

Multi-Site PV Forecasting Across Climate Zones and Prediction Horizons



Niamh McMullin

Division of Industrial Electrical Engineering and Automation
Faculty of Engineering, Lund University

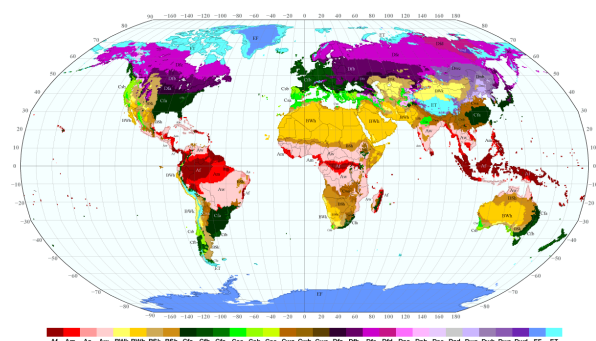
Öken, medelhav eller mulet - varför klimatet avgör hur väl vi kan förutspå solkraft

POPULÄRVETENSKAPLIG SAMMANFATTNING AV **Niamh McMullin**

ALLT FLER SOLPANELER TÄCKER VÄRLDENS TAK OCH FÄLT - MEN HUR VÄL KAN VI EGENTLIGEN FÖRUTSPÅ DERAS PRODUKTION? I DETTA ARBETE HAR OLIKA PROGNOSEMODELLER UTVÄRDERATS ÖVER BÅDE KLIMATZONER OCH TIDSHORISONTER, FÖR ATT GE EN MER HELTÄCKANDE BILD AV HUR TILLFÖRLITLIGA DESSA MODELLER FAKTISKT ÄR. RESULTATET ÄR TYDLIGT: MODELLVALET SPELAR FÖRVÅNANSVÄRT LITEN ROLL JÄMFÖRT MED KLIMATZONEN DÄR PROGNOSEN GÖRS.

I takt med den gröna energiomställningen har intresset för solkraft ökat. Detta kanske låter bekant - plötsligt har grannen, kollegan, ja kanske till och med du själv, solpaneler på taket. Problemet med solkraft är att produktionen varierar kraftigt beroende på väder och årstid, vilket kan orsaka driftstörningar i elnätet. Därför har prognostisering av solkraftsproduktion blivit allt viktigare inom forskningen - bättre prognoser kan nämligen bidra till ett säkrare och mer stabilt elnät.

I detta examensarbete har olika prognosmodeller utvärderats över både klimatzon och tidshorisont. Medan tidshorisonter är viktiga för elmarknaden - där till exempel dygnsprognoser används för dagen-före-marknaden, och kortare prognoser används för intradagsmarknaden - så jämförs olika klimatzoner för att undvika att dra slutsatser som är specifika för en enskild solcellsanläggning eller ett begränsat geografiskt område.



För att göra denna jämförelse krävs ett sätt att kategorisera världens klimat. Här används Köppen-Geiger-systemet, som delar in världen i 25 olika klimatzoner. I arbetet har data från fyra klimatzoner använts: de soliga och torra klimaten i Australien och Portugal, samt de mer variabla och molniga klimaten i USA och Kanada.

Resultaten visar att modellvalet spelar förvåansvärt liten roll - avancerade neurala nätverk presterar ungefär lika bra som enklare modeller som Random Forest. Den stora skillnaden finns istället mellan klimatzonerna: de soliga och torra klimaten i Australien och Portugal ger genomgående bättre prognoser än de mer variabla klimaten i USA och Kanada, och detta gäller över i princip alla månader och modeller. Ett tydligt mönster är att regnigare säsonger konsekvent är svårare att förutsäga - ökat molntäcke och snabba väderskiften introducerar en oförutsägbarhet som ingen modell riktigt klarar av att hantera.

Vad gäller tidshorisonter pekar resultaten mot samma slutsats oavsett om prognosen gäller nästa kvart eller hela nästa dag: felet följer solens naturliga dagscykel och är i genomsnitt störst mitt på dagen. En intressant iakttagelse är att modellerna tenderar att anta att nästa dag blir solig och molnfri när de saknar tillgång till framtida väderdata - ett slags optimistiskt standardantagande som kan ge stora fel när verkligheten ser annorlunda ut.

I soliga klimat är detta standardantagande nästan självkorrigering - klara dagar är ju just vad dessa klimat oftast erbjuder. Men även när modellerna förses med väderprognoser kvarstår fördelen: väderprognoser är troligtvis mer tillförlitliga i soliga klimat, eftersom det är enklare att förutspå ytterligare en solig dag än att avgöra exakt när ett regnband drar in.

Sammantaget tyder resultaten på att klimatet där en solcellsanläggning befinner sig inte bara avgör hur svårt det är att göra en bra prognos, utan även hur snabbt felet växer ju längre fram i tiden man försöker blicka. I stabila, soliga klimat är behovet av avancerad prognosinfrastruktur därmed begränsat. I variabla klimat, däremot, är osäkerheten i dagen-före-prognosen sannolikt så pass stor att intradagsmarknaden spelar en viktig roll som korrigeringsmekanism, och värdet av bättre prognosverktyg blir därför högre.

Multi-Site PV Forecasting Across Climate Zones and Prediction Horizons

Niamh McMullin

Thesis for the degree of Master of Science in Engineering Physics



LUND
UNIVERSITY

Supervisor Huan Li
huan.li@iea.lth.se

Examiner Ferdinand Grimm
ferdinand.grimm@iea.lth.se

Abstract

In light of the ongoing energy transition, solar photovoltaic (PV) systems are becoming increasingly widespread. The intermittent nature of solar power introduces challenges to grid operation, which is why PV power forecasting has emerged as an important research topic. While existing research mostly focuses on single-site forecasting, this thesis compares sites from four different Köppen-Geiger climate zones to assess geographical generalisability, alongside prediction horizons relevant to the day-ahead and intraday market.

Six models are compared: Naïve Persistence, Random Forest, XGBoost, Ridge Regressor, SVR and LSTM, with all hyperparameters optimised using Optuna. While differences between models are modest, forecasting accuracy varies substantially across climatic regions. The sunnier zones, BWh (Subtropical desert) and Csa (Mediterranean), consistently outperform the more variable BSk (Mid-latitude steppe) and Dfb (Humid continental). Within each zone, accuracy deteriorates during rainier seasons due to increased cloud cover and weather variability.

