Predictive Modelling of Air Quality in Madrid

Undergraduate Dissertation

Business Analytics

Student: Blanca Herreros de Tejada Lobo

Tutor: Ana Lazcano de Rojas

ACKNOWLEDGEMENTS

This undergraduate dissertation reflects the work I have done during my time at the WorldWide Observatory for Smart Cities. My mentor José Antonio Ondiviela and my partner Marta Meneses have helped me uncover a passion for understanding city dynamics - I have embarked on the incredible journey of Smart Cities. Cities are much more than cluster concrete structures. A city is its people, its culture, its dynamism, and its sustainability. This list could go on endlessly.

I would like to thank Ana Lazcano for guiding and supporting me throughout the process of creating this project. I am grateful for her recommendations, her time, and her patience. She has helped make this Undergraduate Thesis a reality.

Finally, I would like to thank my family. Without them, I would not be who I am or where I am. I thank my mother for teaching me that the best reward one can aspire to is to reap the fruits of one's efforts. I thank my father for teaching me that passion, patience, and self-confidence make everything possible. I thank my sister for teaching me to grow, always appreciating the good things I have.

ABSTRACT

As global concerns about climate change and deteriorating air quality intensify, the European Environment Agency (EEA) and other international organizations are making copious efforts to undo the damage that so many human activities and industries have done to our ecosystems, especially to the air we breathe.

The Barcelona Institute for Global Health (*Instituto de Salud Global de Barcelona*) annually publishes a ranking that studies mortality attributable to air pollution in more than 1,000 European cities. The Spanish capital, Madrid, leads the ranking associated with deaths caused by nitrogen dioxide.

This end-of-degree dissertation provides a holistic assessment of Madrid City Council's current air quality system. It is demonstrated that this system is rather rudimentary and needs urgent actualization. Not only is this air quality control system only composed of 24 static measurement stations, but also, the data is vastly incomplete. Furthermore, two predictive models have been developed (an ARIMA Time Series and an LSTM recurrent neural network) to study how time series models adapt to this type of data. These models highlight the importance for Madrid's City Council to have a robust air quality control system. The results of both predictive models are used to make recommendations to the City Council on improving its air quality system. A stronger air quality system will allow Madrid's City Council to act proactively in reducing pollution and making efficient energy use.

Keywords: LSTM, Recurrent Neural Network, Time Series Model, ARIMA, Nitrogen Dioxide, Madrid

Universidad Francisco de Vitoria UFV Madrid

sas

ÍNDICE

Figure Index

Table Index

1. INTRODUCTION

In October 1948, a town called Donora (PA, United States) found itself enclosed by a poisonous fog. This provoked the death of 20 people. Likewise, it generated cardiovascular and respiratory diseases among more than 14,000 inhabitants of the town.

Donora was an industrial enclave housing multiple zinc smelting plants and steel mills. Poor weather conditions and the town's unfortunate location between hills led to a dense cloud (containing sulfuric acid and carbon monoxide) forming over the town. Despite the worrying circumstances, the industrial plants kept operating. After five days, only the rain successfully diluted the contaminants suspended in the air.

This event triggered numerous activist movements raising awareness on the importance of unpolluted air for health. Likewise, this led to the first regulations concerning pollutants in the atmosphere. The Air Pollution Control Act of 1955 facilitated the financing of federal research projects on pollution and air quality. This decree motivated many others from different states and countries. Several organizations are currently carrying out exhaustive controls on the level of contamination in different areas of the world. The three predominant institutions are the European Environment Agency (EEA), China National Environmental Monitoring Center (CNEMC), and the Environmental Protection Agency (EPA) in the United States.

Today, climate change is omnipresent. Authorities and institutions are making efforts to improve their sustainable footprint. Among other examples is the 2030 Agenda for Sustainable Development. Approved by the UN in 2015, it envisions a kinder and safer world for everyone. Of the seventeen objectives described by the United Nations, seven refer to improving the environment. Thus, demonstrating the weight and relevance of the topic.

An aspect of extreme concern relating to climate change is atmospheric pollution. The dirt found in the air is not only responsible for respiratory diseases, but also leads to neurodegenerative illnesses and reproductive problems, among others. The WHO “recognizes that air pollution is a critical risk factor for noncommunicable diseases (NCDs), accounting for an estimated one quarter (24%) of all adult deaths from heart disease, 25% of stroke deaths, 43% of chronic obstructive pulmonary disease deaths, and 29% of lung cancer deaths” (WHO, 2018). In turn, this leads to incremental economic costs. Currently, the expense incurred in health by diseases directly related to poor air quality is estimated to oscillate between 2 and 4 billion dollars annually.

The Madrid City Council has a sensor system that monitors the level of various polluting gases in the air. Their objective is to carry a continuous and exhaustive control of air quality. Additionally, regulations enforce the city council to implement several action plans to avoid gases exceeding their recommended value.

2. THEORETIC BACKGROUND

2.1. REGULATION

Air pollution is any alteration of the natural characteristics of the atmosphere caused by a chemical, physical or biological agent. These agents can arise naturally or be caused by human activities. The pollutants that generate the greatest concern (and are therefore subject to stricter regulations) due to their impact on our health and the environment are:

1. Suspended particles (PM2.5 & PM10).
2. Ozone (O_3)
3. Nitrogen Dioxide (NO_2)
4. Sulfur Dioxide (SO_2)
5. Carbon monoxide (CO)

All these air pollutants, at off-target levels, markedly increase morbidity and mortality rates. According to the WHO, air pollution causes approximately 4.2 million premature deaths per year. That is why the adoption of policies support the use of cleaner modes of transport, improved energy efficiency in housing, electricity generation and industry, and better management of municipal waste would reduce some of the main sources of air pollution in cities (WHO, 2021). The following is a study of the regulations (both European and Spanish) related to air quality requirements.

2.1.1. EUROPEAN LEGISLATION

European legislation includes four main directives on air quality:

1. Directive 2008/50/CE[1]

This directive replaces the Macro Directive and its three previously relevant daughter directives:

- Directive 96/62/CE[2]
- Directive 1999/30/CE[3]
- Directive 2000/69/CE[4]
- Directive 2002/CE[5]

[1] Directive 2008/50/CE of the European Parliament and of the Council of 21 May 2008 on ambient air quality and cleaner air for Europe.

[2] Directive 96/62/CE of the Council of 27 September 1996 on ambient air quality assessment and management (former Framework Directive)

[3] Directive 1999/30/CE of the Council Directive of April 22, 1999 relating to limit values for sulfur dioxide, nitrogen dioxide and oxides of nitrogen, particulate matter and lead in ambient air (1st "daughter" Directive)

[4] Directive 2000/69/CE of the European Parliament and of the Council of 16 November 2000 relating to limit values for benzene and carbon monoxide in ambient air (2nd "daughter" Directive)

[5] Directive 2002/CE of the European Parliament and of the Council of 12 February 2002 relating to ozone in ambient air (3rd "daughter" Directive).

Following WHO recommendations, this law establishes objectives and requirements relevant to air quality assessment in Europe. In addition, among other measures, a number of restrictions on airborne particles smaller than 2.5 micrometers were introduced.

2. Directive 2004/107/CE[6]

This law was derived from the original Macro Directive. Its main objective is to control the levels of certain pollutants in the air, thus reducing the adverse effects caused by them. Among others, the pollutants mentioned in this directive include polycyclic aromatic hydrocarbons. These gases arise through the incomplete combustion of organic matter. In addition, we find restrictions for nickel and arsenic.

3. Directive 2015/1480/EU[7]

Amends various annexes of Directive 2008/50, establishing guidelines for air quality measurement centers, reference methods and data validation.

4. Commission Implementing Decision 2011/850/EU[8]

Establishes the format in which the European Commission states shall notify the European Commission of their ambient air quality. It also establishes the measurement protocol for each pollutant.

- Council Decision 97/101/EC[9]
- Commission Decision 2004/224/EC[10]
- Commission Decision 2004/461/EC[11]

2.1.2. NATIONAL LEGISLATION

In addition, Spanish regulations on air quality include the following guidelines:

1. Act 34/2007[12]:

Through this law, the government aims to maintain optimal levels of polluting gases, thus reaching an acceptable level of air quality, and reducing the possible risks that these may have on

[6] Directive 2004/107/CE of the European Parliament and of the Council of 15 December 2004 relating to arsenic, cadmium, mercury, nickel and polycyclic aromatic hydrocarbons in ambient air.

[7] Directive 2015/1480/EU, of the Commission of 28 August 2015 amending several annexes to Directives 2004/107/CE and 2008/50/CE of the European Parliament and of the Council laying down rules on reference methods, data validation and location of sampling points for ambient air quality assessment.

[8] Commission Implementing Decision 2011/850/EU of 12 December 2011 laying down provisions for Directives 2004/107/EC and 2008/50/EC of the European Parliament and of the Council as regards reciprocal exchange of information and reporting on ambient air quality.

[9] Council Decision 97/101/EC of 27 January 1997 establishing a reciprocal exchange of information and data from networks and individual stations measuring ambient air pollution in the Member States.

[10] Commission Decision 2004/224/EC of 20 February 2004 laying down arrangements for the submission of information on plans or programs required under Council Directive 96/62/EC relating to limit values for certain pollutants in ambient air.

[11] And Commission Decision 2004/461/EC of 29 April 2004 on the questionnaire to be used for annual reporting on ambient air quality assessment under Council Directives 96/62/EC and 1999/30/EC and Directives 2000/69/EC and 2002/3/EC of the European Parliament and of the Council.

[12] Law 34/2007, of November 15, 2007, on air quality and atmospheric protection.

health. Through this law the government defines the stipulation of air quality systems and defines the regulatory scope for the elaboration of air quality plans.

2. Real Decreto 102/2011[13]

Esta normativa adapta las directivas europeas del 2008/50/CE y 2004/107/CE al sistema jurídico español. Por lo tanto, esta directriz tiene el principal objetivo de reducir los riesgos medioambientales y para la salud humana de los contaminantes descritos en el dictamen. Posteriormente, el Real Decreto fue modificado en varias ocasiones:

- el Real Decreto 678/2014[14]
- el Real Decreto 39/2017[15]

This regulation adapts the European Directives 2008/50/EC and 2004/107/EC to the Spanish legal system. Therefore, this directive has the main objective of reducing the environmental and human health risks of the pollutants described in the opinion. Subsequently, the Royal Decree was amended on several occasions:

- Royal Decree 678/2014
- Royal Decree 39/2017

3. Order TEC/351/2019[16]

Finally, Order TEC/351/2019, following the guidelines of the European air quality index ("Air Quality Index"), approves the National Air Quality Index. Using a common air quality index facilitates the understanding of the air quality system by the citizens.

[13] Royal Decree 102/2011, of January 28, on the improvement of air quality.

[14] Royal Decree 678/2014, of August 1, amending Royal Decree 102/2011, of January 28, on the improvement of air quality, to modify the quality objectives for carbon sulfide established in the single transitory provision.

[15] Royal Decree 39/2017, of January 27, amending Royal Decree 102/2011, of January 28, on the improvement of air quality, to transpose into Spanish law Directive 2015/1480, which establishes rules on reference methods, data validation and location of measurement points for the assessment of ambient air quality and incorporates the new information exchange requirements established in Decision 2011/850/EU. In addition, this Royal Decree provides the approval of a National Air Quality Index to inform citizens, in a clear and homogeneous manner throughout the country, about the quality of the air they are breathing at any given moment.

[16] Order TEC/351/2019, of March 18, approving the National Air Quality Index.

2.2. GASES OF INTEREST TO THE STUDY

Having studied the current regulations on air quality and atmospheric pollution, we will analyze the origin and harmful impacts (both for the environment and for health) of those gases that pollute our atmosphere. According to Order TEC/351/2019, of March 18, which approves the National Air Quality Index, the following pollutants are taken into account:

2.2.1. Particulate Matter ($PM_{2,5}$ & PM_{10}):

Solid particles and liquid droplets found suspended in the air constitute airborne particles. Examples of such particles include dust, ash, and soot. There are two categorizations of airborne particles: PM2.5 and PM10. The main difference between the two is the size. The latter is composed of a set of particles with a size equal to or smaller than 10 micrometers. Contrastingly, PM2.5 is a set of particles with a dimension equal to or smaller than 2.5 micrometers. The harmful health effects of these particles are directly related to their size, since the smaller they are, the easier it is for them to penetrate our lungs. The smallest particles can even enter our bloodstream. Among other symptoms, particulate matter is an aggravating factor in cardiovascular and respiratory system diseases and can lead to heart attacks or premature death in people with pre-existing conditions. Environmentally, this type of particle contributes to the lack of visibility and the creation of acid rain in cities with a high level of pollution.

2.2.2. Ozone (O_3):

Ozone is a gas composed of three oxygen atoms. We find this gas at different altitudes in the atmosphere, predominantly in the troposphere and stratosphere. Ozone can be both harmful and beneficial; it all depends on where it is found and what its concentration is.

First, stratospheric ozone reacts with nitrogen and oxygen (abundant in this part of the atmosphere), protecting the earth from ultraviolet rays. Some adverse effects of overexposure to ultraviolet rays include blindness and skin cancer. This is why ozone is considered vital to our planet and its ecosystems.

However, tropospheric ozone has adverse effects on our health and environment. Among other symptoms, this type of ozone can aggravate the condition of asthmatic people. In addition, it can lead to irritation of the pharynx, eyes, and neck. Ground-level ozone is generated through chemical reactions between nitrogen oxides (NOx) and volatile organic compounds (VOCs). Pollutants emitted by automobile combustion, refineries, or power plants react (with each other) when exposed to long periods of solar radiation. Therefore, tropospheric ozone levels increase significantly in summer, posing a greater risk to the population.

2.2.3. Nitrogen Dioxide (NO2):

Nitrogen dioxide belongs to a group of highly reactive gases known as nitrogen oxides (NOx). This gas is mainly created by the combustion of gasoline and other types of fuels. Therefore, the main emitters of this type of pollutant are different types of motored vehicles (cars, buses, motorcycles...). Breathing air with high concentrations of NO2 can irritate our respiratory tract. Additionally, exposure to this pollutant for extended periods can aggravate certain respiratory-related diseases. In particular, nitrogen dioxide worsens asthma, causing symptoms such as coughing and shortness of breath. Occasionally, this results in hospitalization. Environmentally, nitrogen dioxide is one of the main generators of acid rain (along with sulfur dioxide).

2.2.4. Sulfur Dioxide (SO_2):

Industrial facilities, such as power plants that burn fossil fuels, are the main culprits of high levels of SO_2 in the atmosphere. Less significant emitters of sulfur dioxide include natural sources such as volcanoes and vehicles burning fuels with high sulfur content.

Exposure to SO2 for a short period can damage the human respiratory system, making it difficult to breathe. People with asthma, especially children, are extremely sensitive to the effects of SO2. High concentrations of sulfur dioxide in the air provoke the formation of other sulfur oxides (SOx). These sulfur oxides can react with other compounds in the atmosphere to form small particles. These particles contribute to particulate matter pollution (PM2.5 and PM10), penetrating deep into the lungs and contaminating our bloodstreams. Along with nitrogen dioxide, sulfur dioxide affects the acidity of rainfall. Thus, deteriorating of coastal and river ecosystems.

.

2.2.5. Carbon Monoxide (CO):

Carbon monoxide is a colorless, clear gas. If inhaled in large quantities, it can be harmful to our health. As with previously studied pollutants, this gas arises with incomplete combustion. That is when fuels such as gasoline burn without sufficient oxygen. Once again, the largest generators of this pollutant are vehicles that use carbon-based fuels (gasoline, kerosene...) as their energy source. In addition, some elements in our homes can increase the risk of inhaling this gas. A clear example is fireplaces.

Breathing air with high concentrations of CO reduces the amount of oxygen transported through our bloodstream to critical organs such as the heart and brain. High levels of CO can cause dizziness, confusion, loss of consciousness, and even death.

Fortunately, open spaces limit the amount of CO available to inhale. However, elevated levels of carbon monoxide outdoors can be of concern for people with cardiovascular diseases. People who have a reduced ability to transport oxygenated blood to their hearts in situations where the heart needs more oxygen than usual are especially vulnerable to the effects of CO when exercising or under increased stress.

2.3. AIR QUALITY SYSTEM: MADRID CITY COUNCIL

The Department of Environment and Planning of the Community of Madrid has made copious efforts to improve air quality and reduce the amount of polluting gases in the atmosphere. The current plan, called MADRID 360, "was born to reduce polluting emissions in the capital city, transforming it into a sustainable city."

At present, and as can be seen in the first image, the city of Madrid has a total of 24 static air quality monitoring stations, divided into five zones. In addition, as explained in the Madrid 360 plan, each station is categorized as Suburban, Urban Background, or Urban Traffic:

Table 1: Categories of Air Quality Stations

Type of Station	Location	Color
Suburban	Outskirts.	
Urban	Urban areas	
Urban with high levels of traffic.	Urban areas exposed to high levels of traffic	

These stations are distributed throughout the city, and in addition to measuring the previously mentioned pollutants, they also measure levels of Lead (Pb), Benzene (C6H6), Arsenic, Cadmium, Nickel and Benzo(a)pyrene.

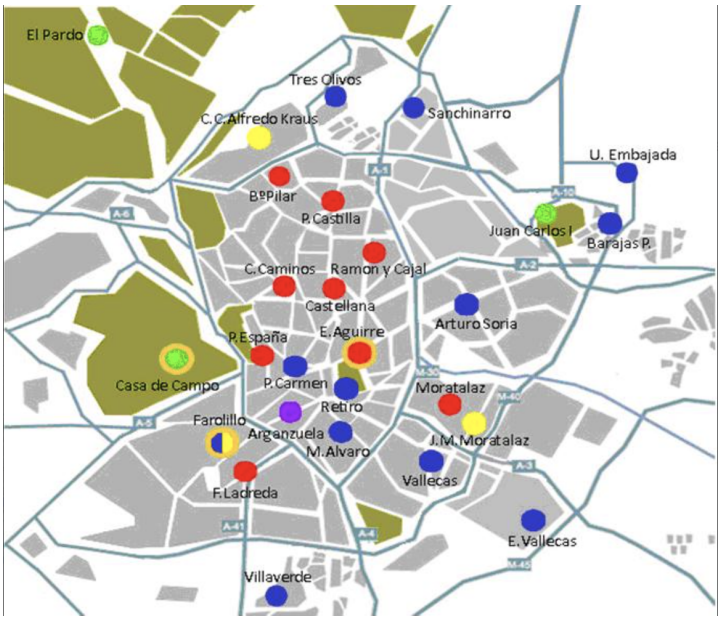

Figure 1: Corresponding Locations of Air Quality Stations

The main strategic lines of the Madrid City Council's air quality measurement system are:

- To reduce pollution by sectors of the city,
- Increase energy efficiency
- Create a reference framework for decision-making by different public administrations that use the data acquired by this network of stations.

2.4. THEORETICAL EXPLANATION OF SELECTED MODELS

2.4.1. Time Series (ARIMA)

The first model developed is a type of time series forecast. ARIMA models work with chronologically ordered sets of data (López, 2020). In other words, not only is the data relevant, but the timing of each occurrence also matters. Time series usually have three main characteristics:

- **Trend**: Long-term movement of the time series.
- **Seasonal component:** Cyclical component that repeats over time. Generally in the short term.
- **Residual (White Noise):** White noise refers to the assumption of the randomness of the data within the same series (constant mean and variance). As will be discussed below, auto-regressive (AR) and moving average (MA) make up for this lack of statistical correlation.

