

After we have predicted the past values to see if they are accurate, we are interested now in making forecasts about the future days. We make predictions on a 7-day basis. We use the last 60 values of the closing price in order to predict the next value. Every time we get a new value, we add that in the list and continue to make forecasts (taking into account the predicted ones as well) until we have a set of seven values. Next, we try to plot them and study the trend they give.

To illustrate the future closing price predictions we have chosen a random week and have predicted through 3 different models the closing price for each day of that week. The results given in *Figure 5* show the true values for each of the 5 days of that week (there is no activity during Saturday and Sunday) and the predicted value for each day. We can see that the *LSTM ADV(FCHI+AEX+BFX)* model predictions are closer to the true values and in the table below the chart we can inspect every predicted value. It needs to be mentioned that LSTM as a regression model allows us to make value predictions, contrary to SVM, where we made a classification of the prediction. We also can see that the *LSTM Vanilla* version is the least accurate in the sense that its value predictions have greater difference with the true values compared to other models. In this case,



	11/16/2020	11/17/2020	11/18/2020	11/19/2020	11/20/2020
True Values	0.80	1.00	1.20	1.10	1.10
LSTM ADV (FCHI+AEX+BFX)	0.70	0.90	1.10	1.20	1.10
LSTM Vanilla(FCHI)	0.08	0.20	0.50	0.50	0.50
LSTM ADV(FCHI+GR+IB)	0.7	0.8	1	0.9	0.9

Figure 6: *LSTM 5 days closing price trend predictions according to different models*

this implies that data augmentation in other models plays a significant role in generalization performance. But, the true intention of the future forecasts is not directly the closing price prediction. We are more interested in closing price trend predictions because that will allow us to make investment decisions. A lot of indicators can be used to make these trend predictions and all of them are based on the closing price. In our case, we have chosen a moving average with 15 days window size in order to make comparisons between our three different models in approximating the real values trend.

The trend value for a given day is calculated as the relative difference with the value of the previous day given in percentage. For the chosen week all the trend values are positive and relatively small (not more than 1.2%) which may be interpreted as a stable week in terms of price volatility. It can be noticed that *LSTM ADV(FCHI+AEX+BFX)* has a similar trend behaviour compared with the real values trend although with a negative difference of 0.2%. The *LSTM Vanilla(FCHI)* version is the most inaccurate among the 3 models but still it manages to maintain positive values of trend prediction. These values may not be interpreted as signals to invest since the upward trend percentage is relatively small and may be regarded as sideways trend.

3.3.3 Results

In comparing the combinations between IBEX, GDAXI and Oil, we have come to the conclusion that the lowest RMSE error is achieved for:

- FCHI, IBEX, GDAXI and Oil
- FCHI, IBEX and GDAXI

Between these, while the first does have the lowest RMSE (with the second being not far behind), it also seems to be over-fitting the closing price during the testing phase. In any case, we decided to choose both of these scenarios while doing the predictions.

On the other hand, after testing combinations between AEX and BFX datasets, we achieve the best results with the lowest RMSE for the combination of the three: FCHI, AEX and BFX.

Moreover, we observe that the base version of LSTM has a lower performance, a higher RMSE in comparison to the LSTM Advanced models.

4 Credit risk assessment

In this section we study the credit risk assessment problem.

4.1 Overdraft dataset

4.1.1 Context of the data

Overdraft is a type of revolving credit, granted from the bank to a client, typically when its account reaches zero. The difference between an overdraft and a loan consists in the fact that the overdraft has a fixed duration of one or two years and a fixed fee that the client has to pay each month until the end of the maturity period, when the client is obliged to pay the total amount of the credit.

4.1.2 Data Analysis

Target label

There are 11397 samples in this dataset, of which 10672 samples are good clients and 725 samples are bad clients thus the data is heavily imbalanced, which will definitely reduce the accuracy of model training. Therefore, we should solve the problem of this imbalance in pre-processing step as much as possible.

The following resampling method can change the ratio of positive and negative classes of the original data, that is, reduce the degree of class imbalance.

- Undersampling: Re-extract from the classes with more samples, and only keep a part of these samples.
- Oversampling: copy some points in the minority class to increase its basic number.
- Generate synthetic data: create new synthetic points from a few classes to increase its base.

All these methods have only one purpose: to rebalance (partially or fully) the data set. Purposefully choosing a new ratio can lead to some related methods, but if you do not further consider the essence of the problem and just rebalance the classes, then this process may be meaningless. So in simple terms, when we modify the data set by resampling, we are changing the distribution of the data.

In addition to resampling, we can also add multiple other features to the data set to make the data set richer, so that we may get better accuracy results. Compared with the resampling method, this method uses more information-enriched raw data instead of changing the meaning of the original data.

