Forecasting bike-sharing demand in Seoul: a comprehensive analysis

Federico Salerno

Dissertation written under the supervision of professor Pedro Afonso Fernandes

Dissertation submitted in partial fulfilment of requirements for the MSc in Business Analytics, at the Universidade Católica Portuguesa, May 2024.

Forecasting bike-sharing demand in Seoul: a comprehensive analysis

Federico Salerno

Abstract

Bike sharing programs represent the future of mobility, contributing to the creation of a "green" economy and a more sustainable future.
Providing the city with a stable and accurate supply of bicycles is a major concern for the city of Seoul. The objective of this research is to provide useful insights on how effectively forecast bike sharing demand through automated processes. Side goals are related with the difference between models' performances as well as with drawing causal effects.

Examining the public rented bicycles in the city, time series forecasting is implemented through different methods, exploring both parametric and non parametric models such as Seasonal ARIMAs with exogenous variables, Multiple Linear Regression and Support Vector Regression. This study takes into consideration mostly weather-related features, consistently with previous literature.
Rides are intuitively influenced by features like temperature as well as by time effects that occur in certain periods of the year. After inspecting the different relationship between response variable and features, models were fit and tested. Consistently with regression errors measured on test set, SVR can be considered the best model for the aim of this research.

Keywords: Time series; Econometrics, Machine learning; Forecasting

Forecasting bike-sharing demand in Seoul: a comprehensive analysis

Federico Salerno

Resumo

Os sistemas de partilha de bicicletas representam o futuro da mobilidade, contribuindo para a criação de uma economia "verde" e para um futuro mais sustentável. Neste âmbito, é preocupação da cidade de Seoul assegurar uma oferta estável e adequada de bicicletas.

O principal objetivo desta investigação é providenciar resultados úteis sobre como prever a procura de bicicletas partilhadas com recurso a métodos automáticos. Os objetivos secundários passam pela análise comparada de performance dos vários modelos preditivos considerados e pela identificação de efeitos causais.

Com base em dados sobre a partilha de bicicletas em Seoul, foram implementados diferentes modelos paramétricos e não paramétricos de séries cronológicas, incluindo ARIMA sazonais, regressões lineares múltiplas e de vetores de suporte. Em coerência com a literatura existente sobre o tema, foram tidas em consideração variáveis relacionadas com o estado do tempo meteorológico.

De facto, as viagens de bicicleta são influenciadas por fatores como a temperatura, bem como por efeitos que ocorrem em certas épocas do ano. Após a exploração das relações entre as variáveis dependente e independentes, foram estimados e testados diferentes modelos. De acordo com os erros de previsão medidos na subamostra de teste, as regressões de vetores de suporte parecem ser a melhor aproximação para prever a procura por bicicletas partilhadas.

Palavras chave: Séries cronológicas; Econometria; Aprendizagem automática; Previsão.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Objectives and Motivation

Bike sharing schemes have developed around the world at a fast pace throughout years, affecting severely transit use, reducing greenhouse gases emissions and enhancing public health in general. Nowadays, sustainability has become a key priority. Customers are more aware and constantly looking for alternative lifestyle practices to reduce their own impact on the environment. In order to help them to make an informed choice, operators need to leverage on the impressive amount of data available in the modern era of 'micro-mobility'.

Forecasting bike sharing demand through standard econometric and Machine Learning algorithms can help achieve positive and tangible results under an economic, social and environmental perspective. The challenges that our society will be facing in the next years in terms of sustainability require the continuous development of resources that we already have. Together with governments, enterprises play a major role. By pursuing effective strategies and adopting clear Corporate Social Responsibility (CSR) frameworks, private companies can lead the transition to a more sustainable economy.

In this context, there have been few generations of bike-sharing programs, considering that the first known program began in 1965 in Amsterdam: ordinary, white bikes were provided for public use, however results were not appealing as bicycles were discarded into canals or stolen for private purposes (DeMaio, 2009). Many years later, the development of Bike Sharing Systems (BSSs) made the service more and more accessible as well as cheaper: nowadays, users expect to easily pick up a bike at any time and anywhere. ICT-based bike-sharing programs offer researchers huge access to large-scale ridership data.

To get an idea of the phenomenon, the number of cities operating a BSS has increased from 13 in 2004 to 855 as of 2014 and the global bike share fleet accounts for about 946 000 bicycles, most of it in China and the South-East Asia region (Fishman, 2016).

However, the user frequency among people who are members of some BSS is still not that high, overall. For instance, in an Australian study, almost half (46%) of annual members recorded no trips and only about 10% use the system on a daily basis as well: bike sharing is still considered as an alternative way of transportation but with impressive room of improvement (Fishman, 2016).

Difficulties can occur in the provision chain of BSSs due to demand/supply changes, weather conditions, urban transportation problems and other factors such as safety concerns (DeMaio, 2009). Thus, due to uneven distribution of users' demand in time and space, it is crucial for operators to understand how to effectively forecast demand changes through a data-driven approach.

Moreover, an accurate understanding of the relative impact of different factors that seems to be strongly influential such as poor weather, darkness and flat terrain is crucial (Winters et al., 2011). Being aware of significant factors as well as to accurately forecast demand is definitely crucial in the decision-making process of any transportation service, giving to the operators the chance to take action coherently with user behaviour and needs. The objective of this research is to understand whether the bike demand can be accurately predicted using several Machine Learning methods as well as standard econometric models.

1.2 Research questions

What is the best model to forecast bike-sharing demand, using time series data? How does the performance of these methods differ? What are the main factors affecting bike sharing demand?

1.3 Hypotheses and Scope of Analysis

Providing commuters with a stable supply of bicycles is a major concern for public providers and private companies, nowadays. The thesis focuses on the comparison of different Machine Learning models as well as standard econometric ones, with the ultimate purpose of forecasting bike-sharing demand on a daily basis in the city of Seoul in a timespan of one year, namely 2018.

Moreover, it aims to investigate to what extent features are strong predictors of bike-sharing demand. Section 2 provides a summary concerning the provision of the different bike sharing schemes throughout years, further business developments as well as reporting how previous literature tackled the problem. Section 3 highlights the methodology, while the dataset is presented in Section 4. Subsequently, modelling phase is explained in Section 5 and the discussion in Section 6.

2 Literature Review

2.1 Bike Sharing models of provision

Since bike-sharing's inception, many different providers have existed, mostly governments, private companies and universities. In the government case, the authority operates the service as it would do with any other public transit service, thus relying on public finances. The benefit is related with the greater control on the service provided by the public authority. Also, they could also leverage on advertising, with the benefit of being convenient and cost-effective as they could not afford to provide the service otherwise. A detriment could be related to an issue of moral hazard. The advertising company usually do not take advantage from the cash flows generated by the service, so the incentive is to operate the program no matter what are the challenges; we should also consider the financial burden that advertising companies have to sustain.

This is well-stated by the general director of JCDecaux, world leader in outdoor advertising, after facing several difficulties in the provision of a joint bike sharing program with the City of Paris: *"It is simple. All the receipts go to the city. All the expenses are ours"* (DeMaio, 2009).

The public authority may also not have the know-how and experience required to manage and make it sustainable under a financial perspective. Differently from the government-owned bike sharing programs, the service for private and for-profit organization consists of an entrepreneurial activity with different business models. Among all, consumption-based as well as "free and freemium" are the most common, noting that advertisment has always been a key component of the revenue streams as well. A benefit can be that it can operate with little bureaucracy involvement (only having local support to use public space), but that also means not having enough public subsidies as well as funding in a "penny market" that strongly relies on volumes and with many competitors.

Finally, universities were pioneers in the provision of bike sharing programs and it is easy to understand why: bicycles are an effective, easy way of moving into campuses and a good alternative to walking. So, the educational institution provides the service with the benefit of expanding its intra-campus transit objectives without relying on the public authority to offer sufficient bicycles available on campus. A drawback could be related to a problem of compatibility between locality's and university's systems.

Considering the disruptive nature of the service nowadays (as part of the "sharing economy"), companies initially focuses on low-end and niche markets of transportation through low-cost bike renting services; they still can meet the needs of consumers from low-end markets and the previous non consumers of public transportation in the attributes that these consumers value. For instance, solving the point-to-point commute problem in short distance, the so-called "last mile" problem (Si et al., 2021).
Consequently, surviving bike-sharing companies have continued to improve the mainstream attributes and gradually penetrated into the urban transportation market and become indispensable.

2.2 Docked vs dockless BSSs and business developments

Within bike sharing programs, there are two substantial types of implementation: traditional docked bike sharing systems and innovative, disruptive dock-less bicycles. Based on the comparison of these two generation of bike-sharing systems, similarities and differences can be related to these subsequent summarised aspects: service usage, "last mile" implications, accessibility and distribution (Chen et al., 2020).

There have been previous researches with the goal of studying the social impact and mobility patterns of bike-sharing programs as well as methods regarding the bikes' redistribution among urban centers. By definition, docked and dock-less bike-sharing systems are fundamentally built in a different way; the real advantage of dock-less bike sharing (DBS) is related to its free-floating characteristics. The core equipment is the smart lock that the quick response code allows users to borrow and return the bikes anytime and anywhere as well as GPS device that can easily locate real-time positions. That has actually implications on temporal and spatial differences of usage.

The exploratory analysis conducted by Grant McKenzie offers important insights that show how urban cyclist interact with their city, using a panel of US cities (McKenzie, 2018). It suggests that docked bikes are more frequently used for commuting to and from work than the dock-less ones. Also, it demonstrates that docked ridership is more focused on the central/business districts of the city than dock-less bikes. One interesting finding is that predominantly residential regions (having lower household income) use less bike sharing services, no matter what the typology. While both services can be considered relatively inexpensive, these bike-sharing services use digital payments as well as credit

cards as the basis of payment, making it less likely that lower income individuals can use them, according to (McKenzie, 2018).