Regarding forecast horizons, all models outperform Persistence at 15 minutes but underperform it at 24 hours. A Rolling Forecast framework reveals that LSTM forecasts converge towards a generic diurnal PV profile when future weather data is unavailable. A Perfect Rolling Forecast framework using a Bi-LSTM with perfect weather inputs isolates meteorological uncertainty as a primary driver of horizon-dependent accuracy loss.

The findings suggest that climatic context is a more fundamental determinant of forecasting accuracy than model choice, and that meteorological variability drives forecast error regardless of horizon or model complexity - contributing to a more generalised understanding of PV forecasting across diverse climatic conditions.

Sammanfattning

I den pågående energiomställningen blir solcellsanläggningar (PV) allt vanligare. Solcellers intermittenta natur leder till utmaningar i nätet kopplade till driftsäkerhet, vilket är varför PV-prognostisering har blivit ett viktigt forskningsområde. Medan mycket forskning redan finns på prognostisering av enskilda solcellsanläggningar jämför denna masteruppsats anläggningar från fyra olika Köppen-Geiger klimatzoner för att analysera geografisk generaliserbarhet, tillsammans med tidshorisonter relevanta för dagen före- samt intradagsmarknaden.

Prognostiseringsprestanda över klimatzoner undersöks med hjälp av sex modeller: Naïve Persistence, Random Forest, XGBoost, Ridge Regressor, SVR och LSTM, där alla hyperparametrar optimeras med Optuna. Skillnaderna mellan modellerna är relativt små, medan en betydligt större skillnad finns mellan klimatzonerna. De soligare klimaten BWh (Varmökenklimat) och Csa (Medelhavsklimat) presterar genomgående bättre än de mer variabla BSk (Kallhalvökenklimat) och Dfb (Fuktigt kontinentalklimat). Prognostiseringsprestandan försämras också markant under regnperioder, då ökad molntäckning och snabba väderväxlingar leder till större oförutsägbarhet.

Vad gäller tidshorisonter överträffar samtliga modeller Persistence för 15-minutershorisonten men presterar sämre för 24-timmarshorisonten. Ett Rolling Forecast-ramverk visar att LSTM-prognoser tenderar att konvergera mot en generisk daglig PV-profil utan tillgång till framtida väderdata. Ett Perfect Rolling Forecast-ramverk, där Bi-LSTM förses med perfekt väderinformation, isolerar meteorologisk osäkerhet som en primär drivkraft bakom horisontberoende noggrannhetsförsämring.

Resultaten tyder på att klimatologiskt sammanhang är en mer grundläggande faktor för prognostiseringsnoggrannhet än modellval, och att meteorologisk variabilitet fungerar som en oberoende felkälla oavsett horisontlängd eller modellkomplexitet - vilket bidrar till en mer generaliserad förståelse av PV-prognostisering över varierande klimatförhållanden.

Acknowledgements

This master's thesis marks the final part of my degree in Engineering Physics at Lunds Tekniska Högskola (LTH), and was carried out at the Division of Industrial Electrical Engineering and Automation.

I would like to express my deepest gratitude to my supervisor Huan Li, who has supported me throughout this process with regular meetings and valuable insights. I would also like to thank my friends and family for their continuous support and encouragement.

This thesis has been a pleasure to write, where I have had the opportunity to broaden my knowledge and feel inspired by the endless opportunities that the energy field has to offer.

Lund, May 2026

Niamh McMullin

Contents

1	Introduction	8
1.1	Purpose and Research Questions	8
1.2	Delimitations	9
1.3	Disposition	9
2	Literature Review	10
3	Theoretical Background	12
3.1	Solar Photovoltaic (PV) Systems	12
3.1.1	How Photovoltaic Cells Work	12
3.1.2	Factors Affecting PV Output	12
3.1.3	Grid Integration of PV Systems	13
3.1.4	Battery Storage in PV Systems	13
3.2	Energy Markets	14
3.2.1	Day-Ahead Market	14
3.2.2	Intraday Market	14
3.3	Köppen-Geiger Climate Classification	15
3.4	Forecasting Models	16
3.4.1	Persistence Models	16
3.4.2	Random Forest	16
3.4.3	eXtreme Gradient Boosting	17
3.4.4	Ridge Regression	17
3.4.5	Support Vector Regression	18
3.4.6	Long Short-Term Memory	19
3.4.7	Bi-Directional Long Short-Term Memory	21
3.5	Hyperparameter Optimisation	21
3.5.1	Optuna	21
3.5.2	Tree-structured Parzen Estimator Algorithm	22
3.5.3	Time Series Cross-Validation	22
3.6	Scaling	23
3.7	Evaluation Metrics	23
4	Data Collection	24
5	Data Preprocessing	26
5.1	Australia Site	26
5.1.1	Missing Timestamps and Values	27
5.1.2	Exploration of Local Weather Variables	28
5.1.3	Importing Data from JRC	29
5.1.4	Merging Data	29
5.1.5	Feature Engineering and Resampling	30
5.2	USA Site	32
5.2.1	Daylight Saving Time	33
5.2.2	Missing Timestamps and Values	33
5.2.3	Importing Data from JRC and Merging Data	33
5.2.4	Feature Engineering and Resampling	34
5.3	Portugal Site	35
5.3.1	Missing Timestamps and Values	36
5.3.2	Exploration of Local Weather Variables	36
5.3.3	Importing Data from JRC and Merging Data	36

5.3.4	Daylight Saving Time	36
5.3.5	Feature Engineering	38
5.4	Canada Site	39
5.4.1	Missing Timestamps and Values	40
5.4.2	Importing Data from JRC and Merging Data	40
5.4.3	Daylight Saving Time	41
5.4.4	Feature Engineering	41
5.5	Feature Selection	41
6	Forecasting Performance Across Climate Zones	46
6.1	Methodology	46
6.1.1	Naïve Persistence	46
6.1.2	ML Models	46
6.1.3	LSTM (DL)	48
6.2	BWh - Australia	50
6.3	BSk - USA	51
6.4	Csa - Portugal	53
6.5	Dfb - Canada	54
6.6	Comparative Analysis	56
6.6.1	Cross-Country Performance	56
6.6.2	Impact of Enhanced Resolution and Features	60
7	Forecasting Performance Across Time Horizons	62
7.1	Exploratory Horizon Experiments	62
7.1.1	Methodology	62
7.1.2	15-Minute Horizon	64
7.1.3	24-Hour Horizon	65
7.2	Rolling Forecast Strategy	66
7.2.1	Methodology	67
7.2.2	Day-Ahead Forecasting	68
7.2.3	Intraday Forecasting	69
7.2.4	Remarks on Forecast Performance	70
7.3	Rolling Forecast Strategy with Perfect Weather Information	70
7.3.1	Methodology	70
7.3.2	Day-Ahead Forecasting	71
7.3.3	Intraday Forecasting	71
8	General Discussion	73
9	Conclusions	74
10	Future Work	75
	Appendix	80
A	Example of Decision Tree	80
B	Pearson Correlation Heatmaps	81
C	Pairplots	87

