Exploring improvements of wind power forecasts using Convolutional Neural Networks and Time Series Analysis

Jakob Nabialek

Master's thesis
2022:E37

Lund University

Faculty of Engineering
Centre for Mathematical Sciences
Mathematical Statistics

Centrum Scientiarum Mathematicarum

Acknowledgements

I would like to express my gratitude to the Physical Trading desk at Axpo's office in Malmö and especially Thomas Hamfelt. The guidance within the workings of the energy market, access to data and approaches to data manipulation were key to conducting this thesis.

I would also like to thank my supervisor Carl Hvarfner for the meaningful discussions and guidance within the concepts used in this thesis.

Abstract

Due to environmental considerations, volumes of renewable power production are rapidly growing, and its share of the energy pool is increasing. The intermittent nature of wind power, being one of the main renewable energy sources, is a challenge when generating production forecasts. Accurate forecasts are necessary for the electrical grid to be kept in balance as the development of wind power continues. Power traders have a great incentive in achieving low forecast errors to decrease their imbalance costs.

The basics of the energy market and wind power predictions are presented in this thesis. Thereafter, the possibilities of improving medium-term wind power forecasts are explored. Two separate models, based on the Kalman filter and Convolutional Neural Networks are implemented on historical production data with an existing black box model as a baseline comparison. The results are varying, and a conclusion is drawn that adding real-time production data as an input into the models can be beneficiary for the accuracy.

Contents

Chapter 1

Introduction

Electric power is a crucial commodity in many functional aspects of society. As many other commodities, power is traded on an exchange where the price is set by a function of supply and demand. In the Nordic countries that exchange is Nord Pool. The nature of electric power differs from other exchange traded commodities in the sense that it cannot be stored, in any significant amounts, and therefore has to be produced and consumed simultaneously [6]. This has many implications on the dynamics of the market [28] and therefore also on the needs of market participants [31].

As the share of renewable, and often intermittent, power sources is increasing in the energy pool; the market dynamics are becoming more dependent on their production patterns. Since 1997 the global wind power capacity has grown from 7.5 GW to 733 GW by 2020 [35]. Simultaneously the access to fossil fuels is becoming increasingly restrictive and costly [13, 30]. This has led to an increment of prices and their volatility in the regulating market [34]. Therefore, for actors in the energy market who are managing wind power production, forecasts of high quality are becoming increasingly important. There is also an interest to achieve this on a societal level as the market is favored by production levels being estimated sufficiently, allowing for production to be planned. In the mission to become independent of fossil fuels through the continual development of renewable energy solutions, forecasts of high quality are an important factor.

1.1 Objective

This thesis explores and implements possible improvements for wind power medium-term production forecasts used to trade on the spot market. These are forecasts made before 12:00 (mid-day) for each hour the following day.

1.2 Axpo Nordic

Historical data for the necessary analysis is made available to the author by Axpo Nordic. The company manages a portfolio partly consisting of wind power in the Nordic region. The power produced in these units is sold through Nord Pool's spot market. As Axpo Nordic is a balance responsible party (BRP), they bear the financial responsibility for the power sold to the market being equal to the power supplied. Therefore, Axpo Nordic uses forecasts to trade the managed wind power on the spot market.

1.3 Scope

Currently the forecasts are made using a state of the art black box model. This forecast will in this thesis be referred to as the baseline forecast. The inner workings of the model generating the baseline forecast are unknown and the model is therefore referred to as a black box.

The concepts used for exploring the possibility of improving the baseline forecasts are based on the Kalman filter as well as artificial neural networks. Two seperate models, based on the Kalman filter and convolutional neural networks are designed in this thesis. These are designed on and applied on historical data of a wind farm. The main input into these models is the baseline forecast but recent production data (real-time production data) will also be a factor. Generally, in existing literature and studies, recent production data is not used for wind power forecasts at longer time horizons than a few hours. This thesis explores the possibility of extending that window.

Chapter 2

Background

2.1 Power market

Energy trading on Nord Pool is done in 15 countries over 21 bidding zones. The bidding zones are geographical areas to which network constraints are applied. Sweden, for example, is split into four bidding zones: SE1, SE2, SE3 and SE4. More than 300 buyers and sellers place over 2000 orders daily on Nord Pool's spot market [24].

The market is divided into three sub-markets which are the spot market, intraday market and regulating market. Trading on the spot market is done daily on an hour basis for the 24 hours of the next day. At 10:00 CET available capacities of the grid are published, and bids can be submitted until 12:00 CET. Orders are thereafter matched and the price is set by a function of supply and demand with limitations of the grid imposed [24].

The intraday market allows market participants to trade closer to the time of delivery. It is a continuous market where trading takes place until one hour before delivery. The demand for such a market stems from energy production and consumption which cannot be accurately planned or forecasted at the time of the spot bids.

It is the ultimate responsibility of the transmission system operator (TSO) to keep the system in balance. That is done through regulating production and consumption with the aim to keep the frequency of the system at a constant level. In the Nordics the frequency is kept at 50 Hz ($\pm 0, 1$). Imbalances between power sold/bought, on the spot and intraday market, and delivered/consumed occur. These imbalances lead to the TSO regulating production and/or consumption. The cost of the regulation is thereafter settled between the TSO and the BRP in the regulating market. It is therefore in the interest of the BRP to keep the imbalances low.

2.2 Production types

Power production types are broadly divided into the categories of intermittent, dispatchable and base load power. Base power consists of power production types which cannot change their power output quickly, such as nuclear or coal power plants. Historically base load sources covered the minimum level that an electric grid demanded over time [2]. These are partly being replaced by intermittent power sources as that is the nature of many renewable power sources. Intermittent power cannot be planned as the sources produce sporadically. Among these are, for example wind and solar power. Dispatchable power sources have short startup times and their production can be regulated on demand. The demand for dispatchable power sources is driven by the need keep the system in balance. There are many types of dispatchable power sources, both renewable and conventional, as for example natural gas turbines and hydroelectric facilities [14]. Some production types such as hydro electrical power can serve as both base load and dispatchable power. Such is the function of hydro electrical power the Swedish grid, which has a similar load to nuclear power as presented in Figure 2.1.

	Characteristics	Examples
Base load	Power sources that have long start up and shut down times. They can cover the minimum level the grid demands over time.	Coal, nuclear, geothermal and combined cycle.
Intermittent	Sources that produce sporadically.	Wind and solar
Dispatchable	Sources that have short start up and shut down times. Used to regulate the balance in the system.	Grid batteries, hydroelectric and natural gas

Table 2.1: Production types and their characteristics

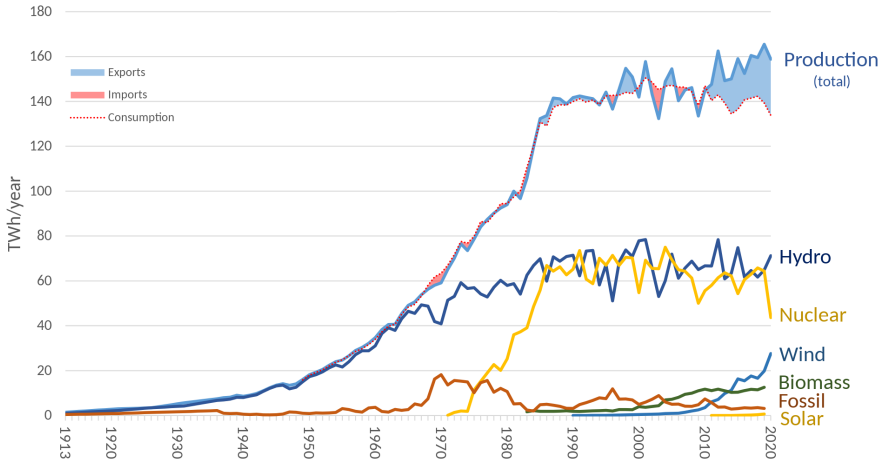

Figure 2.1: Electricity production in Sweden [37]

2.3 Wind power prediction

As wind power is an intermittent source of power, there is a challenge in creating accurate predictions of production. The production levels are characterized by their stochastic nature [4] with high volatility [7]. They are a factor of multiple weather-related factors such as wind speed, temperature and pressure [9]. Most models are therefore based on using numerical weather prediction (NWP) as input [15].

