UCLouvain

LOUVAIN
School of Management

Louvain School of Management

Forecasting short term European CO_2 returns using high frequency gas data: a machine learning approach

First and foremost, I would like to thank my supervisor: Professor Leonardo Iania, for his precious feedback and his support in the choice of the thesis as well as its realization. In addition, several other people assisted me at various stages of my Master's thesis, and I would like to thank them for their contribution. Laurent Remy, External communication Manager at Fluxys, introduced me to the European gas network and explained me how to navigate across the various data sources related to the gas flows. Panagiotis Varelas, PHD student in the field of Machine Learning and Energy systems at the Université Libre de Bruxelles, recommended some best practices in the field of Machine Learning, which were of great help for the neophyte that I was. Moreover, I had the opportunity to discuss with Professor Nathan Lassance, professor of Big Data for finance at the Louvain School of Management, about how to implement some of the course materials. Last, but not least, I am grateful to my parents for their support throughout the year. On one hand, my mother, via long discussions and proof-reading, enabled me to synthesize my thoughts and reorganize them in a compelling way. On the other hand, my father for his recommendations with regards to the transformation of energy units and for connecting me with experts in their field.

This Master's thesis was a challenge for me, and would have not been possible without their input. Thank you very much for everything.

Abstract : In this Master thesis, machine learning algorithms are exploited to forecast weekly returns of the European Trading System during phases III and IV, and study the impact of physical gas flows on the forecasts. Additionally, a popular dimension reduction technique called Principal Component Analysis is used together with the Marchenko-Pastur theorem. The findings are twofold: 1) information about the gas flows do not improve the accuracy of the forecasts during the phase III, but tend to improve them during the phase IV, 2) selection of the relevant set of Principle Components via the Marchenko-Pastur theorem does not outperform arbitrary thresholds of variance explained (95-99%) when the hyperparameters of the machine learning algorithms are finely tuned.

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
Louvain School of Management

Place des Doyens, 1 bte L2.01.01, 1348 Louvain-la-Neuve
Boulevard Emile Devreux 6, 6000 Charleroi, Belgique
Chaussée de Binche 151, 7000 Mons, Belgique

www.uclouvain.be/lsm

1 Introduction

Human activity, powered by fossil fuel, has emitted large amounts of greenhouse gases in the atmosphere since the industrial revolution. Greenhouse gases are responsible for the global warming of the earth to such an extent that nations decided in 2015 to curb emissions to preserve the global warming below +1.5° Celsius compared to the pre-industrial period, during the Paris Agreement (United Nations, n.d.).

Well before that, in 2005, to give clear direction and incentives to industries in Europe, the European Union (EU) decided to introduce the EU Emissions Trading System (EU ETS). The system covers around 40% of the union's emissions and is an important tool to achieve EU's greenhouse gas emissions reduction objectives as it gives a price signal to industries. The scheme currently works as a cap-and-trade system, reducing the cap every year and offering the opportunity to trade emissions quotas freely.

After a pilot phase (2005-2007) that showed severe dysfunctions at the time, the EU ETS followed an iterative evolution throughout the remaining phases. Currently, the ETS has entered its fourth phase (phase IV) spanning from 2021 to 2030. The phase IV is characterized by a linear annual decrease of 2.2% of the emission quotas. In addition, the EU created the Market Stability Reserve (MSR), which began to operate in 2019, to better balance the supply and demand of emission allowances. However, large economic downturns, such as the great financial crisis or the covid crisis, decreased the demand for emissions allowances, while the supply of the allowance remained the same. It created large surplus of emission allowances in the scheme, undermining short-term improvement.

As the carbon price has become the main signal for industries covered by the EU ETS, its prediction has been the subject of numerous research works. The prediction of the price is of great importance for companies willing to hedge themselves against the price volatility of carbon allowances. In addition, Tan et al. (2020) highlighted the opportunities of diversification for portfolio managers, given that the carbon market and non-energy financial markets are connected.

In this master thesis the carbon price is forecasted with the help of machine learning algorithms and alternative physical data in the form of the physical European gas flows. We test whether the addition of physical data improve the forecasting accuracy of the carbon price compared to models using solely financial data.

The forecasting of the carbon price is a very complex exercise, and we faced many challenges. The first challenge was the complexity of the methods used to forecast the carbon prices. Having no prior knowledge of forecasting methods or machine learning algorithms, understanding the technicalities behind them and actually produce a forecast was not easy. Second, the data preparation was heavier than thought. It is often said that 90% of the work is data preparation and 10% of the work is actually producing and analysing the results, we can only confirm it. Finding the right data, making the right and proper transformation after a thorough understanding of the data were definitely the heaviest part of the master thesis. Finally, we also faced the well-known "black box" challenge. Performant algorithms or forecasting frameworks is one thing, interpretable ones is yet another.

After the introduction to our work, the remaining sections are the following: Section 2 provides a thorough review of the literature about the carbon market and the forecasting methods used, Section 3 presents the data as well as the methodology, Section 4 presents and discusses the results, finally Section 5 highlights the limits of this study, and Section 6 concludes it.

2 Literature Review

The EU ETS consists of a cap-and-trade system spanning on 4 phases. The first phase started in 2005 and consisted in a pilot project. We are currently in the 4th phase with record-high prices, but it has not always been the case and it is important to understand the iterative construction of the EU ETS. The literature review explains the journey of the EU ETS, then explains the different drivers and explanatory variables of the carbon prices. To finish, a review of the different methods of forecasting used to predict the prices of the carbon is presented. We then expose the goal of the master thesis and our contribution to the literature.

2.1 Review of the Evolution of the EU ETS

The first phase (2005-2007) consisted of a pilot project. The amount of carbon allowances was defined by the member states under the National Allocation Plan (NAP). The allowances were computed thanks to historical emissions, which were very imprecise (Sato et al., 2022). Most of the allowances were given for free, through a process called "grandfathering". Because of the lack of data, allocations proved to be too high for the need of the market participants creating a huge surplus of allowances. That important surplus combined with the interdiction of "banking" the allowances (i.e., to keep the allowances for later) lead to a sharp drop of the price (Sato et al., 2022). The first phase ended with important lessons that shaped the second phase.

The second phase (2008-2012), strong of the data collected during the pilot phase, sharpened the allocation of free allowances, and the number of allowances auctioned increased from 5% to 10% (Wettestad, 2014). However, despite a willingness form the EC to support the carbon price, the Great Financial Crisis (GFC) and ambitious energy policies in several state members drove the price down again (Sato et al., 2022). During the first two phases, the member states had much flexibility, and the EC only exercised a watchdog role in the EU ETS. The EC decided to take more actions in the following phases (Wettestad, 2014).

The third phase (2013-2020) was significantly different from the previous phases. Several changes had been made to make the carbon market more efficient. First, the EC resolved the problems of windfall profits of the power producers (see Sijm et al., 2006) by ordering the auctions of all the allowances for the power sector in 2013. Other sectors will progressively follow the system of auctions until 2027, year in which all allowances will be auctioned. Certain sectors still benefit currently from the "grandfathering" process because of two reasons: i) They are highly carbon intensive; ii) certain industries might become uncompetitive in case the carbon price is too high and could decide to relocate their activities outside Europe. Indeed, it is much easier to delocalise the production of goods that can be transporter by boats than electricity, which must be consumed locally. The EC decided to give some time to the industries to adapt (Sato et al., 2022). Second, a Linear Reduction Factor (LRF) of 1.74% of the allowances supplied was introduced to fasten the transition (Ellerman et al., 2016). Third, the Market Stability Reserve (MSR) was introduced to regulate the supply of allowances in case of a major demand shocks such as the Great Financial and covid-19 crises (Sato et al., 2022). In a nutshell, the MSR reduces the supply of allowances auctioned if the number of allowances in circulations exceeds a certain threshold (see Bruninx et al., 2020 for an extensive review of the MSR and its impacts). Second to last, new actors such as the aviation in 2012, or the petrochemical, aluminium, and ammonia industries in 2013 entered in the scope of the EU ETS (Aatola et al., 2013). Finally, the length of the phase changed as well. Political uncertainties have weighted on the price of carbon allowances during the transition periods between the phases. As a result, the EC decided to extend the length of the period to reduce the uncertainty and drive long-term investment (Sato et al., 2022).

The fourth phase (2021-2030) started with stronger decarbonizing ambition. The annual linear decrease in the amount of carbon allowances increased from 1.74 % to 2.2 %. In addition, the MSR is fully implemented, and the number of allowances to be withheld are going to increase according to Sato et al. (2022).

2.2 Drivers of the EU ETS

Even though it is commonly known that carbon emissions are caused by energy use and economic activity, which have the effect of raising the demand for carbon allowances, there are still a lot of unanswered questions regarding their actual impact on the carbon market. On that matter, the principal barriers to clarity are the iterative construction and the profound changes that the EU ETS has undergone. In this chapter, we review several studies to understand the drivers on the carbon market, their impacts and the obstacles that prevent to reach an absolute consensus.

Very early in the literature, the role of the energy was identified as a major driver. For instance, Chevallier (2009) highlighted the importance of the power producers in driving the price of carbon through a *fuel switching* effect. The *fuel switching* effect consists of the opportunity to switch between coal, a carbon-intensive source, and gas for power producers to meet their emission reduction targets (see also Benze and Trück, 2009). A few years later, Aatola et al. (2013) found that the oil and gas markets had a positive effect on the carbon price, while the coal had a negative effect during phases I and II. In this context, positive means an increase in price and a negative impact means a decrease in the price of the carbon allowances. These findings confirmed the study of Bredin and Muckley (2011).

Moreover, Chevallier (2011a) brought a nuance around the impact of the energy sector on the carbon price. He noted that it depends on the state of the economy and found that oil and coal shocks had a positive impact during a booming economy but had a negative impact during a bursting economy. To the best of our knowledge, this study is the first to show potential asymmetries in the impacts of the energy markets.

Hammoudeh et al. (2014b) also introduced a nuance of the effect of the energy markets on the carbon price. They studied the short-term and long-term effects of several energy variables during the first two phases and the beginning of phase III. The authors noted that an oil shock had a short-term positive effect on the carbon price, which then tends to have a negative effect over the longer-term. They also found an overall negative impact of the gas price on the carbon prices, which is inconsistent with some previous studies (see Bredin & Muckley, 2011; Aatola et al., 2013). The general finding is that higher energy prices lead to less economic activity, which then reduces the demand for carbon allowances (Hammoudeh et al., 2014b).

Hammoudeh et al. (2014a) studied the impact of the coal, gas, oil, and electricity prices at different quantiles of the distribution of the carbon prices. Their findings are the following: the oil market negatively affects the carbon market at the top quantile of the distribution (i.e., when the carbon price is high). In addition, the authors found that the gas had a positive impact on the carbon market when the latter is low, but a positive one when the latter is high. With regards to coal shocks, it always exercises a negative impact on the carbon price. In general, they stress that the energy markets have nonlinear effects on the carbon price.

To better understand the exact connexions between the energy markets and the carbon market, X. P. Tan and Wang (2017) proposed a new economic model, which links energy consumption, economic activity and the demand for carbon allowances. They introduced the *aggregated demand effect,* the *production restraint effect* on top of the *substitution effect* already identified. The *aggregated demand effect* relies on the link between energy, carbon emissions and economic activity. During an economic expansion, more energy is consumed, which increases the demand for carbon allowances. The *production restraint effect* is active when the *substitution effect* becomes less important at very high carbon prices. With higher marginal costs, the industrial plants cannot switch fuels, and prefer to restrain their production. Accordingly, the demand for carbon allowances is lower. The authors report that the energy markets do not affect homogeneously the carbon market across the phases. For instance, coal and gas impact the carbon price via the *production restraint* effect during the first phase, while the *aggregated demand* seems to drive the second phase during the economic recovery post GFC. Thanks to new transmission channels, the authors clarify the time-varying impacts of the energy markets on the carbon price.

Duan et al. (2021) used the very same transmission channels to explain that energy prices tend to have stronger impacts when the carbon prices are low and vice versa. The authors justified the use of additional transmission channels because the *switching fuel* or *substitution* effect suffers some serious shortcomings. It fails to consider the effect of the oil on the carbon prices. Chevallier et al. (2019) even questioned this mechanism as they found that high carbon prices alone do not provide enough incentives for power producers to switch from coal to natural gas.

In addition to the dynamic state of the economic situation that results in different effects of the energy markets on the carbon prices, the energy market, and more specifically the power markets have been experiencing dramatic changes across the phases of the EU ETS. As a reminder, the electricity prices in Europe are formed via the *merit order* principle (i.e., the technology with the lowest marginal cost delivers first while the most expensive technology

that satisfies the remaining demand fixes the prices). Coal used to have a large share in the generation mix. However, the gas seemed to have recently become the dominant peak technology (i.e., flexible technology with high variable cost that bridges the gap left by the base load technologies such as the nuclear and the renewable). At the European level, gas-powered units outproduced their coal-powered units in the middle of the 2018 year. Coal electricity production declined from 850 TWh in 2017 to 550 TWh in 2020 in Europe (IEA, 2020).

The link between the power and carbon markets can be symbolized by two metrics: the clean spark spread and the dark spread. The clean spark spread refers to the profit margin acquired by a gas-fired power plant, accounting for both the cost of natural gas and emissions allowances. On the other hand, the dark spread pertains to the same concept, but for a coal-fired power plant. These concepts are inherently related to the *substitution effect*, and the literature found that some of the energy drivers are not robust to the introduction of variables linked to the power market. For example, Hammoudeh et al. (2014b) found that coal shocks have no effect on the carbon price if the electricity price is incorporated in the model. Conversely, Batten et al. (2021) exposed that it is, instead, the price of the natural gas that has no impact on the carbon prices when the electricity price and the clean spark spread are included. The reason behind the contradiction between the two studies might be the current European phase-out of coal is slowly replaced by gas.

Duan et al. (2021) found that energy markets have an overall negative impact on the carbon market (see also Hammoudeh et al., 2014b) with an asymmetric intensity across the carbon price distribution. The reasoning is the following: an increase in energy prices leads the industrials to reduce their output given an increase in their marginal costs. Therefore, energy prices have a negative effect on the carbon price. However, during a booming economy, the demand for the output of industrials is high, which incentivize the industrials to keep producing. As a result, energy price shocks moderately impact the carbon market during a booming economy characterized by a high carbon price. However, during a bursting economy the demand for industrial output is severely tightened, which removes the incentive to keep producing if the costs increase.

The previous findings of Duan et al. (2021) confirm the study of Zhu et al. (2019). They performed a multiscale analysis of the drivers of the carbon markets to have a more granular view on the long-term and short-term drivers of the carbon market. They found that the oil and natural gas markets negatively impact the carbon price in the long-term. The reasoning is the

following: higher energy costs disfavour the long-term economic development, which in return affects negatively the demand for carbon allowances while the opposite is true. The finding is consistent with the study of Batten et al. (2021). The reasoning is confirmed by the positive long-term effect of the stock market on the carbon price. The stock market being a good proxy of the long-term economic development. Lovcha et al. (2022) confirmed the long-term effects of economical, oil and gas variables, but found a positive impact of the oil market, which contradicts the previous results (Zhu et al., 2019).

Despite the established link between the energy, power and carbon markets, the main drivers identified only explain a variable, and often small, part of the volatility throughout the phases. Therefore, other drivers must be identified. Weather variables seem to have no predictive power on the carbon prices, except for the unanticipated changes in temperature (Batten et al., 2021). This seems counterintuitive given the more and more important share of the renewables in the electricity mix in Europe (IEA, 2021). But it seems that the energy markets reflect already all the information coming from weather variables but fail logically to incorporate unforeseen weather events. Given that the carbon price volatility will increase due to bigger temperature volatility caused by climate change, one can expect an increasing cost of options-based hedging strategies (Batten et al., 2021).

