UNIVERSITY OF TARTU
Faculty of Science and Technology
Institute of Computer Science
Computer Science Curriculum

Denys Kolomiiets

A Competitive Scenario Forecaster using XGBoost and Gaussian Copula

Master's Thesis (30 ECTS)

Supervisor(s): Novin Shahroudi, MSc
Meelis Kull, PhD

Tartu 2023

A Competitive Scenario Forecaster using XGBoost and Gaussian Copula

Abstract:
In recent years scenario forecasting has been explored and developed by multiple authors. It is a useful technique for setting such as renewable energy production, which is extremely important for a society transitioning from fossil fuel energy generation. Currently, one of the methods to approach the task of scenario forecasting are generative models. The primary goal of this thesis is to develop an approach that outperforms the current best model, using the decision tree model method. This work also discusses possible improvements for decision tree models in scenario forecasting setting. Our approach has surpassed the performance of generative models, making it a solid new baseline for future researchers to beat.

Keywords:
XGBoost, Gaussian Copula, Quantile forecasting, Scenario forecasting, Energy forecasting, Time series

CERCS: P176 Artificial Intelligence

Konkurentsivõimeline stsenaariumide prognoosija kasutades XGBoosti ja Gaussi koopulat

Lühikokkuvõte:
Viimastel aastatel on stsenaariumiprognoosimist uuritud ja arendatud mitme autori poolt. See on kasulik tehnika taastuvenergia tootmise jaoks ja on äärmiselt oluline ühiskonnale, mis läheb üle fossiilkütuste kasutamiselt energia tootmisel. Praegu on stsenaariumiprognoosimise ülesande lahendamiseks üheks meetodiks generatiivsed mudelid. Selle magistritöö peamine eesmärk on välja töötada lähenemisviis, mis ületab praegu parima mudeli, kasutades otsustuspuu meetodit. Käesolevas töös arutatakse ka võimalikke täiustusi otsustuspuu mudelites stsenaariumiprognoosimisel. Meie lähenemisviis on ületanud generatiivsete mudelite jõudluse, muutes selle kindlaks uueks baasjooneks tulevastele teadlastele.

Võtmesõnad:
XGBoost, Gaussian Copula, Kvantiliprognoosid, Stsenaariumiprognoosid, Energiaprognoosid, Ajareasarjade prognoosimine

CERCS: P176 Tehisintellekt

Contents

1 Introduction

Probabilistic forecast is an approach that is useful in many tasks that have to take uncertainty into account. There could be many potential uses for such methods, but our work focuses on renewable energy production and demand analysis. Since energy production facilities, such as solar and wind farms, rely on the weather, the amount of energy produced is inherently unpredictable. Renewable energy systems must be supplemented with either energy storage facilities or less desirable non-renewable generators. The cost and risk of renewable energy sources must be analyzed for a more reliable energy supply and better energy efficiency. Thus, better forecasting and analysis methods are incredibly important for faster and wider adoption of renewable energy sources. This has been the aim of many researchers and competitions, such as GEFCom2014.

Our work focuses on scenario forecasting, which is a way to represent uncertainty in forecasts. This thesis complements the work of [DWL+22] by comparing it to a more established algorithm, XGBoost [CG16], in the same setup as theirs. The experiments have shown that current State of the Art models do not outperform the XGBoost algorithm with a copula-based scenario generation technique.

Section 2 describes the dataset used, methods used in modelling, and previous work on this thesis is based. It is important to establish the background to understand later chapters properly. Section 3 describes the differences between our work and [DWL+22], mainly data representation and evaluation metrics used. The section describes the implementation details of the practical part in-depth. Section 5 analyses the results of experiments and speculates about the differences in performance between tracks and models. Section 6 concludes our work and summarizes the results. It also states the possibility of future research and development of probabilistic methods of the XGBoost algorithm.

2 Background

2.1 GEFCom2014

Most of the field of forecasting is focused on marginal or point forecasts. However, there are different applications where an event's probability must be considered. One such application is the renewable energy industry, which is inherently uncertain, so the probabilistic methods make more sense.

GEFCom2014 was a competition aimed at gathering Computer Science students and researchers to improve forecasting approaches in renewable energy setting [HPF+16] using probabilistic forecasting techniques. The organizers of the competition emphasize the probabilistic forecasting methodologies, their application to the energy sector, and different maturity levels between techniques as one of the challenges aimed to be addressed by the Competition. As such, the competition has included a diverse set of time-series datasets called tracks.

The three datasets employed were:

1. Load track - history of load demand, collected over 5 years, 2006 - Jan 2012 in America. Input consists of 25 parameters of temperature readings. Output is a load value for the hour.

2. Wind track - power production of a wind farm, collected over 2 years, 2012-Jan 2014 in Australia. It has 10 different zones, depending on wind speed and direction at 10m and 100m altitudes. An output parameter is the amount of power produced.

3. Solar track - power production of the solar farm, collected over 2 years, Apr 2012-Jul 2014 in Australia. Has 3 different zones. Depends on multiple weather features.

Competitors have employed a lot of different algorithms. Still, notably, XGBoost and Random forest trees have shown good performance on multiple tracks, implemented by teams such as **dmlab** for wind and solar tracks. Interestingly, none of the top teams used Tree-based models on the loaded track. This could be due to the lower performance of such models, as the loaded track is the only one where XGBoost did not outperform state-of-the-art models.

Forecast quality was evaluated with **Pinball loss**, averaged over 99 quantiles. Detail description of this method can be found in subsection 2.4.1. This is the same setup as in [DWL+22] and our work.

2.1.1 Load track

The aim of the GEFCom2014-L was to forecast the quantiles of hourly loads for a US utility on a rolling basis. The forecast horizon was one month. Hourly historical

load and weather data for the utility were provided. In addition to the data provided by the competition organizer, the contestants were also allowed to use US federal holiday information. However, it hasn't been used in our work, as [DWL+22] have not used it to train the models. Thus, for the purposes of a fairer comparison, holidays have not been included. Figure 1 shows the load distribution across time, and 2 shows the relationship between temperature and load consumed.

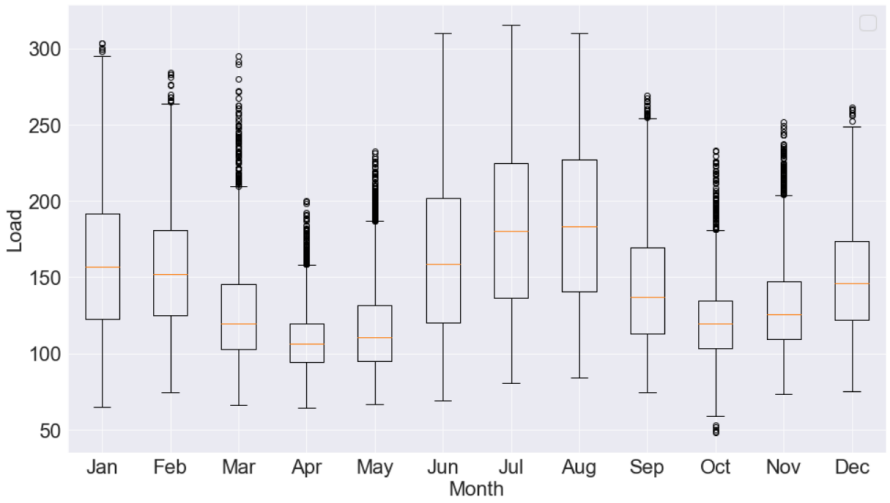


Figure 1. Load distribution

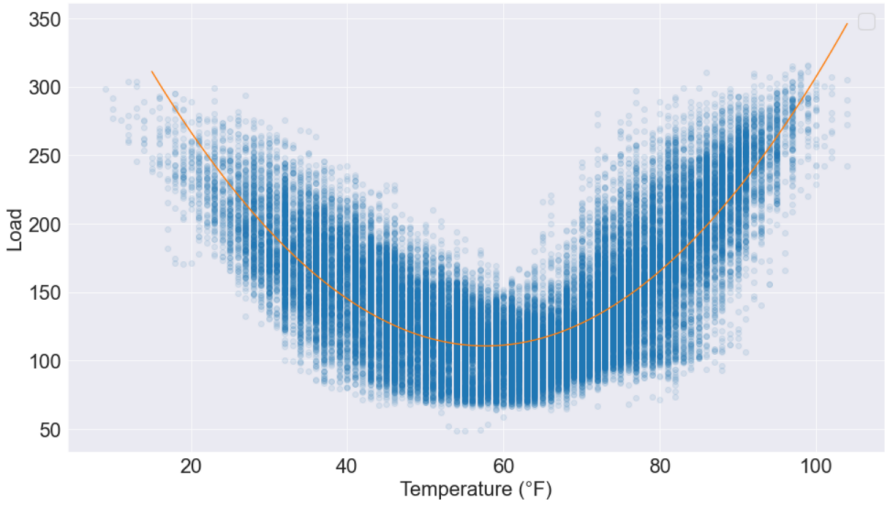


Figure 2. Relationship between load and temperature

From these two figures, it can be clearly seen that Load is highly dependent on temperature in a U-shaped manner. Thus, it can be assumed that in the winter month and the hottest times of summer, the load will be the highest. Figure 1 shows exactly that, where high peaks occur in winter months (around January) and summer months (around July), with troughs in between. And while seasonal features could make sense, the temperature has already been provided as the main input feature, thus making any seasonal features redundant.