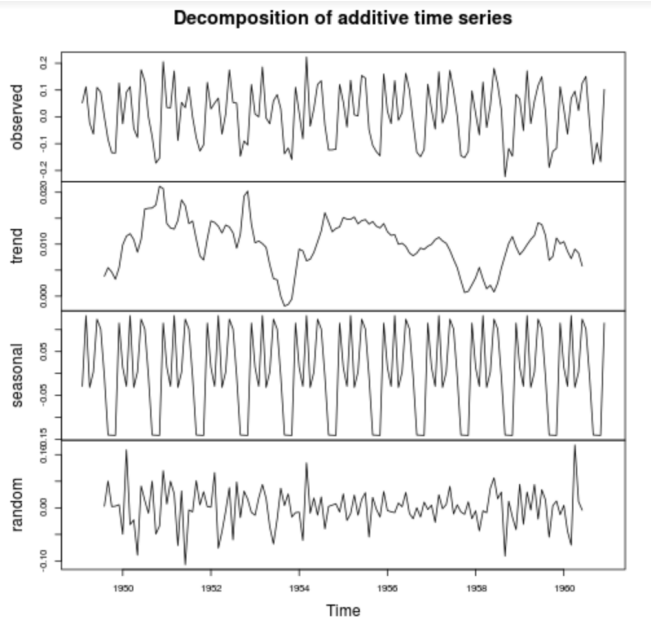

Figure 2: Time Series Decomposition

2.4.1.1. Auto-regressive Models: AR(p)

Auto-regressive models predict the future value of the dependent variable as a function of previous values of itself plus an error or random variable; AR(p) is an auto-regressive model that uses the value of p previous periods for the estimation of the dependent variable, y_t. The mathematical representation of this model is as follows:

$$y_t = \mu + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \ldots + \phi_p y_{t-p} + \epsilon_t$$

Being:

- $\mu, \phi_1, \phi_2, \phi_p$ constant parameters.
- $y_{t-1}, y_{t-2}, y_{t-p}$ the values of the dependent variable in previous lags.
- ϵ_t the error term or random variable. It is normally distributed (constant mean and covariance) and has constant variance.

An auto-regressive model of order one, AR(1), represents the simplest case of this type of model ($y_t = c + \phi_1 y_{t-1} + \epsilon_t$). It incorporates a single lag of the variable of interest plus a random error.

2.4.1.2. Moving Average Model (MA(q))

Moving average models, MA(q), present an auto-regression where the regressors are the error terms of each period *t* (Rodó, 2020). That is, the dependent variable is defined following a constant value μ and is adjusted by the error in each of the previous periods. The mathematical expression of the moving average models is as follows:

$$y_t = \mu + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \ldots + \phi_q \epsilon_{t-q} + \epsilon_t$$

Being:

- $\mu, \theta_1, \theta_2, \theta_q$ constant parameters.
- $\epsilon_{t-1}, \epsilon_{t-2}, \epsilon_{t-q}$ the residual terms in each of the preceding periods
- ϵ_t the error term or random variable. It is normally distributed with constant mean and covariance.

It is important to note that the MA(q) model is stationary. This means that each independent variable (the prior errors) is completely random. Furthermore, the set of independent observations follows a Normal Distribution (zero mean and constant variance).

Finally, for the interpretation of results in SAS, it is important to highlight that the value of theta presented in this tool is inverted.

2.4.1.3. Modelos ARMA(p,q)

The ARMA(p,q) model is a combination of a p-order auto-regressive model and a q-order moving average model. That is, in this model, in addition to incorporating p previous values of the same dependent variable, q error terms from previous estimates are also included. Its mathematical expression is:

$$y_t = \mu + \sum_{i=1}^{p} \phi_i y_{t-i} + \epsilon_t + \sum_{i=1}^{q} \theta_i \epsilon_{t-i}$$

Being:

- μ, θ_i, ϕ_q constant parameters.
- y_{t-i} the values of the dependent variable in previous lags
- ϵ_{t-i} the residual terms in each of the preceding periods.
- ϵ_t the error term or random variable. It is normally distributed with constant mean and covariance.

The simplest ARMA model, AR(1,1), takes exclusively one auto-regressive value and a single moving average value into account: $y_t = c + \phi_1 y_{t-1} + \varepsilon_t + \theta_1 \varepsilon_{t-1}$.

2.4.1.4. Stationarity

As previously mentioned, an important aspect of the ARMA(p,q) model is stationarity. In other words, it has a constant mean and variance of the data over time. The predictive ability of the model depends on the characteristic of stationarity; a constant mean, and a variance proving heteroscedasticity.

If an ARMA model is not stationary, then the data will not behave similarly in the future.

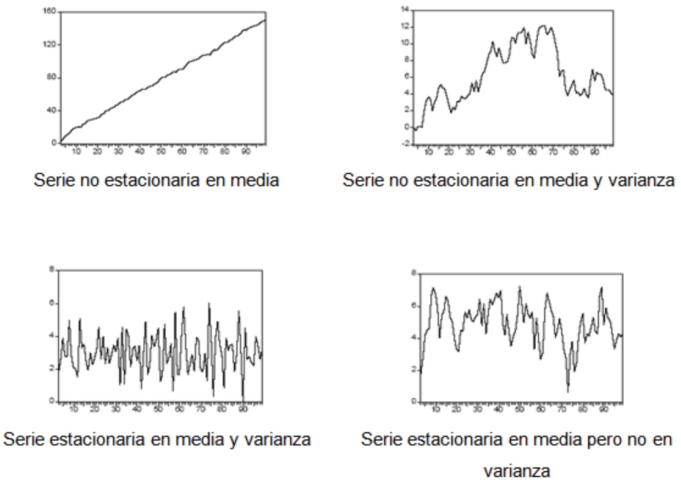

Figure 3: Stationarity Data (Parra, 2022)

The best example of stationarity is white noise decomposition within a time series. White noise follows a Normal Distribution N(μ,σ2) with covariance equal to 0.

The stationarity of the data is a factor of special importance. The time series models generated below assume the stationarity of the data used. If the characteristics of the process change over time, it will be difficult to represent the series for past and future time intervals using a simple linear model. Thus, not being able to make reliable forecasts for the variable under study (Parra, 2019). To correct for non-stationarity in a data set, differencing is used.

2.4.1.5. ARIMA (Autoregressive Integrated Moving Average Model)

An ARIMA (p,d,q) model or integrated auto-regressive moving average of order (p,d,q) model is similar to the ARMA model studied above. The difference between the two models lies in the stationarity of their data. The acronym "I" in ARIMA represents the prior integration performed on the data to ensure its stationarity. That is, "an ARIMA (p,d,q) model is obtained after applying the difference operator "d" times to a non-stationary process until arriving at a stationary and invertible ARMA (p,q) process" (Lafuente, 2020). The best example of stationarity is white noise decomposition within a time series. White noise follows a Normal Distribution N(μ,σ2) with covariance equal to 0.

2.4.1.6. ¿How do I choose the order of an ARIMA model?

2.4.1.6.1. Autocorrelation function (AFC):

The autocorrelation function measures the ratio of autocovariance of y_t and y_{t-k} divided by the variance of the dependent variable y_t. In other words, the autocorrelation function, as its name suggests, measures the level of autocorrelation between y_t and y_{t-k}.

$$FAC(k) = \rho_k = \frac{Cov(y_t, y_{t-k})}{Var(y_t)}$$

Due to the indirect effects exerted by the variables in this model, the function decreases exponentially.

2.4.1.6.2. Partial Autocorrelation Function (PACF)

Contrastingly, the partial autocorrelation function measures the direct relationship between y_t and y_{t-k}. Unlike AFC, this function does not take into account the indirect effects of intermediate variables.

$$FACP(k) = Corr[y_t - E * (y_t | y_{t-1}, \ldots, y_{t-k+1}), y_{t-k}]$$

To check whether a model is adequate, the autocorrelation function and the partial autocorrelation function are taken into account. The combined behavior of both functions describes which model will

be optimal. The joint behavior of both functions can lead to various model types, as explained in Table 2.

Table 2: Model Selection using AFC & PACF

	AR(p) or ARIMA(p,d,0)	**MA(q) or ARIMA(0,d,q)**	**ARMA(p,q)**
AFC	Decreases exponentially	After "q" significant coefficients, the rest rapidly cancels out.	Decreases Exponentially.
PAFC	After "p" significant coefficients, the rest rapidly cancels out.	Decreases Exponentially.	Decreases Exponentially.

2.4.2. LSTM (Long-Short Term Memory)

2.4.2.1. MACHINE LEARNING

As the second proposal for the air quality predictive model, a LSTM (Long-Short Term Memory) model will be used. This type of algorithm belongs to the branch of Deep Learning algorithms within the Machine Learning field.

Computers, despite having superior quantitative skills (a mathematical problem that takes us minutes to solve can be solved in nanoseconds by a computer), lack reasoning, creativity, and sentimentality. The main objective of Machine Learning is to give computers learning and reasoning capabilities that simulate those human beings possess.

2.4.2.2. TRADITIONAL DEEP LEARNING MODELS

2.4.2.2.1. Structure

Deep Learning Models present a specific branch within Machine Learning. This type of algorithms (as shown in Figure 4) have a layered structure of nodes and edges. This structure tries to simulate the distribution of neurons and their connections within our brain.

Neural networks have three layers: the Input Layer, one or more Hidden Layers, and finally the Output Layer. Broadly speaking, neural networks receive a series of inputs, which travel from the input layer, through the different hidden layers (undergoing various operations), finally generating an output.

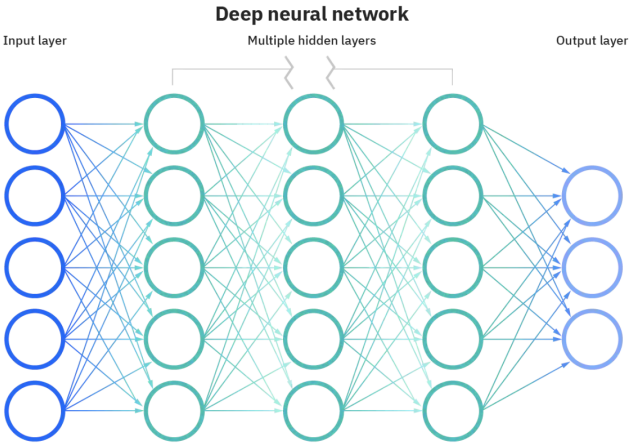

Figure 4: Structure of a Traditional Deep Neural Network (IBM Education, 2022)

In a complete neural network, each node connects to all the neurons in the previous layer. To better understand how neural networks work, it is useful to visualize each neuron as a linear regression that receives certain inputs. For example, the inputs can be the output values of the neurons in the previous layer. A neuron with three input values would have the following structure:

$$\sum_{i=1}^{m} w_i x_i + bias = w_1 x_1 + w_2 x_2 + w_3 x_3 + bias$$

The inputs received by each neuron have their respective weight (coefficient w), which can be seen as the relative importance of each input. Each weight is accompanied by the coefficient x, the activation function of the previous layer. In other words, x represents the probability that this neuron receives the input from the neuron to which each weight corresponds. In addition, each node (neuron) has a bias value (or threshold value), an indicator of the probability of activation of the neuron in question. Each linear regression generates an output, which is processed by an activation function. This function defines whether a neuron is activated, thus passing its output to adjacent layers. An example of an activation function would be:

$$output = f(x) = 1 \; if \sum_{i=1}^{m} w_i x_i + bias \geq 0$$

OR

$$output = f(x) = 0 \; if \sum_{i=1}^{m} w_i x_i + bias < 0$$

In this particular example, the activation function is linear. It passes to the next layer the information of those neurons with an output equal to or greater than zero. Other examples of activation functions include:

- **Sigmoid**

$$sigmoid(x) = \frac{1}{1+e^{-x}}$$

Being:
- e Euler's constant = 2,71828

By using this activation function, all the values of a neural network will be bounded between 0 and 1. This greatly reduces the impact that a single input variable can have on the output of a neuron.

- **Tanh**

$$tan(x) = \frac{2}{(1+e^{-2x}) - 1}$$

Being:
- e Euler's constant = 2,71828

The Tanh activation function behaves similarly to the sigmoid function. Both functions lack linearity. However, unlike the sigmoid function, when using tanh, the output of a neuron oscillates between -1 and 1.

- **Relu (Rectified Linear Unit)**

$$relu(x) = x,\ x \geq 0 \textbf{ OR } relu(x) = 0,\ x < 0$$

This last activation function only accepts information from neurons whose result is greater than or equal to zero. However, the result of this type of function is not limited to values below 1, and the result may exceed this value.

Next, the training of a neural network is studied.

2.4.2.2.2. Training the model

The learning process of a neural network consists of finding the optimal values for both the different weights of each input and the bias of each neuron. Thus, the error of the predictions generated by the model is minimized. When training a neural network, two concepts are highly relevant*: forward propagation and back propagation.*

To simplify the training explanation, a neural network with three layers (input, hidden, layer) and one neuron per layer is assumed:

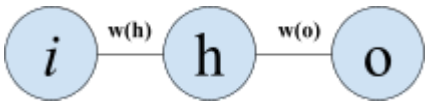

Figure 5: Neural Network Example

Relevant information of the neural network:

- An input layer, a hidden layer and an output layer. Each layer has only one neuron.
- The desired output of the model is 1.
- Each neuron depends on the weight and bias explained above, in addition to the activation function of the previous neuron. For example, the output neuron represents the function: $Output = w_o x_h + bias_o$

As previously mentioned, traditional neural networks work with information by means of *feed forward propagation*; feeding the next layer with certain inputs. When starting to train the model, the weights and bias values that determine the output of a neuron are generated randomly. Once the entire neural network has been traversed, use is made of the mean squared error (MSE) function to evaluate the accuracy of the generated results, thus calculating its error.

MSE Function:

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

Being:

- m: the number of samples used in calculating the error
- $\hat{y}$: the achieved output value
- y: the desired output value

Both $\hat{y}$ and y are linear regressions with format $y = mx + b$, as portrayed in Figure 5.

The Mean Squared Error (MSE) function calculates the sum of squared differences between the value obtained and the desired value of the neural network output. For example, taking into account the neural network in Figure 5, its cost function would be: $MSE = (o - 1)^2$; where 'o' is the output generated by the last neuron (and 1, as previously stated, is the desired value).

The lower the result of the cost function, the better trained the model is. As the model is trained, a decreasing cost function indicates "learning" or adjustment of the model to the data. As discussed below, the next training step consists of modifying the weights and bias of each neuron (within each layer) to reduce the cost function as much as possible.

The *gradient descent* algorithm adjusts the weights and bias values of the model. The objective of this process is to calculate the derivative of the cost function (in relation to w_o), thus calculating its gradient. Knowing the gradient of the MSE allows the model to know in which direction the minimum (or local minimum) of the cost function lies. This will help select which

modifications of the weights and bias value are appropriate to modify the output of the model relative to the minimum of the mean square error function.

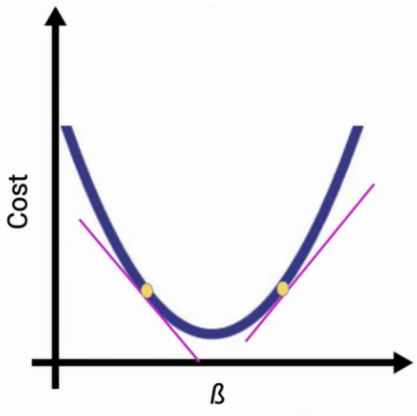

Figure 6: Gradient Descent Example

- If MSE's derivative is **negative**, the minimum (or local minima) will be to the right.
- If MSE's derivative is **positive**, the minimum (or local minima) will be to the left.

Once the first phase of forward propagation has been completed and the cost function calculated, the model will interleave this process with a process called backward propagation. Broadly speaking, backward propagation consists of modifying the weights and bias value of each neuron (in relation to the gradient descent calculated in each layer), thus improving the output result. Unlike forward propagation, this process starts at the output layer, progressively adjusting the weights and biases of the neurons in each layer until reaching the input layer. At this point, the model repeats the process as many times as it has been assigned.

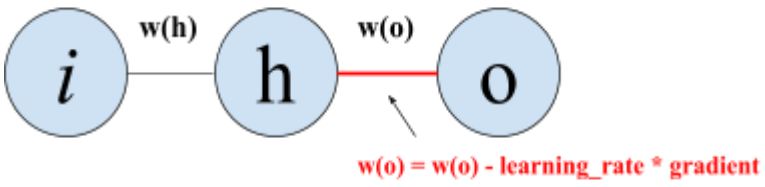

Figure 7: Backward Propagation Example

$$gradient = \frac{\vartheta(MSE)}{\vartheta w(o)}$$

In image 7, taking into account the simple neural network designed previously, it can be seen how the relative weight (**w(o)**) at the output of neuron **h** is modified as a function of the calculated gradient and a learning_rate value defined by the neural network creator (usually 0.1). This process is repeated to modify the value of **w(h)**. In this case, and using the chain rule, the gradient used to modify **w(h)** would be:

$$gradient = \frac{\vartheta(MSE)}{\vartheta w(h)} = \frac{\vartheta(MSE)}{\vartheta(h)} * \frac{\vartheta(h)}{\vartheta w(h)}$$

To conclude, the learning process of a neural network can be summarized in three steps:

- Step 1: Collect data input to the model.
- Step 2: Perform the forward propagation process, thus calculating the loss of the model.
- Step 3: Use the gradients to modify the weights and bias value of the different neurons in the model using back propagation.

2.4.2.3. RECURRENT NEURAL NETWORKS

A Recurrent Neural Network follows the general structure of a traditional Deep Learning model, using nodes and edges. This simulates, once again, the behavior of the human brain. However, recurrent neural networks are often used with temporal data. That is, this type of model assumes the existence of sequentiality in the data.

As can be seen in Figure 8, each neuron in the hidden layer has two different types of input. Firstly, they receive the output of the neurons in the previous layer (following the structure of traditional neural networks). However, there is a second input to each neuron; feedback from itself. In other words, the output generated by a neuron is reinserted as input to itself, allowing information to persist in the network.

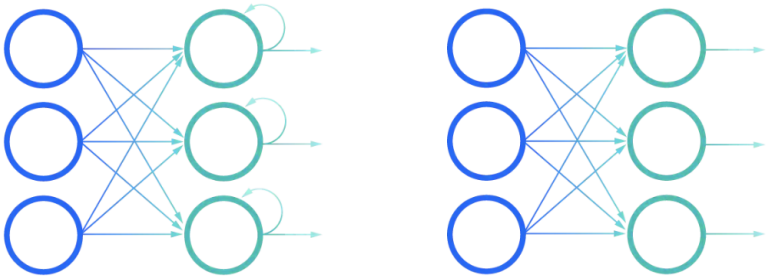

Figure 8: Recurrent Neural Networks vs. Feedforward Neural Network (IBM Education, 2022)

Figure 9 presents a recurrent neural network unwrapped over time. This image shows the inputs and outputs of a single neuron within a recurrent neural network at different time periods ($t_0, t_1, t_2 \ldots$).