For the moment, we have reviewed essentially demand-side drivers. Most of the literature has focussed on understanding the drivers of the demand-side for the very simple reason that the supply is locked by policies defined by the member states in a first time, and by the EC in a second time form phase III onwards (Koop and Tole, 2013). However, another reason is that supply-based can be summarized as policy uncertainty, which are very much abstract and difficult to quantify (Batten et al. 2021)

Nevertheless, the supply has a huge impact on the prices. Bruninx et al. (2020) studied the impact of the implementation of the MSR and the accelerated linear decrease of allowances implemented in 2020. They found that the combined effect could decrease by 41% the cumulative emissions compared to the unchanged policies.

To sum up, the drivers of the demand for the carbon allowances have changed over the phases for three reasons: the economic situation (boom or burst) was heterogeneous during the phases, the power market is more and more driven by renewables and gas-powered unit at the expense

of coal-powered units, and finally for the very simple reason that the EU ETS has evolved in scope and rules.

In addition, the energy markets have asymmetrical effect on the carbon markets with regards to the intensity (strong or moderate) and the sign (positive or negative) based on the quantile of the distribution of the carbon price.

Finally, all these complex relationships have represented only a small and variable portion of the variability of the carbon prices, in which a large part might be hard-to-quantify speculative behaviours and political uncertainties. We expect the exercise of forecasting the carbon price to be, to say the least, tricky and complex. The next section investigates the methods used to overcome some of the challenges.

2.3 Evolution of the forecasting methods for the carbon prices

After having defined the EU ETS and understood the driving variables, we can investigate the different techniques used to forecast the carbon prices. In the beginning of the EU ETS, several research groups applied multiple linear models such as the Autoregressive Moving average (ARMA) or the Generalized Autoregressive Conditional Heteroskedasticity (GARCH). For example, García-Martos et al. (2013) used a multivariate extension of the ARMA method called Vector ARMA (VARMA) to predict the CO2 prices as well as other commodity prices. Byun and Cho (2013) proposed a GARCH model to forecast the volatility of day-ahead carbon price futures based on the volatility of several commodities.

As highlighted in the previous section, carbon prices are influenced by several drivers, at different time scales and with asymmetric effects based on the distribution of carbon prices. It results in a very noisy market with non-linear patterns (see for example Benz & Trück, 2009 or Duan et al., 2021 among others).

Consequently, the true signals of the time-series might be mixed, which prevents to achieve accurate predictions due to the different timescales of the factors. To deal with that problem, several pre-processing methods have been introduced to increase the forecasting accuracy. Zhu (2012) decomposed the carbon price time-series into Intrinsic Modes Functions (IMFs), using the Empirical Mode Decomposition (EMD). An IMF fulfils two conditions: First, the discrepancy between the count of extremum points and zero crossings should not exceed one. Second, the arithmetic mean of the upper and lower bounds must be precisely zero, as outlined by the work of Huang et al. (1998).

This new method is particularly useful for non-stationary and non-linear time-series as it uses no *a priori* assumptions (Rehman and Mandic, 2010). Thanks to encouraging results, several other works used improved versions of the EMD. One of the main shortcomings of the EMD is mode mixing (i.e., the IMFs can contain more than one oscillatory signal or similar signal might be in different IMFs) according to Zhu et al. (2016). The Ensemble Empirical Mode Decomposition (EEMD), or the Complete ensemble empirical mode decomposition (CEEMD) with adaptive noise (CEEMDAN) are non-exhaustive examples of methods that aim at solutioning this problem (See for example Zhu et al., 2016; Zhang et al., 2018; Wang et al., 2021).

Classical statistical methods relying on linear models are not suitable to forecast the carbon markets because of the inherent non-linear patterns (Huang et al., 2021). Therefore, Chevallier (2011) proposed to use a non-parametric model to forecast the carbon prices during phase I. The non-parametric model outperformed several other linear models.

Several researchers used machine learning models, which can model non-linear patterns, instead of econometric models to improve the accuracy of the forecast. Zhu (2012) used an Artificial Neural Network (ANN) to forecast carbon prices during phase II which outperformed linear models. Jaramillo-Morán, Fernández-Martínez, García, et al. (2021) compared three techniques being Multilayer Perceptron (MLP), Long Short-Term Memory (LSTM) and Extreme Gradient Boosting (XGBOOST). They reported that the best performing algorithm is the MLP combined with EMD. They concluded that MLP provides more robust forecasts in short and mid-term horizons compared to LSTM despite of the ability of the latter to capture long-term dependencies thanks to its "memory" cells. Zhu et al. (2021) trained a Least Squares Support Vector Machine (LSSVM) to forecast spot and future prices of carbon. LLSVM is a version of Support Vector Machine (SVM), which was first proposed by Suykens and Vandewalle (1999), and which consists of solving linear equations instead of a quadratic programming problem. Zhu et al. (2021) found that the model had a high accuracy performance overall, but also a very good directional accuracy.

Extreme Machine Learning (ELM) combined with several decomposition methods has also been used to forecast carbon prices in China and Europe. The results showed that the ELM provides great accuracy performances (Sun & Zhang, 2018). ELM is inspired by the human brain, and biological learning. A human brain does not change its "structure" based on the task

performed. In other words, there is no human interference in the choice of the brain structure. Yet, human brain performs various tasks (i.e., feature selection, regression, classification, etc.) and with almost zero learning time with regards to specific samples. Inspired by these features, ELM was designed to have only one parameter to manually tune (the number of hidden nodes) and an accelerated learning process (ELM only learns the weights between the hidden layer and the output layer during the training period). ELM is characterized by randomly assigned biases and non-linear functions. The pre-defined "structure" makes it almost problem-and-data agnostic, as the human brain (G. Huang, 2014).

With the use of machine learning, powerful optimizations algorithms are needed to "tune" the parameters of the models. This step is called hyperparameter tuning. In the literature of carbon forecasting, one class of algorithm is primarily used: Meta-heuristic optimization algorithms. Those algorithms are popular because they can solve the local optimum problem and be applied to a wide range of problems among other advantages (Mirjalili and Lewis, 2016).

This class of algorithms provide a general framework to solve problems and is inspired by the nature-based behaviours or phenomena. By mimicking natural behaviours, researchers expect the algorithms to find optimal solutions as one can find in the natural world. (Kamrani and Gonzalez, 2003).

For instance, Zhu (2012) optimized the parameters of its ANN with a Genetic Algorithm (GA). The idea behind GAs is to reproduce the behaviour of natural selection. The algorithm starts with an ensemble of random solutions that evolve towards more optimal solutions based on the Darwinian natural selection process (See Kamrani & Gonzalez, 2003 for an extensive explanation). Moreover, Zhu and Wei (2013) use the Particle Swarm Optimization (PSO) to tune the parameters of a Least Squares Support Vector Machine (LLSVM). Also, Sun and Zhang (2018) used the Whale Optimization Algorithm (WOA) to optimize the parameters of a series of machine learning models. The WOA is inspired by the hunting strategies of whales (Mirjalili & Lewis, 2016).

One can then combine the different techniques mentioned above to create advanced forecasting frameworks. Zhang et al. (2022) proposed a dynamic self-learning N-A MEMD-COSWOA-ELM. They create a rolling-window and self-learning ELM, because of its fast-learning and generalization superiority, optimized by an extension of the WOA after having decomposed the data into several IMFs with a noise-assisted multivariate version of the EMD. The rolling-

window and self-learning feature alleviate two commonly encountered problems: overfitting and the dynamic character of the factors influencing the carbon markets (see for example batten et al., 2021). The model proposed outperformed by at least 50% all other approaches.

Furthermore, machine learning techniques outperform their econometric peers on a standalone basis because of the high non-stationarity and non-linearity of the carbon price. But they might not deliver superior performances in all cases. For instance, GARCH gives very satisfactory results when forecasting short-term volatility (see for example Byun & Cho, 2013). In addition, machine learning techniques are sometimes considered as a "black-box".

Researchers have therefore created hybrid forecasting frameworks to improve the accuracy of the predictions. Hybrid frameworks are interesting because they take advantage of several models to outperform standalone techniques. For example, Zhang et al. (2018) proposed a hybrid forecasting framework using the CEEMD to transform the time-series into IMFs, which are then reassembled into three components: long-term, periodic, and random. The authors then used different forecasting models based on their characteristics to forecast independently each of the components. For instance, a GARCH model is used to forecast the random component, while a certain type of Neural network, optimized by a meta-heuristic algorithm called Ant Colony, forecasts the periodic component. Each model is fed with specific variables identified in the literature. Eventually, all the forecasted components are reassembled. Their results proved much better than other tested frameworks.

Another example is proposed by Huang et al. (2021) who used a Variational Mode Decomposition (VMD) to dissect carbon prices into different modes, sort them into high-frequency and low-frequency. Then, they used a GARCH model to forecast the high-frequency components, and a LSTM to forecast the low-frequency components. By doing so, they combined the short-term power of a GARCH while enjoying the ability of the LSTM to "remember" long-term dependencies. Eventually, they feed a LSTM with the forecasting of the high and low frequency components to achieve the final forecasting. Their model is named VMD-GARCH/LSTM-LSTM and showed significantly better results that machine learning or econometric techniques used alone.

2.4 Contribution to the literature

The review of the literature highlights the difficulty to forecast carbon prices, due to noisy and non-stationary time-series, non-linear and time-varying patterns of the carbon prices, the lack of historical and relevant data because of unpredictable regulatory changes among other

challenges. To overcome part of the problems, research have focused on ever more advanced and complex forecasting frameworks. However, very few research works have investigated the potential of new predictive variables despite the largely unexplained variability of the carbon prices.

This master thesis aims at contributing to the literature in two ways. First, we bring additional understanding of the forecasting of carbon prices in the fourth phase of the EU ETS, as few researches have been made so far. With record-high prices and a huge volatility, the phase IV is singularly different from the previous ones. Second, and conversely to the focus from the literature, the master thesis will investigate further the relationship between carbon prices and physical flows of the commodities in the form of daily European gas data. As the current energy market has experienced intense disruptions due to recent events. Price can be easily manipulated by government actions as the recent discussions at the European level around a cap on the prices of the gas prove (Financial Times, 2022). We therefore assume that the prices do not always provide the right signal with regards to the carbon emissions, but that physical flows in the real world do, which should eventually drive the price of the carbon prices.

We aim at linking financial and physical world for two reasons. First, the variability of the carbon prices explained by the current drivers is relatively low in some cases, which pledges for the discovery of new drivers. Second, and more abstract, we would like to emphasize that behind financial prices, there exists an underlying physical reality. We believe that researchers have everything to win by adopting techniques and embracing the knowledge of several disciplines. The example of the use of signal processing methods developed for physical applications or the example of the transposition of biological phenomena to computer-based optimization program have encouraged us to do so.

3 Data & Methodology

This section describes briefly the methodology used to forecast the returns of the European Union Allowances (EUA).The methodology is composed of four main steps: the data preparation, the dimension reduction to obtain the predictive variables, the models training and the forecasts and finally the evaluation of the forecasts. The different steps are illustrated in Figure 1.

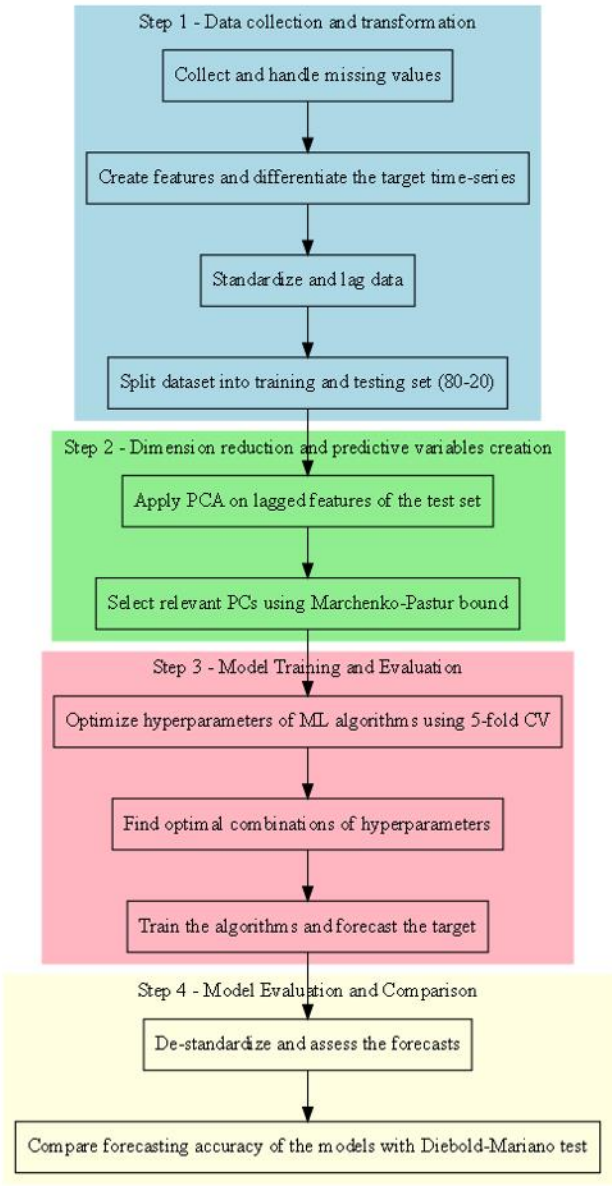

Figure 1: The methodology used to forecast the carbon market consists of 4 successive phases

In a first step, the data are collected from Bloomberg and from the Gas Infrastructure Europe (GIE) website. The missing values are replaced by the average of the days just before and after. The data are then transformed to create meaningful features and make the target time-series stationary. After the transformation, the data are standardized, lagged, and split into a training and testing set according to an 80-20 division.

Second, the dimensions of the new data set are reduced thanks to a Principal Component Analysis (PCA). We use the Marchenko-Pastur theorem to determine the relevant set of Principle Components (PCs) to keep in order to predict the dependent variables.

In the third step , a 5-fold cross validation respecting the time-order of the time-series is applied to optimize the hyperparameters of three machine learning models and find the optimal

combinations that should yield the best out-of-sample performance. Then, the machine learning algorithms are trained with the optimal set of hyperparameters and forecast the carbon market.

Finally, the de-standardized predictions are assessed with the help of several metrics commonly used in the literature. And a Diebold-Mariano test enables to compare the forecasting accuracies of the different models, with and without the data to test our hypothesis.

The following section presents with more details the different steps of the methodology to ensure reproducibility.

3.1 Data collection and transformation

In this section, the first step of the methodology is explained with an emphasis on the transformations applied on the data, and the reasoning behind them.

3.1.1 Univariate analysis

The dependent variable is the "EUA Daily Futures Index" (EUR/ton), which tracks the price of the daily future contracts of carbon allowances traded on the Intercontinental Exchange (ICE). The data are retrieved from Bloomberg (ICEDEU3). This choice is motivated by the increased stability of the future price compared to the spot price of the carbon (Dual et al., 2021). The evolution of the dependent variable is shown in Figure 2.

