Forecasting of Corporate Revenues with Machine Learning Models versus Traditional Methods in the Digital Industry

João Ribeiro dos Santos Pattenden

152021005

Dissertation written under the supervision of Professor Pedro Afonso Fernandes

Dissertation submitted in partial fulfilment of requirements for the MSc in Business Analytics, at Católica-Lisbon School of Business & Economics

1^{st} June 2023

Forecasting of Corporate Revenues with Machine Learning Models versus Traditional Methods in the Digital Industry

João Ribeiro dos Santos Pattenden

Abstract

There has been a growing interest in the applicability of machine learning models in corporate sales and revenues forecasting. Past research has found promising results in this field, which show that these models might outperform more traditional methods. In this thesis, three real-world datasets containing information about the revenues and other features of Amazon, Microsoft and Netflix in the last two decades are investigated to forecast the revenues of these digital companies. Firstly, we apply different pre-processing techniques on the data, which include seasonal differencing using logarithm transformations. Then, some more classical time-series methods including Autoregressive model or order 1 are built. Moreover, different machine learning models including Partial Least Squares and Deep Neural network are applied. Finally, an empirical comparison of the models is performed using metrics such as Mean Absolute Error and Akaike Information Criterion. The results show that Autoregressive model of order 1 outperforms all the other models in terms of revenues forecasting accuracy in all datasets. Particularly, comparing with the benchmark machine learning model in each dataset, this method is able to reduce the error by more than 12 % and up to 72 %. Although these findings require further research to address any possible limitations, they provide insights on the performance of several models in revenues forecasting of digital firms, which can be a valuable tool for the decision-making process of businesses in this industry.

Keywords: Machine Learning; Sales and revenues forecasting; Digital companies; Pre-processing; Time Series; Empirical comparison; Accuracy;

Previsão de Receitas de Empresas com Modelos de Machine Learning versus Métodos Tradicionais na Indústria Digital

João Ribeiro dos Santos Pattenden

Resumo

Tem havido um interesse crescente no uso de modelos de machine learning para a previsão de vendas e receitas das empresas. Pesquisas recentes revelaram resultados promissores, que mostram que estes modelos podem superar métodos de previsão mais clássicos. Nesta tese, são investigadas três bases de dados que contêm informação sobre as receitas e outros atributos das empresas Amazon, Microsoft e Netflix nas últimas duas décadas, com o objetivo de prever as suas receitas. Começamos por aplicar diversas técnicas de pré-processamento dos dados, que incluem as diferenças homólogas de logaritmos. Posteriormente, são implementados alguns métodos mais clássicos de previsão de séries temporais como o modelo autorregressivo de ordem 1. São desenvolvidos também diferentes modelos de machine learning como o modelo dos mínimos quadrados parciais e redes neuronais. Por fim, é feita uma comparação dos modelos, utilizando diferentes métricas como o erro médio absoluto e o critério de informação de Akaike. Os resultados mostram que o modelo autorregressivo de ordem 1 tem a melhor performance na previsão das receitas nas três bases de dados. Comparando com o melhor modelo de machine learning em cada uma das bases de dados, este método consegue reduzir o erro em mais de 12 % e até 72 %. Embora estes resultados precisem de investigações adicionais, de modo a abordar possíveis limitações, dão-nos uma perceção geral sobre o desempenho de diversos modelos na previsão de receitas de empresas digitais, o que pode representar uma contribuição valiosa para a tomada de decisões dos negócios desta indústria.

Palavras-chave: Machine Learning; Previsão de vendas e receitas; Pré-processamento; Séries Temporais; Comparação; Empresas digitais;

Acknowledgments

I would like to thank my advisor Pedro Afonso Fernandes for his support and availability during the development of this thesis. His guidance was fundamental for the completion of this research.

I am also thankful to my family and friends for their encouragement and patiente, which helped me maintain my focus throughout this thesis.

Finally, I want to express my gratitude to all that have contributed, directly or indirectly, to the finalization of this study.

Contents

Acronyms

R^2 Coefficient of Determination. 5, 9

AIC Akaike Information Criterion. 24, 25, 27

AR1 Autoregressive model of order 1. 2, 20, 27–30

AR1MAX Autoregressive model of order 1 with multiple exogenous variables. 20, 28

ARIMA AutoRegressive Integrated Moving Average. 4–6, 8, 9

CNN Convolutional Neural network. 4, 8, 9

Conv-LSTM Convolutional Long short-term memory. 4, 5

D&SF Demand and Sales Forecasting. 1, 4

DNN Deep Neural network. 21, 23, 27

ETS Exponential Smoothing. 8, 9

FCL-Net Fusion Convolutional Long short-term memory network. 4, 5

GBT Gradient Boosted Tree. 5

GLM Generalized Linear Model. 5

k-NN k-nearest neighbours. 8

LASSO Least Absolute Shrinkage and Selection Operator. 21–23

LSTM Long short-term memory. 4, 8

M5P M5 Model trees. 8

MAE Mean Absolute Error. 5, 6, 8–10, 24, 27, 28

MSE Mean Squared Error. 24, 25, 27, 28

OLS Ordinary Least Squares. 21, 22

PLS Partial Least Squares. 21, 23, 27, 28

Q1 first quarter. 15, 16, 29

Q2 second quarter. 15

Q3 third quarter. 15

Q4 fourth quarter. 15

R&D research and development. 13, 16

RMSE Root Mean Squared Error. 5, 7–9, 24, 25, 27, 28

SCM Supply Chain Management. 1, 4

SVM Support Vector Machine. 8, 21, 22, 27

XGBoost eXtreme Gradient Boosting. 4, 6, 7

List of Figures

List of Tables

List of Equations

1 Introduction

1.1 Overview

In recent years, companies have been undergoing some major and disruptive changes in the market due to a higher level of uncertainty and volatility, which was caused by various events, namely the Covid-19 pandemic (Szczygielski et al., 2022). One of the areas that firms need to focus to address the challenges imposed by today's competitive business environment is Demand and Sales Forecasting (D&SF).

D&SF is a core area in businesses since it helps firms to balance supply and demand, which leads to a more efficient resource allocation. By anticipating the demand for their products and services, firms can improve their performance in several stages of the supply chain such as production planning, inventory planning, financial planning, marketing, and pricing (Nguyen et al., 2021). Being Supply Chain Management (SCM) a critical factor to leverage firm's success (Cadavid et al., 2018), accurate demand forecasts become vital for companies to succeed, meet their goals, and ensure their long-term impact and value.

For instance, accurate predictions can prevent firms from having excessive inventory or stock shortages, which would lead to excessive inventory costs, and to lost sales and consumer dissatisfaction, respectively (Huang et al., 2019). Hence, precise forecasts may help firms to manage more effectively their costs by optimizing their production schedules and inventory levels to match the expected demand from their customers.

Regarding financial planning, accurate sales predictions can help businesses to plan their budgets and set targets for their performance. At a marketing level, it might help monitor the competition, decide whether it is advantageous to enter a new market and develop products that align with costumer needs, which, consequently, contribute to increased customer satisfaction (Gupta, 2013). In general, sales and revenues forecasting helps firms to make knowledge-based and data-driven decisions, which can ultimately contribute to improved outcomes.

1.2 Motivation

Traditionally, firms have been using time series methods to forecast sales. However, this approach relies heavily on historical data and because of the recent market instability, which is causing major and structural changes in data patterns, there could be other models more suitable to make accurate predictions of future sales values.

Having said this, it becomes crucial for firms to invest in advanced analytics and automation to produce real time forecasts that reflect effective responses to constantly changing market conditions (Hürtgen et al., 2020).

The usage of machine learning models to forecast sales and revenues has emerged in recent years and several researchers have found promising results. Hence, there is a growing interest to test whether these methods can deliver more accurate and reliable forecasts than more traditional techniques.

1.3 Research question and hypotheses

This thesis aims to compare the performance between more traditional time series methods and machine learning models in revenues forecasting of the digital industry and determine which approach is better for this task. Specifically, the research question that this thesis seeks to answer is: **Can machine learning models improve the accuracy of time series methods such as Autoregressive model of order 1 (AR1) in forecasting the revenues of firms belonging to the digital industry?**

To answer this question, a comprehensive review of the literature on the methods used for sales and revenues forecasting will be conducted, highlighting the recent usage of machine learning techniques to approach this task and its comparison with more traditional time-series methods. Then, some real-world revenues datasets from the digital industry will be used to build traditional time-series and machine learning models. The out-of-sample performance of each model will then be evaluated based on different metrics. Finally, the strengths and limitations of each model will be discussed and, the model with the best performance will be selected.

Based on the above discussion and research question, two hypotheses about the methods used to forecast revenues in the digital industry will be proposed.

Hypothesis 1: The best model to forecast revenues in most of the datasets selected will be a machine learning model.

Hypothesis 2: There is not going to be a significant difference (less than 5 %) between machine learning models and time series methods like AR1 in terms of accuracy of revenues predictions in the digital industry.

These two hypotheses will be tested and verified at the end of this paper after conducting an in-depth analysis that compares the outcomes of the different models. The results of this study can provide valuable insights about the benefits and weaknesses of different models when forecasting revenues, which can lead businesses to make more informed decisions about their revenues forecasting strategies and therefore, maximize their overall efficiency.

2 Literature review

2.1 Overview

In this section, several papers that discuss the usage of machine learning techniques as an alternative solution for time series methods in sales and revenues forecasting will be examined. We will start looking at the emerging machine learning practices to forecast sales and revenues in a business context and the correspondent findings will be reviewed. This way, the literature review seeks to explore the current developments in sales and revenues forecasting research, which may be meaningful for businesses that intend to apply these methods in real-world environments.