There are multiple methods used to transfer these inputs into predicting power output [15], which are presented in this section. To minimize the imbalance costs on the regulating market, market participants have an incentive to increase the quality of their forecasts.

In the literature, wind power forecasting is oftentimes categorized on the basis of methodology applied as well as the predictive time-horizon. The methodology is divided into the categories of physical and statistical methods. The time-horizons are divided into the categories [33] shown in Table 2.2.

Horizon	Range	Usage
Very short-term	Few seconds to 30 minutes	- Regulation Actions
Short-term	30 minutes to 6 hours	- Economic Load Dispatch Planning - Load Increment/Decrement Decisions
Medium-term	6 hours to 1 day	- Generator Online/Offline Decisions - Operational Security in Spot Electricity Market
Long-term	1 day to 1 month	- Unit Commitment Decisions - Reserve Requirement Decisions - Maintenance Scheduling to Obtain Optimal Operating Cost

Table 2.2: Horizon categories for wind power forecasts [33]

2.3.1 Physical methods

The physical methods are based upon using a model of the physical description of a wind turbine and its surrounding [15]. The physical model for the turbine is oftentimes provided by the turbine manufacturer. The main obstacle to using such model is to accurately predict wind speeds at the location of the turbine. Since wind farms are not necessarily located at the precise locations where meteorological forecasts are made, the data needs to be approximated or extrapolated to the desired location and turbine height. This includes modelling for obstacles which can affect the wind speed. The predicted wind speed at the location and height of the turbine is thereafter transformed into power output. This can be represented by a power curve where a certain power output is expected at a certain wind speed as seen in Figure 2.2.

Physical methods perform well in the equivalent to short, medium-term and long-term forecasting [26, 4]. For shorter time horizons recent production data becomes more important [15, 26].

An advantage the physical methods hold is that there is no need for historical production data. This method is therefore useful for newly built wind farms.

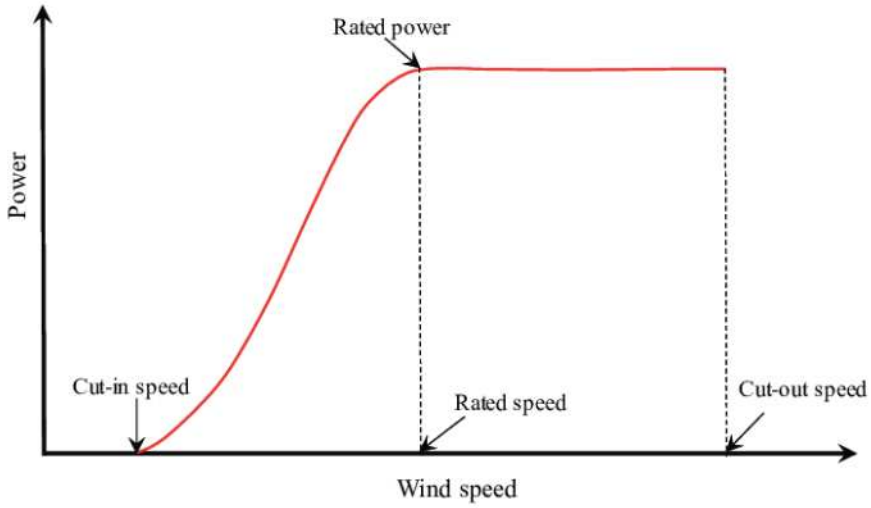

Figure 2.2: Power curve of a wind turbine. Cut-in speed refers to the wind speed at which the power plant starts to produce power. Rated speed and power refer to the wind speed respectively production output where the power plant reaches maximum power. Cut-out speed refers to the wind speed at which the power generation is stopped. [29]

2.3.2 Statistical methods

Statistical methods differ from the physical methods in the sense that they do not seek to find the physically described relation in the conversion of the modelled wind speed into power output. The relation between inputs and outputs is sought through training a model on historical data. Inputs may vary but these are generally based on meteorological data and NWPs. The advantage to this method is that wind speed does not have to be interpolated to the specific site of the wind turbine [1, 15]. Recent production data, if available, can also be taken into account to add an autoregressive factor to the model. Oftentimes it is not possible to access the current production of a wind farm and an autoregressive/moving average approach is therefore not always applicable.

An example would be that a prediction for time {t+k} $\hat{p}_{t+k|t}$ considers NWP data for up to time {t+k} and measurements, {M}, for up to time t. The function f is a generic function dependent on specific method used.

$$\hat{p}_{t+k|t} = f\left(\{NWP_\tau\}_{\tau=t}^{t+k},\ \{\mathrm{M}\}_{\tau=0}^{t}\right) \tag{2.1}$$

There are multiple approaches for this method. Models can for example be based on ARMA-models [5] as well as artificial neural networks [19]. Simpler models are generally used for shorter time horizons where recent power output

data can be a predictive factor. This includes the autoregressive model (AR), the moving average model (MA) and the autoregressive moving average model (ARMA). They have the following forms:

AR(p):

$$X_t = c + \sum_{i=1}^{p} \varphi_i X_{t-i} + \varepsilon_t \tag{2.2}$$

MA(q):

$$X_t = \mu + \varepsilon_t + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} \tag{2.3}$$

ARMA(p,q):

$$X_t = c + \varepsilon_t + \sum_{i=1}^{p} \varphi_i X_{t-i} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} \tag{2.4}$$

Where c is a constant, φ and θ model parameters, ε white noise, μ the expectation of X_t, p is the order of the autoregressive model and q is the order of the moving average model.

For time horizons longer than short-term, the more advanced models are preferred as they allow for the handling of more inputs than recent production data. This is possible using more complex time series models such as the Kalman filter. Artificial neural networks can also be used to design such model. The theory behind these methods will be explained in the next chapter.

2.4 Related work

A lot of previous work has been done in the area of wind power prediction. The Kalman filter is applied in many papers [3, 21, 17] for different prediction horizons. The models in these studies mainly use NWPs as input for prediction generation further ahead than the short-term horizon.

Most work regarding usage of convolutional neural networks (CNN) in wind power forecasting is rather new and comparatively good results are achieved [10, 11]. The area of using CNNs for time series analysis in general has grown recently. These models perform well with the explanation [10, 11] that the CNN is good at handling large sets of data. This is particularly important when using NWP as input, which is not the case in this thesis as the baseline forecast incorporates such factors. The 1D CNN can be used to generate predictions in an approach inspired by the physical methods where the specific wind speeds and directions are calculated [10]. ANNs in general can also be used to create models based on statistical methods [11] which is more relevant for this thesis.

Some models [11] incorporate multiple machine learning algorithms for different modelling tasks.

Furthermore, the CNN developed in this thesis allows for the rate of the output signal to be the same as that of the input signal. Related work [23] shows that such models are effective in the sense of performance against difficulty to train.

Chapter 3

Model theory

3.1 Kalman filter

The Kalman filter [12] is an algorithm that can estimate the state of a system based on measurements over time. As it is a dynamic model, it can accurately make predictions for systems which change over time. It can also handle multiple inputs. Therefore, it is a suitable model for prediction of wind power production which tends to display this kind of behavior.