There's also a theme of business models. When dock-less bike sharing was first launched, its business logic was that the enterprises provided customers with cheap and convenient bicycle rental service through purchasing and offering bikes uniformly. Companies were charging fees based on the time of use. This kind of rental service seemed to be of high frequency and could create a very strong cash flow. However, such business model was immature, with many problems and lack of a stable and effective developing path. Thus, players in the market (such as ofo, Mobike, Hellobike) spent huge amount of money on subsidies for users (through special offering, coupouns, discounts and so on) and that influx of capital has made bike-sharing companies to no longer compete through continuous development. At a certain point, they were not even aiming to profits, but just constantly spending money and subsidize because it seemed that there would always be profitable investments to cover the financial burden of capital expenses and operating costs; thus, virtually becoming more a financing tool than following normal business logic. In this framework, it is not surprising the high failure rate of companies in the market.

Based on (Si et al., 2021), the future developments of the business model of bike sharing needs to grasp three key elements: *value creation, value delivery and value capture.* Basically, its intrinsic value lies in providing low price, great convenience and an effective solution to the point-to-point demand in short distance commute. In order to implement an effective value chain, bike-sharing companies should massively improve the quality of their bikes and attributes (for instance new mopeds that allows users to ride effortlessly) as well as managing cost through technology and process improvements. Value capture can be assessed through different strategies such as: personalization, asset sharing, an agile and adaptive organization as well as a collaborative ecosystem.

2.3 Bike sharing demand forecasting

Solving the mismatch problem of bikes is crucial and in order to achieve that in a business-oriented perspective it is important to understand which factors affects the demand. Many researches have been conducted throughout years for different purposes, namely forecasting, prediction as well as for exploratory data objectives.

Previous literature addressed these problems by looking at: weather conditions, built environment, public transportation, socio-demographic attributes, temporal factors and safety (Eren and Uz, 2020). Looking at the aforementioned research paper, it states that, in addition to the long-term seasonal effects of climate on bike ridership, weather conditions change continuously in the short term and may affect the demand for trip generation.

It is determined that "there is a positive correlation between the temperature increase and the bike sharing demand" (Eren and Uz, 2020; Fishman, 2016; Winters et al., 2011). Specifically, the trip production is positively correlated when the weather temperature is between 0-20°C, reaching a maximum level when the range is 20°-30°C.

(Kim, 2018) took a step further by collecting data also from spatial features (residential, commercial, industrial, university districts), aiming to investigate calendar effects due to weekends, public holidays and school holidays. The findings of this study are that there was no significant difference in the number of bike rentals on weekends and normal working days, but there was a reduction of trips during the weekend mornings. Surprisingly, the bike sharing rentals on public holidays has decreased as well as the effect of school holidays does not impact the usage in a significant way; it does not affect the usage due to the impossibility of children under a certain age to become members.
To summarise, previous literature thoroughly investigated the effects of different type of features on the bike sharing ridership, with an important stress on weather-related factors which are reasonably crucial to be considered in this particular matter.

The methods and the granularity adopted by previous studies matter as well: the topic has been discussed in different spatial scales, namely on a city level, single station level and station clustering level such as, for instance, in (Chen et al., 2016) and (Feng et al., 2018), where stations were classified into different clusters and predicted station-based demand for the different category. Results showed that clustering better captures the local demand than on a city-level for merely prediction purposes, with the possible limitation of mismatching local differences between different single-stations.

Among the studies on demand forecasting, time-series regression models have been applied. By analyzing mobility patterns of *Bicing*, an urban community bicycle program in Barcelona, (Kaltenbrunner et al., 2010) used a classic autoregressive moving average (ARMA) for forecasting purposes; a limitation is given by the fact the aforementioned model requires the strong assumption of stationality of the time-series.

Consequently, (Yoon et al., 2012) assessed the problem by adopting an auto-regressive integrated moving average (ARIMA). Further developments of these time-series regression based models have been implemented; (Yu et al., 2023) proposed a hybrid model that relies on SARIMA, in order to effectively capture seasonal patterns (calendar effects due to holiday/weekend). All these models are designed to capture linear dependencies in the time-series data, but may not be enough where past values and past errors can not linearly predict future values.

Thus, latest developments in machine and deep learning represent a helpful tool to enhance forecasting performances as well as stressing on the importance of an appropriate feature selection.

Interestingly, (Peláez-Rodríguez et al., 2024) provided a ML approach to solve the same problem and with the same scope (*Bicing* problem) presented previously (Kaltenbrunner et al., 2010). It is crucial that models receive information of past demand, otherwise significant deviations and outliers may be expected. Then, the database construction with the addition of exogenous variables is useful when solving time-series forecasting. In details, factors were categorized in categorical, meteorological and seasonal, with the latter having a negative impact on the target variable. After the application of a feature selection algorithm that enables to select optimal predictor variables, many models were considered: MLR (Multiple Linear Regression), SVR (Support Vector Regression), RF (Random Forest), LSTM (Long-Short Term Memory), Lasso Regression, CNN (Convolutional Neural Network). In details, RF recorded the best score by having the highest R^2 and the lowest score for both RMSE and MAE. These models as well as performance metrics will be discussed in details in a subsequent, dedicated chapter.

Moreover, (Boufidis et al., 2020) provided a comprehensive study on machine and deep learning methods' performance. To tackle this problem of predicting bike rentals, the study aims to compare different regression metrics (in details: MAE, MSE, RMSE, R^2) for a panel of models. Overall, Random Forest and Gradient Boosting models seems

to fit the train set better than other algorithms, recording lower errors and higher R^2 values. However, generalization's ability must be seen on out-of-sample performance, where Gradient Boosting produces slightly less errors with respect to the rest of models. Interestingly, neural networks records the worst fit both for train and test sets.

However, a combined approach has showed significant results in (Yu et al., 2023); evidence suggests that LSTMs usually outperform standard parametric models but using both in a hybrid model tremendously reduce regression errors, enhancing the performance. To summarise, previous literature achieved results by reasonably prioritizing weather-related features as strong predictors, by adding exogenous/categorical variables to catch seasonal, time-related effects (holidays, weekend, specific hour within the day) as well as by understanding linear dependencies within data in order to adopt the fittest model among standard econometric methods and latest developments in machine and deep learning algorithms.

3 Methodology

The methodology adopted to address the problem consists of five linked phases: *business understanding, data understanding, data pre-processing, modelling* and *valuation*.

In *business understanding*, the aim is to identify the best approach under a business and operating perspective. The topic is addressed as a Supervised Learning problem; data, output variable are labelled and there is an associated response measurement y_i for each observation of predictor measurements (James et al., 2023). On the other hand, Unsupervised Learning would be a good fit for the detection of underlying and unobserved patterns in data without an associated response measurement (Customer Segmentation with Clustering algorithms, for instance).

In this case, the aim is to properly fit a model where the dependent variable is related with predictors and to inspect the relationship between them, through inference (James et al., 2023). Within Supervised Learning framework, the problem could either be a *regression* or *classification* problem. In this case, we want to forecast a quantitative, continuous response variable, namely *Rented Bike Count*. Thus, we'll be considering regression models that will be evaluated according to appropriate performance metrics that will be discussed in the subsequent chapters.

Data understanding and pre-processing's objective is to inspect variables in the dataset and incorporate new, useful features (transformations, dummy variables) in order to enhance modelling accuracy.

Then, in *modelling*, parametric and non-parametric models were trained and fitted, namely Seasonal ARIMAs (SARIMAs), Support Vector Regression (SVR) and Time Series Linear Regression (TSLM). Also, data is split into two different train and test sets, accordingly with modelling best practices.

Finally, the goodness of fit and out-of-sample validation of the adopted models are measured in the *valuation* phase through different performance metrics. In details, MAE (*mean absolute error*), RMSE (*root-mean-square error*) and MAD (*median absolute deviation*) are considered a good fit for valuation purposes of a regression problem for time series analysis.

3.1 Methods

3.1.1 Multiple Linear Regression (MLR)

The underlying assumption that holds in time series linear regression is that there is a linear relationship between the explained variable (thus, the time series of interest y_t) and the predictors.

The simplest linear regression is the SLR (Simple Linear Regression) that allows just one single predictor in the model. More sophisticated models such as MLR (Multiple Linear Regression) are more commonly used in predictive analytics as well as time series forecasting.

The standard mathematical form of a multiple regression model is:

$$y_t = \beta_0 + \beta_1 x_{1,t} + \beta_2 x_{2,t} + ... + \beta_k x_{k,t} \tag{1}$$

where y is the variable to be forecasted and $x_1, ..., x_k$ are the k predictor variables; each of the predictor variables must be numerical (Hyndman and Athanasopoulos, 2021).

The coefficients $\beta_1, ..., \beta_k$ measure the effect of each predictor after taking into account the effects of all the other predictors in the model and that is called *marginal effect* of the predictor variables (Hyndman and Athanasopoulos, 2021). Among all estimators, coefficients are commonly estimated through *OLS (Ordinary Least Squares)*; the estimation is computed through data and it returns the least value for the sum of squared errors. Please consider that the main research question focuses on forecasting bike sharing demand and not on conducting statistical inference (thus, the effect of each factor on rented bicycles). Panel data track the same unit over multiple time periods and the lack of independence is a violation of standard OLS assumptions (that normally does not occur in pooled cross-sections), leading to wrong and misleading *p-value* computation. Coefficients can be still unbiased but the significant (or not) effect of each factor maybe misleading and unreliable in this context.

Concerning the assumptions for the Linear Model itself, the aforementioned assumption of linearity is not the only one that holds; errors must have zero mean for unbiased forecasting as well as a constant variance to accurately produce prediction intervals. Also, they should not be autocorrelated (meaning extra information still left out) and correlated with predictors, which should not be random variables.