2.2 Current trends in sales/revenues forecasting

To achieve success in today's competitive and challenging business environment, it is vital for businesses to focus on an effective SCM. Being D&SF an important key area in SCM, firms should adapt their supply chain to cutting-edge technologies to enhance their performance, reduce their costs and improve their services (Cadavid et al., 2018). In view of this, several applications of machine learning in SCM have been implemented, placing emphasis on the specific area of D&SF.

2.2.1 On-demand ride services

Our first example is the presentation of a new deep learning approach to predict the short-term passenger demand for on-demand ride services (Ke et al., 2017). The demand for these services is quite complex because many dependencies need to be considered simultaneously, namely spatial, temporal, and exogenous dependencies. A Fusion Convolutional Long short-term memory network (FCL-Net) was purposed to tackle the three interdependencies within a single end-to-end learning framework.

In this study, data from an on-demand ride service platform in China was used. Different models, including FCL-Net, were built to forecast the demand, namely the AutoRegressive Integrated Moving Average (ARIMA) model, eXtreme Gradient Boosting (XGBoost), Long short-term memory (LSTM), Convolutional Long short-term memory (Conv-LSTM) and Convolutional Neural network (CNN). By using different metrics to compare the performance of these models such as Root Mean Squared Error (RMSE), Coefficient of Determination (R^2) and Mean Absolute Error (MAE), it was possible to observe that the FCL-Net model had the best scores, outperforming significantly the more traditional approaches such as ARIMA and the other machine learning algorithms. Specifically, FCL-Net was able to reduce the RMSE of ARIMA by 53.6 %. Moreover, the authors highlighted the inclusion of exogenous variables like travel time rate, time-of-day, day-of-week, and weather conditions, in addition to the passenger demand itself, as a promising technique. This is evidenced by the fact that FCL-Net achieved a relatively 48.3 % lower RMSE than Conv-LSTM with only the historical demand intensity.

Overall, it was concluded that this new approach could give more accurate and reliable predictions, which might, ultimately, improve the efficiency of these services by contributing to the improvement of costumers' experience and satisfaction.

2.2.2 E-commerce – fashion store

Nowadays, businesses deal with a vast amount of data that is expected to grow further in an exponential manner (Cheriyan et al., 2018). As a result, more traditional forecasting methods may not be appropriate to deal with the high degree of complexity of the sales data. In fact, these authors proposed that the application of machine learning techniques could be a more reliable and efficient way of getting more accurate predictions considering the current trends.

They used a dataset containing three consecutive years of sales data from an e-fashion store to train and test different machine learning-models such as the Generalized Linear Model (GLM), Decision Tree (DT) and Gradient Boosted Tree (GBT). The performance of the three models was then compared, based on different metrics such as accuracy rate, error rate, precision, recall and Kappa. The results showed that GBT reached the highest accuracy rate (98%) and minimum error rate (2%), outperforming the other two models. In particular, GBT achieved a 34 % lower error rate than the GLM. Hence, the authors concluded that the usage of machine learning techniques could be part of an intelligent sales prediction system to handle a growing volume of data.

2.2.3 Retail industry – pharmacy retail sector

Other example that evaluates whether the usage of machine learning techniques could improve the accuracy and performance of predictive models for sales time series forecasting is a study conducted by Pavlyshenko (2019), which assessed the applicability of machine learning models for sales forecasting. To accomplish that, he used a sales dataset from Rossmann stores. The author began to present some techniques to pre-process the data, which included handling missing values, removing outliers, one-hot encoding (technique that converts categorical data into binary vectors) and feature scaling (process that normalizes the range of input features).

After pre-processing the data, he built several models such as Linear Regression, Random Forest, and XGBoost, and evaluated their performance using as metrics a relative MAE. In addition, he used a technique called stacking, which combined the predictions of the different machine-learning models into a final forecast.

For that purpose, he divided the data into training and validation sets. Then, he trained each individual model on the training set and used its predictions on the validation set as an input to an ensemble model, that was trained to predict the final sales forecast. This way, the strengths of the different models were combined into one, which ultimately outperformed all the individual models in terms of accuracy. The validation and out-of-sample errors of the technique Stacking were 1.2 % lower than ARIMA's.

Hence, the author concluded that the usage of machine learning models, particularly the combination of different ones into an ensemble model, i.e., stacking, was an efficient way to get more accurate and reliable sales predictions.

2.2.4 E-commerce – merchandise store

As we can verify, one can state that machine learning techniques show promise in forecasting sales accurately. Moreover, they might be a viable alternative for the traditional methods, since they can be used in many situations in which classic demand forecasting methods might not be suitable. In fact, the time series methods assume the availability of sales data for a certain time period. When businesses want to launch new products or offer new services, there is obviously no historical data available. However, firms still need to know in advance how they are going to design their products/services and which strategies will be implemented, bearing in mind the optimization of logistics, supply chain management and wholesale management (Smirnov and Sudakov, 2021).

This is even more important when the decisions involved entail a significant investment. Smirnov and Sudakov (2021) published a paper in which they proposed several machine learning methods to forecast new product demand, by using data about new product demand from the Ozon online store.

As input data of the algorithm, they used attributes like price, name, category and text description of the product. Although they could not use data about the history of each item's sales, they used historical data grouped by categories. Some examples were "average brand sales within a group of products in the first week of products in stock" and "the ratio of price to average price of products from the same subtype". The problem was treated as a regression task and the authors presented Gradient Boosting to solve it, namely XGBoost, Light Gradient Boosting Machine (LightGBM) and Categorical Boosting (CatBoost) packages. The metric standard RMSE was then selected to evaluate the performance of the different models in forecasting the sales for this new product.

The best accuracy achieved at testing data set was RMSE = 4.00129, but due to the lack of published articles on the same dataset, it was not possible to compare the results directly. Nevertheless, the authors concluded that machine learning models could be a valuable tool to forecast the demand for new products without any marketing research or historical data.

2.2.5 Publishing industry

Another example of an application of machine learning in a forecast of newly product demand was the study conducted to predict the sales of new published books in a real business environment (Castillo et al., 2017).

The goal of this work was to forecast the sales for new launched books in the Spanish market by considering several "pre-sales variables", which are basically variables that are under the control of the publisher before the books get released to the bookstores. The main intuition behind it was to provide publishers with a tool that helped them to decide how many prints they were going to do of each new book. The desired effect was that they did not print too many since it implied great losses of the investment or did not print enough because it could lead to lost sales and, potentially, customers.

Since there is no specific historical data, time-series were not suitable to approach this task. Instead, the authors proposed several machine learning methods to answer this question. They started to consider data about other previously published books and took, as a reference, a set of features related to the book publication process that are known before a new book is launched.

Several pre-processing techniques were applied to the data, including feature selection to discard less relevant features. Afterwards, several prediction models were obtained for the dataset and evaluated by means of metrics such as MAE, RMSE, Relative Absolute Error (RAE) and Root Relative Squared Error (RRSE). Among models like M5 Model trees (M5P), k-nearest neighbours (k-NN), Random Forest, Linear Regression and Support Vector Machine (SVM) for regression, the M5P model obtained the best results. For instance, the RMSE of M5P was 24.55 % lower than k-NN's. In fact, the authors highlighted that this model could learn and tackle regression tasks that had high dimensionality and categorical variables efficiently. Moreover, it generated representations that are comprehensible and interpretable for the decision-making process without the need of human intervention.

Therefore, the study provided valuable insights on how to use machine learning to predict sales for new products, since there is no historical data available in these cases. Hence, this can be an effective instrument that firms and businesses could incorporate in the decision-making process related to the launch of new products and services.

2.2.6 Retail industry – retail store

Although time-series forecasting methods are not suitable for some situations, even when you can apply them, research shows that machine learning methods may achieve more accurate predictions, as previously discussed in other examples.

Ensafi et al. (2022) conducted a study in which they applied several predictive models to forecast the sales of seasonal items, namely furniture, using a real-world dataset. The authors began to use some more classical time-series methods such as Seasonal Autoregressive Integrated Moving Average (SARIMA), Auto Regressive Moving Average (ARMA), ARIMA and Triple Exponential Smoothing (ETS). Then, more advanced methods were applied, including Prophet, LSTM and CNN.

Before building the models, some pre-processing techniques were applied to the dataset, such as time-series aggregation, which consisted of resampling the daily data into monthly frequency and using the average daily sales. The performance of all models was then compared based on different accuracy metrics like RMSE and Mean Absolute Percentage Error (MAPE). The results demonstrated that the Stacked LSTM method outperformed the other models. Actually, it was able to reduce the RMSE of ARIMA by 54.5 %.

The authors concluded that neural networks time-series forecasting are quite competitive when comparing with more traditional time-series methods. In addition, they found evidence that models such as CNN and Prophet might be suitable candidates to forecast sales accurately, especially for seasonal items.

2.2.7 Retail industry – Walmart

An older study conducted on a sales dataset from Walmart had already reached a similar conclusion (CATAL et al., 2019). The authors built several models to provide businesses and enterprises with a means to make accurate sales. These models included some machine learning methods such as Linear Regression, Bayesian Regression, Neural Network Regression, Decision Forest Regression and Boosted Tree Regression and also some more traditional time series techniques like Seasonal ARIMA, Non-Seasonal ARIMA, Seasonal ETS, Non -Seasonal ETS, Naive Method, Average Method and Drift Method.

The main goal was to make a comparative analysis between regression methods in machine learning and time-series techniques. To achieve this, several metrics were chosen to estimate the error of the predictions, namely MAE, RMSE and R^2.