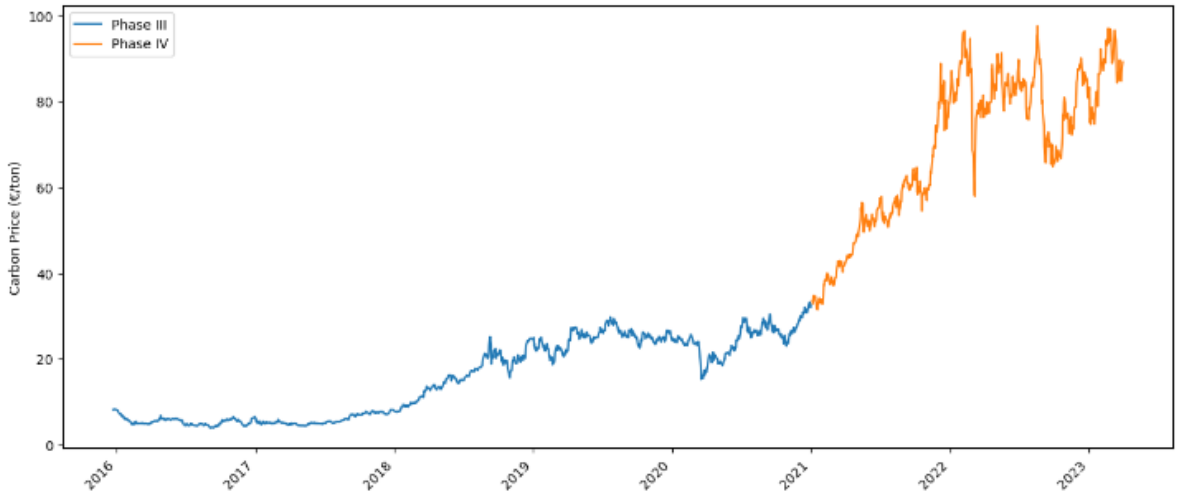

Figure 2: The carbon price (€/ton) time-series displays a pronounced upward trend between 2015 and 2023.

The data spans from 25-12-2015 to 31-03-2023, amounting to 1873 data points on two different phases (III & IV) of the EU ETS with the 01-01-2021 symbolizing the transition from one phase to another. As one can see in Figure 2, the time-series seems to display an upward trend with regards to its mean. As a result, the time-series might not be stationary. As a reminder, a stationary time-series is characterized by a constant mean and volatility over time.

To check whether a time-series is stationary, Dickey & Fuller (1979) proposed a test whose null hypothesis is the presence of a unit root in the time-series. Without going into details, a unit-root signifies that the time-series follows a random-walk process and does not revert to its existing mean, implying that the time-series is not stationary. If the null hypothesis is rejected, it means that the time-series is stationary. The p-value of the Augmented Dickey-Fuller (ADF) test for a lag of 24 day is equal to 0.99, larger than the commonly used threshold of 5%. As a result, we cannot reject the null hypothesis. Theoretically, it does not strictly validate that there exists a unit root, and that the time-series is non-stationary. However, practically speaking, the graphical evidence is sufficient in our case and the test only provides supportive evidence that does not infirm our intuition. All the information with regards to the moments of the distribution and the values of the test performed can be found in Table 1.

To solve the issue of non-stationarity, we must differentiate the time-series. Consequently, the time-series is transformed in "log returns" to achieve stationarity. Thanks to this log-transformation, we also hope to tend towards a normal distribution of the returns. The log returns are obtained via the following transformation :

$$Log\ returns = \log\left(\frac{X_t}{X_{t-l}}\right), \quad t \in \mathbb{Z}^+, l \in \mathbb{Z}^+$$

To compute weekly log returns, l is equal to 5, and for "monthly" log returns l is equal to 20, for instance. After the weekly log transformation, the statistical value of the ADF test is equal to -19.95, and the p-value is very close to 0. We can therefore reject the null hypothesis. It confirms that the time-series is stationary for weekly log returns.

In addition to the ADF test, we also check whether the data are normally distributed thanks to the Jarque-Bera test. The test poses as null hypothesis that the third moment, or skewness, and fourth moment, or kurtosis, have the same values than the ones of normal distribution, being 0 and 3 respectively (Jarque & Bera, 1980). Figure 3 shows the new time-series after the log-transformation.

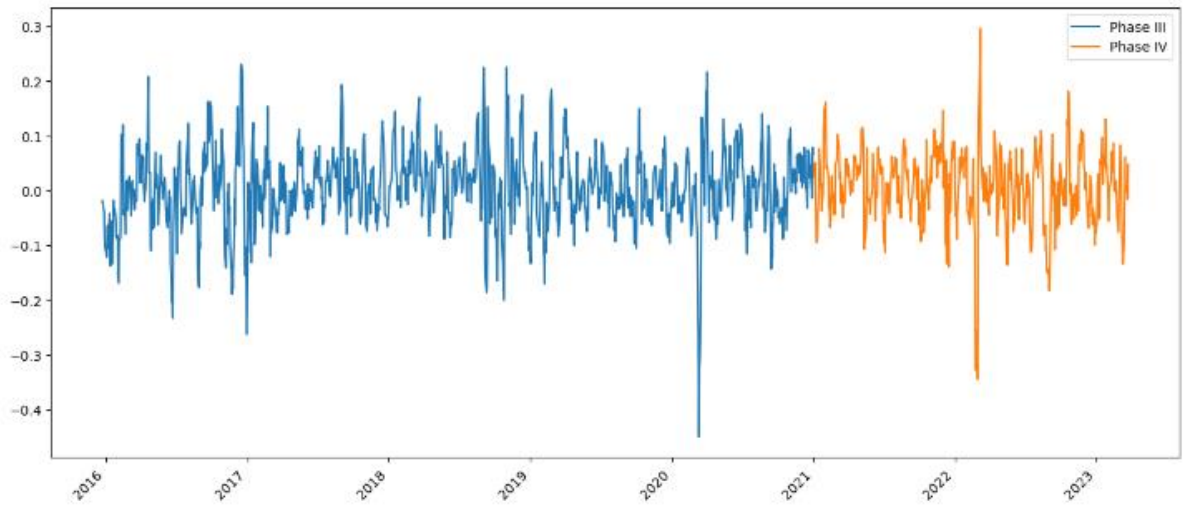

Figure 3: The weekly log returns time-series has no upward trends, and seems stationary.

The p-values are close to 0, which signifies that the data do not have the same skewness and kurtosis as a normal distribution. Figure 4 compares the distribution of the weekly log returns with a normal distribution defined by the very same mean and variance.

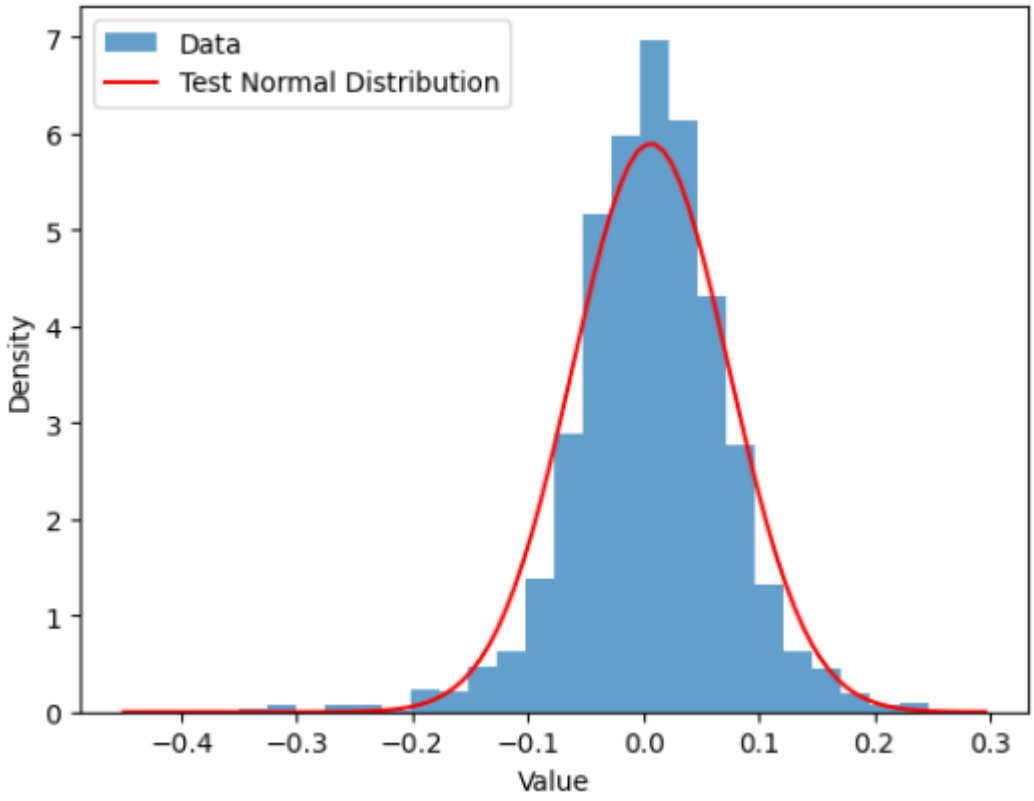

Figure 4: The weekly log returns data displays negative skewness and excess kurtosis.

The distribution of the weekly log returns is characterized by a slight asymmetry and many extreme values or “fat tails”. Table 1 provides the values of the four first moments of the distribution of the data before and after transformation, as well as the results of the statistical tests performed. The values of the skewness and kurtosis confirm the visual interpretation of Figure 4. After the log-transformation, the time-series is effectively stationary, but depart

largely from a normal distribution mainly because of a substantial excess of kurtosis as the p-values indicate.

	Mean	Variance	Skewness	Kurtosis	JB		ADF	
					Test value	P-value	Test value	P-value
Daily Price	32.08	777.02	0.92	-0.54	\	\	0.64	0.99
Weekly Log-Returns	1.20E-03	9.28E-04	-0.40	3.65	1089.48	2.64E-237	-19.95	0.00

Table 1: The four first moments of the distribution of the carbon time-series before and after differentiation with the values of the JB and ADF statistical tests and their p-value

3.1.2 Independent variables

To predict the weekly log returns, several independent variables are used. The independent variables can be grouped in two categories: the financial data and the physical data.

3.1.2.1 Financial data

Based on the review of literature, it seems that the most impactful variables come from the energy markets and the economic activity. Therefore, we used the futures of the Brent, AP12 Rotterdam and TTF and the Euro Stoxx 50 (SX5E) as a proxy of the expectations of the economic situation. The energy markets variables come from Bloomberg and are namely: the Generic 1st 'MO' Future Comdty (EUR/bbl) (MO1), Generic 1st 'MFE' Future Comdty (EUR/ton) (MFE1) and NL TTF Nat Gas Month 1 Index (EUR/MWh) (TTFG1MON). Figure 5 shows the time-series of the different financial variables collected.

Figure 5: Evolution of energy commodities and the EuroStoxx 50. The recent energy crisis caused havocs in the financial and energy markets.

3.1.2.2 Physical data

3.1.2.2.1 Choice of the data

Additionally, physical data are added to the model to alleviate potential weakness in the price signals and to be closer to the physical reality of the economic activity. Physical data represents the daily flow of gas in Europe. The values are retrieved from the website of the Aggregated Gas Storage Inventory (AGSI)[1] for the gas flowing via pipeline and from the website of the Aggregated LNG Storage Inventory (ALSI)[2] for the liquefied natural gas (LNG). Optimally, we should have added the information of the net import of gas via pipeline from non-Eu countries. However, we decided to exclude them because the values are noisy, hard-to-interpret, and subject to errors. Since we could not make sense of it, we preferred to discard them. The loss of information from the discard of the net imports of gas from non-EU countries should

[1] Available at: https://agsi.gie.eu/
[2] Available at: https://alsi.gie.eu/

not be significant. For instance, any deficit or surplus of imports of gas should *in fine* be reflected in the level of storage.

The master thesis focuses solely on the gas analysis because of the granularity, high-frequency, and quality of the data. Unfortunately, information about the European oil and coal consumption exists only on a monthly basis. The exercise of forecasting high-frequency variables with low-frequency data becomes tricky and requires more advanced methods that are out of the scope of this work. For the interested reader, one example is the study of Foroni et al. (2023) who use monthly macro-economic variables with monthly frequency to predict daily electricity prices with an advanced Bayesian version of the Mixed Data Sampling (MIDAS) model. The particularity of MIDAS is its ability to handle the difference in frequencies between the dependent and independent variables (Ghysels et al., 2006).

3.1.2.2.2 Feature extraction

Among the plentiful numbers of data available online, only a subset is interesting for this study. This section describes how a sub-set of relevant variables is transformed to end up with two features that will be used to predict the carbon price.

First, we would like to have a measure of the energy stored under the form of gas. The amount of gas stored in the storage facilities is expressed in TWh, and the amount of gas stored under the form of LNG is expressed in cubic meters. We are interested in the sum of the two measures to have a total amount of energy stored under the form of gas.

To transform cubic meters of LNG in TWh, some assumptions are required. In reality, LNG from various regions has different gross calorific values (e.g. Russian LNG has 55.25 MJ per kg whereas Norwegian LNG has 52.22 MJ per kg). For simplicity, we assume a gross calorific value of 50 000 kJ. We then know that the density of 1 m^3 of LNG is 450 kg (IEA, 2023). We can derive the quantity of energy per cubic meter of LNG by multiplying the mass of the LNG by the energy per kg of gas:

$$1m^3 of\ LNG\ =\ 450\ kg\ \times \frac{50\ 000\ kJ}{3600\ kJ/kWh}\ =\ 6255\ kWh\ =\ 6.255\ MWh$$

Secondly, it is essential to have a measure of the exchange of gas flows between the storage capacity and the pipeline network that eventually end up being consumed by the industry or the households. The entry and exit of gas from the storage facilities as well as the exit of gas from the LNG facilities are given in the initial dataset. Summing them up (with correct signs) gives

a new measure of the net injection of gas in the system in terawatt hour per day (TWh/d), which is the net inflows coming from the reservoirs of gas and the LNG facilities. Note that the difference in storage between two days is not equal to the sum of the gas coming out because of the assumptions with regards to the calorific values of the LNG. Finally, to avoid seasonality effects, the relative value year on year are considered for each day. To do so, the difference between one value and its yearly-lagged (252 days) counterpart is computed. Figure 6 represents the two new physical features created.

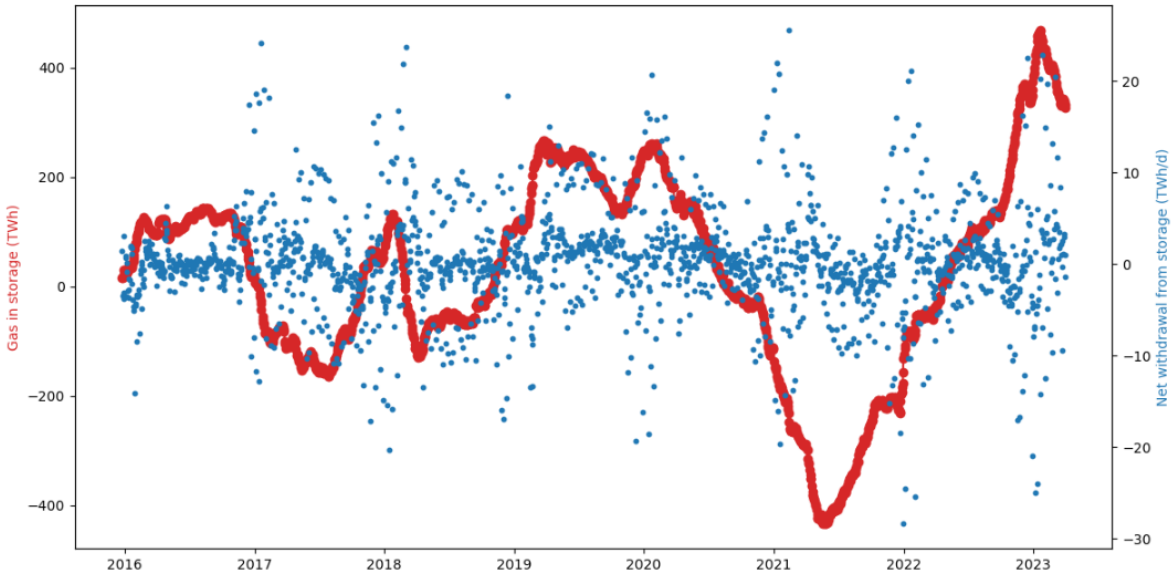

Figure 6: Evolution of the relative measure of Storage and Net Withdrawal of gas over the years.

