Master's in Industrial Technologies

Master's Final Thesis

Solar and Wind Power Generation Modeling for the Spanish Electrical System

Author: Fernando Celaya

Supervisors: Andrés Ramos and Jesús David Gómez

November 5, 2024

Abstract

This work evaluates models for long-term modeling of hourly capacity factors for solar PV, solar TH, and wind power, aiming to support grid planners and operators in a context of higher renewable integration. Twelve models were tested across distribution fit, temporal consistency, coherence, and extreme value accuracy. Findings reveal model suitability varies: the N-HiTS best captures solar PV frequency components which are the main driving factor for this series, iTransformer excels by best handling solar TH higher complexity, and VARMAX is optimal for wind's lower signal to noise ratio. A custom SERA loss function improved extreme wind event prediction, though challenges remain in accurately modeling wind data.

Keywords — Renewable energy modeling, solar, wind, Spanish electrical system, time series forecasting, loss function, model evaluation.

Acknowledgements

I would like to express my deepest gratitude to my supervisors, Andrés Ramos and Jesús David Gómez, for their guidance and support during this project which helped it to become a more interesting and valuable work than what it would have been without them.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Background

For the last few decades, the global energy landscape has been undergoing a significant transformation, driven by climate change and the need to reduce greenhouse gas emissions in order to reduce its impact. Europe has been at the forefront of this transformation and Spain, as its fifth-largest economy, has had a significant part in this. In fact, Spain has shown drive of its own by being at the forefront of many of these initiatives with a robust commitment to renewable energy sources. This country has set ambitious targets for renewable energy integration, setting a target of having 74% of its energy coming from renewable generation facilities by 2030 and a 42% share of renewables in energy end use, as per the 2021 Spain Energy Policy Review Agency, 2021. Spain is particularly well positioned for this transition due in part to its favorable geographical conditions. Being one of the southernmost parts of Europe with a Mediterranean climate allows for long periods of intense sunshine, and its extensive coast also provides good conditions for regular and powerful wind.

Like any other industrialized country, historically Spain has relied heavily on fossil fuels. The recent growth of renewable energy has been driven by several factors, including technological advancements, a reduction in costs driven by the economies of scale in production mainly in China and by government policies. Amongst these policies, the first big push came from the Royal Decree 661/2007 through which the production of renewable energy in Spain was regulated T. y. C. Ministerio de Industria, 2007. In this legislation, a premium was awarded to producers of renewable energy in order to incentivize investment in the sector. However, as soon as 2010 these premiums started to be reduced due to the general deficit in the sector and the financial constraints imposed on the state by the 2008 financial crisis. These reductions culminated in 2013 with a drop in 40% of all premiums available at that point. This led to the bankruptcy of a great number of companies which had invested heavily mainly in solar power plants and which relied on these subsidies for their required profitability. In 2015 the sector was further regulated through what came to be known as the "Sun Tax" E. y. T. Ministerio de Industria, 2015. This royal decree imposed a tax on self-consumption systems – individuals or companies installing solar PV panels to generate part of their own electricity – in order to compensate for the additional system maintenance needed by these systems. Since then, thanks to the drop in costs these subsidies have stopped being necessary with solar energy being profitable without government intervention, and some argue even the most profitable source of energy Glenk and Reichelstein, 2022. This advantageous financial landscape, together with a recent regulatory push driven mainly from the European Union, has prompted a new golden age of investment in renewable energy in Spain. In Figure 1 the growing trend in investment can be clearly seen, except for the drop in 2021 created by the Covid-19 crisis.

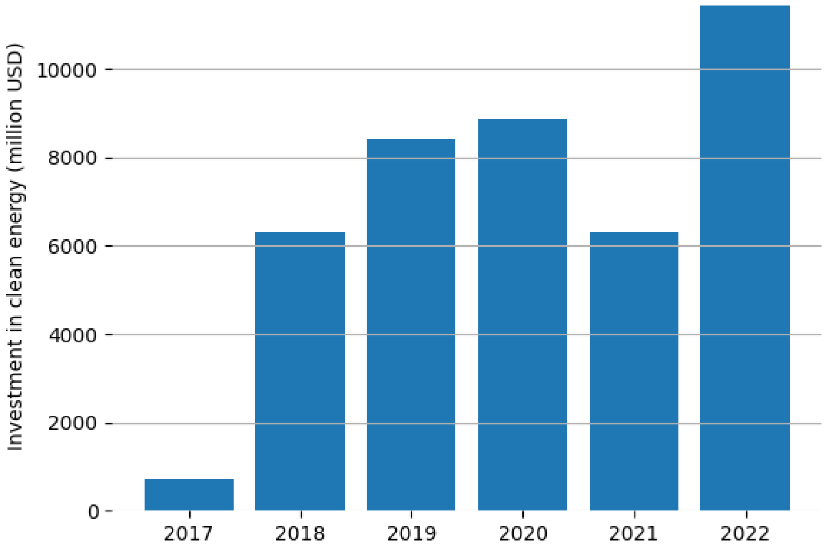

Figure 1: Investment in the Spanish clean energy sector, in million USD. Fernández, 2024

This showcases the obvious interest there is in renewable energy in general in Spain. However, when talking about renewable energy it is very important to outline the different renewable energy resources and their corresponding technologies Ellabban et al., 2014:

- **Biomass**: Biomass energy is derived from organic materials, mainly plant and animal matter such as wood, agricultural crops, waste, and algae. It can be burned directly to produce heat and power a thermal power plant or it can be converted into biofuels like bioethanol or biodiesel. These fuels are then used in thermal power plants or for other uses like vehicle powering. It is very versatile with many of the main benefits of fossil fuels and can be used to reduce waste. However, sustainable sourcing of the biomass is critical, as doing it in any other way can lead to deforestation or other environmental issues.

- **Geothermal**: Geothermal energy harnesses the higher and more constant temperatures found deep below the Earth's surface, which originate from the planet's hot core due to the high pressure and the radioactive decay of different materials. Geothermal plants directly source hot water from underwater sources or pump fluids deep into the surface, heating it and retrieving it afterwards. These hot fluids can be used in a thermal power plant, as part of the cycle to preheat the working fluid or directly to heat up some buildings. It provides a constant and reliable energy supply not dependent on weather conditions and it has a small land footprint. However, it is very location-specific, working best in regions with significant tectonic activity and with easily drillable materials in the ground.

- **Hydropower**: Hydroelectric power is the power of flowing water. This water is generally stored in a dam and released to flow through turbines which generate electricity, although other systems which do not require dams and use water flowing in rivers are also possible. Worth mentioning are also pumped storage plants, which although are not strictly a renewable energy resource, they can be used as energy storage systems. Hydroelectric power has high efficiency and can provide electricity on demand with very short warm up periods, however the initial investment to build the dam is very costly, it cannot be placed anywhere and can disrupt the local ecosystems.

- **Marine**: Marine power leverages different sources of energy available in the seas and oceans. There are six main energy sources within this category:

  - Wave: It transforms the kinetic energy of waves in their up and down movements into electrical energy.

  - Tidal range: It leverages the difference in height between the high and low tide to generate electricity.

  - Tidal currents: It harnesses the horizontal currents caused by the rising and falling of the tide.

  - Ocean currents: Similar to the previous source, however it leverages currents not necessarily caused by tides, but caused by the dynamics of the ocean.

  - OTEC: Ocean thermal energy conversion exploits the temperature difference between the warmer surface water and the colder deep water as the hot and cold sources of a thermal cycle to generate electricity.

  - Salinity gradients: It leverages the difference in saline concentration between different areas of the ocean.

  Many of these energy sources are still undergoing intensive research and development and are still in the prototype and demonstration stage.

- **Solar**: Solar energy leverages the energy of solar radiation reaching us from the sun in several ways:

  - Photovoltaic: Solar PV systems directly convert the energy of the photons sent by the sun into direct current through the photoelectric effect Einstein, 1905. The PV cells, generally made out of silicon, have an efficiency of around 20%. However, they are very modular and can be produced at scale, helping achieve the tremendous drop in cost that has been seen during the last decade.
  - Thermal: Solar thermal power leverages the heating capacity of the solar radiation to harness its power. Although it is often used to directly heat up buildings, it can also be used to generate electricity. Concentrating solar power (CSP) produces electricity by concentrating the solar irradiance through mirrors and lenses to heat a liquid which is used in a thermal power cycle.

- **Wind**: Wind energy refers to the kinetic energy of wind currents. This energy can be transformed through wind turbines into electrical energy. Wind power is generally obtained through two different types of setups. There are on-shore and offshore power plants, with the former being installed in land and the latter in seas or oceans near the coast. Offshore installations have some advantages, like more powerful and constant wind, no land erosion and less landscape visual degradation, however its installation is more costly.

As it can be seen, there are many renewable energy resources and technologies that can be used to harness them and aid in the energy transition. However, not all of them are equivalent, with different technologies having different levels of maturity, profitability, capital requirements, etc. That is why generally the renewable energy mix is not very uniform.

In Figure 2 it can be seen how solar, wind and hydro energy covers more than 95% of the energy generation of 2023 – as percentage of the total generation of 134.321 GWh of renewable energy – while other sources like geothermal and biomass are almost negligible.

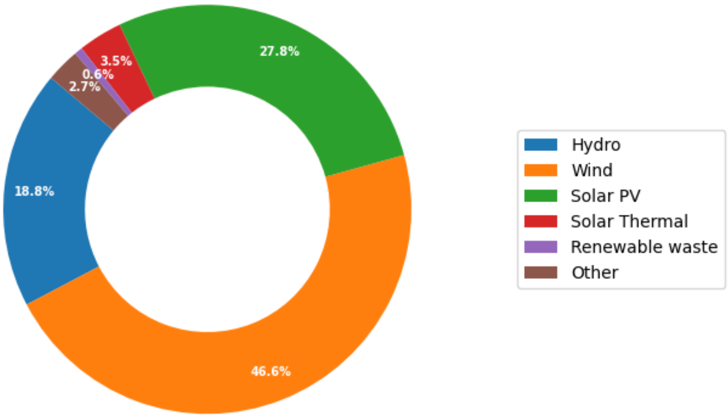

Figure 2: Generated renewable electricity mix in Spain in 2023. *Renewable electricity generation*, 2024

1.2 Motivation

In the previous section in Figure 2 a promising picture of the renewable energy generation in Spain could be seen, even more when the total generation is compared with the non-renewable generation of 132.486 GWh in Spain (*Renewable electricity generation*, 2024), showing how renewable energy generation surpassed the 50% mark.

However, the generation mix shown is also a source of concern. A 77.9% of the total renewable energy mix was produced through solar and wind energy. Solar and wind energy share some characteristics which lead to some grave challenges to the overall electrical system. Mainly, these characteristics and consequent challenges are:

- **Variability**: These energy sources are intermittent. That is, one cannot choose to turn on a solar power plant at any moment, as its output depends on the solar input, which cannot be decided by the operator. This is translated into a very variable pattern of generation with high peaks and low troughs.

  This variable pattern of production with rapid ramps leads to the need of having other power plants with also rapid ramp ups which can step up after a sudden fall of solar or wind production. These other plants also need to be very flexible to adapt to the wide changes in generation from these renewable resources.

- **Uncertainty**: Not only can these technologies not be discretionary dispatched, but often it is difficult to know in advance what their behavior will be at some point in the future. The further out into the future one looks, the less certain the sun and wind forecasts become, and thus the less certain the production output forecasts become.

  Many of the problems of variability are accentuated by the fact that this variability is uncertain. Given the grid players are uncertain about the renewable generation, large reserves need to be allocated in case the prediction fails. This in turn leads to more expensive electricity for the end consumer if the same level of reliability is required.

- **Location specific**: The renewable resources are often geographically concentrated. For example, sun is more prominent in the south of Spain and wind around the coast.

  Having many of these power plants geographically concentrated leads to several problems when all of them are producing at high outputs at the same time. Power lines connecting the renewable rich area with the rest can become overloaded forcing shutdowns and cutoffs. Furthermore, great voltage differences between areas in the grid can appear.

- **Generator technology**: The induction generators generally used in wind turbines or the inverters in solar power plants behave differently to the synchronous generators used in thermal power plants, which leads to problems in frequency control as will be outlined below.

  This difference in generator technology leads to several problems. The first one being that these power plants are unable of providing the primary reserve matching supply and demand at the expense of frequency changes that synchronous generators supply automatically. Furthermore, these generators do not provide the same reactive power supply as the synchronous generators, and in the case of induction generators in wind turbines they in fact consume reactive power in order to function. It has also been hypothesized how these generators can lead to angular instability Vittal and Ayyanar, 2013. The generators of wind turbines have also been shown to lead to problems of power quality due to the injection of different harmonics Muljadi et al., 2006.

- **Low capacity factor**: Due to the variability in the resources, the capacity factor of installations leveraging sun and wind is generally lower to that of other non-renewable technologies.

Even more importantly, the greater the share of overall energy that comes from solar and wind sources the more pronounced these challenges will become. For more details regarding these characteristics and their consequences, refer to Ahmed et al., 2020, Kumar et al., 2016 and Steen et al., 2014.

Grid operators need to consider these challenges at a short and medium term when operating the grid, and planners should also consider it when planning the long term of the grid.

The first two characteristics, the variability and uncertainty of these resources, leads in fact to a great problem in long term planning. When deciding where to locate which power plants and of what capacity, it is important to have accurate models of all resources. In fact, many of the optimization models used in long term planning require forecast models of these resources, which is very challenging due to the aforementioned characteristics. The lack of realistic and accurate models of solar and wind energy generation is precisely what this work aims to solve.

1.3 Objective and scope

By looking at the overview of the wind and solar energy presented in subsection 1.2 Motivation and understanding the objectives and challenges of grid planners – long term distribution of resources to ensure the grid's effectiveness and reliability – the need for a model that can be used by these planners has become apparent.

To be more precise, for such a model to be useful it must accurately represent the wind and solar energy generation marginal and joint distributions. The characteristics the model must fulfill are the following:

- **Modeled variable**: The variable to be modeled is the hourly capacity factor of wind, solar PV and solar thermal energy generation. The capacity factor f_t can be calculated as

  $$f_t = \frac{E_t}{P_t t} \tag{1}$$

  Where E_t is the energy generated in a given timeframe, P_t is the power output rating – that is the rated installed capacity – and t is the length of the timeframe. As it can be seen, the capacity factor is a unitless ratio.

  It has been chosen as the modeled variable due to having several beneficial characteristics. Numerically, $f_t \in [0, 1]$ which makes it easier to model. It also is able to show the power generation without regarding the installed capacity. The installed capacity is often determined by policy decisions, awarded licenses or discretionary investment decisions by electricity companies and is thus harder to accurately model. The capacity factor however allows for an accurate representation of the generated power.

The hourly frequency has been chosen as it is short enough to allow for significant changes throughout the day, but is not short enough for noise to become a significant part of the variable. Furthermore, this is the frequency used by most market players, with the electrical market quoting hourly values for example.

The wind, solar PV and solar thermal capacity factors have been chosen as independent variables because the three of them have significantly different driving factors, making them have different dynamics that can be captured by individual models. Even the solar PV and solar thermal have different dynamics, thanks to the inertia provided by the storage of heat in solar thermal power.

- **Input variables**: The only exogenous variable that will be used in all models are the historical values of these capacity factors and temporal variables such as the month or hour. A purely time series approach has been taken, as opposed to other physical approaches that leverage weather forecasts. The intuition for this decision is that the information provided by the weather or any other significant variable should be integrated in the capacity factor, and relying on other external variables forces a two step approach where a model for those variables needs to be created and then a model for the capacity factor needs to be created, as there are no accurate forecasts for any of the other possible exogenous variables with a long enough timeframe.

- **Geographical scope**: The capacity factor will be that of the national Spanish power system.

- **Temporal scope**: The model should be able to forecast the capacity factor for a period of 1 to 10 years into the future, with hourly values.

- **Methodological scope**: Statistical, machine learning and deep learning models will be studied as candidates to modeling this variable.

2 Literature Review

Given the great and increasing importance of renewable energy, it is of no surprise that many scholars have already shown great interest in all aspects surrounding it. From the development of the technology, to the analysis of its use, there is an incredible amount of papers regarding the topic. Given this astonishing breadth, this literature review will focus strictly on modeling and forecasting papers.

Regarding the data used by the model, two main types of models have been observed. There are models which predict power data based on physical data such as wind speed, temperature, rainfall, humidity, etc. for solar power such as in G. C. Wang et al., 2019 with other models even using images obtained from satellites like in Shen et al., 2018. On the other hand, there are models which try to predict the target variable's values from historical data or non-physical exogenous variables. These will be the main focus of the literature review given how this is the data available for the present work. Finally, there are also some hybrid models leveraging both.

There is also great diversity with respect to the predicted variable. With respect to wind, some works focus on predicting wind speed as in He et al., 2020 while others on wind power as in L. Wang et al., 2020. For the solar resource, predicted variables include solar irradiance as in Alzahrani et al., 2017 also expressed as solar irradiation as in Sorkun et al., 2017, solar power as in Suresh et al., 2020 or more specifically solar PV power as in Sharadga et al., 2020 or solar thermal power as in J. Wang et al., 2020.

Starting with the data preprocessing methods, most papers use Min-Max scaling like in Ju et al., 2019 where it is used in the forecasting of solar PV to normalize all strictly positive data, while Zero-Mean normalization is used on data with positive and negative values. Regarding wrong or missing value handling most papers report removing such records, although some papers such as Peng et al., 2020 use linear imputation. Regarding the removal of outliers, some papers such as Sharadga et al., 2020 use the Hampel filter while others like Zang et al., 2020 interpret as outliers any values exceeding three times the standard deviation and impute it with a quadratic spline. Some papers also use data augmentation for the model to be able to focus on specific weather conditions, like F. Wang et al., 2018 where 10 categories of weather are uncovered and a Generative Adversarial Network (GAN) is used to generate more data for some of these categories to solve the data imbalance present in the dataset. In some other cases some additional features are calculated and included in the dataset, like in Yin et al., 2019 where the sine and cosine of the wind direction is calculated and included. Other techniques found in the literature are the clustering of the data to divide it into seasons or weather conditions like in Alzahrani et al., 2017 or to assign labels such as in F. Wang et al., 2020.

Regarding the prediction time horizon there is also great diversity in the published literature. Some papers like Hu et al., 2015 or J. Chen et al., 2019 focus on short term forecast horizons on the order of 10 minutes to a few hours. Others like Aly, 2020 focus on a medium horizon of 1 week for its wind speed and power prediction. There are few where the horizon is considered long term like Ray et al., 2020, which predicts PV power with a multi-year horizon. However, this paper like others found with a long term horizon use a monthly resolution, so the forecast horizon, although long in the temporal sense, is not that many steps ahead.