Terminology

- AC = Alternating Current
- AI = Artificial Intelligence
- ANN = Artificial Neural Network
- BESS = Battery Energy Storage Systems
- Bi-LSTM = Bi-directional Long Short-Term Memory
- DA = Day-Ahead market
- DC = Direct Current
- DL = Deep Learning
- DST = Daylight Saving Time
- EI = Expected Improvement
- JRC = Joint Research Centre (European Commission)
- LSTM = Long Short-Term Memory
- MAE = Mean Absolute Error
- ML = Machine Learning
- MTU = Market Time Unit
- nRMSE = normalised Root Mean Square Error
- PV = Photovoltaic
- PVGIS = Photovoltaic Geographical Information System
- RF = Random Forest
- RMSE = Root Mean Square Error (RMSE)
- RNN = Recurrent Neural Network
- sMAPE = symmetric Mean Absolute Percentage Error
- SVR = Support Vector Regressor
- TPE = Tree-structured Parzen Estimator
- XGBoost = eXtreme Gradient Boosting

1 Introduction

In the midst of climate change, fossil fuel energy resource depletion and an increase of global energy consumption [1], the energy transition is critical. The sustainable energy transition refers to a structural transformation of energy production, distribution, and consumption, aimed at replacing fossil fuels with renewable energy sources. [2] As part of the energy transition, distributed energy resources, including solar photovoltaic (PV) systems, are becoming increasingly widespread. However, the intermittent and fluctuating nature of solar energy introduces significant challenges to the safe and stable operation of the power grid [3]. PV power forecasting has therefore emerged as one of the key approaches to address these challenges and is receiving increased research interest. [4]

PV power production is affected by various factors, including solar irradiation, shading, soiling, temperature, degradation of PV module, PV module tilt angle, and more. [5] Some of these factors are weather-dependent and vary across time and location, causing power swings and system instability. [1] Accurate short-term forecasting of PV generation is therefore crucial, as it enables smarter operational decisions - such as when to store energy in batteries, when to feed it into the grid, or when to curtail production - thereby enhancing the stability and efficiency of micro grids. Different machine learning approaches have been proposed for accurate short-term PV forecasting in recent years, including statistical models [6], Machine Learning (ML), Deep Learning (DL) [7], and hybrid approaches [8].

A problem with statistical models, such as Autoregressive, Moving Average and ARIMA, is that they can have difficulties with finding complex non-linear relationships and adjusting to fast-changing variables (such as wind speed or sunlight exposure). [6] AI models are often much better at this, making them superior when handling non-linear, high-dimensional energy data. [9] AI models can be divided into ML and DL models. ML typically refers to algorithms that learn patterns from data using features engineered by humans. DL, on the other hand, represents a more hierarchical and organised form of learning, where Artificial Neural Networks (ANNs) with multiple hidden layers automatically extract features and learn complex, non-linear relationships. [7] Hybrid approaches refer to a combination of several deep learning and machine learning models. [8]

While a lot of research already exists on PV forecasting, many papers focus on single sites, limiting geographical generalisability. [10] Climatic conditions such as solar elevation variability, temperature, seasonal patterns and cloud coverage differ significantly across regions. Therefore, comparing model performance across multiple climatic contexts is essential to understand whether forecasting approaches capture underlying physical and statistical relationships or merely adapt to site-specific characteristics.

Furthermore, prediction horizons play an interesting role in energy forecasting. Participation in day-ahead markets requires forecasts to be submitted up to 36 hours before delivery [11], while intraday markets allow for shorter-notice adjustments as real-time conditions evolve [12]. This distinction has direct implications for the value of a forecast: a highly accurate short-term prediction may be of limited use if it arrives too late to inform a day-ahead bid, while a longer-horizon forecast - even if less precise - may still carry significant value by enabling timely market participation. These market structures therefore impose concrete requirements on the temporal reach of forecasting models, and motivate a systematic assessment of how forecasting performance evolves as the prediction horizon extends.

1.1 Purpose and Research Questions

The purpose of this thesis is to analyse how model accuracy is affected by varying climatic conditions and increasing prediction horizons. While horizon length is directly linked to DA and intraday market applications, the inclusion of multiple climates allows for evaluating model robustness and generalisability under structurally different weather dynamics. This is important to avoid conclusions that are site-specific and to assess whether certain models remain stable under different levels of meteorological variability. The thesis will therefore examine how forecasting difficulty - introduced through both longer horizons and climate heterogeneity - impacts model performance.

The research questions for this thesis are the following:

- **RQ1:** How do different machine learning forecasting models perform across PV sites located in different climatic regions?
- **RQ2:** How does forecast accuracy vary across different prediction horizons?

1.2 Delimitations

Given time constraints and data availability, some delimitations were made.

Only four datasets were included, each representing a distinct climate zone. Since data collection, preprocessing and model training associated with each dataset is time-intensive, four datasets was determined to be a practical upper bound. It should be noted that only a single dataset is used per climate zone; it is therefore assumed that each dataset is reasonably representative of its respective climate, acknowledging that forecasting performance may also be influenced by site-specific factors such as local data quality and instrumentation.

Beyond the choice of datasets, only a few forecasting models were selected for comparison. These were identified through a literature review as strong candidates for short-term PV forecasting, and no claim is made that they constitute an exhaustive set. Rather than optimising individual models beyond the hyperparameter tuning performed via Optuna - which employed a TPE sampler with RMSE as the objective metric - the aim was to provide each model with an equal opportunity to reach its potential, prioritising fair comparison over peak performance. Concretely, this means that hyperparameter tuning via Optuna was applied consistently across all models using the same number of trials, search strategy, and objective metric, rather than manually fine-tuning individual models beyond what this process yielded.

Regarding forecast horizons, the study was deliberately scoped to horizons relevant for day-ahead and intraday market participation, meaning that both very short and longer-term forecasts were excluded.