Organizers of the competition also cited three other challenges imposed by Load track:

1. Weather station selection. Competition organizers provided 25 weather stations but no geographical data to identify their locations. It was done so competitors would develop an advanced algorithm to identify and select better weather stations.

2. Multi-horizon load forecasting. The one-month ahead load forecasting was chosen as a competition topic so that contestants have room to develop short-term load forecasts to improve a few days ahead forecast.

3. Scenario generation. Organizers expected that some competitors would investigate the possibility of scenario-generation methods. It was claimed that ten years of data provided would be enough to evaluate that method.

2.1.2 Wind track

The objective of the probabilistic wind power forecasting track in GEFCom2014 was to make predictions about the wind power generation of ten wind farms located in Australia. This was done by predicting the wind power generation 24 hours in advance for ten different zones that corresponded to the ten wind farms on a continuous basis. The wind power output series from these wind farms are shown in Figure 3. The locations of these 10 wind farms were not disclosed during GEFCom2014. The forecasts were to be expressed in the form of a set of 99 quantiles, with various nominal proportions between 0 and 1.

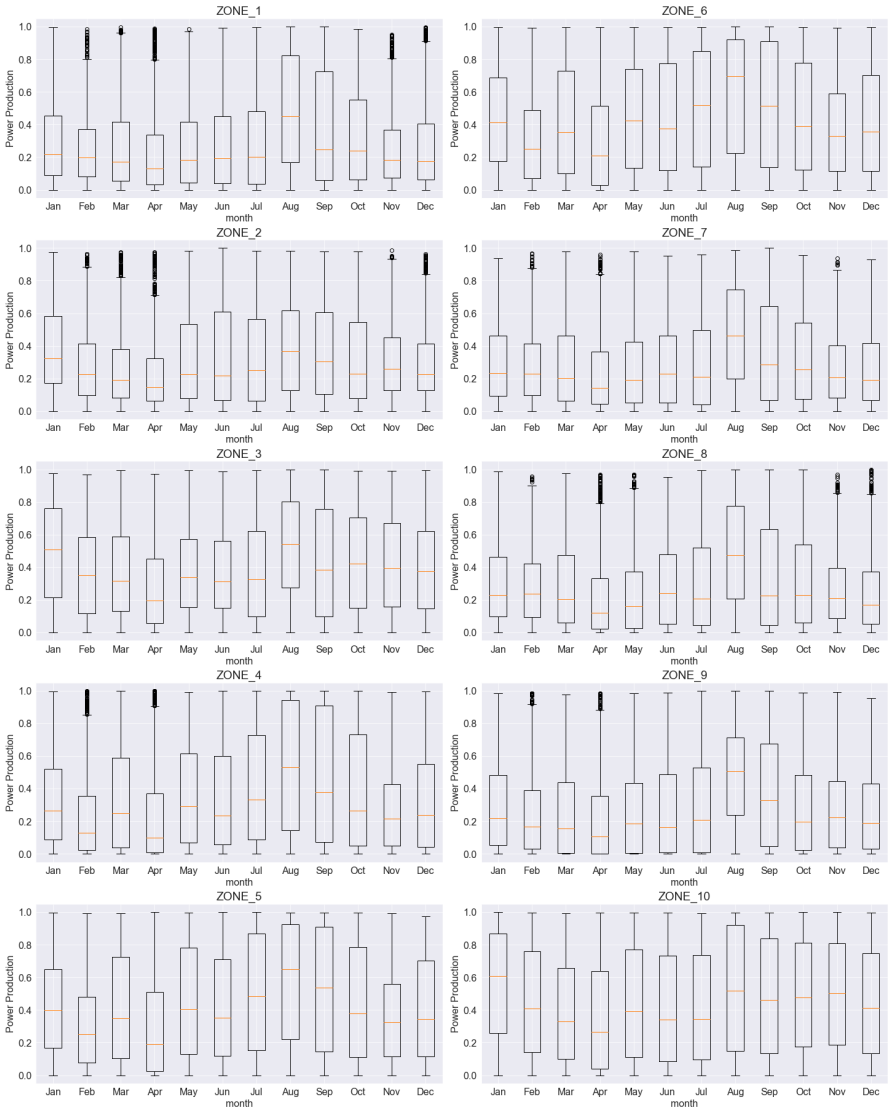


Figure 3. Distribution of wind speed

The input parameters for the forecasting model comprised wind speed forecasts, which were obtained from the European Centre for Medium-range Weather Forecasts (ECMWF). The forecasts provided wind speed estimates for two different heights, 10 meters and 100 meters above ground level, for both the zonal and meridional wind components. The projections of the wind vector represented these components onto the west-east and south-north axes, respectively. Figure 4 shows the scatter plots between wind power generation and wind speeds. Overall, it's hard to point out any concrete relationship s in either of the two figures. Though, it can be pointed out that overall all

power-to-speed relationship follow a U-shape, with more power produced towards the leftmost and rightmost side of the graph.

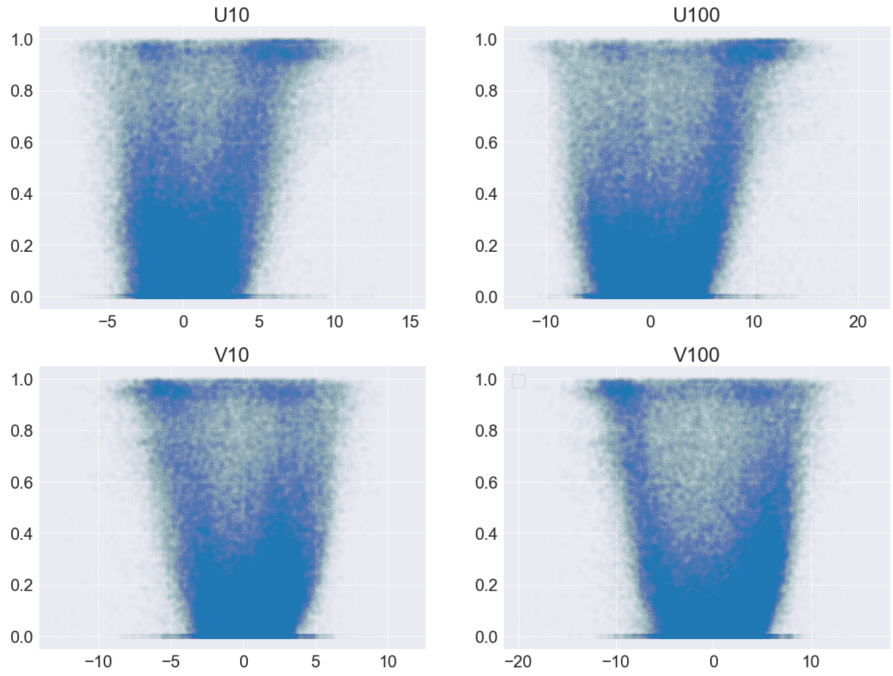


Figure 4. Relationship between wind speed and power production

2.1.3 Solar Track

The design of the probabilistic solar power forecasting problem in GEFCom2014 was similar to that of the wind track. The forecasting task involved predicting the solar power generation on a rolling basis, 24 hours ahead of time, for three different solar power plants located within a specific region of Australia. The solar power generation profiles are shown in Figure 5. The exact locations of these plants were undisclosed during the competition. The forecasts were expressed as 99 quantiles with nominal proportions ranging from 0 to 1. Participants had access to weather forecasts for 12 weather variables obtained from the European Centre for Medium-range Weather Forecasts (ECMWF). These variables are summarized in Table 1.