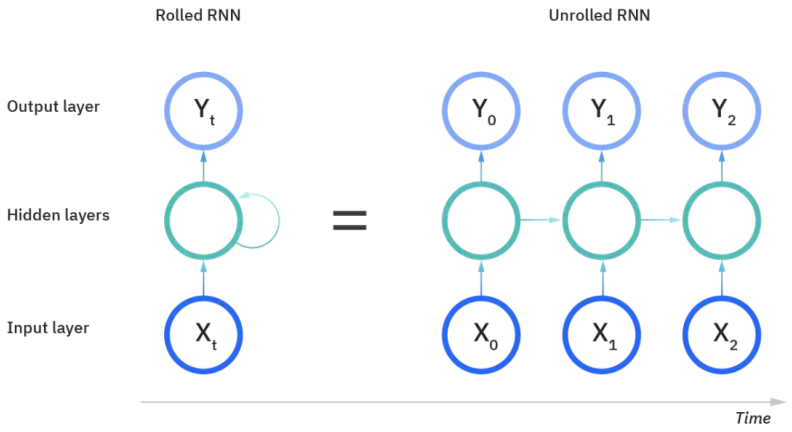

Figure 9: Unwrapped Neural Network (IBM Education, 2022)

To preserve the state of a neuron at previous points in time, neural networks have a "memory" called a memory cell. Although recurrent neural networks can capture dependencies between short-term data, this type of model can encounter problems when modeling long data sequences. This problem is known as vanishing gradient descent, which implies a worsening of memory over time.

2.4.2.4. LSTM

The LSTM (Long Short Term Memory) model, introduced by Sepp Hochreiter and Juergen Schmidhuber, is, roughly speaking, a type of Recurrent Neural Network. As they explain in their paper "Long-Short Term Memory", published in 1997, the advantage of this model is that it solves the short-term memory problem of traditional RNN models. The use of LSTM memory allows the neural network to identify important information, leaving aside whatever is irrelevant to the model. The structure of each neuron in an LSTM model is shown in Figure 10.

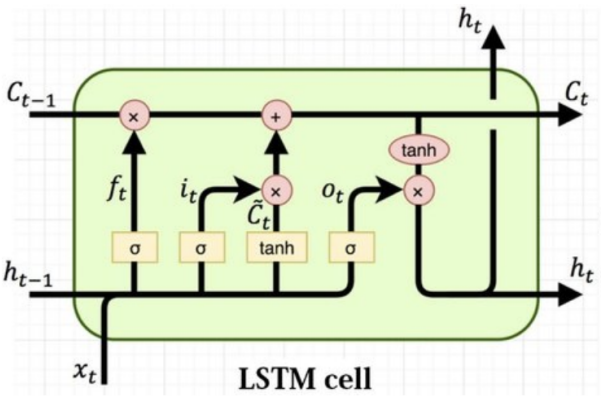

Figure 10: LSTM Cell (Stack Exchange, 2020)

An important characteristic about this type of neuron is its two states. The cell state (c_t) can be understood as the long-term memory of the model. This is the component that differentiates LSTMs from RNNs. On the other hand, the hidden state contains the information that comes from previous layers of the model. It is therefore available in both RNN and LSTM models.

2.5. SELECTED TOOLS

2.5.1. SAS

Throughout this project, SAS has been used for both formatting data and developing the ARIMA time series development.

2.5.2. PYTHON

To create the LSTM model, Python and its libraries pandas, numpy, matplotlib, statsmodel and torch.nn, among others, have been used.

3. DATA ENGINEERING

3.1. Data Selection

As previously mentioned, one of the main objectives of this dissertation is to develop two predictive models (ARMA/ARIMA and LSTM) of air quality in the city of Madrid. This study will help us understand how time series models predict future pollution values.

Therefore, this data engineering section is pivotal for subsequent analysis. Not only do we have to find relevant data, but also match the data structure required by both models. To prepare the data for use in the predictive models, we have used SAS; as previously stated.

The City Council of Madrid has an Open Data Catalog which collects information relevant to the dynamics of the Spanish capital city (traffic, cultural activities, medical emergency services...). The objective of this platform, as indicated by the municipality's government, is to promote access to municipal data boosting the development of creative tools to attract and serve the citizens of Madrid (Ayuntamiento de Madrid, 2016). This service has been available to citizens since March 2014.

Among the catalog of available data, we find a wide selection of information related to Madrid's air quality. Relevant data includes daily, hourly and real-time indicators of the different air pollutants. Each air quality station studies a fixed set of pollutants (as described in Table 2 of the Appendix).

To develop both predictive models, hourly air quality data has been selected as the indicator of interest. The selected data sources use the arithmetic mean of ten-minute intervals recorded every hour (Ayuntamiento de Madrid, 2016), to collect each observation. That is, such database collects the values of each pollutant on an hourly basis during the 24 hours of the day. Likewise, some stations have sensors used to record various climatological indicators at each time of the day (Table 2, Appendix).

In relation to the selected data, it is important to note that the Madrid City Council does not have a homogeneous data measurement system. That is, not all stations collect each of the climatological indicators nor all the pollutant indicators.

3.2. Data Structure

Before formatting the data, it is important to know the structure of the data. The format of the hourly air quality files is discussed below:

3.2.1. Structure of Hourly Data Files: Air Quality

The hourly air quality data from the Madrid City Council has been downloaded in CSV format. Each record in the file has the following structure:

PROVINCIA	MUNICIPIO	ESTACION	MAGNITUD	PUNTO_MUESTREO	ANO	MES	DIA	H01	V01	H02	V02
28	79	4	1	28079004_1_38	2019	1	1	23	V	17	V

Figure 11: Structure of each individual observation obtained by Air Quality Sensors.

Since we are working with data from the Municipality of Madrid, the value of the *PROVINCIA* (province) and *MUNICIPIO* (municipality) fields is the same in all records. Additionally, the variable *ESTACION* (station) identifies which of the 24 air quality stations (Table 2, Appendix) has collected this data. In addition, the *MAGNITUD* (magnitude) field distinguishes which pollutant the observation in question studies. For example, in the particular case of Figure 11, this observation presents the hourly values of sulfur dioxide (magnitude 1) collected at Plaza de España on January 1, 2019.

The observation *PUNTO_MUESTREO* (sampling point) simply unifies the values of the fields mentioned, adding the measurement technique used (presented by the last two digits).

Finally, each observation has twenty-four value fields and twenty-four validation fields. That is, variables H01-H24 correspond to the data obtained at each hour of the day (H01 being one o'clock in the morning). On the other hand, for an observation to be considered valid, it must contain a "V" in each validation field (V01-V24).

3.3. Data Formatting

Data formatting plays an essential role in the preparation of files for the development of both predictive models. As previously mentioned, there are a total of 24 air quality measurement stations in Madrid. Although these stations together measure a total of 17 pollutant gases, our final file will only contain the values relevant to those pollutants relevant to the National Air Quality Index; Particulate Matter (PM2.5 & PM10), Ozone (O_3), Nitrogen Dioxide (NO_2), Sulfur Dioxide (SO_2) and Carbon Monoxide (CO) (Table 6, Appendix).

Total rows: 31368 Total columns: 56 Rows 1-100

	PROVINCIA	MUNICIPIO	ESTACION	MAGNITUD	PUNTO_MUESTREO	ANO	MES	DIA	H01	V01	H02	V02
1	28	79	102	81	31JUL02:10:38:00	2021	1	1	0.98	V	1.4	V
2	28	79	102	81	31JUL02:10:38:00	2021	1	2	1.18	V	1.65	V
3	28	79	102	81	31JUL02:10:38:00	2021	1	3	0.62	V	0.55	V
4	28	79	102	81	31JUL02:10:38:00	2021	1	4	1.93	V	1.82	V
5	28	79	102	81	31JUL02:10:38:00	2021	1	5	1.62	V	0.78	V
6	28	79	102	81	31JUL02:10:38:00	2021	1	6	1.1	V	1.12	V
7	28	79	102	81	31JUL02:10:38:00	2021	1	7	2.28	V	2.5	V
8	28	79	102	81	31JUL02:10:38:00	2021	1	8	2.38	V	2.83	V
9	28	79	102	81	31JUL02:10:38:00	2021	1	9	0	V	0	V
10	28	79	102	81	31JUL02:10:38:00	2021	1	10	1.23	V	1.02	V

Figure 12: Pre-Formatting Data Structure

3.3.1. Data Fromatting: Air Quality

Time series use a time variable and a single dependent variable. Initially, our dataset has 24 study variables for each observation (H01-H24). Therefore, the final objective of the formatting will be to transform the data so that the first file contains the arithmetic mean of the 24 hourly values as the dependent variable (Figure 13), while the second file has a transposed format of the values to end up with a single H variable and a single time variable (which in addition to detailing the date includes a time stamp), as can be seen in Figure 14.

Total rows: 8660 Total columns: 4

	ESTACION	MAGNITUD	FECHA	MEDIA_DIARIA_CONT
1	4	8	01/01/2021	9.79
2	4	8	02/01/2021	23.04
3	4	8	03/01/2021	29.58
4	4	8	04/01/2021	29.25
5	4	8	05/01/2021	54.13
6	4	8	06/01/2021	34
7	4	8	07/01/2021	34.08
8	4	8	08/01/2021	23.63
9	4	8	11/01/2021	25.5
10	4	8	12/01/2021	60.33

Figure 13: Post-Formatting File DATOS_CONT_MEDIADIARIA (Daily Pollution Avg.)

We begin by formatting the air quality data file with the objective of creating two datasets that allow working with time series forecasts. One will use the daily mean of the pollutant as the dependent variable, while the other will use the hourly values. The steps followed to transform the initial data structure (shown in Figure 11) into the daily data structure shown in Figure 13 are as follows:

1. First, extraneous variables are discarded. As I have commented above, all air quality stations are located within the Municipality of Madrid. Therefore, all observations contain the same data in the variables *PROVINCIA* and *MUNICIPIO*, so both columns are eliminated. In addition, the *PUNTO_MUESTREO* column is eliminated, since its content is irrelevant in the creation of the predictive models.
2. Columns V01-V24 indicate the validity of each hourly pollution variable. That is, it indicates if data has been collected correctly and if its veracity has been checked. Therefore, once the validity check of each observation data has been performed, only those observations that have validated hourly data are kept in the dataset. Columns V01-V24 are eliminated.
3. The cells *DIA* (day) *MES* (month) and *AÑO* (year) have been combined into a single variable called *FECHA* (date). Moreover, its format has been modified, allowing SAS to interpret it as a date, which will facilitate the use of this variable in time series forecasting.
4. According to Order TEC/351/2019, of March 18, which approves the National Air Quality Index, the following pollutants are taken into account when creating the model.
    a. Sulfur dioxide (SO): Represented in observations with Magnitude 1.
    b. Nitrogen dioxide (NO2): Represented in observations with Magnitude 8.
    c. Carbon monoxide (CO2): Represented in observations with Magnitude 6.
    d. Ozone (O3): Represented in observations with Magnitude 14.
    e. Total suspended particles (PM2.5 and PM10): Represented in the observations with Magnitudes 9 and 10.

To perform both predictive models we work only with the data relevant to these gases and particulate pollutants. Therefore, we eliminate the observations that have as *MAGNITUD* a value other than 1,8,6,14,9 and 10.

5. Additionally, only the observations related to nitrogen dioxide (Magnitude 8) are retained in the dataset. This is the only variable measured in all stations, so studying it will allow us to compare the results obtained.

Up to this point, the transformations have been common both for the hourly values file and for the file containing the daily mean of each pollutant as the dependent variable. From this point on, two new datasets are created diverging from the original dataset. These datasets are called *DATOS_CONT_MEDIADIARIA* (contamination data for daily mean) and *DATOS_CONT_DATOSHORARIOS* (hourly pollution data).

1. The first dataset, "DATOS_CONT_MEDIADIARIA" (Figure 13), as its name indicates, contains the daily average as a reference point for the value of each pollutant on each date. To complete this file, the variable "DATOS_CONT_MEDIADIARIA" has been created. This variable contains the arithmetic mean of the hourly values of each pollutant (H01-H24), each

day. Once this variable has been created, columns H01-H24 have been eliminated to avoid redundancy.

	ESTACION	MAGNITUD	VALOR_HORARIO_CONT	FECHA_HORA
1	4	8	10	01JAN2021:01:00:00
2	4	8	16	01JAN2021:02:00:00
3	4	8	7	01JAN2021:03:00:00
4	4	8	5	01JAN2021:04:00:00
5	4	8	6	01JAN2021:05:00:00
6	4	8	8	01JAN2021:06:00:00
7	4	8	14	01JAN2021:07:00:00
8	4	8	21	01JAN2021:08:00:00
9	4	8	13	01JAN2021:09:00:00
10	4	8	13	01JAN2021:10:00:00

Figure 14: Post-Formatting File DATOS_CONT_DATOSHORARIOS (Hourly Pollution Value)

2. On the other hand, to create the file *DATOS_CONT_DATOSHORARIOS* (Image 8), the original file has been transposed following the steps detailed below:
    a. First, two new variables have been created. *VALOR_HORARIO_CONT* (Hourly contamination value) and *HORA* (time)
    b. *VALOR_HORARIO_CONT*:
        i. Variable is initialized to **0** for every observation.
        ii. Creation of a temporary variable "n" which, thanks to a while loop n<25, can contain values between 1 and 24 (representing each hour of the day). The objective of this loop is to create 24 copies of each observation, storing the contamination value corresponding to each hour of the day in *VALOR_HORARIO_CONT* sequentially. For example, if n=1 the value corresponding to 1:00:00 AM will be stored in *VALOR_HORARIO_CONT*. So on and so forth until 12:00 midnight.
        iii. After the previous steps have been completed, each observation will contain 8 equivalent rows, the only difference being *VALOR_HORARIO_CONT*.

    c. HORA:
        i. Variable is initialized to **0** for every observation.
        ii. As the previously mentioned loop is executed, the value represented in the time variable "n" is transferred to this variable, changing its format, so that SAS identifies it as a time variable.

iii. Combino *HORA* y *FECHA*, creando una única variable temporal que facilite la implementación de series temporales.I combine *HORA* and *FECHA*, creating a single time variable that facilitates the implementation of time series.

d. Once these two variables have been successfully created, variables H01 through H24 are deleted, as well as variable n.

Once the two files have been created, the data formatting stage is completed. The data is now ready for analysis using time series forecasting.

4. DATA ANALYSIS: Predictive models

4.1. Introduction to Elaborated Models

To develop these predictive models of air quality, nitrogen dioxide remains our pollutant of interest. As previously explained, it is the gas that represents high risk levels. Furthermore, this study will narrow its prediction to pollution levels at *Plaza de Castilla.* This station was selected as reference, due to its urban location and dense traffic in its vicinity. Both predictive models will assess levels of this pollutant between January 2017 and March 2022. In addition, to compare the generated results, both models will have a data distribution in training and test datasets of 80% and 20% respectively.

4.2. ARIMA

As mentioned above, the first model is based on the concept of time series. This predictive model has been performed both on the file containing the daily averages (*DATOS_CONT_MEDIADIARIA*) of the NO2 level at each station, and on the dataset that includes the hourly data (*DATOS_CONT_DATOSHORARIOS*). The objective is to see with which data the model fits best, thus using that dataset for the LSTM model. For the purpose of synthesis when explaining the development of the model, the results obtained for the daily dataset will be shown. The results of the ARIMA model for the hourly data set can be found in section 7 of the Appendix.

4.2.1. Modelos ARIMA en SAS.

4.2.1.1. Descriptive Statistics

Before digging deeper into the model, it is interesting to plot descriptive graphs, which allow us to observe the real behavior of nitrogen dioxide levels at the *Plaza de Castilla* station throughout the selected time frame.

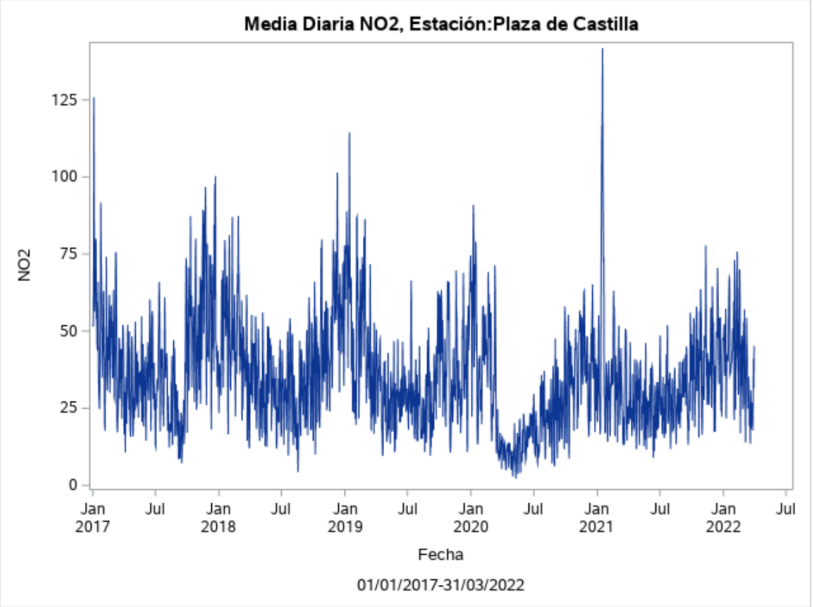

Figure 15: Daily NO2 Average Pza. Castilla (January 2017- March 2022)

Figure 15 shows a high oscillation in nitrogen dioxide levels over time. This may be a good indicator of seasonality. For example, a potential trigger for the oscillation of NO2 levels may be the variation in traffic levels over a working day. As explained above, nitrogen dioxide is generated by incomplete combustion of certain fuels, including gasoline and kerosene. Therefore, if the level of traffic increases (as is often the case during rush hour) on the roads surrounding a station, it is understandable that the station will detect a higher presence of NO2.

Regarding the long-term trend, the graph does not reflect significant trends. That is, it appears that throughout the year, despite short-term oscillations, the data show neither positive nor negative long-term patterns, with NO2 levels remaining relatively constant over time.

The procedure used in SAS to develop an ARIMA time series is called PROC ARIMA. As described by Box and Jenkins in 1976, this procedure follows three phases: identification, estimation and diagnosis, and finally prediction. An ARIMA model is developed below using each of these phases.

4.2.1.2. Dataset Division

The dataset has been divided into train (80%) and test (20%). The model is trained using the first data set, while further demonstrating the validity of the model using the second.

4.2.1.3. Phase 1: Identification

In this first phase, the IDENTIFY statement is used, with which, in addition to studying which model is appropriate to predict the dependent variable, the stationarity of the time series can also be examined using the Augmented Dickey-Fuller (ADF) test. It is recalled that the ARIMA model

assumes stationarity in the data, so performing the Augmented Dickey-Fuller test is of particular relevance. If the data are non-stationary, differences will have to be performed.

The first result obtained through the identification statement is a summary of the most relevant descriptive statistics for each file. As shown in Figure 16, the mean nitrogen dioxide level (between 2017 and 2021) in Plaza de Castilla is $36.07 \mu g/m^3$. It also has a standard deviation of $18.75\ \mu g/m^3$. Broadly speaking, taking into account the air quality scales presented by the EEA (Table 1 of the Appendix), it is concluded that the average level of nitrogen dioxide does not seem to pose any risk to the health of the people of Madrid, and therefore, the air quality level in Plaza de Castilla could be considered appropriate.

Name of Variable = MEDIA_DIARIA_CONT	
Mean of Working Series	36.60731
Standard Deviation	18.75406
Number of Observations	1532

Figure 16: Descriptive Statistics on Daily NO2 Average, Pza. Castilla

Additionally, the identification phase produces a panel of graphs used to study the autocorrelation of the time series and perform a subsequent trend analysis (Figure 17). This panel of data is generated taking into account "n" previous periods (or n "lags"). SAS offers the user the possibility to define the number of periods to study through the use of the NLAG option. By default, NLAG is 24; and for this study, the value of NLAG has not been modified.