1.3 Disposition

The overall structure of the thesis is presented below:

- **Section 3 (Theoretical Background):** This section presents all relevant theory used in the thesis.
- **Section 4 (Data Collection):** This section describes how data was collected, where all datasets originate from, and a description of each PV site and location.
- **Section 5 (Data Preprocessing):** This section provides a thorough description of how each dataset was preprocessed, and an explanation of how features were selected.
- **Section 6 (Forecasting Performance Across Climate Zones):** This section addresses RQ1 by evaluating forecasting performance across all climate zones, followed by a cross-country analysis to identify whether patterns generalise beyond individual contexts. The impact of feature selection and temporal resolution is then examined to disentangle the factors driving model performance.
- **Section 7 (Forecasting Performance Across Time Horizons):** This section addresses RQ2 by evaluating how forecasting accuracy varies across different prediction horizons. Initially, only two time horizons (15 minutes and 24 hours) are evaluated in the so-called Exploratory Horizons Experiments, and the findings are then used to develop the primary Rolling Forecast framework. Finally, future weather information is added and the performance is assessed once more.
- **Section 8 (General Discussion):** This section discusses the results from the previous two sections, and how they interplay with each other.
- **Section 9 (Conclusions):** This section aims to conclude the findings made in the thesis.
- **Section 10 (Future Work):** This section proposes areas for future research.

2 Literature Review

Solar PV power forecasting has received considerable research attention in recent years, with a wide range of machine learning and deep learning models proposed and evaluated. However, the majority of existing studies focus on a single site or a geographically limited region, and few systematically examine how forecasting performance varies across both climatic conditions and prediction horizons.

Two studies come close to the scope of this thesis by addressing both climate diversity and multiple prediction horizons. Carlak and Karabanova [13] evaluated three machine learning models - ANN, SVR, and Random Forest - for weather forecasting across two climatically distinct locations, a Mediterranean site in Turkey and a temperate continental site in Russia, with forecast horizons ranging from 1 to 10 days ahead. While the study provides valuable insight into how algorithm-climate interactions affect forecasting accuracy, it targets meteorological variables such as temperature, wind speed, and global horizontal irradiance rather than PV power output directly, and covers only two climate zones. Aduama et al. [14] introduced a leave-one-region-out benchmark for PV power forecasting across eight globally distributed sites spanning diverse Köppen climate zones, evaluating LSTM, Basic Transformer, and Temporal Fusion Transformer architectures across horizons of 1 to 6 hours. The study demonstrates that model performance varies considerably across climatic regions and that architecture choice should reflect geographic context. However, the focus on transfer learning and short-term horizons of up to 6 hours leaves the question of day-ahead and intraday forecasting performance across climate zones largely unaddressed.

To the authors' knowledge, no additional studies have simultaneously examined PV forecasting performance across both climate zones and prediction horizons. Related work has instead addressed these dimensions separately, as presented in Table 1. This indicates a clear gap in the literature, which this thesis aims to address.

Table 1: Summary of reviewed PV forecasting studies. Climate Diversity indicates whether the study sites are located in different Köppen-Geiger climate zones.

Study	Sites	Climate Diversity	Horizons	Models	Evaluation Metrics
[13]	2	Yes	1 to 10 days	ANN, SVR & RF	RMSE, MAE & R^2
[14]	8	Yes	1 to 6 hours	LSTM, Basic Transformer, TFT	MAE, RMSE & nRMSE
[15]	4	Yes	No	BiLSTM-LSTM	MSE, MAE, RMSE, R^2
[16]	6	Partially, within Italy	1 to 3 days	Analog Ensembles, Deterministic Model, Hybrid model with MLP & kNN/QRF	RMSE, MAE & MBE
[17]	25	Partially, within USA	5 mins, 1 hour, 1 day & 3 days	(s)ARIMA, (s)ARIMAX, MLR, SVR, PSO, Average & Re	MASE
[18]	68	Partially, within Cyprus	1 to 73 days	62 statistical models & 27 deep learning models	RMSE, nRMSE & MAE
[19]	1	No	1, 6, 12 & 24 hours	LSTM, GRU & hybrid LSTM-GRU	MAE, MSE & R^2
[20]	1	No	1 day, 1 week, 2 weeks & 1 month	LR, Bayesian Regression, SVR, ANN RF, XGBoost, RF-ANN, RF-XGBoost & XGBoost-ANN	RMSE, MAE & R^2

3 Theoretical Background

The following section presents the theoretical background of this thesis. Where mathematical formulations are provided, they describe the underlying algorithms of the models used - as implemented in `Scikit-learn` and `TensorFlow/Keras` - rather than custom implementations. All variables and expressions are therefore inherent to the respective algorithms, and are included to aid the reader's understanding of how each model operates internally.

3.1 Solar Photovoltaic (PV) Systems

3.1.1 How Photovoltaic Cells Work

Solar panels, also known as photovoltaic panels, consist of multiple PV cells that are connected to each other in both series and parallel. [21] PV cells are made of semiconductor material, often silicon, and are designed to absorb photons. [22] When a photon hits the surface of the cell, it transfers its energy to electrons, causing them to flow through the circuit as an electrical current. [23] PV cells produce Direct Current (DC), and need to be converted to Alternating Current (AC) before it can be fed into the grid or a household. This is done through an inverter. [22]

Figure 1 shows a schematic diagram of a PV system feeding into the grid.

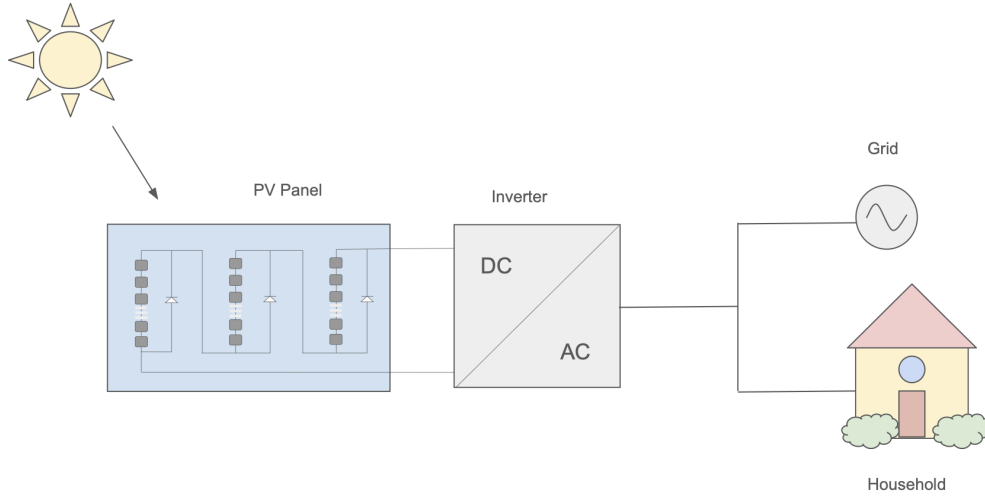


Figure 1: Schematic diagram of a PV system feeding into the grid.

