Detecting crisis event with Gradient Boosting Decision Trees

Eric Benhamou, Jean Jacques Ohana, David Saltiel, Beatrice Guez

Detecting crisis event with Gradient Boosting Decision Trees

Eric Benhamou[1,2,3], **Jean Jacques Ohana**[3], **David Saltiel**[3] , **Beatrice Guez**[3]
[1]Miles Lamsade Dauphine University, France, [2]EB Ai Advisory, France, [3]Ai For Alpha, France

Abstract

Financial markets allocation is a difficult task as the method needs to dramatically change its behavior when facing very rare black swan events like crises that shift market regime. In order to address this challenge, we present a gradient boosting decision trees (GBDT) approach to predict large price drops in equity indexes from a set of 150 technical, fundamental and macroeconomic features. We report an improved accuracy of GBDT over other machine learning (ML) methods on the S&P 500 futures prices. We show that retaining fewer and carefully selected features provides improvements across all ML approaches. We show that this model has a strong predictive power. We train the model from 2000 to 2014, a period where various crises have been observed and use a validation period of 3 years to find hyperparameters. The fitted model timely forecasts the Covid crisis giving us a planning method for early detection of potential future crises.

Introduction

Numerous studies on stock markets have shown that in a normal regime, equity markets are rising steadily as investors get rewarded for risk-taking (Siegel 2007). This is related to the so called equity risk premium (see (Mehra and Prescott 1985a) or (Fama 1965)) that creates an upward trend in stocks markets. More specifically, U.S. stocks have returned 6.5 to 7% inflation-adjusted annual returns over the last 200 years. This makes a passive buy-and-hold investment strategy hard to outperform, unless one is able to accurately plan when stock markets exit the normal rising regime. This dramatic shift from normal to Black swan crisis regime makes the planning exercise of portfolio allocation very arduous. In order to address this crucial challenge, we devise a gradient boosting decision trees (GBDT) approach to predict large price drops in equity indexes. The idea of simply determining if stock markets are in a normal or a 'crash regime' is motivated by two major observations:

- The exercise of identifying two simple regimes is much more realistic than trying to forecast stock market returns themselves, as stock markets are notoriously non-stationary and unpredictable especially when facing tail events (Taleb 2007).

- There are repeated patterns in the alternance of financial booms and busts (Sornette 2003). We can highlight two stylized facts well documented in the economics literature. First, some features portend a crisis regime in the near future, such as herding behavior and leverage increase, resulting in overextended upward trends in equity markets (Rodriguez and Sbuelz 2006). These features take their ground in various psychological biases (Kahneman 2011). Second, stock market crashes are often led by an increase in credit spreads, a flight-to-quality and a slump in cyclical commodities prices (Caballero and Krishnamurthy 2008).

As presented in (Samitas, Kampouris, and Dimitris 2020), machine learning and in particular planning techniques in machine learning can provide an early signal to predict financial crises. The article emphasizes that regional crashes may spread to the whole market, increase the probability of re-occurrence of crises in the near term and show universal and characteristic behavior that machine learning can capture. Likewise, (Gu, Kelly, and Xiu 2020) proved that machine learning techniques are able to extract and identify dominant predictive signals, that includes variations on momentum, liquidity, and volatility. They show that machine learning methods are able to provide predictive gains thanks to capturing nonlinear interactions missed by other methods.The aim of this article is to present a machine learning planning algorithm that captures universal and reproducible behaviours to timely invest in and divest from equity markets.

On another theme, (Benhamou et al. 2021b), (Benhamou et al. 2021a), (Benhamou et al. 2020c) or (Benhamou et al. 2020b) showed that planning techniques using deep reinforcement learning are a good alternative to traditional portfolio methods. However, if one wants specifically to target crisis detection, a good alternative is rather to tackle this planning exercise as a supervised learning problem. We therefore devise a planning algorithm specifically geared toward crisis detection that have the following characteristics:

- The planning method leverages more than 150 features ranging from traditional financial and economics variables to technical variables.
- It aims specifically at determining if we are in normal or crisis regime by classifying future returns position below

or above the historical 5 percentile level

- In the method, we tackle the issue of imbalanced data and feature selection by doing sequential features selection. We find that this feature selection is critical to achieve good model performance.

Using modern machine-learning terminology, determining whether stock markets are in a normal or crash regime from various fundamental and technical features, is referred to as a 'classification problem'. The problem is stated as follows. We say that a market is in normal regime if its return is above its 5 percentile. In spirit, it is similar to the exercise of looking at images that have cats and dogs and using raw pixels to determine to which class (cat or dog) the image belongs. Because the answer can depend on the time horizon considered, we examine in turn each of the 15, 20 and 25 days horizons. The choice of these horizons is motivated by the fact that 20 business days correspond to a month, a period that is sufficient to capture a significant price trend, but not long enough to be influenced by exogenous events that would deteriorate the predictive power of the model. For each market and horizon, a model that predicts the crash probability is derived. Each model is then weighed proportionally to its past Sharpe Ratio over the last two years on the training set. The concept of mixing various models is referred to as an 'ensemble approach' and is known to bring robustness and precision to machine learning models (see for instance (Breiman 1996). The underlying machine learning model is a model available out-of-the-shelves, making the approach understandable and not too complicated to reproduce.