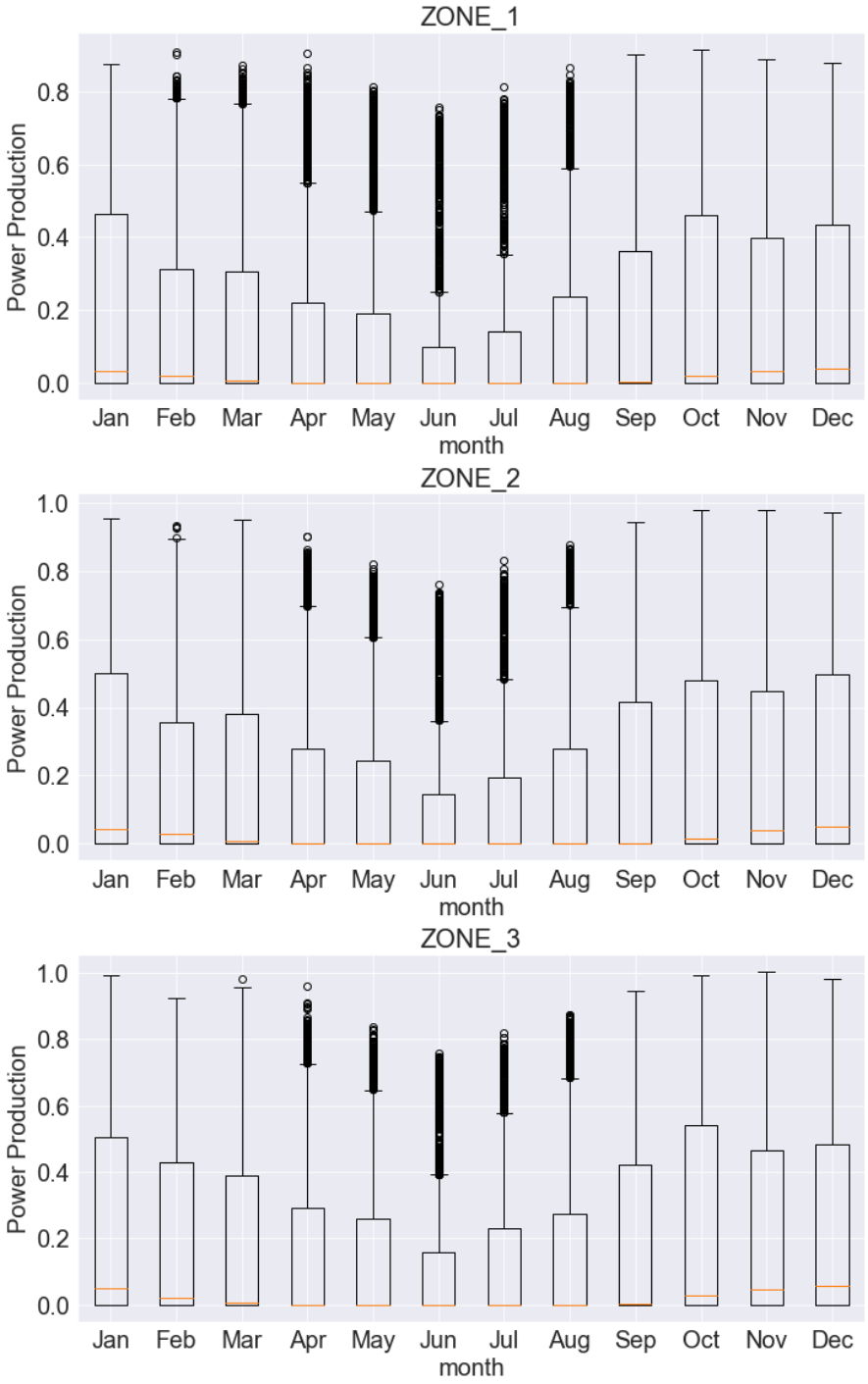


Figure 5. Distribution of Solar power production

Variable name	Units	Comments
Total column liquid water (tclw)	$kg * m^{-2}$	Vertical integral of cloud liquid water content
Total column ice water (tciw)	$kg * m^{-2}$	Vertical integral of cloud ice water content
Surface pressure (SP)	Pa	
Relative humidity at 1000 mbar (r)	%	Relative humidity is defined with respect to saturation of the mixed phase
Total cloud cover (TCC)	0-1	Total cloud cover derived from model levels using the model's overlap assumption
10-metre U wind component (10u)	$m * s^{-1}$	
10-metre V wind component (10v)	$m * s^{-1}$	
2-metre temperature (2T)	K	
Surface solar rad down (SSRD)	$J * m^{-2}$	Accumulated field
Surface thermal rad down (STRD)	$J * m^{-2}$	Accumulated field
Top net solar rad (TSR)	$J * m^{-2}$	Net solar radiation at the top of the atmosphere. Accumulated field
Total precipitation (TP)	m	Convective precipitation + stratiform precipitation (CP + LSP). Accumulated field

Table 1. Solar track variables

2.2 Dumas et al. contribution

The paper [DWL+22] introduced the current state-of-the-art model for scenario forecasting. Another major impact of [DWL+22] work is the improvement of the evaluation process, which combines the scenarios of all three tracks and produces monetary evaluation.

The three models considered in the original paper are: Normalizing flows [KSJ+16]; Variational autoencoders (VAE) [QHD+20]; Generative adversarial networks (GANs) [GPAM+14]. While the inner working of these models is irrelevant to our work, a performance comparison is the most interesting part between these three. NFs have outperformed the other two models on 2 out of 3 tracks, as well as generating the highest net profit. VAE outperformed the NFs on the Wind track, making it the second-best out of the 3 proposed.

A second important contribution of [DWL+22] was a thorough qualitative and quantitative evaluation setup, which used 8 different evaluation metrics. However, if a model

outperformed others, all evaluation metrics were consistent in their judgement. Therefore, the two most important for the task of scenario forecasting have been chosen for our work.

The quantitative part was a setup representing a downstream task of a machine learning setup. It aimed to optimize pricing for an energy provider, relying on generated scenarios. It is crucial for the final model performance analysis, as it combines the results of all three tracks and simulates real-world cost-benefit analysis. It also gives researchers a good estimation of the potential upside of a model under development over the existing one.

2.3 XGBoost algorithm

XGBoost, eXtreme Gradient Boosting [CG16], is a high-performance machine learning algorithm that ensembles multiple Decision Trees to approximate certain functions. The simplest representation of decision trees can be seen in Figure 6. The basic idea is to combine multiple weak function approximations to develop a more resilient one. The paper [CG16] describes the algorithm in use, optimization of that algorithm, implementation of different scalability features and analyses the resulting performance compared to existing solutions.

To decrease overfitting, it uses shrinkage [Fri02], which reduces the weight of newly added trees by a certain factor. Another technique is feature subsampling [Bre01], as well as supported instance subsampling.

The algorithm is a tree-ensembling model employing second-order objective [FHT00]. For certain data, it creates a number of trees. Each leaf contains a continuous score. For a given example, each tree calculates the value for corresponding leaves, which is summed up afterwards, giving a score for the example.

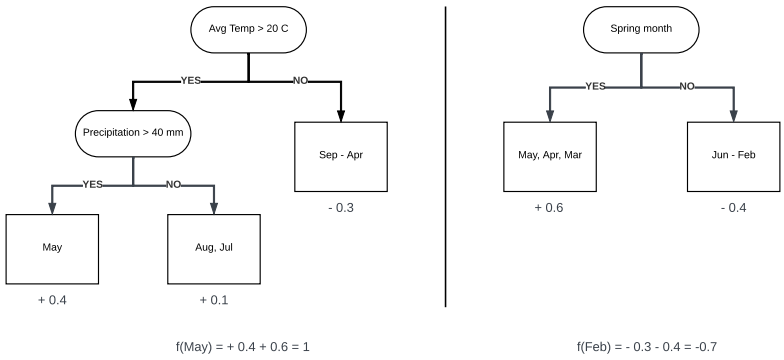


Figure 6. XGBoost algorithm representation

Since such a model contains functions as parameters, it cannot be optimized using traditional gradient descent algorithms. It is optimized in an additive manner, adding the tree that improves predictions the most in a greedy manner.