3.1.2 Factors Affecting PV Output

The power output of a photovoltaic (PV) system can be approximated using the following formula:

$$P_{pv} = (N_p N_s) P_{pv,STC} f_{pv} \frac{G}{G_{STC}} [1 + \alpha_p (T_c - T_{c,STC})] \quad (1)$$

where

- P_{pv} = solar PV output
- N_s = number of modules in series

- N_p = number of modules in parallel
- $P_{pv,STC}$ = solar PV output under Standard Test Conditions
- f_{pv} = derating factor (to account for dust accumulation, shading, ageing, and wiring losses)
- G = solar irradiance
- G_{STC} = solar irradiance under Standard Test Conditions
- α_p = temperature coefficient of power
- T_c = module temperature
- $T_{c,STC}$ = module temperature under Standard Test Conditions [24]

The number of cells in series (N_s) and in parallel (N_p) is constant for a given PV system. P_{pv} is also constant, as it is given by the manufacturer, and G and T_{STC} are standard values defined as 1000 W/m^2 and 25°C . [25].

The irradiance G is typically a driving factor of short-term variability, as it depends on cloud coverage, time of day, season, etc. The module temperature T_c varies with ambient temperature, wind and irradiance, and a higher temperature causes a decreased efficiency, which is reflected by the temperature coefficient α_p . [26] Lastly, the derating factor f_{pv} accounts for various system losses and uncertainties, such as soiling, shading and ageing. [24]

The parameters G , T_c , α_p and f_{pv} illustrate the complexity of accurately estimating P_{pv} , since they are influenced by several interrelated factors, many of which are weather-related.

It is also worth mentioning that climate change directly affects the efficiency of photovoltaics. For example, rising air temperatures reduces the efficiency of the PV modules, and changed weather patterns can increase the uncertainty of solar energy availability. [9]

3.1.3 Grid Integration of PV Systems

While PV systems contribute with many advantages when coupled with the grid, they also come with notable challenges. Firstly, the rapid introduction of an increasing number of PV systems in the grid can lead to technical difficulties such as grid stability problems and voltage fluctuations. Moreover, since solar power relies on daily cycles and weather, the supply and demand can be hard to handle, highlighting the need for innovative energy storage systems. [6]

3.1.4 Battery Storage in PV Systems

Battery Energy Storage Systems (BESS) can be combined with PV systems to overcome the intermittency of PV power production. [27] As a general rule of thumb, PV production is highest during daytime and negligible at night. However, electricity demand often peaks in the morning and evening, which may lead to a temporal mismatch between generation and consumption. BESS can help mitigate this mismatch by storing excess PV generation and supplying electricity during periods of higher demand. [28]

BESS can also be used to sell energy to the grid, which may support grid stability by supplying electricity during periods of high demand. [27] Accurate PV forecasting is therefore highly relevant for efficient battery operation and energy management, as it can ensure optimal charge and discharge scheduling.

3.2 Energy Markets

The electricity market exists to balance supply and demand in real time, as production and consumption on the physical grid must remain in constant equilibrium. There are several markets that work together to obtain equilibrium, all with different time horizons:

- **Forward market:** up to 10 years before delivery
- **Day-ahead market:** 12-36 hours before delivery
- **Intraday market:** up to 1 hour before delivery
- **Balancing market:** during and immediately prior to delivery [29]

This study will focus on the day-ahead market and the intraday market. It should be noted that the market context presented in this section is based on the Swedish and European electricity markets, with Nord Pool and Svenska kraftnät as primary references. Since the study later compares datasets from non-European countries, this may seem inconsistent. However, the sole purpose of introducing energy markets here is to motivate the selected time horizons - not to characterise the electricity markets of the countries from which the datasets originate. The datasets were selected to represent a variety of climatic conditions, and the underlying market structures of those countries are considered outside the scope of this study.

3.2.1 Day-Ahead Market

The largest volume of electricity is traded in the Day-Ahead (DA) market, where the spot-price for the following day is decided through an auction. [11] Nord Pool publishes the available capacities on interconnectors and in the grid at 10:00 CET, and actors have until 12:00 CET to submit their final bids for delivery times the next day. [30] On September 30th 2025, the Market Time Unit (MTU) was changed to 15 minutes instead of 60 minutes in the European DA market [31]. This means that the final bids entail how much actors want to buy/sell each 15 minutes between 00:00 and 23:45 the following day. [11]

Once the orders are submitted, they are matched with other orders in an algorithm called Euphemia. Here, the price for each timestamp and bidding zone is set, where the curves for sell price and buy price meet and network constraints are taken into account. Prices are announced to the market at 12:45 CET or later, where individual results are reported to each seller and buyer. These actors then have an obligation to deliver and consume the sold or purchased electricity. [30]

3.2.2 Intraday Market

The intraday market serves as a complement to the DA market, giving actors the opportunity to adjust their positions closer to delivery - for instance, in response to unexpected changes in weather-dependent production. The intraday market consists of three auctions (IDA1, IDA2 and IDA3) and a continuous market. The three auctions take place at 15:00 CET the previous day, 22:00 CET the previous day and 10:00 CET the same day, and they work similarly to the DA-market. [12]

The continuous market starts at 15:20 CET the previous day and closes one hour before delivery. [12] Like the DA market, the MTU of the Nordic intraday market was changed from 60 minutes to 15 minutes on March 18th 2025 [31]. The prices are set individually for each transaction, in a pay-as-bid principle, where the highest buy price and lowest sell price come first. [32] This thesis will only consider the continuous market, and will not discuss the three auctions further.

3.3 Köppen-Geiger Climate Classification

Climate is defined by World Meteorological Organization (WMO) as "the average weather conditions for a particular location over a long period of time, ranging from months to thousands or millions of years". WMO uses a 30-year period for averaging climate variables [33], which is widely considered as the classical time frame for climate discussions. [34]

The Köppen-Geiger climate classification system is used to categorise regions based on general temperature profile. The major categories are:

- **A:** Tropical climates
- **B:** Dry climates
- **C:** Moist Subtropical Mid-Latitude Climates
- **D:** Moist Continental Mid-Latitude Climates
- **E:** Polar Climates
- **H:** Highlands [35]

Additionally, there are subdivisions that further divide each category into smaller groups based on perception and temperature patterns. These subdivisions are represented by a second and a third letter. [36]

Figure 2 shows a world map of the Köppen-Geiger climate classification.