The estimated dynamics of the current state of a system can be expressed using the state-space representation, which is a model of a system based on input, output and state variables. The Kalman filter can be a tool to estimate the unknown state variables, such as future variables which then can be used to generate predictions based on inputs. This is done using the observations of the input and output of the system. The linear state-space representation can be expressed as in Eq. 3.1 and 3.2:

$$\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t + \mathbf{e}_t \tag{3.1}$$

$$\mathbf{y}_t = \mathbf{C}\mathbf{x}_t + \mathbf{w}_t \tag{3.2}$$

where $\mathbf{y}_t$ is the m-dimensional measurement vector at time t and $\mathbf{x}_t$ is the internal n-dimensional state vector. The matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ are known matrices of appropriate dimensions. The k-dimensional input vector $\mathbf{u}_t$ contains all measurable inputs to the system, whereas $\mathbf{e}_t$ and $\mathbf{w}_t$ denote the n-dimensional process noise and the m-dimensional measurement noise, respectively.

To form the Kalman filter [12], let us introduce

$$\mathbf{Y}_t = \begin{bmatrix} \mathbf{y}_1^T & \cdots & \mathbf{y}_t^T \end{bmatrix}^T \tag{3.3}$$

which is a vector including observations up to time t. The optimal prediction of the state vector at time $t + k$, given $\mathbf{Y}_t$, can be formed as

$$\hat{\mathbf{x}}_{t+k|t} = \mathbf{E}\left\{\mathbf{x}_{t+k} \mid \mathbf{Y}_t\right\} \tag{3.4}$$

where $\hat{\mathbf{x}}_{t+k|t}$ is the k-step prediction of the state vector x_t. Then the prediction errors are defined as

$$\tilde{\mathbf{x}}_{t+k|t} = \mathbf{x}_{t+k} - \hat{\mathbf{x}}_{t+k|t} \tag{3.5}$$

$$\tilde{\mathbf{y}}_{t+k|t} = \mathbf{y}_{t+k} - \hat{\mathbf{y}}_{t+k|t} = \mathbf{y}_{t+k} - \mathbf{C}\hat{\mathbf{x}}_{t+k|t} \tag{3.6}$$

The optimal linear reconstruction, $\hat{\mathbf{x}}_{t|t}$, and one-step prediction, $\hat{\mathbf{x}}_{t+1|t}$, of the vector x_t for the linear state space representation can be computed recursively using

$$\hat{\mathbf{x}}_{t|t} = \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t \left(\mathbf{y}_t - \mathbf{C}\hat{\mathbf{x}}_{t|t-1}\right) \tag{3.7}$$

$$\hat{\mathbf{x}}_{t+1|t} = \mathbf{A}\hat{\mathbf{x}}_{t|t} + \mathbf{B}\mathbf{u}_t \tag{3.8}$$

where the Kalman gain, $\mathbf{K_t}$ is formed as

$$\mathbf{K}_t = \mathbf{R}^{x,x}_{t|t-1}\mathbf{C}^T \left[\boldsymbol{R}^{y,y}_{t|t-1}\right]^{-1} \tag{3.9}$$

with

$$\begin{aligned} \mathbf{R}^{x,x}_{t|t} &= \mathbf{R}^{x,x}_{t|t-1} - \mathbf{K}_t\mathbf{R}^{y,y}_{t|t-1}\mathbf{K}_t^T \\ &= \mathbf{R}^{x,x}_{t|t-1} - \mathbf{K}_t\mathbf{C}\mathbf{R}^{x,x}_{t|t-1} \\ &= \left(\mathbf{I} - \mathbf{K}_t\mathbf{C}\right)\mathbf{R}^{x,x}_{t|t-1} \end{aligned} \tag{3.10}$$

$$\mathbf{R}^{x,x}_{t+1|t} = \mathbf{A}\mathbf{R}^{x,x}_{t|t}\mathbf{A}^T + \mathbf{R}_e \tag{3.11}$$

$$\mathbf{R}^{y,y}_{t+1|t} = \mathbf{C}\mathbf{R}^{x,x}_{t+1|t}\mathbf{C}^T + \mathbf{R}_w \tag{3.12}$$

As initial conditions one should select

$$\hat{\mathbf{x}}_{1|0} = E\left\{\mathbf{x}_1\right\} = \mathbf{m}_0 \tag{3.13}$$

$$\mathbf{R}^{x,x}_{1|0} = V\left\{\mathbf{x}_1\right\} = \mathbf{V}_0 \tag{3.14}$$

This thesis will be working with predicting further in time than one step ahead. The Kalman filter can be modified to compute multi-step predictions recursively the following way:

$$\hat{\mathbf{x}}_{t+k+1|t} = \mathbf{A}\hat{\mathbf{x}}_{t+k|t} + \mathbf{B}\hat{\mathbf{u}}_{t+k|t} \tag{3.15}$$

$$\mathbf{R}^{x,x}_{t+k+1|t} = \mathbf{A}\mathbf{R}^{x,x}_{t+k|t}\mathbf{A}^T + \mathbf{B}\mathbf{R}^{u,u}_{t+k|t}\mathbf{B}^T + \mathbf{R}_e \tag{3.16}$$

3.2 Artifical neural networks

Among the more advanced statistical models for wind power forecasting, many different types of artificial neural networks (ANN) have been researched on [18, 8, 20, 16]. These are for example the multilayer perceptron [18], wavelet neural network [8], Elman neural network (ENN) [20], long-short-term memory (LSTM) [20] and convolutional neural network (CNN) [16].

ANNs are a subset of machine learning whose intention is to imitate how the brain uses synapses and neurons to process information. ANNs can be characterized as consisting of an input layer, one or more hidden layers and

an output layer. These layers consist of one or more nodes. An ANN with such design is called a multilayer perceptron. This architecture allows for the modelling of complex non-linear relationships through the learning and training process.

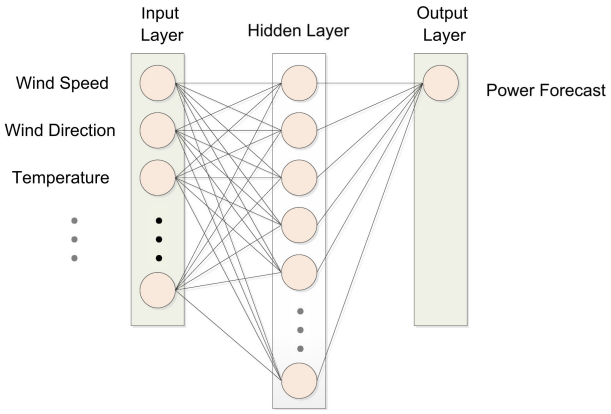

Figure 3.1: ANN approach for wind speed and power forecasts, adapted from the original source [15]

Figure 3.1 represents a fully connected feed-forward network. A fully connected network connects all nodes to each other between the layers. This thesis exclusively studies feed-forward networks where connections between nodes do not form a cycle. The output y (Power Forecast), in the case of the network in Figure 3.1, is given by Eq. 3.17.

$$y = \varphi_o \left(\sum_{j=0}^{J} w_j \varphi_h \left(\sum_{k=0}^{K} \tilde{w}_{jk} x_k \right) \right) \tag{3.17}$$

where x_k is input k, φ_o respectively φ_h the activation function for the output and hidden layer and w respectively $\tilde{w}$ weight for each node. The activation function can consist of various functions depending on the problem being solved. It is typically a non-linear function, such as the Rectified Linear Unit (ReLU) (Eq. 3.18), resulting in a transformed output value. Each neuron computes an output which depends on the input and the weight expresses the importance of respective inputs. The weights are set through the training process.

$$f(x) = x^+ = \max(0, x) \tag{3.18}$$

The network learns patterns and relations between the input and output by minimizing a loss function. In regression related problems a loss function commonly used is the mean squared error (MSE) which calculates the average

of the squares of the errors (Eq. 3.19).