Time characteristics analysis

For a financial risk control task, time is a very important variable in a general sense. So we decided to deal with *Date of Disbursment* which means the date of loan issuance and *Maturity date* which means the date of repayment.

We subtracted the *Maturity date* from the *Date of Disbursment* to create a new integer type characteristic *issueDateDuration* that represents the time limit for the customer to repay the loan. In addition, we also recorded that the initial time of loan issuance was May 20, 2014. The new feature *issueDateDT* created by subtracting this start date from *Date of Disbursment* corresponds to different customers to a certain extent.

Distribution of numerical variables

For some numerical variables such as *Monthly Income*, the magnitude of the data is too large, about 10000. Secondly, their original distribution does not conform to a normal distribution. We performed a log mathematical processing, which can reduce the order of magnitude and make the distribution more in line with the normal distribution. We also performed the same treatment for the *Approved loan*, *Other installments of other loans*, and *issueDateDuration* features.

Feature derivation

By combining multiple features, we can get some more representative new features. For example, we can divide the time spent in the current position by age to get a ratio value. The new value created may express its relationship with the classification label to a certain extent. We've newly created features :

$$\frac{Monthly\ Installment\ of\ the\ loan}{Approved\ loan} \tag{1}$$

$$\frac{Approved\ loan}{Monthly\ Income} \tag{2}$$

$$\frac{Number\ of\ years\ at\ the\ current\ address}{Age} \tag{3}$$

are all positively correlated related to the classification's label.

Data bucketing

Binning is mainly used to discretize continuous variables and merge discrete variables with multiple categorical values. From the perspective of model effects, feature bucketing is mainly used to reduce the complexity of variables, reduce the impact of variable noise on the model, and improve the correlation between independent variables and dependent variables. This makes the model more stable. The discretized features have strong robustness to abnormal data and are not easily affected by extreme values. And it can avoid the influence of meaningless fluctuations in the characteristics on the model, and the model will be more stable. The binning methods can be divided into two categories: unsupervised method and supervised method. The main unsupervised binning methods are equidistant and equal-frequency. The supervised binning methods include chi-square binning and decision tree binning. The unsupervised binning method ignores the type of the instance, and it is risky that the instance falls in the correct interval.

As a result, we use decision tree binning and information entropy to bin the variables of *Age*, *Current years at job* and *Number of years at the current address*.

After binning, there are two important criteria to judge the quality of binning. First, the continuous bin must be monotonous. The rate of bad sample in different bins should have a trend, either increasing or decreasing. In addition, WOE (weight of evidence) and IV (information value) values can be used to judge the quality of a bin. The IV value reflects the proportion of the number of individuals in the current group to the overall number, and its influence on the predictive ability of variables. Therefore, we often use the IV value to determine the degree of influence of the feature on the target variable. The larger the value, the higher the correlation between the feature and the target variable.

Feature selection

Through the above feature engineering, we have a total of 32 features. A correct mathematical model should be simple in form. We hope to be able to select features that have a better interpretation of the target variable from the constructed features, so that the model has a stronger generalization ability. Through feature selection, we hope to improve the accuracy of prediction, construct faster, lower-consuming prediction models, and have a better understanding and interpretation of the model.

In theory, there are three methods for feature selection: filter, wrapper and embedded.

1. Filter: According to divergence or relevance, each feature is scored, thresholds are set, and features are selected. Commonly used methods include variance selection, pearson correlation coefficient method, chi-square test method and so on.

2. Wrapper: According to the objective function (usually the prediction effect score), select several features at a time, or exclude several features.

The basic idea of stepwise regression is to introduce variables one by one, and each time a variable is introduced, the selected variables must be tested one by one. When the originally introduced variable becomes no longer significant due to the introduction of later variables, it is removed. This process is repeated until no significant variables are selected into the equation, and all significant independent variables are eliminated from the regression equation. Stepwise regression selection feature is widely used in traditional risk control modeling, but it is more complicated to use, and the time cost is high. So we are not making a choice here.

3. Embedded : Firstly use some machine learning algorithms and models for training, get the weight coefficient of each feature, and select the feature from large to small according to the coefficient. Common methods include regular term feature selection and tree model feature selection. The L1 regular method has the characteristic of sparse solution, so it naturally has the characteristic of feature selection. The SelectFromModel class in the feature_selection library combined with the logistic regression model can be used to select features. The learning algorithm of the tree model uses a heuristic method, using indicators such as information gain or information gain ratio or Gini index as the criteria for selecting features, and recursively selecting the best features. The SelectFromModel class combined with the GBDT model can be used.

Here we combined the results of 4 models for feature selection. They are Pearson correlation coefficient method, variance selection, logistic regression with L1 penalty term and GradientBoostingClassifier.