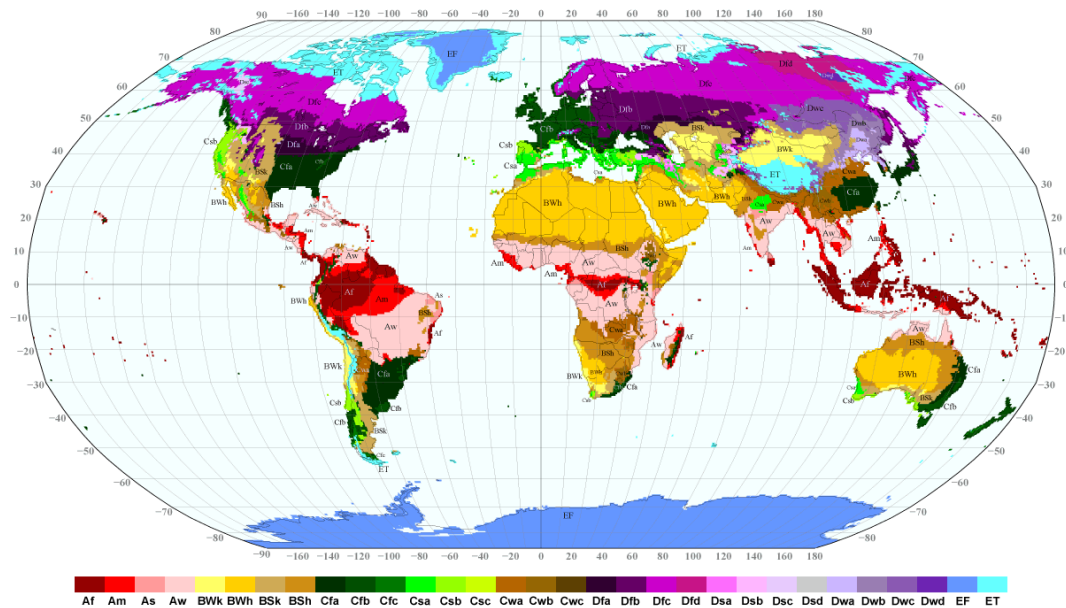


Figure 2: Köppen-Geiger climate classification. [37]

3.4 Forecasting Models

3.4.1 Persistence Models

A Persistence model is an extremely simple model, where it outputs the value of a previous time as the predicted value. [3] Persistence models are a great way to evaluate more complex machine learning models; if advanced models perform worse than, or not significantly better than, a Persistence model, this suggests that they offer limited practical benefit.

A *Naïve Persistence* model outputs the current timestamp as the previous timestamp's actual value. [7] In other words:

$$\hat{y}_{t+1} = y_t \quad (2)$$

The Naïve Persistence model is well-suited for stable weather but struggles with short-term variability. [7]

A *Seasonal Persistence* model uses the timestamp hour from the previous day as the forecast:

$$\hat{y}_{t+24h} = y_t \quad (3)$$

Seasonal Persistence captures daily solar cycles but has difficulties with day-to-day weather variation. [7]

3.4.2 Random Forest

Among ML models, Random Forest (RF) stands out for its effectiveness and ability to capture non-linear relationships. RF is an ensemble learning technique which combines multiple decision trees to produce a forecast. [9] A decision tree is a model which makes predictions by recursively splitting the data based on input features. A decision tree uses a greedy algorithm, meaning it doesn't take the complete tree into account when splitting, but rather tries to obtain a model which explains the data as well as possible for each individual split. [38] Therefore, decision trees are known to have a high variance, which means that small changes in the training data can result in significantly different tree structures. [39]

To reduce variance of decision trees, bagging ('*bootstrap aggregating*') is introduced. The main idea of bagging is to reduce the variance of the base model, without increasing its bias. This is done by artificially creating multiple datasets from a single dataset of the same size. [38] An example of bagging could be:

Original dataset:

$$[1, 2, 3, 4, 5]$$

Bootstrap datasets:

$$[1, 2, 2, 5, 5]$$

$$[1, 1, 4, 5, 5]$$

$$[2, 3, 3, 4, 4]$$

Each bootstrap dataset is used to train an ensemble of decision trees. Their predictions are then averaged, which subsequently reduces the variance and keeps the bias low. [38]

The problem with bagging is that trees can end up having a very high correlation with each other. This is not ideal; if the trees are too similar, they might learn the same mistakes. To prevent this, additional randomness is

introduced in a method referred to as *Random Forests*. In a Random Forest, the decision trees are trained using bagging, but with one significant difference. Each time a node is about to be split during training, all possible features $x_1, \dots, x_p$ are not considered as splitting variables. Instead, a random subset is chosen consisting of $q \leq p$ inputs, where these are the only variables considered. [38]

To illustrate this, it is easy to imagine an ensemble of decision trees where there is one dominant feature. If the trees would be constructed using plain bagging, it is likely that all trees would pick this dominant feature as the first splitting variable, making them identical after the first split. However, using RF instead, some of the ensemble members would not have access to this dominant variable at the first split, ultimately forcing them to split according to another feature. Even though this does not necessarily improve the performance of an individual tree, it may become beneficial later in the splitting process, and when averaging across many trees, the overall performance can improve. [38]

An example of a single decision tree, trained on one of the datasets used in this thesis, is provided in Appendix A. From this, the intuition behind using decision trees for PV forecasting becomes apparent: solar output is largely governed by a hierarchy of meteorological conditions, which maps naturally onto a sequential series of if-then rules. For instance, the tree may first split on whether solar irradiance exceeds a certain threshold, followed by further splits on humidity or time of day.