The initial experiments done in this study were performed for only one store and one department of the Walmart Company since the application of time series analysis techniques was not suitable for the entire dataset. In this first approach, the Decision Forest Regression achieved the highest scores, outperforming the other models, including the more traditional time-series methods. Specifically, this method was able to reduce the MAE and RMSE of Seasonal ARIMA by 5.92 % and 5.77 %, respectively.

In a second approach, all the departments of Walmart firm were tested with only the regression methods. The best performer was the Boosted Decision Tree Regression, which attained a MAE of 1669.10, RMSE of 3696.59 and R^2 of 0.97. Comparing with Linear Regression, this approach decreased the MAE by 32.7 % and the RMSE by 15.31 %.

Overall, the paper provided a comprehensive benchmarking of several techniques, including time series and regression for sales forecasting.

2.2.8 Retail food sector – bakery chain

We can also look at a recent paper prepared by Huber and Stuckenschmidt (2020), in which the authors proposed the usage of a machine learning approach to forecast the daily demand for different product categories at a store level for a bakery chain. The authors placed an emphasis on calendric special days, namely public holidays, days either side of public holidays and other special events that affect consumer behaviour. In these days, the customer demand fluctuates significantly from the other days and therefore, they stated that a special attention should be given to these days in the demand forecasting process.

Several methods were built using a dataset containing the daily point of sales of a bakery chain. These included time series methods and machine learning models like Gradient-Boosted Decision Trees and Artificial Neural Networks.

The authors compared the performance of the different models by relying on metrics such as Mean Absolute Scaled Error (MASE) and MAE. It was demonstrated that machine learning methods gave more accurate predictions than base-line approaches such as time series models, for this scenario. By conducting a deeper examination, they concluded that machine learning models could achieve lower forecasting errors by detecting patterns from the data that the other models were unable to do. Specifically, the machine learning models showed substantial gains in their performance when forecasting to neighbouring days of public holidays. The error was reduced by more than 10 % and up to 20 % in these cases comparing with adjusted time series models.

Hence, the authors concluded that machine learning models could be a suitable alternative to state-of-the-art time series methods for daily demand forecasting in the retail industry, particularly in special days. This provides a useful framework for retailers that seek to enhance their demand forecasting capabilities to achieve better predictions, namely in calendric special days.

2.2.9 Summary

In view of the above arguments, one can state that the usage of machine learning to forecast sales has been emerging recently, which results from their applicability to different scenarios that sometimes more traditional methods like time series models might fail to handle.

After looking at the results from the different studies that were conducted lately, it is evident that machine learning techniques show potential in forecasting sales for several industries, namely transportation, retail, e-commerce, pharmacy, and publishing. Moreover, these techniques are capable of dealing with different situations such as large and complex data, lack of historical data in case of launching a new product or service, demand seasonality, demand on special days, among others.

Their comparison to time series methods (Table 1) suggest that machine learning models may achieve more accurate predictions than the former. However, it is important to note that in general authors had to use more sophisticated machine learning techniques in order to get considerable gains when comparing with more traditional approaches. Moreover, this might not apply to every situation or business scenario. Finally, one can state that the implementation of these techniques in firms can be costly, as it requires experts in the field, a substantial amount of time to pre-process the data and train the model, and computational power (Hürtgen et al., 2020). The interpretation of these models might also be challenging due to the lack of intuition and explanation of their predictions, which can limit their application in the decision-making process (Elshawi et al., 2019).

Models/Papers	On-demand ride services	Retail store	Walmart
FCL-Net	**0.016**	–	–
ARIMA	0.0345	282.50	–
Stacked LSTM	–	**128.51**	–
Decision Forest Regression	–	–	**3439.48**
SARIMA	–	235.58	3650.08

Table 1: Comparison of Root Mean Squared Error between time-series and machine learning models

Overall, this section highlighted that machine learning techniques might be a viable option to forecast sales by showing their superiority over traditional time series methods in certain scenarios. However, further research is needed to address whether the results depend on the business context as these methods require a substantial investment, and their results might be hard to interpret.

Hence, these models will be tested in different sales datasets from firms of the digital industry and their performance will be evaluated. The best model will then be selected, and it will be verified whether machine learning models can outperform the more traditional approaches in this specific setting.

It is important to highlight that several authors used tree models to predict sales values in their studies. However, these models assume independence between observations (Taddy, 2019). As we are dealing with time series, this choice might be questionable since we usually cannot assume that the observations are independent from each other (Cryer and Chan, 2008), i.e. what happened yesterday will likely influence what happens today and tomorrow. Because of that, this approach will not be considered in this thesis.

3 Methodology, methods and tools

3.1 Methodology

This thesis followed a quantitative approach. The research design was a comparative analysis between machine learning models and time series methods in revenues forecasting in the digital industry. To accomplish that, numerical data from digital companies' financial reports and available in public sources was collected and analysed by using different statistical and computer-based techniques.

3.2 Methods

3.2.1 Data collection and description

Data containing the information of the revenues of Amazon, Microsoft and Netflix was extracted from the respective financial reports available in Statista and Macro Trends[1]. The datasets used in this research contain quarterly data on firms' revenues ($\mathrm{REVENUE}$), research and development (R&D) expenses (RDEV) and annual data on marketing expenditure (MKTG) and number of employees (EMPL). The Netflix dataset includes an additional attribute, the quarterly number of subscribers ($\mathrm{SUBSCRIBERS}$). The datasets descriptions are as follows:

1. **Amazon Dataset:** this dataset contains quarterly and annual data on Amazon's revenues, R&D expenses, marketing expenditure and the number of employees from 2009 to 2022.

2. **Microsoft Dataset:** this dataset contains quarterly and annual data on Microsoft's revenues, R&D expenses, marketing expenditure and the number of employees from 2009 to 2022.

3. **Netflix Dataset:** this dataset contains quarterly and annual data on Netflix's revenues, R&D expenses, marketing expenditure, number of employees and number of subscribers from 2014 to 2022.

The datasets provide an overview of the companies' revenues performance over the past two decades (Figure 1) and enable the analysis of the impact of different features such as R&D (Figure 2), marketing expenses (Figure 3), and number of employees (Figure 4) on revenues.

[1] https://www.statista.com/ and https://www.macrotrends.net/

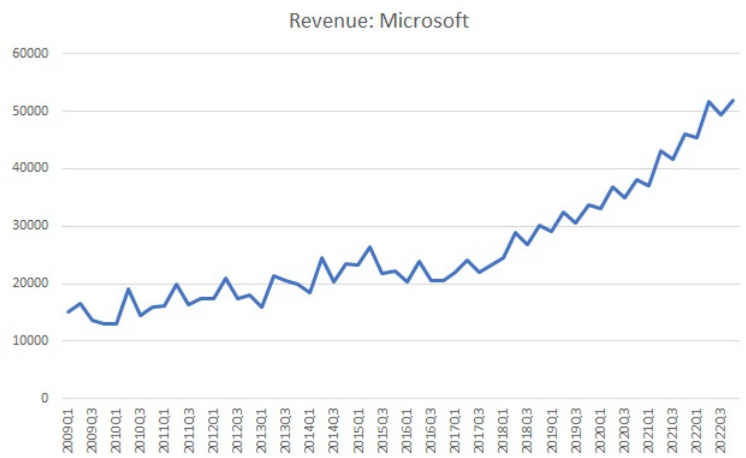

Figure 1: Microsoft's quarterly revenue between 2009 and 2022

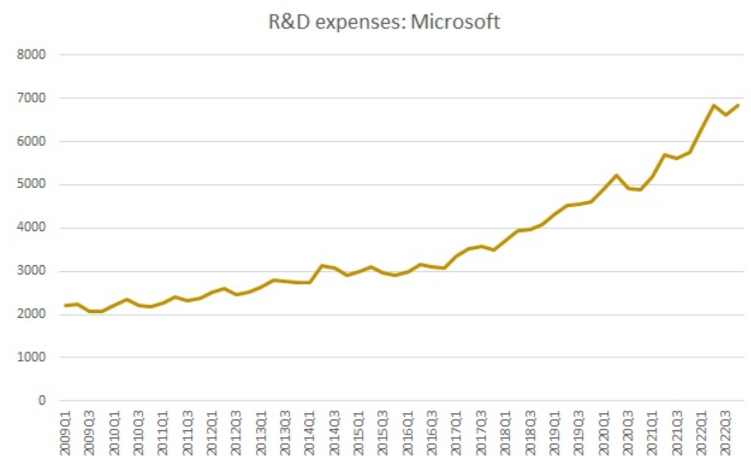

Figure 2: Microsoft's quarterly research and development expenses between 2009 and 2022