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3.19}$$

The minimization of the loss function is done by updating the weights in the network. The training process is an optimization problem. A commonly used optimization algorithm is gradient descent learning. It works through first initiating the weights with small random numbers and a learning rate η. For each pattern n in the network, the output (Eq. 3.20) and the gradient of the error function (E) with respect to the weight is computed (Eq. 3.21). The weight is updated according to the product of the learning rate with the sum of the gradients (Eq. 3.22).

$$y_n = y(\mathbf{x}_n) \tag{3.20}$$

$$\frac{\partial E}{\partial w_n} = -x_n \frac{\partial E}{\partial y_n}\varphi_0'(w_n x_n) \tag{3.21}$$

$$\omega_k \to \omega_k - \eta \sum_n \frac{\partial E}{\partial w_n} \tag{3.22}$$

An obstacle in the training process is to not overfit the model [25]. That is to not create a model which corresponds with the data it is being trained on too well in the sense that it will no longer perform reliably on additional data. A cause for overfitting is creating a model with too many parameters [25]. Therefore, a frequent regularization technique to manage this issue in ANNs is to, randomly, remove weights in the model during the training process. This process is called dropout [25]. Overfitting can be visible in a graph where the loss for the training and validation data is graphed for each epoch (the number of times the algorithm works through the data set). If the validation loss increases during the duration of training, it is a sign of overfitting.

3.2.1 Convolutional neural networks

Convolutional neural networks are a class of artificial neural networks. Although they are commonly used for image analysis [25] they have a specific strength within time series analysis as well [23, 11]. CNNs can make predictions based on a fixed-width history which makes them perform well in modelling processes that change over time.

Convolutional layers allow for fewer connections between layers and therefore less parameters to train. Not all input nodes affect all output nodes. In image recognition the theory is that pixels closer to each other have a higher resemblance. For time series data that too can be the case.

CNNs, as the name suggests, deal with convolution in the hidden layers which are called convolutional layers. Convolution is an operation on two functions

that produce a third function that expresses how the shape of one is modified by the other [36].

$$(f * g)_t = \int_{x=0}^{t} f(t-x)g(x)dx \tag{3.23}$$

In descrete time it is denoted as follows.

$$(f * g)_t = \sum_{j=1}^{m} g(j) \cdot f(t-j+m/2) \tag{3.24}$$

Convolution is done in the dimension of the inputs, which in the case of image recognition is two dimensional and in the case of time series data is one dimensional. Therefore, this thesis deals with the 1D CNN. The convolution is done between the input and the filter, called a kernel, in the layer. Practically this operation works in the way that we are sliding the kernel over the input vector. For each position of the kernel, we multiply the overlapping values of the kernel and vector together and add up the results. This sum of products will be the value of the output at the point in the input vector where the kernel is centered [32]. See example in Figure 3.1. The multiplicative factor in the kernel is the trained parameter and therefore the equivalent of a weight.

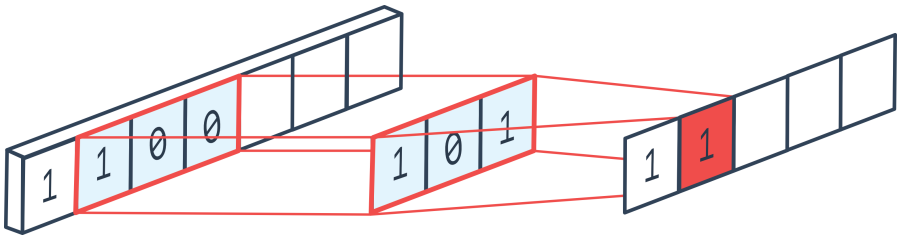

Figure 3.2: A 1D convolution with a kernel sized 3 and stride 1 [27]

It is common for CNNs to include a pooling layer after convolution to decrease the size of the output and only keep the significant parts. This is not done in this thesis as there is an advantage in keeping the input and output layer the same length [23]. Furthermore, it makes the process of rolling window predictions less complicated.

At the point in the network where the convolution-related operations are complete the data is inserted into a flattening layer for the data to be suitable for some ANN layer ahead. This is particularly the case for 2D convolutions that lead to a 2D output which is not be a suitable input into most layers outside of a CNN. In the case of a 1D convolution on a 1D input the flattening process yields the same result. However, a 1D convolutional layer can have multiple 1D inputs and therefore there is a necessity for the flattening operation. A dense layer, which is the most commonly used layer in ANNs consisting of nodes with weights, may thereafter be added to the model before the output layer.

Chapter 4

Method

4.1 Data description and selection

For the data analysis and prediction generation the author received access to a database containing historical production data, forecasts and other information for multiple production units or wind farms. One production unit/wind farm would be chosen for the prediction generation.

Validated production data is not reported to the power producer or manager by the TSO until after 14 days after power delivery. This is an obstacle in the case of models based on recent power production data. However, some production units allow for access to real-time production data. Unfortunately, data errors occur and the real-time data is oftentimes not an exact match to the validated production data. Therefore, two main requirements were decided by the author, in the choice of unit to analyze, to maximize the possibility of improving forecast qualities. Historical data needed to be clean and have as few unreported production stops as possible. Preferably it would be at full capacity for the whole period during which data is selected. Also, the real-time data needed to be accurate, historically. A wind farm in Finland was chosen to be analyzed upon.

The data was extracted using SQL and stored in the format presented in the Table 4.1. The stored baseline forecasts are at a spot level and therefore at a rolling 14-38 time steps. To keep the prediction horizons constant the days going into and out of daylight savings time were handled through the the removal or addition of one hour. In the case of adding one hour, the value was set as the average of the hours before and after. In the case of removing one hour, the remaining hour was set to the average of itself and the removed hour. Through this manipulation of data all days were set to being 24-hours long. As this is a very small part of the data set it is likely not going to alter the quality of the results.

The baseline forecasts are generated using a black box method and therefore it is unknown whether they are based on a physical or statistical method. For

Date and time	Baseline Forecast	Metered data	Realtime data
...	...	...	...
2021-05-11 00:00	3.768	4.425	4.426
2021-05-11 01:00	2.742	3.981	3.989
...	...	...	...

Table 4.1: A slice representation of the data used for the analysis. Mock data. Values are expressed in megawatts (MW).

forecasts at this time horizon, medium-term, real-time data is seldom used as an input [15]. This is limited to very short-term and short-term forecasts.

Due to autocorrelation seemingly being present in the baseline forecasts this thesis differs in choosing the approach of incorporating production data through the usage of real-time data. A variety of errors could possibly be assessed with this approach. In the data sample there are periods when either under- and/or over-forecasting is present for many consecutive hours. Peaks in production can consecutively be under-forecasted. Although the existence of some autocorrelation may be expected for predictions at this time horizon, the autocorrelation extends further than the prediction horizon in some parts of the data set. In Figure 4.1 and 4.2 it is visible that there is significant autocorrelation (95 % confidence interval) in the prediction errors up to a lag of around 400. Therefore, it is likely that real-time data is not used as an input into the baseline-model. It is therefore possible that using real-time data could be beneficial and could produce a forecast of better quality.