Related works

Our work can be related to the ever growing theme of using machine learning in financial markets. Indeed, with increasing competition and pace in the financial markets, robust forecasting methods has become a vital subject for asset managers. The promise of machine learning algorithms to offer a way to find and model non-linearity in time series has attracted lot of attention and efforts that can be traced back as early as the late 2000's where machine learning started to pick up. Instead of listing the large amount of works, we will refer readers to various works that reviewed the existing literature in chronological order.

In 2009, (Atsalakis and Valavanis 2009) surveyed already more than 100 related published articles using neural and neuro-fuzzy techniques derived and applied to forecast stock markets, or discussing classifications of financial market data and forecasting methods. In 2010, (Li and Ma 2010) gave a survey on the application of artificial neural networks in forecasting financial market prices, including exchange rates, stock prices, and financial crisis prediction as well as option pricing. And the stream of machine learning was not only based on neural network but also generic and evolutionary algorithms as reviewed in (Aguilar-Rivera, Valenzuela-Rendón, and Rodríguez-Ortiz 2015).

More recently, (Xing, Cambria, and Welsch 2018) reviewed the application of cutting-edge NLP techniques for financial forecasting, using text from financial news or twitters. (Rundo et al. 2019) covered the wider topic of usage of machine learning techniques, including deep learning, to financial portfolio allocation and optimization systems. (Nti, Adekoya, and Weyori 2019) focused on the usage of support vector machine and artificial neural networks to forecast prices and regimes based on fundamental and technical analysis. Later on, (Shah, Isah, and Zulkernine 2019) discussed some of the challenges and research opportunities, including issues for algorithmic trading, back testing and live testing on single stocks and more generally prediction in financial market. Finally, (Sezer, Gudelek, and Ozbayoglu 2019) reviewed not only deep learning methods but also other machine learning methods to forecast financial times. As the hype has been recently mostly on deep learning, it is not a surprise that most of their reviewed works are on deep learning. The only work cited that is gradient boosted decision tree is (Krauss, Do, and Huck 2017)

Recently a growing theme of planning method for portfolio allocation using deep reinforcement learning has emerged either applied to a few strategies (Benhamou et al. 2021b), or to the appropriate timing of hedging strategies (Benhamou et al. 2020b), (Benhamou et al. 2020c), (Benhamou et al. 2021c). The challenge in this approach is twofold: deep reinforcement learning does not specifically address the question of forecasting regime and can be less efficient than a supervised learning method geared toward crisis prediction. Model explanability can also be challenging as opposed to GBDT methods that can leverage Shapley values and various explainable AI methods (Zhang, Yi, and Chen 2020) or (Ohana et al. 2021).

Another stream of research in machine learning applied to finance have been to review the best algorithms for predicting financial markets. With only a few exceptions, these papers argue that deep networks outperform traditional machine learning techniques, like support vector machine or logistic regression. There is however the notable exception of (Ballings et al. 2015) that argue that Random Forest is the best algorithm when compared with peers like Support Vector Machines, Kernel Factory, AdaBoost, Neural Networks, K-Nearest Neighbors and Logistic Regression. Indeed, for high frequency trading and a large amount of input data types coming from financial news and twitter, it comes at no surprise that deep learning is the method of choice as it can incorporate large amount of input data types and in particular text inputs. But when it comes to small data set like daily data with properly formatted data from times series, the real choice of the best machine learning is not so obvious.

Interestingly, Gradient boosting decision trees (GBDT) are almost non-existent in the financial market forecasting literature. One can argue that GBDT are well known to suffer from over fitting when tackling regression problems. However, they are the method of choice for classification problems as reported by the machine learning platform Kaggle. In finance, the only space where GBDT are really cited in the literature is the credit scoring and retail banking. For instance, (Brown and Mues 2012) or (Marceau et al. 2019) reported that GBDT are the best ML method for this specific task as they can cope with limited amount of data and very imbalanced classes.

If we are interested in classifying stock market into two regimes: a normal rising one and a crisis one, we are precisely facing very imbalanced classes and a binary classification challenge. In addition, if we are looking at daily observations, we have also a machine learning problem with limited number of data. This two points can hinder seriously the performance of deep learning algorithms that are well known to be data greedy. Hence, our work has consisted in researching whether GBDT can provide a suitable method to identify regimes in stock markets. In addition, as a byproduct, GBDT provide explicit rules (even if they can be quite complex) as opposed to deep learning making it an ideal candidate to investigate regime qualification for stock markets. In this work, we apply our methodology to the US S&P 500 future. Naturally, this can be easily transposed and extended to other main stock markets like the Nasdaq, the Eurostoxx, the FTSE, the Nikkei or the MSCI Emerging future.