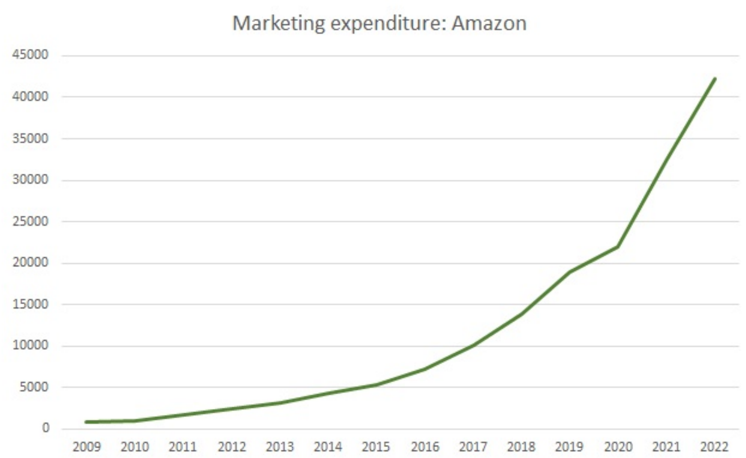

Figure 3: Amazon's annual marketing expenses between 2009 and 2022

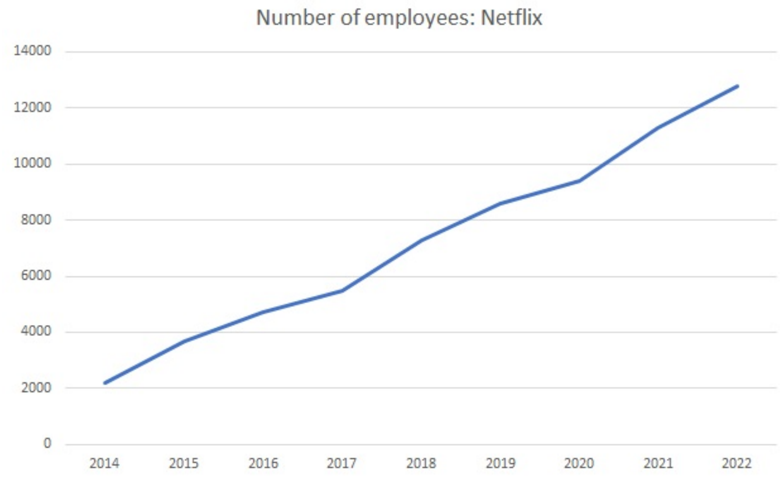

Figure 4: Netflix's annual number of employees between 2014 and 2022

3.2.2 Data preprocessing

To increase the number of data points, annual data of marketing expenditure and number of employees from the three firms was transformed into quarterly data by using a centered moving average of five periods with equal weights (MA5) and linear interpolation, respectively. In the case of the marketing expenditure, we started by dividing the annual data by 4 to have the quarterly average and then the following formula was applied:

$$\mathrm{MA}_5 = 0.2x_{t-2} + 0.2x_{t-1} + 0.2x_t + 0.2x_{t+1} + 0.2x_{t+2} \tag{1}$$

where t refers to the quarter for which the formula is being computed, $t-1$ and $t+1$ to the previous and following quarter, respectively.

It is important to note that to be able to use this formula on the values of the first two quarters of 2009, and the last two quarters of 2022, we took as reference values for 2008 third quarter (Q3) and fourth quarter (Q4) the result of the division of the annual value of 2009 by four. For the first quarter (Q1) and second quarter (Q2) of 2023, we assumed the variation between 2022 and 2023 to be the same as from 2021 to 2022 by applying the following formula:

$$mktg\_exp_{2023Q} = mktg\_exp_{2022Q4} + mktg\_exp_{2022Q1} - mktg\_exp_{2021Q4} \tag{2}$$

where $mktg\_exp$ refers to the marketing expenditure and $mktg\_exp_{2023Q}$ to the average quarterly marketing expenditure of 2023.

Regarding the number of employees, we used linear interpolation, as mentioned above, and started to compute the value for 2009 Q1:

$$employees_{2009Q1} = employees_{2008} + \frac{employees_{2009} - employees_{2008}}{4} \tag{3}$$

Then, for the next quarters we applied the following formula:

$$employees_{y,t} = employees_{y,t-1} + \frac{employees_{y} - employees_{y-1}}{4} \tag{4}$$

where y and t refer to the year and quarter for which the formula is being computed, respectively. $employees_y$ and $employees_{y-1}$ correspond to yearly data of the number of employees in periods y and $y - 1$

Afterwards, we evaluated whether the data had seasonality. It was possible to observe that Microsoft's revenues and R&D expenses (Figures 1 and 3) as well as Amazon's revenues (Figure 5) showed strong seasonal fluctuations over time. To solve this, we proceeded to the seasonal adjustment with TRAMO-SEATS, using gretl.

Figure 5: Amazon's quarterly revenue between 2009 and 2022

In addition, we checked whether the series were stationary, using the Dickey-Fuller test (Enders, 1995). This test checks the for the presence of a unit root (no stationarity) and there are different regression equations that can be used for this purpose. As the series of this thesis exhibited an increasing trend over time (Figures 1, 2, 3, 4, 5), we used the equation that included a drift and a linear time trend (Equation 5):

$$\Delta y_t = a_0 + \gamma y_{t-1} + a_2 t + \varepsilon_t \tag{5}$$

where Δy_t is the change in the value of y from one period to the next, a_0 is the constant term, γ is the influence of y previous period's value on the current period's change in y, t is the linear time trend and ε_t is the error term.

The null hypothesis of this test states that the y_t series contains a unit root in the series, which implies that the sequence is not stationary. We had to compare the p-values of the statistical test to check whether we rejected or failed to reject the null hypothesis. This procedure was done in gretl and the following p-values were estimated:

Series/Datasets	Amazon	Microsoft	Netflix
REVENUE	0.9542	0.9961	0.9495
RDEV	1	1	0.2012
MKTG	1	0.9994	0.9754
EMPL	1	0.995	0.791
SUBSCRIBERS	–	–	0.4151

Table 2: P-values estimated from the Dickey-Fuller test on the series of all datasets

As we can verify, all the p-values estimated from this test are greater than the conventional levels of statistical significance (0.1, 0.05 and 0.01), which implies that there is not sufficient evidence to reject the null hypothesis. Hence, we conclude that the series are considered non-stationary, which means that we have a trend that affects the behaviour of the series over time. This can make the process of forecasting more difficult and therefore, we attempted to make the data stationary by applying the seasonal difference of logarithms in all the variables (Equation 6):

$$x_{y,t} = 100 \ln(x_{y,t}) - 100 \ln(x_{y-1,t}) \tag{6}$$

where x refers to the variable for which the formula is being computed and y and t to the current year and quarter, respectively.

It is important to note that we opted to apply year-over-year differences to further address any seasonality that the data could have. The multiplication by 100 aimed at simplifying the interpretation in terms of growth rates. With this transformation, the Amazon and Microsoft observations from 2009 and the Netflix observations from 2014 were lost.

Despite our effort to remove the trend through the logarithm transformation, we reassessed the stationarity of the series by using the Dickey Fuller test with constant and trend to verify whether the former procedure was able to eliminate the trend from the series. The following p-values were estimated:

Series/Datasets	Amazon	Microsoft	Netflix
REVENUE	0.7456	0.0134	0.9505
RDEV	0.8679	0.4267	0.0444
MKTG	0.0652	0.6197	0.1371
EMPL	0.0048	0.3301	0.0257
SUBSCRIBERS	–	–	0.8537

Table 3: P-values estimated from the Dickey-Fuller test on the series of all datasets after the logarithm transformation

It is possible to see that although there was still not enough evidence to reject the null hypothesis in most series at the conventional levels of statistical significance, there was a general decrease in the p-values. This means that the evidence against the null hypothesis was increasing, which suggests that the series could be closer to be stationary. In addition, in some series such as EMPL from Amazon, REVENUE from Microsoft, RDEV and EMPL from Netflix, we could reject the null hypothesis at a 5 % level, for example.

The fact that we were still unable to reject the hypothesis of non-stationarity at the conventional levels of statistical significance in most cases after applying the logarithm transformation can be related to the abnormally high growth of the revenues and activities of the digital industry during the pandemic. For instance, revenues of Amazon increased significantly during this period and therefore, the series remains non-stationary even after the transformation is applied. In an attempt to make the series stationary, it was further applied first differences on the series, and many series continued to be non-stationary and therefore, we decided not to apply any further transformations besides the one from equation 6.

3.2.3 Cross Validation on time series

Cross-validation consists into randomly splitting the data into two subsets, a training set and a validation set (James et al., 2021). The model is trained with the observations of the training set and then the fitted model is used to predict the values for the validation set. This way, we are able to see how well the model generalizes to new data since its performance is evaluated on the validation set, which contains unseen data of the model (out-of-sample).

One approach to apply cross-validation is k-fold cross validation, which consists into randomly splitting the data into k subsets of approximately equal size. The model is trained on k-1 folds and the remaining fold (validation set) is used to evaluate the performance of the model. This process is repeated k times, and, in each time, a different group of observations (fold) is treated as a validation set. To estimate the model's performance, the average performance across all folds can be computed.

When dealing with time series, this method has some particularities. For instance, the splitting of the data into training and testing set must not be completely random, since it can break down the temporal relationship within the series (Chen et al., 2023). It does not seem reasonable to forecast the past using future-looking data. The temporal relationship within the series should be preserved in training and validation sets.

To address this issue, the rolling window technique was used. This method involves selecting a window of fixed size and then moving it across the time series (Figure 6). For each window, the models are trained using the training set and then applied to the test fold where the prediction metrics are computed (Bitencourt et al., 2021). The performance of each model is the average error value measured across all test folds.

The number of folds (k) selected in this thesis was 12, which is approximately 20 % of the observations. It is important to note that the Covid-19 pandemic outbreak comprised the years between 2020 and 2022, which are the ending years of our series and part of our test period.

The economic and social impact of this pandemic had no precedents in the history and because of that, this period might not be representative of the normal conditions in the market. For that reason, we tried to increase the number of folds from 12 to 16 so that the test period included more of the pre-pandemic period. As the models' performance did not vary significantly and we did not have many observations, we opted to maintain k = 12.