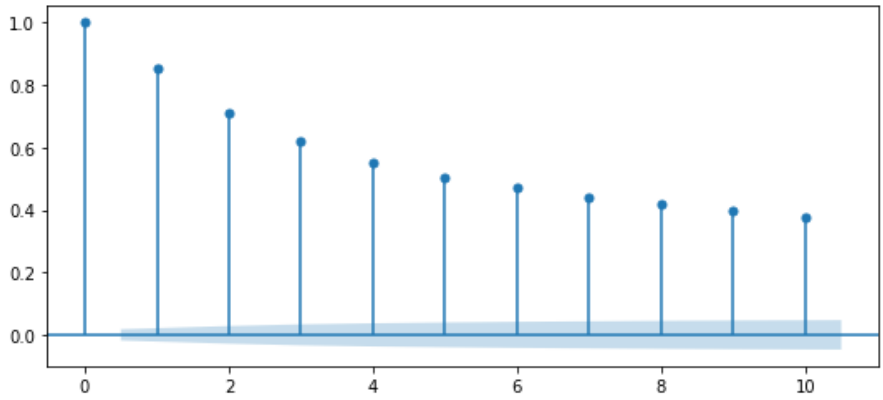

Figure 4.1: Autocorrelation function (ACF) graph up to lag 10. Light blue field represents the area where the autocorrelation is insignificant at a 95 % confidence interval

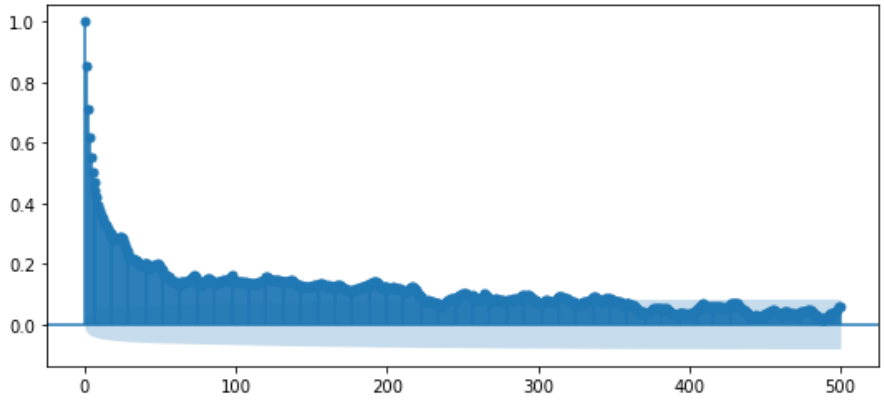

Figure 4.2: Autocorrelation function graph up to lag 500. Light blue field represents the area where the autocorrelation is insignificant at a 95 % confidence interval

4.1.1 Data split

The data used is of a length of 11688 which corresponds to the number of hours of data. This data is split up into a training, validation and test set. The data is spanning over all times of the year. It is outside the aim of this thesis to create different models for different times of the year as the correlation between power output and season is low.

- Training data - length 6000
- Validation data - length 1920
- Test data - length 3768

4.2 Rolling window predictions

The output of the predictions generated is in the same format as the input baseline forecast. Rolling window predictions were performed and a 38-time step prediction was generated with data available up until the hour starting 09:00 CET. Using the data $[x_0, \ldots, x_t]$ we make the prediction $[\hat{x}_{t+1}, \ldots, \hat{x}_{t+38}]$. Out of the 38 predictions the last 24 predictions were selected corresponding to each hour the next day. This was then done continually for the length of the data to be predicted upon. Due to the issue of validated production data not being available until after a period of 14 days after the time of production, recent data was replaced by real-time data. Fortunately, the differences in this particular example were very low.

4.3 Implementation of the Kalman filter

The process of implementing the Kalman filter was done using Python. We create a custom Kalman filter from the basics due to some unique properties of the problem being solved. As for example generating rolling window predictions.

To compute the rolling window predictions, the Kalman filter is firstly designed to create one-step predictions. The data is continually, in a loop, read from the data presented in Table 4.1. The approach for time series modelling is oftentimes to normalize the data through the removal of the mean of the training data set and division by the standard distribution. Such approach led to greater errors in this example and was therefore not done in the final model.

Row matrix $\mathbf{C}$ contains all relevant inputs and measurements. This is a modification of the model described in section 3.1 where these are kept in separate matrices (Eq. 3.1 - 3.2). The elements of the matrix are the baseline forecast for the predicted data point, real-time production data for the last 200 data points and the prediction error for the last 200 data points. For each iteration in the Kalman filter, when a new prediction is made, $\mathbf{C}$ is updated the following way,

$$\mathbf{C}_t = \begin{bmatrix} b_{t+1} & \cdots & b_{t+1} & y_{t-200} & \cdots & y_t & e_{t-200} & \cdots & e_t \end{bmatrix} \tag{4.1}$$

where b_{t+1} is the one-step baseline-forecast at time t, y_t is real-time data for time t and e_t is the prediction error for time t.

This then allows for the calculation of the Kalman gain $\mathbf{K}_t$ (Eq. 3.9) through the update of the reconstruction error variance $\mathbf{R}^{x,x}_{t|t-1}$ (Eq. 3.11) and the sample variance $\mathbf{R}^{y,y}_{t|t-1}$ (Eq. 3.12). Thereafter it is possible to compute the reconstruction (Eq. 3.7) and perform the one-step prediction (Eq. 3.8).

The prediction further in time is made through using the current estimation of the state space but updating the $\mathbf{C}$ matrix and making the relevant baseline forecast data points and prediction errors visible for the model (Eq 4.2). Using a loop, predictions at a horizon of 14-38 time steps were generated and stored in an array.

$$\mathbf{C}_t = \begin{bmatrix} b_{t+1} & \cdots & b_{t+k} & y_{t-200} & \cdots & y_t & e_{t-200} & \cdots & e_t \end{bmatrix} \tag{4.2}$$

4.3.1 Components

Through the process of acquiring sufficient results on the test and validation data the $\mathbf{C}$ matrix and the following components were set.

- $R_w = 7$
- $\mathbf{R}_e = 10^{-11} I_{500}$
- $\hat{\mathbf{x}}_{1|0} = 4I_{500}$
- $\mathbf{R}^{x,x}_{1|0} = I_{500}$
- $\mathbf{A} = I_{500}$

Where 500 is the number of components in $\mathbf{C}$

4.4 Implementation of the Convolutional Neural Network

The CNN-model was implemented using the Python library TensorFlow. The data had to be pre-processed for the model to be able to interpret it. The data, presented in Table 4.1, was transformed into arrays of the size (38, 2) which corresponds to the batch size of data being inserted during each iteration of passing data through the model. The batch array consists of two rows containing the real-time data for the last 38 hours and the baseline forecast for the following 38 hours which are the ones being predicted upon. Similarly, the labels were transformed into arrays of size (38, 1) and contained the production data points for which the model was being trained. This is depicted in Figure 4.4.

The output is thereafter of a length of 38 and the last 24 datapoints, corresponding to each hour the next day are extracted and stored.

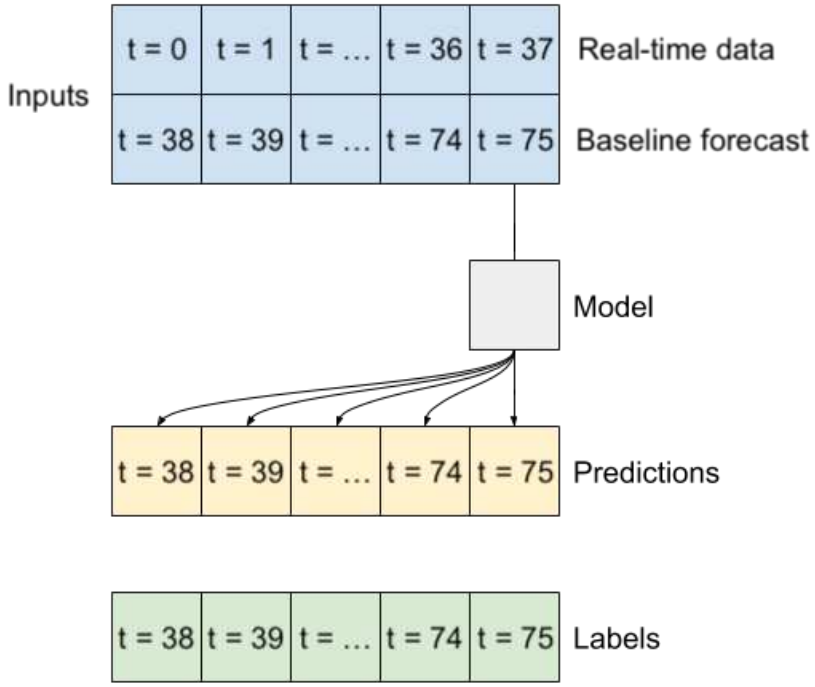

Figure 4.3: Figure representing the CNN data flow

Oftentimes, in regression type problems or time series predictions using ANNs the data is normalized. During the training process such approach did not lead to better results and therefore the final model does not incorporate normalization of the inputs. In the case that the process being modelled has a clear daily or yearly periodicity, the date and time can be transformed using sine and cosine to be used as an interpretable input. As the baseline prediction errors were deemed to not have a significant dependency on these factors it was decided to not proceed with this approach.