As for the type of model used, it is quite astonishing the diversity of models that have been applied to the task of forecasting solar and wind power or related variables. Within the statistical methods' category, some authors like Pedro and Coimbra, 2012 implement an ARIMA model, Bouzerdoum et al., 2013 extends it to a SARIMA model combined with an SVM or Haddad et al., 2019 implements a vanilla SARIMA model, Jain et al., 2022 implements a VARMAX used for wind power forecasting by forecasting it together with other variables determined to be causal through a Granger test. As additional less conventional statistical methods, Delgado et al., 2024 leverages a Hidden Markov Model (HMM) combined with an LSTM for PV power generation forecasting, Frimane et al., 2022 uses an infinite HMM for short term solar irradiance forecasting and L. Wang et al., 2022 uses a Markov switching GARCH-MIDAS model for forecasting renewable stock volatility in short and long term horizons. Regarding Machine Learning based models, Abuella and Chowdhury, 2016 uses Support Vector Regression (SVR) for a one-year ahead hourly granularity forecast of solar power, others like Aksoy and Genc, 2023 implements three different gradient boosting models for solar power plant generation forecasting. However, the most explored type of models is without a doubt the Deep Learning or Neural Network based models. The models that have been implemented for solar and wind generation start at plain Feed Forward Neural Networks (FFNN) as in Pedro and Coimbra, 2012. Convolutional Neural Networks which are usually used for 2 or 3 dimensional data can be adapted for 1 dimension and used for solar PV power forecasting as in Huang and Kuo, 2019. Recurrent Neural Networks (RNN) are specially well suited for these types of forecasting tasks given their ability to retain temporal relationships. Within the family of RNN, Long Short Term Memory networks (LSTM) are implemented by itself as in Abdel-Nasser and Mahmoud, 2017 or Hossain and Mahmood, 2020, they are stacked as in Liang et al., 2018 or implemented in an autoencoder architecture as in Suresh et al., 2020. Other RNN architectures like Gated Recurrent Unit (GRU) networks are also implemented in Sorkun et al., 2017 for solar irradiation forecasting. Deep Belief Networks (DBN) in which hidden layers have connection between layers but not between units in a given layer are also used for forecasting wind speed in Lin et al., 2019 or for solar PV power in Neo et al., 2017. More recently, given the huge popularity of Transformer models for speech processing, some papers exploring their use for renewable energy forecasting have appeared, like Kim et al., 2024 which uses a transformer architecture for solar PV power forecasting. A special approach used by some regarding deep learning is that of transfer learning, where a pretrained model is employed on a different target domain like Sarmas et al., 2022 using TL for solar power forecasting. Another big family of models that is widely explored in the literature is that of ensemble models. These models combine two or more basic models through some weighting scheme, stacking architecture, voting model or more sophisticated combination method. An example of a paper using such models is the one implemented by H. Liu et al., 2019 which combines a CNN, GRU network and SVR for wind speed forecasting.

Regarding evaluation metrics, there is also some diversity in the kind of metrics that are used. The work by Mystakidis et al., 2024 gives a good overview on some widely used metrics, mentioning the coefficient of determination (R^2), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Root Mean Squared Error (RMSE), Coefficient of Variation of the Root Mean Squared Error (CVRMSE) and Normalized Root Mean Squared Error (NRMSE).

3 Data Description

3.1 Data Sources

The datasets needed for the exploration of the time series and the fitting of the models have been obtained from the ESIOS – System Operator's Electronic Information System. This is an online portal where the System Operator, the REE – Spanish Electrical Grid – uploads all relevant information for the operation of the grid.

In order to accomplish everything outlined in subsection 1.3 six different datasets are needed. These datasets are:

- Real time wind energy hourly generation
- Real time solar PV energy hourly generation
- Real time solar thermal energy hourly generation
- Wind energy monthly installed capacity
- Solar PV energy monthly installed capacity
- Solar thermal energy monthly installed capacity

3.2 Data Preprocessing

The obtained datasets are already very clean, so no missing or obviously wrong values were found on any of them. The installed capacity is obtained on a monthly basis as there is not any lower granularity in the ESIOS portal. With these datasets, the capacity factor is obtained as explained in equation (1) by dividing the generated electricity by the installed capacity times the period time frame. The time frame in this case is always one hour. Like this the hourly capacity factor for the Spanish electrical system for the three different technologies is obtained. Note that the data obtained spans from mid 2015 until end of 2023, since no previous data is available.

For the models which need it, that is the models where error terms are Gaussian or which require some form of normalization, an additional Box-Cox transformation has been added to the data, in order to obtain these Gaussian errors and normalization. This will be explained more in depth in the Methodology section.

3.3 Exploratory data analysis

Now that it is well understood where the data comes from, an initial analysis of the data to understand what the capacity factor of each looks like will be performed.

3.3.1 Initial visualization

The first variable to be visualized will be that of the solar PV capacity factor, which can be seen in

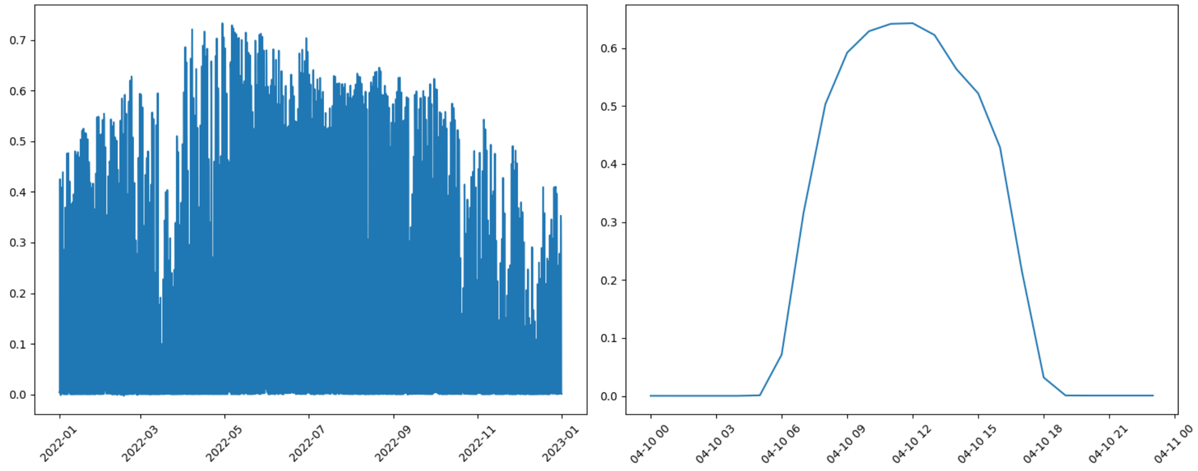

Figure 3: Yearly and daily profiles of the solar PV capacity factor, taken for 2022 and the 10th April 2022 respectively.

The main features of the series can be well visualized here. The yearly seasonality is very clear, with higher peak values in summer and lower peak values in winter, although this is altered by periods of lower sunlight where peaks are reduced below the typical seasonal peak, like at the beginning of March. The daily pattern is also very clear, with values of practically zero – it is not exactly zero due to some background light and noise in the measurement – at night and a generation in accordance with the position of the sun in its ecliptic.

The solar thermal profile, although belonging to the same renewable resource as the solar PV, is slightly different.

Several differences between these profiles shown in Figure 4 and the previous ones are apparent. The yearly pattern, even though it follows roughly the same seasonality with higher

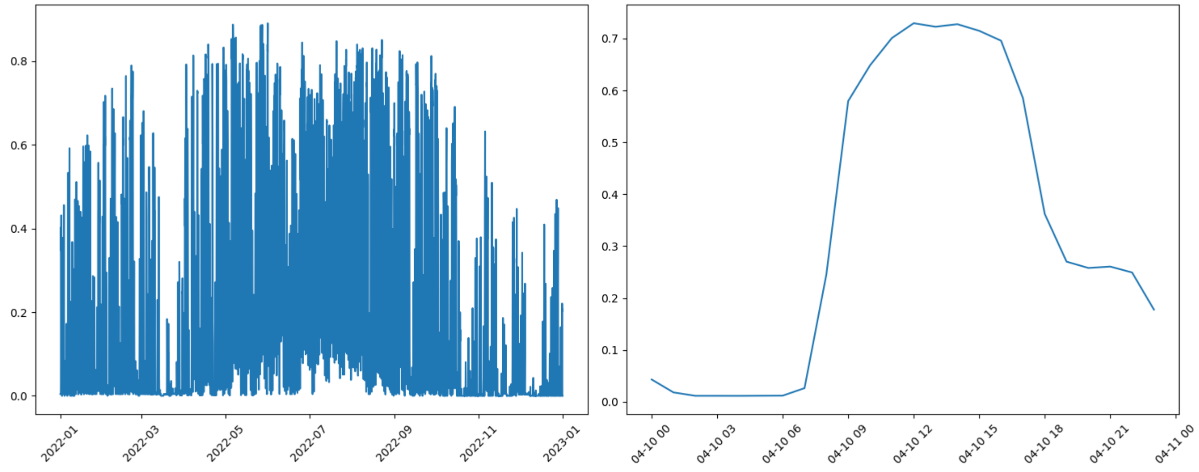

Figure 4: Yearly and daily profiles of the solar thermal capacity factor, taken for 2022 and the 10th April 2022 respectively.

peaks in summer is much more volatile than that of the solar PV. Furthermore, there are whole periods of several days when generation does not go to zero, due to the inertia of the thermal system. The highest peaks are also higher for the solar thermal, reaching generation values much closer to its rated capacity in the summer than the solar PV. In the daily pattern the mentioned inertia can be clearly seen, where apart from the general sinusoidal throughout the day it can be seen how generation remains above zero at night.

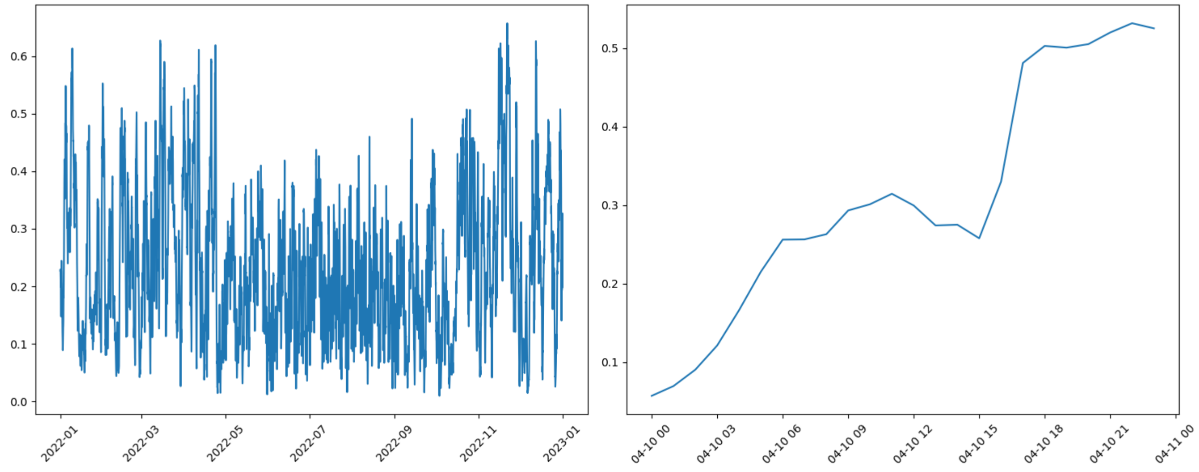

Figure 5: Yearly and daily profiles of the wind capacity factor, taken for 2022 and the 10th April 2022 respectively.

The wind profile shown in Figure 5 is the most different of the three. The yearly profile is the most volatile of the three, with less of a clear seasonal pattern, although it can be seen how the mean values in summer seem to be lower than those in the winter. In the daily profile there are not any obvious patterns which can be seen at first view.

3.3.2 Individual statistical characteristics

After an initial view of the general time series for the three variables, a deeper look at the main statistical properties of these individual series is pertinent. For this deeper look a KDE and cumulative KDE will be shown and the main moments will be calculated.

The Kernel Density Estimate (KDE) is a non-parametric way of estimating the probability density function (PDF) of a random variable. It works by placing a kernel – that is a smooth, symmetric function – at each data point and then summing these kernels to produce a smooth estimate of the PDF. In more precise terms, the estimation of the PDF at x denoted by $\hat{f}\left(x\right)$ is calculated as

$$\hat{f}\left(x\right) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{2}$$

where n is the number of data points, $K\left(\cdot\right)$ is the kernel function, h is the bandwidth – a parameter used to approximate the width of the kernel and thus how smooth or sensitive to individual points the estimation is – and x_i are the individual data points.

The chosen kernel function has been the Gaussian kernel, which is calculated as

$$K\left(x\right) = \frac{1}{\sqrt{2\pi}} \exp -\frac{1}{2}x^2 \tag{3}$$

The method selected for estimating the optimal bandwidth has been the method outlined in Scott, 1979.

The cumulative KDE is an analogous method used for calculating the cumulative density function (CDF) instead of the PDF. It is calculated as

$$\hat{F}\left(x\right) = \frac{1}{n}\sum_{i=1}^{n} \int_{-\infty}^{x} \frac{1}{h} K\left(\frac{t - x_i}{h}\right) dt \tag{4}$$

The KDE and cumulative KDE of all three variables can be seen in Figure 6

Several observations can be made regarding these plots. First, the high density of zero like values in both solar plots is clear, with an even higher density in the solar PV. In fact, it can be seen how both KDE have non-zero values for x values below zero, however this only happens due to the symmetry and width of the Gaussian kernel, which for the points at zero sums some value for points below zero. Apart from the mode at zero, both plots have another peak near their maximum value, at around 0.6 and 0.9 for the solar PV and solar thermal respectively.

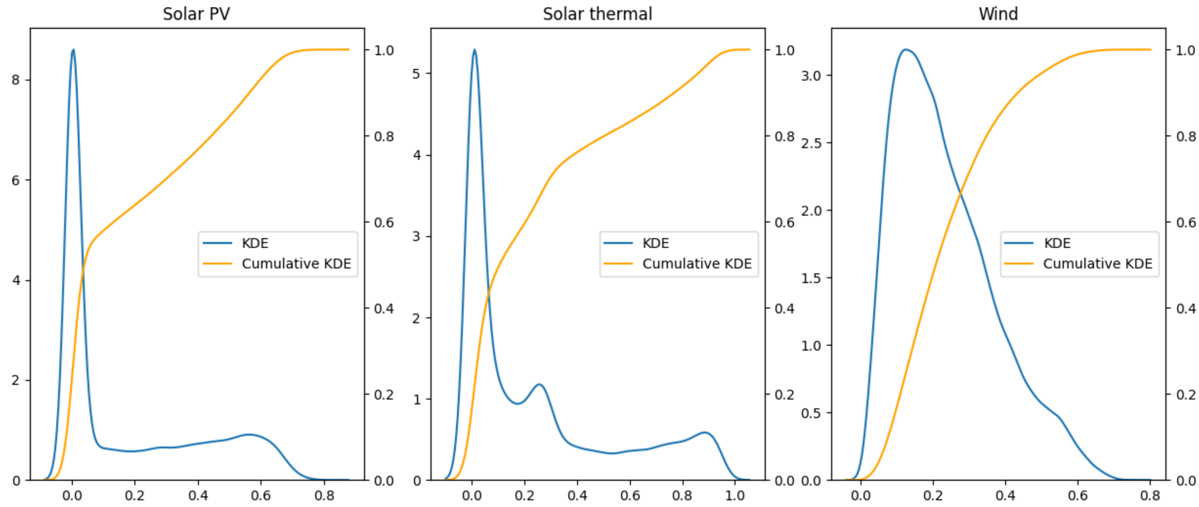

Figure 6: KDE and cumulative KDE of the three capacity factors

This is due to the shape of the solar ecliptic, where the sun remains at its highest point for longer than it is at any given point during ascent or descent. The solar thermal PDF also has a significant peak at around 0.3, probably due to this being the power at which the power plants run with the stored heat at night. This is in fact the point where the daily curve in Figure 4 can be seen operating at night. As for the wind PDF and CDF, it is a much smoother plot, with the mode around 0.1 and then slowly and gradually tapers of.

In order to have a more quantitative view of these distributions, metrics relating to the first four moments are shown in Table 1

Metric	Value		
	Solar PV	Solar Thermal	Wind
Mean	0.18	0.24	0.23
Standard Deviation	0.23	0.29	0.14
Skew	0.88	1.12	0.78
Kurtosis	-0.79	-0.08	0.01

Table 1: Main moments of the three capacity factors

Through these measures it can be seen how some intuitions obtained when looking at the time series plot of the different capacity factors were wrong. For example, the wind capacity factor is not the most volatile as it seemed, but in fact has the lowest standard deviation. This could be however due to the high variability of the solar profile, and taking away the seasonality of the three variables these metrics could be different.

3.3.3 Lagged relationships

Another key factor about the three series is their seasonality and how a given value at t is influenced by past values at $t-1, t-2...$. In order to explore these relationships, two different metrics will be used.

The first one is the autocorrelation. While correlation as expressed by the Pearson correlation coefficient is generally calculated like

$$\rho(X,Y) = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{\sum_{i=1}^{n}\left(x_i-\bar{x}\right)\left(y_i-\bar{y}\right)}{\sqrt{\sum_{i=1}^{n}\left(x_i-\bar{x}\right)^2}\sqrt{\sum_{i=1}^{n}\left(y_i-\bar{y}\right)^2}} \tag{5}$$

The autocorrelation function of a series quantifies the correlation of a series with lagged values of itself at different lags. For a given lag k it is thus calculated like

$$ACF\left(k\right) = \frac{\sum_{t=1}^{N-k}\left(x_t-\bar{x}\right)\left(x_{t+k}-\bar{x}\right)}{\sum_{t=1}^{N}\left(x_t-\bar{x}\right)^2} \tag{6}$$

The correlation coefficient has some known drawbacks however. It only measures linear relationships, so any dependence apart from this very simple relationship cannot be captured through a high correlation coefficient. That is what the new correlation coefficient proposed by Chatterjee, 2020 aims to solve, and why it has been included here. This new correlation coefficient, capable of capturing non-linear relationships and with some very interesting statistical properties is calculated as

$$\xi(X,Y) = 1 - \frac{3\sum_{i=1}^{n-1}\left|r_{i+1}-r_i\right|}{n^2-1} \tag{7}$$

Where the given pairs $\left(x_1,y_1\right),\left(x_2,y_2\right)...\left(x_n,y_n\right)$ have been arranged so that $x_1 \leq ... \leq x_n$. Here r_i denotes the rank of y_i. That is, the number of j such that $y_j \leq y_i$. If there are ties among the x values, the formula is slightly different, however with the capacity factor being a continuous variable it can be assumed that no ties will be present. Finally, note how even though $\rho \in [-1,1]$ the new correlation coefficient makes no distinction between positive and negative relationships and $\xi \in [0,1]$.

With this new correlation coefficient, a new autocorrelation function is constructed in the same way as in (6), calculating the correlation of the series with its lagged values for each lag. The following figure shows both autocorrelation functions for the three series.

Several observations can be made from the two autocorrelation functions in Figure 7. The

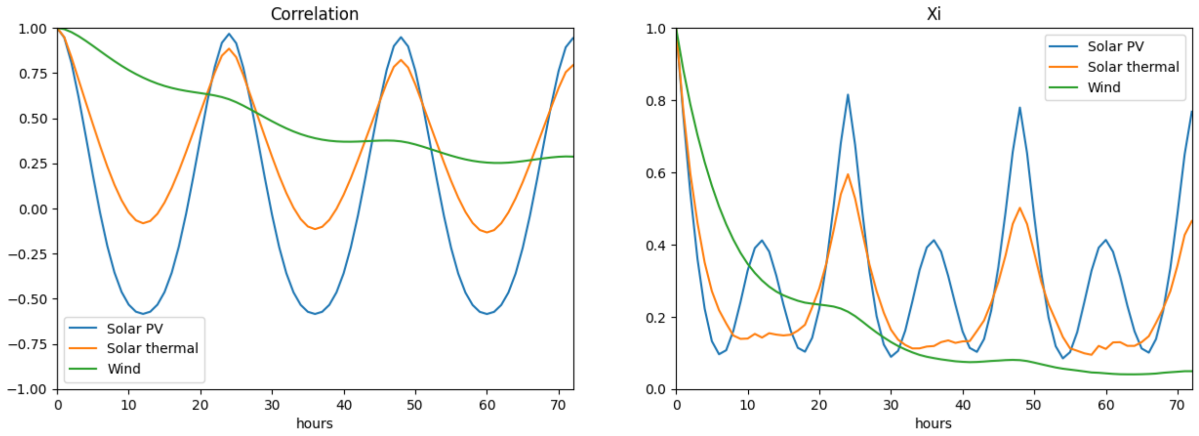

Figure 7: Autocorrelation functions with the two correlation coefficients for the three capacity factors at different hourly lags

seasonality in both solar series is again made apparent, with high autocorrelations at the 24, 48 and 72 hour mark showcasing the importance of the previous days' values at the same hour for a given day, but also at the 12, 36 and 60 hour mark. This second autocorrelation, negative in the case of Pearson correlation, denotes the relationship between opposite moments of the day between different days. It is also noted how the correlation in the case of the solar PV is much more persistent, with the correlation peaks remaining more or less constant for the three days, while for the solar thermal this correlation tapers off a little faster. It is also interesting to see how the correlation at the 12, 36 and 60 hour mark is a lot lower for the solar thermal, showcasing how this relationship between values at a lag of 12 hours practically – although not completely – disappears, again due to the inertia of the thermal system which makes the solar thermal cycle less related to the sun's trajectory. As for the wind autocorrelation functions, it is much more linear and smooth, with the relevance of past values reducing gradually as the hours go by. There is a slight daily seasonality as well with the autocorrelation at daily lags being slightly above what could be expected from the linear trend. However, this has a practically negligible relevance compared with the solar series. In fact, for the wind capacity factor it can be seen how the immediately previous values have much higher significance, while previous lags at the daily or semidaily periods hold much less significance.