3.1.2 Seasonal ARIMA (SARIMA)

SARIMA is a parametric statistical method commonly used for time series forecasting purposes. In order to get a deeper understanding of this model, it is important to inspect non-seasonal ARIMA and its *autoregressive* and *moving average* components first.
Autoregressive models use a linear combination of lagged values in order to predict the target variable. Specifically,

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + ... + \phi_p y_{t-p} + \epsilon_t \tag{2}$$

where ϵ_t is a white noise; we can refer to it as an autoregressive model of order p, AR(p). Autoregressive models are remarkably flexible at handling a wide range of different time series patterns (Hyndman and Athanasopoulos, 2021). Recalling the aforementioned formula, we'll consider an AR(1) model in order to explain the different parameters' scenarios, as explained in (Hyndman and Athanasopoulos, 2021):

- when $\phi_1 = 0$, y_t is equivalent to white noise,
- when $\phi_1 = 1$ and $c = 0$, the dependant variable is a random walk.
  The best prediction becomes the last observation such as for stock prices,
- when $\phi_1 = 1$ and $c \neq 0$ is equivalent with a random walk with drift,
- when $\phi_1 < 0$, the dependant variable tends to oscillate between positive and negative values.

It is important to consider that these models are generally linked to an assumption of *stationarity* of data. A stationary time series is one whose properties do not depend on the time at which the series is observed, thus the trend and seasonality can not affect the value of the time series at different lags (Hyndman and Athanasopoulos, 2021). In details, data must record a constant mean, constant variance as well as a homogeneous autocorrelation pattern over time. However, the assumption is strong especially for real-case scenarios as fluctuations (seasonality, trends, cycles) generally occur and that is valid also for the specific topic of this research. The problem can be tackled by using some techniques; the time series of interest can be pre-processed by differencing and log-scaling the response variable to make the mean constant and reduce the variance throughout time (Hyndman and Athanasopoulos, 2021).

Moving Average models consist of the construction of a linear relationship between past forecasts errors and the target variable y_t, which can be thought of as a weighted moving average of the aforementioned errors (Hyndman and Athanasopoulos, 2021).
In mathematical terms,

$$y_t = c + \epsilon_t + \theta_1\epsilon_{t-1} + \theta_2\epsilon_{t-2} + ... + \theta_q\epsilon_{t-q} \tag{3}$$

The Moving Average model of order q, MA(q), takes into consideration past forecasts' errors that are not observed. *Invertibility* is required in order to have stable and unique moving average models. The concepts of *stationarity* and *invertibility* are strongly linked with each other. The latter refers to the property of a MA(q) to be revised as an AR of infinite order, under certain conditions.

Differently from standard ARMA (p,q) model, ARIMA(p,d,q) has another parameter that has to be set, which refers to the level of differencing involved; that's why it is an *Autoregressive Integrated Moving Average* model. Even though differencing can be helpful in making the time series stationary, it could be not enough for standard ARIMAs. These models may need a seasonal component in order to properly deal with seasonality, thus Seasonal ARIMAs are not disregarded in this context. By adding seasonal terms $(P, D, Q)_m$ (where m refers to the seasonal period of the data), it is possible to model different seasonal patterns (weekly, monthly, quarterly and so on). In order to optimally set models' parameters, ACF and PACF analysis must be implemented.

Autocorrelation Function (ACF) computes the correlation between a time-series and its lagged values, showing how much past values influence current ones. Partial Autocorrelation Function (PACF) measures the correlation between the time series and its lagged values while controlling for the indirect effects of shorter lags. Both ACF and PACF range from -1 to 1. Thus, zero, positive or negative autocorrelation can occur between current and past values of the time-series.

3.1.3 Support Vector Regression (SVR)

SVR belongs to the larger family of Support Vector Machines (SVM), which are algorithms that are commonly used for classification tasks. SVR relies on the same mechanisms that work for classification SVMs, with some exceptions. SVMs are the result of a continuous development that starts with the *Maximal Margin Classifier*; in a context of a binary

response variable, this classifier is constructed upon the choice of several separating hyperplanes.

What is a hyperplane? The definition given by (James et al., 2023) states that it is a flat affine subspace of dimension $p - 1$. In practical terms, in two dimensions, a hyperplane is a flat one-dimensional subspace (so, a line) and, in three dimensions, a hyperplane is a plane (James et al., 2023). However, there are infinite possibilities of hyperplanes that can effectively separates our observations. The *Maximal Margin Classifier* rely on a separating hyperplane that is farthest from training observations; computing the perpendicular distance from each training observation to a given separating hyperplane, the *maximal margin hyperplane* is the one that has the farthest minimum distance to the training observations (James et al., 2023).

However, it is possible that two classes can not be separated by a hyperplane and *maximal margin classifier* is very sensitive to changes in observations as well; this problem can lead to a classifier that overfits data and that is not effective with new, unseen data. Support Vector Machines (SVMs) try to solve this issue by setting a margin of tolerance, making it possible for an observation to be on the wrong side of the margin as well as of the hyperplane, resulting in greater robustness to individual observations and a better classification of most of the training instances (James et al., 2023). This model rely on support margin classifiers but in an enlarged feature space, leveraging on kernels. These tools are mathematical functions (e.g. linear, polynomial, sigmoid) that transform original data in a higher dimensional space, facilitating the choice of a optimal separating hyperplane. Using kernels, SVMs can better handle complex non-linear relationships, compared to other models.

Similarly, SVR aims to find the best hyperplane that maximises the distance with the closest observation and that - at the same time - minimise the error, in a continuous space. It can be considered as an extension of Support Vector Machines (SVM). Differently from Multiple Linear Regression (MLR), the coefficients are not OLS-estimated but SVR seeks coefficients that minimize another type of loss function, where only residuals larger in absolute value than some positive constant contribute to the loss function (James et al., 2023). It builds the regression model based on the *supports*. One of its strengths is that it is effective in the modelling of non-linear relationships; being a non parametric model, it does not make assumptions on the nature of the relationship and it does not

rely on a pre-determined number of parameters, making it a powerful method to model non-linearity.

3.2 Performance assessment

The performance of the models is assessed by considering the closeness of the forecasts from the actual values, following the Supervised Learning approach. For time series regression analysis, three main errors are computed: RMSE, MAE and MAD. Lower errors suggest a better fit.

3.2.1 RMSE

Root-Mean-Square Error (RMSE) is the square root of MSE (*mean squared error*), where the latter is computed as:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{4}$$

The deviations from actual values are squared, summarized and then averaged, resulting in a positive value. Due to the squared differences, MSE amplifies larger errors and one of its disadvantages is the lack of scale-dependency. Considering that errors are squared and the unit as well, it would be more useful for comprehension purposes to consider the root of this computation:

$$RMSE = \sqrt{MSE} \tag{5}$$

RMSE now matches the unit of the considered variable. Please note that it is sensitive to the presence of outliers.

3.2.2 MAE

Mean Absolute Error (MAE) is computed as the mean of the absolute differences between forecasts and actual values. It is calculated on the same unit of the variable and it does not amplify large errors compared to RMSE.

In mathematical terms,

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{6}$$

3.2.3 MAD

Median Absolute Deviation (MAD) is calculated as the median of the absolute errors.

$$MAD = median(|y_i - \hat{y}_i|) \tag{7}$$

As we are considering the median, it is robust to the presence of large errors. It does not consider squared differences, thus it computed on the same unit of the variable that it is considered.

R^2 is not considered in this analysis. It is commonly used to assess the goodness-of-fit of a linear model, by calculating the proportion of variance explained. However, it could not be suitable for a time series regression analysis due to violations of assumptions.
Time series are not usually stationary, recording inconstant mean, variance and errors correlated overtime that may inflate the R^2 computation to a misleadingly higher value, suggesting a better fit even though it is not actually recorded.

4 Case Study: Seoul's Bike Sharing Demand

4.1 Data Overview

The dataset taken into consideration provides detailed information concerning Seoul's bike sharing activity; it is an open-source dataset available on the UC Irvine Machine Learning Repository as well as on Kaggle. The granularity is hourly-related and the target variable considered is *Rented Bike Count*; each observation in the dataset provides information concerning bicycles rented each hour in a timespan of about one year, namely 2018. The data preparation phase and exploratory data analysis was implemented in Python, while the modelling and valuation phases were coded using RStudio. The raw data's variables and the code will be provided in the Appendix.

4.2 Data Pre-processing & EDA

The raw data records 8760 observations with 14 variables, mostly with information concerning weathers' features. It starts from 01/12/2017 and end on 30/11/2018. As already cited, *Rented Bike Count* is the response variable of our analysis. On average, the bicycles rented overtime are about 705 per hour; however, the distribution is not normal but left-skewed with the presence of some outliers, thus it would be more informative to see another measure of central tendency robust to outliers such as the median. In this case, the median accounts for 505 bicycles. The distribution's minimum and maximum are respectively 0 and 3556, with half of the records between 191 and 1065. Also, there are some zeros, namely 295 just in the response variable. The dataset does not record missing values, overall. Aiming to inspect time seasonality on different levels, four different variables are created: *Dayofweek, Quarter, Month, Dayofyear*.

Please note that the dataset was, then, aggregated on a daily basis. Forecasting bikes rented hourly lead to no results, after a thorough and challenging analysis; the multiple levels of seasonality are difficult to incorporate in the models, but that will be discussed specifically in the *Limitations* section of this research.

Daily aggregation allows to eliminate the hourly component of seasonality; the aggregated dataset now records one observation per day. The trade-off is to lose observations, restricting the dataset to 365 instances. Due to the presence of seasonality at multiple levels, the time series is not stationary yet. Hourly seasonality was eliminated through ag-

gregation. Considering that *Seasons* variable is categorical, dummy variables are created in order to control for its effect. Also, log-scaling the time series of interest was helpful to make the variance more constant overtime and differencing was useful for constant mean purposes. First-level differencing was implemented on a daily basis, so we are now considering day by day changes in the number of bike rented.