We chose finally 20 features :

Age, Marital Status, Education Type, Number of years at the current address, Employment Type, Current years at job, Credit Registry Information, Monthly Income, Reference, Sex, issueDateDuration, Monthly Income log, Monthly Installment of the loan log, CREDIT INCOME PERCENT, INSTALL CREDIT PERCENT, JOB AGE PERCENT, ADDRESS AGE PERCENT, Current years at jobbins, Number of years at the current address bins and issueDateDT.

4.1.3 Model exploration & Performance evaluation

After selecting and preprocessing the features, we needed to choose the more appropriate model to do the predictions. Ensemble-based models such as random forests are known to give good results on imbalanced data, moreover there are so called "balanced" versions of the algorithms that are designed to work especially well on this kind of data. Here we are going give an overview of the different models tested and their performances.

Balanced Random Forest

The main idea behind Random Forest machine learning models is to optimize many random small decision trees on the data, and then doing a majority voting of the results of these sub-models in order to have the final prediction, these kind of models have proven to be robust in the case of imbalanced labels.

Balanced Random Forest models are based on Random Forest, but the input data is subjected to either oversampling (artificial samples from the minority class are created) or undersampling (samples from the majority class are filtered so that the classes are equally represented.

Improved BRF

The idea behind this approach is to use multiple Balanced Random Forest classifiers to improve classification results, the way this is done is that a first classifier is trained on a dataset, then the samples for which the estimated probability of belonging to a class of another is high are removed from the dataset, and a new model is trained on the remaining data, this can be done with has much steps as necessary (typically 3).

Other Balanced Ensemble Methods

Other balanced versions of ensemble methods (of which random forest classifiers are part) were explored, we won't go into the technical details of each of them, but these include :

- Balanced AdaBoost (called RUSBoost in the following part)
- Easy Ensemble
- Balanced Bagging

Numerical results

Here are the results we got for different models, we see that Balanced Random Forests give slightly better results and that is why this model was kept.

About the performance measures :

- Specificity indicates the proportion of negatives that are correctly identified
- Sensibility indicated the proportion of negatives that are correctly identified
- Accuracy is the global proportion of correctly identified samples
- Receiver operating characteristic Area under curve (ROC AUC) is the area under a curve that indicates the diagnostic ability of a binary classifier as its discrimination threshold is varied, it is an other way to measure the overall performance of a model

Method	Specificity	Sensibility	Accuracy	ROC AUC
BRF	0.732	0.701	0.704	0.716
Easy Ensemble	0.733	0.682	0.689	0.711
Balanced Bagging	0.715	0.685	0.691	0.706
RUSBoost	0.705	0.671	0.702	0.689

4.1.4 Model explainability

The bank has to justify the decision of approving or not approving the credit to the client, therefore we have applied the SHAP method for explaining the results of the chosen Machine Learning model.

SHAP (Shapley Additive explanations) [12] is a method used to explain individual predictions by computing the contribution of each feature to the prediction. SHAP is based on the game theoretically optimal Shapley Values. The feature values of a data instance act as players in a coalition. Shapley values tell us how to fairly distribute the "payout" (= the prediction) among the features.

SHAP specifies the explanation as:

$$g\left(z'\right)=\phi_0+\sum_{j=1}^{M}\phi_j z_j' \tag{4}$$

where g is the explanation model, $z' \in \{0,1\}^M$ is the coalition vector, M is the maximum coalition size and $\phi_j \in R$ is the feature attribution for a feature j, the Shapley values.

KernelSHAP estimates for an instance x the contributions of each feature value to the prediction. KernelSHAP consists of 5 steps:
Sample coalitions $z_k' \in \{0,1\}^M, \quad k \in \{1,\ldots,K\}$(1 = feature present in coalition, 0 = feature absent)

- Get prediction for each z_k' by first converting z_k' to the original feature space and then applying model $f: f\left(h_x\left(z_k'\right)\right)$

- Compute the weight for each z_k' with the SHAP kernel.

- Fit weighted linear model.

- Return Shapley values ϕ_k, the coefficients from the linear model.

[13] proposed TreeSHAP, a variant of SHAP for tree-based machine learning models such as decision trees, random forests and gradient boosted trees.TreeSHAP was introduced as a fast and model-specific alternative to KernelSHAP.