3.3.4 Seasonal statistical characteristics

Even though the characteristics shown in previous sections already provides very significant information, it seems like many of these characteristics may be seasonally dependent. That is, they may not be the same in summer and in winter, or at night and during the day. That is why the previous analyses will be performed again but seasonally separating the datasets. Both daily and yearly splits will be done, where ”day” is the period from 06:00 to 18:00 UTC and ”night” is the period from 18:00 to 06:00 UTC. On the yearly split and taking into account the day’s number within a year, spring goes from the 3rd to the 5th month, summer from the 6th to the 8th, autumn from the 9th to the 11th and winter from the 12th to the 2nd, both inclusive in all cases. Note how these are meteorological seasons, and not astronomical seasons based on the solstices and equinoxes.

The yearly seasonal breakdown can be seen in Figure 8.

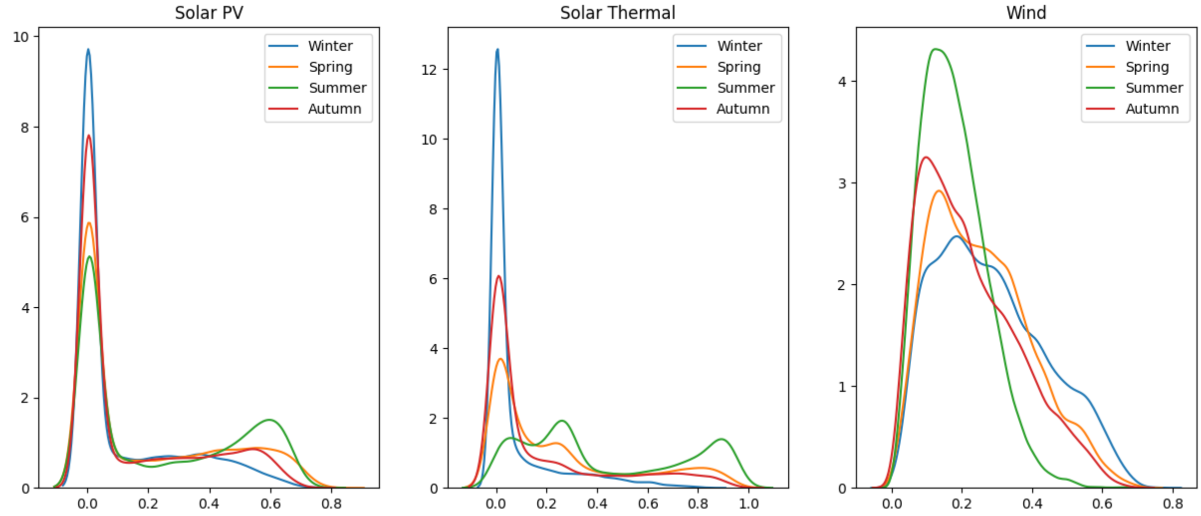

Figure 8: KDE plot for the three capacity factors for the different yearly seasons

For the solar PV it can be seen how on one extreme in winter the concentration of zeroes is highest, with very low density in the previously seen peak at 0.6, while in the summer the concentration of zeroes is still high – due to the night hours – but much less, and the peak at 0.6 can be clearly seen. Spring and autumn are found in between these values, with spring having a density of zeroes slightly above summer but a much lower peak at 0.6. For solar thermal the difference in density at zero is much higher, with winter having higher density and the rest of the seasons having a much lower one. In fact, the summer density at zero is very low, much more than other peaks. The peak at 0.9 can again only be seen in the summer, and the peak at 0.3 can be seen mainly for summer but also slightly for spring. As for the wind distribution, summer seems to be the one with the lowest volatility and lowest mean, while winter has the widest PDF and apparently mean.

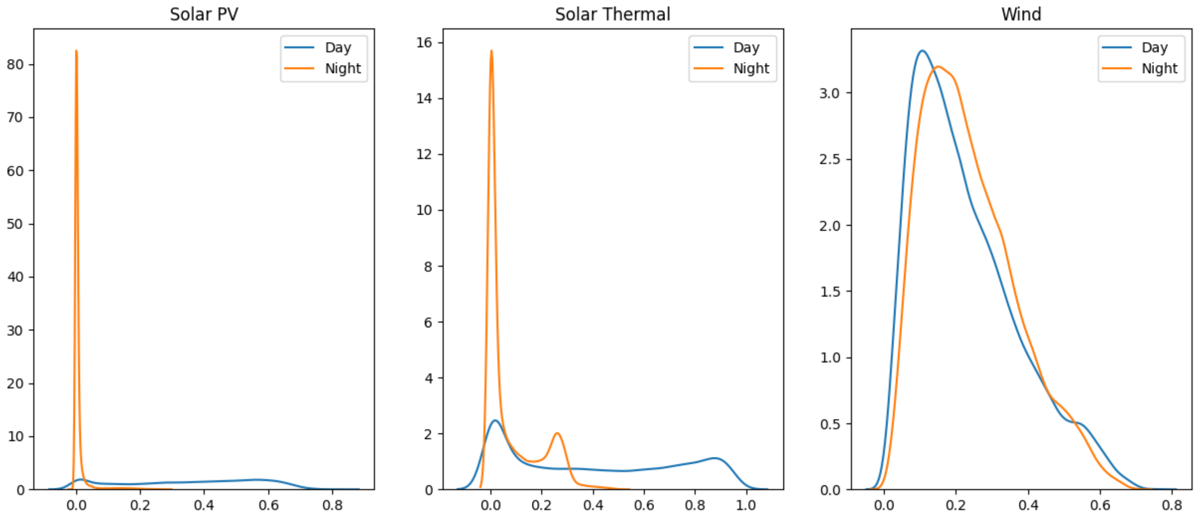

Figure 9: KDE plot for the three capacity factors for the different daily periods

In Figure 9 a similar representation for the KDE can be seen but for the split between night and day. For solar PV the plot is not very interesting, with the night having all its density at the zero point, with the day density having a slight peak at zero – as the exact times for sunset and sunrise vary during the year and the period defined here as ”day” can in fact have some nighttime – and the already known one at 0.6. For solar thermal it can be seen how the peak at 0.9 is exclusively a daytime phenomenon while the peak at 0.3 is a nighttime phenomenon, confirming the hypothesis that this was the power generated at night with the stored heat from the day. Regarding the wind, it can be seen how the difference between day and night is practically negligible, with day exhibiting a slightly more ”summer” like behavior – that is, slightly less volatile and lower mean.

3.3.5 Multivariate analysis

Up until this point every analysis has been performed on a per variable basis, examining their individual properties. However, when thinking about the true implications for the power grid it becomes clear that the joint characteristics are also fundamental. For a given wind and solar distributions, the consequences of the grid are not the same if peaks in both happen at the same time or if they complement each other. That is why the joint characteristics will be studied, and they will mainly be studied through cross correlation and through copula analysis. Some other analysis, like through Granger causality as outlined in Granger, 1969 were attempted, but no new relevant information that wasn't uncovered through cross correlation was found.

Cross correlation follows a principle very similar to the autocorrelation, but instead of measuring the correlation of a series with lagged versions of itself, the correlation of a variable with lagged versions of a different variable are measured. That way, something similar to the predictive value of the second variable on the first can be understood. Again, the two different correlation coefficients already explained were used. Note how for Person's correlation the relationship for (X, Y) is the same as for (Y, X). However, when using lagged values the correlation does depend on which variable is lagged. For the new correlation coefficient the directionality matters, as the effect of one variable on a second is not the same as that of the second on the first one.

In Figure 10 the lagged cross correlation between each pair of variables can be seen. For each of the three variables, it is shown how the lagged versions of the other two are correlated to the given variable, for each of the two correlation coefficients.

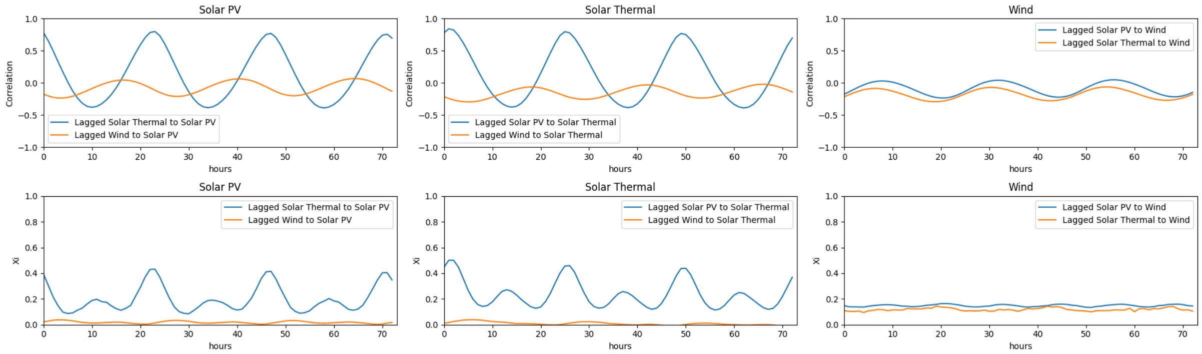

Figure 10: Lagged cross correlation between each pair of capacity factors

The similarity between the solar PV and solar thermal is clearly shown, with high correlations in both directions. Interestingly, the solar thermal capacity factor is more highly correlated with the 1-lag solar PV than with the simultaneous solar PV, probably due to the time it takes for the working fluid to be heated up, while the PV cells are able to convert that sunlight into power instantaneously. The second notable observation is that the predictive power of solar on wind seems to be higher than that of wind on any solar when looking at the Xi. Also, that correlation of the two lagged solar variables does not seem to decay for the examined 3-day period, and a very slight lag effect is seen where the daily seasonality of the sunlight is seen in the correlation, with wind being negatively correlated to simultaneous solar power but positively correlated to solar power at a lag of around 8 hours.

The second method used to see how the three variables compare to each other will be using copulas. Copulas are multivariate functions used to model the dependence between several random variables. Consider the random vector (X_1, X_2, X_3) representing the observations of capacity factor for solar PV, solar thermal and wind energy for a given point in time. The marginal CDF of each of the variables is represented as $F_i(x) = P(X_i \leq x)$, which when applied to the random vector yields the marginals $(U_1, U_2, U_3) = (F_1(X_1), F_2(X_2), F_3(X_3)) \in [0,1]^3$. The copula $C(\cdot)$ is the joint cumulative distribution function such that

$$C(u_1, u_2, u_3) = P(U_1 \leq u_1, U_2 \leq u_2, U_3 \leq u_3) \tag{8}$$

The copula contains all information on the dependence structure between the three series, while the marginal cumulative distribution functions contain all information on the marginal distribution of the variables.

In this analysis, the marginal CDF used has been the Gaussian cumulative KDE outlined in (4). The copula used has been a Gaussian copula, which relies on the multivariate normal distribution.

For the fitting process, a cumulative Gaussian KDE is fitted for each of the variables and then the univariate values are transformed according to that CDF. Then, the normal percent point function (PPF) – the inverse of the CDF – is applied to transformed variable to normalize it. Then, a multivariate Gaussian is fitted on the normalized data.

The correlation matrix of the Gaussian copula can be seen in Table 2.

	Solar PV	Solar thermal	Wind
Solar PV	1.00	0.72	-0.19
Solar thermal	0.72	1.00	-0.23
Wind	-0.19	-0.23	1.00

Table 2: Correlation matrix of the Gaussian copula with Gaussian KDE marginals

It can be seen how in the copula representation the strong positive correlation between both solar series and the negative correlation between wind and both solar series still stands.

4 Methodology

In this section several points regarding the methodology used in the work will be presented. The general approach taken to modeling, as well as the models themselves will be explained. The metrics used to evaluate the goodness of fit of the different models will also be presented.

4.1 Overview of modeling approach

In order to understand why the models that will be presented below have been chosen and why the evaluation metrics have been chosen, it is worth first taking a look at the requirements of the modeling task and the general criteria that determine what a good result is.

4.1.1 Modeling task characteristics

The general technical characteristics of the task, from the objectives and the time series characteristics already explored are the following:

- **Modeling horizon**: The horizon to be modeled is of between 1 and 10 years, or between 8.760 and 87.600 hourly time steps.

- **Available data**: The data available for the training and testing tasks consists of slightly below 10 years of data, or slightly below 87.600 observations. This data has been consistently recorded at a 1-hour time step interval.

- **Seasonality**: There has been found to be high seasonality, both at the daily and yearly frequencies.

- **Exogenous variables**: A certain predictive power of some series with respect to the others has been found. Thus, the capacity of incorporating the use of exogenous variables or multivariate modeling approaches are specially interesting.

- **Extreme values**: There is a special interest in having accurate modeling of the extreme values, specially in the wind capacity factor.

- **Point wise forecast**: The forecasts produced by the model can be point wise predictions, there is no need to produce probabilistic forecasts for each time step.

4.1.2 Modeling criteria

Starting with the objectives already outlined in the section called Objective and scope, here these objectives will be broken down into more specific criteria that the best model must aim to meet in the best possible way. The specific characteristics the model must meet are the following:

- **Seasonality**: The chosen models must be able of incorporating seasonality into its forecast.
- **Multiple variables**: The chosen models should be able to incorporate multiple variables, be it through a multivariate modeling, through historical exogenous variables and/or through future exogenous variables. Through the modeling task it will be made apparent whether incorporating the information of the other series helps in modeling a given series.
- **Complex relationships**: The temporal relationships of a series with itself and the relationships of a series with the other variables are probably more complex than linear relationships. That is why a model capable of incorporating complex non-linear temporal and intervariate relationships will probably have an advantage.
- **Multiperiod modeling**: As it has already been mentioned, the modeling period will encompass between 8.760 and 87.600 time steps. The model must be able to create such forecasts. But going further, if the prediction is created by autoregressively estimating the next step, there is a compounding in errors over time Selim et al., 2020. By estimating multiple time steps in one prediction pass, the number of autoregressive steps can be reduced and with it the compounding of errors can be reduced too. Therefore, models with bigger single-step prediction window should have an advantage.
- **Long and short term temporal relationships**: The model must be able to incorporate both long term and short term temporal relationships.
- **Robustness**: The model must be resistant to overfitting, given the parameter size of some models and the number of available training data points.
- **Extreme value accuracy**: The model must be specially sensitive to extreme values, be it through the flexibility of incorporating a custom loss function or through its intrinsic

characteristics.

- **Computational efficiency**: Given the long modeling horizon, the model must be efficient both in prediction time and in memory used when training and when incorporating historical data in prediction.

4.2 Models

Now that the criteria used to select potential models has been outlined, the different models that will be evaluated can be presented. These models have been divided by model family or type. Here below, the different model types and each of the models and their characteristics and main working principles are explained.

4.2.1 Statistical models

These models are based on classical statistical methods to analyze and predict temporal series. They use techniques like regression analysis, autocorrelation and spectral analysis to model and forecast temporal variables. The ones that will be used here are:

- **Linear Regression**: This is the simplest model and it will be used as a naive benchmark. It models a given capacity factor value for one of the series as a linear combination of several exogenous variables. In this case, those variables are the values of Fourier sine and cosine terms, some lagged values of the given time series and some lagged values of other time series, which in this case will be the other capacity factors. The main parameters of the model are thus:

  - Seasonal frequencies: That is the frequencies for which the Fourier terms will be calculated. It can be daily, yearly, both, etc.
  - Fourier harmonics: For each of the seasonalities, several harmonics of Fourier terms can be calculated, creating more nuanced patterns the more harmonics are added.
  - Autoregressive order: That is, what lags of the modeled series to incorporate.
  - Exogenous lags: That is, what lags of the other two series to incorporate.

The mathematical formulation for the model thus is:

$$Y_t = \beta_0 + \sum_{\tau}\sum_{k}\left[\gamma_{\tau,k}\sin\left(\frac{2\pi kt}{\tau}\right) + \delta_{\tau,k}\cos\left(\frac{2\pi kt}{\tau}\right)\right] + \\ + \sum_{l}\phi_l Y_{t-l} + \sum_{v}\sum_{p}\theta_{v,p}X_{v,t-p} + \epsilon_t \quad (9)$$

where τ is each of the seasonality periods, k is each of the fourier harmonics, l is each of the self lags, v is each of the other two capacity factors and p is each of the lags used for the two other capacity factors.

This model operates on several assumptions which are worth mentioning. The first is that all relationships – those with the seasonal terms, with lagged values, with the other capacity factors, etc. – are assumed to be linear. The error term ϵ_t is assumed to be independent and identically distributed (i.i.d.) with constant variance (homocedastic) and Gaussian, although there is no evidence that this is true for the three modeled series. That is why the Box-Cox transform is used before fitting the model.

- **ARIMAX**: The AutoRegressive Integrated Moving Average with eXogenous variables model is an extension to the previous model. It incorporates both autoregressive and moving average components to the endogenous time series and allows for time integration, which extends the purely autoregressive view in the previous model, as well as some exogenous variables. It provides a more complex representation of the relationship of the time series with its lagged values than the linear regression model presented above. To the previously presented parameter, these new ones are added:

  - Integration order: This represents the number of times the endogenous time series is differentiated.
  - Moving average order: That is, what moving average error terms are incorporated.

  The new model formulation is:

$$Y_t = -\left(\Delta^d Y_t - Y_t\right) + \beta_0 + \sum_{\tau}\sum_{k}\left[\gamma_{\tau,k}\sin\left(\frac{2\pi kt}{\tau}\right) + \delta_{\tau,k}\cos\left(\frac{2\pi kt}{\tau}\right)\right] + \\ + \sum_{l}\phi_l \Delta^d Y_{t-l} + \sum_{q}\theta_q\epsilon_{t-q} + \sum_{v}\sum_{p}\theta_{v,p}X_{v,t-p} + \epsilon_t \quad (10)$$

where d is the order of differentiation with $\Delta Y_t = Y_t - Y_{t-1}$ and q is lags of the error term added for the moving average.

- **VARMAX**: The Vector AutoRegressive Moving Average with eXogenous variables is a modification of the ARIMAX model, but without the integration component and turning the single variable endogenous variable into a vector variable. This model jointly predicts all capacity factor variables, directly creating the joint distribution. The model is formulated as:

$$Y_t = \beta_0 + \sum_{\tau} \sum_{k} \left[\gamma_{\tau,k} \sin\left(\frac{2\pi k t}{\tau}\right) + \delta_{\tau,k} \cos\left(\frac{2\pi k t}{\tau}\right) \right] + \\ + \sum_{l} \phi_l Y_{t-l} + \sum_{q} \theta_q \epsilon_{t-q} + \epsilon_t \quad (11)$$

where Y_t here is not a single variable data point but a three-dimensional vector containing all three capacity factors. The coefficients are also vector coefficients.

4.2.2 Machine Learning models

Machine learning models are a class of models designed to learn more complex patterns from data with less constraining assumptions and generally more parameters. Even if these models are quite general and are not specifically designed for time series modeling, they can be readily used for this purpose. For this specific use case, these models excel at capturing complex and non-linear relationships that statistical methods may overlook, they can therefore outperform the former methods when these characteristics are present or when data is highly non-stationary or with intricate dependencies. The ones that have been implemented are:

- **SVM**: Support Vector Machines introduced in Cortes and Vapnik, 1995 are ML models that can be used for either classification or regression. They work by finding the optimal hyperplane that separates the data points into distinct classes or predicts values with the maximum margin. In the specific case of regression the model is known as Support Vector Regression (SVR). Its objective is to find a function $f(x)$ that deviates from the empirical values y_i by at most a margin ϵ ensuring the model remains as flat as possible.