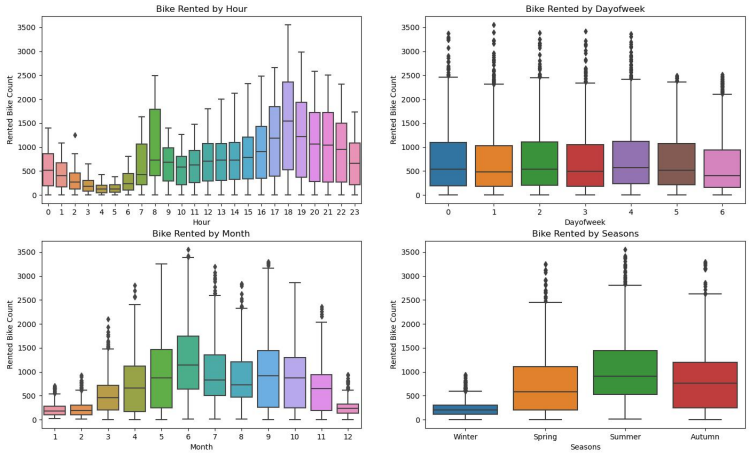


Figure 1: Seasonality at different time granularity and *Seasons*

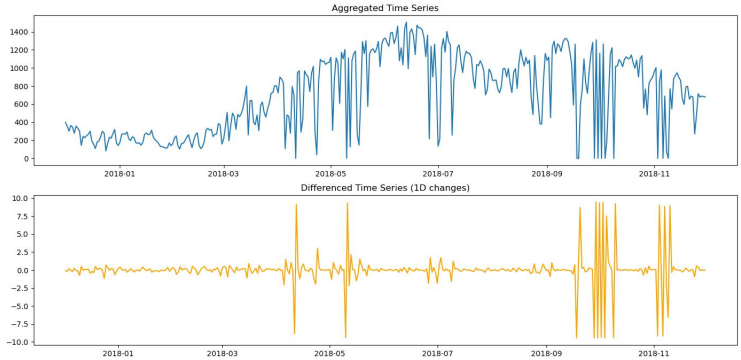


Figure 2: Daily time series before and after preprocessing: log-scaling and differencing

Other types of differences have been computed, as well. Initially, the main focus was on the changes between weeks, resulting in losing more observations (first 7 are lost, instead of just one) and not improving autocorrelation pattern throughout lags. Then, second-differencing was computed but it did not improve results; the major drawback of an over-differenciated time series is the lack of interpretability. As we can see from Figure 1, the count of bike rented is, on average, different between months and seasons, therefore it would be useful to include them in the *modelling* phase. There's no significant difference on average between days of the week, though.

Another important step is to understand if covariates are correlated between each other. Collinearity may occur and it could worsen forecasting accuracy. Also, it may provide useful insights regarding variables that can be included in the models as predictors. A stepwise approach was adopted; starting from simpler models with fewer covariates can help the interpretability of the models themselves, ceteris paribus. For this purpose, a correlation matrix was computed:

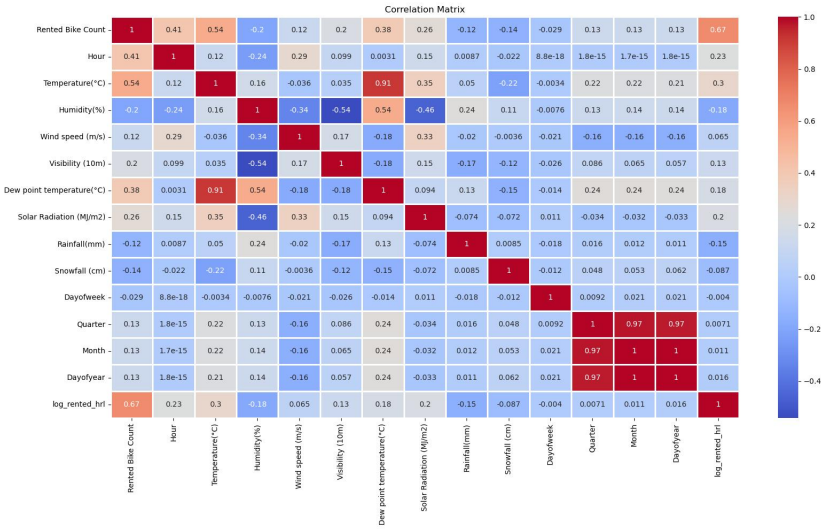


Figure 3: Correlation matrix

Not considering time variables and transformations, *Temperature (C°)* and *Dew point temperature (°C)* show a significant level of correlation between them and the target variable, recording respectively 0.54 and 0.38 as Pearson correlation coefficient. Being pretty similar, they are highly correlated so I decided to exclude one of them in order to avoid collinearity; *Temperature (°C)* is more correlated with the response variable and it is a more reliable, thorough measure of the current state of weather.

Intuitively, people are more encouraged to ride bicycles with warmer temperature and seasons, rather than during the coolest periods of the year.

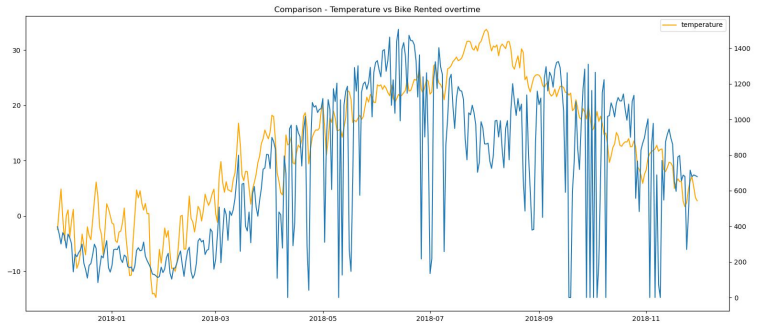


Figure 4: Contrasting *Temperature* and *Rented Bike Count* over time

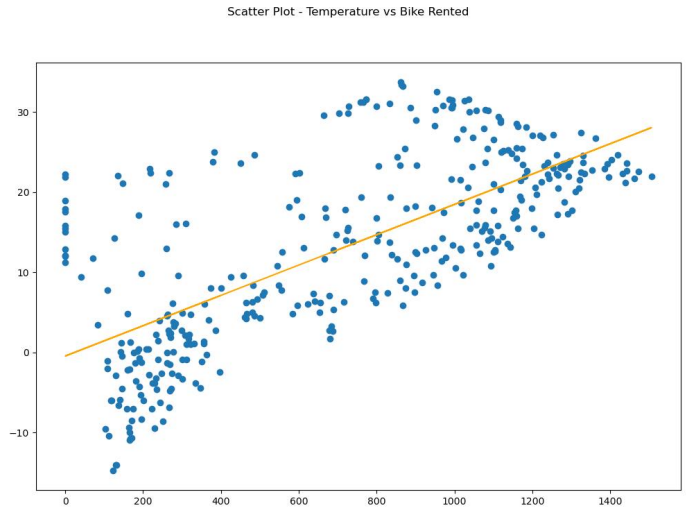


Figure 5: Scatter plot contrasting *Temperature* and *Rented Bike Count*

The aforementioned covariate and the time series of interest have, overall, a certain similar cycle and seasonality throughout time as well as a slight linear relationship.
At the end of this phase, the dataset pre-processed in Python is saved as a *.csv* file and then read on RStudio for the *modelling* and *valuation* phases.

5 Modelling & Findings

The pre-processed, aggregated dataset records a new variable, namely *sdd* which refers to 'Seasonal Daily Differencing'; the new response variable represents the daily changes of *Rented Bike Count*, in the aftermath of log-scaling transformation. Among several Machine Learning and standard statistical parametric models, SARIMAs would be a good fit for our problem thanks to its proven ability to effectively incorporate seasonal effects (Yu et al., 2023). More precisely, we are also including some variables that have been previously inspected in the Exploratory Data Analysis. We are now considering seasonal ARIMA models with exogenous variables (SARIMAX); *Month* and *Temperature (°C)* are included as covariates in the model, aiming to control for monthly seasonality as well as for the correlation between the time series of interest and the lattest. After an ex-post comparison, including *Seasons* dummies did not improve their performance, thus it is preferable to include less complex models, ceteris paribus.

In order to have a benchmark to facilitate comparison between models, TSLM (Time Series Linear Model) was included; coefficients are OLS-estimated and it includes *Month*, *Temperature (°C)* plus *Seasons* dummies. Due to collinearity issues, one out of four *Seasons* dummies was dropped, thus only three are considered. Then, a non parametric algorithm was included; Support Vector Regression (SVR) is a type of Support Vector Machine for regression tasks. Its strength is that it can more efficiently deal with non-linear relationships that link variables in the data better than standard parametric models; both TSLM and SARIMAXs rely on the assumption of linearity between response variable and predictors, but that is not valid for non parametric models such as SVR. These methods do not make assumptions on the relationship that occur between output variables and predictors. On the other hand, it may be more computationally challenging as there isn't a restricted number of parameters to be estimated.

In order to set SARIMAs parameters and to double check stationarity, I inspected ACF and PACF plots; the former can help in the choice of both seasonal and non seasonal MA parameters and the latter can be useful for the AR component, as a rule of thumb. On the one hand, the significant spike recorded at lag 1 of ACF suggests a non seasonal MA(1). On the other hand, PACF has a tail-off shape until the 5th lag; there are some slightly significant spikes after that lag but I started with AR(3) and then eventually incrementing.

Summarizing, we are considering two seasonal ARIMAs (3,1,1), respectively with (1,1,0) and (0,1,1) as seasonal components that control for the annual periodicity of the time series.