Contribution

Our contributions are threefold:

- We specify a valid methodology using GBDT to do planning in financial markets and in particular to determine regimes in financial markets, based on a combination of more than 150 features including financial metrics, macro economics, risk aversion, price and technical indicators. Not only does this provide a suitable explanation for current equity levels thanks to features analysis but it also provides a tool to attempt for early signals should a turn point in the market come.
- We discuss in greater details technical subtleties for imbalance data sets and features selection that is key for the success of this methods. We show that for many other machine learning algorithm, selecting fewer very specific features provides improvement across all methods.
- Finally, we compare this methodology with other machine learning (ML) methods and report improved accuracy of GBDT over other ML methods on the S& P 500 future. We finally apply the method to other equity index futures and show that the method works accross all these 7 equity index futures.

Data used

We screen more than 150 variables summarized in figure 1, belonging to the following categories:

- The Risk Aversion metrics include the equities' and G10/emerging currencies' implied volatilities, the High Yield corporate credit bonds credit spreads, and the shape of the VIX forward curve, defined as the ratio of the VIX Spot over the VIX three-month forward. These indicators characterize the financial assets' liquidity conditions or the accessibility of funding, two complementary measures of risk appetite.
- Financial metrics include the one month, six months and one year growth of Earnings per Share, Price/Earnings and Price/Sales for each equity index. These indicators predict the earnings and sales growth cycle, while providing an insight into valuation multiples changes.
- Macroeconomic indicators consist of the Citigroup Economic Surprise indices in the main economic zones (US, EU, Japan, Emerging, Worldwide). These indicators convey the cycle of positive or negative economic surprises on a daily basis.
- US Yields change (10 years yield, 2 years yield, 10 year breakeven, US Libor) over the same horizons: one month, six months and one year. A change in yields may either reflect the business cycle, the inflation cycle, or the monetary stance of the Federal Reserve.
- The steepness of the US yield curve is also computed as a difference between the government bond yield rate and the short term LIBOR rate on two distinct maturities (10 years, 2 years). This indicator is a well-known predictor of the economic cycle as it computes the spread between long term and short term rates.
- Technical indicators comprise the put/call ratio (as provided by the CBOE), and the market breadth (the percentage of individual stocks above their respective 200 days Moving Average) on the six equity indices and the MSCI World ACWI. The Put/Call ratio may reflect extreme optimism or pessimism in the investors' consensus while market breadth characterizes the unweighted average participation of individual stocks among the global equity indices.
- Last but not least, technical indicators from various asset classes are analyzed:
  - Excess returns of six equity indices, BCOM Energy and Industrial Metals, FX Emerging Bloomberg Index Excess Return (reflecting the aggregate evolution of 8 emerging currencies vs. the dollar), dollar index, as computed by the ICE US. Returns are computed over the same time horizons as before (one month, six months and one year),
  - Historical volatilities, computed over horizons of 10, 20 and 30 days,
  - Distance to 250 days and 500 days moving average.
  - Sharpe Ratios of all the above-mentioned assets, evaluated over horizons of 6 months and 1 year.

Cyclical commodities, the dollar index as well as emerging currencies are often leading indicators of the economic cycle. Furthermore, cyclical asset returns and volatilities may either be used procyclically or countercyclically to predict an incoming crisis. It is well known that in case of positive or negative bubble bursts, there are extreme market reversals (see (Kent and Moskowitz 2015)). Overall, 102 features are built upon the above variables. These features are used to predict the crash probability in each of the six equity markets.

These features capture the universal behaviors documented in (Kahneman 2011), namely herding and trending behavior, cross-market contagions, leverage procyclicality etc. They also contained a mix of fundamental and technical indicators to capture the two main approaches used in the asset management industry.

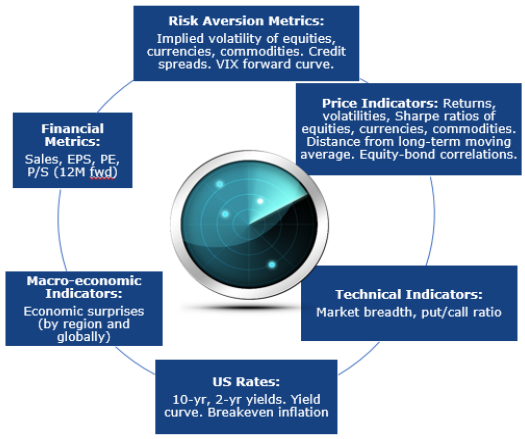

Figure 1: Features used

Why GBDT?

The motivations for Gradient boosting decision trees (GBDT) are multiple:

- GBDT are well know methods to provide state of the art ML methods for small data sets and classification problems. They are supposed to perform better than their state of the art brother, Deep Learning methods, for small data sets. In particular, GBDT methods have been one of the preferred methods from Kagglers and have won multiple challenges.
- GBDT methods can handle data without any prior re-scaling as opposed to logistic regression or any penalized methods. Hence they are less sensitive to data re-scaling
- they can cope with imbalanced data sets as detailed in section .
- when using the leaf-wise use leaf-wise tree growth compared to level-wise tree growth, they provide very fast training.