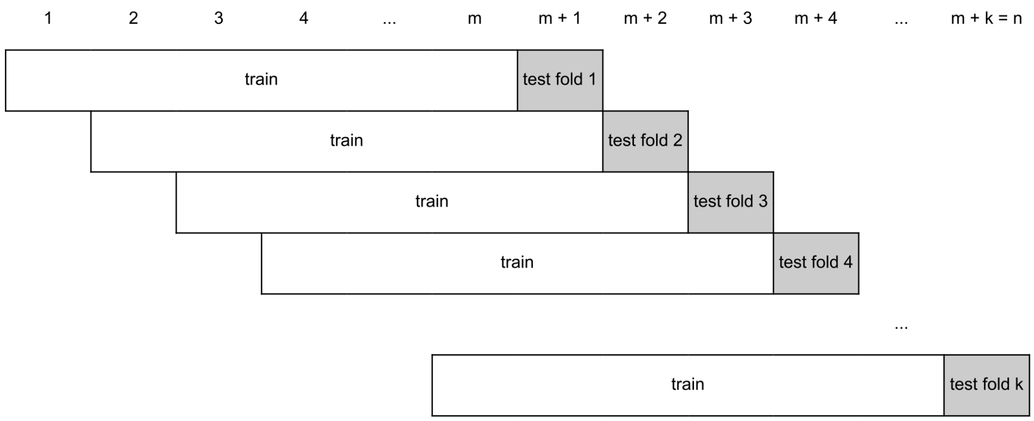

Figure 6: Out-Of-Sample cross-validation with time series (schematic representation)

3.2.4 Time series models

The first two models that have been produced were **AR1** and **Autoregressive model of order 1 with multiple exogenous variables (AR1MAX)**. The autoregressive models seek to explain the present value of a series by using its past values (Siegel and Wagner, 2022).

3.2.4.1 Autoregressive model of order 1

AR1 was the first model to be built. It uses the previous value of a series to predict its present value:

$$\hat{y}_t = \beta_0 + \beta_1 y_{t-1} + \varepsilon_t \tag{7}$$

where $\hat{y}_t$ is the value of the series in period t, β_0 is the interception term and β_1 is the autoregressive coefficient, which describes the relationship between y's current value and and its value in period $t-1$. $\varepsilon_t \sim WN(0, \sigma^2)$, i.e. ε_t is a white noise series in which each observation in period t is a random variable that follows a normal distribution with mean 0 and variance σ^2. Also, for each $s < t$, ε_t is uncorrelated with y_s.

3.2.4.2 Autoregressive model of order 1 with multiple exogenous variables

AR1MAX is an extension of AR1. In addition to the variable's past value, it considers the values of exogenous variables into the explanation of a series (Fung et al., 2003):

$$\hat{y}_t = \beta_0 + \beta_1 y_{t-1} + \sum_{i=1}^{r} \beta_i x_{ti} + \varepsilon_t \tag{8}$$

where $\hat{y}_t$ is the value of the series in period t, β_0 is the interception term, β_1 is the autoregressive coefficient, $\sum_{i=1}^{r} \beta_i x_{ti}$ is a sumproduct between the regression coefficients β_i and the exogenous variables x_{ti}, and ε_t is the random noise series.

3.2.5 Machine learning models

The machine learning models that were produced in this thesis were **Ordinary Least Squares (OLS)**, **Least Absolute Shrinkage and Selection Operator (LASSO)**, **SVM**, **Partial Least Squares (PLS)** and **Deep Neural network (DNN)**.

3.2.5.1 Ordinary Least Squares

OLS is a widely used technique for estimating coefficients of linear regression models, which describe the relationship between one or multiple predictors (independent variables) and a dependent variable (James et al., 2021). The values of the coefficients are unknown and must be estimated. This method uses the following equation:

$$Y_t = \beta_0 + \sum_{i=1}^{r} \beta_i x_{ti} + \epsilon_t \tag{9}$$

where Y_t is the value of the dependent variable in period t, β_0 is the interception term, $\sum_{i=1}^{r} \beta_i x_{ti}$ is the sumproduct between the regression coefficients β_i and the predictors x_{ti}, and ϵ_t refers to the error term at time t, which is unknown. β_i is interpreted as the average effect of a one unit increase in x_{ti} on Y_t, holding all other predictors fixed.

The regression parameters are estimated through the minimization of the sum of squared residuals (RSS), which correspond to the difference between the actual values (Y_t) and the predicted values ($\hat{Y}_t$) at period t obtained from the regression equation (9):

$$\min \text{RSS} = \min \sum_{t}^{j} (Y_t - \hat{Y}_t)^2 \tag{10}$$

3.2.5.2 Least Absolute Shrinkage and Selection Operator

LASSO is a regression technique used to estimate the coefficients that introduces the concept of regularization to enhance the forecasting accuracy. Particularly, it performs variable selection, which consists of selecting important predictors from a set of potential predictors (James et al., 2021). As a result, the models generated from the LASSO are usually less complex and easier to interpret. Contrary to OLS, LASSO seeks to minimize the following equation:

$$\min \mathrm{RSS} + \lambda \sum_{j=1}^{p} |\beta_j| \tag{11}$$

where RSS is the sum of squared residuals, λ is the tunning parameter and $\sum_{j=1}^{p} |\beta_j|$ is the sum of the absolute values of the coefficients.

As we can see, LASSO is an extension of OLS that imposes a penalty on the size of the coefficients (λ), which might force them to be zero in some cases.

3.2.5.3 Support Vector Machine

SVM is a machine learning approach that can be used for both regression and classification tasks. In our particular case, it will be used for a regression problem. Its main goal is to find a hyperplane in an N-dimensional space, where N is the number of attributes, whose distance between the hyperplane and the closest data points is maximized, while minimizing the prediction error (Awad and Khanna, 2015).

SVM maps the data to a higher feature space, using *kernels*. A *kernel* is a function that quantifies the similarity of two observations (James et al., 2021). With the usage of this function, the model can handle non-linear relationships between the input variables and the dependent variable, which makes it an effective tool for regression problems in which we have complex relationships between the predictors and the target variable. The function of a SVM is the following:

$$y_t = \alpha + \sum_{i=1}^{n} (\beta_i \cdot K(x_t, x_{it})) \tag{12}$$

where y_t is the target variable at period t, α is the intercept, β_i is the coefficient associated with each support vector x_i and $K(x, x_i)$ is the kernel function that measures the similarity between observations x and x_i at period t.

3.2.5.4 Partial Least Squares

PLS is a statistical technique that combines features from principal component analysis and multiple regression (Abdi, 2003). It aims to model the relationship between a dependent variable and one or more predictors. It is an useful tool to deal with multicolinearity among the explanatory variables.

Similarly to LASSO, this model also performs variable selection. Particularly, we regress the dependent variable y on to each dimension of x (predictors) independently and have the resulting regression coefficients (β). Then a factor v is created, which aggregates the first-order effect of each predictor x on y. The marginal regression is then performed:

$$y_i = \alpha + \beta * v_i + e_i \tag{13}$$

where α is the interception term and e_i represents the residuals.

PLS applies this regression successively to the residuals e_i in order to improve the forecast until it reaches a minimum of number of predictors and observations (Taddy, 2019). This process is called *boosting*.

3.2.5.5 Deep Neural Network

DNN is a machine learning algorithm that is part of the class of artificial neural networks (Liu et al., 2022). This model is capable of extracting features at different levels of abstraction, which allows it to detect and learn complex patterns in the data.

In this study, a feedforward neural network (FFNN) will be used. The architecture of this model consists of several neurons, which are logically arranged into two or more layers that interact with each other through weighted connections (Goh, 1995). Specifically, we have an input layer to which the data is presented and an output layer that holds the response to the given input. The intermediate layers (or hidden layers), enable these models to represent the associations between patterns.

This network maps the input data to predictions and then a loss function compares these predictions to the targets, producing a loss value. The optimizer then uses the loss value to update the network's weights in a process that is called *backpropagation*.

3.2.6 Measures of forecast error

Measures of forecast error indicate how effectively and accurately the models are making predictions and enable the comparison between them. In this thesis, the performance of the models will be evaluated by four widely used measures: **MAE**, **Mean Squared Error (MSE)**, **RMSE** and **Akaike Information Criterion (AIC)**. The following equations (14, 15, 16, 17, 18) were used to calculate these estimators where y_i is the actual value and $\hat{y}_i$ the predicted value for period i. The number of observations is represented by n.

3.2.6.1 Mean Absolute Error

To calculate the MAE, we computed the average of the absolute differences between the predicted values and the actual values:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{14}$$

The main advantage of this metric is that it is easy to compute and understand. In addition, the error value units match the predicted target value units (Schneider and Xhafa, 2022). However, for the comparison between series with different units, it is not advisable. Also, MAE gives the same weight to all the errors, so it is not suitable when we want to penalize outliers (Chai and Draxler, 2014).

3.2.6.2 Mean Squared Error

To calculate the MSE, we computed the average of the squared errors, i.e., the difference between the predicted and actual values.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{15}$$

This metric can be useful when we want to give a particular attention to outliers. It is widely used due to its relevance in statistical modelling. However, the fact that it heavily weights large errors comparing to small ones can constitute a disadvantage depending on the situation (Hyndman and Koehler, 2006). Also, the squaring leads to the loss of the unit.

3.2.6.3 Root Mean Squared Error

Following this, we can proceed to the calculation of RMSE in which we computed the square root of the average of the squared errors. In other words, RMSE was computed by taking the square root of MSE.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{16}$$

By taking the square root of MSE, we avoided the loss of the unit. Although it is sensitive to outliers, some authors such as Chai and Draxler (2014) recommended the usage of RMSE and acknowledged its relevance in measuring the performance of the models.

3.2.6.4 Akaike Information Criterion

Finally, we computed the AIC following the version proposed by Greene (2012):

$$\text{AIC} = ln\left(\frac{\epsilon'\epsilon}{k}\right) + \frac{2q}{k} \tag{17}$$

where ϵ is a column vector with the out-of-sample errors, k is the number of folds and q is the number of parameters of each forecasting model.