One can notice that, despite the absence of information about the net imports of gas by pipeline from non-EU countries, the level of storage itself is a good measure of the wider gas flows. For example, one can observe that the levels of storage are superior to the average for the years 2020, for instance. It corresponds to the beginning of covid-19 crisis, and the following reduction in gas consumption. Another example is the very low levels observed in the end of the year 2021, which correspond to the beginning of the energy crisis (The Economist, 2021). As a result, we argue that the data from the supply itself are not needed as the storage features testify for it. In total, the number of features amounts to 7 with the addition of the 2 physical variables.

3.1.2.2.3 Re-indexing the data

The financial markets are open 252 days a year while gas is (normally) flowing 365 days a year. It creates discontinuities and could pose problems when creating the time-series. Deleting the surplus of information is wrong as we know that events happening outside of the market opening hours have logically an impact at the reopening. Therefore, we decided to incorporate

the information from the gas features lying outside of market opening days in the earliest prior market day. For example, the information happening during the weekend is associated with the data of the Friday so that their information is incorporated in the prediction of the value on Monday.

However, we cannot "summarize" the data as we wish. For the sake of consistency, the data are summarized through the followings: the level of gas storage of the latest day replaces the value of the earliest prior market day while the net injections of gas in the system are summed up. As a result, the model has the most up to date stock of gas and the total flows coming in or out of the storage facilities.

3.2 Data preparation

After having collected and transformed the data, constructed the features and our target, we still need to create the actual input that will eventually feed our predictive models. The following section explains the different steps to achieve it.

3.2.1 Standardization of the data

The data are expressed in fundamentally different units and have different scales. As a result, the forecasting model might be more influenced by one feature at the expense of another if their values are not re-scaled. One way to re-scale the data is to standardize with the following formula:

$$\boldsymbol{X}_{cs} = \frac{X_{ij} - \overline{X}_k}{\sigma_k} for\ i \in [1,\ldots,N], for\ j \in [1,\ldots,K]$$

3.2.2 Time-series construction

Zhao et al., (2018) stressed that it is important to have information at different time lags to predict the carbon price, and that the optimal lag might be different for different variables. Therefore a lag of 20 days (around a month of market data) is chosen to construct a new lagged dataset. The new lagged dataset consists of 140 independent variables (i.e., the 7 features with a lag of 20 days) and one dependent variable (i.e., the weekly carbon log return).

It is very unlikely that all the 140 features are essential to forecast the weekly log returns of the carbon market. Therefore, a Principal Component Analysis (PCA) is applied on the features to create a new set of variables that "summarizes" the data while keeping most of the variance of the data.

3.2.3 Dimension reduction via Principal Component Analysis

The PCA is a powerful reduction dimension tool that finds new orthogonal components, called principal components (PCs), that keep most of the variance to project the data on those new dimensions. PCA "summarizes" the variance of a data set on a lower subset of dimensions. The intuition is that the very same dataset could be described with *p* new principal components instead of the *k* actual variables. In Python, the function PCA from the module Scickit-learn is a practical way to apply PCA to the features (Pedregosa et al., 2011).

To find the new PCs, the data must be centred beforehand. In our case, the data are standardized because the variables are expressed in different units. Once the data are standardized, the correlation matrix of the dataset can be computed. Please note that if the data were only centred, the PCA would have been done on the covariance matrix. The correlation matrix $\boldsymbol{R}$ with *K x K* dimensions is as follow:

$$\boldsymbol{R} = \frac{1}{n}\boldsymbol{X}_{CS}\boldsymbol{X}_{CS} = \begin{bmatrix} \sigma_1 & r_{12} & r_{13} & \cdots & r_{1K} \\ r_{21} & \sigma_2 & r_{23} & \cdots & r_{2K} \\ r_{31} & r_{32} & \sigma_3 & \cdots & r_{3K} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{K1} & r_{K2} & r_{K3} & \cdots & \sigma_K \end{bmatrix}$$

With R_{ij} representing the correlation between the *i*th and *j*th standardized columns of the Matrix $\boldsymbol{X_{cs}}$. By solving this equation, the matrix of eigen vectors $\boldsymbol{V}$ of $\boldsymbol{R}$ is found:

$$(\boldsymbol{R} - \lambda \boldsymbol{I}_{\boldsymbol{p}}) \cdot \boldsymbol{v} = 0$$

With λ being the eigen values of $\boldsymbol{R}$. It turns out that the eigen vectors form the orthogonal basis of the new space of dimensions *P* on which the data are projected to form the new PCs. The new PCs are equal to $\boldsymbol{Y} = \boldsymbol{X_{cs}V}$. The first PC has the largest eigen value λ, and represents the largest part of the variance of the dataset.

The PCA brings additional benefit to our predictions. First, it fastens the training time of the different models, which is interesting as our computational power is limited (Rudnik et al., 2022). In addition, it removes potential problems arising from collinearity. However, if one

keeps all the components, the prediction should not be improved as all the noise contained in the time-series will still be in the PCs. Therefore, we propose to select only a subset of relevant components to reduce the noise and improve the predictions.

3.2.4 Selection of the relevant Principal components via Marchenko-Pastur bound

A common, but arbitrary, criteria to decide the number of the first *p* PCs is to set a threshold of variance explained by subset of PCs. In general, one can decide to keep 95% of the variance, for instance. The percentage of variance explained by the *p* first components can be computed as follow:

$$\frac{\sum_{i=1}^{P}\lambda_i}{\sum_{i=1}^{K}\lambda_i} \geq \alpha \; with \; \alpha \in [0,1], P \leq K$$

However, this method is not very appealing as it depends on an arbitrary choice. In our case, with models containing different sets of predictive variables, this method is not satisfactory.

To avoid a "fit-for-all" solution that would most likely not yield an optimal solution, we decided to use the bound of Marchenko-Pastur (1967) to select the relevant subset of PCs. The authors derived a theorem indicating that for a random matrix with *N* observations and *K* variables independent and identically distributed with zero mean and variance σ^2, the eigen values λ of the covariance matrix of the random matrix converge towards a certain density function as *N* and $K \to \infty$ with $1 < N/K < \infty$. The density function of the eigen values λ is the following:

$$f(\lambda) = \begin{cases} \frac{N}{K}\frac{\sqrt{(\lambda_{\max}-\lambda)(\lambda-\lambda_{\min})}}{2\pi\lambda\sigma^2}, & \text{if } \lambda \in [\lambda_{\min}, \lambda_{\max}] \\ 0, & \text{otherwise} \end{cases}$$

With

$$\lambda_{\max} = \sigma^2\left(1+\sqrt{\frac{K}{N}}\right)^2$$

$$\lambda_{\min} = \sigma^2\left(1-\sqrt{\frac{K}{N}}\right)^2$$

The idea is that if the eigen values of the covariance matrix lie within the two bounds, and follow the above-mentioned distribution, they represent random signal or noise. A practical way is to therefore take the PCs whose eigen values do not follow the distribution, meaning that they are non-random and provide a meaningful information.

If the data are standardized, we obtain the eigen values of the correlation matrix ***R***. As a result, the upper bound becomes $\lambda_{max} = \left(1 + \sqrt{\frac{K}{N}}\right)^2$. We only keep the PCs whose eigen values are superior to the upper bound.

3.3 Model training and forecasts

3.3.1 Machine learning algorithms and the naïve strategy

3.3.1.1 XGBOOST

Extreme Gradient Boosting, or XGBOOST, is a model composed of an ensemble of trees. Each tree assigns a score, which are then summed up to make the final prediction score. To build the trees, one must minimize the sum of an error function and a regularization term that penalizes the complexity of the trees.

$$Minimize\ \frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2 + \gamma T + \frac{1}{2}\lambda\sum_{j=1}^{T}w_j^2$$

With γ penalizing the number of leaves, T the number of the trees and λ controlling the value of the weights given to the leaves in each tree by penalizing the $l2$ norm of the weights.

XGBOOST relies on gradient boosting to improve the forecasting of the different trees. The equations representing the iterative gradient boosting are out of the scope of the Master thesis, but in a nutshell, the algorithm adds iteratively the best forecasting that optimally minimizes the objective function. The weights of the leaves are then adjusted based on the optimal improvement and the new computation of the objective function. Gradient boosting is called a greedy algorithm because it looks naively for the best direct improvement of the objective function. XGBOOST, therefore, adds new trees that are built onto the errors made by previous trees to optimize the weights. The interested reader can find the missing mathematical equations of the gradient boosting as well as a complete explanation of the algorithm in the paper of Chen

and Guestrin (2016). In our code, we use the Python version of the open-source package available in Github[1] (*XGBoost Python Package*, n.d.).

3.3.1.2 Support vector regression

The Support Vector Regression (SVR) consists of an optimization problem under constraint that can be written as follow:

$$Minimize: \sum_{i=1}^{N} (|y_i - f(x_i)| - \varepsilon)^2 + \frac{1}{2} C \sum_{i=1}^{N} w_i^2$$
$$Subject\ to: |y_i - f(x_i)| \leq \varepsilon, \ \forall\ i = 1,2,...,N$$

Similarly to XGBOOST, the objective function is a trade-off between the forecasting accuracy and a regularization term, only on the *l*2 norm in this case. The ε term represents a certain tolerance margin, called the epsilon tube. The function *f(.)* is an important choice to optimize the model. As mentioned in the literature review, carbon price follows non-linear patterns. Therefore, we opt for different non-linear functions such as sigmoid or radial basis, which is the one by default in the SVR function from the Scickit-learn module that we use (Pedregosa et al., 2011).

In addition, the objective function contains a regularization term with a penalty on the *l2* norm. The larger the ε and *C*, the more robust the model is. Intuitively, the constraint ensures that the absolute value of the error is effectively smaller than the tolerance tube. Finally, the Hyperparameters ε , *C* and the function *f(.)* will be chosen to optimally leverage the power of the algorithm.

3.3.1.3 Neural Network

Neural networks (NN) are characterized by an input layer, an output layer with a single neuron in this case and a hidden layer with several neurons. The different layers are connected by weights and the neurons have a function of activation to introduce some non-linearities. The objective of the NN is to minimize a measure of the error, which is the Mean Squared Error (MSE) in our case:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i))^2$$

[1] Available at: https://github.com/dmlc/xgboost

The weights and biases are optimized during the training with the backpropagation algorithm (Rumelhart et al., 1988). The algorithm bears its name because the information is propagated from the output layer to the input layer. In practice, the backpropagation algorithm is combined with more advanced optimisation algorithms to find and adjust the optimal weights while respecting the principle of backpropagation. For instance, the MLPRegressor function in Sickit-learn (Pedregosa et al., 2011) uses by default an algorithm called Adam proposed by Kingma and Ba (2014). To control the complexity of the model and avoid overfitting, the number of layers and neurons per layer is very important and will be tuned to optimize the accuracy.

3.3.1.4 Naïve Strategy

To verify that the Machine learning algorithms, and our forecasting strategy displays some forecasting power, it is compared with a naïve strategy. We first thought about applying a strategy forecasting 0% returns at each day. However, this strategy is not very performant for the reasons explained in Section 3.4.1.4. Therefore, we use a strategy, which consists of using the previous results of the independent variable as a forecast. This method effectively compares whether the models can learn from the other predictive variables, instead of solely relying on the time-series itself. Indeed, as the models have access to the previous weekly log returns of the independent variable, we expect them to use this information greatly in combination with the other variables to outperform the naïve strategy.

3.3.2 Hyperparameters tuning via Cross-validation

To obtain the best combination of hyperparameters, an adaptation of the cross-validation method that respects the order of the time-series is used. Classical cross-validation does not consider the time-order of the data, and simply shuffle the blocks randomly. In time-series, this situation would create a look-ahead bias.

We therefore use a particular cross-validation method that splits the training set into five different folds and uses the n first folds to predict the n+1 fold, for n belonging to [1-4]. In this case, the folds are ordered chronologically to respect the time-order of the time-series. Figure 7 illustrates graphically an example of the cross-validation method.

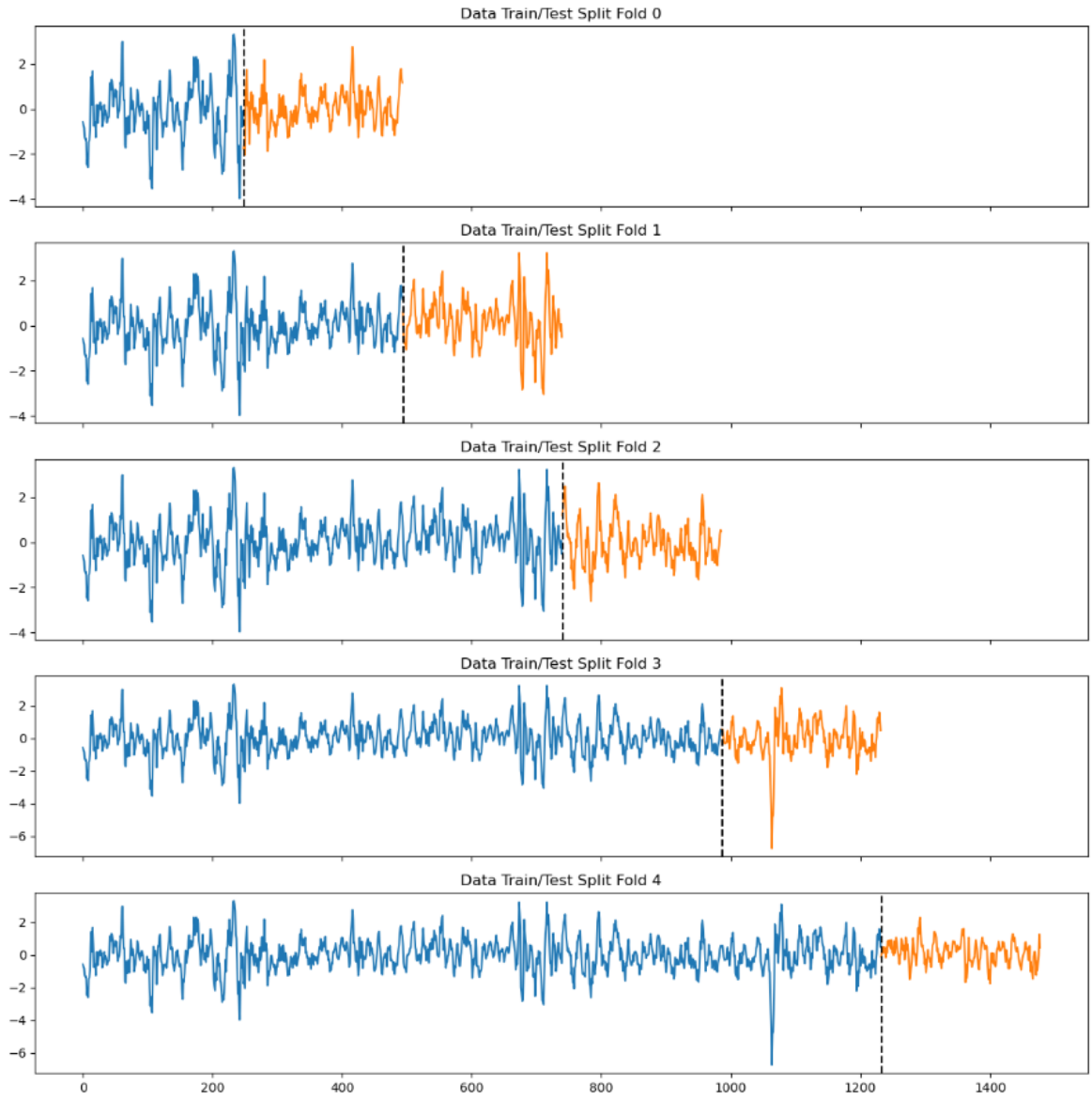

Figure 7: Example of the cross-validation method for weekly log returns in the full sample. The points in blue represent the training folds and the orange represent the testing fold.