That is, preventing overfitting. The function takes the form:

$$f(x) = w^T \phi(x) + b \tag{12}$$

where w is the weight vector, b is the bias term and $\phi(x)$ is a transformation that maps the input data into a higher-dimensional space. The SVR algorithm solves the following optimization problem:

$$\begin{aligned} \min_{w,\xi_i,\xi_i^*} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \\ \text{s.t.} \quad & y_i - f(x_i) \leq \epsilon + \xi_i \\ & f(x_i) - y_i \leq \epsilon + \xi_i^* \\ & \xi_i, \xi_i^* \geq 0 \end{aligned} \tag{13}$$

where ξ_i and ξ_i^* are slack terms allowing for some flexibility in prediction errors and C is the regularization parameter that controls the trade off between the flatness of the function and how well it fits the data. Solving the optimization problem provides:

$$f(x) = \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) K(x_i, x) + b \tag{14}$$

where α_i and α_i^* are the optimal Lagrange multipliers corresponding to the slack variables ξ_i and ξ_i^* and $K(\cdot)$ is the Kernel function used to calculate the distance between predicted and empirical data points and which provides the needed non linearities to the function. In this case, the Radial Basis Function has been used with

$$K(x_i, x_j) = e^{-\gamma\|x_i - x_j\|^2} \tag{15}$$

where γ controls the spread of the kernel.

As it can be seen, the SVR is quite useful as it includes non-linear relationships, has built in robustness to overfitting and is quite scalable. However, it is quite sensitive to the choice of parameters – mainly that of kernel, C and ϵ – and assumes the relationship between input space and prediction remains constant in time, so it is not that appropriate for non-stationary time series.

In the implementation used in this work, the input space is equivalent to that of the classical statistical models, comprised of some lags of the data, some Fourier decompositions at different levels and some lags of the other time series.

- **XGBoost**: eXtreme Gradient Boosting introduced by T. Chen and Guestrin, 2016 is a highly efficient and scalable implementation within the paradigm of decision trees. Decision trees are another ML framework where starting from a root node, the dataset or output space is split into two or more subsets based on a specific value of a selected feature. For example, a very simple example would be to say that on average the predicted solar capacity factor is 0.3 (root node). If it is daytime the predicted value is 0.6 and if it is nighttime the predicted value is 0.0 (two subsequent nodes). This split can be further refined with more levels on different values of different features, providing a more accurate and non-linear prediction based on the input values. Trees are built through an algorithm which at each node selects the feature that best separates the data based on a criterion such as entropy. This model is quite simple but allows for non-linearity and is very robust to different scales in input features. However, it can easily overfit on the training data and the results can be quite unstable.

  Returning to XGBoost, this model builds sequential trees where each new tree corrects the errors of the previous one in what is known as an ensemble of trees. It also implements several improvements such as regularization, tree pruning or parallelization. The model thus builds a prediction like

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i) \tag{16}$$

  where $\hat{y}_i^{(t)}$ is the prediction after t trees and $f_t(\cdot)$ is the tree added at level t. Each individual tree is constructed optimizing an objective function which both minimizes the loss function between true and predicted value and also has a regularization term which prevents overfitting. The objective thus becomes

$$\min_{\theta} \sum_{i=1}^{n} L(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \tag{17}$$

  with the regularization term for each tree k being

$$\Omega(f_k) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{18}$$

with T being the number of nodes in the tree, w_j being the weight of leaf j and γ and λ being regularization parameters.

This model has some of the same advantages as the ability of capturing non-linear relationships and built in regularization, but it also allows for some features like custom loss functions.

In the implementation used for the XGboost, the input space is again equivalent to that of the classical statistical models with some lags of the data, some Fourier decompositions at different levels and some lags of the other time series.

4.2.3 Neural Network models

Neural networks are computational models inspired by the brain's morphology. They are formed by *neurons*, which are the basic unit which take several weighted inputs, sums them and adds some bias and applies an activation function which provides the non linearities in the model. Many of these neurons are connected across several layers, and end up providing a final prediction. For each layer, the mathematical expression can be represented as

$$A = \sigma\left(WX + B\right) \tag{19}$$

where A is the output of the layer, X is the input vector, resulting from the previous layer or the input itself, W is the weight vector, B is the bias and $\sigma(\cdot)$ is the activation function to be applied element wise to the resulting vector. This function can be a ReLU, tanh or sigmoid among others.

Neural networks have been shown to be universal approximators by Hornik et al., 1989, meaning that any arbitrary function can be perfectly approximated by a feedforward neural network with at least one hidden layer. In practice however there is hardly enough data to provide such an approximation and overfitting is quite easy due to the number of parameters. This has lead to the appearance of many NN architectures, trying to adapt them to specific use cases. These are the NN models that have been implemented for this work.

- **N-BEATS**: The Neural Basis Expansion Analysis for interpretable Time Series forecasting is another relatively new NN architecture focused on time series modeling through deep learning, mainly on the univariate case. Its architecture can be summarized through the following diagram obtained from the mentioned paper: Through Figure 11 it can be seen how the model works. There are several stacks, each created by blocks of fully connected feedforward networks. Each block predicts a part of the forecast with the information from the lookback period. The fully connected stack in the block predicts both the forecast and the backcast – data in the lookback period. The backcast is subtracted to the previously available lookback period, so each block tries to model the forecast from the residuals of the previous block in a very similar way to how each of the trees in the XGBoost predicts the error of the previous tree. The output of each of the blocks is summed to create the stack output, and all the stack's outputs is aggregated to create the final forecast. This architecture is supposed to automatically decompose the time series into different components like trend and seasonality in an interpretable way. This model has demonstrated state-of-the-art performance in several time series benchmarks such as the M3 and M4. It is also domain agnostic, making no assumptions on stationarity or seasonality and it works well without any feature engineering. The training is also more efficient than for an LSTM, as the lack of recurrent feedback loops allows for parallelization of the training. It also allows for a multistep forecasting window.

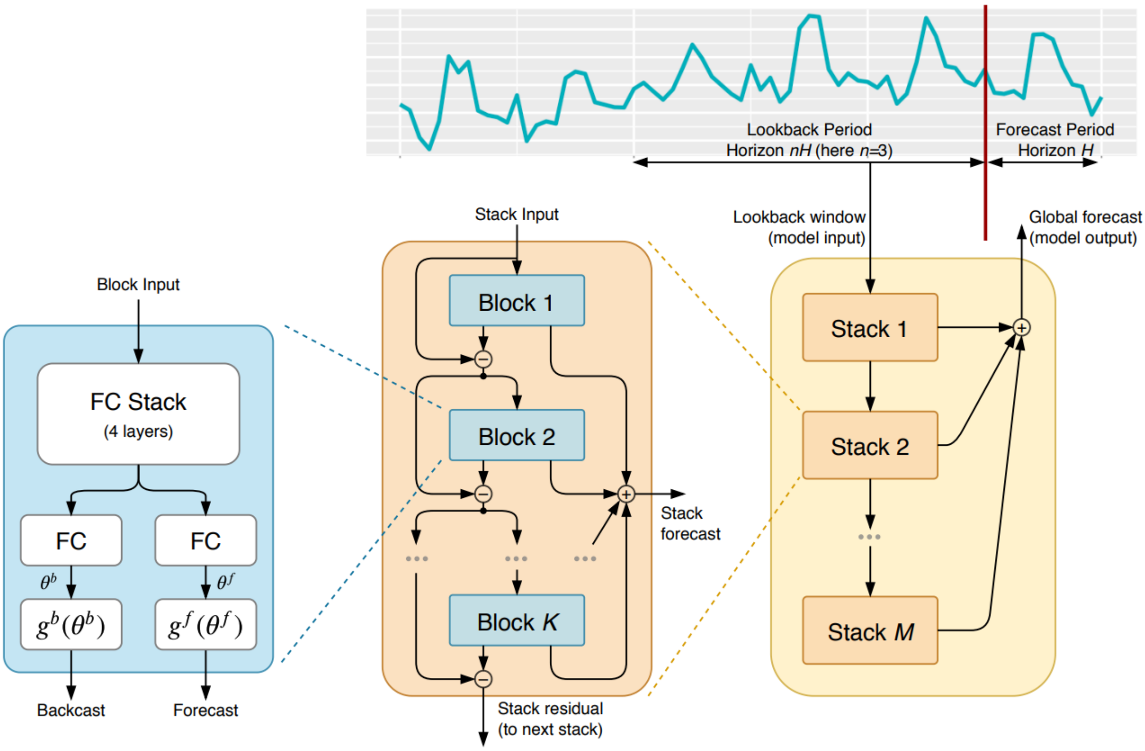

Figure 11: N-BEATS architecture

- **N-HiTS**: The Neural Hierarchical Interpolation for Time Series forecasting is another quite a recent architecture developed by Challu et al., 2023. It was developed with the explicit goal of modeling long term relationships in time series data in a more memory and computation efficient way. The architecture overview can be seen in the following diagram, obtained from the mentioned paper: As it can be seen in Figure 12, the architecture is quite similar to the N-BEATS model with some added complexity. The model works first by sampling the historical window used in forecasting at different frequencies. Then, a series of MLP (feedforward NN) stacks are used in an ensemble way to forecast the following data point at each of the different forecasting frequencies. In the ensemble, each of the MLP stacks produces a prediction of the historic data which is subtracted from the previous historic data, so each MLP block exclusively tries to approximate the residuals of the previous one – very similarly to how each tree reduced the error of the previous tree in the XGBoost model. Then, the predictions across all frequencies are interpolated to provide a prediction for each of the minimal frequency data points.

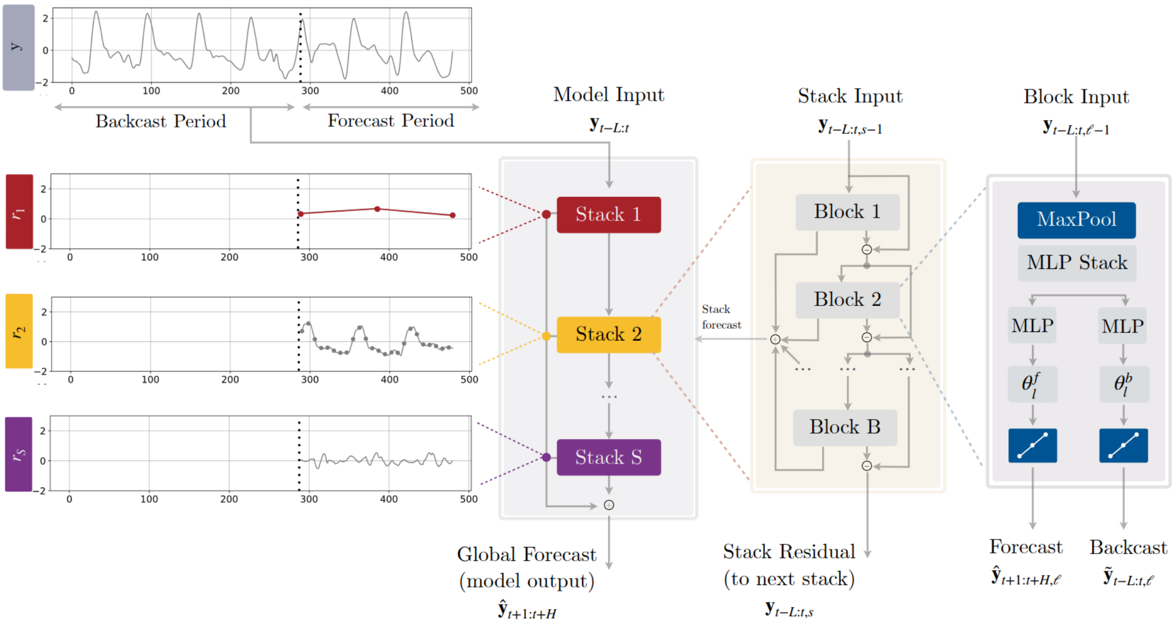

Figure 12: N-HiTS architecture

This architecture allows each stack to focus on capturing the relationships at a different frequency, and it does it in a very efficient way, not really needing all data points for all relationships. This allows for very good performance in several long term time series as shown in the mentioned paper. Even if this is exclusively a univariate model, it has been shown to outperform several state of the art multivariate models. It remains to be seen however if following an exclusively univariate effect the cross relationships between all three series can be maintained.

Also note how given the architecture, in this case it is not necessary to explicitly include time related variables such as the Fourier decompositions as inputs, as these relationships are automatically learned through the different frequencies' sampling.

- **TimeMixer**: The TimeMixer model was prpoposed by S. Wang et al., 2024 in 2024. This MLP architecture leverages the idea that time series present different patters for different sampling scales. It mixes information across different time steps in a much more efficient way than other sequential models like LSTM or attention based models. The TimeMixer model has Past-Decomposable-Mixing (PDM) blocks which for different resolutions extracts the trend and seasonal components – through bottom up and top down mixing respectively. After several PDM blocks, each of the different scales has different information with different prediction capabilities. Thus, a Future-Multipredictor-Mixing block combines the predictions of these different scales. The FMM can be thus understood as an ensemble of multiple predictors where each predictor leverages the information of a different scale. This is probably better understood through the diagram presented by the authors in the paper. This architecture is furthermore inherently multivariate, also mixing information across different variables and thus capturing their interactions. Additionally, thanks to its MLP based architecture it is very computationally efficient compared to other recurrent or attention based models, being capable of having longer input sequences and faster training and execution times.

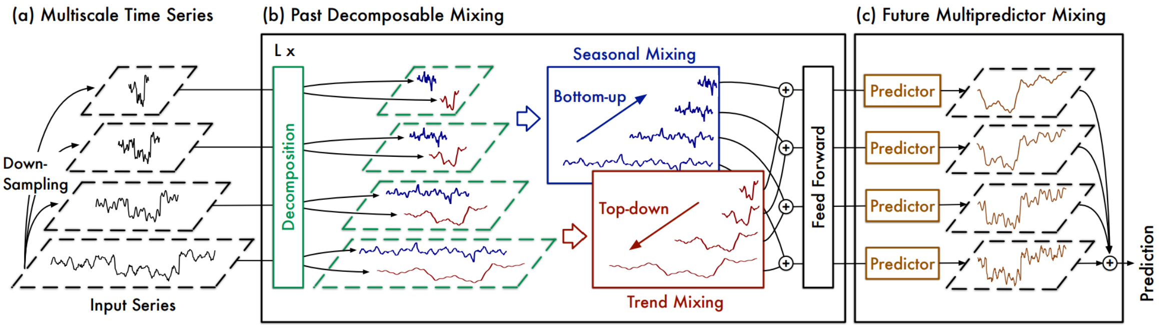

Figure 13: TimeMixer simplified architecture

4.2.4 Transformer models

Transformers are in reality also a type of deep learning or NN model. However, it incorporates a series of new key components that make it different enough to merit its own category. Introduced by Vaswani et al., 2017 in the legendary *Attention is All You Need* paper that sparked the LLM revolution, the transformer architecture is a NN architecture that incorporates a self-attention mechanism used to process data. This self-attention mechanism is in simple terms a way of weighting the importance of different elements in a sequence relative to each other. This allows for a weighted sum of the input sequence focusing on relevant past information. It also contains other key components like the Positional Encoding, which is added to the input sequence to provide information about the position of each element in the such sequence. This general architecture is immensely flexible and has been applied successfully to many problems, most notably to natural language generation. This basic model has also sparked a series of subsequent models with some modifications that make them specially suited for some tasks such as time series modeling. The architectures used in this work are:

- **TFT**: Temporal Fusion Transformers introduced by Lim et al., 2020 are a specific implementation of the transformer architecture used for time series modeling. It has three main components. The first is a set of Variable Selection Networks (VSN) which select the most relevant input variables. These can be static variables, dynamic historically known variables or dynamic generally known variables – where not only the past values are known, but future values are also known. The selected input goes into an LSTM based encoder which captures short term dependencies. Finally, a self-attention based decoder captures long term dependencies using transformers. All of this is coordinated through different gating mechanisms that modulate the flow of information between the different layers. A final key feature of this architecture is its in built ability to produce quantile forecasts. This however will not be specially useful for the task at hand. A more in depth overview of the architecture can be seen in Figure 14, obtained from the aforementioned paper. The TFT architecture has some very interesting features like explicitly capturing short and long term dependencies, regularization through the gating mechanisms, multistep prediction or interpretability of variable importance. However, this comes at the expense of high computational cost, high model complexity and high data requirements.

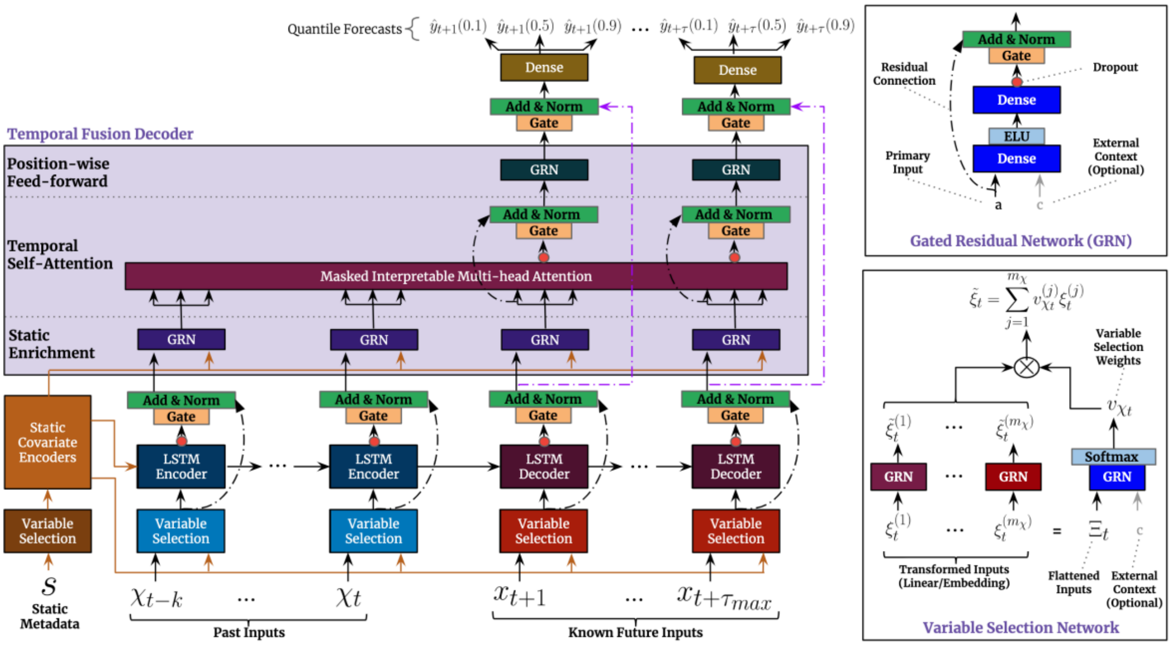
Quantile Forecasts
$\hat{y}_{t+1}(0.1)$ $\hat{y}_{t+1}(0.5)$ $\hat{y}_{t+1}(0.9)$ ... $\hat{y}_{t+\tau}(0.1)$ $\hat{y}_{t+\tau}(0.5)$ $\hat{y}_{t+\tau}(0.9)$
Dense
Add & Norm
Gate
Temporal Fusion Decoder
Position-wise Feed-forward
GRN
Temporal Self-Attention
Masked Interpretable Multi-head Attention
Static Enrichment
Static Covariate Encoders
LSTM Encoder
LSTM Decoder
Variable Selection
s
Static Metadata
χ_{t-k} ... χ_t
x_{t+1} ... $x_{t+\tau_{max}}$
Past Inputs
Known Future Inputs
Residual Connection
Dropout
ELU
Primary Input
External Context (Optional)
a
c
Gated Residual Network (GRN)
$\tilde{\xi}_t = \sum_{j=1}^{m_\chi} v_{\chi_t}^{(j)} \tilde{\xi}_t^{(j)}$
Variable Selection Weights
v_{χ_t}
$\tilde{\xi}_t^{(1)}$ ... $\tilde{\xi}_t^{(m_\chi)}$
Softmax
$\xi_t^{(1)}$ ... $\xi_t^{(m_\chi)}$ = Ξ_t
Transformed Inputs (Linear/Embedding)
Flattened Inputs
External Context (Optional)
Variable Selection Network

Figure 14: TFT architecture

- **Informer**: A different transformer based architecture developed by H. Zhou et al., 2021 specifically developed for forecasting of long sequence time-series through a more efficient implementation. The main advances introduced in this model are the use of a *ProbSparse* self-attention mechanism which has a time and memory complexity of $\mathcal{O}(n \log n)$ instead of $\mathcal{O}\left(n^2\right)$ – with n being the lookback period –, a decoder which predicts the long time-series sequence in one operation rather than step by step and an input method specifically developed to handle extremely long input sequences.