- Global Explainability
  We can calculate global feature importance by averaging the absolute Shapley values per feature across the data. For our overdraft model the most important features are as in the plot below:

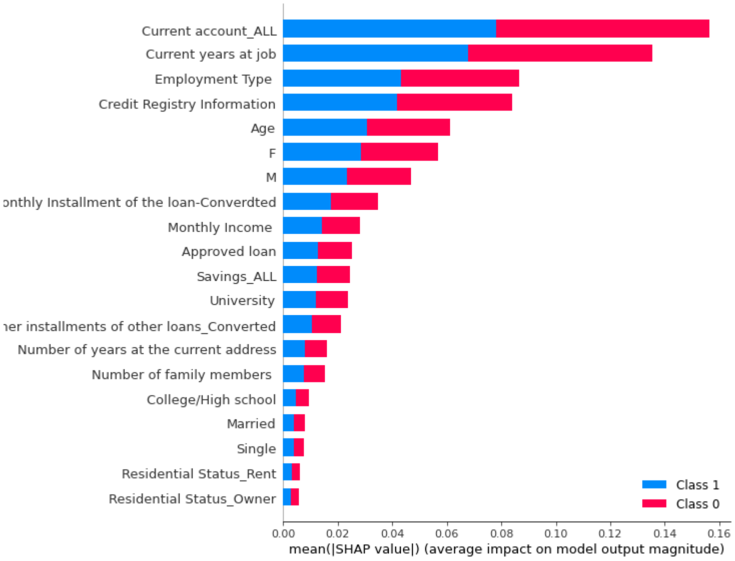


Figure 7: *Global feature importance for overdraft*

- Local Explainability

  Shap values can also be calculated for individual cases, so that for a new client apart from predicting client's capability for paying or not paying the credit we can also explain the impact of each variable in the decision made.

  We have tested with a particular client and the results are as in the image below: Shap values of 5 most important features are as in the table below:

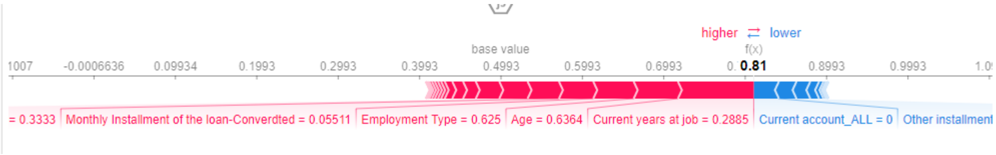


Figure 8: *Local feature importance*

Feature	Shap Value
Current years at job	9.31414336e-02
Age	5.43103166e-02
Employement Type	4.96683509e-02
Monthly Installment of the loan-Converted	4.21402153e-02
Credit Registry	3.85204589e-02

4.2 Generalization of the model

4.2.1 Exploration of datasets

In order to obtain a more general model, which would be capable of dealing with different types of client's data, we have analysed all the available datasets related to the credit assessment's problem.
We have focused on finding a dataset containing a large number of samples and features in the interest of obtaining better generalization capabilities.
Some of the datatsets that we have taken in consideration are presented below:

- UCI repository
  The first dataset we explored is the dataset from UCI repository with all possible dataset for machine learning. We found five different dataset of Australian Credit Approval, German Credit Data, Default of Japan Dataset, Default of Polish Dataset and Default of Credit Card Clients Dataset.
  But after our analysis of the data set, most of the customer groups targeted by these data are not in the European region, and there are not many features that can be shared. The features that can be shared only include credit, age, and monthly income. But when we only use these features, the accuracy of classification will be greatly reduced.

- Home Credit Default Risk
  This dataset is huge, it contains 220 features 307512 clients. We kept the most accurate features but the accuracy results are very low. The reason behind is the fact that we can't use in our context the three most important variable according to Random Forest.

- PAKDD 2010
  We don't seem to get significant results on this dataset using only features relevant to our use case. Also, since the dataset seems to come from Brazil, all the monetary quantities are in réal.

- Fast credit
  This dataset contains only 11 features but they are all usable in our context :
  Gender, Married, Dependents, Self Employed, Applicant Income, Coapplicant Income, Loan Amount, Loan Term, Credit History, Property are
  The main drawback here is the limited number of samples.

Lending club Dataset

We finally selected 13 features for the final model analysis. Including the loan required by the customer, the customer's own information, and the traceability of the loan phenomenon of the customer in the past.

Feature	Explanation
loan-amnt	The amount the customer needs to borrow
int-rate	The interest rate value that the customer hopes to obtain after negotiating with the lender
term	Duration of the loan, 36 months or 60 months are available
emp-length	Length of time the customer has worked in the current position
home-ownership	Customer's housing status, whether it is mortgaged, leased or owned, etc
annual-inc	Customer's annual income
Purpose	Purpose of loans, including debt consolidation, repaying bank card loans, improving housing, etc
delinq-2yrs	Inquire whether the customer has overdue payment in the past two years
inq-last-6mths	Times of the customer has applied for credit in the past 6 months
open-acc	Number of open credit lines in the borrower's credit file
pub-rec	Number of derogatory public records
application-type	Whether the loan is an individual application or a joint application with two co-borrowers
acc-now-delinq	Number of accounts in arrears

4.2.2 Final model

Model exploration was also done for the model used for generalization, and as for the overdraft case, Balanced Random Forest was retained due to performance and relative simplicity.