More practically, a grid of values is defined for several hyperparameters, one combinations of hyperparameters is randomly selected. Then, the model makes predictions on the remaining blocs, and the mean of the test RMSE is reported.

These steps are repeated between 150 and 300 times depending on the model to train. As our computational power is limited, the number of iterations is limited to avoid endless testing periods. The best model out of the different combinations is selected and trained on the whole training set based on the 80-20 split. The grids search containing the hyperparameters of the 3 machine learning algorithms are displayed in Appendix 8.1.

3.4 Evaluation of the performance of the forecasting methods

3.4.1 Evaluation metrics

After having trained an optimal model, its performances must be evaluated. To do so, four standard and widely used performance metrics are considered: R^2, MAE, MAPE and RMSE. It is important to have several metrics to have a critical and complete view of the performance of the models.

3.4.1.1 Mean Squared error and Root Mean Squared Error

The Mean Squared Error (MSE) is a good metrics to spot very extreme values as it squares the error. The idea is that rare and large errors will be penalized more than small frequent errors (Chicco et al., 2021). The MSE has also a rooted version with the Root Mean Squared Error (RMSE). Their formulas are the following:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

$$RMSE = \sqrt{MSE}$$

3.4.1.2 Mean Average Error

The Mean Average Error (MAE) does not penalize too much the outliers conversely to the MSE. The formula is the following:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|Y_i - f(x_i)|$$

3.4.1.3 Mean Average Percentage Error

The Mean Average Percentage Error (MAPE) is the percentage version of the MAE. It allows to compare the errors in relative terms. The formula is the following:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{Y_i - f(x_i)}{Y_i}\right| \times 100$$

3.4.1.4 Coefficient of determination or R^2

The coefficient of determination or R^2 indicates the proportion of variability of the dependent variable explained by the independent variables. R^2 can be computed with the following formula:

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (Y_i - f(x_i))^2}{\sum_{i=1}^{n} (Y_i - \overline{Y})^2}$$

The coefficient of determination can be rewritten as $1 - \frac{Mean\ Squared\ errors}{Mean\ Total\ sum\ of\ squares}$ by dividing the Total squared errors and the Total sum of squares by *n.* As a result, if the MSE is equal to the MST, the R^2 score becomes 0. In other words, if one forecasts a straight line equal to the mean of the test set, the R^2 score becomes equal to 0. It also means that if the MSE is higher than the MST, the R^2 scores might become negative. Since the average of the weekly log returns is slightly above 0 as shown in Table 1, a naïve strategy consisting of forecasting a 0% return would have led to a larger MSE than the MST, and negative R^2. Consequently, we discarded the 0% naïve strategy.

3.4.2 Accuracy performance comparisons with the Diebold-Mariano test

To verify whether the physical data improve our forecasts, we compare the accuracy of the algorithms with and without the physical data. To do so, we must use a statistical test. Since we adopt machine learning algorithms because of their flexibility with regards to the assumption of the distribution of the data, we also use a test that do not assume an *a priori* distribution of the residuals or of the data.

To compare our models, we use a test proposed by Diebold and Mariano (2002). The test computes a loss function of the forecasted errors of two different set of forecasts. Then, it tests whether the two losses are different. With the incorporation of a loss function, the test enables to incorporate an asymmetry in the forecasting error that one would like to avoid, penalizing more the errors in one direction or the other. In our case, there is no a priori reason to penalize one error or the other, therefore we use the MSE of the forecasted sets as our loss function.

The Diebold-Mariano (DM) test is constructed on the difference between the two sets of forecasted errors divided by the square root of their autocovariance. The null hypothesis is that the result of the difference of the two loss functions is equal to 0. We use the python package called dieboldmariano[1] created by Edoardo Annunziata (2021).

4 Results

In this section, the results obtained after a series of tests comparing the performances of the different models are presented and discussed. We analyse the forecasting of the weekly log

[1] Available at: https://github.com/edoannunziata/dieboldmariano

returns using Marchenko-Pastur to retain the optimal number of PCs, and compare the results with a naïve strategy. Then, some robustness tests are performed to verify whether our findings still prove to be correct with respect to a change in the dependent variable and to different sub-samples. Finally, we analyse whether the Marchenko-Pastur bound improves the accuracy compared to some arbitrarily thresholds of variance explained by the PCs.

4.1 Results for the full sample

The accuracy of the weekly log returns forecasted by the three machine learning models with and without physical data can be found in Table 2. Additional information with regards to the weekly log returns can be found in Appendix 8.2.

		RMSE	MSE	MAPE	MAE	R2
XGBOOST	with	4.68E-02	2.19E-03	2.20E+02	3.46E-02	0.59
	without	4.63E-02	2.14E-03	2.36E+02	3.45E-02	0.60
NN	with	4.00E-02	1.60E-03	1.07E+03	3.02E-02	0.70
	without	4.00E-02	1.60E-03	2.86E+03	2.97E-02	0.70
SVR	with	4.75E-02	2.26E-03	1.68E+03	3.64E-02	0.57
	without	3.86E-02	1.49E-03	3.70E+03	2.84E-02	0.72
NAIVE STRATEGY	N/A	4.86E-02	2.36E-03	1.34E+03	3.51E-02	0.56

Table 2: Forecast performances of the three machine learning algorithms with and without physical data.

Before starting the analysis, one can observe that the MAPE values are extremely high. This phenomenon is explained by close-to-zero values of the weekly log returns predicted. As a result, the MAPE, by construction, skyrockets even for very small errors. This is a well-known weakness of the MAPE (see Kim & Kim, 2016), and therefore we decided to exclude it from the analysis.

4.1.1 Comparison of the Machine Learning algorithms

With and without physical data, the Naïve Strategy is consistently the poorest performer, which is reassuring. Nevertheless, it is not much worse than the other machine learning algorithms used. For instance, the MAE of XGBOOST without the physical data only slightly outperforms the naïve strategy with a 1.4% lower MAE.

With regards to the other algorithms, the SVR without the physical data provides the best results as its MAE is 5% lower than the second-best model, being the NN without the physical data.

Table 3 provides the values of two-sided DM tests comparing accuracy with and without the physical data. The null hypothesis is that the forecasts are the same while the alternative hypothesis is that the accuracy of the forecasts is different. To improve the readability of the Tables with DM results, the p-values below the threshold of 5% are highlighted in green (improvement of the performances) or red (worsening of the performances). A quick way to understand the signification of "different accuracy", given how the test is constructed, is to look at the sign of the test value. A positive value means a decrease in accuracy while a negative values indicates an increase in accuracy. We will use this convention for the rest of the Master's Thesis.

The p-values confirm that the results of XGBOOST are worse than the ones of its peers in most of the cases. In Table 2, SVR without the physical data seems to have the best results. Yet, the p-value shows that we cannot reject the hypothesis that the accuracy of SVR is truly different than the one of the NN. Consequently, we cannot infer anything. Conversely, the p-value of 1.06E-07 comparing SVR and NN with the physical data shows without doubt that the accuracy is different, which means that NN is more accurate given the positive value of the test.

		Diebold-Mariano	
		Test value	p-value
XGBOOST ≠ SVR	with	-0.61	0.55
	without	4.23	2.94E-05
XGBOOST ≠ NN	with	3.20	1.48E-03
	wihtout	3.86	1.34E-04
SVR ≠ NN	with	5.42	1.06E-07
	without	-1.84	0.07

Table 3: Statistical values and p-values of a two-sided Diebold Mariano test comparing the forecasting accuracy of the machine learning algorithms.

XGBOOST almost consistently underperforms its peers, which make it the worst performer. Indeed, comparing the accuracy of XGBOOST and the Naïve Strategy indicates that we cannot be certain that they are truly different. Moreover, NN and SVR have very close performances with an advantage for the NN, which shows more robustness with regards to the input data. Indeed, the standard deviation of the MAE for the NN is 3.84E-04, 10 times lower than for SVR (5.7E-03).

4.1.2 The effect of the physical data

At first glance, the results found in Table 2 show that the addition of the physical data impacts negatively the performances. To be sure, a DM test is performed to compare the accuracy of

the algorithms with and without the data. The null hypothesis is that the accuracy is the same, the alternative hypothesis is that it is different. Results can be found in Table 4.

	Diebdold-Mariano	
	Test value	p-value
XGBOOST	0.86	0.80
NN	-0.02	0.49
SVR	6.19	1.61E-09

Table 4: Statistical values and p-values of a two-sided Diebold Mariano comparing the forecasting accuracy with and without the physical data

The results confirm only partly the previous results as only the SVR proves to perform worse with the physical data. In Figure 8, one can see the predictions of the SVR with and without the physical data.

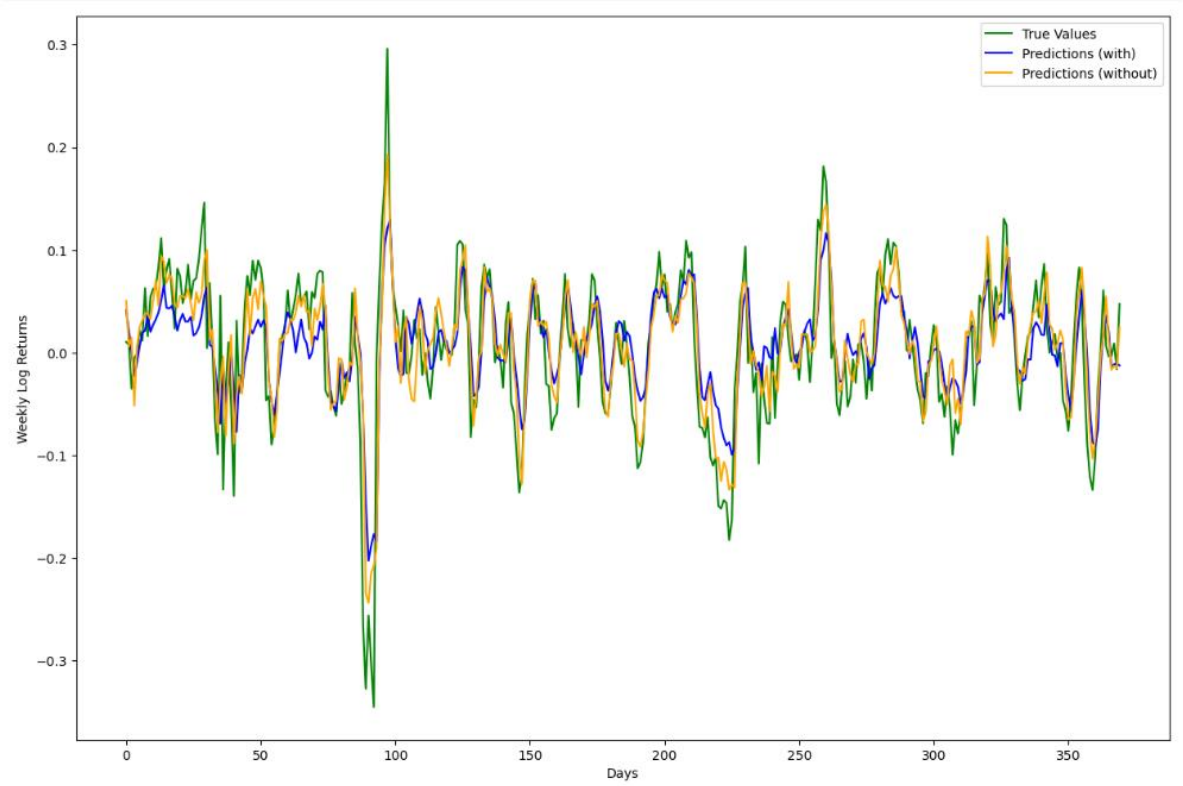

Figure 8: SVR predictions of the weekly log returns with and without the physical data on the full sample. Without the physical data SVR seems to systematically be closer to the real value than with the physical data.

For the two other algorithms, the p-values prevent us to conclude anything. We therefore do not know whether physical data truly impacts the accuracy of the predictions. In the following sections, we investigate whether these results remain true for different samples and dependent variables.

4.2 Robustness checks

4.2.1 Robust to subsample analysis

The literature review highlights the time-varying pattern of the variables driving the carbon markets, and the dynamic evolution of the EU ETS. Therefore, testing our hypothesis on a sub-sample might add some robustness to the results. In Machine learning, there is a strong trade-off between the quantity and the quality of the data. More data might confuse the machine as the patterns might change over time. However, too little data and the machine cannot properly "learn" the patterns. Consequently, the choice of the data, and therefore of the subsample, is highly important.

We cut the dataset in two at the date of the introduction of the MSR. This choice is motivated by the fact that the MSR is a genuine game changer for the EU ETS as it significantly reduces the total amount of emissions over the years (Bruninx et al., 2020). We therefore assume that the introduction has changed the dynamic and patterns of the carbon market. We denote the first subsample as "Pre-MSR" (24-12-2015 to 31-12-2018, 778 observations), and the second one as "Post-MSR"(02-01-2019 to 31-03-2023, 1095 observations). The performance metrics of the sub-sample analysis is presented in Table 5 and Table 6. Additional information about the subsample can be found in Appendix 8.2.

Pre-MSR		RMSE	MSE	MAPE	MAE	R2
XGBOOST	with	5.50E-02	3.03E-03	1.12E+02	3.89E-02	0.49
	without	5.18E-02	2.69E-03	1.11E+02	3.69E-02	0.55
NN	with	7.68E-02	5.90E-03	1.21E+02	5.66E-02	0.01
	without	7.68E-02	5.90E-03	1.21E+02	5.66E-02	0.01
SVR	with	7.23E-02	5.23E-03	1.13E+02	5.30E-02	0.13
	without	4.98E-02	2.48E-03	1.03E+02	3.54E-02	0.59
Naive Strategy	N/A	5.10E-02	2.60E-03	1.40E+02	3.60E-02	0.57

Table 5: Forecasts performances of three machine learning algorithms with and without physical data on the subsample "Pre-MSR"

Post-MSR		RMSE	MSE	MAPE	MAE	R2
XGBOOST	with	4.04E-02	1.63E-03	1.00E+03	3.29E-02	0.59
	without	4.01E-02	1.61E-03	1.58E+03	3.23E-02	0.60
NN	with	4.30E-02	1.85E-03	4.63E+03	3.43E-02	0.54
	without	4.89E-02	2.39E-03	1.48E+03	3.87E-02	0.41
SVR	with	5.91E-02	3.49E-03	8.57E+02	4.51E-02	0.13
	without	5.71E-02	3.26E-03	2.31E+03	4.39E-02	0.19
Naive Strategy	N/A	3.88E-02	1.50E-03	1.36E+03	3.12E-02	0.63

Table 6: Forecasts performances of three machine learning algorithms with and without physical data on the subsample "Post-MSR"

First of all, conversely to the full sample case, the naïve strategy achieves the best and second best results in the Pre and Post MSR samples. The decreasing performances from the machine

learning algorithms could be explained by two elements. First, the Pre-MSR sample contains very few data points for algorithms that might require much more information to learn appropriate patterns. The lack of data increases the variance of the estimators, which gives non-robust out-of-sample performances. Second, the hyperparameter tuning via grid search CV and cross-validation do not guarantee to find a global optimum. Due to limited computational power, every combination of the hyperparameters on the grid was not tested, which is already a subset of the limitless vector of values for each hyperparameter. In other words, the hyperparameter tuning step brings some randomness to the results. One algorithm might find an optimal combination in a sample, and misses it in another. It is a weakness of this work, which is explained in Section 5.1. For instance, the NN during the Pre-MSR sample offers poor results with a R^2 score close to 0, meaning that it is only slightly better than a straight line. It is most likely due to the lack of data and the sub-optimal hyperparameters combination.