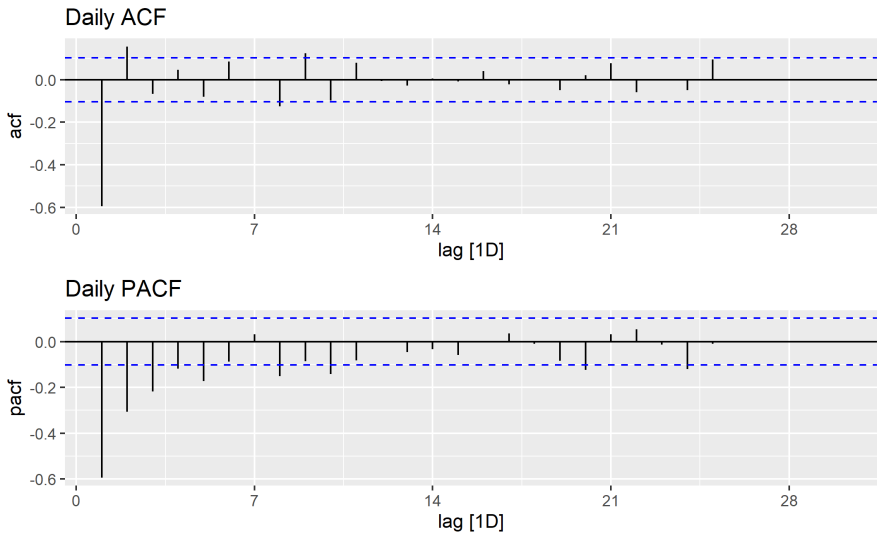


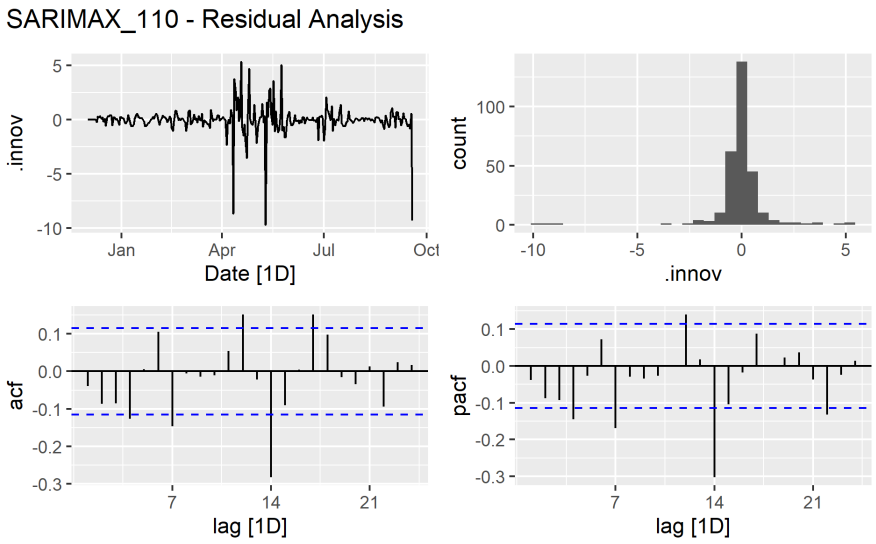
Figure 6: Daily differenced *Rented Bike Count* ACF and PACF plot

The dataset was split into a training and a test set, with a 80%-20% split accordingly with modelling best practices; the former accounts for 291 observations, while the test set accounts for 73 instances (for a total of 364). This is consistent because we are now considering daily differences, so the first observation is removed as it produces a NA value. Out-of-sample validation is guaranteed by testing on unseen data; however, cross-validation would be helpful to strengthen generalization. K-fold cross-validation, which is commonly used, can not be implemented due to the dependency of data throughout time (it would not make sense to predict past values using future values, by dividing all the series in k-folds and resampling), thus it is more challenging to implement time series cross-validation. This issue will be discussed more in details in the *Limitations* section of the research.

After the training phase, residual analysis was implemented in order to check the fitness of the models. In details, residuals should not record any pattern (white noise), thus having zero mean and homoskedasticity. The requirement of normality in the residuals is not strictly necessary for our purpose of forecasting, however it can be helpful for the models' consistency. Zero-mean condition indicates that the model does not systematically under/overestimate the dependant variable, otherwise the model would produce biased

estimates. That's also true for homoskedasticity; residuals' variance should be constant overtime in order to have consistent performances with all the values of the time series. Finally, residuals should be white noise; if there are significant spikes, the model is not able to incorporate all the information but there are still underlying patterns other than random error (which can not be predicted).

Looking at the residuals of the models considered for our purpose, SVR and $SARIMAX_{011}$ recorded the best results. Overall, all models are zero-mean centered and quite normally distributed; however, there are some outliers (larger errors in the distribution of residuals), probably due to the lack of homoskedasticity at certain levels of the time series (between April and May, right before October). Being systematic and consistent in all models, differencing and log-scaling was not enough to reduce variability of dependent variable in the aforementioned time periods. This could be a problem, as models eventually will not perform homogeneously in all time periods. A possible solution could be the fitting of a median model that can mitigate this issue. The periods that record high peaks in residuals maybe inspected apart for further studies.



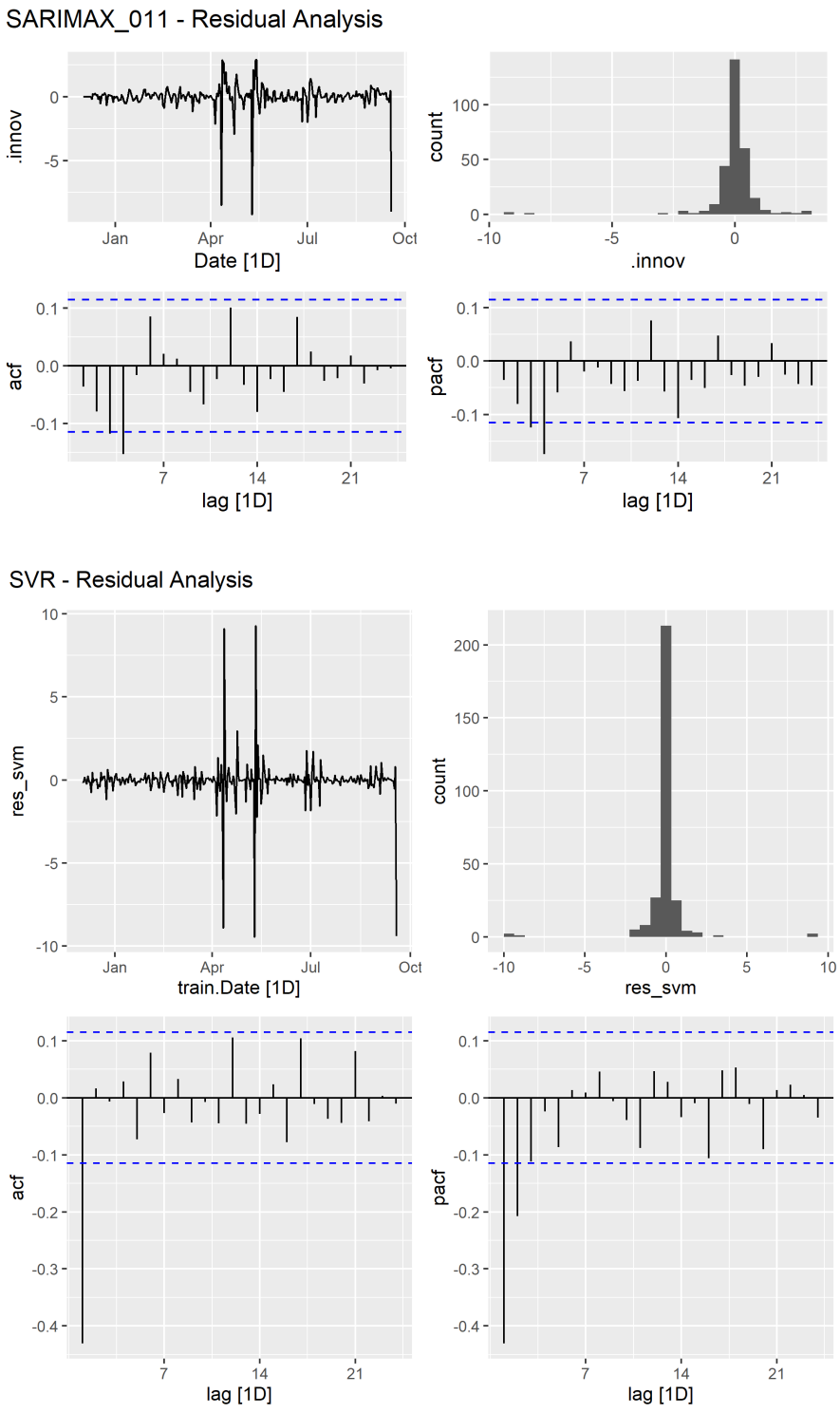


Figure 7: Residual analysis comparison between SARIMAs and SVR

Looking at autocorrelation plots, residuals can be considered white noise, with the exception of $SARIMAX_{110}$. The latter records some significant spikes in both plots and that also happens every 7 days, suggesting an underlying weekly and biweekly pattern that it is left out from the model. There's a slightly significant spike in $SARIMAX_{011}$

as well but that can occur randomly; overall, the residuals are not correlated with each other as they lie within the confidence interval. The same statement can be valid for SVR, where ACF records high negative correlation at first lag but residuals can be assumed as white noise for next lags.

In the aftermath of residual analysis, trained models are tested on unseen data for out-of-sample validation; the test set starts from 2018-09-19 and ends on the 2018-11-30, covering the final months of the dataset. As already mentioned, the performance metrics considered are three standard regression metrics, namely RMSE, MAD and MAE. A forecast method that minimises the MAE will lead to forecasts of the median, while minimising the RMSE leads to forecasts of the mean (Hyndman and Athanasopoulos, 2021).