Methodology

In a normal regime, equity markets are rising as investors are paid for their risks. This has been referred to as the equity premium in the financial economics literature (Mehra and Prescott 1985b). However, there are subsequent down turns when financial markets are in panic and falling. Hence, we can simply assume that there are two regimes for equity markets:

- a *normal* regime where an asset manager should be long to benefit from the long bias of equity markets.
- and a *crisis* regime, where an asset manager should either reduce its equity exposure or even sell short it if the strategy is a long short one.

We formally say that we are in crisis regime if returns are below the historical 5 percentile computed on the training data set. The parameter 5 is not taken randomly but has been validated historically to provide meaningful levels, indicative of real panic and more importantly forecastable. For instance for the S&P 500 market, typical levels are returns at minus 6 to minus 5 percents over a period of 15 days. To make our prediction whether the coming 15 days return will be below 5 percentile (hence be classified as in crisis regime), we use more than 150 features described later on as they deserve a full description. Simply speaking these 150 features are variables ranging from implied volatility of equities, currencies, commodities, credit and VIX forward curve, to financial metrics indicators like 12 month forward estimates for sales, earning per share, price earning, macro economics surprise indexes (like the aggregated Citigroup index that compiles and Z-scores most important economic difference for major figures like ISM numbers, non farm payrolls, unemployment rates, etc).

We are looking explicitly at only two regimes with a specific focus on tailed events on the returns distribution because we found that it is easier to characterize extreme returns than to predict returns using our set of financial features. In machine learning language, our regime detection problem is a pure supervised learning exercise, with two classes classification. Hence the probability of being in the normal regime is precisely the opposite of the crisis regime probability.

In the rest, we assume daily price data are denoted by P_t. The return over a period p is simply given by the corresponding percentage change over the period: $R_t^d = P_t/P_{t-d} - 1$. The crisis regime is determined by the subset of events where returns are lower or equal to the historical 5 percentile or centile denoted by C. Returns that are below this threshold are labeled 1 while the label value for the normal regime is set to 0. Using traditional binary classification formalism, we denote the training data $X = \{x_i\}_i = 1^N$ with $x_i \in \mathbb{R}^D$ and their corresponding labels $Y = \{y_i\}_{i=1}^N$ with $y_i \in 0, 1$. The goal of our classification is to find the best *classification* function $F^*(x)$ according to the sum of some specific loss function $\mathcal{L}(y_i, F(x_i))$ as follows:

$$F^* = \underset{F}{\operatorname{argmin}} \sum_{i=1}^{N} \mathcal{L}(y_i, F(x_i))$$

Gradient boosting considers the function estimation of F to be in additive form where T is the number of boosted rounds:

$$F(x) = \sum_{m=1}^{T} f_m(x)$$

where T is the number of iterations. The set of weak learners $f_m(x)$ are designed in an incremental fashion. At the m-th stage, the newly added function, f_m is chosen to optimize the aggregated loss while keeping the previous found weak learners $\{f_j\}_{j=1}^{m-1}$ fixed. Each function f_m belongs to a set of parameterized base learners that are modeled as decision trees. Hence, in GBDT, there is an obvious design choice between taking a large number of boosted round and very simple based decision trees or a limited number of base learners but of large size. In other words, we can decide to use a small boosted round and a large decision trees whose complexity is mostly driven by its maximum depth or we can alternatively choose a large boosted round and very simple

decision trees. In our experience, it is better to take small decision trees to avoid over-fitting and an important number of boosted round. In our experiment, we use 500 boosted rounds. The intuition between this design choice is to prefer a large crowd of experts that can not memorize data and hence should not over fit compared to a small number of strong experts that are represented by large decision trees. If these trees go wrong, their failure is not averaged as opposed to the first solution. Typical implementations of GBDT are XGBoost as presented in (Chen and Guestrin 2016), LightGBM as presented (Ke et al. 2017), or Catboost as presented (Prokhorenkova et al. 2018). We tested both XGBoost and LightGBM and found an improvement in terms of speed of three time faster for LighGBM compared to XGBoost for similar learning performances. Hence, in the rest of the paper, whenever we will be mentioning GBDT, it will be indeed LightGBM.

To make experiments, we take daily historical returns for the S&P 500 merged back-adjusted futures prices. Our daily observations are from 01Jan2003 to 15Jan2021. We split our data into three subsets:

- a train data set from 01Jan2003 to 31Dec2014
- a validation data set used to find best hyper-parameters from 01Jan2015 to 31Dec2017
- and a test data set from 01Jan2018 to 15 Jan2021

GBDT hyperparamers

GBDT have a lot of hyper parameters to specify. To our experience, the following hyper parameters are very relevant for imbalanced data sets and need to be fine tuned using evolutionary optimisations as presented in (Benhamou et al. 2019):

- min sum hessian in leaf
- min gain to split
- feature fraction
- bagging fraction
- lambda l2

There is a parameter playing a central role in the proper use of GBDT which is the max depth. On the S&P 500 future, we found that very small trees with a max depth of one performs better over time than any larger tree. These 5 parameters mentioned above are determined as the best hyper parameters on the validation set.