4.4.1 Model parameters

The CNN model consists of an input layer, which has no trainable parameters, and the ones presented in Table 4.2.

Layer type	Output Shape	Parameters
Conv1D	(37, 2048)	14336
Flatten	(75776)	0
Dense	(2048)	155191296
Dense	(38)	77862
Total		155,283,494

Table 4.2: The layers, their output shape and number of parameters of the CNN model. All parameters are trainable.

The model consists of only one convolutional layer as that led to the most sufficient results. The number of filters in that layer is 2048 and the kernel size is 2, which specifies the length of the convolution window.

Activation functions are used in the dense layers where the first one uses a Rectified Linear Unit (ReLU) and the second one uses a linear activation function. The loss is measured as mean squared error and the optimizer used is Adam with a learning rate of 10^{-4}.

4.5 Evaluation metrics

To evaluate the accuracy of predictions, a set of statistical evaluation methods can be used. A common method is the mean absolute error (MAE) which expresses the mean error of paired observations in absolute terms.

$$\text{MAE} = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n} \tag{4.3}$$

y_i represents the actual value while $\hat{y}_i$ represents the prediction.

It is preferable, for the result in this thesis to be comparable to future work, to use a normalized evaluation metric. The MAE would not be comparable to forecasts of wind farms of different sizes. It is also in the interest of trade secrets to not reveal the capacity of the analyzed wind farm. Therefore, the normalized mean absolute error is a fitting metric in this case, which also has been used in previous work.

$$\text{NMAE} = \frac{MAE}{\frac{\sum_{i=1}^{n} y_i}{n}} \tag{4.4}$$

Chapter 5

Results

The results of applying the Kalman filter and CNN model on the analyzed wind farm are presented below using NMAE. The results are presented for the training-, validation- and testing data.

For about the first 1000 data points of the training data, some of the parameters of the Kalman filter are stabilizing. For the models to be comparable, the first 1000 datapoints and predictions are removed from the evaluation set for the test data.

Figure 5.1 presents the training and validation loss per epoch for the CNN. The training loss decreases during training. It is desirable for the validation loss to do that as well. As seen in the graph, the validation loss decreases initially to later stabilize. It is expected for the validation loss to be greater than the training loss. Therefore this model is considered to be sufficient.

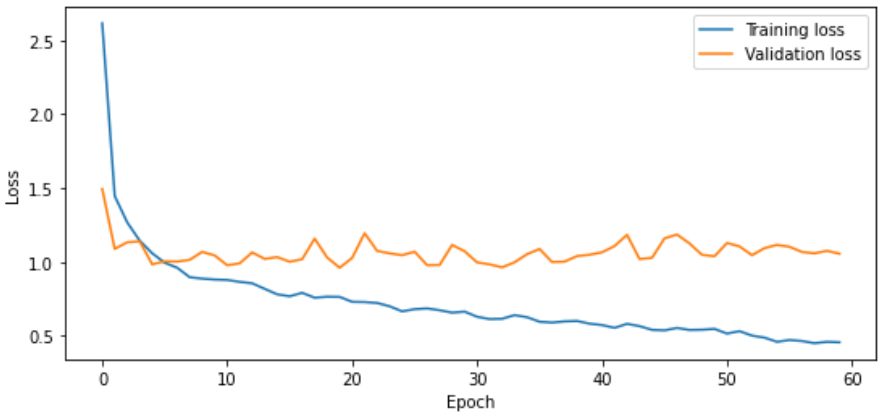

Figure 5.1: The loss for each epoch during the training and validation of the model.

Prediction model	NMAE(%)	Data set
Kalman filter	41.72 %	Training
CNN	20.79 %	Training
Baseline	41.34 %	Training

Table 5.1: NMAE for training data

Prediction model	NMAE(%)	Data set
Kalman filter	35.60 %	Validation
CNN	32.44 %	Validation
Baseline	30.57 %	Validation

Table 5.2: NMAE for validation data

Prediction model	NMAE(%)	Data set
Kalman filter	43.90 %	Test
CNN	41.58 %	Test
Baseline	43.80 %	Test

Table 5.3: NMAE for test data

Figures below (5.2 - 5.7) present samples of the predictions, actual production and baseline forecasts on the different data sets. These plots are not for the whole data sets but picked for periods found interesting, which is explained further in the discussion chapter. The axis for production volume is removed due to confidentiality.