In addition, without performing DM test, we can observe that the performances of the algorithms are variable, and there is no clear outperforming algorithm. This observation pledges in favour of using multiple algorithms to perform a task (i.e forecasting, classification…), and obtain more robust results. This method, called ensemble learning, follows the intuition that we do not know which algorithm will perform best. Therefore, we should use multiple algorithms to obtain better and more robust performances. Note that XGBOOST uses a form of ensemble learning called boosting, which consists of several predictors that learn from their peer mistakes to improve the results.

4.2.1.1 The effect of physical data

Table 7 contains the values of the DM tests for the two subsamples comparing the accuracy with and without the physical data. The alternative hypothesis is that the forecasts are different. The p-values confirm that the forecasts are different in some cases such as for XGBOOST in Pre-MSR or SVR in Post-MSR. We observe that that the physical data worsen the performances most of the time. The only cases the physical data improves the accuracy is for the NN during the Post-MSR sample. This confirms that the physical data confuse the algorithms, and do not help them to sharpen their predictions.

	Pre-MSR		Post-MSR	
	Diebdold-Mariano		Diebdold-Mariano	
	Test value	p-value	Test value	p-value
XGBOOST	3.28	1.30E-03	0.42	0.68
NN	1.87	6.38E-02	-2.39	1.74E-02
SVR	4.35	2.54E-05	2.65	8.59E-03

Table 7: Statistical values and p-values of a two-sided Diebold Mariano test comparing the forecasting accuracy with and without the physical data

4.2.2 Robust to a new dependent variable

The literature review also highlighted the impacts of the energy markets on the carbon price at different horizons of time. Therefore, it is important to identify a potential impact on a (relatively) longer horizon. We chose "monthly" (i.e., lag of 20 days) log returns to study a potential longer-term impact. We therefore accordingly change our feature to incorporate the 20 previous monthly log returns instead of the 20 previous weekly log returns. Table 8 compares the performances of the 3 algorithms and the naïve strategy with and without the physical data on the full sample.

		RMSE	MSE	MAPE	MAE	R2
XGBOOST	with	5.57E-02	3.10E-03	9.72E+01	4.06E-02	0.82
	without	4.93E-02	2.43E-03	9.56E+01	3.70E-02	0.86
NN	with	8.17E-02	6.67E-03	1.30E+02	6.31E-02	0.61
	without	6.33E-02	4.01E-03	1.27E+02	5.05E-02	0.76
SVR	with	1.19E-01	1.41E-02	1.89E+02	9.12E-02	0.17
	without	5.12E-02	2.62E-03	9.62E+01	3.87E-02	0.85
Naive Strategy	N/A	4.83E-02	2.34E-03	1.04E+02	3.62E-02	0.86

Table 8: Forecasts performances for monthly log returns with and without physical data on the full sample.

No strategy manages to outperform the naïve strategy. The closest competitor is XGBOOST without the physical data with a 2% higher MAE. This result shows that our machine learning algorithms fail to identify any relevant signals, and are only overfitting noise. We can conclude so because our algorithms use the same feature as the naïve strategy combined with other features (lag and additional variables such as gas price, etc.). The underperformance therefore means our models fail to incorporate the relevant patterns, if any. Additional results and information about the "monthly" log returns can be found in Appendix 8.3.

4.2.2.1 The effect of the physical data

In Table 9 , the results of the DM tests are displayed. The alternative hypothesis is that the forecasts are different as previously.

	Full sample		Pre-MSR		Post-MSR	
	Diebdold-Mariano		Diebdold-Mariano		Diebdold-Mariano	
	Test value	p-value	Test value	p-value	Test value	p-value
XGBOOST	3.34	9.33E-04	-2.48	1.43E-02	-6.94	4.70E-11
NN	6.17	1.77E-09	6.43	1.68E-09	9.34	1.44E-17
SVR	11.02	1.40E-24	6.94	1.17E-10	-7.18	1.17E-11

Table 9: Statistical values and p-values of a two-sided Diebold Mariano test comparing the forecasting accuracy with and without the physical data for monthly log returns.

Not only the models fail to outperform the naïve strategy, but the addition of the physical data worsens once again the performances. The p-values mean that for the threshold of 5%, the addition of the physical implies that the accuracies of the predictions are worse most of the time.

Yet, interestingly, if there is an improvement in the accuracy, it takes place during the sample Post-MSR. This sample is characterized by huge energy commodity volatility -as one can see in Figure 5- due to several exogeneous factors. It is also the case for the weekly log returns, although it is less clear. Our hypothesis is that during market turmoil, physical data might prove relevant to be included in order to obtain better performances.

4.3 The impact of the Marchenko-Pastur bound

To study the impact of the Marchenko-Pastur bound, we decided to use XGBOOST because it i) converges and provides a "satisfactory" solution, which is not the case for the NN in some sub-samples and ii) is less sensitive to the input variables compared to the SVR based on the results in Tables Table 2,Table 5,Table 6 and Table 8. Since, we test whether the choice of the PCs that ultimately fix the number of input variables has any impact on its performances, it is important to have an algorithm that provides satisfactory results in all cases, and that is robust to the input features.

As a result, if XGBOOST proves to be impacted by the number of PCs selected, we can assume that it is also the case for less robust algorithms. However, if XGBOOST proves to not be impacted, we cannot generalize this finding as it does not mean that other algorithms are not affected. As one can see on the Figure 9, there is a substantial amount of PCs that represent tiny portions of the variance.

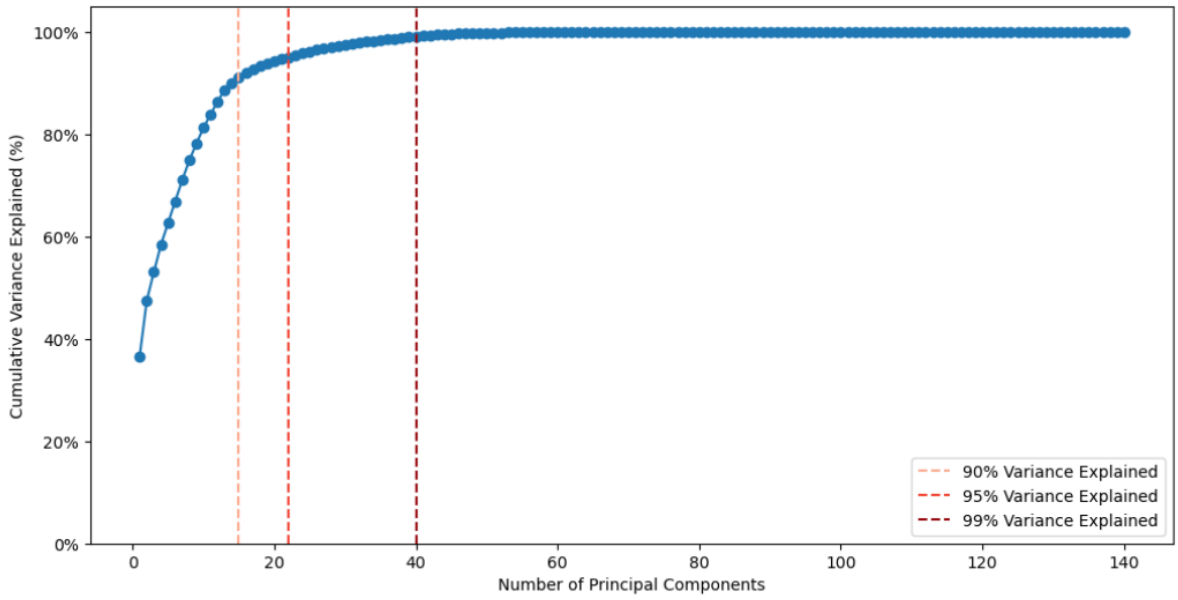

Figure 9: The variance of the data explained in function of the number of Principal Components. The number of PCs increases exponentially as the variance explained approaches 100%. Example of the weekly log returns in full sample with the physical data.

4.3.1 Weekly log returns

The results of the two-sided Diebold-Mariano test comparing the performances of XGBOOST with and without the Marchenko-Pastur bound can be found in Table 10. The alternative hypothesis is that the forecasts of XGBOOST with Marchenko-Pastur filtering the PCs are different than the ones without.

		95% variance explained		99% variance explained	
		Diebold-Mariano		Diebold-Mariano	
		Test value	P-value	Test value	P-value
Full	with	-1.61	0.11	-1.61	0.11
	without	0.00	1.00	1.40	0.16
Pre-MSR	with	-3.20	0.00	-3.20	0.00
	without	0.00	1.00	0.00	1.00
Post-MSR	with	-0.15	0.88	-0.30	0.76
	without	-2.44	0.02	1.82	0.07

Table 10: Statistical values and p-values of a two-sided Diebold Mariano test comparing the forecasting accuracy of the forecasts with the Marchenko-Pastur bound and without for different percentages of variance explained by the PCs for weekly log returns.

Two things can be implied. First, the Marchenko-Pastur bound seem to outperform arbitrarily thresholds, which is coherent to the theory. Second, we can see that the test value is equal to 0 in some cases. This happens when two forecasts are identical, meaning that the algorithm reaches the same output whatever the filtering method used. In the next section, we check whether we obtain similar results for the monthly log returns. Additional information can be found in Appendix 8.4 as well.

4.3.2 Monthly log returns

In Table 11, one can find the results of the two-sided Diebold-Mariano test comparing the performances of XGBOOST with and without the Marchenko-Pastur bound. The alternative hypothesis is that the forecasts of XGBOOST with Marchenko-Pastur filtering the PCs are different than the ones without.

		95% variance explained		99% variance explained	
		Diebold-Mariano		Diebold-Mariano	
		Test value	P-value	Test value	P-value
Full	with	3.41	0.00	3.41	0.00
	without	0.00	1.00	1.40	0.16
Pre-MSR	with	-2.76	0.01	1.61	0.11
	without	2.05	0.04	2.05	0.04
Post-MSR	with	-6.24	0.00	-6.84	0.00
	without	0.00	1.00	6.93	0.00

Table 11: Statistical values and p-values of a two-sided Diebold Mariano test comparing the forecasting accuracy of the forecasts with Marchenko-Pastur bound and without for different percentages of variance explained by the PCs for monthly log returns.

The picture emerging from the results is quite different than for weekly log returns. For monthly log returns, the abundance of values in red means that the Marchenko-Pastur bound is less performant than arbitrarily thresholds. One can notice that the test values seem to increase with the variance explained, which would signify that the more variance explained, the better the accuracy of XGBOOST. This phenomenon is also present for weekly log returns, yet is less clear.

This is not coherent with our intuition, and the random matrix theory. However, here are two potential explanations. First, Marchenko-Pastur bound without treatment of the eigenvalues of the covariance matrix might lead to a miscomputation of the bound because of the distortion that the eigenvalues undergo. As a result, meaningful signal might be lost because of a bad estimation of the eigenvalues. Actually, this is a common problem, and Stein (1975) provides notably a robust estimator of the eigenvalues that converges towards the true eigenvalues as the number of observations reaches infinity. While this might explain why Marchenko-Pastur underperforms arbitrarily thresholds, how can we explain that the 99% threshold outperform the 95% threshold? Randomness of the hyperparameter tuning step as already mentioned is a track that we cannot dismiss. Indeed, the tuning might find a more accurate combination for the 99% threshold than for the 95% threshold. While randomness might be the reason, it might also be the case that XGBOOST panoply of hyperparameters offer more robustness, and is more performant than simple de-noising methods such as the Marchenko-Pastur bound.

We believe that our implementation of the Marchenko-Pastur bound was far from optimal, and that an improved version or other more powerful feature selection methods must be combined with fine-tuned machine learning algorithms. This should enhance the performances, as well as saving a substantial amount of time.

5 Limits

5.1 Technical limits of the models

The models proposed in this work suffer some technical limitations. First, the hyperparameters were tuned using a 5-fold cross-validation method and a grid search. Although it is not wrong to do so, the lack of computational power forced us to explore a limited number of combinations of hyperparameters, which most likely reached a suboptimal combination. Second, even though carbon price has time-varying patterns, the models were not dynamically train . To the best of our knowledge, the dynamic forecasting method of Zhang et al. (2022) seems to be the most performant one so far with excellent results as already mentioned in the review of literature. Finally, the Marchenko-Pastur bound was naively applied, and more advanced de-noising, and feature selections methods could greatly enhance the performances.

5.2 Look-ahead bias

There exists a (minor) look-ahead bias in our study given the fact that we standardize the training and testing sets together. As a result, there is an implicit revelation of future information, which might have impacted our results to the upside. To solve this issue, the implementation of an independent validation set or a time-varying standardization method are two potential solutions for future implementations.

5.3 Lack of comparisons with other studies

Most of the literature focusses on forecasting day ahead prices. We forecast weekly or monthly log returns, which do not allow a like-for-like comparison. We could have transformed the log returns in daily prices, and then assessed the forecasted daily prices, but it does not provide a relevant comparison as we forecast one week ahead. In addition, very few papers that forecasted phase IV have been published so far, which do not help either. As highlighted in Section 4.2.1, sample matters a lot for the performance of the algorithms. To allow a genuine like-for-like comparison, we should have applied previous methodologies and algorithms to our data sets, which is practically impossible for us. This lack of comparison motivated the addition of a naïve strategy to have a point of comparison. This being said, we do not believe that our model can

compete with other models presented during the literature review given all the limits mentioned in Section 5.1.

5.4 Lack of explainability

Machine learning models are identified as “black-boxes”, which means that they severely lack readability. Conversely to simpler econometric methods, Machine Learning algorithms are harder to analyse et require specific methods to properly be studied. One of the methods is the use of Shapley values as shown in the study of Lundberg and Lee (2017). Having not used such methods, we must rely on assumptions and potential explanations instead of assertive explanations. In addition, the randomness of the hyperparameter tuning step signifies that we struggle to have a definitive answer to our questions.

For instance, we argue that physical data worsen the performances most of the time, unless potentially during the Post-MSR sample. Are the differences in performances actually related to the physical data or the hyperparameter tuning period ? Is it related to the hyperparameter tuning step or to a very specific choice of the sub-samples or the independent variables? In general, we found it very difficult to conclude anything from the battery of tests we performed. If we had to redo the master thesis, we would definitely use Shapley values or likewise methods to “open” the black box, and understand it further.

6 Conclusion

Our objective was to forecast carbon returns with machine learning methods and study the impact of an alternative source of data on the accuracy. Our results show that the addition of the two physical data variables do not systematically improve the accuracy of the forecasts. Worse, it might lead to a decrease in accuracy. The most likely reason is that the gas market incorporates all the information, making our two physical data noisy redundant features. Yet, it is worth mentioning that if the physical data generate superior performances, it seems to take place during the sample “Post-MSR” which spans over the recent energy market turbulences. During very extreme energy markets volatility, physical data might prove worth to be considered. The reason is that physical flows can hardly be falsifiable and are the main cause for the price of the underlying commodity to skyrocket or for the carbon to be emitted. Therefore, it might prove to be a good indicator during market irrationality or extreme market turmoil. We can only encourage the European Union and other regions of the world to publish more granular data about physical flows.