Process of features selection

Using all the raw features would add too much noise to our model and would lead to bias decision. We thus need to select or extract the main meaning full features. As we can see in figure 2, we do so by removing the features in 2 steps.

- Based on gradient boosting trees, we rank the features by importance or contribution.
- We then pay attention to the severity of multicollinearity in an ordinary least squares regression analysis by computing the variance inflation factor (VIF) to remove colinear features. Considering a linear model $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + .. + \beta_n X_n + \epsilon$, the VIF is equal to $\frac{1}{1-R_j^2}$, with R_j^2 the multiple 2 for the regression of X_j. The VIF reflects all other factors that influence the uncertainty in the coefficient estimates.

At the end of this 2-part process, we only keep 33% of the initial dataset.

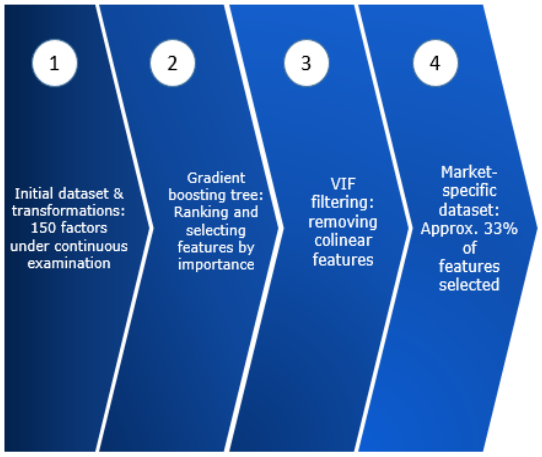

Figure 2: Features selection process

It is interesting to validate that removing many data makes the model more robust and less prone to overfiting. In the next section, we will validate this point experimentally.

Results

Model presentation

Although our work is mostly describing the GBDT model, we compare it against common machine learning models. Hence we compare our model with four other models:

- RBF SVM that is a support vector model with a radial basis function kernel denoted and with a γ parameter of 2 and a C parameter of 1. We use the sklearn implementation. The two hyper parameters γ and C are found on the validation set.
- a Random Forest model whose max depth is taken to 1 and its boosted round to 500. On purpose, we take similar parameters as for our GBDT model so that we benefit from the averaging principle of taking a large boosted round and small decision trees. We found that for annual validation data set ranging from year 2015 on-wards and for the S&P 500 markets, the combination of a small max depth and a large number of boosted rounds performs well.
- a first deep learning model, referred in our experiment as Deep FC (for fully connected layers) that is naive built with three fully connected layers (64, 32 and one for the final layer) with a drop out in of 5 % between and Relu activation, whose implementation details rely on tensorflow keras 2.0
- a second more advance deep learning model consisting of two layers referred in our experiment as Deep LSTM: a

64 nodes LSTM layer followed by a 5% dropout followed by a 32 nodes dense layer followed by a dense layer with a single node and a sigmoid activation.

For both deep learning models, we use a standard Adam optimizer whose benefit is combine adaptive gradient descent with root mean square propagation (Kingma and Ba 2014).

For each model, we train them either using the full data set of features or only the remaining features that are resulting from the features selection process as described in 2. Hence, for each model, we add a suffix ' raw' or ' FS' to specify if the model is trained on the full data set or after features selections. We provide the performance of these models according to different metrics, namely accuracy, precision, recall, f1-score, auc and auc-pr in tables 1 and 2. The GBDT with features selection is among all metrics superior and outperform the deep learning model based on LSTM validating our assumption that on small and imbalanced data set, GBDT outperform deep learning models. In tables 3 and 4, we measure the improvement of the model with feature selection. We specifically make the difference between the value obtained for the model with feature selection and the same model without feature selection. We can see that using a lower and more sparse number of feature improves dramatically the performance of all models, as measured by the AUC and AUC pr metric.

AUC graphics

Figure 3 provides the ROC Curve for the two best performing models, namely the GBDT and the Deep learning LSTM model with features selection. Simply said, ROC curves enables to visualize and analyse the relationship between precision and recall and to stress test the model whether it makes more error of type I or error of type II when trying to find the right answer. The receiver operating characteristic (ROC) curve plots the true positive rate (sensitivity) on the vertical axis against the false positive rate (1 - specificity, fall-out) on the horizontal axis for all possible threshold values. We can notice that the two curves are well above the *blind guess* benchmark that is represented by the dotted red line. This effectively demonstrates that these two models have some predictability power, although being far from a perfect score that will be represented by a half square. The ROC curve also gives some intuition whether a model is rather concentrating on accuracy or recall precision. In an ideal world, if the ROC curve of the model was above all other models' ROC curve, it will Pareto dominates all other and will be the best choice without any doubt. Here, we see that the area under the curve for the GBDT with features selection is 0.83 to be compared with 0.74 which is the one of the second best model, namely the Deep LSTM model with also Features selection. The curve of the first best model GBDT represented in blue is mostly over the one of the second best model the Deep LSTM model. This indicates that in most situations, we expect this model to perform better than the Deep LSTM model