Class	precision	recall	F1-score	AUC
-1	0.66	0.28	0.39	0.64
1	0.62	0.89	0.73	0.64
accuracy				0.62

4.3 "Credit Simulation" web application

4.3.1 Architecture

The architectural style used for developing the web application is Representational State Transfer (REST).The main project contains two sub projects. One project contains the frontend of our application and the other contains the backend services, representing respectively the client who requests the resources and the server who has the resources.

REST

REST (REpresentational State Transfer) is an architectural style that provides standards between computer systems on the web, facilitating the communication of systems with each other. REST-compliant systems, also known as RESTful systems, are stateless and have a separation of the concerns of client and server.

Angular Framework

Angular is a development platform and application design framework used for developing sophisticated and efficient single-page apps, while Angular Material is an UI component library which offers a variety of ready-to-use components. For developing the frontend of our app we have used some Material components such as forms.

Flask Framework

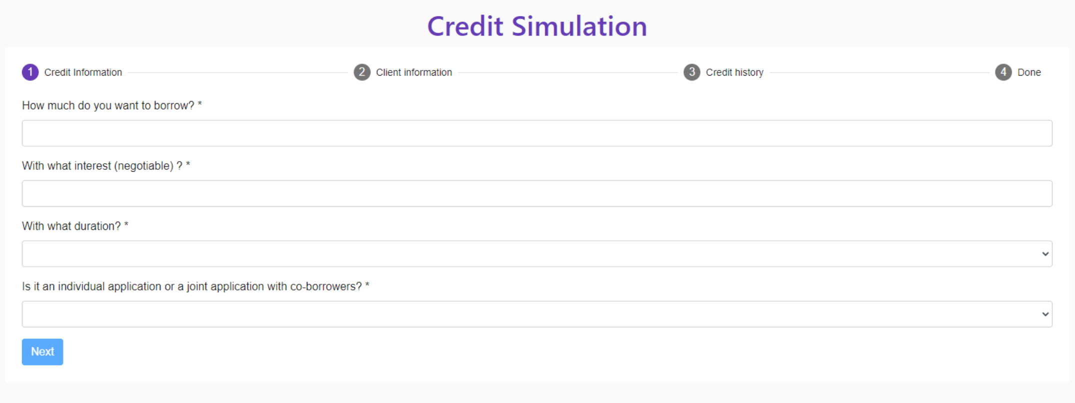
Flask is a web framework which provides tools, libraries and technologies for building web applications. It is written in Python and it is considered as a microframework for not requiring particular tools or libraries. We have used Flask for developing the backend of our project. Flask handles Http Requests coming from the front-end project, while communicating with basic python files which contain our Machine Learning model.

Pickle module

To avoid training our model every time we shut down our Python session, we need to save the classifier we trained and built.For this purpose we have used Python's in-built pickle module which allows us to serialize and deserialize Python objects to compact byte code.

4.3.2 Functionalities

Our app contains a multistep form that requires all the necessary information from the client, so that the client can simulate a credit in order to check his chances of getting an approval from the bank. The form is translated into a JSON object and then sent as a parameter of the http request. The response of this request is the probability that the clients gets an approbation for the credit.



Credit Simulation

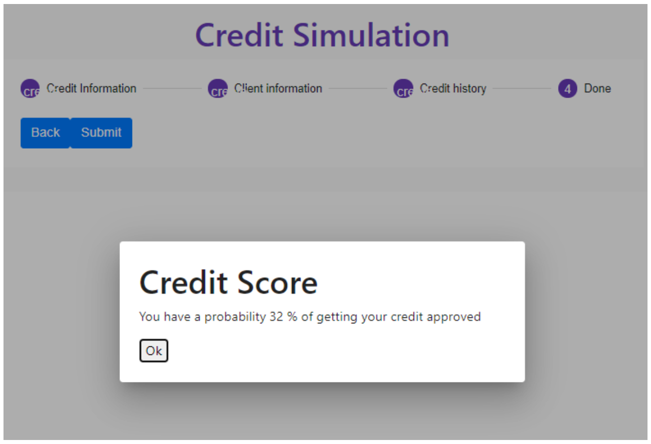
How much do you want to borrow? *

With what interest (negotiable) ? *

With what duration? *

Is it an individual application or a joint application with co-borrowers? *

Figure 9: *Functionalities - Multistep form*



Credit Simulation

Credit Score

You have a probability 32 % of getting your credit approved

Figure 10: *Functionalities - Probability output*

5 Conclusions

In this project, we started by conducting a small study on financial AI applications which helped us create a more clear idea of how the AI and finance world are connected as well as how our two applications are placed.

We then took two problems and applied different Machine Learning models to make predictions.