The errors computed show that SVR outperforms other models in the test set and it can be considered as the best model for our purpose, noting that it is the best in minimizing the forecasts errors. SVR was fitted including the subsequent variables as predictors: *Month*, *Temperature (°C)* and three *Seasons* dummies, namely *Winter, Spring, Summer*. Interestingly, $SARIMAX_{011}$ records similar results but $SARIMAX_{110}$ performs worse than any model, not beating Linear Regression which is the benchmark in this analysis; that's consistent with previous considerations because of its residual analysis' poor results. Thus, it is also crucial to optimally choose parameters because models' performances can be very different. The difference between best and worst performance can be caused by the complexity of models and their assumption's suitability for this type of data.

As inspected previously in the Exploratory Data Analysis, bike sharing demand is largely affected by seasonality at several time components (and not solely) that occur simultaneously. Decomposing the time series should be done thoroughly. Due to the presence of some outliers in the residual analysis (thus, larger residuals with respect to the distribution), RMSE is larger than MAE as expected. As already pointed out, that could be due to the lack of homoskedasticity, resulting in inconsistent performances in certain time periods. Note that MAD is lower in absolute values, being robust to the presence of the aforementioned outliers in the residuals distribution. Results may be influenced by the nature and size of test data; we have to consider that forecasts are computed in the last months of the year that include periods with higher changes in bike sharing demand (thus, higher variance), compared to the whole time series.

The relevance of this performance metrics in the business context is huge. It can impact operational efficiency, inventory management, revenue optimization and customer retention. An efficient allocation of the resources allows providers to perform maintenance in the hours when the number of bike rentals is lower, ensuring better schedules that do not affect the business. Overestimating bicycles needed can cause oversupply, increasing operational costs but without a matched revenue. Bikes being unavailable when needed can frustrate users that may seek alternative competitors, as well.

The service relies on brand reputation, thus an accurate allocation of resources can prevent customers to switch to other providers. Loyal customers can also help expand the brand through word-of-mouth and recommendations, translating in an increased usage. An increased amount of data means that the provider can better understand customers' behavioural patterns and then better position themselves in the market. The strategy concerning revenue optimization can leverage on data analytics, developing dynamic pricing and price discrimination. Prices can be adjusted during high-demand periods and according to behavioural, cycling patterns of the commuters.

Table 1: Forecasts errors on test set

	MAD	MAE	RMSE
SVR	0.04722793	2.276150	4.368714
$SARIMAX_{011}$	0.45218077	2.510343	4.452931
$LinearModel$	0.57440999	2.560005	4.411134
$SARIMAX_{110}$	1.20961165	3.110205	4.509756

6 Discussion

6.1 Limitations

It is important to specify that there are some limitations and constraints in this study. First, the tiny number of observations in the dataset after the daily aggregation can hinder generalization of the models. Forecasting the number of rented bicycles on a hourly basis was the initial objective, however it became challenging due to the presence of several effects of seasonality at different time levels.
The granularity of data was, then, reconsidered on a daily basis through aggregation; the trade-off was, intuitively, losing observations.

Apart from the size of the dataset, generalization is also threatened by the lack of cross-validation. Standard K-fold cross-validation can not be applied in time series analysis, due to the time constraint. In this study, results were not cross-validated, hence the sample on which models are tested is always composed of the last 20% observations in the dataset. Also, previous literature is more focused on the prediction with pooled cross-sectional data, mainly inspecting how Machine Learning models perform as well as drawing causal relationships between response and predictors.

Please note that many models were excluded for forecasting purposes but they were a good fit for predictions with pooled-cross sections, for instance. Specifically, Decision Trees and Random Forests were not considered because they require independence between observations, although they were widely used in other previous researches. Differencing can threaten interpretability of results, as well; an over-differenciated time-series can lead to misinterpretation of results and explainability issues.

Finally, there may be difficulties in the deployment of the models due to data drift. Data drift can reduce the accuracy of the models over time, when applied in real case scenarios. This problem involves changes over time in statistical properties of the distributions of data or in the relationships between the time series of interest and its predictors. This study focuses on a limited dataset that records historical data that can be outdated when applied in real case scenarios.

6.2 Conclusion and further developments

This study provides useful insights regarding bike sharing forecasting in an urban context, by comparing standard econometrics and Machine Learning models.
Bike sharing demand is highly affected by seasonality, recording patterns that repeat at a constant frequency overtime. Cycles and trends components can also occur. However, the former is more difficult to predict due to its non-uniformity. Unlike seasonality, cycles do not have a fixed frequency and their duration can vary significantly, requiring longer historical data. Due to the restricted number of observations in the time-series of interest, it is challenging to determine whether or not there is a cycle happening in the number of bike rentals. Linear trend can be handled by differencing the time-series of interest. In this specific case, seasonality is driven by cyclical factors of a fixed length related to time components and weather variables.

The main challenge is to understand how multiple seasonal patterns that eventually interact with each other can be thoroughly handled. In order to do so, time variables were created and inspected. The periodic nature is recorded over different time scales: hourly, monthly, quarterly. There is no calendar effects due to the presence of weekends as no statistical significant difference was detected between them and working days, in the time span of interest. Hourly seasonal patterns are handled through aggregation, after a careful analysis.

As already mentioned in the *Limitations* section, the granularity of the data was then reconsidered on a daily basis. Monthly seasonality is controlled in the models with the appropriate covariate. Due to collinearity issues, the performance of the models was worse after adding covariates to control for quarterly seasonality. *Quarter* and *Month* record a high Pearson correlation coefficient, showing a strong correlation between each other. Thus, I preferred not to include both. Daily differencing helped mitigate weekly seasonality. If there is a recurring pattern every 7 days, considering the differences (not absolute values, anymore) can be helpful for this purpose.

Finally, the number of bike rentals records statistical significant differences between seasons, suggesting a seasonal effect. Due to its categorical data type, dummy variables for *Seasons* were created in order to better control for its effect. One out of four was dropped in order to avoid collinearity.

The pre-processing part requires an important mention in this study. The distribution of rented bikes is rarely normal-shaped and there are intuitively some outliers; some hours, especially during nights, do not record any rental, while there are some peaks in the demand right before and after working hours. By differencing and log-scaling the response variable, the accuracy of the models tremendously improved.

Concerning the main research question, the best model is Support Vector Regression (SVR), which achieved the best results overall. Comparing all the models, only $SARIMAX_{011}$ recorded similar results on out-of-sample validation. Overall, the benefit of non parametric models is that they can better handle non-linear relationships that may occur between the time series of interest and covariates. The assumption of linearity could be a restriction in this type of analysis, considering that the demand itself is not linear throughout time and there is seasonality at different time levels. The advantage is that there is not a pre-determined number of coefficients being estimated, considering that finding optimal parameters in standard econometrics models is challenging and performances' results can vary. For instance, $SARIMAX_{110}$ and $SARIMAX_{011}$ recorded very different results.

In conclusion, SVR outperformed standard econometric models, proving that Machine Learning algorithms can be an effective tool for forecasting purposes and not just for prediction. Previous literature mostly covered Machine Learning tools to predict bike sharing demand using pooled-cross sections but not for forecasting purpose in a panel data framework, thus this research can be a starting point for further developments.

The business part of interest for bike sharing companies is that they can leverage on algorithms in order to provide a stable supply, cut costs where needed and effectively allocate resources. Considering that the forecasts horizon is very tight (companies need real time adjustments on a hourly basis), further developments can consider hybrid models that combine both standard econometrics and Machine/Deep Learning tools, building a hybrid model that can better leverage on their respective properties. For instance, (Yu et al., 2023) developed a hybrid SARIMA-LSTM model that accurately forecasts bikes on a hourly basis. The ensemble model achieved better results than the respective ones alone. Considering the lack of homoskedasticity in some periods of the year, a similar strategy can be applied to the models considered in this study as well, by implementing a median model. Other further steps could be related to expanding the research with more

years as well as by including spatial features that are not considered in this research.

A special mention goes to the potential contribution of this research to the achievement of several Sustainable Development Goals (SDGs), a UN framework that has become a guidance for the citizenship, companies and countries in terms of sustainability. There are 17 goals set and they are part of the 2030 Agenda for Sustainable Development. Overall, the Agenda is an action plan for the earth, people and prosperity. The aim is to provide a guidance regarding the global challenges that we are facing and that will occur in the next decade, by involving countries and stakeholders in a collaborative partnership. Forecasting bike sharing demand can be potentially helpful for several goals: ensuring good health and well-being, making energy clean and affordable, fostering innovation and infrastructures, taking climate action and it can be an useful resource in the development of sustainable cities and communities. The aforementioned goals are respectively 3,7,9, 13 and 11 in the SDGs framework.

It is acknowledged that physical activity can reduce stress as well as the prevalence of many diseases such as obesity, diabetes and heart issues, promoting a healthier lifestyle *(Good Health and Well Being - SDG n.3)*. A stable supply of bicycles and a better planning can also reduce the number of vehicles on the roads, impacting in a positive way the number of car accidents.

There are some examples that deserve to be mentioned and that can be considered as benchmarks in this context. For instance, Copenhagen and Amsterdam are globally recognized for their extensive bicycles' network and for their cycling culture, representing the forefront of sustainability in Europe as well as being the demonstration of what can be achieved by leveraging on data analytics.

The latter can be effective in the reduction of fossil-fuel reliance, greenhouse gases emissions and in the development of sustainable urban cities. An efficient distribution of bicycles in the city means that fewer vehicle trips are needed, reducing the operational emissions of the service. Through an integration with public transportation, accurate demand forecasting ensures that bicycles are available at crucial transit hubs, solving the "last mile" problem. Commuters can complete the first and last part of their journey without relying on cars, reducing their impact on emissions as well as on pollution. Air quality is strictly related to the latter, leading to a an enhanced quality of life.