2.4 Pinball loss and Energy score

2.4.1 Pinball loss

Pinball loss, also called Quantile loss or Quantile score (QS), is a metric used to assess the performance of a quantile forecast. The loss is similar to the Mean Absolute Error (MAE) function. However, it assigns a higher weight to quantile predictions lower than median quantiles and a lower weight to higher than median quantiles. At quantile τ = 0.5 the loss formula equals to Mean Absolute Error. A more detailed overview of Pinball loss can be found in [BP10].

Let τ be the desired quantile, y be the real value, and z be the predicted value corresponding to quantile τ. Than Pinball loss L_τ is written like this 7:

$$L_\tau(y,z) = \begin{cases} (y-z)\tau, & \text{if } y \geq z \\ (z-y)(1-\tau), & y < z \end{cases}$$

Figure 7. Quantile loss formula

The resulting graph for this formula can be seen in Figure 8.

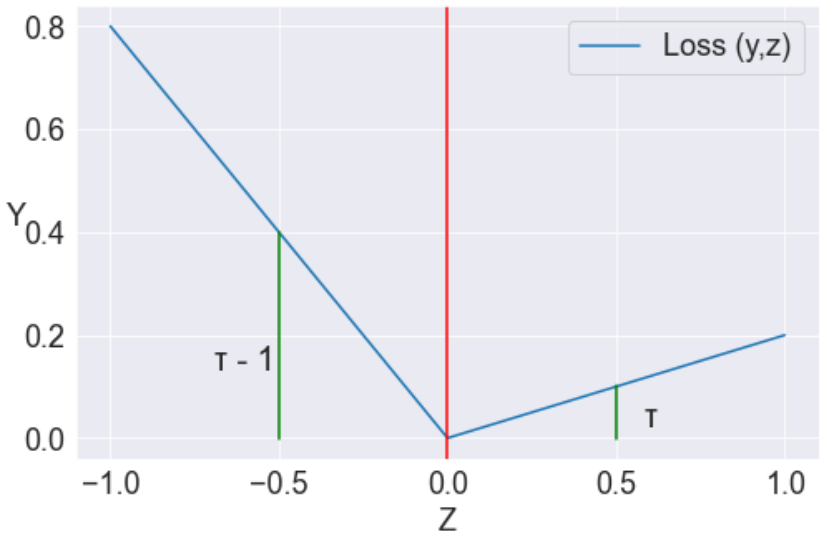


Figure 8. Pinball loss graph

2.4.2 Energy distance

Energy distance, or Energy score (ES), represents the distance between two probability distributions. The idea of energy is analogues to the potential energy of bodies in physics, where the potential energy is zero only if the distance between two objects is zero. And the energy between objects, and by proxy - distributions, increases with the distance between them.

Definition of energy score is as follows: Let X and Y be independent random vectors in R^d , with cumulative distribution function (CDF) F and G, respectively. In what follows, $||\cdot||$ denotes the Euclidean norm (length) of its argument, E denotes the expected value, and a primed random variable X denotes an independent and identically distributed (iid) copy of X; that is, X and X' are iid. Similarly, Y and Y' are iid. The squared energy distance can be defined in terms of expected distances between the random vectors, and the energy distance between distributions F and G is defined as the square root of $D^2(F, G)$ [SR13].

$$D^2(F,G) = 2E||X-Y|| - E||X-X'|| - E||Y-Y'|| \geq 0$$

Figure 9. Energy distance formula

2.5 Copula

Copula functions are used to describe the correlation between random variables [PMN$^+$09]. Consider a random vector $(X_1, X_2, ...X_{24})$ with continuous marginals, where cumulative distribution functions $F_{X_i}(x) = P[X_i \leq x]$ are continuous. A random vector will have uniformly distributed marginals if the probability integral transform is applied to each component.

$$(U_1, U_2, ...U_{24}) = (F_1(X_1), F_2(X_2), ...F_{24}(X_{24}))$$

The copula for $(X_1, X_2, ...X_{24})$ is defined as a joint cumulative distribution function $(U_1, U_2, ...U_{24})$

$$C(u_1, u_2, ...u_{24}) = P[U_1 \leq u_1, U_2 \leq u_2, ...U_{24} \leq u_{24}]$$

The experiments have been used to obtain the dependencies between hours of the day. It is necessary for further scenario generation, as scenarios have to consider previous values by definition. An example of the Copula function in matrix form can be seen in figure 16.

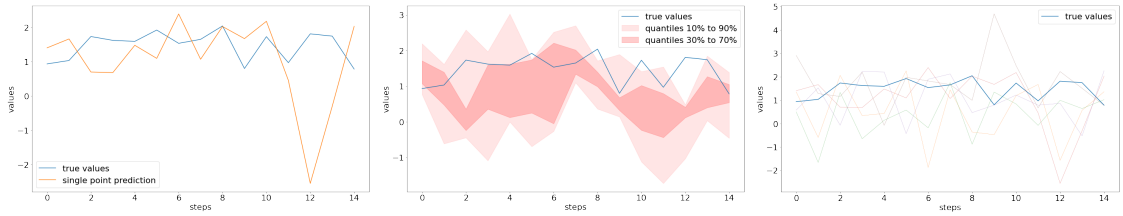
2.6 Scenario Forecast

Scenario forecasting is a method used to anticipate and evaluate various possible future outcomes based on different assumptions about the variables that are likely to impact the situation under analysis. The goal is to develop multiple plausible scenarios, rather than relying on a single prediction of the future, to help decision-makers better prepare for various potential outcomes [MCM+14].

A single scenario is a specific prediction of what the future might look like based on a set of assumptions about how different variables will change over time. However, it is often difficult to accurately predict the future with certainty, and relying on a single scenario can be risky. To address this, scenario forecasting involves developing multiple scenarios that reflect a range of possible futures based on different assumptions about how key variables might evolve.

Examples of a single point, quantile and scenario forecasts can be seen in Figure 10. In the Single point example, there is only true values for each step and predictions for each step. In the quantile figure, there is a predicted distribution for each step. And in the Scenario example, there are 5 different scenarios predicted for all 15 steps.

By generating a range of scenarios, decision-makers can explore the potential implications and consequences of different future outcomes and develop more robust and adaptive strategies that can be better suited to handle various scenarios. Scenario forecasting aims to give organisations a more informed and comprehensive understanding of the risks and opportunities associated with different future outcomes and help them prepare accordingly.



(a) Single Point Forecast (b) Quantile forecast (c) Scenario forecast

Figure 10. Forecast examples

3 Method

[DWL+22] established a solid framework of experiments and assessment of scenario forecasts. It does not rely only on conventional evaluation methods with different quality metrics and measures but also conducts a downstream evaluation based on the monetary evaluation of Scenario Forecasts. This is extremely important for any future research, as it provides a comprehensive overview of each model forecast's strengths and weaknesses. Such an assessment is also tremendously useful in demonstrating the practical implications of research in the field to potential users.

[DWL+22] have not compared results to more established Copula-based approaches. Our work also bridges this gap by using Copula scenario generation in tandem with the well-regarded algorithm XGBoost. However, due to major differences between the approaches, there are several key changes to the setup of the original work. These changes are outlined in this section. Another important contribution of [DWL+22] work is establishing an assessment framework. It considers both the conventional evaluation metrics such as Quantile Score and slightly less known Energy Score, introduced in [GR07], and lays out a process for downstream economic evaluation. This monetary evaluation of forecast quality provides a clear and practical view of the problem of forecasting quality in a much wider picture.

We used Quantile Score and Energy score metrics from [DWL+22] in our work. Only two have been chosen because of their relevancy for specific tasks of Scenario Forecasting, Quantile score representing a Univariate metric and Energy score – a Multivariate metric. These two metrics adequately represent the differences in performance between models. Following the results of [DWL+22], metrics from the same family provide similar performance evaluations.

Other specific metrics have not been, as they were required for the specific experimental setup of [DWL+22]. Statistical analysis was not performed, as it was used to reinforce the results obtained with other metrics.

The process flow of our work is shown in Figure 11

Monetary analysis of predictions is performed by solving optimization problems, using forecasted values as input. It uses licensed Gurobi API. It uses predictions of 50 Test days, each day containing 50 scenarios.

3.1 XGBoost-Gaussian Copula

Extreme Gradient Boosting regressor is a powerful algorithm that outperforms other models in tabular data settings. This has been shown in the work of [BLS+21], where gradient boosting methods have the best and second-best results across multiple datasets. One neural network model shows higher performance on one out of five datasets. This makes gradient boosting a go-to choice for tabular datasets and cases where time series can be converted into tabular.