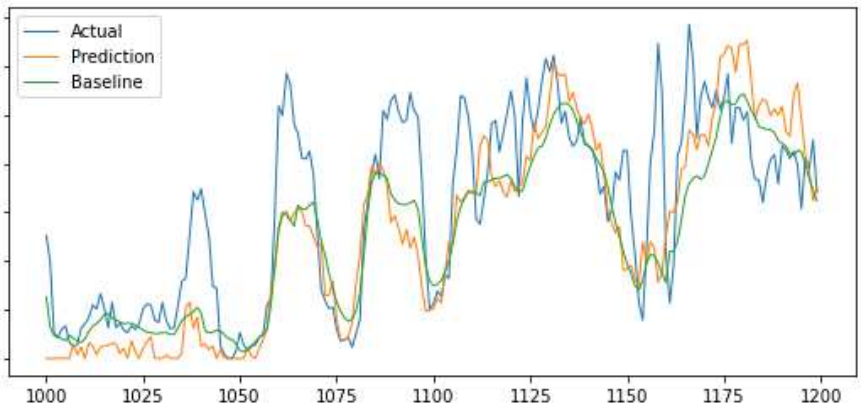

Figure 5.2: Kalman filter on training data

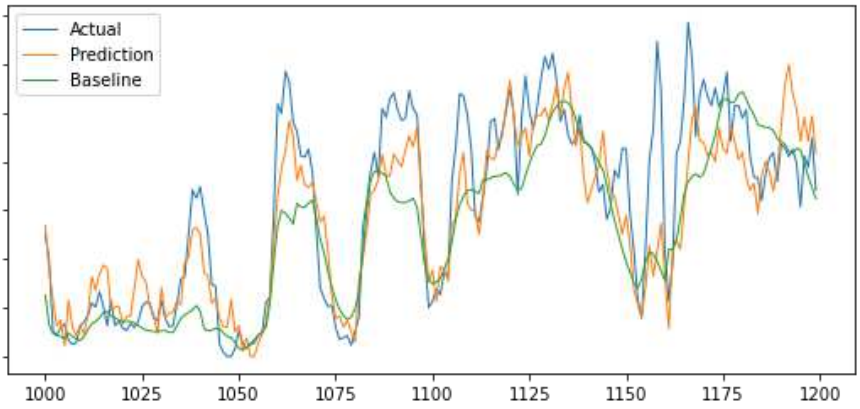

Figure 5.3: CNN on training data

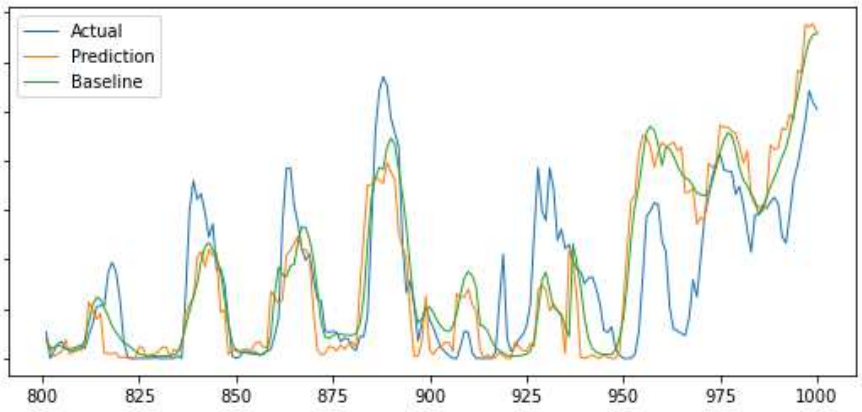

Figure 5.4: Kalman filter on validation data

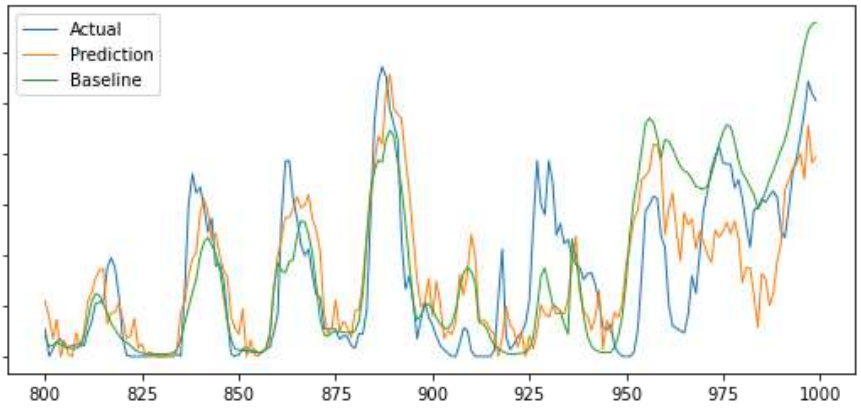

Figure 5.5: CNN on validation data

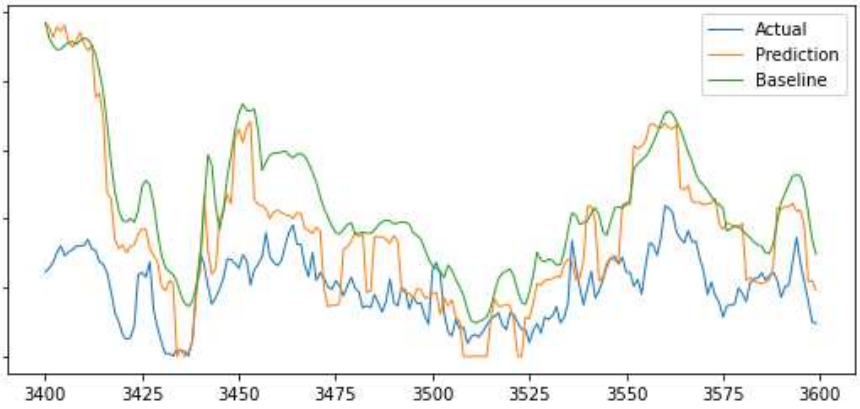

Figure 5.6: Kalman filter on test data

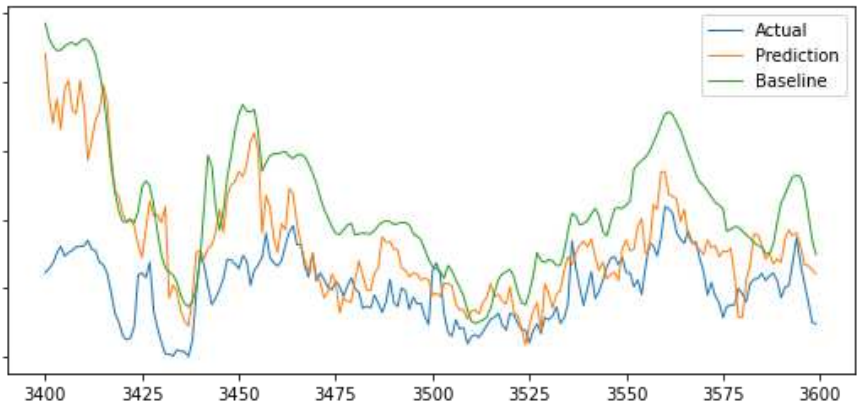

Figure 5.7: CNN on test data

Chapter 6

Discussion and conclusion

The information presented in the results is analyzed and discussed in this chapter. Further on conclusions are drawn and further research is recommended.

6.1 Result analysis

The samples of data presented in the graphs in the results chapter were chosen for times of systemic under- and over-forecasting. It is in these data sets where the advantage of incorporating real-time data should be visible. The CNN presented a higher degree of accuracy, than the baseline forecasts, during these periods of time. This is the case for the Kalman-filter model to some extent, as well. However, this is not necessarily a representation of the overall accuracy which is presented in terms of NMAE. Overall, the Kalman filter model delivers less accuracy for all the data sets while the CNN model varies. As the CNN model uses back-propagation during the training process, the accuracy of the model in this data set is not comparable to the Kalman filter for the same data set. The CNN model delivers slightly less accuracy for the validation data, than the baseline forecast but slightly better results for the test data. A possible cause for this is that there are longer periods of systemic baseline forecast errors in the test data compared to the validation data. These can for example be the result of unreported inactive wind turbines which the baseline model therefore does not take into consideration. However, they can also be a result of a sub-optimal underlying model. It is therefore visible, in the presented graphs, that during periods of time when the quality of the baseline forecast is low, the CNN model performs better.

It is also important to take into consideration that the analyzed wind farm allows for access to accurate real-time data. This is a necessity for the models in this thesis as that is a main input. As mentioned in Section 4.1, validated production data becomes available from the TSO after 14 days, in the case of the Nordic market. Access to real-time data of high quality is not necessarily the case in the wind power and/or power trading industry overall.

6.2 Comparison

For discussion purposes, it would be interesting to compare the models in this thesis and the baseline forecast to models in existing studies. It would mainly be interesting in regard to the ones referenced to in this thesis. Most of the existing studies regarding wind power predictions do not evaluate the results at the specific time frames as this thesis, which are rolling window predictions at a 14-38 hour horizon. The results in most papers are evaluated at a constant prediction horizon. However, a relevant study discusses the characteristics of spot wind power forecast errors in the Nordic region with data from 2014 [22]. The difference is that the forecasts in this study are generated at a 12-36 hour horizon. In all of Finland, the wind power forecast errors were at a MAE as a percentage of capacity at 8.46 %. The equivalent for the test data, in this thesis, is 8.59 % for the CNN and 9.05 % for the baseline forecast. Larger errors at a wind farm basis can be expected as the forecast errors between different wind farms at a country level are not fully correlated. The performance, to that extent in which they can be compared, is therefore similar.

6.3 Conclusion

Generating a better forecast than the baseline forecast, with high confidence, is a difficult task. However, as the autocorrelation of the errors in the baseline forecast varies over time, the CNN model performs better in some instances. The CNN shows better performance during periods of high autocorrelation in baseline prediction errors. Therefore, there could be an advantage to use such a model if the baseline forecast is known to be under-performing for a particular wind farm.

The Kalman filter, in this form, would not be as suitable as a substitute or addition. It does show better performance during times of high autocorrelation as well but the CNN model generally does a better job of this. However, there are possibilities to improve the implementation of the Kalman filter model-wise. The current implementation of the Kalman filter does not predict the state space at the prediction windows. A recursive approach which could approximate the state space at the predicted steps would probably lead to better results, than the current model.

The main conclusion is that there can be value to be added to wind power forecasts at a medium-term horizon through the input of real-time data into the prediction models.

6.4 Further research

Further research into the use of recent production, if available through real-time data, for forecasts could lead to an improvement in quality. As mentioned in Section 4.1, this is not the case for the large part of forecast generation at the time horizon used in this thesis.