For the portfolio management problem, we studied previous and state-of-the-art algorithms and implemented for the French stock market dataset. We used SVM as a Machine Learning algorithm to make predictions about the trend of the stock market. Next, we went further and applied a Deep Learning method (LSTM) to make near-future predictions, something that is out of the capacity of the SVM model implemented during the first part of the project. We implemented and tested two different versions and made 7-day forecasts for the CAC40 stock market. In comparing results, it is safe to say that the LSTM Advanced version outperformed the LSTM base version in both RMSE score and trend prediction capabilities. This result is compatible with the result that was obtained from [11].

For the credit risk assessment problem, we have analysed the overdraft data and after implementing various methods we concluded that the best method to use is Balanced Random Forest. We have also worked on the generalization of the model, such that our model can be implemented to any type of costumer's data. After analysing all the open source datasets related to the credit assessment project, we decided to use the Lending club dataset, considering the big number of samples and features it contains. Finally, we have implemented our Machine Learning model into a web application for credit simulation.

References

[1] 10 Applications of Machine Learning in Finance
`https://algorithmxlab.com/blog/applications-machine-learning-finance`

[2] Top 10 applications of Machine Learning in Finance
`https://medium.com/breathe-publication/top-10-applications-of-machine-learning-in-finance-9bfc911faf3f`

[3] United States Department of the Treasury ”History of Anti-Money Laundering Laws” :
`https://www.fincen.gov/history-anti-money-laundering-laws`

[4] M. Jullum, A. Løland, R. B. Huseby, G. Ånonsen, J. Lorentzen ”Detecting money laundering transactions with machine learning” - Journal of Money Laundering Control (2020)

[5] E.A. Mirestineanu, G. Mesnita “An Analysis of the Most Used Machine Learning Algorithms for Online Fraud Detection” (2019)

[6] L. Guo, F. Shi, J. Tu “Textual analysis and machine leaning: Crack unstructured data in finance and accounting” - The Journal of Finance and Data Science 2 (2016)

[7] Alexandre Kowalczyk Support Vector Machines Succinctly

[8] Haifei Zhang Code : `https://github.com/Haifei-ZHANG/iQuant`

[9] Nelson, David & Pereira, Adriano & de Oliveira, Renato. (2017). Stock market’s price movement prediction with LSTM neural networks. 1419-1426. 10.1109/IJCNN.2017.7966019.

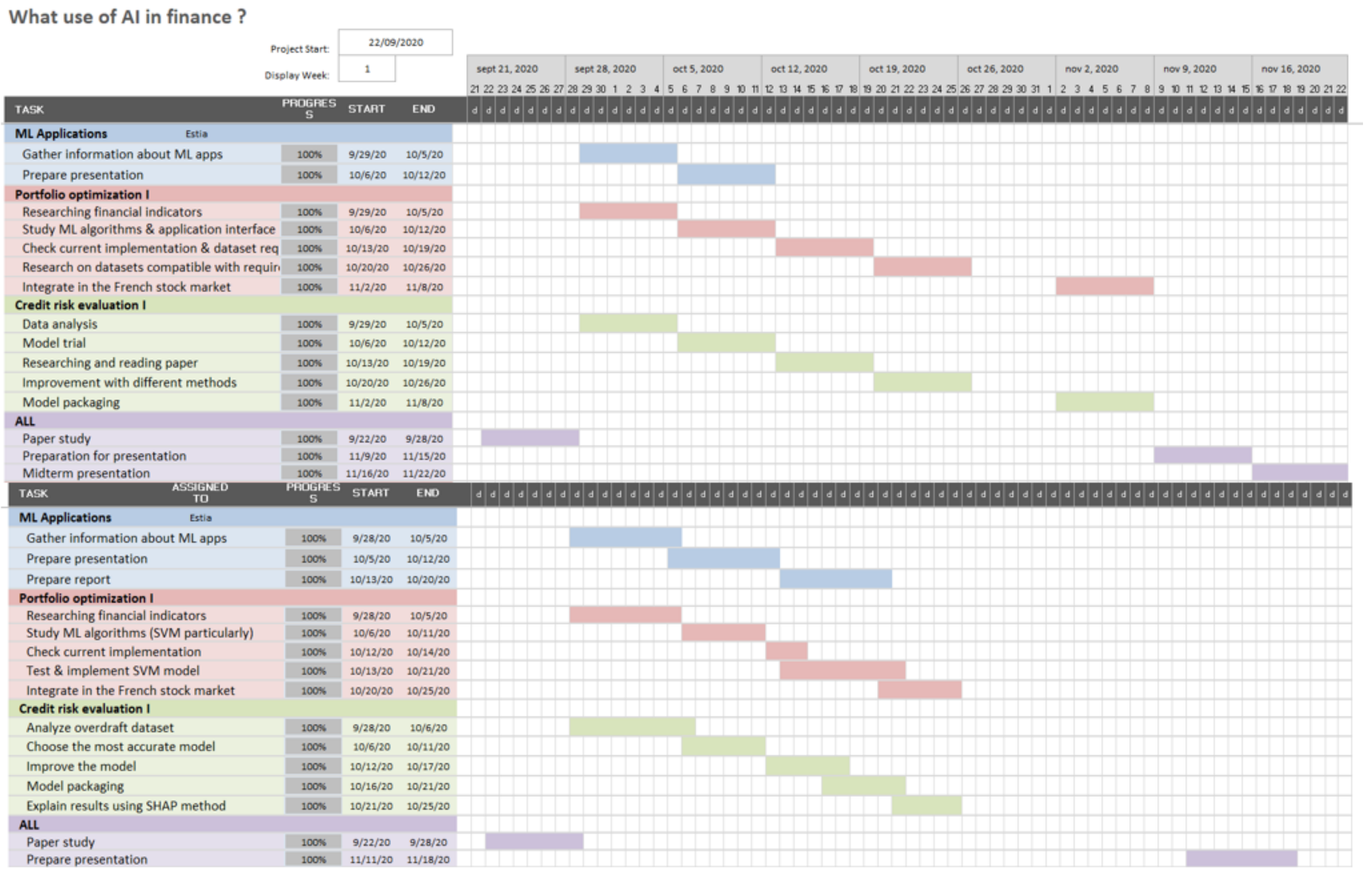
[10] Sezer, Omer & Gudelek, Ugur & Ozbayoglu, Murat. (2019). Financial Time Series Forecasting with Deep Learning : A Systematic Literature Review: 2005-2019.