These solutions are related with the achievement of SDGs n.7,9,13:

Affordable and Clean Energy, Industry & Innovation and Infrastructure, Climate Action. Ensuring bike availability can be a huge resource for underserved and marginalized neighbourhoods, providing an alternative way of transportation where there may be not many left. Bike sharing represents a cheaper way of moving with respect to more expensive ones, such as cars. A stable and extensive network ensures equal opportunities, by guaranteeing accessibility to those services which may be not present in poorer and marginalized areas. Thus, it can be helpful to promote accessibility and inclusivity, which are important for the development of *Sustainable Cities and Communities (SDG n.11).*

Communities have the responsibility to guarantee good education, accessible healthcare and basic services (such as a reliable transportation network) to everybody. Whenever public authorities could not guarantee the aforementioned services, partnerships between public governments and privates could be implemented in order to achieve those goals. Overall, cities can adopt mixed-use developments that set cycling and walking as priorities. Fostering a cycling culture can produce a chain of benefits that work together in the development of a carbon neutral future and a more sustainable way of living.

References

N. Boufidis, A. Nikiforiadis, K. Chrysostomou, and G. Aifadopoulou. Development of a station-level demand prediction and visualization tool to support bike-sharing systems' operators. *Transportation Research Procedia*, 47:51–58, 2020. ISSN 23521465. doi: 10.1016/j.trpro.2020.03.072.

L. Chen, D. Zhang, L. Wang, D. Yang, X. Ma, S. Li, Z. Wu, G. Pan, T.-M.-T. Nguyen, and J. Jakubowicz. Dynamic cluster-based over-demand prediction in bike sharing systems. pages 841–852. ACM, 9 2016. ISBN 9781450344616. doi: 10.1145/2971648.2971652.

Z. Chen, D. van Lierop, and D. Ettema. Dockless bike-sharing systems: what are the implications? *Transport Reviews*, 40:333–353, 5 2020. ISSN 0144-1647. doi: 10.1080/01441647.2019.1710306.

P. DeMaio. Bike-sharing: History, impacts, models of provision, and future. *Journal of Public Transportation*, 12:41–56, 12 2009. ISSN 1077-291X. doi: 10.5038/2375-0901.12.4.3.

E. Eren and V. E. Uz. A review on bike-sharing: The factors affecting bike-sharing demand. *Sustainable Cities and Society*, 54:101882, 3 2020. ISSN 22106707. doi: 10.1016/j.scs.2019.101882.

S. Feng, H. Chen, C. Du, J. Li, and N. Jing. A hierarchical demand prediction method with station clustering for bike sharing system. pages 829–836. IEEE, 6 2018. ISBN 978-1-5386-4210-8. doi: 10.1109/DSC.2018.00133.

E. Fishman. Bikeshare: A review of recent literature. *Transport Reviews*, 36:92–113, 1 2016. ISSN 0144-1647. doi: 10.1080/01441647.2015.1033036.

R. Hyndman and G. Athanasopoulos. *Forecasting: principles and practice*. 3rd edition, 2021.

G. James, D. Witten, T. Hastie, R. Tibshirani, and J. Taylor. *An Introduction to Statistical Learning*, pages 1–13. 2023. doi: 10.1007/978-3-031-38747-0_1.

A. Kaltenbrunner, R. Meza, J. Grivolla, J. Codina, and R. Banchs. Urban cycles and mobility patterns: Exploring and predicting trends in a bicycle-based public transport

system. *Pervasive and Mobile Computing*, 6:455–466, 8 2010. ISSN 15741192. doi: 10.1016/j.pmcj.2010.07.002.

K. Kim. Investigation on the effects of weather and calendar events on bike-sharing according to the trip patterns of bike rentals of stations. *Journal of Transport Geography*, 66:309–320, 1 2018. ISSN 09666923. doi: 10.1016/j.jtrangeo.2018.01.001.

G. McKenzie. Docked vs dockless bike-sharing: Contrasting spatiotemporal patterns. 2018.

C. Peláez-Rodríguez, J. Pérez-Aracil, D. Fister, R. Torres-López, and S. Salcedo-Sanz. Bike sharing and cable car demand forecasting using machine learning and deep learning multivariate time series approaches. *Expert Systems with Applications*, 238:122264, 3 2024. ISSN 09574174. doi: 10.1016/j.eswa.2023.122264.

S. Si, H. Chen, W. Liu, and Y. Yan. Disruptive innovation, business model and sharing economy: the bike-sharing cases in china. *Management Decision*, 59:2674–2692, 10 2021. ISSN 0025-1747. doi: 10.1108/MD-06-2019-0818.

M. Winters, G. Davidson, D. Kao, and K. Teschke. Motivators and deterrents of bicycling: comparing influences on decisions to ride. *Transportation*, 38:153–168, 1 2011. ISSN 0049-4488. doi: 10.1007/s11116-010-9284-y.

J. W. Yoon, F. Pinelli, and F. Calabrese. Cityride: A predictive bike sharing journey advisor. pages 306–311. IEEE, 7 2012. ISBN 978-1-4673-1796-2. doi: 10.1109/MDM.2012.16.

L. Yu, T. Feng, T. Li, and L. Cheng. Demand prediction and optimal allocation of shared bikes around urban rail transit stations. *Urban Rail Transit*, 9:57–71, 3 2023. ISSN 2199-6687. doi: 10.1007/s40864-022-00183-w.

A Appendix

Contents

A.1 Raw Dataset

Table 2: Raw dataset

Variable Name	Type	Unit	Description
Date	Date		Date in year-month-day format
Rented Bike Count	Integer		Count of bikes rented at each hour
Hour	Integer		Hour of the day
Temperature	Continuous	°C	Temperature in Celsius
Humidity	Integer	%	Level of humidity in %
Wind Speed	Continuous	m/s	Level of wind speed in m/s
Visibility	Integer	10m	Visibility measured at 10m of distance
Dew point temperature	Continuous	°C	Dew point temperature in °C
Solar radiation	Continuous	m/s	Solar radiation in m/s
Rainfall	Integer	mm	Amount of rain in mm
Snowfall	Categorical	cm	Level of snow in cm
Seasons	Categorical		Name of the season
Holiday	Binary		Whether it is holiday or not
Functioning Day	Binary		Whether it is functioning day or not

A.2 Data pre-processing & EDA in Python

```
#importing libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
data=pd.read_csv("SeoulBikeData.csv", encoding="unicode-escape")
data.shape

#variable names
data.columns

#first columns
data.head()

#overview
data.describe()

#null/zeros in dataset
data.isna().sum()

#some zeros detected
(data==0).sum()

#converting date in datetime
data['Date'] = pd.to_datetime(data['Date'], format='%d/%m/%Y')

#creating a datetime index on hourly basis
data['Datetime']= data.apply(lambda x:x['Date'] +
pd.Timedelta(hours=int(x['Hour'])), axis=1)
data.set_index('Datetime', inplace=True)

#creating temporal features

data['Dayofweek'] = data.index.dayofweek
data['Quarter']=data.index.quarter
data['Month']=data.index.month
data['Dayofyear']=data.index.dayofyear
```

```
## Daily Aggregating

#aggregating on a daily basis

ag_data=data.groupby('Date').mean('Rented Bike Count')
ag_data

#concatenating seasons on daily dataset
seasons_data=data[['Date','Seasons']]
seasons_data.drop_duplicates(subset='Date',inplace=True)
seasons_data.set_index('Date', inplace=True)
ag_data=pd.merge(ag_data,seasons_data,on='Date')

#value count of seasons
ag_data['Seasons'].value_counts()

#dealing with zeros
ag_data['Rented Bike Count']=ag_data['Rented Bike Count'].replace(0,0.1)

#no zeros remained
ag_data['Rented Bike Count'].isna().sum()

## Log Scaling Target Variable & 1Day Differencing

#log scaling target variable on aggregated data
ag_data['log_rbc']=ag_data['Rented Bike Count'].apply(lambda x: np.log(x))

#log scaling on hourly data
data['Rented Bike Count']=data['Rented Bike Count'].replace(0,0.1)
data['log_rented_hrl']=
data['Rented Bike Count'].apply(lambda x: np.log(x))
```

```
# day differencing
ag_data['sdd']=ag_data['log_rbc'].diff(periods=1)

#preprocessed data as an input for RStudio modelling phase
ag_data.to_csv("prepro-data.csv")

# Data Visualization - EDA

## Hourly vs Daily (Aggregated) Time Series

#comparing original time series and aggregated time series
fig=plt.figure(figsize=(17,8))
ax1=fig.add_subplot(2,1,1)
plt.plot(data['Rented Bike Count'], axes=ax1)
ax1.set_title("Bike Rented Hourly in Seoul in 2018")
ax2=fig.add_subplot(2,1,2)
ax2.set_title('Bike Rented Daily in Seoul in 2018')
plt.plot(ag_data['Rented Bike Count'], axes=ax2)

## Log Scaling and Differencing Target Variable

#comparing original target variable distribution
fig=plt.figure(figsize=(17,10))
plt.title
('Impact of Log-Scaling and Differencing on Rented Bike Distribution')
ax1=fig.add_subplot(2,1,1)
ax2=fig.add_subplot(2,1,2)
ax2.set_title('After')
sns.histplot(data['Rented Bike Count'], ax=ax1,kde=True)
sns.histplot(ag_data['sdd'], ax=ax2,kde=True)
```

```
## Normal Time Series vs 1d Differenced Time Series

#normal time series vs 1D differencing
fig=plt.figure(figsize=(17,8))
ax1=fig.add_subplot(2,1,1)
plt.plot(ag_data['Rented Bike Count'], axes=ax1)
ax1.set_title("Aggregated Time Series")
ax2=fig.add_subplot(2,1,2)
ax2.set_title('Differenced Time Series (1D changes)')
ax2.plot(ag_data['sdd'], color='orange')
#plt.plot(data['Rented Bike Count'])
plt.savefig('sdd_vs_ag.png',format='png')

## Correlation Matrix

#studying correlation between target variable and covariates
fig=plt.figure(figsize=(13,10))
cor_matrix=data.corr()
hm_matrix=sns.heatmap(cor_matrix,annot=True,linewidth=2, cmap='coolwarm')
hm_matrix.set_title('Correlation Matrix')
plt.tight_layout()
plt.savefig('corr_matrix.jpg')

## Inspecting Temperature behaviour overtime

#noticing the quite good correlation with temperature
we plot the variables (target and temperature) against each other
fig=plt.figure(figsize=(19,8))
ax1=fig.add_subplot(1,1,1)
ax2=ax1.twinx()
ax1.set_title('Comparison -- Temperature vs Bike Rented overtime')
#ax2=fig.add_subplot(2,1,2)
```

```
#ax2.set_title('Bike Rented overtime')
ax1.plot(ag_data['Temperature( C )'],color='orange',label='temperature')

ax2.plot(ag_data['Rented Bike Count'], label='bike rented' )
#ax2.legend()
ax1.legend()
#fig.legend()
plt.savefig('temp_vs_bike.png',format='png')

#scatterplot
fig=plt.figure(figsize=(12,8))
plt.scatter(ag_data['Rented Bike Count'], ag_data['Temperature( C )'])
m,b=np.polyfit(ag_data['Rented Bike Count'],
                ag_data['Temperature( C )'],1)
plt.plot(ag_data['Rented Bike Count'],
m*ag_data['Rented Bike Count']+b, color='orange')
fig.suptitle('Scatter Plot — Temperature vs Bike Rented')
plt.savefig('scatter.png',format='png')

## Inspecting Time Seasonalities

#plot to see if there's some type of seasonality in the data
fig,ax=plt.subplots(2,2, figsize=(17,10))

#hourly seasonality
ax[0,0].set_title('Bike Rented by Hour')
sns.boxplot(data=data, x='Hour',y='Rented Bike Count', ax=ax[0,0])

#dayofweek
sns.boxplot(data=data, x='Dayofweek',y='Rented Bike Count', ax=ax[0,1])
ax[0,1].set_title('Bike Rented by Dayofweek')
```

```
#month
sns.boxplot(data=data, x='Month',y='Rented Bike Count', ax=ax[1,0])
ax[1,0].set_title('Bike Rented by Month')

#seasons
sns.boxplot(data=data, x='Seasons',y='Rented Bike Count', ax=ax[1,1])
ax[1,1].set_title("Bike Rented by Seasons")

#save it
plt.savefig('TimeSeasonal.png',format='png')
```