This thesis does not work with the various inputs used in wind power prediction generation, mainly being weather forecasts. The performance of a model built upon the output of a black box model is unlikely to lead to the same results as if the underlying model would have had incorporated further inputs. This is therefore a limitation in this thesis. The models presented in this thesis probably do not use the full potential of such data and further research would be beneficial.

Bibliography

[1] Sanjeev Kumar Aggarwal. Wind power forecasting: A review of statistical models.

[2] K Beckman. Steve holliday, ceo national grid:"the idea of large power stations for baseload is outdated". *Energypost EU*, 11, 2015.

[3] Federico Cassola and Massimiliano Burlando. Wind speed and wind energy forecast through kalman filtering of numerical weather prediction model output. *Applied energy*, 99:154–166, 2012.

[4] Niya Chen, Zheng Qian, Ian T Nabney, and Xiaofeng Meng. Short-term wind power forecasting using gaussian processes. In *Twenty-third international joint conference on artificial intelligence*. Citeseer, 2013.

[5] Peiyuan Chen, Troels Pedersen, Birgitte Bak-Jensen, and Zhe Chen. Arima-based time series model of stochastic wind power generation. *IEEE transactions on power systems*, 25(2):667–676, 2009.

[6] Fernando J De Sisternes, Jesse D Jenkins, and Audun Botterud. The value of energy storage in decarbonizing the electricity sector. *Applied Energy*, 175:368–379, 2016.

[7] Sara Fogelberg and Ewa Lazarczyk. Wind power volatility and its impact on production failures in the nordic electricity market. *Renewable Energy*, 105:96–105, 2017.

[8] Saeid Jafarzadeh Ghoushchi, Sobhan Manjili, Abbas Mardani, and Mahyar Kamali Saraji. An extended new approach for forecasting short-term wind power using modified fuzzy wavelet neural network: A case study in wind power plant. *Energy*, 223:120052, 2021.

[9] Gregor Giebel, Richard Brownsword, George Kariniotakis, Michael Denhard, and Caroline Draxl. *The State-Of-The-Art in Short-Term Prediction of Wind Power: A Literature Overview, 2nd edition*. ANEMOS.plus, 2011. Project funded by the European Commission under the 6th Framework Program, Priority 6.1: Sustainable Energy Systems.

[10] Shubhi Harbola and Volker Coors. One dimensional convolutional neural network architectures for wind prediction. *Energy Conversion and Management*, 195:70–75, 2019.

[11] Ying-Yi Hong and Christian Lian Paulo P Rioflorido. A hybrid deep learning-based neural network for 24-h ahead wind power forecasting. *Applied Energy*, 250:530–539, 2019.

[12] Andreas Jakobsson. *An introduction to time series modeling*. Studentlitteratur, 2019.

[13] Ana Maria Jaller-Makarewicz. As fossil fuel prices skyrocket globally, renewables grow steadily cheaper, Sep 2021.

[14] Jordan Hanania, Kailyn Stenhouse and Jason Donev. Dispatchable source of electricity , 2020.

[15] Jaesung Jung and Robert P Broadwater. Current status and future advances for wind speed and power forecasting. *Renewable and Sustainable Energy Reviews*, 31:762–777, 2014.

[16] Mansoor Khan, Tianqi Liu, and Farhan Ullah. A new hybrid approach to forecast wind power for large scale wind turbine data using deep learning with tensorflow framework and principal component analysis. *Energies*, 12(12):2229, 2019.

[17] M Najafi Khoshrodi, Mohammad Jannati, and Tole Sutikno. A review of wind speed estimation for wind turbine systems based on kalman filter technique. *International Journal of Electrical and Computer Engineering*, 6(4):1406, 2016.

[18] Shuhui Li. Wind power prediction using recurrent multilayer perceptron neural networks. In *2003 IEEE Power Engineering Society General Meeting (IEEE Cat. No. 03CH37491)*, volume 4, pages 2325–2330. IEEE, 2003.

[19] Zi Lin, Xiaolei Liu, and Maurizio Collu. Wind power prediction based on high-frequency scada data along with isolation forest and deep learning neural networks. *International Journal of Electrical Power & Energy Systems*, 118:105835, 2020.

[20] Hui Liu, Xi-wei Mi, and Yan-fei Li. Wind speed forecasting method based on deep learning strategy using empirical wavelet transform, long short term memory neural network and elman neural network. *Energy conversion and management*, 156:498–514, 2018.

[21] Petroula Louka, Georges Galanis, Nils Siebert, Georges Kariniotakis, Petros Katsafados, Ioannis Pytharoulis, and George Kallos. Improvements in wind speed forecasts for wind power prediction purposes using kalman filtering. *Journal of Wind Engineering and Industrial Aerodynamics*, 96(12):2348–2362, 2008.

[22] Jari Johannes Miettinen and Hannele Holttinen. Characteristics of day-ahead wind power forecast errors in nordic countries and benefits of aggregation. *Wind Energy*, 20(6):959–972, 2017.

[23] Roni Mittelman. Time-series modeling with undecimated fully convolutional neural networks. *arXiv preprint arXiv:1508.00317*, 2015.

[24] Nord Pool. Day-ahead market . https://www.nordpoolgroup.com/en/the-power-market/Day-ahead-market/, 2022. Accessed: 2022-06-04.

[25] Edén Ohlsson. Introduction to artificial neural networks and deep learning.

[26] Inci Okumus and Ali Dinler. Current status of wind energy forecasting and a hybrid method for hourly predictions. *Energy Conversion and Management*, 123:362–371, 2016.

[27] Peltarion. 1d convolution. https://peltarion.com/knowledge-center/documentation/modeling-view/build-an-ai-model/blocks/1d-convolution.

[28] Ramiz Qussous, Nick Harder, and Anke Weidlich. Understanding power market dynamics by reflecting market interrelations and flexibility-oriented bidding strategies. *Energies*, 15(2):494, 2022.

[29] Shafiqur Rehman, Md Alam, Luai M Alhems, M Mujahid Rafique, et al. Horizontal axis wind turbine blade design methodologies for efficiency enhancement—a review. *Energies*, 11(3):506, 2018.

[30] Bruce Robertson. Ieefa update: Gas is exiting the global energy mix, providing lessons for emerging markets.

[31] Miadreza Shafie-Khah and João PS Catalão. A stochastic multi-layer agent-based model to study electricity market participants behavior. *IEEE Transactions on Power Systems*, 30(2):867–881, 2014.

[32] Noah Snavely. Cs1114 section 6: Convolution. *Cornell University*, pages 1–6, 2013.

[33] Saurabh S Soman, Hamidreza Zareipour, Om Malik, and Paras Mandal. A review of wind power and wind speed forecasting methods with different time horizons. In *North American power symposium 2010*, pages 1–8. IEEE, 2010.

[34] Petr Spodniak, Kimmo Ollikka, and Samuli Honkapuro. The impact of wind power and electricity demand on the relevance of different short-term electricity markets: The nordic case. *Applied Energy*, 283:116063, 2021.

[35] IRENA Statistics. Renewable energy capacity statistics 2021, 2021.

[36] Wikipedia. Convolution. http://en.wikipedia.org/w/index.php?title=Convolutionoldid=1087802892, 2022. [Online; accessed 06-June-2022].

[37] Wikipedia. Electricity sector in Sweden. http://en.wikipedia.org/w/index.php?title=Electricity%20sector%20in%20Swedenoldid=1088898051, 2022. [Online; accessed 06-June-2022].

Master's Theses in Mathematical Sciences 2022:E37
ISSN 1404-6342

LUTFMS-3442-2022

Mathematical Statistics
Centre for Mathematical Sciences
Lund University
Box 118, SE-221 00 Lund, Sweden

http://www.maths.lu.se/