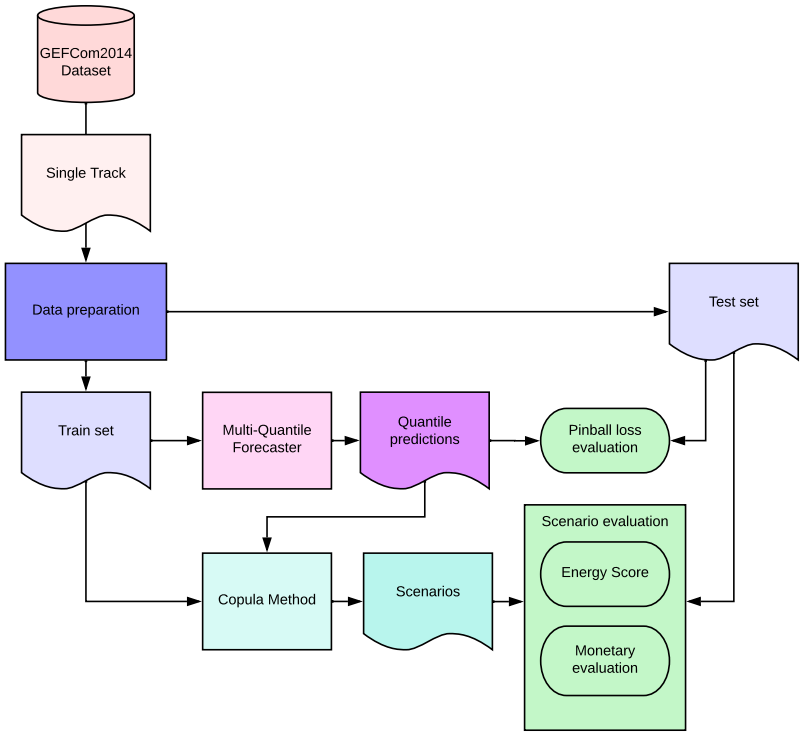


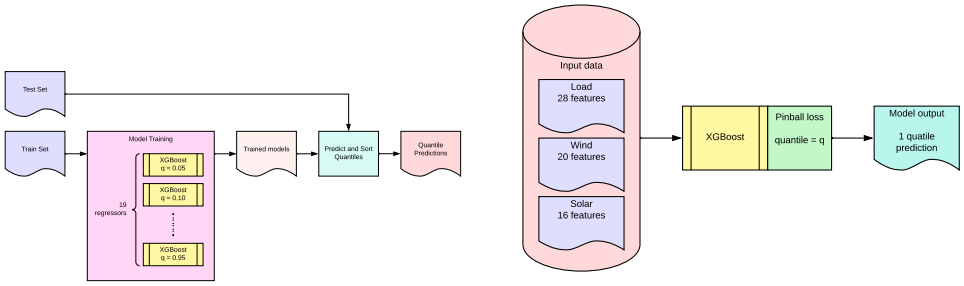
Figure 11. Process flow chart

The dominating performance of XGBoost on tabular data is the reason for the following hypothesis: XGBoost should be one of the main algorithms that a model has to outperform to claim State of the Art status. Therefore, we propose a combination of XGBoost as an established method for time series forecasting and combining it with Gaussian Copula to produce scenarios (XGBoost - GC).

3.1.1 Quantile forecast step

Some implementations of XGBoost allow the performing of multi-label classification and multi-regression tasks. However, it is impossible for the quantile forecasting setting to pre-define a model with multiple target quantiles and train it in a single runtime. The flow chart of quantile predictions can be seen in Figure 12.

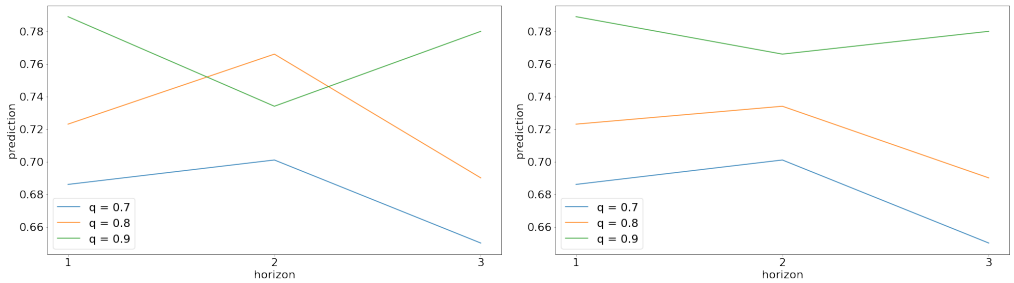
After predicting the test set's data, 99 different models predict the quantiles, so the predictions have to be sorted. Otherwise, a quantile crossing could occur; see the example in Figure 13 at step 2, where quantile 0.8 prediction "crosses" with quantile 0.9. In our work, the model makes a 24-hour prediction. Quantile crossing is undesirable,



(a) Quantile prediction flow (b) Single model flow

Figure 12. Quantile forecast step flow chart

significantly decreasing the forecast's quality and interfering with downstream tasks. It is important that quantiles are strictly increasing, meaning each subsequent prediction of the quantile has to be larger than the previous one. Therefore, Quantile predictions are sorted to fix quantile crossing as a part of the Quantile forecast step. In this case, because numerical precision in the runtime is good enough, it's unlikely that 2 quantiles would have exactly the same values.



(a) Quantile Crossing Example (b) Quantile Crossing Correction

Figure 13. Quantile Correction Toy example

3.1.2 Copula-based Scenario generation

Quantile forecasts of XGBoost can produce marginal forecasts. It could be thought of as independent scenarios on each lead time of the horizon. To produce meaningful scenarios, marginal forecasts must be coupled, and their interdependencies modelled. Therefore, the Gaussian Copula function is used to couple the marginal forecasts. Steps required to achieve such coupling are depicted in Figure 14.

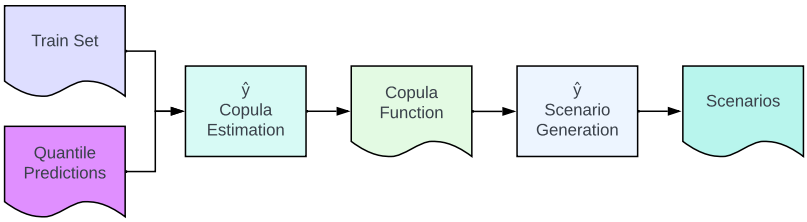


Figure 14. Copula flow chart

The Training set and quantile predictions are used as inputs to generate the copula function.

Copula Estimation is done in several steps:

1. An identity matrix is created, and samples of all training set days are provided
2. The covariance for a single day is calculated and added to the identity matrix with a certain forget factor.
3. The matrix is then standardized to ensure that all values fall in the range between 0 and 1.
4. Steps 2 and 3 are repeated for all the days in the training set.

Forget factor defines how much information is retained from the previous iteration and how much is added from a new day. It balances retaining enough information between iterations while picking new values from a sample. The end result of this step is a Copula Function.

The copula function is used in Scenario Generation to simulate multiple scenarios by drawing random samples from the copula distribution and transforming them using the input Quantile Predictions. This will create scenarios that are consistent with the input probabilistic estimates and capture the dependencies between the variables. This step produces scenarios evaluated with an Energy score and passed down to monetary evaluation.

4 Experiments

[DWL+22] used data to train multiple models capable of producing multiple outputs. The data preparation had to be done differently from the original paper while ensuring the train-test-validation split was the same to guarantee proper evaluation.

The only additional variable included in **Load** was an hour feature. Average load is highly dependent on the hour of the day, and while the transformation in [DWL+22] accounted for it by producing 24 different outputs, XGBoost in a quantile setting can take the hour of the day as a feature as an input. The month of the year has also been added as a feature and the average load of the predicted hour. However, both features had not produced a significant impact, as the month would only reflect the average temperature trend, which is directly accounted for by data inputs. In contrast, the average load of the hour seems to be captured well by the model during training.

Wind track had no significant change in data preparation compared to [DWL+22]. Model has been trained in a setting with just the original features of the GEFCom2014 dataset and with additional features created in [DWL+22] work. Those features are:

1. WS10 and WS100 - wind speed at altitudes of 10 and 100 meters, respectively
2. WE10 and WE100 - wind energy at altitudes of 10 and 100 meters
3. WD10 and WD100 - wind direction at 10 and 100 meters

The model outperformed the Variational Auto Encoder model from [DWL+22] both with and without these six additional features, but there has been a slight increase in average quantile and energy score results.