[11] Lakshminarayanan, S. and John P. McCrae. “A Comparative Study of SVM and LSTM Deep Learning Algorithms for Stock Market Prediction.” AICS (2019).

[12] Lundberg, Scott M., and Su-In Lee ”A unified approach to interpreting model predictions.” Advances in Neural Information Processing Systems” - (2017)

[13] Lundberg, Scott M., Gabriel G. Erion, and Su-In Lee ”Consistent individualized feature attribution for tree ensembles” - (2018)

A Gantt Charts



What use of AI in finance ?

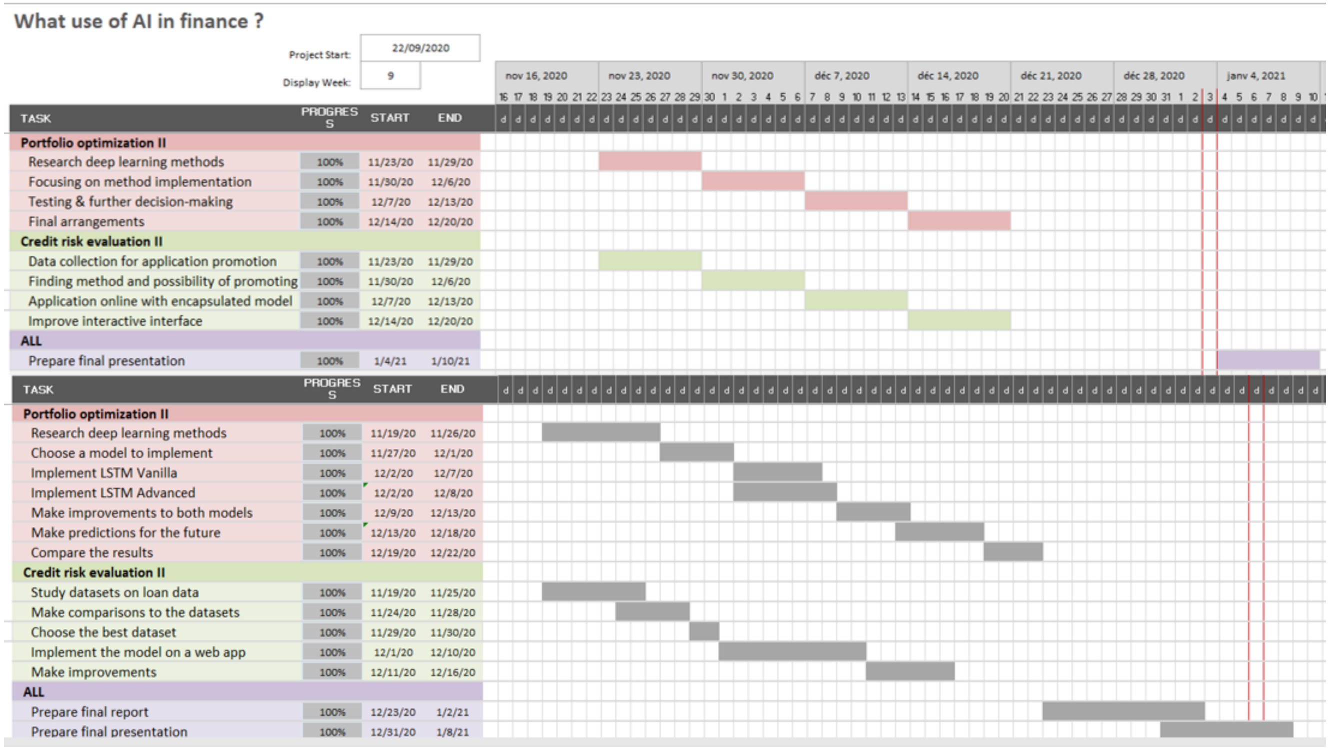
Project Start: 22/09/2020

Display Week: 1

TASK		PROGRESS	START	END
ML Applications	Estia			
Gather information about ML apps		100%	9/29/20	10/5/20
Prepare presentation		100%	10/6/20	10/12/20
Portfolio optimization I				
Researching financial indicators		100%	9/29/20	10/5/20
Study ML algorithms & application interface		100%	10/6/20	10/12/20
Check current implementation & dataset req		100%	10/13/20	10/19/20
Research on datasets compatible with requir		100%	10/20/20	10/26/20
Integrate in the French stock market		100%	11/2/20	11/8/20
Credit risk evaluation I				
Data analysis		100%	9/29/20	10/5/20
Model trial		100%	10/6/20	10/12/20
Researching and reading paper		100%	10/13/20	10/19/20
Improvement with different methods		100%	10/20/20	10/26/20
Model packaging		100%	11/2/20	11/8/20
ALL				
Paper study		100%	9/22/20	9/28/20
Preparation for presentation		100%	11/9/20	11/15/20
Midterm presentation		100%	11/16/20	11/22/20

TASK	ASSIGNED TO	PROGRESS	START	END
ML Applications	Estia			
Gather information about ML apps		100%	9/28/20	10/5/20
Prepare presentation		100%	10/5/20	10/12/20
Prepare report		100%	10/13/20	10/20/20
Portfolio optimization I				
Researching financial indicators		100%	9/28/20	10/5/20
Study ML algorithms (SVM particularly)		100%	10/6/20	10/11/20
Check current implementation		100%	10/12/20	10/14/20
Test & implement SVM model		100%	10/13/20	10/21/20
Integrate in the French stock market		100%	10/20/20	10/25/20