In addition, we did not find that filtering the PCs with Marchenko-Pastur bound reveals important for well-parametrized and robust machine learning algorithms. We suspect that there are more powerful forms of filtering or features selection methods that would have delivered better results. We encourage therefore to find alternative methods combined with well-parametrized and powerful machine learning algorithms to identify the relevant features and find the right degree of robustness.

For future research, it would be interesting to keep looking for additional features to explain a greater portion of the variability of the carbon price movements. One idea could be to create a policy index that would combine the impact of the long-term renewable targets for the energy sector and the uncertainty that market participants face at the approach of a new phases often synonyms of more tightened ETS rules. Another could be to integrate broader financial assets to consider the market participants looking the carbon market as a diversification tool. Another idea would be to combine machine learning computational power with feature permutations to effectively quantify the importance of each variables, and improve the understanding of the price formation of the ETS. Finally, the relationship between the number of variables and the robustness of the machine learning algorithms based on their hyperparameters could be an interesting area for future research to quantify how robust is a given set of hyperparameters combination.

7 Bibliography

Aatola, P., Ollikainen, M., & Toppinen, A. (2013). Price determination in the EU ETS market: Theory and econometric analysis with market fundamentals. *Energy Economics*, *36*, 380–395. https://doi.org/10.1016/j.eneco.2012.09.009

Batten, J. A., Maddox, G. E., & Young, M. R. (2021). Does weather, or energy prices, affect carbon prices? *Energy Economics*, *96*, 105016. https://doi.org/10.1016/j.eneco.2020.105016

Benz, E., & Trück, S. (2009). Modeling the price dynamics of CO2 emission allowances. *Energy Economics*, *31*(1), 4–15. https://doi.org/10.1016/j.eneco.2008.07.003

Bredin, D., & Muckley, C. (2011). An emerging equilibrium in the EU emissions trading scheme. *Energy Economics*, *33*(2), 353–362. https://doi.org/10.1016/j.eneco.2010.06.009

Bruninx, K., Ovaere, M., & Delarue, E. (2020). The long-term impact of the market stability reserve on the EU emission trading system. *Energy Economics*, *89*, 104746. https://doi.org/10.1016/j.eneco.2020.104746

Byun, S. J., & Cho, H. (2013). Forecasting carbon futures volatility using GARCH models with energy volatilities. *Energy Economics*, *40*, 207–221. https://doi.org/10.1016/j.eneco.2013.06.017

Chen, T., & Guestrin, C. (2016). *XGBoost: A Scalable Tree Boosting System*. https://doi.org/10.1145/2939672.2939785

Chevallier, J. (2009). Carbon futures and macroeconomic risk factors: A view from the EU ETS. *Energy Economics*, *31*(4), 614–625. https://doi.org/10.1016/j.eneco.2009.02.008

Chevallier, J. (2011a). A model of carbon price interactions with macroeconomic and energy dynamics. *Energy Economics*, *33*(6), 1295–1312. https://doi.org/10.1016/j.eneco.2011.07.012

Chevallier, J. (2011b). Nonparametric modeling of carbon prices. *Energy Economics*, *33*(6), 1267–1282. https://doi.org/10.1016/j.eneco.2011.03.003

Chevallier, J., Nguyen, D. K., & Reboredo, J. C. (2019). A conditional dependence approach to CO2-energy price relationships. *Energy Economics*, *81*, 812–821. https://doi.org/10.1016/j.eneco.2019.05.010

Chicco, D., Warrens, M. J., & Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ*, *7*, e623. https://doi.org/10.7717/peerj-cs.623

Dickey, D. A., & Fuller, W. A. (1979). Distribution of the Estimators for Autoregressive Time Series with a Unit Root. *Journal of the American Statistical Association*, *74*(366a), 427–431. https://doi.org/10.1080/01621459.1979.10482531

Diebold, F. X., & Mariano, R. S. (2002). Comparing Predictive Accuracy. Journal of Business & Economic Statistics, 20(1), 134–144. http://www.jstor.org/stable/1392155

Duan, K., Ren, X., Shi, Y., Mishra, T., & Yan, C. (2021). The marginal impacts of energy prices on carbon price variations: Evidence from a quantile-on-quantile approach. *Energy Economics*, *95*, 105131. https://doi.org/10.1016/j.eneco.2021.105131

Ellerman, A. D., Marcantonini, C., & Zaklan, A. (2016). The European Union Emissions Trading System: Ten Years and Counting. *Review of Environmental Economics and Policy*, *10*(1), 89–107. https://doi.org/10.1093/reep/rev014

EU Emissions Trading System (EU ETS). (n.d.). Climate Action. https://climate.ec.europa.eu/eu-action/eu-emissions-trading-system-eu-ets_en

Financial Times. (2022, December 19). EU energy ministers reach deal on gas price cap. *Financial Times*. https://www.ft.com/content/5b2ffae4-04d1-4e09-89ce-b85f575d8422

Foroni, C., Ravazzolo, F., & Rossini, L. (2023). Are low frequency macroeconomic variables important for high frequency electricity prices? Economic Modelling, 120, 106160. https://doi.org/10.1016/j.econmod.2022.106160

García-Martos, C., Rodríguez, J. P., & Sánchez, M. L. Z. (2013). Modelling and forecasting fossil fuels, CO2 and electricity prices and their volatilities. *Applied Energy*, *101*, 363–375. https://doi.org/10.1016/j.apenergy.2012.03.046

Gas Infrastructure Europe - AGSI. (n.d.). https://agsi.gie.eu/

Ghysels, E., Santa-Clara, P., & Valkanov, R. (2006). Predicting volatility: getting the most out of return data sampled at different frequencies. *Journal of Econometrics*, *131*(1–2), 59–95. https://doi.org/10.1016/j.jeconom.2005.01.004

Hammoudeh, S., Nguyen, D. K., & Sousa, R. M. (2014a). Energy prices and CO2 emission allowance prices: A quantile regression approach. *Energy Policy*, *70*, 201–206. https://doi.org/10.1016/j.enpol.2014.03.026

Hammoudeh, S., Nguyen, D. K., & Sousa, R. M. (2014b). What explain the short-term dynamics of the prices of CO2 emissions? *Energy Economics*, *46*, 122–135. https://doi.org/10.1016/j.eneco.2014.07.020

Huang, N. E., Shen, Z., Long, S. R., Wu, M. C., Shih, H. H., Zheng, Q., Yen, N.-C., Tung, C. C., & Liu, H. H. (1998). The Empirical Mode Decomposition and the Hilbert Spectrum for Nonlinear and Non-Stationary Time Series Analysis. Proceedings: Mathematical, Physical and Engineering Sciences, 454(1971), 903–995. http://www.jstor.org/stable/53161

Huang, G. (2014). An Insight into Extreme Learning Machines: Random Neurons, Random Features and Kernels. *Cognitive Computation*, *6*(3), 376–390. https://doi.org/10.1007/s12559-014-9255-2

Huang, Y., Dai, X., Wang, Q., & Zhou, D. (2021). A hybrid model for carbon price forecasting using GARCH and long short-term memory network. *Applied Energy*, *285*, 116485. https://doi.org/10.1016/j.apenergy.2021.116485

IEA. (2020). *Electricity Market Report - December 2020 – Analysis - IEA*. Retrieved May 28, 2023, from https://www.iea.org/reports/electricity-market-report-december-2020

IEA. (2023). *Natural Gas Information - Data product - IEA*. Retrieved March 12, 2023, from https://www.iea.org/data-and-statistics/data-product/natural-gas-information#documentation

Jaramillo-Morán, M. A., Fernández-Martínez, D., García, A. G., & Carmona-Fernández, D. (2021). Improving Artificial Intelligence Forecasting Models Performance with Data Preprocessing: European Union Allowance Prices Case Study. *Energies*, *14*(23), 7845. https://doi.org/10.3390/en14237845

Jaramillo-Morán, M. A., Fernández-Martínez, D., García-García, A., & Carmona-Fernández, D. (2021). Improving Artificial Intelligence Forecasting Models Performance with Data Preprocessing: European Union Allowance Prices Case Study. *Energies*, *14*(23), 7845. https://doi.org/10.3390/en14237845

Jaramillo-Morán, M. A., & García-García, A. (2019). Applying Artificial Neural Networks to Forecast European Union Allowance Prices: The Effect of Information from Pollutant-Related Sectors. *Energies*, *12*(23), 4439. https://doi.org/10.3390/en12234439

Jarque, C. M., & Bera, A. K. (1980). Efficient tests for normality, homoscedasticity and serial independence of regression residuals. *Economics Letters*, *6*(3), 255–259. https://doi.org/10.1016/0165-1765(80)90024-5

Kamrani, A. K., & Gonzalez, R. (2003). A genetic algorithm-based solution methodology for modular design. Journal of Intelligent Manufacturing, 14(6), 599-616. https://www.proquest.com/scholarly-journals/genetic-algorithm-based-solution-methodology/docview/200543038/se-2

Kim, S., & Kim, H. (2016). A new metric of absolute percentage error for intermittent demand forecasts. *International Journal of Forecasting*, *32*(3), 669–679. https://doi.org/10.1016/j.ijforecast.2015.12.003

Kingma, D.P., & Ba, J. (2014). Adam: A Method for Stochastic Optimization. CoRR, abs/1412.6980.

Koop, G., & Tole, L. (2013). Forecasting the European Carbon Market. *Journal of the Royal Statistical Society*, *176*(3), 723–741. https://doi.org/10.1111/j.1467-985x.2012.01060.x

Lago, J., De Ridder, F., & De Schutter, B. (2018). Forecasting spot electricity prices: Deep learning approaches and empirical comparison of traditional algorithms. *Applied Energy*, *221*, 386–405. https://doi.org/10.1016/j.apenergy.2018.02.069

Lovcha, Y., Perez-Laborda, A., & Sikora, I. (2022). The determinants of CO2 prices in the EU emission trading system. *Applied Energy*, *305*, 117903. https://doi.org/10.1016/j.apenergy.2021.117903

Lundberg, S. M., Erion, G. G., & Lee, S. (2018). *Consistent individualized feature attribution for tree ensembles.https://doi.org/10.48550/arxiv.1802.03888*

Market Stability Reserve. (n.d.). Climate Action. https://climate.ec.europa.eu/eu-action/eu-emissions-trading-system-eu-ets/market-stability-reserve_en

Mirjalili, S., & Lewis, A. L. (2016). The Whale Optimization Algorithm. *Advances in Engineering Software*, *95*, 51–67. https://doi.org/10.1016/j.advengsoft.2016.01.008

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *HAL (Le Centre Pour La Communication Scientifique Directe)*. https://hal.inria.fr/hal-00650905

Rehman, N., & Mandic, D. P. (2010). Multivariate empirical mode decomposition. Proceedings: Mathematical, Physical and Engineering Sciences, 466(2117), 1291–1302. http://www.jstor.org/stable/25661497

Rehman, N. U., & Mandic, D. P. (2011). Filter Bank Property of Multivariate Empirical Mode Decomposition. *IEEE Transactions on Signal Processing*, *59*(5), 2421–2426. https://doi.org/10.1109/tsp.2011.2106779

Rudnik, K., Hnydiuk-Stefan, A., Kucińska-Landwójtowicz, A., & Mach, U. (2022). Forecasting Day-Ahead Carbon Price by Modelling Its Determinants Using the PCA-Based Approach. *Energies*, *15*(21), 8057. https://doi.org/10.3390/en15218057

Rumelhart, D. E., Hinton, G. E., & Williams, R. B. (1988). Learning representations by back-propagating errors. *Nature*, *323*(6088), 533–536. https://doi.org/10.1038/323533a0

Sato, M., Rafaty, R., Calel, R., & Grubb, M. (2022). Allocation, allocation, allocation! The political economy of the development of the European Union Emissions Trading System. *Wiley Interdisciplinary Reviews: Climate Change*, *13*(5). https://doi.org/10.1002/wcc.796

Sijm, J., Neuhoff, K., & Chen, Y. (2006). CO_2cost pass-through and windfall profits in the power sector. *Climate Policy*, *6*(1), 49–72. https://doi.org/10.1080/14693062.2006.9685588

Stein C (1975) Estimation of a covariance matrix. Rietz Lecture

Sun, W., & Zhang, C. (2018). Analysis and forecasting of the carbon price using multi—resolution singular value decomposition and extreme learning machine optimized by adaptive whale optimization algorithm. Applied Energy, 231, 1354–1371. https://doi.org/10.1016/j.apenergy.2018.09.118

Suykens, J. a. K., Lukas, L., Van Dooren, P., & Vandewalle, J. (1999). Least squares support vector machine classifiers: a large scale algorithm. In European Conference on Circuit Theory and Design (pp. 839–842). https://ci.nii.ac.jp/naid/10015058609

Tan, X. P., & Wang, X. Y. (2017). Dependence changes between the carbon price and its fundamentals: A quantile regression approach. *Applied Energy*, *190*, 306–325. https://doi.org/10.1016/j.apenergy.2016.12.116

Tan, X., Sirichand, K., Vivian, A., & Wang, X. (2020). How connected is the carbon market to energy and financial markets? A systematic analysis of spillovers and dynamics. *Energy Economics*, *90*, 104870. https://doi.org/10.1016/j.eneco.2020.104870

The Economist. (2021, October 16). *The energy shock | Oct 14th 2021 | The Economist*. https://www.economist.com/weeklyedition/2021-10-16

United Nations. (n.d.). *The Paris Agreement*. https://www.un.org/en/climatechange/paris-agreement

Wang, J., Sun, X., Cheng, Q., & Cui, Q. (2021). An innovative random forest-based nonlinear ensemble paradigm of improved feature extraction and deep learning for carbon price forecasting. *Science of the Total Environment*, *762*, 143099. https://doi.org/10.1016/j.scitotenv.2020.143099

Wettestad, J. (2014). Rescuing EU Emissions Trading: Mission Impossible? *Global Environmental Politics*, *14*(2), 64–81. https://doi.org/10.1162/glep_a_00229

XGBoost Python Package. (n.d.). Dmcl XGBOOST Stable. Retrieved April 5, 2023, from https://xgboost.readthedocs.io/en/stable/python/index.html

Zachmann, G., Sgaravatti, G., & McWilliams, B. (2022, December 16). *European natural gas imports*. Bruegel | the Brussels-based Economic Think Tank. https://www.bruegel.org/dataset/european-natural-gas-imports

Zeqiraj, V., Sohag, K., & Soytas, U. (2020). Stock market development and low-carbon economy: The role of innovation and renewable energy. *Energy Economics*, *91*, 104908. https://doi.org/10.1016/j.eneco.2020.104908

Zhang, J., Li, D., Hao, Y., & Tan, Z. (2018). A hybrid model using signal processing technology, econometric models and neural network for carbon spot price forecasting. *Journal of Cleaner Production*, *204*, 958–964. https://doi.org/10.1016/j.jclepro.2018.09.071

Zhang, W., Wu, Z., Zeng, X., & Zhu, C. (2022). An ensemble dynamic self-learning model for multiscale carbon price forecasting. *Energy*, *263*, 125820. https://doi.org/10.1016/j.energy.2022.125820

Zhao, X., Han, M., Ding, L., & Kang, W. (2018). Usefulness of economic and energy data at different frequencies for carbon price forecasting in the EU ETS. *Applied Energy*, *216*, 132–141. https://doi.org/10.1016/j.apenergy.2018.02.003