AIC is a metric that based on in-sample fit estimates a model's ability to predict the future values (Mohammed et al., 2015). It evaluates the quality of the statistical models by selecting the model whose balance between its complexity and the goodness of fit is best. Although the actual values of AIC do not have a real meaning, the best model is the one that has the lowest value. We can evaluate the performance of the remaining models by comparing them with the AIC value of the best model (Portet, 2020).

$$\Delta_i = AIC_i - \min_i AIC_i \tag{18}$$

where i corresponds to the model whose performance is being assessed and $\min_i AIC_i$ corresponds to the AIC of the best model.

The author defined Δ_i as the *information loss* when using model i rather than the best estimated model. Hence, the smaller the Δ_i, the more plausible is model i.

3.3 Tools

The tools used in this thesis included **Microsoft Excel**, **R Studio** and **gretl with TRAMO-SEATS**.

Excel was used to do several steps in the data preprocessing, namely the tranformation of the annual data into quarterly data, the detection of seasonality and stationarity, and the logarithm transformation. Also, it was used to do the exploratory data analysis, which included the creation of charts to get insights about the datasets. Gretl with TRAMO-SEATS was used to adjust the data seasonality of some variables. In addition, gretl was used to perform the Dickey-Fuller tests on the series of the various datasets.

Finally, R was used to build the time-series and machine learning models, which made the predictions for the revenues of the digital firms. In addition, it was used to evaluate the performance of the models in the various datasets. This allowed us to obtain some results that will be showed in the findings section.

4 Findings

It is possible to see that among all the models that were produced, AR1 achieved the best results in the Amazon dataset in all the metrics selected. Among the machine learning models, we highlight DNN, which achieved the lowest errors of MSE, RMSE and AIC. Comparing with this benchmark, AR1 was able to reduce the MSE and RMSE by 61 % and 38 %, respectively. In addition, the *information loss* when using DNN rather than AR1 according to AIC is 1.273. When looking at MAE, the SVM was the best performer among the machine learning models. Nonetheless, AR1 could reduce SVM's MAE by 43 %.

Models/Metrics	MAE	MSE	RMSE	AIC
AR1	**5.038**	**36.94**	**6.078**	**3.776**
AR1MAX	5.243	36.593	6.049	4.267
OLS	9.02	99.087	9.954	5.096
LASSO	8.925	98.112	9.905	5.086
SVM	8.831	102.492	10.124	5.13
PLS	8.362	94.619	9.727	5.05
DNN	8.705	94.574	9.725	5.049
Median	8.621	92.442	9.615	5.027

Table 4: Models performance on the Amazon dataset between 2010 first quarter and 2022 fourth quarter

In the Microsoft dataset, AR1 achieved the lowest errors in all the metrics. Among the machine learning models, PLS achieved the best scores. When comparing these two models, we verify that AR1 was able to reduce PLS's MAE, MSE and RMSE by 13 %, 50 % and 29 %, respectively. Also, the *information loss* when using PLS rather than AR1 is 1.02.

Models/Metrics	MAE	MSE	RMSE	AIC
AR1	**2.569**	**9.097**	**3.016**	**2.375**
AR1MAX	3.678	21.105	4.594	3.716
OLS	3.047	18.76	4.331	3.432
LASSO	3.045	18.594	4.312	3.423
SVM	3.473	20.217	4.496	3.507
PLS	2.937	18.075	4.251	3.395
DNN	6.745	60.518	7.779	4.603
Median	3.029	18.551	4.307	3.421

Table 5: Models performance on the Microsoft dataset between 2010 first quarter and 2022 fourth quarter

In the Netflix dataset, AR1 was again the model that achieved the best results in all the metrics. In particular, it was able to reduce the PLS' MAE, MSE and RMSE by 56 %, 72 % and 47 %, respectively. Furthermore, when using the benchmark machine learning model PLS, which achieved the lowest errors among this type of models, we got an *information loss* of 2.022.

Models/Metrics	MAE	MSE	RMSE	AIC
AR1	**2.697**	**11.063**	**3.326**	**2.654**
AR1MAX	3.695	15.744	3.968	4.006
OLS	6.316	43.556	6.6	4.774
LASSO	6.395	44.671	6.684	4.799
SVM	7.115	54.025	7.35	4.989
PLS	6.063	39.48	6.283	4.676
DNN	7.581	63.685	7.98	5.154
Median	6.305	43.376	6.586	4.77

Table 6: Models performance on the Netflix dataset between 2015 first quarter and 2022 fourth quarter

It is important to note that the addition of exogenous variables to AR1 (AR1MAX) led to higher errors in the three datasets, which may suggest that the series are relatively well explained by their past values. Hence, we concluded that AR1 was the best model to forecast the revenues of the three digital companies, outperforming all the others and therefore, evidence to support our hypotheses was not found.

5 Discussion

The goal of this research was to compare the performance of more traditional time series methods and machine learning models in forecasting the revenues of the digital industry, namely Amazon, Microsoft and Netflix. Based on previous research and current trends in sales/revenues forecasting, we expected that machine learning models would perform better than time series due to their ability to deal with complex data and non-linear relationships. However, our results showed that the best model in all datasets was AR1, which was able to reduce the errors of machine learning models by more than 12 % and up to 72 %. As a result, evidence to support our two initial hypotheses was not found.

These unexpected findings could have been caused by several limitations of the study conducted. One possible limitation was the limited amount of data, especially in the case of Netflix, in which we only have 32 observations. Also, we have small datasets for Amazon and Microsoft with 54 observations each. This might have prevented machine learning models to detect patterns and complex relationships in the series since they need a large amount of data for this purpose, especially, in the case of neural networks.

Furthermore, the attributes selected to explain the evolution of the series may not have been the most relevant. Besides that, our analysis could have excluded other indicators that are more important for the explanation of revenues' performance in the digital industry.

It is important to note that the data for this thesis has been extracted from public sources and therefore, the amount of data and attributes for the selected time periods was limited. The data collection of corporate information is usually a challenging task because most firms do not disclose their financial information due to competition, for example. Also, they have strict privacy protocols that can prevent them from publishing information about the company or their customers. For instance, the addition of data on customer behaviour or firm's internal operations into our models could have improved their performance, namely in the case of machine learning models.

Other possible limitation was the inclusion of the Covid-19 pandemic period in this research (between the Q1 of 2020 and the end of 2022), which may have had a significant impact on the revenues performance of the three firms. As mentioned earlier, the pandemic increased the uncertainty and volatility in the market and its significant economic impact on society might have disrupted the data patterns in the digital industry, which our models may not have been able to capture.

Finally, we saw that the quarterly data of revenues of these three firms exhibited a strong temporal component with seasonality and an increasing trend. Although several tools have been used in this research to deal with the trend and seasonality of the series, it could be the case that there were still traces of seasonality and non-stationarity in the data, which could have affected the machine learning models performance by preventing them to capture these temporal dependencies.

This could also be the reason for the better performance of time series models since these are specifically designed for predicting time series. That is why these models keep being widely used in several industries despite the emergence of machine learning models in revenues and sales forecasting. On the other hand, machine learning models are not specifically designed for time series forecasting and although they have been showing promise in forecasting sales and revenues of several industries, there is a need of further research to address whether they can capture temporal patterns in an effective way. In addition, it is essential to recall that this thesis did not use very sophisticated machine learning techniques, which we had already seen in previous research that it was an important factor to achieve substantial gains when using these models.

Despite these limitations, this study provides valuable insights on the efficiency of time series and machine learning models in forecasting the revenues of digital companies. Relating to our research question, our findings suggest that time series models are more suitable for the task than machine learning models. In other words, it does not seem that machine learning models are able to improve the accuracy of time series methods such as AR1 in the digital industry. However, these results cannot be generalized considering the limitations of this research. Further research is needed to verify whether machine learning models can outperform time series methods in other contexts.

6 Conclusion and further developments

As mentioned before, our study concluded that time-series methods seemed to be more suitable for forecasting the revenues of digital firms, which was a surprising result considering the emergence of machine learning models in this type of tasks in recent years. However, as stated in the discussion section, several limitations could have led to these findings and because of that there is a need to extend this study to address them.

Firstly, it is important that further research explores the impact of more attributes on the revenues of these companies such as customer demographics or industry trends. In other words, the impact of other variables on the revenues performance of this industry should be investigated so as to develop more sophisticated predictive models. It could also be interesting to examine whether macroeconomic indicators such as interest rate have an impact on the revenues of digital companies. In addition, it should be further addressed whether the variables included in this study are the most relevant for the explanation of revenues. This would allow a more comprehensive understanding of the factors that drive the revenues in this industry.

Another aspect that can be further examined is the impact of Covid-19 on revenue forecasting, namely in the digital industry. It could be the case that this has disrupted the data patterns in this industry and therefore, future studies can try to incorporate external factors that somehow measure this impact. Also, the test period of the models can be extended so as to reduce the impact of Covid-19 on revenues forecasting.

Finally, the inclusion of more machine learning models like ensemble, stacking or boosting models could be interesting to test whether their performance can improve the accuracy of time series models in this specific context. Moreover, more time series models can be built in the future to provide a fair comparison. This way, we would have a diverse range of forecasting models, which would enrich this research. The interpretability of these models can also be addressed as it is a crucial point in real-world scenarios. Potentially, future applications of this thesis can constitute a valuable tool for firms in this industry to get more accurate results in their revenues prediction and therefore, make more informed data-driven decisions.

References

H. Abdi. Partial least square regression (pls regression). *Encyclopedia for research methods for the social sciences*, 6(4):792–795, 2003.