  The Informer has many of the same benefits of other transformer based architectures, but with the added feature of being more efficient and thus being able to incorporate a longer lookback period and thus capturing longer range dependencies that may escape some other architectures. All the while being also more efficient at inference time and thus faster in formulating predictions. However, it also shares many of the same disadvantages of other transformer based models like requiring vast amounts of data for proper training or being very resource intensive, even with the added efficiency.

- **FEDFormer**: The Frequency Enhanced Decomposed Transformer presented by T. Zhou et al., 2022 is yet another transformer based architecture specially designed for long sequence time series modeling. It leverages a principle similar to the N-HiTS. It uses the idea that complex time series have a much more compact representation in the frequency domain, which also captures both long term trends and shorter term seasonal components. To act upon this principle, the authors make several changes to the base transformer architecture, substituting the Multi-Headed Self-Attention block with a Frequency Enhanced Block and the Multi-Headed Cross-Attention block with a Frequency Enhanced Attention block. Both of these leverage Fourier transforms within their operations to operate on the frequency domain. However, the authors also provide the option of using the Wavelet Transform (WT) instead of the Fourier Transform (FT). Using the WT which transforms the input to the frequency and time domain, not only to the frequency domain as the FT, allows for a more nuanced although complex representation of the input data. For this work, the FT implementation will be used due to its simpler implementation.

Altering the transformer to operate on the frequency domain also has as a consequence reducing the complexity further down even from the Informer to linear time and memory.

Apart from accentuating the benefits already mentioned in the Informer, in the paper it is demonstrated how the predicted distribution is closer to the ground truth distribution according to the Kolmogorov-Smirnov distribution test, which is a specially interesting characteristic given the work objectives and evaluation metrics explained in subsection 4.4.

- **iTransformer**: The inverted Transformer model introduced by Y. Liu et al., 2023 is another novel architecture based on transformers which is inherently multivariate. To understand its innovations it is convenient to first understand how normal multivariate transformers work. These transformers take the multivariate time series and slice it temporally. Each token is thus a given timestamp with each of the variable's values. The iTransformer takes the opposite approach. Instead of splitting the time series temporally and embedding each of the time steps, it splits the time series per variable and embeds these variables. The attention mechanism is then run on the variate tokens instead of the temporal tokens. The feedforward network then is run on the time series representation, while in traditional transformers it is run on the multivariate representation. To better understand this concept it an be useful to look at Figure 15 obtained from the mentioned paper. Through this change in architecture, the iTransformer gains some benefits with respect to the traditional transformer. It has been observed that for traditional transformers increasing the lookback period does not necessarily translate to better evaluation

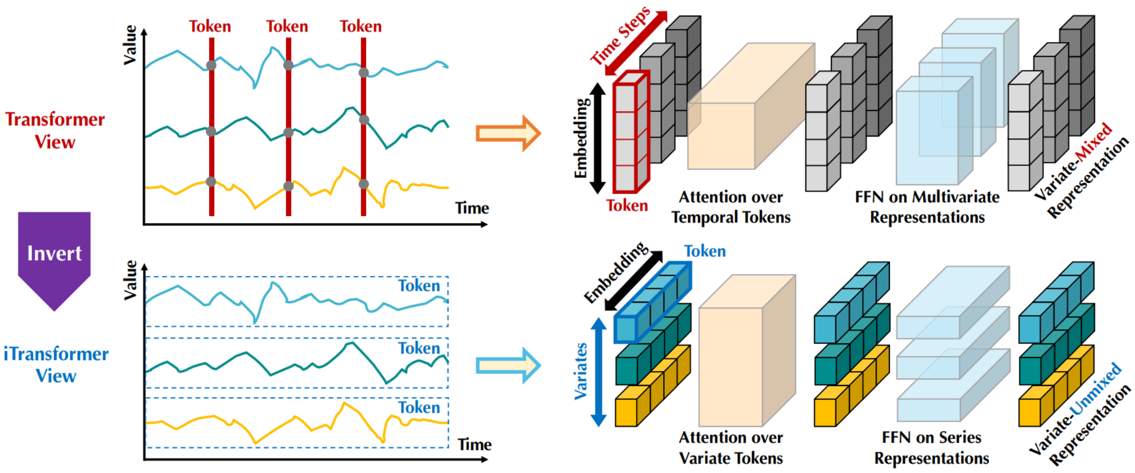
Transformer View
Invert
iTransformer View
Value
Token
Token
Token
Time
Value
Token
Token
Token
Time
Time Steps
Embedding
Token
Attention over Temporal Tokens
FFN on Multivariate Representations
Variate-Mixed Representation
Embedding
Token
Variates
Attention over Variate Tokens
FFN on Series Representations
Variate-Unmixed Representation

Figure 15: Comparison of traditional transformer and inverted transformer embedding and general architecture

metrics due in theory to the *distraction* of the attention mechanism, however by leveraging the MLPs on the temporal dimension the iTransformer can gain more benefit from extended lookback periods. Its architecture also allows for seamless multivariate modeling, allowing it to even better generalize to unseen variables and to easily change the number of variables included in the modeling. In general, the inverted architecture achieves a better performance in most benchmarks as presented not only in the mentioned paper, but also in independent benchmarks like Y. Wang et al., 2024 where the iTransformer is the best performing model for long term forecasting with shared hyperparameters out of the 15 evaluated transformer and NN based models.

It is worth noting how the inverted architecture can be applied to most transformer models, including the ones used here. It is possible for example to create an iInformer model by inverting the dimensions of transformer mechanism in the Informer model. Here however the iTransformer model used is the inverted vanilla transformer architecture, in order to not confound performance with other improvements added by other architectures.

4.2.5 Comparison

After having presented all model classes and the different models that will be implemented themselves, it can be useful to perform a quick comparison and an a priori judgement of which models should outperform the rest, even if all of them will be implemented and evaluated.

It can be seen how there is a very wide range of models from which to choose from – many more than the ones presented here. It is worth noting how here only the base models have been analyzed, and no custom combinations between them have been explored. These combinations can be very useful in combining useful characteristics of two or more base models and creating more accurate predictions as shown in Zou and Yang, 2004.

Out of all models here analyzed, it would seem like the most appropriate due to its intrinsic characteristics is the FEDFormer due to it being able to include a very long forecasting horizon and lookback period in each step, combined with a reasonable robustness that allows it to maintain the capacity to learn long run complex patterns.

All of these characteristics are also shared to a greater or lesser extent by the N-BEATS and N-HiTS, but the FEDFormer can be deployed in a multivariate setup, while N-HiTS and N-BEATS are exclusively univariate. This would force to include the information of the other series as exogenous variables, while a multivariate approach is more straightforward and potentially with less accumulated error for multistep ahead predictions.

The FEDFormer also allows for the use of specific loss functions that could be used to improve accuracy in extreme values.

It is worth noting that given how according to some independent benchmarks the iTransformer generally outperforms the FEDFormer in tasks similar to this one, it is possible that in this case it will also outperform and be the most useful model.

4.3 Loss functions

As previously mentioned, one of the main concerns with long horizon models used to predict solar and wind generation – even more so in the case of wind – is its ability to accurately model the extreme values of the generation distribution. The reasoning for this was already explained in subsection Modeling task characteristics.

One way of enabling models to better approximate these values is through the loss function used to train them. The loss function is a mathematical function that measures how well a model's predictions align with the actual values. It quantifies the errors of the model and guides the learning process, as the model will try to minimize this loss function. Some commonly used loss functions are the Root Mean Squared Error or the Mean Absolute Error for example. The choice of the loss function guides the finally optimal model and thus its performance. A possibility that will be tested in this work is whether modifying the standard loss function to one that penalizes more heavily errors in the extremes can lead to a model with better performance in extreme value metrics. The loss function that will be used for this is the Squared Error Relevance Area introduced in Silva et al., 2022.

4.3.1 SERA

The Squared Error Relevance Function is a loss function that tries to give a bigger emphasis to errors committed in the extremes while still accounting for the performance in the rest of the distribution, thus preventing a severe bias from occurring.

Given a set of points $\mathcal{D} = \{\langle x_i, y_i \rangle\}_{i=1}^{N}$ and a relevance function $\phi : \mathcal{Y} \rightarrow \{0, 1\}$ defined for the target variable Y. Consider the subset $\mathcal{D}^t \subseteq \mathcal{D}$ defined as $\mathcal{D}^t = \{\langle x_i, y_i \rangle \in \mathcal{D} | \phi(y_i) \geq t\}$ the Squared Error Relevance with respect to a given cutoff t (SER_t) is defined as

$$SER_t = \sum_{y_i \in \mathcal{D}^t} (\hat{y}_i - y_i)^2 \tag{20}$$

By integrating this function over all t the $SERA$ is obtained

$$SERA = \int_0^1 SER_t dt \tag{21}$$

In this case the relevance function $\phi(\cdot)$ is heavily inspired on the one used in the referenced paper. It is a Cubic Hermite spline. This is a curve where each segment is a third order polynomial specified by the value and first order derivative at each of the end points. Two intermediate points are chosen according to upper and lower chosen percentiles. For example, the y values for the 10th and 90th percentile values are chosen. These values will be x values for the $\phi(\cdot)$ function with a 0 first degree derivative and 0 or 1 y value, depending on whether the $SERA$ function has been specified to focus on the lower or upper extreme values. If the model is to focus on the upper extreme values, the 90th percentile will have a y value of 1 and the 10th a value of 0. Furthermore, the 0 and 1 points – as these are the theoretical minimum and maximum values for the capacity factor – are determined to have a 0 derivative and a 1 or 0 y value, again depending on whether this is the upper or lower $SERA$.

In order to have a more intuitive understanding of how this custom loss function works and how it affects the predictions of the model, one can look at the small experiment presented in **??**. For this quick demonstration, a sample problem has been created. A set of points have been distributed following a quadratic function. An XGBoost model has been trained to predict the y value of the points based on the x value. However, instead of the usual quadratic error loss, a SERA loss function with a relevance function which disregards the bottom 30% of the data has been chosen. The bottom 30% is given a relevance of 0, the top 60% is given a relevance of 1 and the values in between are given a relevance based on the spline fit between those percentiles. As it can be seen, the XGBoost focuses only on the top 70% of the data, completely disregarding the data points at the bottom. In this toy example it may seem like introducing the SERA loss only worsens the prediction. However, this is only due to the true function being an extremely easy one to fit and thus predict. With more complex functions, allowing the model to focus only on the parts of the distribution one deems more important can really help the model in the fitting of those sections.

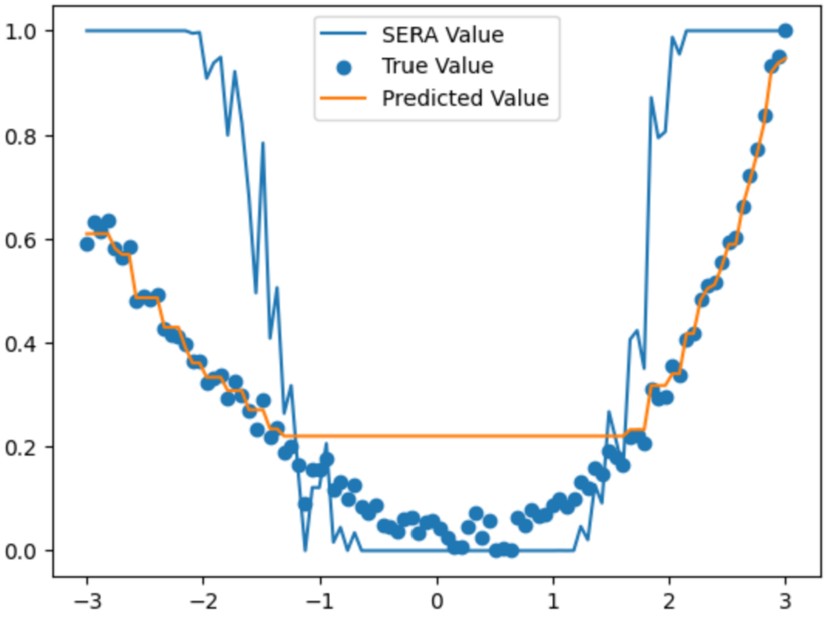

Figure 16: Visualization of the SERA loss and its effects

For the actual implementation of this loss function, note how many of the models mentioned in the Models section are able to work with custom loss functions. However, to analyze whether a custom loss function helps in improving model accuracy for extreme values it is not necessary to retrain all models with this loss function. A single model will be trained with this loss function and its results will be compared to those obtained with the baseline loss function (MAE). The chosen model for the analysis has been the XGBoost given its ease of training and simplicity. In order to implement the *SERA* loss function for the XGBoost library, functions defining its gradient and hessian have been created.

4.4 Evaluation metrics

In order to understand why the following metrics have been chosen for evaluation it is important to keep in mind the overarching goal of the project. The goal of the mentioned models is not to provide an accurate short term forecast, but to provide a model that properly simulates the long term dynamics of the three time series. That is, models that are able of producing long term scenarios of solar and wind generation with univariate and multivariate dynamics equivalent to the real world ones, in order to optimize and test several grid planning possibilities on these scenarios.

In order to best achieve this goal, the best performing model will have to be evaluated on different metrics, each corresponding to different attributes needed for correct whole picture modeling. These are the different attributes that will be tested and the metrics that will be used to test them.

4.4.1 Marginal distributions

The first metric that will be evaluated will be the similarity of the univariate series with the real world data. Note that this similarity is not a point wise similarity. In many relevant papers, the accuracy of a series prediction is calculated through metrics like the RMSE or the MAPE. However, these metrics require that the prediction try to get right the value at each time step. If there is a delay of one time step, or a "cloudy week" comes a week earlier or later the RMSE will give a very poor score even though the general behavior is the same. That is why the approach taken here will be slightly different. The metrics used will be the following:

- **Cramér von Mises criterion**: The Cramér von Mises criterion introduced by Cramér, 1928 measures the distance between two CDF. It is calculated like:

  $$\omega^2 = \int_{-\infty}^{\infty} \left[F_m\left(x\right) - F_e\left(x\right)\right]^2 dF_e\left(x\right) \tag{22}$$

  Where $F_m\left(x\right)$ is the modelled empirical CDF and $F_e\left(x\right)$ is the observed empirical CDF.

- **Kullback-Leibler divergence**: The KL divergence or relative entropy introduced in

Kullback and Leibler, 1951 is a statistical distance that measures how different a probability distribution is from another reference one. It is calculated like

$$D_{KL}\left(P||Q\right) = \sum_{x} P\left(x\right) \log\left(\frac{P\left(x\right)}{Q\left(x\right)}\right) \tag{23}$$

where P is the modelled probability distribution and Q is the reference probability distribution.

In its implementation the probability distributions for the observed and predicted data are calculated through a kernel density estimate with a Gaussian kernel, as explained in the Individual statistical characteristics section. The KL Divergence is then calculated on these PDFs.

- **ACF distance**: Even though the marginal distributions are by themselves a fundamental part of the closeness of the univariate series to reality, they are not the only one. Another key aspect of the series is its temporal self dependency, studied through the ACF of the correlation coefficient and the new correlation coefficient in the section called Lagged relationships. The difference between the ACF of the ξ metric of the empirical data and the model will be measured through a weighted root mean squared error, calculated as:

$$\text{ACFD} = \sqrt{\frac{\sum_{k=0}^{N}|w_k|\left[ACF_m\left(k\right) - ACF_e\left(k\right)\right]^2}{\sum_{k=0}^{N}|w_k|}} \tag{24}$$

where N is the number of periods for which the metric is calculated – 72 in this case, ACF_m is the model's ξ ACF, ACF_e is the empirical ξ ACF and w is the weighting factor, determined to be $w_k = ACF_e\left(k\right)$. This has been chosen so that a higher weight is given to those lags with a higher empirical relevance.

4.4.2 Joint distributions

Now that it is clear how the marginal distributions will be assessed, it is time to look at how the dependence structure between the three series will be tested. There are several metrics that will be used for that purpose:

- **Copula correlation matrix distance**: In the section regarding Multivariate analysis, the correlation matrix of a Gaussian copula representing the dependence structure between the series was calculated. The same matrix will be calculated for samples of each of the models. Then, the distance (CCMD) between both matrices will be calculated. This distance is calculated as:

$$CCMD = \sqrt{\sum_i \sum_j (a_{ij} - b_{ij})^2} \tag{25}$$

where $A = \{a_{ij}\}$ is the correlation matrix of the copula fitted to the modelled data and $B = \{b_{ij}\}$ is the correlation matrix shown in Table 2.

- **CCF distance**: The cross correlation function distance is analogous to the autocorrelation function distance explained above, but for the ξ cross correlation function instead of ACF.

4.4.3 Extreme value analysis

A significantly important aspect of the models is their need to be particularly accurate on the extreme values. Periods of extreme wind or solar generation are particularly interesting to the grid, due to their displacement of other energy sources or their need of them. That is why the accuracy of the models on extreme values will be assessed through these metrics:

- **Conditional Value at Risk distance**: The CVaR, usually used in financial risk management, measures what a certain value on average will be given that a certain threshold has been exceeded. It is calculated as

  $$\mathrm{CVaR}_\alpha(x) = \mathbb{E}\left[X \mid X > \mathrm{VaR}_\alpha\right] \tag{26}$$

  with VaR_α being the Value at Risk for a given certainty. That is, the maximum value not exceeded with a probability α. It is characterized as

  $$\mathrm{VaR}_\alpha = \inf\{x | F_X(x) \geq \alpha\} \tag{27}$$

  The assessment metric will be the difference between the modeled and empirical CVaR calculated as

  $$\mathrm{CVRD} = \frac{\mathrm{CVaR}^m_{95\%}}{\mathrm{CVaR}^e_{95\%}} - 1 \tag{28}$$

where $\text{CVaR}^m_{95\%}$ is the CVaR at a 95% level for the model and $\text{CVaR}^e_{95\%}$ is the empirical CVaR at a 95% level.

- **Tail dependence coefficient**: The tail dependence coefficient is a measure of the co-movements of the tails of their distributions. It can be lower tail dependence or upper tail dependence, with the lower tail dependence calculated as

$$\lambda_l = \lim_{q \to 0} P\left(X_m \leq F_m^{-1}(q) \mid X_e \leq F_e^{-1}(q)\right) \tag{29}$$

  and the upper tail dependence coefficient being calculated as

$$\lambda_u = \lim_{q \to 1} P\left(X_m > F_m^{-1}(q) \mid X_e > F_e^{-1}(q)\right) \tag{30}$$

  with F_m^{-1} being the inverse CDF for the model and F_e^{-1} being the empirical inverse CDF.

  The tail dependence between each modeled variable with its corresponding empirical variable will be studied – e.g. tail dependence between modeled solar PV with empirical solar PV.

  Note how in practice there is not enough data to calculate an infinitesimally continuous F_m^{-1} or F_e^{-1}. Therefore, an approximation will be done where instead of calculating the limit, the coefficient for a $q = 0.05$ or $q = 0.95$ – depending on which tail is being studied – will be used.