3.4.3 eXtreme Gradient Boosting

Like RF, eXtreme Gradient Boosting (XGBoost) is a tree-based optimisation algorithm. [8] However, while RF uses bagging, XGBoost uses (as the name suggests) *boosting*. Boosting is also an ensemble method, and shares some similarities with bagging. The main difference is that boosting uses a sequential construction, meaning that each model tries to learn from the mistakes made by the previous one. *Gradient* boosting means that each new model is trained to minimise the loss function by following the negative gradient of the error with respect to the current predictions. [38] The objective function of which the XGBoost algorithm tries to minimise can be written as:

$$L^{(t)} = \sum_{i=0}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \quad (4)$$

where l is the loss function that represents the error between the predicted data $\hat{y}_i$ and the real data y_i , f_t is the model of the t -th tree, and $\Omega(f_t)$ is the regularisation term. The regularisation term can be further written as:

$$\Omega(f) = \gamma T + \frac{1}{2} \lambda \|\omega\|^2 \quad (5)$$

where T represents the number of tree leaves, ω contains the scores of each leaf, and γ and λ are regularisation hyperparameters. [8] The regularisation term is used to avoid overfitting [38], and is a clever property of XGBoost compared to regular gradient boosting. Furthermore, XGBoost allows for parallel distributed computing of data and has a cache-aware prefetching algorithm. This reduces runtime for large datasets, making XGBoost a popular choice amongst ML models. [40]

3.4.4 Ridge Regression

Ridge regression is a linear regression model with squared error loss, where an L_2 regularisation is added:

$$\hat{\theta} = \operatorname{argmin}_{\theta} \frac{1}{n} \|X\theta - y\|_2^2 + \lambda \|\theta\|_2^2 \quad (6)$$

where the second term is the regularisation term, and λ decides how heavily weights should be penalised. Like already discussed, the regularisation term helps reducing overfitting, improving the model stability. [38]

Ridge regression should not be confused with *kernel ridge regression*. In kernel ridge regression, a kernel function is used to implicitly map the input data into a higher-dimensional feature space, allowing the model to capture non-linear relationships. It is however true that a kernel ridge regression is very similar to ridge regression when using a linear kernel. [38]

3.4.5 Support Vector Regression

Support Vector Regression (SVR) is a non-linear method [7] very similar to kernel ridge regression. [38] The difference is that SVR uses another loss function; in SVR, some of the elements in the dual parameter vector $\hat{\alpha}$ become exactly zero, making the prediction depend only on the data points with $\hat{\alpha}_i \neq 0$:

$$L(y, \hat{y}) = \begin{cases} 0, & \text{if } |y - \hat{y}| < \epsilon \\ |y - \hat{y}| - \epsilon, & \text{otherwise} \end{cases} \quad (7)$$

These non-zero elements of $\hat{\alpha}$ are called *support vectors*. The variable ϵ is a design choice, and Equation 7 describes the so-called ϵ -insensitive tube. The ϵ -insensitive tube means that a small deviation ϵ is allowed between the real and predicted data. It is only the support vectors - the data points that fall outside of the ϵ -insensitive tube - that contribute to making predictions, which can significantly reduce the computational burden. [38]

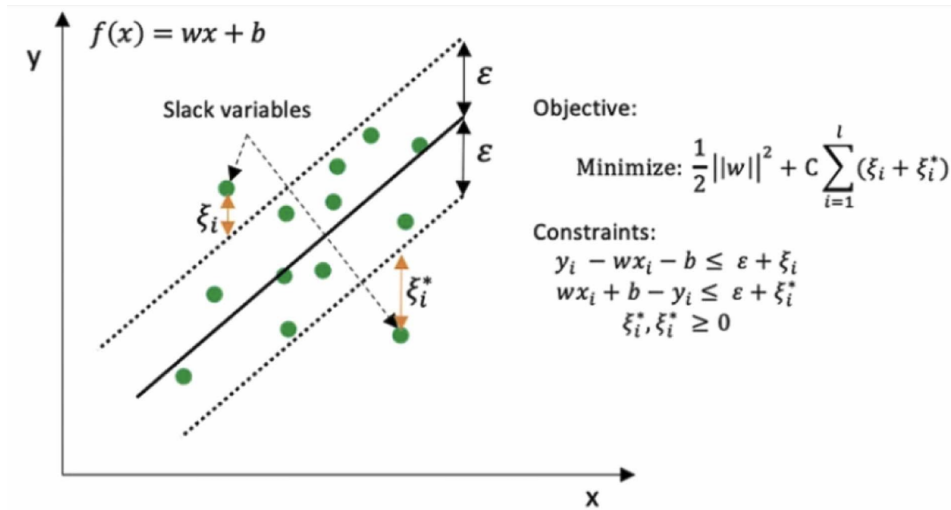


Figure 3: Linear SVR. [7]

Figure 3 demonstrates a linear SVR, where an ϵ -insensitive tube is drawn with dotted lines. Slack variables quantify how far data points lie outside the ϵ -insensitive tube, with ξ_i corresponding to points above the tube and ξ_i^* to points below it. As seen, the objective when finding a function $f(x) = wx + b$ is to minimise:

$$\frac{1}{2} \|w\|^2 + C \sum_{i=1}^l (\xi_i + \xi_i^*) \quad (8)$$

where $\|w\|^2$ penalises large weights w (thereby encouraging low-complexity functions), while the term $C \sum_{i=1}^l (\xi_i + \xi_i^*)$ penalises deviations of training points outside of the ϵ -insensitive tube. [7]

To summarise, SVR is similar to kernel ridge regression except that it introduces an ϵ -insensitive tube in its loss function. By doing so, the model ignores small deviations and only penalises errors that fall outside this region.

As a result, only the support vectors contribute to the final model, which is computationally efficient as only the most informative data points are being looked at in the final prediction.

3.4.6 Long Short-Term Memory

Long Short-Term Memory (LSTM) is a specialised type of Recurrent Neural Network (RNN) - which in turn is a type of Artificial Neural Network (ANN) - with great advantages in time series forecasting because of its efficiency in solving long-term and short-term dependencies. It is therefore widely used in the fields of PV and irradiance forecasting. [3]

Taking a few steps back, the ANN is by some considered the most popular AI approach [24], where the hierarchical form of learning is the foundational technology behind deep learning. [7] ANNs mimic how the human brain processes information, with artificial neurons used for analysing data and finding complex patterns. [41]

The key components of an ANN are an *input layer*, where the network receives information, *hidden layers*, where information is transformed and processed, and an *output layer*, where the final prediction is made. The more hidden layers of an ANN, the more complex patterns it can learn. [41]

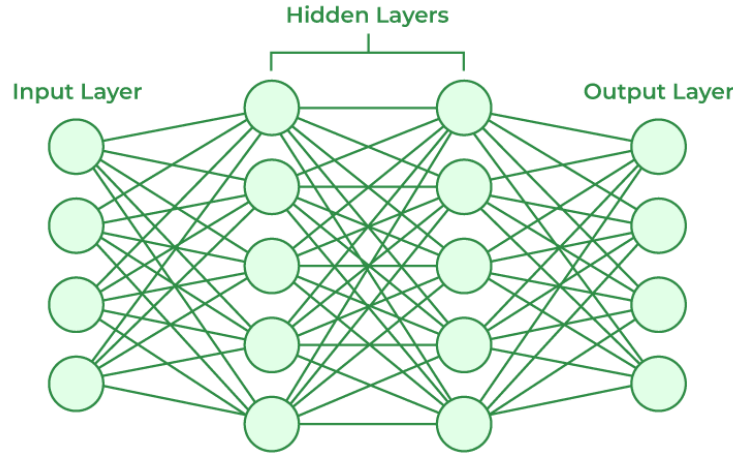


Figure 4: Architecture of Artificial Neural Networks. [41]