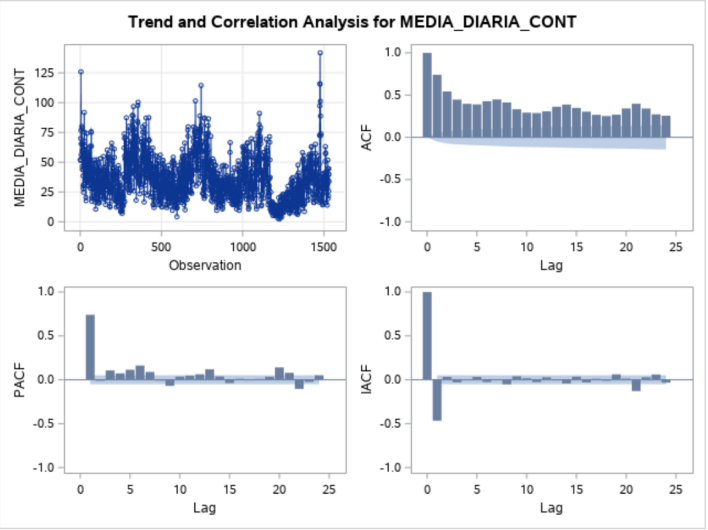

Figure 17: Trend and Correlation Analysis Daily NO2 Average, Pza. Castilla.

Among the graphs presented, we find the Autocorrelation Function (ACF). As previously studied, this graph allows us to understand whether we are dealing with stationary or non-stationary

series. A visual analysis of the Autocorrelation Function of the data set with daily NO2 values (Figure 17) allows us to assume the stationary of this data sequence; the AFC decreases rapidly (and not exponentially).

To check whether the stationarity assumptions are correct, the Dickey-Fuller Stationarity Test is performed; where the null hypothesis indicates non-stationarity. Therefore, if the result presents a significant p-value (at a significance of 95%), it can be concluded that the data is stationary, thus rejecting the null hypothesis. Observing the results presented in Figure 18, and taking into account the p-value of the Dickey-Fuller single mean test, it is confirmed that the null hypothesis can be rejected, thus demonstrating its stationarity.

On the other hand, the Partial Autocorrelation Function (PACF), as previously studied, helps choose the most appropriate predictive model. In the case of daily NO2 values, using AR(1) will be sufficient, because the partial autocorrelation function (PACF) decreases rapidly after the first period, as shown in the PACF plot. However, the PACF plot resulting from the hourly data set indicates that an AR(2) model (Appendix, 7.7.3) is more appropriate for the second model; due to the significance of the first two periods.

Augmented Dickey-Fuller Unit Root Tests							
Type	Lags	Rho	Pr < Rho	Tau	Pr < Tau	F	Pr > F
Zero Mean	0	-82.9666	<.0001	-6.55	<.0001		
	1	-68.6165	<.0001	-5.87	<.0001		
	2	-46.5429	<.0001	-4.83	<.0001		
Single Mean	0	-397.001	0.0001	-15.10	<.0001	114.00	0.0010
	1	-406.028	0.0001	-14.24	<.0001	101.43	0.0010
	2	-320.114	0.0001	-12.06	<.0001	72.73	0.0010
Trend	0	-414.674	0.0001	-15.47	<.0001	119.70	0.0010
	1	-429.850	0.0001	-14.64	<.0001	107.21	0.0010
	2	-342.399	0.0001	-12.41	<.0001	76.99	0.0010

Figure 18: Augmented Dickey-Fuller Test, Daily NO2 Average Pza. Castilla.

Finally, the IDENTIFY statement performs an autocorrelation check for white noise. The null hypothesis of the "Autocorrelation Check for White Noise" states that the autocorrelations of the time series are not significantly different from zero. That is, it is assumed that there is no correlation between the values in the data set. If this hypothesis is rejected, it is confirmed that a time series model (such as ARMA and ARIMA models) would fit the data set, due to the correlation between the data. As can be seen in Figure 19, the p-values are significant for all the lags (with a significance of 99%). Therefore, the null hypothesis can be rejected, confirming that the use of time series is appropriate.

Autocorrelation Check for White Noise									
To Lag	Chi-Square	DF	Pr > ChiSq	Autocorrelations					
6	2359.11	6	<.0001	0.741	0.543	0.447	0.397	0.390	0.427
12	3506.13	12	<.0001	0.448	0.413	0.332	0.291	0.287	0.307
18	4482.15	18	<.0001	0.362	0.388	0.348	0.304	0.267	0.252
24	5420.31	24	<.0001	0.268	0.342	0.398	0.341	0.272	0.256

Figure 19: Autocorrelation Check for White Noise Daily NO2 Average, Pza. Castilla.

4.2.1.4. Phase 2: Estimation and Diagnosis

The second phase of the ARIMA procedure emphasizes the estimation and diagnosis of the data. By means of the "ESTIMATE" statement, within the PROC ARIMA procedure, different models are compared, thus selecting the most appropriate one for the data. In addition, this second phase performs a residual diagnosis. It is necessary that these residuals are normally distributed, N(μ,σ2), with covariance equal to 0. That is, it is important that the residuals are white noise and uncorrelated.

The estimation phase is completed for the data set whose NO2 value represents the daily mean. In the diagnostic phase, it is concluded that the data set will optimally fit an AR(1) model. However, to confirm that this is indeed the case, we proceed to compare this model with the ARIMA(1,0,1) and ARIMA(2,0,0,0) models.

The first outputs generated by the ESTIMATE statement (Figure 20, 21 and 22) are the estimated parameters in each of the models. The t-value shown in each of the graphs indicates the importance of each parameter for this model, and its p-value indicates its significance. Observing the ARIMA(1,0,1) model, it highlights the low significance of a moving average model of degree 1 in our model. Likewise, the ARIMA(2,0,0) model contains a negligible t-value when adding the second degree to the auto-regressive model. In turn, the AR(2) model fails to reject the null hypothesis.

Conditional Least Squares Estimation					
Parameter	Estimate	Standard Error	t Value	Approx Pr > |t|	Lag
MU	36.71410	1.23667	29.69	<.0001	0
AR1,1	0.74070	0.01718	43.12	<.0001	1

Figure 20: Estimated Parameters ARIMA(1,0,0)

Conditional Least Squares Estimation					
Parameter	Estimate	Standard Error	t Value	Approx Pr > |t|	Lag
MU	36.70975	1.21822	30.13	<.0001	0
MA1,1	-0.02144	0.03454	-0.62	0.5349	1
AR1,1	0.73101	0.02357	31.01	<.0001	1

Figure 21: Estimated Parameters ARIMA(1,0,1)

Conditional Least Squares Estimation					
Parameter	Estimate	Standard Error	t Value	Approx Pr > |t|	Lag
MU	36.71080	1.22305	30.02	<.0001	0
AR1,1	0.74925	0.02557	29.30	<.0001	1
AR1,2	-0.01154	0.02557	-0.45	0.6520	2

Figure 22: Estimated Parameters ARIMA(2,0,0)

Next (and as can be seen in Figures 23, 24 and 25), when comparing models, their AIC (Akaike's Information Criterion) and SBC (Schwarz's Bayesian Criterion) statistics are usually compared; the model with the lowest values for these statistics is the most appropriate. Once again, the ARIMA(1,0,0) model turns out to be the most optimal; although not by much.

Constant Estimate	9.519848
Variance Estimate	158.9649
Std Error Estimate	12.60813
AIC	12114.85
SBC	12125.52
Number of Residuals	1532

Figure 23: Goodness of Fit Statistics ARIMA(1,0,0)

Constant Estimate	9.874633
Variance Estimate	159.0394
Std Error Estimate	12.61108
AIC	12116.57
SBC	12132.57
Number of Residuals	1532

Figure 24: Goodness of Fit Statistics ARIMA(1,0,1)

Constant Estimate	9.628821
Variance Estimate	159.0477
Std Error Estimate	12.61141
AIC	12116.65
SBC	12132.65
Number of Residuals	1532

Figure 25: Goodness of Fit Statistics ARIMA(2,0,0)

Furthermore, the table of correlations between estimated parameters (Correlation of Parameter Estimates) is a good resource to check if there is collinearity between the data. Figures 26, 27 and 28 show that the only model without correlation between the estimated parameters is the ARIMA(1,0,0).

Correlations of Parameter Estimates		
Parameter	MU	AR1,1
MU	1.000	0.006
AR1,1	0.006	1.000

Figure 26: Correlation of Parameter Estimates ARIMA(1,0,0)

Correlations of Parameter Estimates			
Parameter	MU	MA1,1	AR1,1
MU	1.000	0.002	0.005
MA1,1	0.002	1.000	0.672
AR1,1	0.005	0.672	1.000

Figure 27: Correlation of Parameter Estimates ARIMA(1,0,1)

Correlations of Parameter Estimates			
Parameter	MU	AR1,1	AR1,2
MU	1.000	0.002	0.002
AR1,1	0.002	1.000	-0.741
AR1,2	0.002	-0.741	1.000

Figure 28: Correlation of Parameter Estimates ARIMA(2,0,0)

In addition to studying the collinearity of the estimated parameters, it is also important to study the autocorrelation of their residuals. The null hypothesis of the test for autocorrelation of residuals indicates the non-correlation of the residuals. However, the significance of test 2 (Figures 29, 30 and 31) in all potential models indicates the existence of some collinearity between the residuals;

due to the rejection of the null hypothesis. This may indicate that these models are not the most appropriate.

Autocorrelation Check of Residuals									
To Lag	Chi-Square	DF	Pr > ChiSq	Autocorrelations					
6	23.92	5	0.0002	0.009	-0.084	-0.011	-0.010	-0.014	0.090
12	92.48	11	<.0001	0.160	0.135	-0.015	-0.017	0.004	-0.012
18	158.15	17	<.0001	0.103	0.164	0.060	0.032	0.004	-0.013
24	276.92	23	<.0001	-0.056	0.082	0.243	0.071	-0.045	0.016
30	348.46	29	<.0001	-0.005	-0.023	0.092	0.178	0.071	-0.001
36	430.40	35	<.0001	-0.004	-0.001	0.000	0.065	0.208	0.069
42	483.89	41	<.0001	-0.024	0.030	-0.038	0.002	0.029	0.174
48	503.41	47	<.0001	0.080	-0.016	0.021	-0.022	-0.005	0.069

Figure 29: Autocorrelation Check for Residuals ARIMA(1,0,0)

Autocorrelation Check of Residuals									
To Lag	Chi-Square	DF	Pr > ChiSq	Autocorrelations					
6	21.66	4	0.0002	-0.002	-0.075	-0.003	-0.005	-0.011	0.091
12	89.69	10	<.0001	0.159	0.136	-0.013	-0.013	0.008	-0.010
18	155.19	16	<.0001	0.103	0.164	0.060	0.035	0.007	-0.009
24	272.20	22	<.0001	-0.053	0.081	0.242	0.070	-0.043	0.020
30	342.97	28	<.0001	-0.001	-0.021	0.092	0.177	0.071	0.001
36	424.19	34	<.0001	-0.000	0.002	0.003	0.065	0.207	0.069
42	477.45	40	<.0001	-0.023	0.034	-0.035	0.006	0.028	0.173
48	496.69	46	<.0001	0.079	-0.015	0.024	-0.020	-0.003	0.069

Figure 30: Autocorrelation Check for Residuals ARIMA(1,0,1)

Autocorrelation Check of Residuals									
To Lag	Chi-Square	DF	Pr > ChiSq	Autocorrelations					
6	22.23	4	0.0002	0.001	-0.078	-0.005	-0.006	-0.012	0.091
12	90.41	10	<.0001	0.159	0.136	-0.014	-0.014	0.007	-0.011
18	155.95	16	<.0001	0.103	0.164	0.060	0.034	0.006	-0.010
24	273.46	22	<.0001	-0.054	0.081	0.243	0.070	-0.044	0.019
30	344.44	28	<.0001	-0.002	-0.022	0.092	0.177	0.071	0.000
36	425.86	34	<.0001	-0.001	0.002	0.002	0.065	0.207	0.069
42	479.17	40	<.0001	-0.023	0.033	-0.036	0.005	0.028	0.173
48	498.49	46	<.0001	0.079	-0.015	0.023	-0.020	-0.003	0.069

Figure 31: Autocorrelation Check for Residuals ARIMA(2,0,0)

Although there is certain correlation between residuals, Figures 32, 33 and 34 show that the best result is obtained when using 6 lags (or periods) for data prediction. This argument is confirmed when taking into account the Q-Q Plot and the normal distribution in Figures 35, 36 and 37 since the residuals fit this distribution quite well (essential characteristic of white noise).

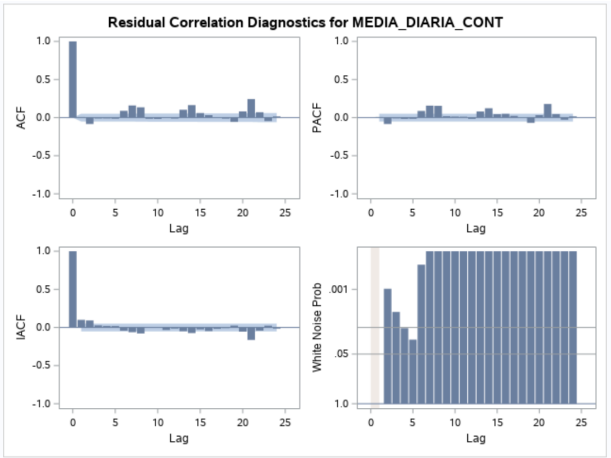

Figure 32: Residual Correlation Diagnostic ARIMA(1,0,0)

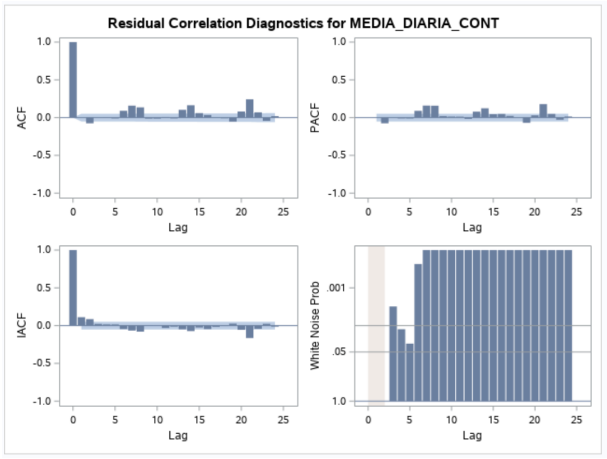

Figure 33: Residual Correlation Diagnostic ARIMA(1,0,1)

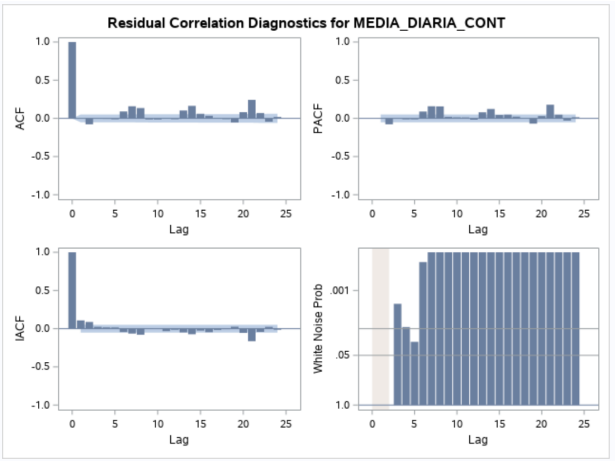

Figure 34: Residual Correlation Diagnostic ARIMA(2,0,0)

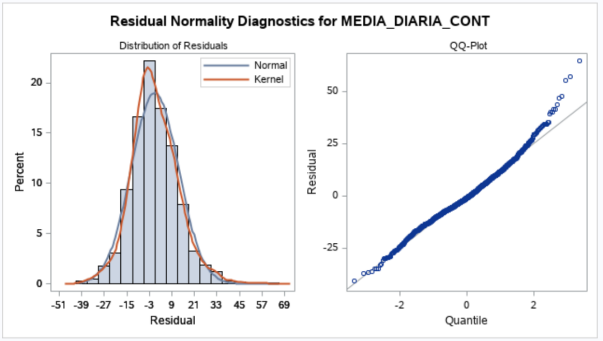

Figure 35: Normality Check of Residuals ARIMA(1,0,0)

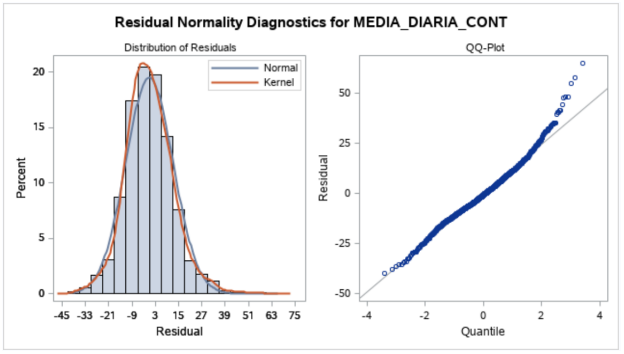

Figure 36: Normality Check of Residuals ARIMA(1,0,1)

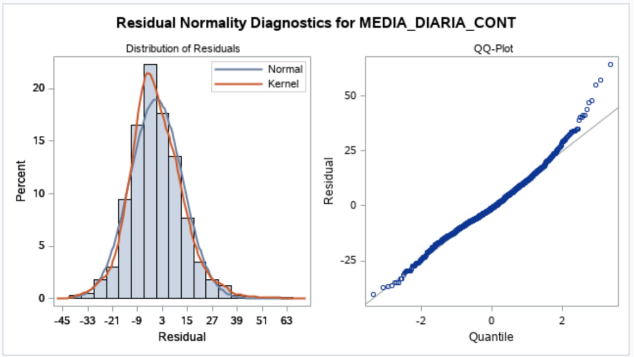

Figure 37: Normality Check of Residuals ARIMA(2,0,0)

It is concluded that the AR(1) model is the most appropriate for this dataset, as had been intuited in the identification phase. Therefore, it is this model that will be used in the last phase of PROC ARIMA: Forecast. Once the estimation phase for the dataset with daily values has been completed, the same procedure is repeated for the dataset with hourly values (Appendix 7); it turns out that the model that best fits the second dataset is an AR(2)

4.2.1.5. Phase 3: Prediction

This last stage of the ARIMA models in SAS uses the FORECAST statement, which allows estimating future values of the time series within a 95% confidence range.

First, this statement is executed for the Train data set, generating also a predictive AR(1) model for the Test data set. The FORECAST statement generates a data file with the predictions estimated by the model on each data set. In addition, the prediction is compared with the actual value, thus generating the variable STD, which represents the standard error of each prediction.