- **Return level distance**: The GEV function is a function often used to model the maxima of sequences of random variables. Its CDF is

$$P\left(GEV\left(\mu, \sigma, \xi\right) \leq x\right) = e^{-t(x)} \tag{31}$$

  with

$$t\left(x\right) \equiv \begin{cases} \left[1 + \xi\left(\frac{x-\mu}{\sigma}\right)\right]^{-\frac{1}{\xi}}, & \text{if } \xi \neq 0, \\ e^{\left(-\frac{x-\mu}{\sigma}\right)}, & \text{if } \xi = 0. \end{cases} \tag{32}$$

  As it can be seen from the formulation above the distribution of maximum values depends on three variables, with ξ governing the tail behavior. Once the GEV distribution is fitted to the data, the return level can be estimated. The return level is the value expected to be exceeded on average once in a given period. If the 100-year return level of wind capacity factor is 0.95, that means that on average there will be one instance every 100 years when the capacity factor exceeds 0.95. The return level z_T is calculated as

$$z_T = \mu + \frac{\sigma}{\xi}\left[\left(-\log\left(1-\frac{1}{T}\right)\right)^{-\xi} - 1\right] \tag{33}$$

Thus, the weekly maxima will be obtained with the data, with which a GEV distribution will be fitted and a 10-year return level will be calculated. The assessment metric will be the percentage difference in return level calculated as

$$\text{RLD} = \frac{z_{10}^{m}}{z_{10}^{e}} - 1 \tag{34}$$

This will be calculated for both the maxima and the minima of the distributions.

The extreme value assessment is particularly important for wind generation, rather than for solar generation as it is much more consistent. Therefore, these values will only be calculated for the wind series.

4.5 Model training and validation

After having laid out the different evaluation metrics that will be used, the method used for training and validating the models will be presented.

A general train-test-validation split will be used. This approach splits the available data in three groups. The first will be used to train the models. The second will be used to test the models and iteratively improve these models, by checking for overfitting and evaluating different parameter setups. The validation data is left as the final evaluation data, where the final metrics that will be presented are calculated. The metrics obtained in this group can be seen as the metrics that are to be expected of the models in a real world context after deployment. A split of 1 year validation data, 1 year testing data and the rest as training data has been chosen, as a 1-year prediction horizon is a long enough horizon to test the long run characteristics of the model and it contains values for all different seasons.

As it can be seen in the Evaluation metrics section there are many evaluation metrics attempting to check for different characteristics of the model. Therefore, using a programmatic approach to model creating through something like a grid search of parameters is not really useful, as it can be difficult to determine in an algorithmic way which results are better than others, as one model can have better results modeling extreme values while others can have better results in modeling cross relationships. Therefore, a manual approach to model improvement with the test set will be employed, and only the best overall model selected in a discretionary way will be evaluated on the validation set and will have its results presented in this work.

Another thing worth mentioning is that no cross validation or bootstrapping methods will be used to evaluate the models. Although these approaches would give a more robust understanding of the validity of the model by being more resistant to anomalies in the test and validation sets, the amount of data available for training – specially for the neural network and transformer based models – is not enough in order to sacrifice even more for testing and validation purposes.

As an alternative approach to cross validation and bootstrapping, the test and validation sets will be evaluated for anomalies. That is, they will be evaluated as if they were a prediction against the previous year. The results are the following:

Solar PV	Period	
	Test	Validation
Cramer von Mises	29.34	9.48
KL divergence	0.0003	0.0004
ACF_ξ distance	0.0127	0.0137
CCMD	0.0112	0.0167
$\text{CCF}_\xi^{SolarTH}$ distance	0.0099	0.0338
CCF_ξ^{Wind} distance	0.0119	0.0372
CVaR^+ distance	-0.0297	-0.0064
Tail dependence coefficient$^+$	0.3676	0.4703
Return level distance$^+$	0.0154	-0.0248

Table 3: Evaluation metrics of the test and validation sets with respect to the previous year for solar PV

At a first glance it can be seen how some metrics seem to be more consistent than others. The Cramer von Mises statistic for example exhibits what seems to be a great difference between one and the other year, while the KL divergence seems to be more reliable exhibiting a much lesser difference. These values will be useful as they represent a baseline of what a reasonable *error* according to the different metrics would be. If when evaluating a model the metric is similar to the ones shown here, the predicted values can be approximated to be close enough to what a the real process would generate.

Note how for solar PV the metrics regarding the extreme values in the left tail have been left out, as the values here are very close to zero and thus these metrics are very noisy. The same will be done for solar TH.

Solar TH	Period	
	Test	Validation
Cramer von Mises	12.86	9.89
KL divergence	0.0102	0.0265
ACF_ξ distance	0.0249	0.0162
CCMD	0.0112	0.0167
$\text{CCF}_\xi^{SolarPV}$ distance	0.0151	0.0155
CCF_ξ^{Wind} distance	0.0201	0.0229
CVaR^+ distance	0.0732	-0.1318
Tail dependence coefficient$^+$	0.3356	0.2968
Return level distance$^+$	0.2928	-0.1504

Table 4: Evaluation metrics of the test and validation sets with respect to the previous year for solar TH

In Table 4 the same metrics shown before but for solar TH can be seen. Again, these will be used as benchmarks for the modeling of solar thermal by the different models. Note how the CCMD metrics are the same, as the copula matrix is the same one for all three series, and the distance is thus also equivalent.

Finally, in Table 5 the same metrics that will be used as benchmarks for wind can be seen. Here, the metrics relating to the extremes in the negative tail have been included since for wind it does hold some interest to see whether the model accurately models periods of low wind generation.

Wind	Period	
	Test	Validation
Cramer von Mises	9.48	5.01
KL divergence	0.0.0018	0.0018
ACF_{ξ} distance	0.0602	0.0630
CCMD	0.0112	0.0167
$\text{CCF}_{\xi}^{SolarPV}$ distance	0.0128	0.0111
$\text{CCF}_{\xi}^{SolarTH}$ distance	0.0136	0.0131
CVaR^{+} distance	-0.0236	-0.0016
CVaR^{-} distance	-0.2461	0.2123
Tail dependence coefficient^{+}	0.0479	0.2443
Tail dependence coefficient^{-}	0.1506	0.1119
Return level distance^{+}	1.3618	-0.0071
Return level distance^{-}	-13.5111	-1.1904

Table 5: Evaluation metrics of the test and validation sets with respect to the previous year for wind

5 Results

After having described all the different implemented models, having formulated a hypothesis about which should be able to outperformed and explained how the models are trained and evaluated, it is now time to present and analyze the obtained results. This section will have four subsections. The first three will focus on each of the three capacity factors, analyzing separately which models outperform and why in each of them. The final section will be devoted to analyzing whether the custom loss function successfully outperforms the base loss function in extreme value modeling.

5.1 Solar PV

The first results that will be presented are those of the photovoltaic solar capacity factor. The results will be presented for all metrics for all evaluated models similarly as was presented in Table 4: Evaluation metrics of the test and validation sets with respect to the previous year for solar TH.

The metrics of the validation set obtained in the Model training and validation section will also be included as a reference point.

Solar PV	Benchmark	Regression	SARIMAX	VARMAX	SVM	XGBoost	N-BEATS	N-HiTS	TimeMixer	TFT	Informer	FEDFormer	iTransformer
Cramer von Mises	29.34	364.6	93.27	295.7	223.3	747.4	69.43	47.81	52.58	2221	485.8	158.6	131.1
KL divergence	0.0002762	0.4844	0.1905	1.692	0.1662	1.586	0.3452	0.003905	0.04968	2.721	1.302	0.5261	0.01814
ACF$_\xi$ distance	0.0127	0.1201	0.1089	0.1163	0.2289	0.1802	0.1018	0.06985	0.04528	0.1769	0.2429	0.1729	0.05605
CCMD	0.01117	0.3291	0.1044	0.2151	0.8375	0.3549	0.1959	0.4296	0.05632	0.5398	0.5675	0.5183	0.2093
CCF$_\xi^{SolarTH}$ distance	0.009855	0.1232	0.08045	0.223	0.1484	0.3242	0.0481	0.1829	0.0826	0.2603	0.1649	0.2608	0.0688
CCF$_\xi^{Wind}$ distance	0.01187	0.07759	0.04984	0.1515	0.1091	0.8558	0.02316	0.01634	0.07532	0.4126	0.06594	0.1383	0.08443
CVaR$^+$ distance	-0.02966	8.394	0.08376	-0.6706	0.6503	-0.7125	1.037	0.003395	0.1888	-0.9442	-0.6828	-0.3569	0.1683
Tail dependence coefficient$^+$	0.3676	0.2671	0.3174	0.000	0.1187	0.03397	0.2831	0.3265	0.2032	0.000	0.04338	0.002283	0.3699
Return level distance$^+$	0.01537	1.857e+07	-0.3411	-1.091	0.7803	-0.9312	4.802	-0.1408	0.7089	-1.09	-0.02178	-0.3841	0.2181

Table 6: Results of the different models for Solar PV

Given the great number of metrics and the difficulty of easily interpreting them, here is another table with the rank of the model for each metric. That is, for each metric, each model will be shown as the 1st, 2nd, 3rd... best model. The benchmark has also been included in this ranking. In most metrics a lower value is better, while for the distances in the extreme values' category it is having a lower absolute value what is important, as the deviation in either sense is negative. The Tail dependence coefficient is the only metric where a higher value is better. In bold, the best model – apart from the benchmark – is highlighted.

Solar PV	Benchmark	Regression	SARIMAX	VARMAX	SVM	XGBoost	N-BEATS	N-HiTS	TimeMixer	TFT	Informer	FEDFormer	iTransformer
Cramer von Mises	(1)	(10)	(5)	(9)	(8)	(12)	(4)	**(2)**	(3)	(13)	(11)	(7)	(6)
KL divergence	(1)	(8)	(6)	(12)	(5)	(11)	(7)	**(2)**	(4)	(13)	(10)	(9)	(3)
ACF$_\xi$ distance	(1)	(8)	(6)	(7)	(12)	(11)	(5)	(4)	**(2)**	(10)	(13)	(9)	(3)
CCMD	(1)	(7)	(3)	(6)	(13)	(8)	(4)	(9)	**(2)**	(11)	(12)	(10)	(5)
CCF$_\xi^{SolarTH}$ distance	(1)	(6)	(4)	(10)	(7)	(13)	**(2)**	(9)	(5)	(11)	(8)	(12)	(3)
CCF$_\xi^{Wind}$ distance	(1)	(7)	(4)	(11)	(9)	(13)	(3)	**(2)**	(6)	(12)	(5)	(10)	(8)
CVaR$^+$ distance	(2)	(13)	(3)	(8)	(7)	(10)	(12)	**(1)**	(5)	(11)	(9)	(6)	(4)
Tail dependence coefficient$^+$	(2)	(6)	(4)	(12)	(8)	(10)	(5)	(3)	(7)	(12)	(9)	(11)	**(1)**
Return level distance$^+$	(1)	(13)	(5)	(11)	(8)	(9)	(12)	(3)	(7)	(10)	**(2)**	(6)	(4)
Total	(11)	(78)	(40)	(86)	(77)	(97)	(54)	**(35)**	(41)	(104)	(79)	(80)	(37)

Table 7: Results of the rank of the different models for Solar PV

At the bottom of Table 9 a Total can be found. This total is the sum of the rank of the model across all metrics and it can be seen as a very simple proxy of the overall goodness of the model across all domains – distribution fit, multivariate coherence and extreme value fit.

The most relevant and straightforward insight that can be obtained from a quick glance at the results is the superiority of the NN based models in general, and more precisely of the N-HiTS model. This is most probably due to the high seasonality of the solar PV series. This model is specially designed to operate at different frequencies, thus extracting all relevant frequency based patterns but without overfitting. When looking at each evaluation category, the N-HiTS seems to be the best regarding the general distribution of the forecast, while the TimeMixer seems to be somewhat better regarding the temporal self-consistency – that is, how correlated a given value should be with its past values – and the N-BEATS seems to be the better one regarding multivariate cross relationship. Finally in the extreme value modeling the N-HiTS again stands as the best model, with the other two NN models doing considerably worse.

In the statistical methods a very big difference between the SARIMAX and the other two models can be seen. The SARIMAX is actually the best model after the N-HiTS and the iTransformer practically tying with the TimeMixer, performing quite well in all evaluation metric categories. This is again thanks to the SARIMAX being able to utilize the frequency components given by the Fourier decompositions without overfitting. The order 1 integration in the SARIMAX however seems to be of critical importance, given how the other two statistical methods perform considerably worse and this is one of the main differences.

Within the transformer based models there seems to also be a great difference between the iTransformer – which is actually the second best model practically tied with the N-HiTS, doing worse in general distribution but better in multivariate coherence – and the rest of the transformer models, with the latter being at a performance level similar to the vanilla Regression model. This could be attributed to the complexity of the models, which could have led them to overfitting. The good performance of the iTransformer with respect to the other architectures is consistent with the benchmarks performed by Y. Wang et al., 2024.

Finally, regarding the machine learning methods the conclusions are similar to those of the transformer based models. Their performance is among the worst of all the models. They seem to have joined the worst characteristics of the simple statistical methods and the more complex NN based methods. The way of encoding seasonal and time varying data is not as sophisticated as for the NN models as they only use the Fourier decompositions, however by adding extra complexity in the way those decompositions are used compared to the statistical methods they seem to overfit and end up performing even worse than these simpler models. Among themselves, it seems like the SVM is better equipped than the XGBoost to model the solar PV data.

Here is a plot of the yearly and weekly for a random week modeled data, together with the actual data. Note how on a moment to moment basis the data does not need to be similar, as the goal is not to accurately predict the hourly value of the capacity factor, but to get a distribution of data with the same characteristics. Note how the modeled data is much more rugged on an hourly basis, while the actual data follows a much smoother hour to hour pattern. However, this model is still more accurate with that ruggedness than other smoother models like the SARIMAX or the VARMAX. On the yearly pattern it can be seen how the model

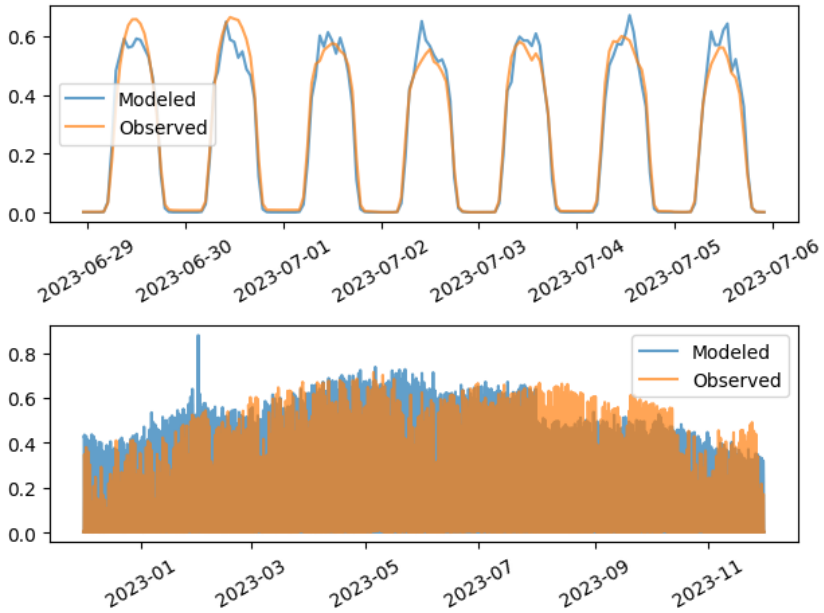

Figure 17: Weekly and yearly predicted and actual solar PV capacity factor for the N-HiTS model

greatly overestimates some extreme positive values and has some sudden drops in capacity factor around August, but the general patterns are maintained.

5.2 Solar TH

For the solar thermal capacity factor series, the results will be shown in the same way as for the Solar PV section. The two tables showing the metrics' values and the rank of the models will be presented, and they will later be discussed.

Solar TH	Benchmark	Regression	SARIMAX	VARMAX	SVM	XGBoost	N-BEATS	N-HiTS	TimeMixer	TFT	Informer	FEDFormer	iTransformer
Cramer von Mises	12.87	239.5	144.6	209.2	93.37	653.9	134	124.2	251.4	699.1	714.9	495.9	26.38
KL divergence	0.01016	0.2558	0.1379	1.014	0.4101	4.31	0.2427	0.05258	0.9605	0.4706	0.8153	1.35	0.04629
ACF$_\xi$ distance	0.02488	0.2246	0.1936	0.1826	0.1988	0.2507	0.1812	0.5174	0.1061	0.1951	0.4585	0.1844	0.1639
CCMD	0.01117	0.3291	0.1044	0.2151	0.8375	0.3549	0.1959	0.4296	0.05632	0.5398	0.5675	0.5183	0.2093
CCF$_\xi^{SolarPV}$ distance	0.01513	0.1008	0.08724	0.223	0.1686	0.2552	0.03605	0.1889	0.09468	0.168	0.2716	0.2363	0.07058
CCF$_\xi^{Wind}$ distance	0.02013	0.08723	0.1107	0.1376	0.1124	0.861	0.118	0.6707	0.07896	0.208	0.5203	0.1325	0.1419
CVaR$^+$ distance	0.0732	2.869	-0.04785	-0.5135	11.09	-0.8908	1.818	-0.1942	-0.6187	5.324	-0.1614	-0.7453	-0.1516
Tail dependence coefficient$^+$	0.3356	0.3676	0.3539	0.2192	0.05479	0.08844	0.3813	0.05251	0.2192	0	0.09589	0.08904	0.3721
Return level distance$^+$	0.2928	17.04	-1.004	-0.9157	1.394e+10	-0.8234	6.143	-1.018	-0.8051	0.8068	0.0144	-0.8311	-0.3048

Table 8: Results of the different models for Solar TH

Again, the model rank table will be shown to better interpret these results.

Solar TH	Benchmark	Regression	SARIMAX	VARMAX	SVM	XGBoost	N-BEATS	N-HiTS	TimeMixer	TFT	Informer	FEDFormer	iTransformer
Cramer von Mises	(1)	(8)	(6)	(7)	(3)	(11)	(5)	(4)	(9)	(12)	(13)	(10)	**(2)**
KL divergence	(1)	(6)	(4)	(11)	(7)	(13)	(5)	(3)	(10)	(8)	(9)	(12)	**(2)**
ACF$_\xi$ distance	(1)	(10)	(7)	(5)	(9)	(11)	(4)	(13)	**(2)**	(8)	(12)	(6)	(3)
CCMD	(1)	(7)	(3)	(6)	(13)	(8)	(4)	(9)	**(2)**	(11)	(12)	(10)	(5)
CCF$_\xi^{SolarPV}$ distance	(1)	(6)	(4)	(10)	(8)	(12)	**(2)**	(9)	(5)	(7)	(13)	(11)	(3)
CCF$_\xi^{Wind}$ distance	(1)	(3)	(4)	(8)	(5)	(13)	(6)	(12)	**(2)**	(10)	(11)	(7)	(9)
CVaR$^+$ distance	(2)	(11)	**(1)**	(6)	(13)	(9)	(10)	(5)	(7)	(12)	(4)	(8)	(3)
Tail dependence coefficient$^+$	(5)	(3)	(4)	(6)	(11)	(10)	**(1)**	(12)	(6)	(13)	(8)	(9)	(2)
Return level distance$^+$	(2)	(12)	(9)	(8)	(13)	(6)	(11)	(10)	(4)	(5)	**(1)**	(7)	(3)
Total	(15)	(66)	(42)	(68)	(82)	(93)	(48)	(77)	(48)	(86)	(83)	(80)	**(32)**

Table 9: Results of the rank of the different models for Solar TH

The first thing that should be taken into account before analyzing these results is that the solar thermal series is very similar to the solar photovoltaic series, therefore similar results should be expected. The only difference being more variability at night and higher dependence on previous values of the cross series, mainly of solar photovoltaic.

Perhaps because of these characteristics there seems to be quite a difference in performance among the top models compared to solar PV. In this case, the iTransformer is clearly the best model. This model is clearly the best in modeling the general distribution of the data, but also does incredibly well in modeling extreme values and quite well on the multivariate characteristics. This could be explained by the fact that the solar thermal series as explained has somewhat of a more complex relationship with lagged values and solar PV and wind values. Thanks to its ability to be stored, its capacity factor is no longer exclusively reliant on weather conditions but also depends on human decisions. That is human decisions that could be affected by how much energy has been stored – highly correlated with past solar PV data – and how much has been or is expected to be produced by the rest of the energy resources. As for the rest of the transformer models, they strongly underperform, only doing comparatively better in extreme value modeling, perhaps again due to being able to capture the complex relationships between these extreme values and past or cross values.