Solar track had similar features added to it as a Load track. The hour feature was as important as in the case of the Load track, while the month feature had no significant result on performance. The reasoning for the monthly feature was similar in both tracks: the data had significant differences across different months, but it has been captured well already by predicting solar irradiation of the surface of the earth for the hour.

The last important part of data preparation was splitting it for training and testing purposes. Since no hyperparameter tuning has been done in our work, the validation part of the split was not used. [DWL+22] has randomly picked 50 days for testing with a certain random state. To reproduce their split on the different formats of data transformation, the dates of those 50 test days have been picked directly from the original split and applied to the new data format, reproducing the original test set.

Multiple models have been trained on 99 different quantiles, from quantile 0.01 to quantile 0.99, with a step of 0.01.

Hyperparameters used are displayed in Table 2. These hyperparameters were hand-picked during several trial runs. The two important factors to balance between were forecasting quality and resource usage, aiming for a training time of 30 minutes per model

and performance better than [DWL+22] on quantile forecasting. Most importantly, the number of estimators and maximum depth were the biggest contributors to a performance increase.

The learning rate shrinks the contribution of each tree by the chosen value. Number of estimators refers to the number of trees generated during the boosting stages. It is limited only by hardware performance, as the XGBoost itself is pretty robust to overfitting. Maximum depth refers to the maximum number of consecutive leaves in a single regression estimator (tree). Minimum leaf samples refers to the number of samples in data required to create a leaf node. Minimum sample split refers to the minimum number of samples required to split the internal node (leaf).

name	value
learning rate	0.05
number of estimators	600
maximum depth	7
minimum leaf samples	9
minimum samples split	9

Table 2. Hyperparameters

The hardware and system used during training are displayed in Table 3. The system used has been a limiting factor during training time, as there is no GPU acceleration for XGBoost is limited to Linux-specific library. The Model used is GradientBoostingRegressor from the ensemble module from the Scikit-Learn library (version 1.0.1).

Operating system	Windows 10
RAM	16 GB
Processor	AMD Ryzen 5 5600 H
Model library	Scikit-learn v. 1.0.1

Table 3. System specifications

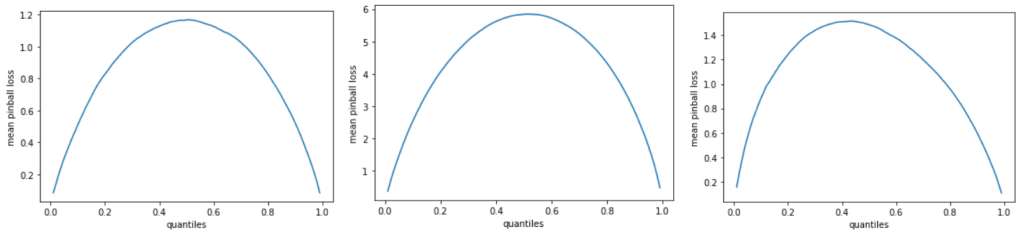
Forget factor of 0.99 has been chosen for generating a copula, which means that the matrix before the update step has a weight factor of 0.99, and the new sample has 0.01 weight in the new matrix. While values less than 0.99 produced unstable results, where covariances did not match between runs. This indicates that the last example impacted the final result too much.

To replicate the setup of [DWL+22], 100 scenarios have been sampled for each of 50 tested dates and assessed with Energy Score.

5 Results and discussion

5.1 Quantile score, copulas and energy score

5.1.1 Quantile score results



(a) Load, $\overline{QS}$ = 0.83 (b) Wind, $\overline{QS}$ = 4.22 (c) Solar, $\overline{QS}$ = 1.11

Figure 15. QS by quantile

Figure 15 shows each track's quantile scores per quantile. Importantly, all tracks have different scales, where (b) Wind has the highest maximum QS on marginal (≈6), while (a) Load has the lowest (≈1.2). This is also indicated by mean $\overline{QS}$, where again they are distributed from highest to lower in order: (b) Wind, (c) Solar, (a) Load.

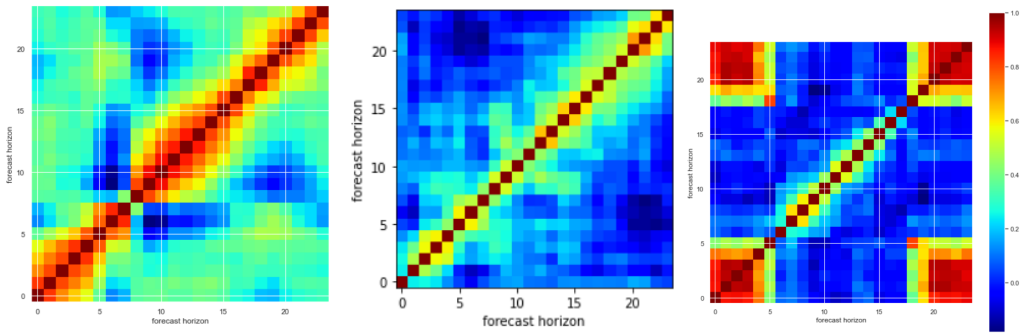
Notably, (c) Solar track differs from the other two, where the maximum QS value is skewed to lower quantiles (≈0.4). This could be due to the difference in the distribution of (c) the Solar dataset. Since there is a rather specific amount of maximum possible energy produced during the day, and there are a lot of days where energy production is at its maximum, the biggest contributor to uncertainty is the lower quantiles. Multiple potential factors could, in turn, cause smaller energy production. Therefore, it is harder to learn the exact values for produced energy. In addition, there could be multiple additional factors not accounted for in input variables, such as unexpected rainfall due to the changes in the wind direction. Therefore, the ballpark of errors concentrates in lower quantiles, resulting in the bias reflected in Figure 15 part (c).

5.1.2 Copula analysis

All tracks had very different distributions of covariances in copula functions, as they are created from covariance between hours of the day. Figure 16 contains the Estimated Covariance Matrix using Multivariate Gaussian Distribution.

Important note: separate covariance matrices for Wind and Solar tracks have been generated for each zone, improving performance from a 54 average energy score to 53 on the Wind track. Impact for a Solar track was less significant, as Zones are more similar for a Solar track. The three matrices in figures 16 represent Copula functions trained on

different tracks. They show the universal copula example, where a single copula have been trained for all zones of the track. Colours closer to brown-red represent higher covariances between hours, while blue represents low covariance. The colour gradient can be seen on the rightmost of the figure. The common feature between all three tracks



(a) Load (b) Wind (c) Solar

Figure 16. Universal copula for each track

is a diagonal of brown colour (covariance = 1) because the covariance of an hour with itself will always be 1. However, that is the only similarity; otherwise, they differ in every other way.

A **Load** track has the highest overall covariance between neighbours inside three distinct "regions", i.e. parts of the day:

1 0-8 hours - night and morning, where load typically is the lowest, as most people sleep during this time or start their day off, which leads to higher consumption around 6-8, and higher covariance during these hours.

2 9-16 hours - are the typical day work hours, where businesses and public establishments are the main drivers of power consumption. Since it has high covariance between neighbours in this region, we can conclude that consumption is pretty stable during this time of the day. During summer, this is the hottest time of the day, necessitating air conditioning and cooling of houses and workplaces.

3 17-23 hours - evening hours, most of the consumption comes from people using more home appliances during this time and requires indoor and outdoor lighting. Covariance is lower relative to the previous time of the day, meaning consumption is less stable throughout these hours.

B **Wind** track overall has a smaller degree of correlation between hours of the day, as it is more chaotic due to the less predictable nature of wind. However, it's

a non-zero correlation level for 3-6 neighbouring hours, so the weather usually stays windy for certain periods. The copula also indicates that some particular hours slightly increase in correlation (2-4, 5-6, 12-15, 20-23), possibly due to local weather patterns.

C **Solar** track is quite distinct from the other two. It has a very high degree of correlation for the first 4-5 hours and the last 5-6 (depending on the time of the year). This is because no power is produced during the nighttime. However, the correlation for the sunny part of the day is relatively low because solar power production has a high natural dependency on the angle of sunlight and the absence of clouds. Both factors can change relatively quickly, thus leading to an overall low correlation during the productive part of the day.

5.1.3 Difference in energy score between zones

Wind track has significant differences in performance between zones, which can be seen in Figure 17. While Solar has a relatively lower gap between zones, shown in Figure 18.