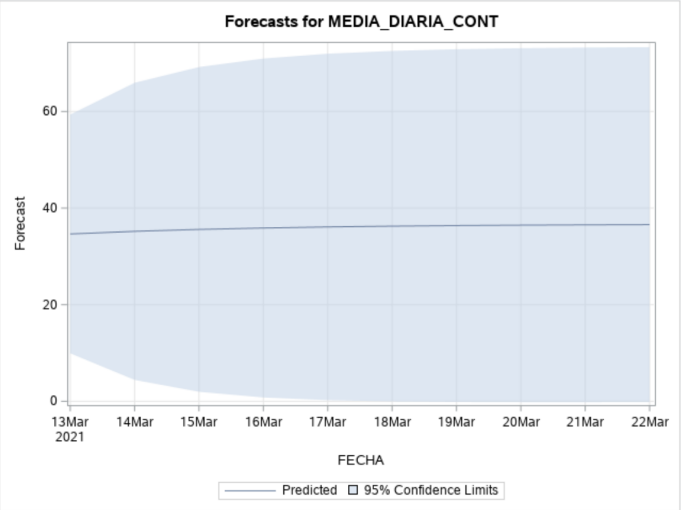

Figure 38: FORECAST results (train)(1)

Forecasts for variable MEDIA_DIARIA_CONT				
Obs	Forecast	Std Error	95% Confidence Limits	
1533	34.6445	12.6081	9.9330	59.3560
1534	35.1811	15.6901	4.4291	65.9332
1535	35.5786	17.1473	1.9706	69.1867
1536	35.8731	17.8964	0.7967	70.9494
1537	36.0911	18.2944	0.2348	71.9475
1538	36.2527	18.5091	-0.0245	72.5298
1539	36.3723	18.6258	-0.1337	72.8783
1540	36.4609	18.6896	-0.1700	73.0919
1541	36.5266	18.7245	-0.1727	73.2259
1542	36.5752	18.7436	-0.1615	73.3119

Figure 39: FORECAST results (train)(2)

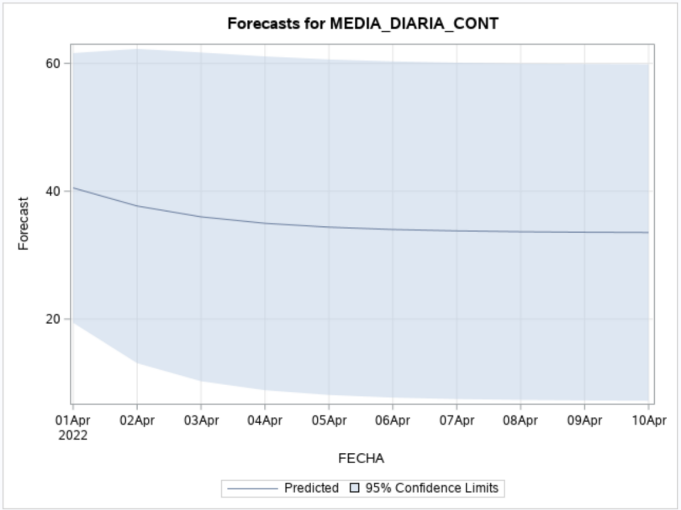

Figure 40: FORECAST results (test)(1)

Forecasts for variable MEDIA_DIARIA_CONT				
Obs	Forecast	Std Error	95% Confidence Limits	
385	40.5176	10.7602	19.4281	61.6071
386	37.6869	12.5382	13.1125	62.2612
387	35.9937	13.1159	10.2869	61.7004
388	34.9809	13.3165	8.8809	61.0808
389	34.3751	13.3876	8.1359	60.6142
390	34.0127	13.4129	7.7239	60.3015
391	33.7960	13.4220	7.4894	60.1025
392	33.6663	13.4252	7.3534	59.9792
393	33.5888	13.4264	7.2736	59.9039
394	33.5424	13.4268	7.2264	59.8583

Figure 41: FORECAST results (test)(2)

Observing the results shown in Figures 39 and 41, it can be seen that the standard error of the predictions of the "test" data set is lower than that generated by training the model. The ARIMA

model, as previously mentioned, has also been repeated for the data set *DATOS_CONT_DATOSHORARIOS*, using an AR(2) model. However, as can be seen in Appendix 7, the results obtained are worse, so the LSTM model has been carried out with the daily mean.

4.3. LSTM

This time we are going to work with a recurrent neural network, LSTM. This type of model presents an alternative use of time series. To create this type of recurrent neural network, the RNN model developed by Roberto Vazquez Lucerga is followed.

Throughout the program, several libraries, classes, methods and functions have been used. A list of the libraries used, their respective functions and, finally, a brief explanation of them can be found in Table 7 of the Appendix.

Making use of the "read_excel" function (pandas), the file of interest for the creation of the dataset is read. Using the parameter "index_col", the column 'DATE' is defined as the index of the data frame. Figures 42 and 43 show the first and last 5 observations of the dataset created:

FECHA	MEDIA_DIARIA_CONT
2017-01-01	52.29
2017-01-02	51.46
2017-01-03	69.96
2017-01-04	125.88
2017-01-05	76.67

Figure 42: Verify Import Data, head()

FECHA	MEDIA_DIARIA_CONT
2022-03-27	17.79
2022-03-28	28.54
2022-03-29	41.08
2022-03-30	31.50
2022-03-31	45.25

Figure 43: Verify Import Data, tail()

Once the data frame is created, the "matplotlib" library is used to generate a descriptive graph. This allows us to see (in broad strokes) the fluctuation of NO2 levels over the last five years (Figure 44).

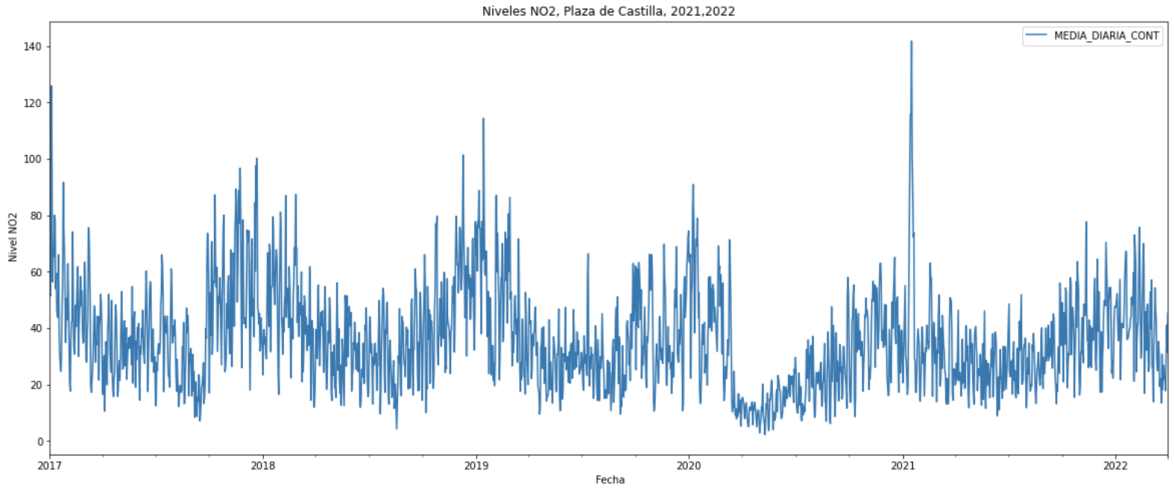

Figure 44: NO2 Levels, Pza. Castilla, 2017-2022

As mentioned in the Data Engineering section, the original data set has been formatted in SAS. However, before creating an LSTM recurrent neural network, it is necessary to perform several additional alterations to the dataset:

1. **One-Hot Encoding:**

Our data set is formed by sequential values. Unlike humans, machine learning models only recognize numeric vectors; a date itself does not add value to the model. To give sequentiality to the variable *FECHA* (date), four new variables are added to the data frame: *dia* (day), *mes* (month), *dia_de_la_semana* (weekday) and *semana_del_anio* (week of the year). To decompose the variable "date" in these columns, the Pandas API **.index** is used. For example, when creating the variable "day" we use .**index.day** (dia = df_datosdiarios_NO2.index.day). This attribute creates a new column from the index ("date") with values between 1 and 31; if the date corresponds to the first day of the month, the observation will have a 1 assigned in "day", and so on until the last day of the month whose value depends on the month to which it refers. This process is repeated for each variable created. The Pandas API, assign, generates a new dataset with the new columns (Figure 45).

Once the new data frame is generated, the method known as one-hot encoding is performed. This method creates n-dimensional vectors for each trait previously generated; n being the number of possible values that an observation can take in each variable. For example, the trait "month" can take values between 1 and 12. Using get_dummies (API, Pandas) each categorical variable is converted into a set of binary variables. In the case of "month", 12 binary variables are created (Figure 46) whose values are only activated if the date corresponds to that month.

FECHA	VALOR_NO2	dia	mes	dia_de_la_semana	semana_del_anio
2017-01-01	52.29	1	1	6	52
2017-01-02	51.46	2	1	0	1
2017-01-03	69.96	3	1	1	1
2017-01-04	125.88	4	1	2	1
2017-01-05	76.67	5	1	3	1
...	...	...	...	...	...
2022-03-27	17.79	27	3	6	12
2022-03-28	28.54	28	3	0	13
2022-03-29	41.08	29	3	1	13
2022-03-30	31.50	30	3	2	13
2022-03-31	45.25	31	3	3	13

Figure 45: Feature Engineering

FECHA	mes_1	mes_2	mes_3	mes_4	mes_5	mes_6	mes_7	mes_8	mes_9	\
2021-03-13	0	0	1	0	0	0	0	0	0	
2021-03-14	0	0	1	0	0	0	0	0	0	
2021-03-15	0	0	1	0	0	0	0	0	0	
2021-03-16	0	0	1	0	0	0	0	0	0	
2021-03-17	0	0	1	0	0	0	0	0	0	
...	...	...	...	...	...	...	...	...	...	
2022-03-27	0	0	1	0	0	0	0	0	0	
2022-03-28	0	0	1	0	0	0	0	0	0	
2022-03-29	0	0	1	0	0	0	0	0	0	
2022-03-30	0	0	1	0	0	0	0	0	0	
2022-03-31	0	0	1	0	0	0	0	0	0	

Figure 46: One-Hot Encoding

2. **Train-Validate-Test Split**:

Before implementing the predictive model, it is important to divide the dataset into three different datasets: **train, test and validate**. The objective of this division, as we know, is to **train** the model on the **train** data set, and use the data frame **test** to see if the generated model adequately predicts extraneous data, without overfitting taking place.

Firstly, the data is divided into train (80%) and test (20%). Then the data frame train is split again into **train** and **validate** data sets (80% and 20% respectively).

When splitting the data, two datasets have been created for each training phase; i.e. y_train and X_train. The data frames y_train, y_val and y_test contain the target variable (the NO2 value) while X_train, X_val and X_test contain the time-variable features. This division has been carried out since transformations are then applied to the data. It is therefore prudent to separate the binary variables from the rest, to avoid the transformations being applied incorrectly.

3. **MinMaxScaler**:

When designing Machine Learning models it is good practice to limit the possible values of incoming variables. Therefore, the MinMaxScaler function belonging to the "sklearn" class has been applied on all data sets. The MinMaxScaler function is as follows:

$$X\_std = (X - X.min(axis = 0)) / (X.max(axis = 0) - X.min(axis = 0))$$

$$X\_scaled = X\_std * (max - min) + min$$

```
            VALOR_NO2
FECHA
2017-01-01      52.29
2017-01-02      51.46
2017-01-03      69.96
2017-01-04     125.88
2017-01-05      76.67
...               ...
2020-05-05      10.50
2020-05-06      20.17
2020-05-07       8.83
2020-05-08       7.88
2020-05-09       5.71

[1225 rows x 1 columns]
------------NEW---------
[[0.40170732]
 [0.39495935]
 [0.54536585]
 ...
 [0.04837398]
 [0.04065041]
 [0.02300813]]
```

Figure 47: MinMaxScaler

This function restricts the values of the variables between zero and one. Additionally, it respects the original distribution of the data. Looking at Figure 47, the transformation of the dataset y_train is reflected. Initially, the dataset, in addition to containing the target variable also contains the date. Once the data is scaled, the MinMaxScaler() function converts the data frame into a single column vector in which the data is bounded between 0 and 1.

4. **Tensor creation**:

As the last step before designing the model, the data frames train, validate and test are converted into tensors. A tensor is the fundamental structure of any neural network. It consists of an array of n-dimensions formed by values of the same type. Moreover, a tensor is an iterable structure. For example, considering the training dataset X_train, we know that it contains 1,225 observations. Moreover, it has 103 columns representing the traits of the time variable. Therefore, the resulting tensor is a multidimensional array of 1,225 x 103.

```
tensor([[1., 0., 0.,  ..., 0., 1., 0.],
        [1., 0., 0.,  ..., 0., 0., 0.],
        [1., 0., 0.,  ..., 0., 0., 0.],
        ...,
        [0., 0., 0.,  ..., 0., 0., 0.],
        [0., 0., 0.,  ..., 0., 0., 0.],
        [0., 0., 0.,  ..., 0., 0., 0.]])
```

Figure 48: X_train Tensor Conversion

5. **Batch Creation:**

In addition, to improve the efficiency of the model, the concept of batches has been applied to this LSTM network. That is, when training the model, instead of feed-forwarding a single value to calculate its mean square error (and continuing with the back-propagation process), the data are grouped by feed-forwarding them in unison. This significantly reduces the computation time. In this case, we have followed the recommendation of the official Pytorch documentation, which indicates that 64 batches is usually the optimal value.

Once the data has been adjusted, the LSTMModel(Module.nn) class (belonging to the nn module of the torch library) is used. This class serves as a template for the creation of any neural network of type LSTM, and therefore represents the backbone of our model. The LSTMModel module contains only two methods:

1) **__init__(self, input_dim, hidden_dim, layer_dim, output_dim, dropout_prob)**

Creates an instance of the model, describing its characteristics. The arguments of this LSTM model are:

a) **input_dim (int):** Number of nodes in the input layer.
b) **hidden_dim (int):** Number of nodes in each hidden layer.
c) **layer_dim (int):** Number of layers in the model.
d) **output_dim (int):** Number of nodes in the output layer.
e) **dropout_prob (float):** The probability of nodes being dropped out.

When designing the model, it is important to carefully select each of these values. For example, in this model, the width of the input layer (input_dim) has been defined so that it has as many nodes as columns in the training dataset. As we have heard so many times throughout our careers, "data analysis is more of an art than a science". That is why different structures have been tested to design the model - especially in relation to the number of layers (layer_dim). The model obtains its best results when using 3 or 5 layers. Since the results are not different from each other, i.e., since the 5-layer model does not have a greater predictive capacity, the model with only 3 layers has been selected (with fewer number of layers, comes greater efficiency). The model designed has a structure similar to that shown in Figure 49:

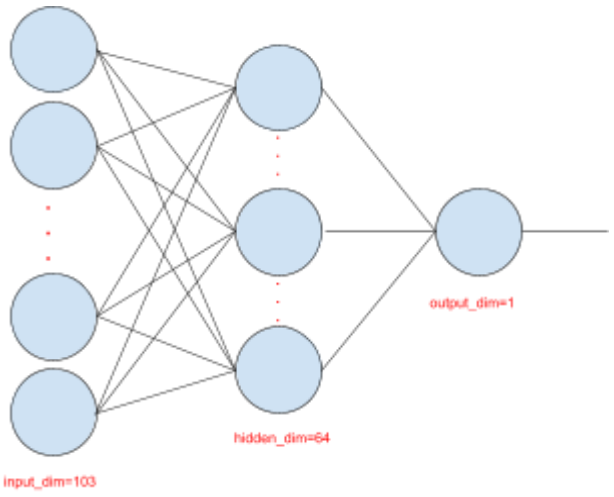

Figure 49: Structure of our LSTM Neural Network

2) forward(self,x):

As every neural network, an LSTM model has forward and backward propagation processes. Pytorch automatically implements the back propagation process, so only the forward propagation process is implemented. This method takes as argument the tensor (torch.Tensor) 'x', whose structure is [batch size, sequence length, input_dim]. It is recalled that batch size is the number of concurrent data to be fed to the model (In Figure 50 the batch size would be two. Depicted in dark blue), while the sequence length is simply the length of the data to be fed to the model (Sequence length in Figure 50 would be 4). The forward method outputs an output tensor which defines the state of the network once the forward propagation is finished.

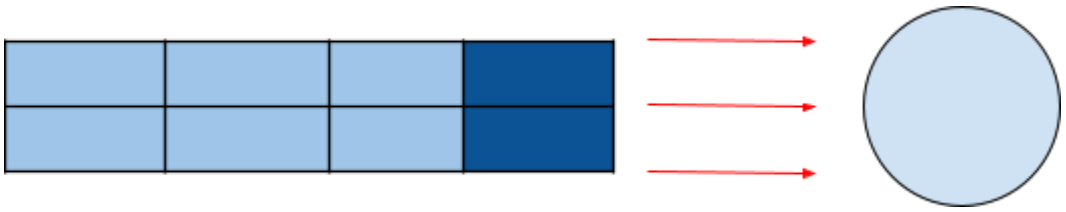

Figure 50: Batch and Sequence Length

The second core class in the development of this LSTM model is the Optimizer class. Within this class, we find the methods necessary to train, validate and demonstrate the model. As previously mentioned, a neural network is trained by oscillating between forward and backward propagation. The optimization class contains five methods that assist in the execution of the model:

1) __init__ (self,model, loss_fn, optimizer)

Once again, the init method initializes the instance of the optimizer.

a) **model**: LSTM previously designed.
b) **loss_fn:** MSE function. It describes the difference between the desired value and the real value obtained.
c) **optimizer:** Type of optimizer. In this case, an Adam Optimizer has been selected over the SGD, given the results obtained when each model was tested.

2) train_step (self, x, y):

Train_step, as the name implies, completes a single training phase; from forward propagation to gradient calculation and backtracking to the first layer. The x and y tensors are of interest as they represent both the data used to train the model and the target values that allow the error function to be calculated.

3) train(self,train_loader,val_loader,batch_size=64,n_epochs=50, n_features=1):

When training an LSTM model, the creator of the model defines a variable called 'n_epoch'. This variable indicates the number of times a model can traverse the entire training dataset. In this model, that translates to the number of times we will call the train_step method. Therefore, 50 passes through the train dataset will be performed.

4) evaluate(self, test_loader, batch_size=1, n_features=1)

A grandes rasgos, este método evalúa el modelo generado usando el dataset test.

5) plot_losses (self)

As its name indicates, this last method is used to create a comparative graph between the errors generated when training the model and when trying to demonstrate its effectiveness. As can be seen in Figure 51, after approximately 10 passes through the training dataset, the training error is approximately 0.01. This reflects the fast learning capability of our model. Once the training phase is done, the validation phase is completed, in which the predictive loss remains relatively constant with an error between 0.02 and 0.03. This shows that there is no over-fitting in the data and that the validations are good.

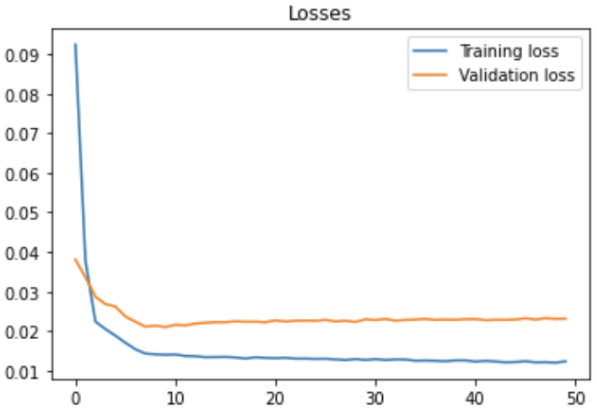

Figure 51: Train and Validation Loss

Once the model has been created and the optimization class has been run to train it, this program generates the table of predictions and values used in the model comparison section to calculate the RMSE prediction error.

FECHA	value	prediction
2021-03-13	31.420000	33.352650
2021-03-14	24.459999	25.881243
2021-03-15	24.169998	39.581089
2021-03-16	14.630000	41.512943
2021-03-17	13.080000	42.013653
...	...	...
2022-03-27	17.789999	22.908422
2022-03-28	28.539999	27.392702
2022-03-29	41.080002	29.810482
2022-03-30	31.500000	29.969570
2022-03-31	45.250000	29.628551

Figure 52: Output LSTM

4.4. Result Comparison

Firstly, it is interesting to compare the results obtained visually. Figures 53 and 54 reveal that the ARIMA model seems to be a better fit to the validation data set. Nevertheless, and broadly speaking, both models seem to predict the trajectory of NO2 quite well. As for the LSTM model, while the predictions generated seem to follow the oscillations of the long-term data, this model seems to be less effective in short-term predictions.