Another surprising part is the drop in performance by the N-HiTS, perhaps due to the higher complexity in this series in the lesser significance of the frequency structure leveraged by this model. The TimeMixer and N-BEATS on the other hand remain really strong performers tying at third place among all models. These models are specially good at capturing the cross relationships between this series and the other two. The TimeMixer model is specially good at this and at capturing the temporal self-consistency of the series, while it is not nearly as good at capturing the general distribution of the data.

Among the statistical models the analysis is very similar to solar PV, with the SARIMAX model being again the best within this family and one of the best overall, with the other two models performing significantly worse.

Regarding the ML models a behavior similar to the last one is observed, with both doing really poorly although slightly better for the SVM, with the only category where XGBoost does better being the extreme value modeling. This is coherent with what has been observed for the transformer models, where models with more parameters capable of capturing more complex patterns but more prone to overfitting do comparatively better in this evaluation category.

Below the same representation as before is given but in this case for the iTransformer model, given how this one is the best in this case. In this case, the same ruggedness problem can be seen, which is common to all NN and transformer based models. However, the solar thermal series is not as smooth as the solar photovoltaic, so it does not stand out as much. It can also be seen how the model accurately models the nighttime generation for solar thermal. As for the yearly pattern, the model seems to greatly underestimate generation during the winter months and it also assumes a much more uniform generation profile than the real one.

5.3 Wind

The last of the three series is the most different compared to the other two, given the much lower seasonality in wind production and its higher degree of dependence on past cross series. The results obtained for the wind series are the following. Note how in this case both the positive

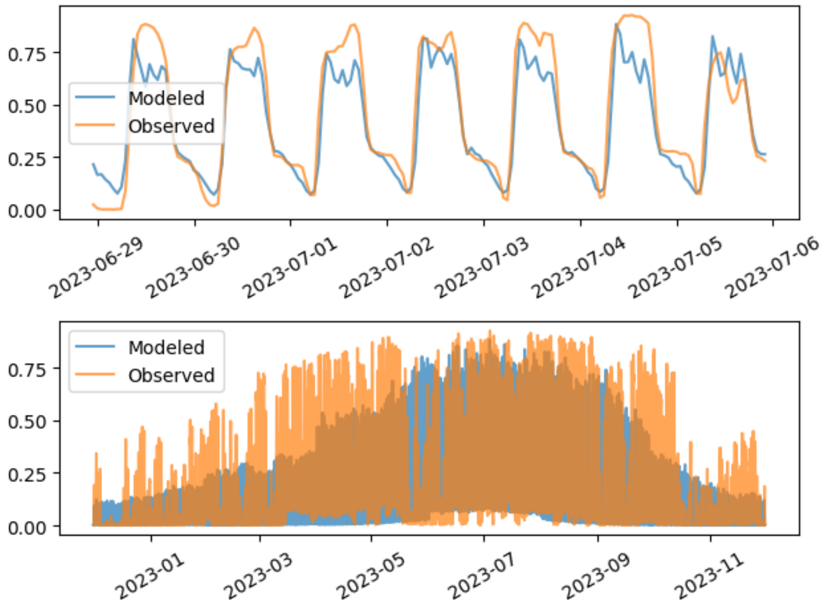

Figure 18: Weekly and yearly predicted and actual solar TH capacity factor for the iTransformer model

and negative extreme values hold some interest as the lower end of the distribution is no longer zero. In fact, moments of very low wind generation are of special interest given these are the moments when the rest of the energy sources would have to step up.

Wind	Benchmark	Regression	SARIMAX	VARMAX	SVM	XGBoost	N-BEATS	N-HiTS	TimeMixer	TFT	Informer	FEDFormer	iTransformer
Cramer von Mises	9.483	1783	1814	360.4	434.2	774.6	381.2	504	553.8	729.6	1283	236	309.9
KL divergence	0.001479	1.109	1.208	0.1067	0.1318	0.3214	0.4356	0.2818	0.2442	0.3298	0.6615	0.1116	0.1838
ACF$_\xi$ distance	0.06016	0.3458	0.3143	0.4598	0.3302	0.6625	0.2918	0.4957	0.2992	0.3452	0.5537	0.3348	0.3312
CCMD	0.01117	0.3291	0.1044	0.2151	0.8375	0.3549	0.1959	0.4296	0.05632	0.5398	0.5675	0.5183	0.2093
CCF$_\xi^{SolarPV}$ distance	0.01282	0.1621	0.1686	0.05547	0.115	0.4045	0.03014	0.03572	0.04976	0.3294	0.4744	0.03619	0.1731
CCF$_\xi^{SolarTH}$ distance	0.01363	0.242	0.2763	0.0691	0.0808	0.525	0.1954	0.7625	0.06438	0.4002	0.7358	0.03586	0.2802
CVaR$^+$ distance	-0.02357	0.4225	0.06328	-0.03603	0.02966	-0.5748	-0.5646	-0.504	-0.3975	-0.5549	-0.1147	0.5273	-0.395
CVaR$^-$ distance	-0.2462	5.582	6.485	3.104	2.912	5.746	2.788	4.813	3.941	5.036	4.858	0.7204	3.297
Tail dependence coefficient$^+$	0.04795	0.09132	0.1256	0.1301	0.1119	0.05028	0.09132	0.105	0.08904	0.05479	0.02727	0.02968	0.07534
Tail dependence coefficient$^-$	0.1507	0.1027	0.1553	0.1484	0.05023	0.05003	0.1119	0.1005	0.09589	0.05251	0.1005	0.04795	0.1872
Return level distance$^+$	1.362	-0.4957	-0.4707	-0.5817	-0.5813	-1.081	-0.8006	-0.9889	-0.8401	-0.9307	-0.4494	0.7511	-0.786
Return level distance$^-$	-13.51	-7.643	-8.188	-4.958	-3.629	-5.709	-2.105	-1.986	-5.443	-6.162	-6.413	-0.6112	-1.679

Table 10: Results of the different models for Wind

The rank of each of the models can be seen in Table 11. Note how in this case, since both the upper and lower extreme metrics are evaluated, the weight of the extreme value metrics in the total rank will be higher and can skew the perceived overall performance of the models when using this metric.

Wind	Benchmark	Regression	SARIMAX	VARMAX	SVM	XGBoost	N-BEATS	N-HiTS	TimeMixer	TFT	Informer	FEDFormer	iTransformer
Cramer von Mises	(1)	(12)	(13)	(4)	(6)	(10)	(5)	(7)	(8)	(9)	(11)	**(2)**	(3)
KL divergence	(1)	(12)	(13)	**(2)**	(4)	(8)	(10)	(7)	(6)	(9)	(11)	(3)	(5)
ACF$_\xi$ distance	(1)	(9)	(4)	(10)	(5)	(13)	**(2)**	(11)	(3)	(8)	(12)	(7)	(6)
CCMD	(1)	(7)	(3)	(6)	(13)	(8)	(4)	(9)	**(2)**	(11)	(12)	(10)	(5)
CCF$_\xi^{SolarPV}$ distance	(1)	(8)	(9)	(6)	(7)	(12)	**(2)**	(3)	(5)	(11)	(13)	(4)	(10)
CCF$_\xi^{SolarTH}$ distance	(1)	(7)	(8)	(4)	(5)	(11)	(6)	(13)	(3)	(10)	(12)	**(2)**	(9)
CVaR$^+$ distance	(1)	(8)	(4)	(3)	**(2)**	(13)	(12)	(9)	(7)	(11)	(5)	(10)	(6)
CVaR$^-$ distance	(1)	(11)	(13)	(5)	(4)	(12)	(3)	(8)	(7)	(10)	(9)	**(2)**	(6)
Tail dependence coefficient$^+$	(11)	(6)	(2)	**(1)**	(3)	(10)	(6)	(4)	(7)	(9)	(13)	(12)	(8)
Tail dependence coefficient$^-$	(3)	(6)	(2)	(4)	(11)	(12)	(5)	(8)	(9)	(10)	(8)	(13)	**(1)**
Return level distance$^+$	(13)	(3)	(2)	(5)	(4)	(12)	(8)	(11)	(9)	(10)	**(1)**	(6)	(7)
Return level distance$^-$	(13)	(11)	(12)	(6)	(5)	(8)	(4)	(3)	(7)	(9)	(10)	**(1)**	(2)
Total	(48)	(100)	(85)	**(56)**	(69)	(129)	(66)	(92)	(73)	(117)	(116)	(72)	(68)

Table 11: Results of the rank of the different models for Wind

At a quick glance it can be seen how the results for the wind series are quite different to those of the other two series. In this case, a different transformer based model – the FEDFormer – is the best performer. However, this overall result is highly skewed by the extreme value prediction metric, where this model does greatly. In the rest of the categories it also does quite good but not comparatively as much. The iTransformer on the other hand drops quite a few positions in the overall ranking, doing not very badly but not being nearly as dominant, with the general distribution being its best category. The TFT and the Informer are again some of the worst performing models.

In this case, the VARMAX is surprisingly the best model. It is the first series in which this model outperforms the SARIMAX model, most surely due to the integration component not being significant in this series due to its much higher stationarity. The results of the whole statistical category are however quite surprising, as it would seem by looking at the VARMAX that the simplicity of the model were an advantage, preventing it from capturing spurious patterns more present in the wind series. However, the other two simple models are significantly worse performers.

Another model with a great improvement in performance is the SVM, for the first time ranking among the top models although not standing out specially in any given category. Again, the XGBoost is the worst performing model.

The N-BEATS and TimeMixer models are again doing very good, with the N-HiTS maintaining its drop in performance in this complex series.

As for the transformers, it would seem like they could maybe benefit from their long run complex pattern capabilities for this series, however as it has already been seen in this series simplicity seems to be the way to go. The iTransformer is again the best out of all transformer based models, followed by the FEDFormer. This latter model is specially good at modelling the lower tail of the distribution. Although it does significantly worse for the upper tail.

One final difference for this series is how worse all models are compared to the benchmark relatively to the other two series. No matter what metric one looks at, the gap between the best model and the benchmark is significantly higher than for the other two series, hinting at the existence of unexploited patterns or modeling criteria that are not being leveraged and that thus do not allow the models to represent the wind series as realistically.

Finally, regarding the evaluation method itself, it is worth noting how for example all the models including the benchmark have a very significant negative Return level distance $^-$. What this is showing is that the return level for the lower tail is greatly underestimated by all models. If all models including the benchmark show the same bias, it could show that there may be a problem with the data used for evaluation itself, where the lower tail distribution was not representative and thus was underestimated by all models.

The comparison seen in Figure 19 is by far the most discouraging one. It can be seen how the modeled profile distribution has nothing to do with the actual one. The variance in the modeled data is much lower than the real one, with all of the data being concentrated around a yearly seasonality which only captures very basic yearly and daily profiles – with the daily ones being present only in the summer, but the model showing them for the whole year.

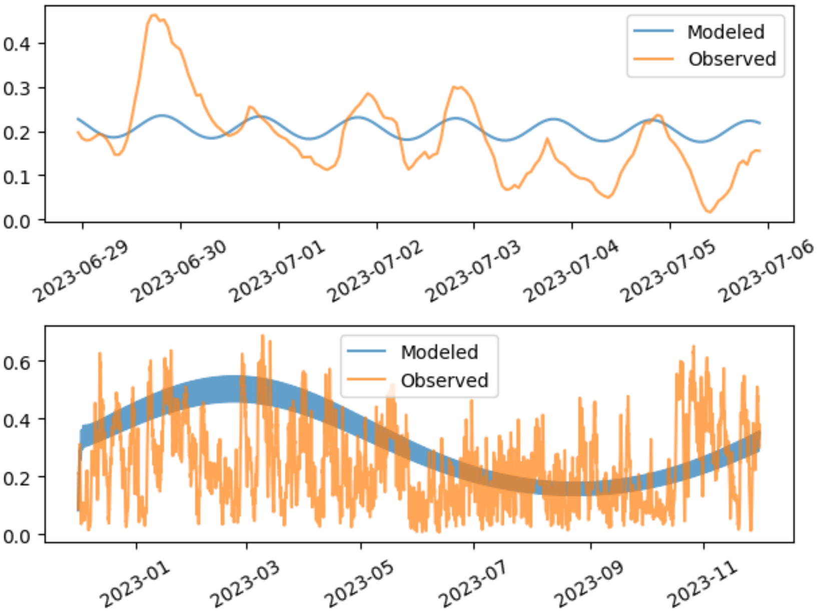

Figure 19: Weekly and yearly predicted and actual wind capacity factor for the VARMAX model

It is not very promising that this is the best model that could be found, although using other models not as constrained by the seasonality and simplicity of the VARMAX does not show much improvement as seen in Figure 20. The same crowding of the predicted data around the mean can be seen, with the model being unable to create data with the appropriate variance.

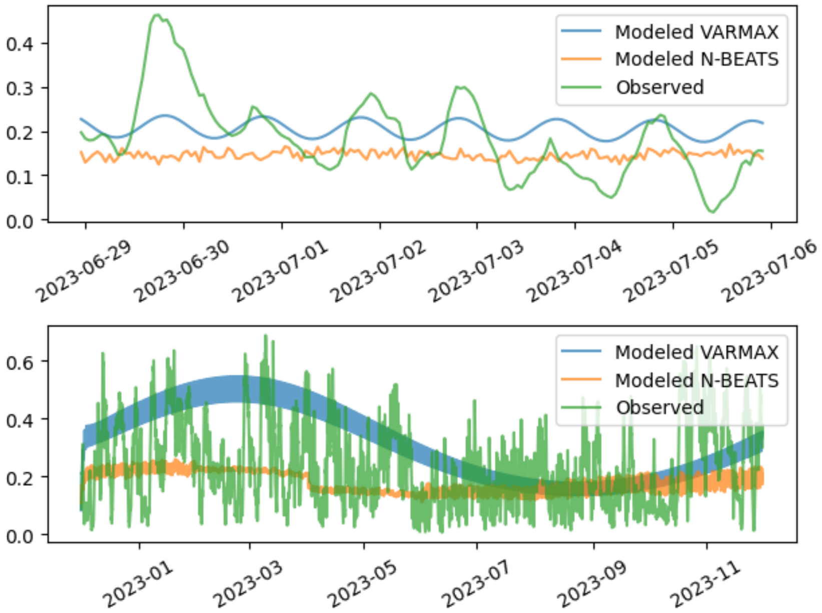

Figure 20: Weekly and yearly predicted and actual wind capacity factor for the VARMAX and N-BEATS model

This does shed some light on some necessary changes for future works. It is clear that the wind capacity factor has a really prominent random component and a low signal-to-noise ratio. The loss functions that have been used, which are mainly MAE and SE as they are the standard for these type of tasks, make the model extremely risk averse in their predictions and when they cannot accurately predict the wind values they end up predicting a kind of mean. For more realistic modeling of wind data it seems like different loss functions that don't penalize point errors that much need to be used.

5.4 Custom loss function

After having analyzed the results for all models in the three capacity factor series, it is now time to analyze the results regarding the hypothesis of the custom loss function. That is, can the behavior of the model be steered through a custom loss function – in this case the SERA loss – to better predict extreme values? As explained in the **??** section, an XGBoost model has been trained using this loss function with the same hyperparameters as the base XGBoost model. In fact, two such models have been implemented, one focusing on the upper tail extreme values and one focusing on the lower tail extreme values. The relevance function in both cases has been to give a relevance of 1 to the top or bottom 10%, a relevance of 0 to the opposite 10% and the rest of the distribution is given a relevance based on the spline interpolation.

Note that the modeling has been done using the custom loss only for the wind modeling, given this is the series where there is the highest practical interest to accurately predict extreme events. The results are as follows.

Wind	XGBoost	XGBoost$^-$	XGBoost$^+$
Cramer von Mises	774.6	1332	1978
KL divergence	0.3214	0.6029	1.678
ACF$_\xi$ distance	0.6625	0.6622	0.2574
CCMD	0.3549	0.3584	0.4519
CCF$_\xi^{SolarPV}$ distance	0.4045	0.3901	0.3306
CCF$_\xi^{SolarTH}$ distance	0.525	0.5031	0.4071
CVaR$^+$ distance	-0.5748	-0.4043	-0.235
CVaR$^-$ distance	5.746	8.46	11.03
Tail dependence coefficient$^+$	0.05028	0.05008	0.05974
Tail dependence coefficient$^-$	0.05003	0.04982	0.01963
Return level distance$^+$	-1.081	-1.206	-0.81
Return level distance$^-$	-5.709	-6.974	-8.092

Table 12: Results of the XGBoost models with SERA loss functions for Wind

Even though these results are much easier to read and interpret than the ones where all the models are included, the rank will still be provided in order to help interpret the results as much as possible.

Wind	XGBoost	XGBoost$^-$	XGBoost$^+$
Cramer von Mises	**(1)**	(2)	(3)
KL divergence	**(1)**	(2)	(3)
ACF$_\xi$ distance	(3)	(2)	**(1)**
CCMD	**(1)**	(2)	(3)
CCF$_\xi^{SolarPV}$ distance	(3)	(2)	**(1)**
CCF$_\xi^{SolarTH}$ distance	(3)	(2)	**(1)**
CVaR$^+$ distance	(3)	(2)	**(1)**
CVaR$^-$ distance	**(1)**	(2)	(3)
Tail dependence coefficient$^+$	(2)	(3)	**(1)**
Tail dependence coefficient$^-$	**(1)**	(2)	(3)
Return level distance$^+$	(2)	(3)	**(1)**
Return level distance$^-$	**(1)**	(2)	(3)
Total	**(22)**	(26)	(24)

Table 13: Results of the XGBoost models' rank with SERA loss functions for Wind

The first insight that can be obtained from these results is that, unsurprisingly, the unbiased XGBoost model seems to be the one best representing the general distribution of the series. This is to be expected as the other two models will probably have introduced some bias in order to better fit extreme values. However, what is surprising is how the upper tail focused model performs better in all metrics regarding temporal consistency both for the series with itself and with the other two series. This probably means that higher wind capacity factor values are more related to previous wind and solar values than the lower ones.

Interestingly, this holds not only for the self temporal correlation but also for the cross temporal correlation, with this model being also the best one regarding the correlation of the wind values with lagged values of the other two series.

Regarding extreme values, which is actually the whole purpose of introducing the custom loss function, a couple interesting observations can be made. Introducing the SERA loss seems to correctly steer the model into correctly modeling the upper tail of the wind capacity factor. However, this result cannot be replicated for the lower tail and the unbiased model is actually the one with the best performance in the lower extreme values.

6 Conclusion

This work has set out to find out which models are the best suited for long term modeling of solar PV, solar TH and wind hourly capacity factor. This aims to further the literature for long term modeling – given the usual focus on short or medium term for most papers – in order to better aid electrical grid planners and operators to integrate more solar and wind energy resources. In order to do this, 12 different models belonging to 4 different model families are implemented and evaluated along 9 different metrics. These metrics aim to evaluate different aspects of the modeled data: overall distribution fit, temporal consistency, multivariate coherence and extreme value fit.

After a careful analysis it has become apparent how different models are better suited for different variables and different characteristics of such variables. For the solar PV capacity factor the N-HiTS emerges as the best overall model due to its ability to best capture and leverage frequency components, which are the main driving factor of this series. As for the solar thermal capacity factor, due to its more complex self and cross dependence it is best modeled by the iTransformer, which thanks to its architecture is better suited for capturing more complex and long run patterns. The wind series is the most different to the other two. Due to its low signal-to-noise ratio this series is best modeled by a simple statistical model, the VARMAX. This simple statistical model however is able to leverage multivariate information for modeling wind data. The NN based models N-BEATS and TimeMixer – very good performers for the previous two series as well – are however the ones that best capture cross-variable relationships for this series.