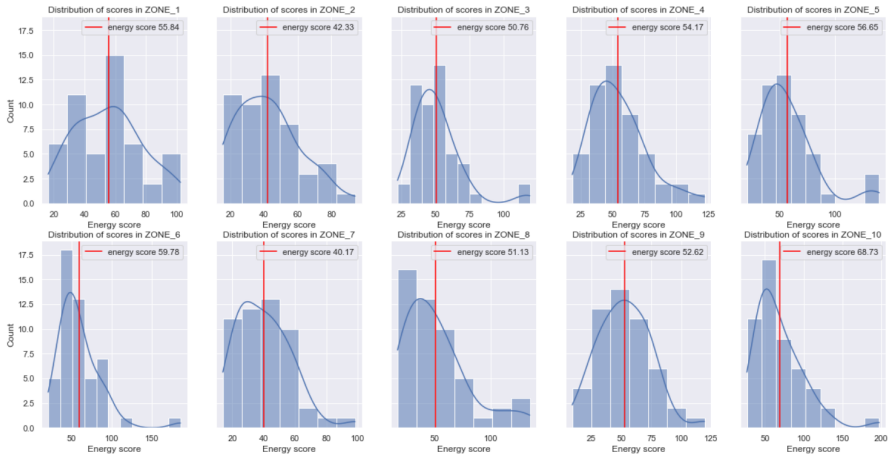


Figure 17. Wind zones energy scores

It is not obvious why Wind zones are so wildly different. It is easy to see that better-performing zones, like 2 and 7, do not have very bad days, while Zone 10 has both worse days on average and one of the worst days captured for all zones overall. This could be due to either unfavourable positioning of the wind turbine, compared to others, or due to mechanical failure on the day. However, it also might be an artefact of a relatively small test subset of only 50 samples, and otherwise, these zones are more comparable with one another.

Figure 18. Solar zones energy scores

5.1.4 Analysis of the best vs worst days and zones

The load track does not have different zones, so only the best and worst days have been analyzed. You can see them in Figure 19. Pretty much the only difference between them is how far the predicted marginal of scenarios is from the actual ground truth. This lowered the overall accuracy of the forecast for the worst day, as true distribution was very far from predicted quantiles, leading to a bad energy score as well.

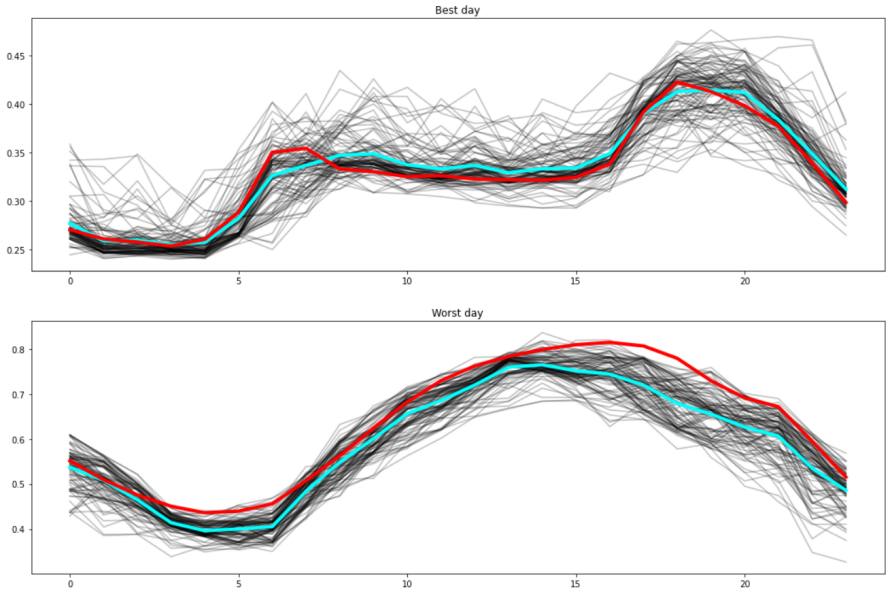


Figure 19. Best and worst days of load track

Wind track has, in general, less of a stable performance than other tracks, and it can be seen in both zones comparison and differences in days in best as well as worst

performing ones in Figure 20. Wind features are plotted in the same graphs for a daily forecast, importantly on a different scale, depicted on the right of the graph. On the worst day of the worst zone, it could be argued that the weather features did not reflect the power production of the day, as features have little to no correlation with actual power produced, while the model was predicting power output relying on the input wind speeds. The situation is quite different for the worst day of the best zone. The wind features have changed rather dramatically between hours 10 and 16 and predicted scenarios have underestimated power output potential.

The situation is fairly similar for the best-performing days of both zones, where the low energy score results from a good approximation of low power production for the day due to forecasted slow winds.

Interestingly, the worst-performing zone copula function graph has a higher correlation at the end of the day, hours 14-24. Perhaps, that might be due to the weather characteristics of the region, which are very different from the other 9 Zones. Due to the model being trained on all examples at once, not being split by zone-basis, it has been a primary contributor to high energy scores. This can also be seen in Figure 17.

5.2 Results comparison

The paper [DWL$^+$22] have used several scoring methods. However, the most applicable to scenario forecasting are the mean quantile score ($\overline{QS}$) and the mean energy score ($\overline{ES}$). Quantile score is a Pinball loss, averaged across all quantiles for a single hour, while Mean Quantile Score is the average of all Quantile Scores of the test set. A single Energy Score entry is assessed for scenarios predicted for a single day, and the Mean Energy Score is averaged over all days in the test set. Table 4 shows the highest performing models from the paper, Normalizing flows (NF) and Variational Autoencoder (VAE), results of random prediction (RAND), and the one tried in our work, Regularizing Gradient Boosting (XGBoost). The comparison has not included Generative Adversarial Network (GAN), as it showed the lowest performance across all tracks in [DWL$^+$22]. XGBoost has outperformed all models on Wind and Solar tracks. However, it hasn't

Track	Score	XGBoost	NF	VAE	RAND
Wind	$\overline{QS}$	**4.22**	4.58	4.45	8.55
	$\overline{ES}$	**53.26**	56.71	54.82	96.15
Solar	$\overline{QS}$	**1.11**	1.19	1.31	2.48
	$\overline{ES}$	**21.7**	23.08	24.65	41.53
Load	$\overline{QS}$	0.83	**0.76**	1.39	3.40
	$\overline{ES}$	9.76	**9.17**	15.11	38.08

Table 4. Score comparison

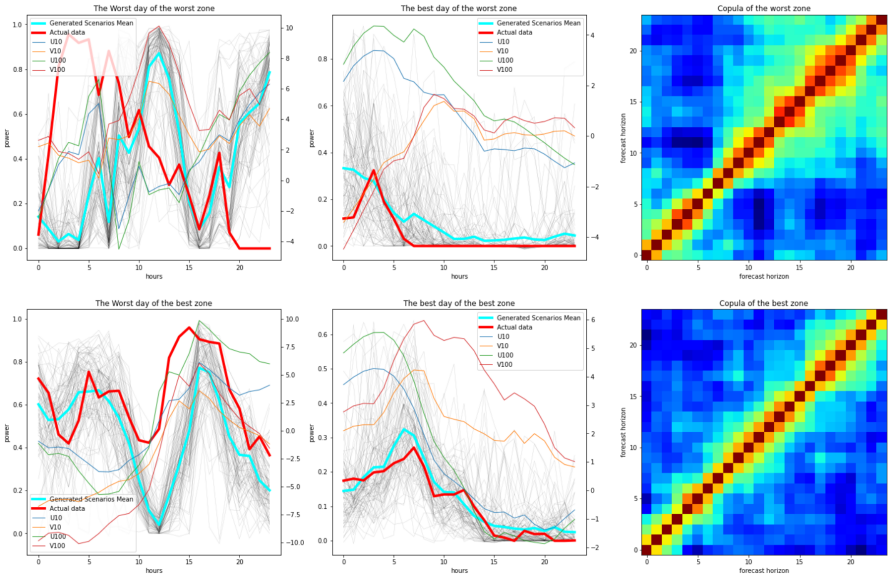


Figure 20. Best and worst days and zones of wind track

The solar track had much less of a difference between the zones in terms of performance, and it had failed in a similar way on the worst forecasted days — by predicting high production on a bad day. It could have been due to general cloudiness, rain, or some other natural cause. For example, dust particles lifted by wind could decrease overall effectiveness.