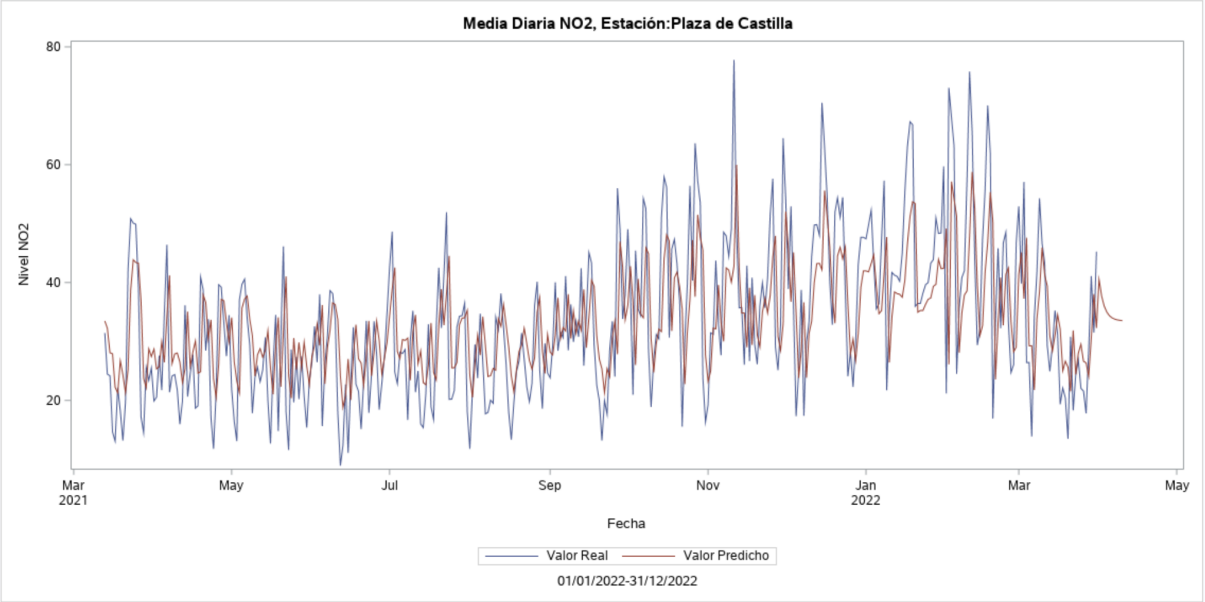

Figure 53: ARIMA Test Forecast Plot

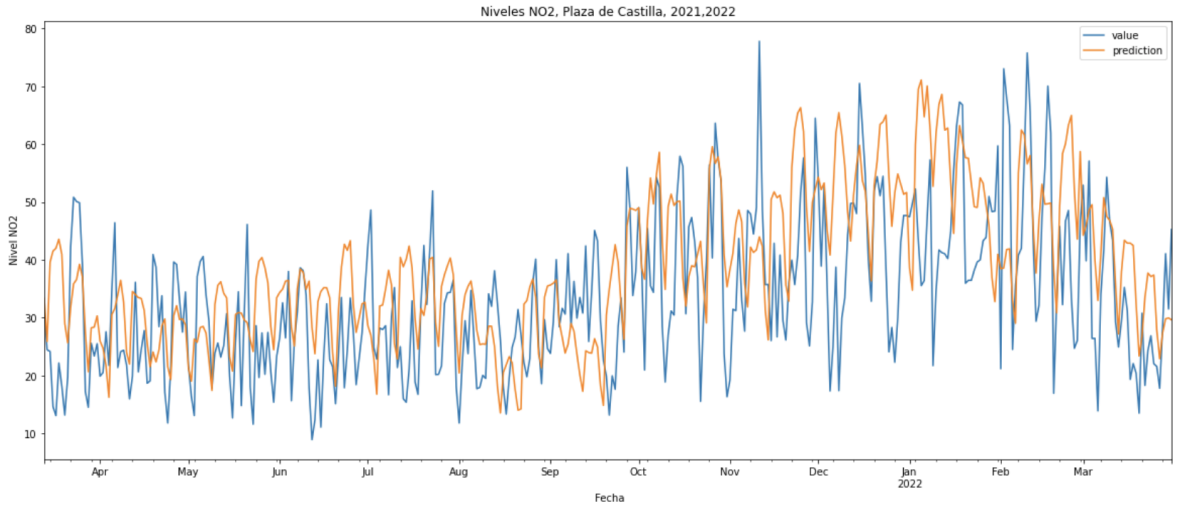

Figure 54: LSTM Test Forecast Plot

To compare both models, we are also interested in the mean error of their predictions. To calculate this deviation from the true value, we use the RMSE measure, or root mean square error. This statistic measures the amount of error between two types of sets; in our case, the actual and predicted values. Having studied the figures above, we can assume that the root mean square error of the LSTM model is higher than the RMSE of the ARIMA model. This is indeed the case (Table 3). Moreover, the error of the ARIMA is comparatively rather small.

Table 3: Comparison Test for ARIMA and LSTM

RMSE ARIMA	***RMSE LSTM***
0.104	3.64

5. OVERALL ANALYSIS AND CONCLUSIONS

5.1. Conclusions & Recommendations

Once having developed the models and demonstrated their predictive capabilities, it is interesting to study the positive impact that a predictive air quality system (coherent and complete) can have on the decision-making process of the Madrid City Council.

Both our LSTM and ARIMA model focus on the prediction of a single pollutant in a single station. Therefore, many of the difficulties usually experienced when dwelling with predictions are removed. When creating a generalized air quality model, working only with static data and limited measurements of each pollutant will result in being rather restrictive. As previously mentioned, the air quality system of the city of Madrid has only 24 (static) stations. Out of these, only 3 collect the entire

scope of pollutants recommended by the European Environment Agency (EEA) in their Air Quality Index. In extreme cases, such as in Vallecas, only one pollutant is measured (as depicted in Table 6 of the Appendix).

Initially, when developing both predictive models, it was intended to implement meteorological parameters to study how a change in weather conditions can affect air quality. However, of the 24 stations, only 14 have climatological sensors. On the other hand, only 7 measure precipitation levels. This is surprising since one of the biggest air pollution reducers is this climatological phenomenon. Rain is formed by the condensation of particles in the air. Condensation nuclei contain everything from pollen to PM2.5 (among other harmful particles). Rain washes pollutant particles to the ground, including pollutant particles (Kaupp and McLachlan, 1998). In other words, rain significantly reduces the level of pollution in the air. Knowing the impact that this and other climatological phenomena can have on the predictive models developed is of vital importance, since their implementation could lead to better decision making by the Madrid City Council. However, the scarce presence of rain gauges (and other meteorological sensors) in the air quality system of the capital has made it very difficult to generate pollution forecasts in Madrid.

When creating the ARIMA model, one of the first things done was study descriptive analytics relating to NO2 levels in the area of Plaza de Castilla (between January 2017 and January 2022). The average level of this pollutant found itself in the lowest range of contamination. This result juxtaposes studies stating that Madrid has the deadliest levels of NO2 across Europe. However, it is also notable that NO2 levels prior to COVID-19 were slightly higher than those after January 2020 (after mobility restrictions were put in place).

Throughout this thesis, the level of Nitrogen Dioxide in the air has been studied, under the assumption that today this pollutant is one of the most worrying among health experts. Moreover, as previously studied, an increase in this pollutant is generally marked by an increase in incomplete combustion in motor vehicles. Having emphasized this specific pollutant, it is interesting to know if there is a correlation between NO2 (and the other pollutants of interest), the number of passengers using public transport (divided into EMT and Metro Suburbano) and the traffic intensity on the different roads in Madrid.

As expected, nitrogen dioxide is positively correlated with an increase in traffic and with an increased use of public transport (Figure 55).

On the other hand, it is striking that there is a positive correlation (greater than 0.5) between the level of NO2 emissions and the use of the suburban metro. As explained by the Community of Madrid, "traveling by subway pollutes five times less than traveling by private vehicle". One possible explanation could be that on days when there is heavy traffic, people take the opportunity to make greater use of public transport. Checking the correlation between Metro_Suburban and any of the traffic variables confirms that this is the case. Therefore, as subway usage increases, we find higher NO2 values.

	Part_Suspension	SO2	CO	O3	NO2	Total_viajeros	EMT	Metro_Suburbano	Interior 1er cinturón	En el 1er cinturón	Entre 1er cinturón y 2º cinturón	En el 2º cinturón	Entre 2º cinturón y M-30	M-30	Entre M-30 y M-40
Part_Suspension	1	0.00388748	-0.01761164	0.13112579	0.20265775	-0.0047065	0.02623763	-0.02502957	0.23021372	0.152486	0.12090492	0.07323163	0.01304768	0.05503335	0.09292722
SO2	0.00388748	1	0.50559689	-0.40893732	0.55415257	0.31755444	0.27366019	0.33944078	0.12580874	0.12834028	0.09757154	0.19740543	0.19981833	0.16774028	0.27175689
CO	-0.01761164	0.50559689	1	-0.86614782	0.85660238	0.43806744	0.4089466	0.44750987	0.25466051	0.34120763	0.33588394	0.34271534	0.40545983	0.34188132	0.39244607
O3	0.13112579	-0.40893732	-0.86614782	1	-0.82927042	-0.36316721	-0.3209992	-0.38289541	-0.15049202	-0.26185496	-0.25801934	-0.26820776	-0.33917033	-0.24072698	-0.29085259
NO2	0.20265775	0.55415257	0.85660238	-0.82927042	1	0.51900125	0.49539721	0.52299442	0.42764882	0.46833779	0.46156643	0.46670748	0.50227507	0.45105704	0.52286129
Total_viajeros	-0.0047065	0.31755444	0.43806744	-0.36316721	0.51900125	1	0.97957824	0.99115162	0.69493462	0.76475786	0.82238007	0.76331156	0.88360323	0.9092958	0.84195556
EMT	0.02623763	0.27366019	0.4089466	-0.3209992	0.49539721	0.97957824	1	0.94422252	0.7402746	0.78351553	0.85053315	0.77402719	0.89724198	0.9242234	0.86533697
Metro_Suburbano	-0.02502957	0.33944078	0.44750987	-0.38289541	0.52299442	0.99115162	0.94422252	1	0.64948491	0.73529766	0.78543065	0.73919271	0.85486852	0.87913651	0.8077191
Interior 1er cinturón	0.23021372	0.12580874	0.25466051	-0.15049202	0.42764882	0.69493462	0.7402746	0.64948491	1	0.93842771	0.92230094	0.83676111	0.85428058	0.81655162	0.91346849
En el 1er cinturón	0.152486	0.12834028	0.34120763	-0.26185496	0.46833779	0.76475786	0.78351553	0.73529766	0.93842771	1	0.97052486	0.87625324	0.93339219	0.85911219	0.9416322
Entre 1er cinturón y 2º cinturón	0.12090492	0.09757154	0.33588394	-0.25801934	0.46156643	0.82238007	0.85053315	0.78543065	0.92230094	0.97052486	1	0.87643084	0.96928915	0.92186159	0.95528434
En el 2º cinturón	0.07323163	0.19740543	0.34271534	-0.26820776	0.46670748	0.76331156	0.77402719	0.73919271	0.83676111	0.87625324	0.87643084	1	0.87969494	0.84212604	0.87438526
Entre 2º cinturón y M-30	0.01304768	0.19981833	0.40545983	-0.33917033	0.50227507	0.88360323	0.89724198	0.85486852	0.85428058	0.93339219	0.96928915	0.87969494	1	0.94877326	0.96113719
M-30	0.05503335	0.16774028	0.34188132	-0.24072698	0.45105704	0.9092958	0.9242234	0.87913651	0.81655162	0.85911219	0.92186159	0.84212604	0.94877326	1	0.92088178
Entre M-30 y M-40	0.09292722	0.27175689	0.39244607	-0.29085259	0.52286129	0.84195556	0.86533697	0.8077191	0.91346849	0.9416322	0.95528434	0.87438526	0.96113719	0.92088178	1

Figure 55 : Correlation Pollutants, Public Transport and Traffic Levels

Considering the points priorly mentioned, the following recommendations are made to improve the air quality measurement and prediction system in Madrid. Implementing a stronger air quality system will give Madrid the opportunity to have deeper insight into their air quality. This will allow them to act proactively in their decision-making:

1. Firstly, multiple obstacles have been encountered when developing predictive models. These obstacles have arisen due to the scarcity of available data (both climatological and of different pollutants). Therefore, it is recommended to the City Council of Madrid to increase data collection at all stations. The implementation of meteorological gauges that allow the analysis of different climatological qualities should be encouraged. Furthermore, the collection (and consequent availability) of pollutants related to the European Air Quality Index should be guaranteed at all stations in Madrid.
2. As previously mentioned, Madrid is pointed out by the Barcelona Institute of Global Health as the European city with the highest NO2 pollution levels. Despite this study showing adequate levels of N02 in Plaza de Castilla (one of the areas with denser traffic in Madrid), there is indeed a positive correlation between traffic oscillation and N02 levels in the capital city. It is therefore recommended to add not only static but also mobile data by installing sensors on the roof of vehicles. This type of practice is already in place in European cities such as Antwerp, where they have implemented the sensEURcity project to increase the reliability of their systems (Van Poppel, 2022).
3. A pollution map with numerous static and dynamic measurements is recommended. The current one is based on an extrapolation by triangulation of the 24 available

stations. However, cities such as Brussels have implemented a system of sensors implemented in citizens' windows. The city has more than 3000 users in collaboration with this project. Other cities, such as London, have been able to identify pollution hot spots through the incorporation of dynamic sensors. For example, in the British capital, it was concluded that a high pollution point are school entrances. Numerous cars commuting to pick up or drop off children at school resulted in unhealthy air pollution levels. The practice of picking up children by car was consequentially banned. The upfront cost of investing in dynamic air quality sensors might be daunting. However, it can be beneficial in the long run.

4. As technology advances, new computation methods offer countless possibilities to innovate in this sector. One of the most recent innovations, designed by Barcelona's Supercomputation Center (BSC-CNS). Monarch, "will be one of the most advanced atmospheric air quality models in Europe that will contribute to the Copernicus Atmosphere Monitoring Service, the European Union's Earth observation program"(). This model makes accurate predictions up to four days past the prediction time. Implementing supercomputer technology, such as the one developed at CNS, would complement the sensor system currently at play with information obtained from environmental satellites.

In conclusion, it is recommended that the City Council increases their investment in an air quality prediction system that fully reflects pollutants of interest and climatological phenomena that may affect the level of these pollutants. Not only should this system be adjusted due to the measures and restrictions imposed by the EU, but also for the welfare of the people of Madrid.

6. APPENDIX

6.1. Table 1: Air Quality Scale

Table 1: Air Quality Scale					
Contaminants	**Excellent**	**Good**	**Average**	**Poor**	**Very Poor**
PM2,5	0-15	16-30	31-55	56-110	>110
PM10	0-25	26-50	51-90	91-180	>180
Nitrogen Dioxide (NO2)	0-50	51-100	101-200	201-400	>400
Ozone (O3)	0-60	61-120	121-180	181-240	>240
Sulfur Dioxide (SO2)	0-50	51-100	101-350	351-500	>500

6.2. Table 2: Data on Air Quality Stations (Madrid City Council)

Table 2: Datos de los Estaciones de calidad del Aire del Ayuntamiento de Madrid				
Area	**Code**	**Name**	**Pollutants Measure**	**Climatological Indicators measured**
Zona 01 (Interior M-30)	28079035	Pza. del Carmen	- Sulfur Dioxide - Carbon Monoxide - Nitrogen Dioxide - Nitrogen monoxide - Ozone	- Temperature - Humidity
	28079004	Pza. de España	- Sulfur Dioxide - Carbon Monoxide - Nitrogen Dioxide - Nitrogen monoxide	- Temperature
	28079039	Barrio del Pilar	- Nitrogen Dioxide - Nitrogen monoxide - Ozone	- Rainfall - Temperature - Humidity
	28079008	Escuelas Aguirre	- Sulfur Dioxide - Carbon Monoxide - Nitrogen Dioxide - Nitrogen monoxide - Particulate Matter PM10 - Particulate Matter PM2.5 - Ozone - Benzene	- Temperature - Humidity
	28079038	Cuatro Caminos	- Nitrogen Dioxide - Nitrogen monoxide - Particulate Matter PM10	- Temperature - Humidity

			- Particulate Matter PM2.5 - Benzene	
	28079011	Av. Ramón y Cajal	- Benzene - Nitrogen Dioxide	
	28079047	Méndez Álvaro	- Nitrogen Dioxide - Particulate Matter PM10 - Particulate Matter PM2.5	
	28079048	Paseo de la Castellana	- Nitrogen Dioxide - Particulate Matter PM10 - Particulate Matter PM2.5	
	28079049	Retiro	- Nitrogen Dioxide - Ozone	
	28079050	Pza. Castilla	- Nitrogen Dioxide - Particulate Matter PM10 - Particulate Matter PM2.5	
Zona 02 (Sureste)	28079013	Vallecas	- Nitrogen Dioxide - Particulate Matter PM10	
	28079036	Moratalaz	- Sulfur Dioxide - Nitrogen Dioxide - Nitrogen monoxide - Particulate Matter PM10	- Rainfall - Humidity
	28079054	Ensanche Vallecas	- Nitrogen Dioxide - Nitrogen monoxide - Ozone	- Wind direction - Wind speed - Rainfall - Solar Radiation - Temperature - Humidity
Zona 03 (Noreste)	28079016	Arturo Soria	- Nitrogen Dioxide - Nitrogen monoxide - Ozone	- Rainfall - Humidity
	28079027	Barajas Pueblo	- Nitrogen Dioxide - Ozone	
	28079055	Urb.Embajada (Barajas)	- Nitrogen Dioxide - Benzene - Particulate Matter PM10	
	28079057	Sanchinarro	- Sulfur Dioxide - Nitrogen Dioxide	

			- Particulate Matter PM10 - Particulate Matter PM2.5	
	28079059	Parque Juan Carlos I	- Nitrogen Dioxide - Nitrogen monoxide - Ozone	- Wind speed - Wind direction - Rainfall - Barometric pressure - Humidity - Solar Radiation - Temperature
	28079060	Tres Olivos	- Nitrogen Dioxide - Ozone - Particulate Matter PM10	
Zona 05 (Suroeste)	28079017	Villaverde Alto	- Nitrogen Dioxide - Ozone	
	28079056	Plaza Elíptica	- Carbon Monoxide - Nitrogen Dioxide - Nitrogen monoxide - Particulate Matter PM10 - Particulate Matter PM2.5	- Wind speed - Wind direction - Temperature - Humidity - Barometric pressure - Rainfall
	28079018	C/ Farolillo	- Carbon Monoxide - Nitrogen Dioxide - Nitrogen monoxide - Particulate Matter PM10 - Particulate Matter PM2.5 - Ozone - Benzene	- Temperature
Zona 04 (Noroeste)	28079024	Casa de Campo	- Nitrogen Dioxide - Nitrogen monoxide - Particulate Matter PM10 - Particulate Matter PM2.5 - Ozone - Benzene	- Wind speed - Wind direction - Rainfall - Barometric pressure - Humidity - Solar Radiation - Temperature
	28079058	El Pardo	- Nitrogen monoxide - Nitrogen Dioxide - Ozone	- Temperature - Humidity

6.3. Table 3: Pollutants, Units of Measurement and, Measurement Techniques

Table 3: Pollutants, Units of Measurement and, Measurement Techniques		
Pollutant	**Units of Measure**	**Measurement Technique**
Particulate Matter ($PM_{2,5}$ & PM_{10})	µg/m 3	Beta Absorption
Ozone (O_3)	µg/m 3	Ultraviolet Photometry
Nitrogen Dioxide (NO_2)	µg/m 3	Chemiluminescence
Sulfur Dioxide (SO_2)	µg/m 3	Ultraviolet Fluorescence
Carbon Monoxide (CO)	µg/m 3	Infrared Absorption
Lead (Pb)	µg/m 3	Particulate Matter PM10 capture in a filter.
Benzene (C6H6)	µg/m 3	Particulate Matter PM10 capture in a filter.
Arsenic (As)	µg/m 3	Particulate Matter PM10 capture in a filter.
Cadmium (Cd)	µg/m 3	Particulate Matter PM10 capture in a filter.
Nickel (Ni)	µg/m 3	Particulate Matter PM10 capture in a filter.
Benzo(a)pyrene (B(a)P)	µg/m 3	Particulate Matter PM10 capture in a filter.