Dealing with imbalanced data

Machine learning algorithms work best when samples number in each class are about equal. However, when one or

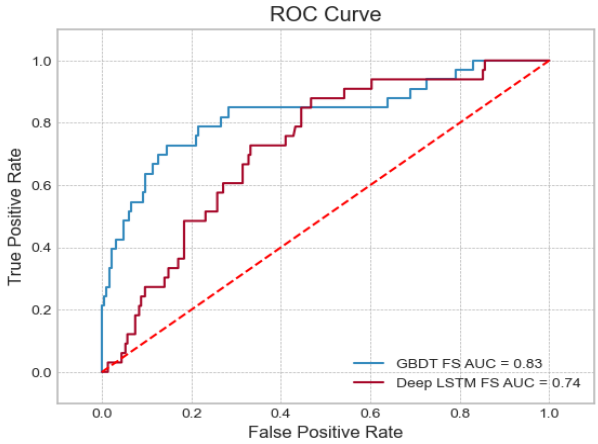

Figure 3: ROC Curve of the two best models

Table 1: Model performance

Model	accuracy	precision	recall
GBDT FS	**0.89**	**0.55**	**0.55**
Deep LSTM FS	0.87	0.06	0.02
RBF SVM FS	0.87	0.03	0.07
Random Forest FS	0.87	0.03	0.07
Deep FC FS	0.87	0.01	0.02
Deep LSTM Raw	0.84	0.37	0.33
RBF SVM Raw	0.87	0.02	0.01
Random Forest Raw	0.86	0.30	0.09
GBDT Raw	0.86	0.20	0.03
Deep FC Raw	0.85	0.07	0.05

Table 2: Model performance

Model	f1-score	auc	auc-pr
GBDT FS	**0.35**	**0.83**	**0.58**
Deep LSTM FS	0.13	0.74	0.56
RBF SVM FS	0.13	0.50	0.56
Random Forest FS	0.13	0.54	0.56
Deep FC FS	0.13	0.50	0.56
Deep LSTM Raw	0.21	0.63	0.39
RBF SVM Raw	0.13	0.50	0.36
Random Forest Raw	0.14	0.53	0.25
GBDT Raw	0.13	0.51	0.18
Deep FC Raw	0.13	0.49	0.06

Table 3: Improvement with features selection

Model	accuracy	precision	recall
GBDT	0.02	0.35	0.52
Deep LSTM	0.03	-0.31	-0.31
RBF SVM	-	0.01	0.06
Random Forest	0.02	-0.27	-0.02
Deep FC	0.02	-0.06	-0.03

Table 4: Improvement with features selection

Model	f1-score	auc	auc-pr
GBDT	0.23	0.32	0.41
Deep LSTM	-0.08	0.11	0.17
RBF SVM	-	-	0.20
Random Forest	-0.02	0.01	0.31
Deep FC	-	0.01	0.50

more classes are very rare, many models don't work too well at identifying the minority classes. In our case, we have very imbalanced class as the crisis regime only occurs 5 percents of the time. Hence the ratio between the normal regime and the crisis regime occurrence is 20! This is a highly imbalanced supervised learning binary classification and can not be done using standard accuracy metric. To avoid this drawback, first, we use the ROC AUC as a loss metric. The ROC AUC metrics is a good balance between precision and recall and hence accounts well for imbalanced data sets. We also weight more the crisis regime occurrence by playing with the *scale_pos_weight* parameter in LightGBM and set it to 20 which is the ratio between the class labeled 0 and the class labeled 1.

Application to the Covid crisis

We provide in figure 4 the out of sample probabilities in connection with the evolution of the price of the S&P 500 merged back adjusted rolled future. In order to smooth the probability, we compute its mean over a rolling window of one week. We see that the probability spikes in end of February indicating a regime of crisis that is progressively turn down to normal regime in mid to end of March. Again in June, we see a spike in our crisis probability indicating a deterioration of market conditions.

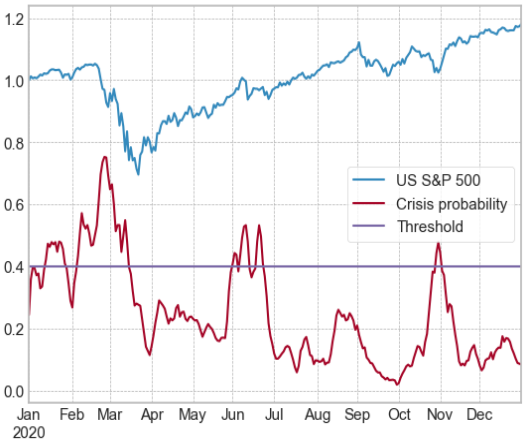

Figure 4: Mean over a rolling window of 5 observation of the probabilities of crash

Can it act as an early indicator of future crisis?