A.3 Pre-processed variables

Table 3: New variables

Variable Name	Type	Description
Dayofweek	Integer	Number of the day in the week
Month	Integer	Number of the month in the year
Quarter	Integer	Number of the quarter in the year
Dayofyear	Integer	Number of the day in the year
log rbc	Continuous	Natural logarithm of *Rented Bike Count*
sdd	Continuous	One-day differences in *log rbc*
Winter dummy*	Binomial	Dummy for *Winter* season
Spring dummy*	Binomial	Dummy for *Spring* season
Summer dummy*	Binomial	Dummy for *Summer* season
Autumn dummy*	Binomial	Dummy for *Autumn* season

*dummies are created on R in the subsequent phase of *Modelling & Valuation in R*

A.4 Modelling & Valuation in R

```
#importing libraries
```

```
library(fpp3)
library(e1071)
library(patchwork)
library(knitr)
library(kableExtra)

#reading pre processed file
df <- read.csv('prepro-data.csv',fileEncoding = "latin1")
df <- drop_na(df)
df<-rename(df, "temp"="Temperature.    .C.",
                "windspeed"="Wind.speed..m.s.",
                "solaradiation"="Solar.Radiation..MJ.m2.",
                "visibility"="Visibility..10m.",
                "humidity"="Visibility..10m.",
                "dewpoint"= "Dew.point.temperature.    .C.",
                "snowfall"= "Snowfall..cm."
                 )

#variables
colnames(df)

#make Datetime readble
df$Date <- as.Date(df$Date)

#checking class
class(df$Date)

#make it a tsibble
df.ts<-df %>% as_tsibble(index=Date)

#PACF and ACF
daily_acf<-df.ts %>%
```

```
  ACF(sdd) %>% autoplot()+labs(title="Daily ACF")

daily_pacf<-df.ts %>%
  PACF(sdd) %>% autoplot()+labs(title='Daily PACF')

combined_auto<-daily_acf/daily_pacf
combined_auto
#ggsave(filename='combined_auto.png',plot=combined_auto)

#creating Seasonal dummies
df.ts<-df.ts %>% mutate(
                    "Winter_dummy"=ifelse(Seasons =="Winter",1,0),
                    "Spring_dummy"=ifelse(Seasons =="Spring",1,0),
                    "Summer_dummy"=ifelse(Seasons =="Summer",1,0),
                    "Autumn_dummy"=ifelse(Seasons =="Autumn",1,0))

#splitting train and test (80% train and 20%)
threshold<-round(nrow(df.ts)*(80/100))
train<-df.ts%>% slice(1:threshold)
test<-df.ts %>% slice((threshold+1):nrow(df.ts))



# Modelling and Forecasting with SVR

#modelling with SVR
mod_svm<-svm(sdd ~ Winter_dummy+Spring_dummy+
Summer_dummy+Month+temp, data=train)

#predicting
fore_svm<-predict(mod_svm,newdata=test)

#errors
```

```
error_svm <- test$sdd - fore_svm

#fitting parametric models
training<-train %>%
model ("SARIMAX_110"=ARIMA(sdd ~ pdq(3,1,1)+PDQ(1,1,0)+Month+temp),
       "SARIMAX_011"=ARIMA(sdd ~ pdq(3,1,1)+PDQ(0,1,1)+Month+temp),
       "Linear Model"=TSLM(sdd ~ Winter_dummy+Spring_dummy+
       Summer_dummy+Month+temp),
       )


# Residual Analysis

#SARIMA_110 Residual Analysis
aug_sarima110<-training[1] %>% augment()
res_sa110_overtime<-aug_sarima110 %>% autoplot(.innov)
res_sa110_hist<-aug_sarima110 %>%
                ggplot(aes(.innov))+geom_histogram(bins=30)
res_sa110_acf<-aug_sarima110 %>% ACF(.innov) %>% autoplot()
res_sa110_pacf<-aug_sarima110 %>% PACF(.innov) %>% autoplot()

sa110_res_analysis<-(res_sa110_overtime+res_sa110_hist)/
                    (res_sa110_acf+res_sa110_pacf)
                    +plot_annotation(
                    title='SARIMAX_110 - Residual Analysis')
sa110_res_analysis
#ggsave('sa110_res_analysis.png',plot=sa110_res_analysis)

#SARIMA_011 Residual Analysis
aug_sarima011<-training[2] %>% augment()
res_sa011_overtime<-aug_sarima011 %>% autoplot(.innov)
res_sa011_hist<-aug_sarima011 %>%
```

```
                    ggplot(aes(.innov))+geom_histogram(bins=30)
res_sa011_acf<-aug_sarima011 %>% ACF(.innov) %>% autoplot()
res_sa011_pacf<-aug_sarima011%>% PACF(.innov) %>% autoplot()

sa011_res_analysis<-(res_sa011_overtime+res_sa011_hist)/
                        (res_sa011_acf+res_sa011_pacf)
                        +plot_annotation(
                        title='SARIMAX_011 - Residual Analysis')

sa011_res_analysis
#ggsave('sa011_res_analysis.png',plot=sa0011_res_analysis)

#Linear Model Residual Analysis
aug_tslm<-training[3] %>% augment()
res_tslm<-aug_tslm %>% autoplot(.innov)
res_tslm_hist<-aug_tslm %>% ggplot(aes(.innov))+geom_histogram(bins=30)
res_tslm_acf<-aug_tslm %>% ACF(.innov) %>% autoplot()
res_tslm_pacf<-aug_tslm%>% PACF(.innov) %>% autoplot()

(res_tslm+res_tslm_hist)/(res_tslm_acf+res_tslm_pacf)
+plot_annotation(title='Linear Model - Residual Analysis')

#SVR Residual Analysis
res_svm<-mod_svm %>% residuals()
fitted_svm<-mod_svm %>% fitted()
svm_df<-data.frame(train$Date,fitted_svm,res_svm) %>%
                    as_tsibble(index=train.Date)
svm_df_overtime<-svm_df %>% autoplot(res_svm)
svm_df_hist<-svm_df %>% ggplot(aes(x=res_svm))+geom_histogram()
svm_df_acf<-svm_df %>% ACF(res_svm) %>% autoplot()
svm_df_pacf<-svm_df %>% PACF(res_svm) %>% autoplot()
```

```
SVR_res_analysis<-(svm_df_overtime+svm_df_hist)/
                  (svm_df_acf+svm_df_pacf)
                  +plot_annotation(title='SVR - Residual Analysis')
SVR_res_analysis

# Forecasting on test data

#Parametric models forecasts on test set
train.results<-training %>% forecast(test)

#visualizing
train.results %>%
            autoplot(df.ts, level=NULL)
            +plot_annotation(title = 'Forecasts vs Actuals - Parametric')

#tibble with SVM metrics
MSE_svm<-sum(error_svm*error_svm)/nrow(test)

svm_eval<-tibble(.model="SVR",
                 MAE= sum(abs(error_svm))/nrow(test),
                 RMSE=sqrt(MSE_svm),
                 MAD=median(error_svm))



#tibble with SVR
performance_metrics<-train.results %>%
  fabletools::accuracy(test, list(MAE = MAE, RMSE = RMSE)) %>%
  select(-.type) %>%
  merge(
    tibble(.model = train.results$.model,
           forecast_error = train.results$.mean - test$sdd) %>%
      group_by(.model) %>%
```

```
        summarise(MAD = median(abs(forecast_error)))
    ) %>%
    relocate(.model, MAD) %>% rbind(svm_eval) %>% arrange(MAD)

performance_metrics
```