6.4. Table 4: Pollutants Collected by the Stations of Air Quality of Madrid

Table 4: Pollutants Collected by the Stations of Air Quality of Madrid		
Code	**Pollutant**	**Units of Measure**
01	Sulfur Dioxide (SO2)	µg/m 3
06	Carbon Monoxide (CO)	mg/m3
07	Nitrogen Monoxide (NO)	µg/m3
08	Nitrogen Dioxide (NO2)	µg/m 3
09	Particles < 2.5 µm PM2.5	µg/m3
10	Particles < 10 µm (PM10)	µg/m3

12	Nitrogen Oxides (NOx)	µg/m3
14	Ozone (O3)	µg/m3
20	Toluene (TOL)	µg/m3
30	Benzene (BEN)	µg/m3
35	EtilBenzene (EBE)	µg/m3
37	Metaxylene (MXY)	µg/m3
38	Paraxylene (PXY)	µg/m3
39	Orthoxylene (OXY)	µg/m3
42	Total hydrocarbons (hexane) (TCH)	mg/m3
43	Methane (CH4)	mg/m3
44	Non-methane hydrocarbons (hexane) (NMHC)	mg/m3

6.5. Table 5: Meteorological Parameters Collected by the Air Quality Stations

Table 5: Meteorological Parameters Collected by the Air Quality Stations		
Code	**Parameter**	**Unidad de Medida**
80	Ultraviolet Radiation	Mw/m2
81	Wind Speed	m/s
82	Wind Direction	-
83	Temperature	oC
86	Humidity	%
87	Barometric Pressure	mb
88	Solar Radiation	W/m2
89	Rainfall	l/m2

6.6. Table 6: Pollutants of interest measured at each station

Table 6: Pollutants of interest measured at each station							
Code	**Station**	**PM2.5**	**PM10**	**O3**	**NO2**	**SO2**	**CO**
28079035	Pza. del Carmen						
28079004	Pza. de España						
28079039	Barrio del Pilar						
28079008	Escuelas Aguirre						
28079038	Cuatro Caminos						
28079011	Av. Ramón y Cajal						
28079047	Méndez Álvaro						
28079048	Paseo de la Castellana						
28079049	Retiro						
28079050	Pza. Castilla						
28079013	Vallecas						
28079036	Moratalaz						
28079054	Ensanche Vallecas						
28079016	Arturo Soria						
28079027	Barajas Pueblo						
28079055	Urb.Embajada (Barajas)						
28079057	Sanchinarro						
28079059	Parque Juan Carlos I						
28079060	Tres Olivos						
28079017	Villaverde Alto						
28079056	Plaza Elíptica						
28079018	C/ Farolillo						
28079024	Casa de Campo						
28079058	El Pardo						

6.7. ARIMA: DATOS_CONT_DATOSHORARIOS

6.7.1. Hourly Values NO2, Pza. Castilla (January 2017- March 2022)

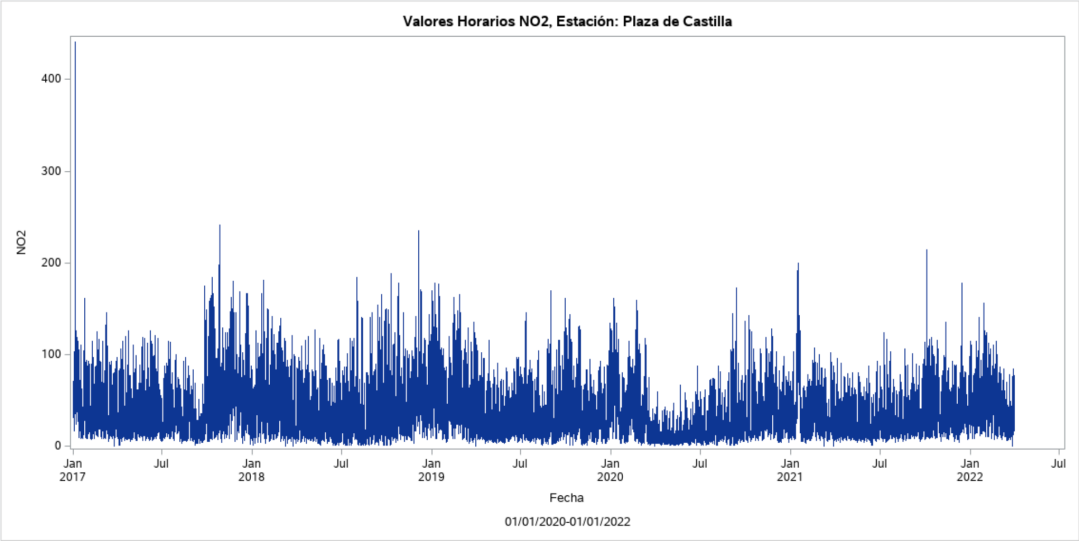

Figure 1: Hourly Values NO2, Pza. Castilla (January 2017- March 2022)

6.7.2. Descriptive Statistics Hourly Values NO2, Pza. Castilla

Name of Variable = VALOR_HORARIO_CONT	
Mean of Working Series	36.60239
Standard Deviation	27.64031
Number of Observations	36787

Figure 2: Descriptive Statistics Hourly Values NO2, Pza. Castilla

6.7.3. Correlation Analysis, Hourly Values NO2, Pza. Castilla.

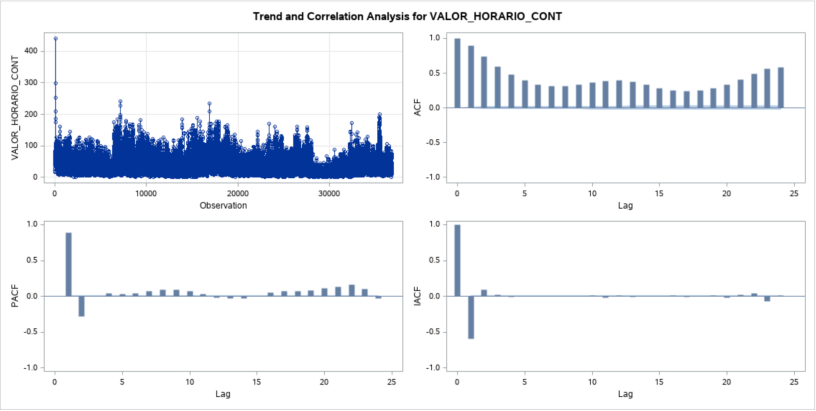

Figure 4: Correlation Analysis, Hourly Values NO2, Pza. Castilla.

6.7.4. Augmented Dickey-Fuller Test, Hourly Values NO2, Pza. Castilla

Augmented Dickey-Fuller Unit Root Tests							
Type	Lags	Rho	Pr < Rho	Tau	Pr < Tau	F	Pr > F
Zero Mean	0	-1446.90	0.0001	-27.17	<.0001		
	1	-2335.66	0.0001	-34.18	<.0001		
	2	-2053.61	0.0001	-31.61	<.0001		
Single Mean	0	-3983.55	0.0001	-45.89	<.0001	1052.82	0.0010
	1	-7081.36	0.0001	-59.50	<.0001	1770.22	0.0010
	2	-7022.34	0.0001	-56.61	<.0001	1602.47	0.0010
Trend	0	-4064.36	0.0001	-46.38	<.0001	1075.42	0.0010
	1	-7248.12	0.0001	-60.20	<.0001	1811.85	0.0010
	2	-7220.08	0.0001	-57.33	<.0001	1643.50	0.0010

Figure 5: Augmented Dickey-Fuller Stationarity Test, Hourly Values NO2, Pza. Castilla

6.7.5. Autocorrelation Check for White Noise, Hourly Values NO2, Pza. Castilla

Autocorrelation Check for White Noise									
To Lag	Chi-Square	DF	Pr > ChiSq	Autocorrelations					
6	9999.99	6	<.0001	0.892	0.738	0.593	0.478	0.394	0.338
12	9999.99	12	<.0001	0.310	0.309	0.330	0.361	0.387	0.391
18	9999.99	18	<.0001	0.371	0.332	0.287	0.254	0.243	0.254
24	9999.99	24	<.0001	0.285	0.337	0.406	0.487	0.559	0.588

Figure 6: Autocorrelation Check for White Noise, Hourly Values NO2, Pza. Castilla

6.7.6. Parameter Estimation

Conditional Least Squares Estimation					
Parameter	Estimate	Standard Error	t Value	Approx Pr > |t|	Lag
MU	36.60054	0.45282	80.83	<.0001	0
MA1,1	-0.01089	0.01868	-0.58	0.5597	1
AR1,1	1.13133	0.01798	62.93	<.0001	1
AR1,2	-0.27103	0.01620	-16.73	<.0001	2

Figure 7: Parameter Estimation ARIMA(2,0,0)

Conditional Least Squares Estimation					
Parameter	Estimate	Standard Error	t Value	Approx Pr > |t|	Lag
MU	36.60054	0.45282	80.83	<.0001	0
MA1,1	-0.01089	0.01868	-0.58	0.5597	1
AR1,1	1.13133	0.01798	62.93	<.0001	1
AR1,2	-0.27103	0.01620	-16.73	<.0001	2

Figure 8: Parameter Estimation ARIMA(2,0,1)

Conditional Least Squares Estimation					
Parameter	Estimate	Standard Error	t Value	Approx Pr > |t|	Lag
MU	36.60036	0.45852	79.82	<.0001	0
MA1,1	-0.23031	0.05568	-4.14	<.0001	1
MA1,2	-0.07977	0.01684	-4.74	<.0001	2
AR1,1	0.91155	0.05579	16.34	<.0001	1
AR1,2	-0.09027	0.04699	-1.92	0.0547	2

Figure 9: Parameter Estimation ARIMA(2,0,2)

6.7.7. Goodness of Fit Statistics

Constant Estimate	5.0732
Variance Estimate	144.2514
Std Error Estimate	12.01047
AIC	287288.7
SBC	287314.2
Number of Residuals	36787

Figure 10: Goodness of Fit Statistics ARIMA(2,0,0)

Constant Estimate	5.113005
Variance Estimate	144.2536
Std Error Estimate	12.01056
AIC	287290.2
SBC	287324.3
Number of Residuals	36787

Figure 11: Goodness of Fit Statistics ARIMA(2,0,1)

Constant Estimate	6.541281
Variance Estimate	144.1459
Std Error Estimate	12.00608
AIC	287263.8
SBC	287306.3
Number of Residuals	36787

Figure 12: Goodness of Fit Statistics ARIMA(2,0,2)

6.7.8. Correlation of Parameter Estimates

Correlations of Parameter Estimates			
Parameter	MU	AR1,1	AR1,2
MU	1.000	0.000	-0.000
AR1,1	0.000	1.000	-0.892
AR1,2	-0.000	-0.892	1.000

Figure 13: Correlation of Parameter Estimates ARIMA(2,0,0)

Correlations of Parameter Estimates				
Parameter	MU	MA1,1	AR1,1	AR1,2
MU	1.000	0.000	0.000	-0.000
MA1,1	0.000	1.000	0.960	-0.951
AR1,1	0.000	0.960	1.000	-0.990
AR1,2	-0.000	-0.951	-0.990	1.000

Figure 14: Correlation of Parameter Estimates ARIMA(2,0,1)

Correlations of Parameter Estimates					
Parameter	MU	MA1,1	MA1,2	AR1,1	AR1,2
MU	1.000	0.000	0.000	0.000	-0.000
MA1,1	0.000	1.000	0.943	0.996	-0.994
MA1,2	0.000	0.943	1.000	0.942	-0.930
AR1,1	0.000	0.996	0.942	1.000	-0.998
AR1,2	-0.000	-0.994	-0.930	-0.998	1.000

Figure 15: Correlation of Parameter Estimates ARIMA(2,0,2)

6.7.9. Autocorrelation Check for Residuals

Autocorrelation Check of Residuals									
To Lag	Chi-Square	DF	Pr > ChiSq	Autocorrelations					
6	35.92	4	<.0001	0.001	0.007	-0.021	-0.012	-0.010	-0.015
12	840.01	10	<.0001	-0.023	-0.020	0.009	0.047	0.096	0.097
18	1243.95	16	<.0001	0.087	0.047	-0.003	-0.029	-0.017	-0.010
24	4510.27	22	<.0001	-0.010	0.001	-0.008	0.049	0.151	0.252
30	5514.18	28	<.0001	0.156	0.039	-0.011	-0.018	-0.003	-0.031
36	6110.67	34	<.0001	-0.019	-0.024	0.000	0.034	0.082	0.085
42	6498.18	40	<.0001	0.084	0.041	-0.005	-0.029	-0.023	-0.020
48	8811.39	46	<.0001	-0.008	-0.009	-0.017	0.031	0.131	0.210

Figure 16: Autocorrelation Check for Residuals ARIMA(2,0,0)

Autocorrelation Check of Residuals									
To Lag	Chi-Square	DF	Pr > ChiSq	Autocorrelations					
6	40.17	3	<.0001	0.000	0.009	-0.022	-0.013	-0.011	-0.016
12	844.03	9	<.0001	-0.024	-0.020	0.009	0.047	0.096	0.097
18	1248.99	15	<.0001	0.087	0.047	-0.003	-0.029	-0.017	-0.011
24	4516.82	21	<.0001	-0.011	0.000	-0.008	0.049	0.151	0.252
30	5524.78	27	<.0001	0.156	0.040	-0.011	-0.018	-0.004	-0.031
36	6122.34	33	<.0001	-0.019	-0.025	0.000	0.034	0.082	0.085
42	6511.91	39	<.0001	0.084	0.041	-0.005	-0.029	-0.024	-0.021
48	8826.04	45	<.0001	-0.008	-0.010	-0.017	0.031	0.131	0.210

Figure 17: Autocorrelation Check for Residuals ARIMA(2,0,1)

Autocorrelation Check of Residuals									
To Lag	Chi-Square	DF	Pr > ChiSq	Autocorrelations					
6	35.18	2	<.0001	0.000	-0.001	-0.002	-0.017	-0.016	-0.021
12	822.90	8	<.0001	-0.027	-0.022	0.008	0.046	0.094	0.095
18	1226.31	14	<.0001	0.085	0.047	-0.004	-0.031	-0.019	-0.014
24	4469.73	20	<.0001	-0.013	0.000	-0.006	0.050	0.151	0.250
30	5477.10	26	<.0001	0.155	0.040	-0.009	-0.018	-0.006	-0.034
36	6066.79	32	<.0001	-0.022	-0.025	-0.001	0.034	0.081	0.084
42	6458.72	38	<.0001	0.083	0.041	-0.005	-0.029	-0.026	-0.024
48	8754.97	44	<.0001	-0.010	-0.009	-0.015	0.032	0.131	0.209

Figure 18: Autocorrelation Check for Residuals ARIMA(2,0,2)

6.7.10. Residual Correlation Diagnostic

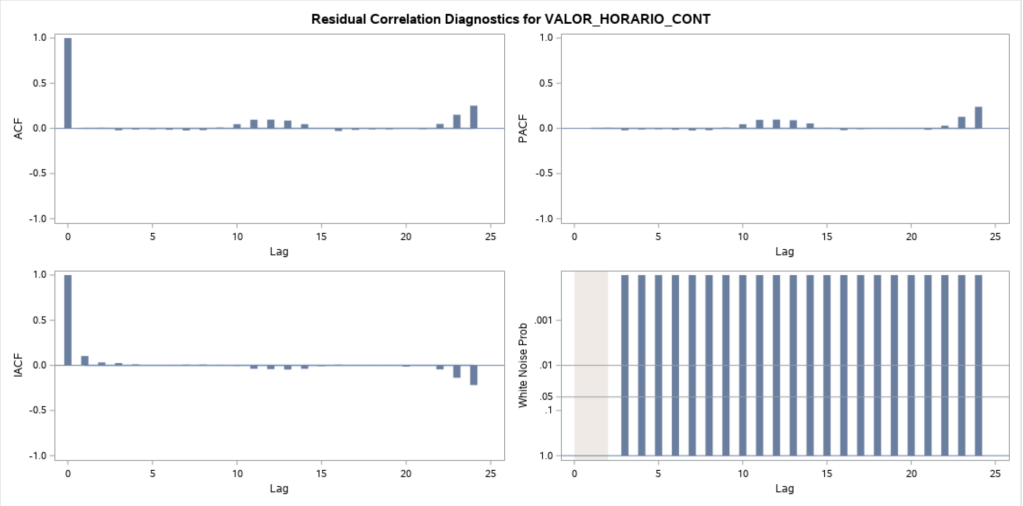

Figure 19: Residual Correlation Diagnostic ARIMA(2,0,0)

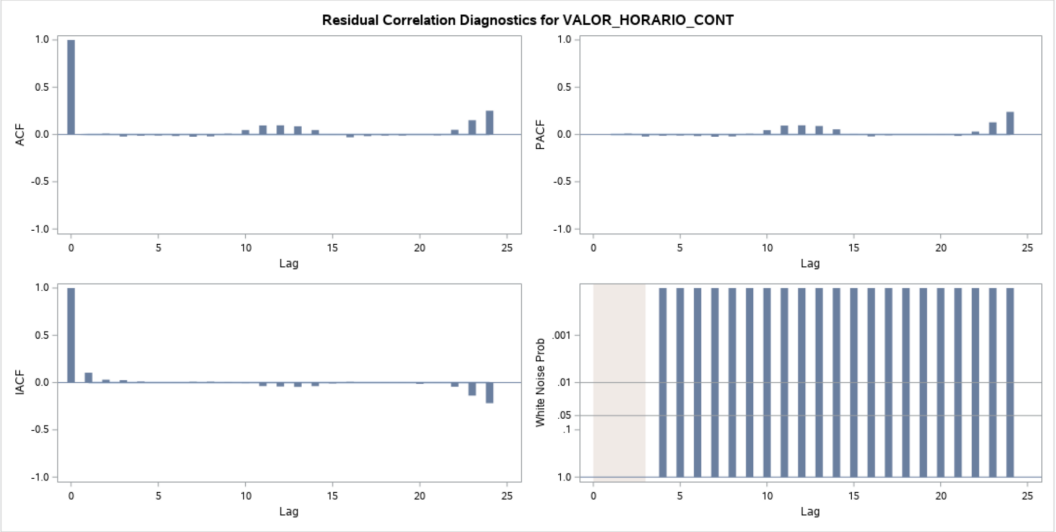

Figure 20: Residual Correlation Diagnostic ARIMA(2,0,1)