Although the subject of this paper is to examine if a crisis model is effective or not, we can do a simple test to check if the planning model can be an early indicator of future crisis. Hence, we perform a simple strategy consisting in deleveraging as soon as we reach a level of 40 % for the crisis probability. The objective here is by no means to provide an investment strategy as this is beyond the scope of this work and would require some other machine learning techniques likes the ones around deep reinforcement learning to use this early indicator signal as presented in (Benhamou et al. 2020a) (Benhamou et al. 2021a) or (Benhamou et al. 2020c).

The goal of this simple strategy that deleverages as soon as we reach the 40% threshold for the crisis probability is to validate that this crisis probability is an early indicator of future crisis. To be very realistic, we apply a 5 bps transaction cost in this strategy. We see that this simple method provides a powerful way to identify crisis and to deleverage accordingly as shown by the figure 5.

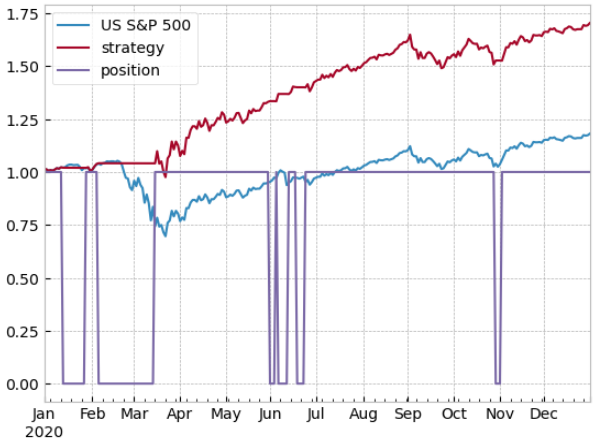

Figure 5: Simple strategy

Conclusion

In conclusion, in this work, we see that GBDT methods can provide a machine learning answer to the planning problem of determining in which regime a market is. Using a simple approach of two modes, GBDT is able to learn from past data and classify financial markets in normal and crisis regimes. When applied to the S&P 500, the method gives high AUC score providing some evidence that the machine is able to learn from previous crisis. We also report that GBDT report improved accuracy over other ML methods, as the problem is a highly imbalance classification problem with a limited number of observation.

References

Aguilar-Rivera, R.; Valenzuela-Rendón, M.; and Rodríguez-Ortiz, J. 2015. Genetic algorithms and Darwinian approaches in financial applications: A survey. *Expert Systems with Applications* 42(21): 7684–7697. ISSN 0957-4174.

Atsalakis, G. S.; and Valavanis, K. P. 2009. Surveying stock market forecasting techniques – Part II: Soft computing methods. *Expert Systems with Applications* 36(3, Part 2): 5932–5941.

Ballings, M.; den Poel, D. V.; Hespeels, N.; and Gryp, R. 2015. Evaluating multiple classifiers for stock price direction prediction. *Expert Systems with Applications* 42(20): 7046–7056. ISSN 0957-4174. doi:https://doi.org/10.1016/j.eswa.2015.05.013.

Benhamou, E.; Saltiel, D.; Ohana, J.-J.; and Atif, J. 2021a. Detecting and adapting to crisis pattern with context based Deep Reinforcement Learning. In *International Conference on Pattern Recognition (ICPR)*. IEEE Computer Society.

Benhamou, E.; Saltiel, D.; Ohana, J. J.; Atif, J.; and Laraki, R. 2021b. Deep Reinforcement Learning (DRL) for Portfolio Allocation. In Dong, Y.; Ifrim, G.; Mladenić, D.; Saunders, C.; and Van Hoecke, S., eds., *Machine Learning and Knowledge Discovery in Databases. Applied Data Science and Demo Track*, 527–531. Cham: Springer International Publishing.

Benhamou, E.; Saltiel, D.; Ungari, S.; and Abhishek Mukhopadhyay, Jamal Atif, R. L. 2021c. Knowledge discovery with Deep RL for selecting financial hedges. In *AAAI: KDF*. AAAI Press.

Benhamou, E.; Saltiel, D.; Ungari, S.; and Mukhopadhyay, A. 2020a. AAMDRL: Augmented Asset Management with Deep Reinforcement Learning. *arXiv* .

Benhamou, E.; Saltiel, D.; Ungari, S.; and Mukhopadhyay, A. 2020b. Bridging the gap between Markowitz planning and deep reinforcement learning. In *Proceedings of the 30th International Conference on Automated Planning and Scheduling (ICAPS): PRL*. AAAI Press.

Benhamou, E.; Saltiel, D.; Ungari, S.; and Mukhopadhyay, A. 2020c. Time your hedge with Deep Reinforcement Learning. In *Proceedings of the 30th International Conference on Automated Planning and Scheduling (ICAPS): FinPlan*. AAAI Press.