M. Awad and R. Khanna. *Support Vector Regression*, pages 67–80. Apress, Berkeley, CA, 2015. ISBN 978-1-4302-5990-9. doi: 10.1007/978-1-4302-5990-9_4. URL https://doi.org/10.1007/978-1-4302-5990-9_4.

H. V. Bitencourt, L. A. F. de Souza, M. C. dos Santos, P. C. de Lima e Silva, and F. G. Guimarães. Combining embeddings and fuzzy time series for high-dimensional time series forecasting in internet of energy applications. 12 2021. URL http://arxiv.org/abs/2112.02140.

J. P. U. Cadavid, S. Lamouri, and B. Grabot. Trends in machine learning applied to demand sales forecasting: A review, 7 2018. URL https://hal.science/hal-01881362.

P. A. Castillo, A. M. Mora, H. Faris, J. J. Merelo, P. García-Sánchez, A. J. Fernández-Ares, P. D. las Cuevas, and M. I. García-Arenas. Applying computational intelligence methods for predicting the sales of newly published books in a real editorial business management environment. *Knowledge-Based Systems*, 115:133–151, 1 2017. ISSN 09507051. doi: 10.1016/j.knosys.2016.10.019. URL https://www.sciencedirect.com/science/article/pii/S0950705116304026.

C. CATAL, K. ECE, B. Arslan, and A. Akbulut. Benchmarking of regression algorithms and time series analysis techniques for sales forecasting. *Balkan Journal of Electrical and Computer Engineering*, pages 20–26, 1 2019. ISSN 2147-284X. doi: 10.17694/bajece.494920. URL https://www.researchgate.net/publication/330762801_Benchmarking_of_Regression_Algorithms_and_Time_Series_Analysis_Techniques_for_Sales_Forecasting.

T. Chai and R. R. Draxler. Root mean square error (rmse) or mean absolute error (mae)? *Geosci. Model Dev. Discuss*, 7:1525–1534, 2014. doi: 10.5194/gmdd-7-1525-2014. URL www.geosci-model-dev-discuss.net/7/1525/2014/.

W. J. Chen, J. J. Yao, and Y. H. Shao. Volatility forecasting using deep neural network with time-series feature embedding. *Economic Research-Ekonomska Istrazivanja*, 36:1377–1401, 2023. ISSN 1331677X. doi: 10.1080/1331677X.2022.2089192. URL https://www.researchgate.net/publication/358045499_Neural_Network-Based_Financial_Volatility_Forecasting_A_Systematic_Review.

S. Cheriyan, S. Ibrahim, S. Mohanan, and S. Treesa. Intelligent sales prediction using machine learning techniques. *IEEE*, pages 1–6, 8 2018. URL https://ieeexplore.ieee.org/document/8659115.

J. D. Cryer and K.-S. Chan. *Time Series Analysis With Applications in R*. Springer Science+Business Media, LLC, second edition edition, 1 2008. ISBN 978-0-387-75958-6. URL https://link.springer.com/book/10.1007/978-0-387-75959-3.

R. Elshawi, M. H. Al-Mallah, and S. Sakr. On the interpretability of machine learning-based model for predicting hypertension. *BMC Medical Informatics and Decision Making*, 19, 7 2019. ISSN 14726947. doi: 10.1186/s12911-019-0874-0. URL https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-019-0874-0.

W. Enders. *Applied Econometric Time Series*. John Wiley Sons, New York, 4th edition, 1995. URL https://new.mmf.lnu.edu.ua/wp-content/uploads/2018/03/enders_applied_econometric_time_series.pdf.

Y. Ensafi, S. H. Amin, G. Zhang, and B. Shah. Time-series forecasting of seasonal items sales using machine learning – a comparative analysis. *International Journal of Information Management Data Insights*, 2, 4 2022. ISSN 26670968. doi: 10.1016/j.jjimei.2022.100058. URL https://www.sciencedirect.com/science/article/pii/S2667096822000027.

E. H. Fung, Y. K. Wong, H. F. Ho, and M. P. Mignolet. Modelling and prediction of machining errors using armax and narmax structures. *Applied Mathematical Modelling*, 27:611–627, 2003. ISSN 0307904X. doi: 10.1016/S0307-904X(03)00071-4. URL https://www.sciencedirect.com/science/article/pii/S0307904X03000714?ref=cra_js_challenge&fr=RR-1.

A. T. C. Goh. Back-propagation neural networks for modeling complex systems, 1995.

W. H. Greene. *Econometric Analysis*. Prentice Hall, Upper Saddle River, New Jersey, 7th edition, 2012. URL https://www.researchgate.net/file.PostFileLoader.html?id=568181165cd9e37af18b458f&assetKey=AS:311705391828994@1451327765378.

A. Gupta. Sales forecasting market potential: Best practices in india. *International Journal of Advanced Marketing and Research*, 1:1–7, 9 2013. URL http://management.cloud-journals.com/index.php/IJAMR/article/view/Mgmt-25.

T. Huang, R. Fildes, and D. Soopramanien. Forecasting retailer product sales in the presence of structural change. *European Journal of Operational Research*, 279:459–470, 12 2019. ISSN 03772217. doi: 10.1016/j.ejor.2019.06.011. URL https://www.sciencedirect.com/science/article/pii/S0377221719304862.

J. Huber and H. Stuckenschmidt. Daily retail demand forecasting using machine learning with emphasis on calendric special days. *International Journal of Forecasting*, 36:1420–1438, 10 2020. ISSN 01692070. doi: 10.1016/j.ijforecast.2020.02.005. URL https://www.sciencedirect.com/science/article/pii/S0169207020300224.

R. J. Hyndman and A. B. Koehler. Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22:679–688, 10 2006. ISSN 01692070. doi: 10.1016/j.ijforecast.2006.03.001. URL https://www.sciencedirect.com/science/article/pii/S0169207006000239.

H. Hürtgen, F. Plaschke, K. Sauer-Sidor, and N. Wittmann. Predictive sales forecasting: Is your finance function up to code? *McKinsey Company*, pages 1–5, 8 2020. URL https://www.mckinsey.com/capabilities/strategy-and-corporate-finance/our-insights/predictive-sales-forecasting-is-your-finance-function-up-to-code.

G. James, D. Witten, T. Hastie, and R. Tibshirani. *An Introduction to Statistical Learning with Applications in R*. Springer, 2nd edition, 2021. URL https://www.statlearning.com/.

J. Ke, H. Zheng, H. Yang, and X. M. Chen. Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach. *Transportation Research Part C: Emerging Technologies*, 85:591–608, 12 2017. ISSN 0968090X. doi: 10.1016/j.trc.2017.10.016. URL https://www.sciencedirect.com/science/article/pii/S0968090X17302899.

X.-Y. Liu, Y. Fang, L. Yang, Z. Li, and A. Walid. Chapter 9 - high-performance tensor decompositions for compressing and accelerating deep neural networks. In Y. Liu, editor, *Tensors for Data Processing*, pages 293–340. Academic Press, 2022. ISBN 978-0-12-824447-0. doi: https://doi.org/10.1016/B978-0-12-824447-0.00015-7. URL https://www.sciencedirect.com/science/article/pii/B9780128244470000157.

E. A. Mohammed, C. Naugler, and B. H. Far. Emerging business intelligence framework for a clinical laboratory through big data analytics. *Emerging Trends in Computational Biology, Bioinformatics, and Systems Biology: Algorithms and Software Tools*, pages 577–602, 1 2015. doi: 10.1016/B978-0-12-802508-6.00032-6. URL https://www.researchgate.net/publication/283551999_Emerging_Business_Intelligence_Framework_for_a_Clinical_Laboratory_Through_Big_Data_Analytics.

H. D. Nguyen, K. P. Tran, S. Thomassey, and M. Hamad. Forecasting and anomaly detection approaches using lstm and lstm autoencoder techniques with the applications in supply chain management. *International Journal of Information Management*, 57, 4 2021. ISSN 02684012. doi: 10.1016/j.ijinfomgt.2020.102282. URL https://www.sciencedirect.com/science/article/pii/S026840122031481X.

B. M. Pavlyshenko. Machine-learning models for sales time series forecasting. *data*, 4:1–11, 1 2019. URL https://www.researchgate.net/publication/330484523_Machine-Learning_Models_for_Sales_Time_Series_Forecasting.

S. Portet. A primer on model selection using the akaike information criterion. *Infectious Disease Modelling*, 5:111–128, 1 2020. ISSN 24680427. doi: 10.1016/j.idm.2019.12.010. URL https://www.sciencedirect.com/science/article/pii/S2468042719300508.

P. Schneider and F. Xhafa. Anomaly detection: Concepts and methods. *Anomaly Detection and Complex Event Processing over IoT Data Streams*, pages 49–66, 1 2022. doi: 10.1016/B978-0-12-823818-9.00013-4. URL https://linkinghub.elsevier.com/retrieve/pii/B9780128238189000134.

A. F. Siegel and M. R. Wagner. Time series: Understanding changes over time. *Practical Business Statistics*, pages 445–482, 1 2022. doi: 10.1016/B978-0-12-820025-4.00014-2. URL https://linkinghub.elsevier.com/retrieve/pii/B9780128200254000142.

P. S. Smirnov and V. A. Sudakov. Forecasting new product demand using machine learning. volume 1925. IOP Publishing Ltd, 6 2021. doi: 10.1088/1742-6596/1925/1/012033. URL https://iopscience.iop.org/article/10.1088/1742-6596/1925/1/012033.

J. J. Szczygielski, A. Charteris, P. R. Bwanya, and J. Brzeszczyński. The impact and role of covid-19 uncertainty: A global industry analysis. *International Review of Financial Analysis*, 80, 3 2022. ISSN 10575219. doi: 10.1016/j.irfa.2021.101837. URL https://www.sciencedirect.com/science/article/pii/S1057521921001708.