The work also demonstrates the usefulness of implementing the SERA as a custom loss function for extreme value modeling in the wind series. This loss function properly steers a sample XGBoost model to better predict extreme wind events in the upper tail of the distribution at the expense of worse overall distribution fit. As an unintended side effect, the temporal consistency metrics also improved for the upper tail biased model. These results are however not replicated for the lower tail of the distribution, where the base model remains the best predictor for extreme values.

This work also highlights the shortcomings of the implemented models in simulating wind data. The performance of the best model is comparatively much worse than that of the best model of the solar PV and solar TH series. After close analysis, it is hypothesized that the gap in performance could be improved by choosing a loss function that does not penalize incorrect point wise predictions so harshly.

Appendices

A Code implementation

The code used in this work has been made publicly accessible. That is, the implementation of the different models, but also all the code used to train and evaluate them, the preliminary study of the datasets, etc. Even the latex version of this written document has been made accessible. All that code can be accessed via GitHub at https://github.com/fcelya/solar-wind-generation. The only elements that have not been made accessible are the datasets used to run the strategy and to estimate the factor parameters. They could not be uploaded due to GitHub's limitations, and enough information has been given in section 3 for anyone to be able to recreate those datasets.
The hope is that by making this code easily accessible, more people will be tempted to explore it, play with it, tackle the problem of long term solar and wind energy modeling and the ideas presented in this work can be further explored by the community at large.

Some code used to generate the results in this work is not present in the mentioned repository. Due to the computing power needed to train the neural network and transformer based models, these models have been trained in Google Colab and Kaggle Notebook environments, leveraging their freely accessible GPU and TPU infrastructure. For many of these neural network and transformer based models, they have been implemented through the open source library NeuralForecast created by Nixtla. For more information refer to Olivares et al., 2022.

B SDG alignment

We hope this work can also help in the advancement of the Sustainable Development Goals as presented by the United Nations in 2015. In particular, this work aims to be specially helpful in advancing the following goals.

- **(SDG 7) Affordable and clean energy**: By modeling and forecasting solar and wind capacity factor – and therefore generation – this work aims to contribute to better integrate renewable energy into the Spanish electrical grid and elsewhere if applicable. Accurate modeling can help in planning infrastructure investments, optimizing resource allocation and reducing fossil fuel reliance, which should drive up affordability and accessibility to clean energy.
- **(SDG 11) Sustainable cities and communities**: Renewable electricity is a key factor in the sustainability transition for cities and communities. Long-term modeling is crucial in ensuring a stable electricity supply for these communities by helping private and public planners make more accurate decisions.
- **(SDG 9) Industry, innovation and infrastructure**: By leveraging some of the latest models in long term time series forecasting, this work aims to bring an innovative approach to modernizing the electrical infrastructure in Spain.
- **(SDG 13) Climate action**: By facilitating the adoption of renewable energy, this work supports the sustainability transition to low carbon energy, directly fighting one of the main drivers of climate change. Long term renewable energy modeling is a very valuable tool for private and public players in reducing greenhouse gas emissions and meeting climate targets.
- **(SDG 8) Decent work and economic growth**: By improving the reliability of the energy infrastructure, this work aims to also help foster economic growth mainly but not only in industrial sectors, which in turn should help create employment opportunities directly in the energy sector and also in all of those helped by having a more robust and affordable electrical grid.

References

Abdel-Nasser, M., & Mahmoud, K. (2017). Accurate photovoltaic power forecasting models using deep lstm-rnn. *Neural Computing and Applications*, *31*(7), 2727–2740. https://doi.org/https://doi.org/10.1007/s00521-017-3225-z

Abuella, M., & Chowdhury, B. (2016). *Solar power forecasting using support vector regression*. https://arxiv.org/pdf/1703.09851

Agency, I. E. (2021). *Spain 2021 energy policy review*. https://iea.blob.core.windows.net/assets/2f405ae0-4617-4e16-884c-7956d1945f64/Spain2021.pdf

Ahmed, S. D., Al-Ismail, F. S. M., Shafiullah, Al-Sulaiman, F. A., & El-Amin, I. M. (2020). Grid integration challenges of wind energy: A review. *IEEE Access*, *8*, 10857–10878. https://doi.org/https://doi.org/10.1109/access.2020.2964896

Aksoy, N., & Genc, I. (2023). Predictive models development using gradient boosting based methods for solar power plants. *Journal of Computational Science*, *67*, 101958–101958. https://doi.org/https://doi.org/10.1016/j.jocs.2023.101958

Aly, H. H. (2020). A novel deep learning intelligent clustered hybrid models for wind speed and power forecasting. *Energy*, *213*, 118773–118773. https://doi.org/https://doi.org/10.1016/j.energy.2020.118773

Alzahrani, A., Shamsi, P., Dagli, C., & Ferdowsi, M. (2017). Solar irradiance forecasting using deep neural networks. *Procedia Computer Science*, *114*, 304–313. https://doi.org/https://doi.org/10.1016/j.procs.2017.09.045

Bouzerdoum, M., Mellit, A., & Pavan, A. M. (2013). A hybrid model (sarima–svm) for short-term power forecasting of a small-scale grid-connected photovoltaic plant. *Solar Energy*, *98*, 226–235. https://doi.org/https://doi.org/10.1016/j.solener.2013.10.002

Challu, C., Olivares, K. G., Oreshkin, B. N., Garza Ramirez, F., Mergenthaler Canseco, M., & Dubrawski, A. (2023). Nhits: Neural hierarchical interpolation for time series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, *37*(6), 6989–6997. https://doi.org/https://doi.org/10.1609/aaai.v37i6.25854

Chatterjee, S. (2020). A new coefficient of correlation. *Journal of the American Statistical Association*. https://doi.org/https://doi.org/10.1080//01621459.2020.1758115

Chen, J., Zhu, Q., Li, H., Zhu, L., Shi, D., Li, Y., Duan, X., & Liu, Y. (2019). Learning heterogeneous features jointly: A deep end-to-end framework for multi-step short-term wind power prediction. *IEEE Transactions on Sustainable Energy*, *11*(3), 1761–1772. https://doi.org/https://doi.org/10.1109/tste.2019.2940590

Chen, T., & Guestrin, C. (2016). Xgboost. *arXiv (Cornell University)*. https://doi.org/https://doi.org/10.1145/2939672.2939785

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297. https://doi.org/https://doi.org/10.1007/bf00994018

Cramér, H. (1928). On the composition of elementary errors. *Scandinavian Actuarial Journal*, *1928*(1), 13–74. https://doi.org/10.1080/03461238.1928.10416862

Delgado, C. J., Alfaro-Mejía, E., Manian, V., O'Neill-Carrillo, E., & Andrade, F. (2024). Photovoltaic power generation forecasting with hidden markov model and long short-term memory in miso and siso configurations. *Energies*, *17*(3), 668–668. https://doi.org/https://doi.org/10.3390/en17030668

Einstein, A. (1905). Über einen die erzeugung und verwandlung des lichtes betreffenden heuristischen gesichtspunkt. *Annalen der Physik*, *322*(6), 132–148. https://doi.org/https://doi.org/10.1002/andp.19053220607

Ellabban, O., Abu-Rub, H., & Blaabjerg, F. (2014). Renewable energy resources: Current status, future prospects and their enabling technology. *Renewable and Sustainable Energy Reviews*, *39*, 748–764. https://doi.org/https://doi.org/10.1016/j.rser.2014.07.113

Fernández, L. (2024). https://www.statista.com/statistics/1253518/clean-energy-investments-spain/

Frimane, Â., Munkhammar, J., & van der Meer, D. (2022). Infinite hidden markov model for short-term solar irradiance forecasting. *Solar Energy*, *244*, 331–342. https://doi.org/https://doi.org/10.1016/j.solener.2022.08.041

Glenk, G., & Reichelstein, S. (2022). The economic dynamics of competing power generation sources. *Renewable and Sustainable Energy Reviews*, *168*, 112758–112758. https://doi.org/https://doi.org/10.1016/j.rser.2022.112758

Granger, C. W. J. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, *37*(3), 424–438. https://www.jstor.org/stable/1912791

Haddad, M., Nicod, J.-M., Mainassara, Y. B., Rabehasaina, L., Al Masry, Z., & Pera, M.-C. (2019). Wind and solar forecasting for renewable energy system using sarima-based model. https://publiweb.femto-st.fr/tntnet/entries/16409/documents/author/data

He, X., Nie, Y., Guo, H., & Wang, J. (2020). Research on a novel combination system on the basis of deep learning and swarm intelligence optimization algorithm for wind speed forecasting. *IEEE Access*, *8*, 51482–51499. https://doi.org/https://doi.org/10.1109/access.2020.2980562

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*(5), 359–366. https://doi.org/https://doi.org/10.1016/0893-6080(89)90020-8

Hossain, M. S., & Mahmood, H. (2020). Short-term photovoltaic power forecasting using an lstm neural network and synthetic weather forecast. *IEEE Access*, *8*, 172524–172533. https://doi.org/https://doi.org/10.1109/access.2020.3024901

Hu, Q., Zhang, R., & Zhou, Y. (2015). Transfer learning for short-term wind speed prediction with deep neural networks. *Renewable Energy*, *85*, 83–95. https://doi.org/https://doi.org/10.1016/j.renene.2015.06.034

Huang, C.-J., & Kuo, P.-H. (2019). Multiple-input deep convolutional neural network model for short-term photovoltaic power forecasting. *IEEE Access*, *7*, 74822–74834. https://doi.org/https://doi.org/10.1109/access.2019.2921238

Jain, S., Sanjay, N., Pawan, N., Arora, R., & Behera, P. (2022). Wind power forecasting using varmax. *2022 2nd International Conference on Technological Advancements* *in Computational Sciences (ICTACS)*. https://doi.org/https://doi.org/10.1109/ictacs56270.2022.9988181

Ju, Y., Sun, G., Chen, Q., Zhang, M., Zhu, H., & Rehman, M. U. (2019). A model combining convolutional neural network and lightgbm algorithm for ultra-short-term wind power forecasting. *IEEE Access*, *7*, 28309–28318. https://doi.org/https://doi.org/10.1109/access.2019.2901920

Kim, J., Obregon, J., Park, H., & Jung, J.-Y. (2024). Multi-step photovoltaic power forecasting using transformer and recurrent neural networks. *Renewable and Sustainable Energy Reviews*, *200*, 114479–114479. https://doi.org/https://doi.org/10.1016/j.rser.2024.114479

Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, *22*(1), 79–86. https://doi.org/https://doi.org/10.1214/aoms/1177729694

Kumar, V., Pandey, A. S., & Sinha, S. K. (2016). Grid integration and power quality issues of wind and solar energy system: A review. https://doi.org/https://doi.org/10.1109/iceteeses.2016.7581355

Liang, S., Nguyen, L., & Jin, F. (2018). A multi-variable stacked long-short term memory network for wind speed forecasting. *arXiv (Cornell University)*. https://doi.org/https://doi.org/10.1109/bigdata.2018.8622332

Lim, B., Arık, S. Ö., Loeff, N., & Pfister, T. (2020). Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, *37*(4), 1748–1764. https://doi.org/https://doi.org/10.1016/j.ijforecast.2021.03.012

Lin, K.-P., Pai, P.-F., & Ting, Y.-J. (2019). Deep belief networks with genetic algorithms in forecasting wind speed. *IEEE Access*, *7*, 99244–99253. https://doi.org/https://doi.org/10.1109/access.2019.2929542

Liu, H., Mi, X., Li, Y., Duan, Z., & Xu, Y. (2019). Smart wind speed deep learning based multi-step forecasting model using singular spectrum analysis, convolutional gated recurrent unit network and support vector regression. *Renewable Energy*, *143*, 842–854. https://doi.org/https://doi.org/10.1016/j.renene.2019.05.039

Liu, Y., Hu, T., Zhang, H., Wu, H., Wang, S., Ma, L., & Long, M. (2023). Itransformer: Inverted transformers are effective for time series forecasting. *arXiv.org*. https://arxiv.org/abs/2310.06625

Ministerio de Industria, E. y. T. (2015). https://www.boe.es/buscar/act.php?id=BOE-A-2015-10927

Ministerio de Industria, T. y. C. (2007). https://www.boe.es/buscar/act.php?id=BOE-A-2007-10556

Muljadi, E., Butterfield, C., Chacon, J., & Romanowitz, H. (2006). Power quality aspects in a wind power plant: Preprint. https://digital.library.unt.edu/ark:/67531/metadc878686/m2/1/high_res_d/875771.pdf

Mystakidis, A., Koukaras, P., Tsalikidis, N., Ioannidis, D., & Tjortjis, C. (2024). Energy forecasting: A comprehensive review of techniques and technologies. *Energies*, *17*(7), 1662–1662. https://doi.org/https://doi.org/10.3390/en17071662

Neo, Y. Q., Teo, T. T., Woo, W. L., Logenthiran, T., & Sharma, A. (2017). Forecasting of photovoltaic power using deep belief network, 1189–1194. https://doi.org/https://doi.org/10.1109/tencon.2017.8228038

Olivares, K. G., Challú, C., Garza, F., Canseco, M. M., & Dubrawski, A. (2022). NeuralForecast: User friendly state-of-the-art neural forecasting models. https://github.com/Nixtla/neuralforecast

Pedro, H. T., & Coimbra, C. F. (2012). Assessment of forecasting techniques for solar power production with no exogenous inputs. *Solar Energy*, *86*(7), 2017–2028. https://doi.org/https://doi.org/10.1016/j.solener.2012.04.004

Peng, Z., Peng, S., Fu, L., Lu, B., Tang, J., Wang, K., & Li, W. (2020). A novel deep learning ensemble model with data denoising for short-term wind speed forecasting. *Energy Conversion and Management*, *207*, 112524–112524. https://doi.org/https://doi.org/10.1016/j.enconman.2020.112524

Ray, B., Shah, R., Islam, M. R., & Islam, S. (2020). A new data driven long-term solar yield analysis model of photovoltaic power plants. *IEEE Access*, *8*, 136223–136233. https://doi.org/https://doi.org/10.1109/access.2020.3011982

Renewable electricity generation. (2024). Red Eléctrica. https://www.sistemaelectrico-ree.es/en/spanish-electricity-system/generation/renewable-electricity-generation

Sarmas, E., Dimitropoulos, N., Marinakis, V., Mylona, Z., & Doukas, H. (2022). Transfer learning strategies for solar power forecasting under data scarcity. *Scientific Reports*, *12*(1). https://doi.org/https://doi.org/10.1038/s41598-022-18516-x

Scott, D. W. (1979). On optimal and data-based histograms. *Biometrika*, *66*(3), 605–610. https://doi.org/https://doi.org/10.1093/biomet/66.3.605

Selim, M., Zhou, R., Feng, W., & Alam, O. (2020). Reducing error propagation for long term energy forecasting using multivariate prediction. In G. Lee & Y. Jin (Eds.), *Proceedings of 35th international conference on computers and their applications* (pp. 161–169). EasyChair. https://doi.org/10.29007/mbb7

Sharadga, H., Hajimirza, S., & Balog, R. S. (2020). Time series forecasting of solar power generation for large-scale photovoltaic plants. *Renewable Energy*, *150*, 797–807. https://doi.org/https://doi.org/10.1016/j.renene.2019.12.131

Shen, Y., Yao, W., Wen, J., He, H., & Jiang, L. (2018). Resilient wide-area damping control using grhdp to tolerate communication failures. *IEEE Transactions on Smart Grid*, *10*(3), 2547–2557. https://doi.org/https://doi.org/10.1109/tsg.2018.2803822

Silva, A., Ribeiro, R. P., & Moniz, N. (2022). Model optimization in imbalanced regression. *arXiv.org*. https://arxiv.org/abs/2206.09991

Sorkun, M. C., Paoli, C., & Incel, O. D. (2017). Time series forecasting on solar irradiation using deep learning. *IEEE*. https://ieeexplore.ieee.org/document/8266215

Steen, D., Goop, J., Göransson, L., Nursbo, S., Brolin, M., Sarasini, S., & Kåberger, T. (2014). *9 challenges of integrating solar and wind into the electricity grid*. https://core.ac.uk/download/pdf/70609781.pdf

Suresh, V., Janik, P., Guerrero, J. M., Leonowicz, Z., & Sikorski, T. (2020). Microgrid energy management system with embedded deep learning forecaster and combined optimizer. *IEEE Access*, *8*, 202225–202239. https://doi.org/https://doi.org/10.1109/access.2020.3036131

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. https://arxiv.org/pdf/1706.03762

Vittal, V., & Ayyanar, R. (2013). Grid integration and dynamic impact of wind energy. *SpringerLink*. https://doi.org/https://doi.org/10.1007-978-1-4419-9323-6

Wang, F., Xuan, Z., Zhen, Z., Li, K., Wang, T., & Shi, M. (2020). A day-ahead pv power forecasting method based on lstm-rnn model and time correlation modification under partial daily pattern prediction framework. *Energy Conversion and Management*, *212*, 112766–112766. https://doi.org/https://doi.org/10.1016/j.enconman.2020.112766

Wang, F., Zhang, Z., Liu, C., Yu, Y., Pang, S., Duić, N., Shafie-khah, M., & Catalão, J. P. (2018). Generative adversarial networks and convolutional neural networks based weather classification model for day ahead short-term photovoltaic power forecasting. *Energy Conversion and Management*, *181*, 443–462. https://doi.org/https://doi.org/10.1016/j.enconman.2018.11.074

Wang, G. C., Urquhart, B., & Kleissl, J. (2019). Cloud base height estimates from sky imagery and a network of pyranometers. *Solar Energy*, *184*, 594–609. https://doi.org/https://doi.org/10.1016/j.solener.2019.03.101

Wang, J., Guo, L., Zhang, C., Song, L., Duan, J., & Duan, L. (2020). Thermal power forecasting of solar power tower system by combining mechanism modeling and deep learning method. *Energy*, *208*, 118403–118403. https://doi.org/https://doi.org/10.1016/j.energy.2020.118403

Wang, L., Tao, R., Hu, H., & Zeng, Y.-R. (2020). Effective wind power prediction using novel deep learning network: Stacked independently recurrent autoencoder. *Renewable Energy*, *164*, 642–655. https://doi.org/https://doi.org/10.1016/j.renene.2020.09.108

Wang, L., Wu, J., Cao, Y., & Hong, Y. (2022). Forecasting renewable energy stock volatility using short and long-term markov switching garch-midas models: Either, neither or both? *Energy Economics*, *111*, 106056–106056. https://doi.org/https://doi.org/10.1016/j.eneco.2022.106056

Wang, S., Wu, H., Shi, X., Hu, T., Luo, H., Ma, L., Zhang, J., & Zhou, J. (2024). Timemixer: Decomposable multiscale mixing for time series forecasting. *The Twelfth International Conference on Learning Representations*. https://openreview.net/pdf?id=7oLshfEIC2

Wang, Y., Wu, H., Dong, J., Liu, Y., Long, M., & Wang, J. (2024). Deep time series models: A comprehensive survey and benchmark.

Yin, H., Ou, Z., Huang, S., & Meng, A. (2019). A cascaded deep learning wind power prediction approach based on a two-layer of mode decomposition. *Energy*, *189*, 116316–116316. https://doi.org/https://doi.org/10.1016/j.energy.2019.116316

Zang, H., Cheng, L., Ding, T., Cheung, K. W., Wang, M., Wei, Z., & Sun, G. (2020). Application of functional deep belief network for estimating daily global solar radiation: A case study in china. *Energy*, *191*, 116502. https://doi.org/https://doi.org/10.1016/j.energy.2019.116502

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., & Zhang, W. (2021). Informer: Beyond efficient transformer for long sequence time-series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, *35*(12), 11106–11115. https://doi.org/https://doi.org/10.1609/aaai.v35i12.17325

Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., & Jin, R. (2022). Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. https://arxiv.org/pdf/2201.12740

Zou, H., & Yang, Y. (2004). Combining time series models for forecasting. *International Journal of Forecasting*, *20*(1), 69–84. https://doi.org/https://doi.org/10.1016/s0169-2070(03)00004-9