Benhamou, E.; Saltiel, D.; Vérel, S.; and Teytaud, F. 2019. BCMA-ES: A Bayesian approach to CMA-ES. *CoRR* abs/1904.01401.

Breiman, L. 1996. Bagging predictors. *Machine Learning Journal* 24(2): 123–140.

Brown, I.; and Mues, C. 2012. An experimental comparison of classification algorithms for imbalanced credit scoring data sets. *Expert Systems with Applications* 39(3): 3446–3453. ISSN 0957-4174.

Caballero, R.; and Krishnamurthy, A. 2008. Collective Risk Management in a Flight to Quality Episode. *The Journal of Finance* 63(5): 2195–2230.

Chen, T.; and Guestrin, C. 2016. XGBoost: A Scalable Tree Boosting System. *CoRR* abs/1603.02754.

Fama, E. F. 1965. The behavior of stock-market prices. *The journal of Business* 38(1): 34–105.

Gu, S.; Kelly, B.; and Xiu, D. 2020. Empirical Asset Pricing via Machine Learning. *The Review of Financial Studies* 33(5): 2223–2273.

Kahneman, D. 2011. *Thinking Fast and Slow*. New York: Farrar, Straus and Giroux.

Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; and Liu, T.-Y. 2017. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30, 3146–3154. Curran Associates, Inc.

Kent, D.; and Moskowitz, T. J. 2015. Momentum Crashes. *Journal of Financial Economics* .

Kingma, D.; and Ba, J. 2014. Adam: A Method for Stochastic Optimization.

Krauss, C.; Do, X. A.; and Huck, N. 2017. Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. *European Journal of Operational Research* 259(2): 689–702.

Li, Y.; and Ma, W. 2010. Applications of Artificial Neural Networks in Financial Economics: A Survey. In *2010 International Symposium on Computational Intelligence and Design*, volume 1, 211–214.

Marceau, L.; Qiu, L.; Vandewiele, N.; and Charton, E. 2019. A comparison of Deep Learning performances with others machine learning algorithms on credit scoring unbalanced data. *CoRR* abs/1907.12363.

Mehra, R.; and Prescott, E. 1985a. The equity premium: A puzzle. *Journal of Monetary Economics* 2(15): 145–161.

Mehra, R.; and Prescott, E. 1985b. The equity premium: A puzzle. *Journal of Monetary Economics* 15(2): 145–161.

Nti, I. K.; Adekoya, A. F.; and Weyori, B. A. 2019. A systematic review of fundamental and technical analysis of stock market predictions. *Artificial Intelligence Review* 1–51.

Ohana, J.-J.; Ohana, S.; Benhamou, E.; Saltiel, D.; and Guez, B. 2021. Explainable AI Models Applied to the Multi-Agent Environment of Financial Markets. In *AAMAS EXTRAAMAS workshop*.

Prokhorenkova, L.; Gusev, G.; Vorobev, A.; Dorogush, A. V.; and Gulin, A. 2018. CatBoost: unbiased boosting with categorical features. In Bengio, S.; Wallach, H.; Larochelle, H.; Grauman, K.; Cesa-Bianchi, N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 31, 6638–6648. Curran Associates, Inc.

Rodriguez, K.; and Sbuelz. 2006. Momentum and Mean-Reversion in Strategic Asset Allocation. *Management Science* 55.

Rundo, F.; Trenta, F.; di Stallo, A. L.; and Battiato, S. 2019. Machine Learning for Quantitative Finance Applications: A Survey. *Applied Sciences* 9(24): 5574.

Samitas, A.; Kampouris, E.; and Dimitris, K. 2020. *Machine learning as an early warning system to predict financial crisis International Review of Financial Analysis*, volume 71(C). Elsevier.

Sezer, O. B.; Gudelek, M. U.; and Ozbayoglu, A. M. 2019. Financial Time Series Forecasting with Deep Learning: A Systematic Literature Review: 2005-2019. *arXiv preprint arXiv:1911.13288* .

Shah, D.; Isah, H.; and Zulkernine, F. 2019. Stock Market Analysis: A Review and Taxonomy of Prediction Techniques. *International Journal of Financial Studies* 7(2): 26.

Siegel, J. 2007. *Stocks for the Long Run*. Companies;: McGraw-Hill, 4th edition.

Sornette, D. 2003. *Why stock markets crash: critical events in complex financial systems*. Oxford: Princeton University Press.

Taleb, N. 2007. *The Black Swan*. Random House Publishing.

Xing, F. Z.; Cambria, E.; and Welsch, R. E. 2018. Natural language based financial forecasting: a survey. *Artificial Intelligence Review* 50(1): 49–73.

Zhang, R.; Yi, C.; and Chen, Y. 2020. Explainable Machine Learning for Regime-Based Asset Allocation. In *2020 IEEE International Conference on Big Data (Big Data)*, 5480–5485.