Zhou, F., Huang, Z., & Zhang, C. (2022). Carbon price forecasting based on CEEMDAN and LSTM. *Applied Energy*, *311*, 118601. https://doi.org/10.1016/j.apenergy.2022.118601

Zhou, J., Huo, X., Xu, X., & Li, Y. (2019). Forecasting the Carbon Price Using Extreme-Point Symmetric Mode Decomposition and Extreme Learning Machine Optimized by the Grey Wolf Optimizer Algorithm. *Energies*, *12*(5), 950. https://doi.org/10.3390/en12050950

Zhu, B. (2012). A Novel Multiscale Ensemble Carbon Price Prediction Model Integrating Empirical Mode Decomposition, Genetic Algorithm and Artificial Neural Network. *Energies*, *5*(2), 355–370. https://doi.org/10.3390/en5020355

Zhu, B., Shi, X., Chevallier, J., Wang, P., & Wei, Y. (2016). An Adaptive Multiscale Ensemble Learning Paradigm for Nonstationary and Nonlinear Energy Price Time Series Forecasting. *Journal of Forecasting*, *35*(7), 633–651. https://doi.org/10.1002/for.2395

Zhu, B., & Wei, Y. (2013). Carbon price forecasting with a novel hybrid ARIMA and least squares support vector machines methodology. *Omega*, *41*(3), 517–524. https://doi.org/10.1016/j.omega.2012.06.005

Zhu, B., Ye, S., Han, D. S., Wang, P., He, K., Wei, Y., & Xie, R. (2019). A multiscale analysis for carbon price drivers. *Energy Economics*, *78*, 202–216. https://doi.org/10.1016/j.eneco.2018.11.007

Zhu, B., Ye, S., Wang, P., Chevallier, J., & Wei, Y. (2021). Forecasting carbon price using a multi-objective least squares support vector machine with mixture kernels. *Journal of Forecasting*, *41*(1), 100–117. https://doi.org/10.1002/for.2784

8 Appendix:

8.1 Grid search

8.1.1 XGBOOST

	Max Depth	Learning Rate	Reg Lambda	Reg Alpha	Min split Loss	Base Score
XGBOOST	2.000E+00	5.000E-03	1.000E+00	1.000E+00	0.000E+00	3.000E-01
	3.000E+00	1.000E-02	5.000E+00	2.000E+00	1.000E+00	4.000E-01
	4.000E+00		1.000E+01	1.000E+01	3.000E+00	5.000E-01
	5.000E+00		2.000E+01	2.000E+01	4.000E+00	6.000E-01
	6.000E+00		1.000E+02	1.000E+02	5.000E+00	7.000E-01
	7.000E+00				6.000E+00	
					1.000E+01	

- Max Depth designates the depth of the trees.
- Learning rate is shrinkage intensity of the features weights during the boosting process.
- Reg Lambda is a regularization term on the l2 norm.
- Reg Alpha is a regularization term on the l1 norm.
- Min Split Loss is used to create additional partitions of the data at the end node of a leaf.
- Base score is the starting point of the algorithm.

More information can be found on this website: https://xgboost.readthedocs.io/en/latest/parameter.html

8.1.2 Neural Network

	Architecture	Kernel	Alpha	Learning Rate
Neural Network	(32,50,32)	"Logistic"	1.000E-02	1.000E-03
	(100,100)	"Relu"	1.000E-01	5.000E-03
	(50,50)	"Tanh"	1.000E+00	1.000E-02
	(50,100,50)		5.000E+00	
			1.000E+01	
			2.000E+01	
			1.000E+02	

- Architecture defines the number of layers and the neurons per layer. For instance, (32,50,32) means that there are three layers composed of 32, 50 and 32 neurons.
- Kernel is the activation function for the hidden layers that introduce non-linearity in the algorithm.
- Alpha denotes the strength of the l2 regularization
- Learning Rate is used to update the weights of the neurons

More information can be found on this website: https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPRegressor.html

8.1.3 Support Vector Regression

	C	Kernel	Tol	Epsilon	Coef0	Degree
Support Vector Machine	1.000E-05	"Polynomial"	1.000E-04	1.000E-03	-2.000E+00	1.00
	1.000E-03	"Radial basis"	1.000E-03	1.000E-02	-1.500E+00	2.00
	1.000E-02	"Sigmoid"	1.000E-02	1.000E-01	-1.200E+00	3.00
	1.000E-01		1.000E-01	1.000E+00	-1.000E+00	4.00
	1.000E+00		1.000E+00	1.100E+00	-3.000E-01	5.00
	1.000E+01			1.200E+00	-2.000E-01	6.00
	1.500E+01			1.300E+00	-1.000E-01	7.00
	1.000E+02			1.500E+00	0.000E+00	8.00
				1.600E+00	5.000E-01	
				1.700E+00	1.000E+00	
					2.000E+00	

- C is a regularization term that adds a l2 penalty. The regularisation is inversely proportional to the value of C.
- Kernel is the type of functions used to define the kernel of the algorithm.
- Tol is the tolerance acceptance in the algorithm. It is a stopping criterion.
- Espilon is the tube tolerance in which the errors are not penalized.
- Coef0 is a parameter associated with the kernel functions “Polynomial” and “Sigmoid”.

- Degree is a parameter associated with the kernel function “Polynomial”.

More information can be found on this website: https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html

8.2 Weekly log returns

8.2.1 Moments of the distribution, Jarque-Bera and Augmented Dickey-Fuller tests

	Mean	Variance	Skewness	Kurtosis	JB		ADF	
					Test value	P-value	Test value	P-value
Pre-MSR	7.10E-03	4.93E-03	-0.16	1.14	45.17	1.56E-10	-5.56	1.56E-06
Post-MSR	6.07E-03	4.32E-03	-0.94	5.26	1414.52	6.92E-308	-7.54	3.38E-11

8.2.2 Performances of the samples with number of PCs and variance explained

8.2.2.1 Full sample

	Full	RMSE	MSE	MAPE	MAE	R2	# PCs	Variance explained
XGBOOST	with	4.68E-02	2.19E-03	2.20E+02	3.46E-02	0.59	13	0.886
	without	4.63E-02	2.14E-03	2.36E+02	3.45E-02	0.60	7	0.875
NN	with	4.00E-02	1.60E-03	1.07E+03	3.02E-02	0.70	13	0.886
	without	4.00E-02	1.60E-03	2.86E+03	2.97E-02	0.70	7	0.875
SVR	with	4.75E-02	2.26E-03	1.68E+03	3.64E-02	0.57	13	0.886
	without	3.86E-02	1.49E-03	3.70E+03	2.84E-02	0.72	7	0.875
Naive Strategy	N/A	4.86E-02	2.36E-03	1.34E+03	3.51E-02	0.56	N/A	N/A

8.2.2.2 Pre-MSR

	Pre-MSR	RMSE	MSE	MAPE	MAE	R2	# PCs	Variance explained
XGBOOST	with	5.50E-02	3.03E-03	1.12E+02	3.89E-02	0.49	10	0.856
	without	5.18E-02	2.69E-03	1.11E+02	3.69E-02	0.55	7	0.906
NN	with	7.68E-02	5.90E-03	1.21E+02	5.66E-02	0.01	10	0.856
	without	7.68E-02	5.90E-03	1.21E+02	5.66E-02	0.01	10	0.856
SVR	with	7.23E-02	5.23E-03	1.13E+02	5.30E-02	0.13	10	0.856
	without	4.98E-02	2.48E-03	1.03E+02	3.54E-02	0.59	7	0.906
Naive Strategy	N/A	5.10E-02	2.60E-03	1.40E+02	3.60E-02	0.57	N/A	N/A

8.2.2.3 Post-MSR

	Post-MSR	RMSE	MSE	MAPE	MAE	R2	# PCs	Variance explained
XGBOOST	with	4.04E-02	1.63E-03	1.00E+03	3.29E-02	0.59	11	0.861
	without	4.01E-02	1.61E-03	1.58E+03	3.23E-02	0.60	7	0.895
NN	with	4.30E-02	1.85E-03	4.63E+03	3.43E-02	0.54	11	0.861
	without	4.89E-02	2.39E-03	1.48E+03	3.87E-02	0.41	7	0.895
SVR	with	5.91E-02	3.49E-03	8.57E+02	4.51E-02	0.13	11	0.861
	without	5.71E-02	3.26E-03	2.31E+03	4.39E-02	0.19	7	0.895
Naive Strategy	N/A	3.88E-02	1.50E-03	1.36E+03	3.12E-02	0.63	N/A	N/A

8.3 Monthly log return

8.3.1 *Moments of the distributions, Jarque-Bera and Augmented Dickey-Fuller tests*

	Mean	Variance	Skewness	Kurtosis	JB		ADF	
					Test value	P-value	Test value	P-value
Full sample	2.69E-02	1.63E-02	-0.58	1.09	195.26	3.98E-43	-6.96	9.46E-10
Pre-MSR	2.97E-02	1.98E-02	-0.50	0.42	36.81	1.02E-08	-3.80	2.92E-03
Post-MSR	2.49E-02	1.40E-02	-0.69	1.72	216.80	8.37E-48	-4.67	9.40E-05

8.3.2 *Performances of the samples with number of PCs and variance explained*

8.3.2.1 *Full sample*

	Full	RMSE	MSE	MAPE	MAE	R2	# PCs	Variance explained
XGBOOST	with	5.57E-02	3.10E-03	9.72E+01	4.06E-02	0.82	10	0.894
	without	4.93E-02	2.43E-03	9.56E+01	3.70E-02	0.86	4	0.892
NN	with	8.17E-02	6.67E-03	1.30E+02	6.31E-02	0.61	10	0.894
	without	6.33E-02	4.01E-03	1.27E+02	5.05E-02	0.76	4	0.892
SVR	with	1.19E-01	1.41E-02	1.89E+02	9.12E-02	0.17	10	0.894
	without	5.12E-02	2.62E-03	9.62E+01	3.87E-02	0.85	4	0.892
Naive Strategy	N/A	4.83E-02	2.34E-03	1.04E+02	3.62E-02	0.86	N/A	N/A

8.3.2.2 *Pre-MSR sample*

	Pre-MSR	RMSE	MSE	MAPE	MAE	R2	# PCs	Variance explained
XGBOOST	with	5.22E-02	2.72E-03	8.22E+01	3.81E-02	0.82	9	0.890
	without	5.35E-02	2.86E-03	8.12E+01	3.90E-02	0.81	5	0.933
NN	with	1.18E-01	1.39E-02	1.57E+02	9.36E-02	0.09	9	0.890
	without	6.08E-02	3.70E-03	8.25E+01	4.51E-02	0.76	5	0.933
SVR	with	1.02E-01	1.05E-02	1.26E+02	8.28E-02	0.31	9	0.890
	without	5.49E-02	3.02E-03	8.62E+01	4.05E-02	0.80	5	0.933
Naive Strategy	N/A	4.86E-02	2.36E-03	8.24E+01	3.52E-02	0.85	N/A	N/A

8.3.2.3 *Post-MSR sample*

	Post-MSR	RMSE	MSE	MAPE	MAE	R2	# PCs	Variance explained
XGBOOST	with	4.11E-02	1.69E-03	8.22E+01	3.24E-02	0.89	9	0.876
	without	7.99E-02	6.38E-03	1.06E+02	5.82E-02	0.57	5	0.918
NN	with	1.04E-01	1.08E-02	1.02E+02	8.30E-02	0.27	9	0.876
	without	5.77E-02	3.33E-03	8.86E+01	4.30E-02	0.78	5	0.918
SVR	with	5.33E-02	2.85E-03	1.00E+02	4.06E-02	0.81	9	0.876
	without	1.00E-01	1.00E-02	1.22E+02	7.28E-02	0.33	5	0.918
Naive Strategy	N/A	3.96E-02	1.57E-03	8.87E+01	3.16E-02	0.89	N/A	N/A

8.4 *95-99% variance explained*

8.4.1 *Weekly log returns*

8.4.1.1 *XGBOOST with 95% of variance explained*

		RMSE	MSE	MAPE	MAE	R2	# PCs	Variance explained
Full	with	4.95E-02	2.45E-03	7.71E+02	3.44E-02	0.54	22	0.950
	without	4.63E-02	2.14E-03	2.36E+02	3.45E-02	0.60	12	0.952
Pre-MSR	with	5.69E-02	3.24E-03	1.12E+02	4.02E-02	0.46	20	0.953
	without	5.18E-02	2.69E-03	1.11E+02	3.69E-02	0.55	11	0.951
Post-MSR	with	4.05E-02	1.64E-03	1.90E+03	3.22E-02	0.59	22	0.951
	without	4.18E-02	1.75E-03	1.37E+03	3.31E-02	0.56	12	0.952

8.4.1.2 XGBOOST with 99% of variance explained

		RMSE	MSE	MAPE	MAE	R2	# PCs	Variance explained
Full	with	4.95E-02	2.45E-03	7.71E+02	3.44E-02	0.54	40	0.991
	without	4.50E-02	2.02E-03	6.75E+02	3.30E-02	0.62	23	0.991
Pre-MSR	with	5.69E-02	3.24E-03	1.12E+02	4.02E-02	0.46	41	0.990
	without	5.18E-02	2.69E-03	1.11E+02	3.69E-02	0.55	26	0.991
Post-MSR	with	4.06E-02	1.65E-03	2.58E+03	3.23E-02	0.59	42	0.990
	without	3.95E-02	1.56E-03	1.92E+03	3.18E-02	0.61	26	0.991

8.4.2 Monthly log returns

8.4.2.1 XGBOOST with 95% of variance explained

		RMSE	MSE	MAPE	MAE	R2	# PCs	Variance explained
Full	with	4.99E-02	2.49E-03	9.53E+01	3.73E-02	0.85	17	0.950
	without	4.93E-02	2.43E-03	9.56E+01	3.70E-02	0.86	8	0.959
Pre-MSR	with	6.28E-02	3.94E-03	7.48E+01	4.43E-02	0.74	15	0.952
	without	4.97E-02	2.47E-03	8.10E+01	3.67E-02	0.84	7	0.956
Post-MSR	with	6.82E-02	4.66E-03	8.95E+01	5.00E-02	0.69	18	0.952
	without	7.99E-02	6.38E-03	1.06E+02	5.82E-02	0.57	8	0.953

8.4.2.1.1 XGBOOST with 99% of variance explained

		RMSE	MSE	MAPE	MAE	R2	# PCs	Variance explained
Full	with	4.99E-02	2.49E-03	9.53E+01	3.73E-02	0.85	36	0.990
	without	4.93E-02	2.43E-03	9.56E+01	3.70E-02	0.86	22	0.990
Pre-MSR	with	5.07E-02	2.57E-03	8.06E+01	3.75E-02	0.83	37	0.990
	without	4.97E-02	2.47E-03	8.10E+01	3.67E-02	0.84	24	0.991
Post-MSR	with	7.05E-02	4.97E-03	9.91E+01	5.28E-02	0.67	40	0.990
	without	4.09E-02	1.67E-03	8.24E+01	3.23E-02	0.89	26	0.991

8.4.2.1.2 Two-sided Diebold-Mariano test

		95% variance explained		99% variance explained	
		Diebold-Mariano		Diebold-Mariano	
		Test value	P-value	Test value	P-value
Full	with	3.41	0.00	3.41	0.00
	without	0.00	1.00	1.40	0.16
Pre-MSR	with	-2.76	0.01	1.61	0.11
	without	2.05	0.04	2.05	0.04
Post-MSR	with	-6.24	0.00	-6.84	0.00
	without	0.00	1.00	6.93	0.00