Credit risk evaluation I				
Analyze overdraft dataset		100%	9/28/20	10/6/20
Choose the most accurate model		100%	10/6/20	10/11/20
Improve the model		100%	10/12/20	10/17/20
Model packaging		100%	10/16/20	10/21/20
Explain results using SHAP method		100%	10/21/20	10/25/20
ALL				
Paper study		100%	9/22/20	9/28/20
Prepare presentation		100%	11/11/20	11/18/20

Figure 11: Gantt chart at the start & the end of the project: Part I



What use of AI in finance ?

Project Start: 22/09/2020

Display Week: 9

TASK	PROGRESS	START	END
Portfolio optimization II			
Research deep learning methods	100%	11/23/20	11/29/20
Focusing on method implementation	100%	11/30/20	12/6/20
Testing & further decision-making	100%	12/7/20	12/13/20
Final arrangements	100%	12/14/20	12/20/20
Credit risk evaluation II			
Data collection for application promotion	100%	11/23/20	11/29/20
Finding method and possibility of promoting	100%	11/30/20	12/6/20
Application online with encapsulated model	100%	12/7/20	12/13/20
Improve interactive interface	100%	12/14/20	12/20/20
ALL			
Prepare final presentation	100%	1/4/21	1/10/21

TASK	PROGRESS	START	END
Portfolio optimization II			
Research deep learning methods	100%	11/19/20	11/26/20
Choose a model to implement	100%	11/27/20	12/1/20
Implement LSTM Vanilla	100%	12/2/20	12/7/20
Implement LSTM Advanced	100%	12/2/20	12/8/20
Make improvements to both models	100%	12/9/20	12/13/20
Make predictions for the future	100%	12/13/20	12/18/20
Compare the results	100%	12/19/20	12/22/20
Credit risk evaluation II			
Study datasets on loan data	100%	11/19/20	11/25/20
Make comparisons to the datasets	100%	11/24/20	11/28/20
Choose the best dataset	100%	11/29/20	11/30/20
Implement the model on a web app	100%	12/1/20	12/10/20
Make improvements	100%	12/11/20	12/16/20
ALL			
Prepare final report	100%	12/23/20	1/2/21
Prepare final presentation	100%	12/31/20	1/8/21

Figure 12: Gantt chart at the start & the end of the project: Part II