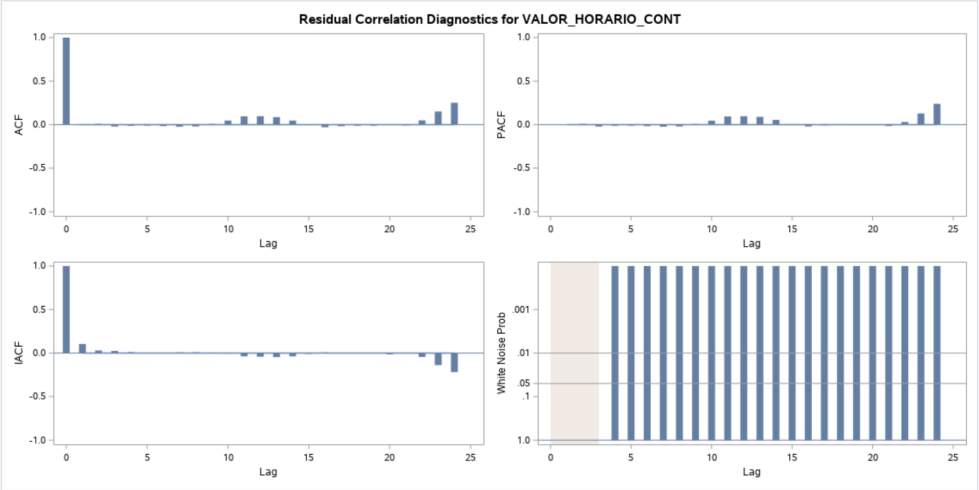

Figure 21: Residual Correlation Diagnostic ARIMA(2,0,2)

6.7.11. Normality Check of Residuals

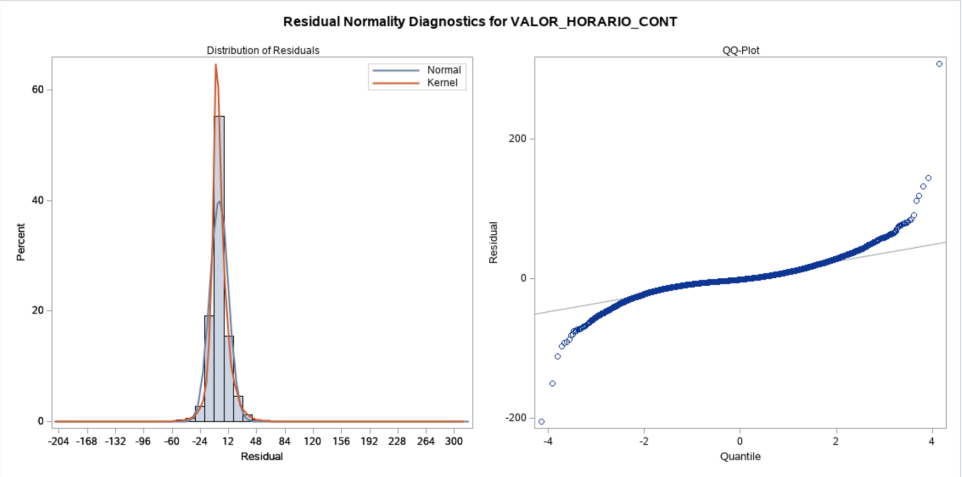

Figure 22: Normality Check of Residuals ARIMA(2,0,0)

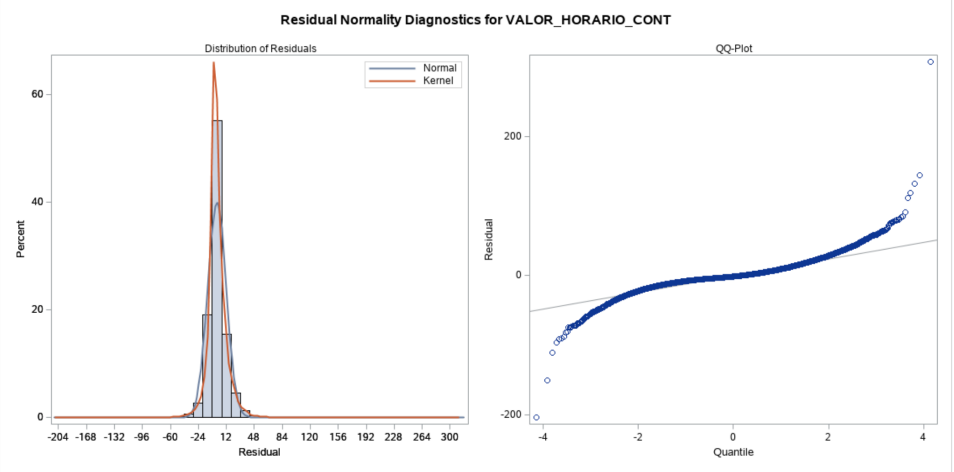

Figure 23: Normality Check of Residuals ARIMA(2,0,1)

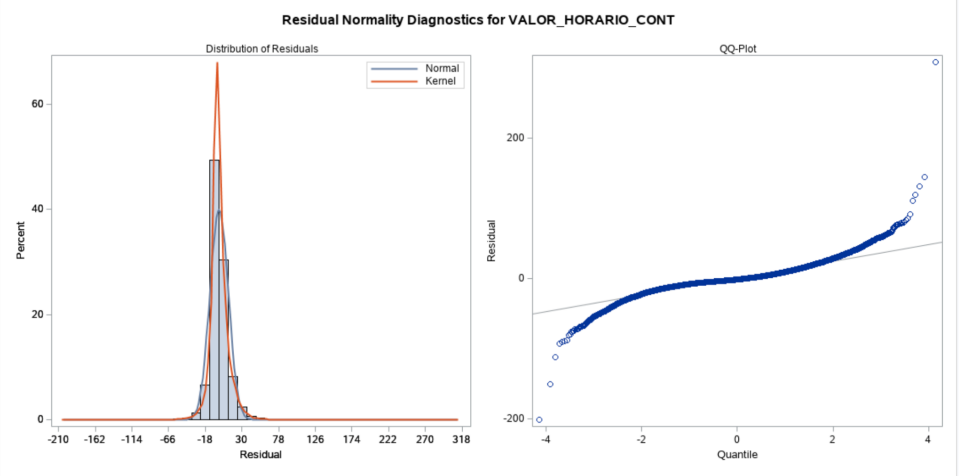

Figure 24: Normality Check of Residuals ARIMA(2,0,2)

7. BIBLIOGRAPHY

A&E Television Networks. (2009, November 13). *Killer smog claims elderly victims*. History.com. Retrieved June 24, 2022, from https://www.history.com/this-day-in-history/killer-smog-claims-elderly-victims

Antwerp Pilot City for project with Air Quality Sensors. VITO. (n.d.). Retrieved June 24, 2022, from https://vito.be/en/news/antwerp-pilot-city-project-air-quality-sensors

Antwerp Pilot City for project with Air Quality Sensors. VITO. (n.d.). Retrieved June 24, 2022, from https://vito.be/en/news/antwerp-pilot-city-project-air-quality-sensors

Análisis de series Temporales Modelos Arima . Sarriko - On. (n.d.). Retrieved June 24, 2022, from https://addi.ehu.es/bitstream/handle/10810/12492/04-09gon.pdf;sequence=1

APMonitorCom. (2020, January 17). *Python 🐍 LSTM Network*. YouTube. Retrieved June 24, 2022, from https://www.youtube.com/watch?v=LiBFV7ptm4M&ab_channel=APMonitor.com

Barboza, A. V. (n.d.). *Serie temporal - definición, qué es y Concepto*. Economipedia. Retrieved June 24, 2022, from https://economipedia.com/definiciones/serie-temporal.html

By: IBM Cloud Education. (n.d.). *What are recurrent neural networks?* IBM. Retrieved June 24, 2022, from https://www.ibm.com/cloud/learn/recurrent-neural-networks

Cabana, E. (2022, February 28). *Ruido Blanco (white noise) en python y en r*. Medium. Retrieved June 24, 2022, from https://ecab-estadistica.medium.com/ruido-blanco-white-noise-en-python-y-en-r-dbd87d4cd1fe

Calidad del Aire. Comunidad de Madrid. (2022, June 23). Retrieved June 24, 2022, from https://www.comunidad.madrid/servicios/urbanismo-medio-ambiente/calidad-aire#:~:text=La%20Red%20de%20Calidad%20del%20Aire%20de%20la%20Comunidad%20de,Di%C3%B3xido%20de%20azufre%20%E2%80%93%20SO2

Calidad del Aire. Comunidad de Madrid. (2022, June 23). Retrieved June 24, 2022, from https://www.comunidad.madrid/servicios/urbanismo-medio-ambiente/calidad-aire#:~:text=La%20Red%20de%20Calidad%20del%20Aire%20de%20la%20Comunidad%20de,Di%C3%B3xido%20de%20azufre%20%E2%80%93%20SO2

Calidad y Evaluación Ambiental. (n.d.). Retrieved June 24, 2022, from https://www.miteco.gob.es/es/calidad-y-evaluacion-ambiental/temas/

Composición de la Atmósfera. (n.d.). Retrieved June 24, 2022, from https://www.imn.ac.cr/documents/10179/27818/Comp-atmosfera.pdf

CSCI 1460 : Introduction to Computational Linguistics. CS146 | Brown University. (n.d.). Retrieved June 24, 2022, from https://cs.brown.edu/courses/csci1460

Deep learning cheatsheet . CS 229 - Deep Learning Cheatsheet. (n.d.). Retrieved June 24, 2022, from https://stanford.edu/~shervine/teaching/cs-229/cheatsheet-deep-learning

Difference between LSTM cell state and Hidden State. Data Science Stack Exchange. (1968, May 1). Retrieved June 24, 2022, from https://datascience.stackexchange.com/questions/82808/difference-between-lstm-cell-state-and-hidden-state

Environmental Protection Agency. (n.d.). *Carbon Monoxide (CO) Pollution in Outdoor Air*. EPA. Retrieved June 24, 2022, from https://www.epa.gov/co-pollution/basic-information-about-carbon-monoxide-co-outdoor-air-pollution#Effects

Environmental Protection Agency. (n.d.). *Criteria Air Pollutants*. EPA. Retrieved June 24, 2022, from https://www.epa.gov/criteria-air-pollutants

Environmental Protection Agency. (n.d.). *Ground-level Ozone Pollution*. EPA. Retrieved June 24, 2022, from https://www.epa.gov/ground-level-ozone-pollution/ground-level-ozone-basics

Environmental Protection Agency. (n.d.). *Health and Environmental Effects of Particulate Matter (PM)*. EPA. Retrieved June 24, 2022, from https://www.epa.gov/pm-pollution/health-and-environmental-effects-particulate-matter-pm

Environmental Protection Agency. (n.d.). *History of Air Pollution*. EPA. Retrieved June 24, 2022, from https://www.epa.gov/air-research/history-air-pollution

Environmental Protection Agency. (n.d.). *Nitrogen Dioxide (NO2) Pollution*. EPA. Retrieved June 24, 2022, from https://www.epa.gov/no2-pollution/basic-information-about-no2#What%20is%20NO2

Environmental Protection Agency. (n.d.). *Particulate Matter (PM) Pollution*. EPA. Retrieved June 24, 2022, from https://www.epa.gov/pm-pollution/particulate-matter-pm-basics#PM

Environmental Protection Agency. (n.d.). *Sulfur Dioxide (SO2) Pollution*. EPA. Retrieved June 24, 2022, from https://www.epa.gov/so2-pollution/sulfur-dioxide-basics#what%20is%20so2

Estrategia de Calidad del Aire y cambio climático de la Comunidad de Madrid (2013-2020). plan azul+. Portal de Transparencia. (2019, September 17). Retrieved June 24, 2022, from https://www.comunidad.madrid/transparencia/informacion-institucional/planes-programas/estrategia-calidad-del-aire-y-cambio-climatico-comunidad

Estrategia de Calidad del Aire y cambio climático de la Comunidad de Madrid (2013-2020). Portal de Transparencia. (2019, September 17). Retrieved June 24, 2022, from https://www.comunidad.madrid/transparencia/informacion-institucional/planes-programas/estrategia-calidad-del-aire-y-cambio-climatico-comunidad

García Martos, D. (n.d.). *Econometría II Grado en finanzas y contabilidad*. Econometra II. Retrieved June 24, 2022, from http://www.est.uc3m.es/esp/nueva_docencia/comp_col_get/lade/Econometria_II_NOdocencia/Documentaci%C3%B3n%20y%20apuntes.html

Kumar, N. (2019, December 18). *Sigmoid neuron - deep neural networks*. Medium. Retrieved June 24, 2022, from https://towardsdatascience.com/sigmoid-neuron-deep-neural-networks-a4cd35b629d7#:~:text=The%2

La Calidad del Aire en la Ciudad de Madrid Durante 2021. (2022). Retrieved June 25, 2022, from https://ecologistasenaccion.org/wp-content/uploads/2022/01/informe-calidad-aire_madrid-2021.pdf

The lancet | the best science for better lives. Premature mortality due to air pollution in European cities: a health impact assessment. (n.d.). Retrieved June 25, 2022, from https://www.thelancet.com/action/showPdf?pii=S2542-5196%2820%2930121-2

*Madrid 360 * Madrid 360*. Madrid 360. (2022, March 11). Retrieved June 24, 2022, from https://www.madrid360.es/

Madrid 360, La Estrategia Para Cumplir con Los Objetivos de Calidad del Aire de la Unión Europea - Ayuntamiento de Madrid. MADRID 360, la estrategia para cumplir con los objetivos de calidad del aire de la Unión Europea - Ayuntamiento de Madrid. (2019, September 9). Retrieved June 24, 2022, from

https://www.madrid.es/portales/munimadrid/es/Inicio/Actualidad/Noticias/MADRID-360-la-estrategia-para-cumplir-con-los-objetivos-de-calidad-del-aire-de-la-Union-Europea/?vgnextfmt=default&vgnextoid=3d6c1609d818d610VgnVCM2000001f4a900aRCRD&vgnextchannel=a12149fa40ec9410VgnVCM100000171f5a0aRCRD

Madrid es la Ciudad Europea con mayor Nivel de Contaminación Por No2 y Mortalidad Asociada. Agencia SINC. (n.d.). Retrieved June 24, 2022, from https://www.agenciasinc.es/Noticias/Madrid-es-la-ciudad-europea-con-mayor-nivel-de-contaminacion-por-NO2-y-mortalidad-asociada#:~:text=de%20la%20Vida-,Madrid%20es%20la%20ciudad%20europea%20con%20mayor%20nivel%20de%20contaminaci%C3%B3n,sus%20efectos%20en%20la%20salud.

Parra, F. (2019, January 5). *Estadística y machine learning con r*. 8 Series Temporales. Retrieved June 24, 2022, from https://bookdown.org/content/2274/series-temporales.html

Portal de calidad del aire. Detalle por estación. (n.d.). Retrieved June 24, 2022, from https://airedemadrid.madrid.es/portales/calidadaire/es/Bases-de-datos-y-publicaciones/Bases-de-datos-de-calidad-del-aire/Detalle-por-estacion/?vgnextfmt=default&vgnextchannel=da2bafcb8f548710VgnVCM1000001d4a900aRCRD

Portal de calidad del aire. Índice de Calidad del Aire. (n.d.). Retrieved June 24, 2022, from https://airedemadrid.madrid.es/portales/calidadaire/es/Bases-de-datos-y-publicaciones/Bases-de-datos-de-calidad-del-aire/Indices-y-zonas/Indice-de-calidad-del-aire/?vgnextfmt=default&vgnextoid=303d635a41187710VgnVCM1000001d4a900aRCRD&vgnextchannel=480285a1259d7710VgnVCM2000001f4a900aRCRD

Portal de Datos Abiertos del Ayuntamiento de Madrid. Datos de uso del portal. (n.d.). Retrieved June 24, 2022, from https://datos.madrid.es/portal/site/egob/menuitem.400a817358ce98c34e937436a8a409

a0/?vgnextoid=d11ce2e5b6801610VgnVCM1000001d4a900aRCRD&vgnextchannel=d11ce2e5b6801610VgnVCM1000001d4a900aRCRD&vgnextfmt=default

Portal de Datos Abiertos del Ayuntamiento de Madrid. En portada. (n.d.). Retrieved June 24, 2022, from https://datos.madrid.es/portal/site/egob

Red de Calidad del Aire de la Comunidad de Madrid. Comunidad de Madrid. (n.d.). Retrieved June 24, 2022, from http://gestiona.madrid.org/azul_internet/html/web/3.htm?ESTADO_MENU=3#:~:text=La%20caracterizaci%C3%B3n%20de%20las%2024,y%20benzo(a)pireno.

Rodó, P. (n.d.). *Modelo Arma*. Economipedia. Retrieved June 24, 2022, from https://economipedia.com/definiciones/modelo-arma.html

Servimedia. (2021, November 10). *Madrid es la ciudad europea con más muertes por aire contaminado de los coches*. ELMUNDO. Retrieved June 24, 2022, from https://www.elmundo.es/ciencia-y-salud/medio-ambiente/2021/11/11/618c5be5fdddff039e8b45e3.html

Smart Cities: Sostenibilidad. Visit 1drv.ms - Microsoft OneDrive. (1969, December 5). Retrieved June 24, 2022, from https://links.giveawayoftheday.com/1drv.ms

Tian, X., Cui, K., Sheu, H.-L., Hsieh, Y.-K., & Yu, F. (2021, July 27). *Effects of rain and snow on the air quality index, PM2.5 levels, and dry deposition flux of PCDD/Fs*. Aerosol and Air Quality Research. Retrieved June 24, 2022, from https://aaqr.org/articles/aaqr-21-06-oa-0158#:~:text=Because%20rain%20depends%20on%20particles,the%20horizontal%20transport%20of%20pollutants.

Time series Arima Models. YouTube. (2013, December 28). Retrieved June 24, 2022, from https://youtu.be/Y2khrpVo6qI

Understanding LSTM networks. Understanding LSTM Networks -- colah's blog. (n.d.). Retrieved June 24, 2022, from https://colah.github.io/posts/2015-08-Understanding-LSTMs/

United Nations. (n.d.). *Objetivos y Metas de Desarrollo Sostenible - Desarrollo Sostenible*. United Nations. Retrieved June 24, 2022, from https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/

World Health Organization. (2018, May 2). *Nueve de Cada Diez personas de Todo El Mundo respiran aire contaminado sin embargo, cada vez hay más países que toman medidas*. World Health Organization. Retrieved June 24, 2022, from https://www.who.int/es/news/item/02-05-2018-9-out-of-10-people-worldwide-breathe-polluted-air-but-more-countries-are-taking-action

World Health Organization. (2021, September 21). *Calidad del Aire Ambiente (exterior)* . World Health Organization. Retrieved June 24, 2022, from https://www.who.int/es/news-room/fact-sheets/detail/ambient-(outdoor)-air-quality-and-health

World Health Organization. (2021, September 22). *Calidad del Aire Ambiente (exterior)*. World Health Organization. Retrieved June 24, 2022, from https://www.who.int/es/news-room/fact-sheets/detail/ambient-(outdoor)-air-quality-and-health

YouTube. (2017, October 16). *Gradient descent, how neural networks learn | Chapter 2, Deep Learning*. YouTube. Retrieved June 24, 2022, from https://www.youtube.com/watch?v=IHZwWFHWa-w&ab_channel=3Blue1Brown

¿Qué es el Ozone troposférico? Medio Ambiente y Sostenibilidad. (n.d.). Retrieved June 24, 2022, from https://mediambient.gencat.cat/es/05_ambits_dactuacio/atmosfera/qualitat_de_laire/av

aluacio/campanya_ozo/que_es_lozo_troposferic/#:~:text=El%20Ozone%20troposf%C3%A9rico%20de%20origen,descargas%20el%C3%A9ctricas%20de%20una%20tormenta.

Índice de Calidad del Aire. (n.d.). Retrieved June 24, 2022, from https://www.miteco.gob.es/es/calidad-y-evaluacion-ambiental/temas/atmosfera-y-calidad-del-aire/calidad-del-aire/ICA.aspx