M. Taddy. *Business Data Science: Combining Machine Learning and Economics to Optimize, Automate, and Accelerate Business Decisions*. McGraw-Hill, 1st edition, 8 2019. URL https://taddylab.com/.

Appendix

Main code used in the thesis

```
#Generic functions


# cv.AR1(y, k)
cv.AR1 <- function(y, k=4){

  y <- as.matrix(y)

  n <- length(y)
  m <- n - k

  pred  <- rep(0,k)
  error <- rep(0,k)

  for (i in 1:k){

    ytrain <- y[i:(m+i-1),1]

    pred[i]  <- AR1(ytrain)

    error[i] <- y[m+i] - pred[i]

  }

  mae <- sum(abs(error))/k
  mse <- sum(error*error)/k
  aic <- log(sum(error*error)/k) + 2/k
```

```
  list("prediction" = pred, "MAE" = mae, "MSE" = mse, "AIC" =
     aic)

}

# cv.AR1MAX(x, y, k)
cv.AR1MAX <- function(x, y, k=4){

  y <- as.matrix(y)
  x <- as.matrix(x)

  n <- length(y)
  m <- n - k

  pred  <- rep(0,k)
  error <- rep(0,k)

  for (i in 1:k){

    ytrain <- y[i:(m+i-1),1]
    xtrain <- x[i:(m+i-1),]

    pred[i]  <- AR1MAX(xtrain, ytrain)

    error[i] <- y[m+i] - pred[i]

  }

  mae <- sum(abs(error))/k
  mse <- sum(error*error)/k

  q <- ncol(x) + 1
```

```
  aic <- log(sum(error*error)/k) + 2*q/k

  list("prediction" = pred, "MAE" = mae, "MSE" = mse, "AIC" =
     aic)

}

# cv.ols(x,y,k)
cv.ols <- function(x,y,k=4){

  n <- length(y)
  m <- n - k

  y <- as.matrix(y)
  x <- as.matrix(x)

  pred  <- rep(0,k)
  error <- rep(0,k)

  for (i in 1:k){

    ytrain <- y[i:(m+i-1),1]
    xtrain <- x[i:(m+i-1),]
    xtest  <- x[m+i,]

    mod <- lm(ytrain ~ xtrain)

    pred[i]  <- xtest %*% coefficients(mod)[-1] + coefficients(
       mod)[1]

    error[i] <- y[m+i] - pred[i]
```

```
  }

  mae <- sum(abs(error))/k
  mse <- sum(error*error)/k

  q <- ncol(x)

  aic <- log(sum(error*error)/k) + 2*q/k

  list("prediction" = pred, "MAE" = mae, "MSE" = mse, "AIC" =
    aic)

}

# cv.lasso(x,y,k)
cv.lasso <- function(x,y,k=4){

  n <- length(y)
  m <- n - k

  y <- as.matrix(y)
  x <- as.matrix(x)

  pred  <- rep(0,k)
  error <- rep(0,k)

  for (i in 1:k){

    ytrain <- y[i:(m+i-1),1]
    xtrain <- x[i:(m+i-1),]
    xtest  <- x[m+i,]
```

```
    mod <- gamlr(xtrain, ytrain, gamma=0)

    pred[i]  <- predict(mod, newdata=t(as.matrix(xtest)))

    error[i] <- y[m+i] - pred[i]

  }

  mae <- sum(abs(error))/k
  mse <- sum(error*error)/k

  q <- ncol(x)

  aic <- log(sum(error*error)/k) + 2*q/k

  list("prediction" = pred, "MAE" = mae, "MSE" = mse, "AIC" =
     aic)

}

# cv.svm(x,y,k)
cv.svm <- function(x,y,k=4){

  n <- length(y)
  m <- n - k

  y <- as.matrix(y)
  x <- as.matrix(x)

  pred  <- rep(0,k)
  error <- rep(0,k)
```

```
  for (i in 1:k){

    ytrain <- y[i:(m+i-1),1]
    xtrain <- x[i:(m+i-1),]
    xtest  <- x[m+i,]

    mod <- ksvm(ytrain ~ xtrain, kernel ="polydot")

    pred[i]  <- predict(mod, newdata=t(as.matrix(xtest)))

    error[i] <- y[m+i] - pred[i]

  }

  mae <- sum(abs(error))/k
  mse <- sum(error*error)/k

  q <- ncol(x)

  aic <- log(sum(error*error)/k) + 2*q/k

  list("prediction" = pred, "MAE" = mae, "MSE" = mse, "AIC" =
     aic)

}

# cv.pls(x,y,k)
cv.pls <- function(x,y,k=4){

  n <- length(y)
  m <- n - k
```

```
y <- as.matrix(y)
x <- as.matrix(x)

q <- ncol(x)

pred  <- rep(0,k)
error <- rep(0,k)

mae_aux <- 1000000000

for (d in 1:5){

  for (i in 1:k){

      ytrain <- y[i:(m+i-1),1]
      xtrain <- x[i:(m+i-1),]
      xtest  <- x[m+i,]

      mod <- pls(xtrain, ytrain, K=d)

      pred[i]  <- predict(mod, newdata=xtest)

      error[i] <- y[m+i] - pred[i]

  }

  mae <- sum(abs(error))/k
  mse <- sum(error*error)/k

  aic <- log(sum(error*error)/k) + 2*q/k
```

```
    if (mae <= mae_aux){

        pred_aux <- pred
        mae_aux  <- mae
        mse_aux  <- mse
        aic_aux  <- aic
        d_aux    <- d
    }
  }

  list("prediction" = pred_aux, "MAE" = mae_aux, "MSE" = mse_
    aux, "AIC" = aic_aux, "directions" = d_aux)

}

# cv.dnn(y,df,k)
cv.dnn <- function(response,df,k=4){

  n <- nrow(df)
  m <- n - k

  pred  <- rep(0,k)
  error <- rep(0,k)

  for (i in 1:k){

    dtrain <- as.h2o(df[i:(m+i-1),])
    dtest  <- as.h2o(df[m+i,])

    mod <- h2o.deeplearning(y = response, training_frame =
        dtrain)
```

```r
    pred[i]  <- as.vector(h2o.predict(mod, newdata = dtest))

    error[i] <- y[m+i] - pred[i]

  }

  mae <- sum(abs(error))/k
  mse <- sum(error*error)/k

  q <- ncol(df) - 1

  aic <- log(sum(error*error)/k) + 2*q/k

  list("prediction" = pred, "MAE" = mae, "MSE" = mse, "AIC" =
     aic)

}

#SAMPLE CODE ABOUT FORECASTING WITH MACHINE LEARNING: AMAZON
   DATASET

# Load the small library of machine learning methods for time
   series
source("librec.R")
source("libmlts.R") #Includes the generic functions

# Quarterly data
data <- read.csv("amazon1.csv")
n <- nrow(data) # Total number of records
dt <- data[,-1]   # Data for training and validation, without
   the first column (OBS)
```

```r
# Response (y) and attributes/covariates (x)
y <- dt[,1]    # REVENUE (var. dep.)
x <- dt[,-1]   # Attributes (var. indep.)
q <- ncol(x)   # Number of attributes


# Number of folds for validation
k <- 12
m <- n - k

# Predictions and errors for cross-validated ML time series
   models
pred_AR1 <- cv.AR1(y,k)

pred_AR1MAX <- cv.AR1MAX(x,y,k)

pred_ols   <- cv.ols(x, y, k)

pred_lasso <- cv.lasso(x, y, k)

pred_svm   <- cv.svm(x, y, k)

pred_pls <- cv.pls(x, y, k)

pred_dnn <- cv.dnn("REVENUE", as.data.frame(dt), k)

# Median model (simple combining procedure)
P <- cbind(AR1=pred_AR1$prediction,
           AR1MAX=pred_AR1MAX$prediction,
           ols=pred_ols$prediction,
           lasso=pred_lasso$prediction,
           svm=pred_svm$prediction,
```

```
            pls=pred_pls$prediction,
            dnn=pred_dnn$prediction)

pred_median <- apply(P,1,median)     # Median by row of P
error_median <- y[(n-k+1):n] - pred_median

MAE_median <- sum(abs(error_median))/k
MSE_median <- sum(error_median*error_median)/k
AIC_median <- log(sum(error_median*error_median)/k) + 2*q/k
P <- cbind(P, median=pred_median)

# Race between models
MAE <- cbind(AR1=pred_AR1$MAE,
             AR1MAX=pred_AR1MAX$MAE,
             ols=pred_ols$MAE,
             lasso=pred_lasso$MAE,
             svm=pred_svm$MAE,
             pls=pred_pls$MAE,
             dnn=pred_dnn$MAE,
             median=MAE_median)

MSE <- cbind(AR1=pred_AR1$MSE,
             AR1MAX=pred_AR1MAX$MSE,
             ols=pred_ols$MSE,
             lasso=pred_lasso$MSE,
             svm=pred_svm$MSE,
             pls=pred_pls$MSE,
             dnn=pred_dnn$MSE,
             median=MSE_median)

RMSE <- sqrt(MSE)
```

```r
AIC <- cbind(AR1=pred_AR1$AIC,
             AR1MAX=pred_AR1MAX$AIC,
             ols=pred_ols$AIC,
             lasso=pred_lasso$AIC,
             svm=pred_svm$AIC,
             pls=pred_pls$AIC,
             dnn=pred_dnn$AIC,
             median=AIC_median)

round(P,1)
round(MAE,3)
round(MSE,3)
round(RMSE,3)
round(AIC,3)
```