The problem with solar track is we do not have direct features to predict such events. Weather forecast of **wind speed** and **likelihood of clouds or precipitation** would substantially increase forecast quality, especially in cases of rain or dust storms. The probability of these events can be used as a feature in the model since it already operates in a probabilistic setup. It is not a given that the model would be able to work with such features well, but inclusion of such features could be a good idea for future experiments.

outperformed the Normalizing Flows model on Load. Though, it is important to note that it has very close results, much closer than Variational Autoencoder. Since fine-tuning hasn't been done for this thesis due to resource limitations, it might be the case that XGBoost could outperform NF with proper technique.

The work of [DWL+22] has also implemented an economic assessment method, which considers planned production, which is the prediction results from Solar and Wind tracks, and price per kilo-Watt $\times$ hour, which is the prediction results for consumption. You can see the results of this optimization in Table 5.

XGBoost has outperformed both the best version of Normalizing Flows (NF-UMNN) and Variational Autoencoders models by 15 thousand Euros. It is a good practical demonstration of how relatively low and seemingly insignificant increases in test scores

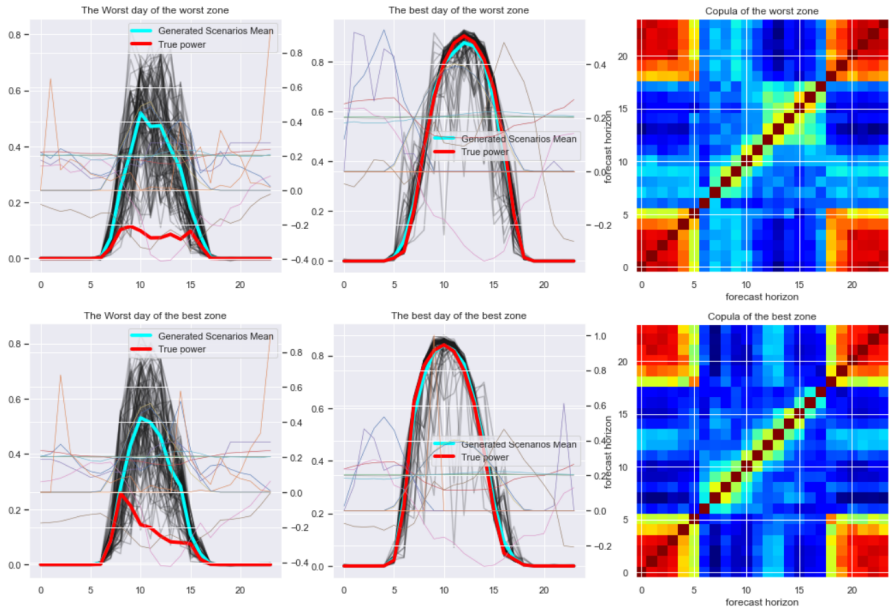


Figure 21. Best and worst days and zones of solar track

XGBoost	NF-UMNN	NF-A	VAE	RAND	O
122	107	101	97	-181	298

Table 5. Profit per model, k€

can result in substantial economic benefits. The random column (RAND) indicates profit for random predictions, which would result in 181 thousand Euros loss. While RAND does not necessarily indicate the maximum possible loss, it still gives an idea about the potential economic damage of a bad model. Oracle (O) column indicates the maximum possible profit if the model had perfect knowledge of the future. It is not perfect, as it doesn't account for uncertainty within data, but it gives a hard upper limit on possible profit. Both random (RAND) and Oracle (O) assessments are part of the monetary assessment that has been created in [DWL+22].

5.3 Resource usage

Training 99 XGBoost quantile models took different amounts of time for each track. The fastest was the Load track, which can be trained in 3.5 hours, with the Solar track being a close second - around 4 hours. Wind track took much longer to train, with 12 hours per experiment.

The assessment of scenarios for downstream tasks by Gurobi optimization software took around 2 hours per experiment.

Overall, the slow training process and inability to train in parallel prevented using a validation set for hyperparameter tuning.

6 Conclusion

After assessing differences between tracks and performance, it can be easily seen that XGBoost models outperformed models presented in [DWL+22] paper. However, the performance on the Load track has been slightly lower than the best-performing model. This could be improved with some model adaptations and tuning. Nevertheless, acquired results can be used as a competitive baseline for future scenario forecasting research, especially those focused on Decision Tree models.

Since the copula-based approaches are the most studied and well-established for scenario generation, combined with high-performance models, such as boosting algorithms, they produce highly accurate predictions. Therefore, we propose using the Copula-based Boosting pipeline as a baseline for future scenario generation model proposals, as they produce very competitive results, only beating which the new models could claim **State of the Art** status.

The first main issue encountered in this work was the adaptation of [DWL+22] code, which was quite specific to the models developed in their paper. The second problem was choosing the XGBoost library to work with. Sklearn did not provide the functionality for a custom validation set, while [CG16] does not yet provide optimizations for the quantile loss.

The code to perform the experiments for this work can be checked in the following link: `https://github.com/lomayka/MSthesis`.

7 Future work

Decision tree algorithms could also be further improved for probabilistic forecasting. Energy Score could be used as a loss function to train the distribution directly. Another possible route is to sample voting trees to assess certain quantiles.

The field of energy forecasting benefits greatly from improvements in forecasting techniques and models. With the ongoing development of XGBoost library version 2.0 [CG16] and optimization of the training process for quantile forecasting, it would be possible to use a validation set during training and train multiple quantiles in a single runtime. This should increase the overall performance of XGBoost in a quantile forecasting setting.

References

[BLS+21] Vadim Borisov, Tobias Leemann, Kathrin Seßler, Johannes Haug, Martin Pawelczyk, and Gjergji Kasneci. Deep neural networks and tabular data: A survey. *CoRR*, abs/2110.01889, 2021.

[BP10] Gérard Biau and Benoît Patra. Sequential quantile prediction of time series, 2010.

[Bre01] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[CG16] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. *CoRR*, abs/1603.02754, 2016.

[DWL+22] Jonathan Dumas, Antoine Wehenkel, Damien Lanaspeze, Bertrand Cornélusse, and Antonio Sutera. A deep generative model for probabilistic energy forecasting in power systems: normalizing flows. *Applied Energy*, 305:117871, 2022.

[FHT00] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Additive logistic regression: a statistical view of boosting (With discussion and a rejoinder by the authors). *The Annals of Statistics*, 28(2):337 – 407, 2000.

[Fri02] Jerome H. Friedman. Stochastic gradient boosting, 2002. Nonlinear Methods and Data Mining.

[GPAM+14] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[GR07] Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

[HPF+16] Tao Hong, Pierre Pinson, Shu Fan, Hamidreza Zareipour, Alberto Troccoli, and Rob J. Hyndman. Probabilistic energy forecasting: Global energy forecasting competition 2014 and beyond. *International Journal of Forecasting*, 32(3):896–913, 2016.

[KSJ+16] Diederik P. Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improving variational inference with inverse autoregressive flow, 2016.

[MCM+14] Juan Miguel Morales González, Antonio J. Conejo, Henrik Madsen, Pierre Pinson, and Marco Zugno. *Integrating Renewables in Electricity Markets: Operational Problems*. International Series in Operations Research and Management Science. Springer, 2014.

[PMN+09] Pierre Pinson, Henrik Madsen, Henrik Nielsen, Georgios Papaefthymiou, and Bernd Klöckl. From probabilistic forecasts to statistical scenarios of short-term wind power production. *Wind Energy*, 12:51 – 62, 01 2009.

[QHD+20] Yuchen Qi, Wei Hu, Yu Dong, Yue Fan, Ling Dong, and Ming Xiao. Optimal configuration of concentrating solar power in multienergy power systems with an improved variational autoencoder. *Applied Energy*, 274:115124, 2020.

[SR13] Gábor J. Székely and Maria L. Rizzo. Energy statistics: A class of statistics based on distances. *Journal of Statistical Planning and Inference*, 143(8):1249–1272, 2013.

Appendix

I. Licence

Non-exclusive licence to reproduce thesis and make thesis public

I, **Denys Kolomiiets**,
*(*author's name)

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to

   reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

   A Competitive Scenario Forecaster using XGBoost and Gaussian Copula,
   *(*title of thesis)

   supervised by Meelis Kull and Novin Shahroudi.
   *(*supervisor's name)

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe on other persons' intellectual property rights or rights arising from the personal data protection legislation.

Denys Kolomiiets
09/05/2023