Recurrent neural networks have the ability to store memory from previous time steps, making them useful for temporal data. The main difference in architecture from ANNs is the presence of a recurrent cell, which can 'remember' information by incorporating the hidden state from previous time steps into the current computation, and thereby enabling the network to capture dependencies across time. [6]

A problem with RNNs is that the weights can increase or shrink exponentially over the iterations, resulting in exploding or vanishing gradients. [6] To solve this, LSTM was introduced, with gating mechanisms that regulate the flow of information, and thereby preventing uncontrolled growth or decay of gradients. The LSTM cell has three gate controls: the *input gate* (i_t), the *output gate* (o_t), and the *forget gate* (f_t). [3]

The input gate i_t determines how much of the new input should be incorporated into the cell state. It combines the current input x_t , previous hidden state m_{t-1} , and previous cell state c_{t-1} , and applies a sigmoid function σ to produce a gating value between 0 and 1, thereby controlling the amount of new information stored in the memory [3]:

$$i_t = \sigma(W_{ix}x_t + W_{im}m_{t-1} + W_{ic}c_{t-1} + b_i) \quad (9)$$

where W are the weight matrices and b is the bias vector.

The forget gate f_t filters out unnecessary details from the cell state, ensuring that the neural network only focuses on relevant old information during training [42]. The forget gate has the following formula [3]:

$$f_t = \sigma(W_{fx}x_t + W_{fm}m_{t-1} + W_{fc}c_{t-1} + b_f) \quad (10)$$

The output gate o_t plays a role in deciding how much of the internal memory should be exposed as output, and in doing so, controlling the extent to which the internal memory contributes to the output at each time step. [42] The output gate can be described by [3]:

$$o_t = \sigma(W_{ox}x_t + W_{om}m_{t-1} + W_{oc}c_{t-1} + b_o) \quad (11)$$

The cell state c_t is affected by both the input gate i_t and the forget gate f_t . The forget gate determines the extent to which the previous memory is preserved and the input gate adds relevant new information [3]:

$$c_t = f_t \odot c_{t-1} + i_t \odot g(W_{cx}x_t + W_{cm}m_{t-1} + b_c) \quad (12)$$

where $\odot$ denotes the Hadamard product (element-wise multiplication).

Furthermore, the hidden state m_t can be described by:

$$m_t = o_t \odot h(c_t) \quad (13)$$

where h usually is a $\tanh$ -function. [3]

Finally, the output y_t is obtained by applying a linear transformation to the hidden state m_t , followed by an activation function φ . This maps the internal representation of the neural network to the final prediction at each time step [3]:

$$y_t = \varphi(W_{ym}m_t + b_y) \quad (14)$$

Figure 5 shows the architecture of the LSTM model, where Equations 9-14 are illustrated visually.

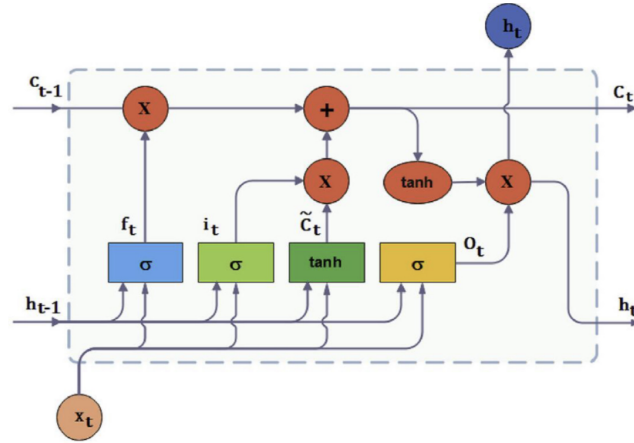


Figure 5: Architecture of LSTM. [3]

3.4.7 Bi-Directional Long Short-Term Memory

The Bi-Directional Long Short-Term Memory (Bi-LSTM) model is an extension of the LSTM model, where its primary feature is its capability of processing sequences in both forward direction and backward direction. This enables the model to capture both past and future temporal dependencies, which is beneficial in sequence-based tasks such as time series forecasting, where the full temporal context of a sequence can enhance prediction performance. [42] Figure 6 shows the architecture of the Bi-LSTM model.

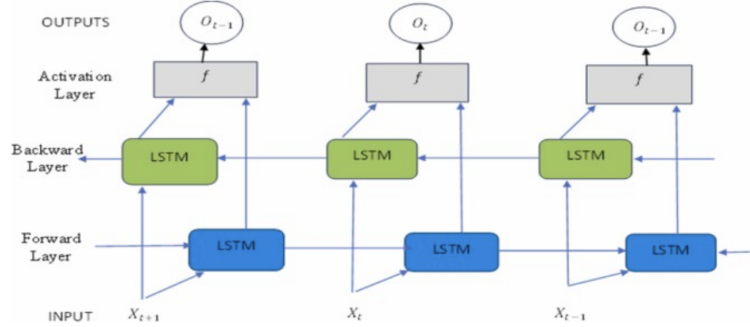


Figure 6: Architecture of Bi-LSTM. [42]

3.5 Hyperparameter Optimisation

Hyperparameter optimisation is very important in machine learning, as some hyperparameters may have a great impact on the model's performance. [43] Finding the optimal hyperparameters for a certain model and dataset may be an intricate process, which is why various hyperparameter optimisation algorithms often are used. [7]

3.5.1 Optuna

Optuna is an open source hyperparameter optimisation framework designed to automate hyperparameter searches through its state-of-the-art algorithms. [44] Unlike `Scikit-learn`'s commonly used `GridSearchCV`, which performs an exhaustive search over a pre-decided set of hyperparameters [7], Optuna uses Bayesian optimisation and adapts its search strategy based on the performance of previous trials, thereby saving computation time. [45]

Optuna's two most basic concepts are called *study* and *trial*. The aim of the study is to find the optimal hyperparameters based on an objective function, and it does so through multiple trials. [44] Optuna manages the exploitation-exploration trade-off by both considering known good hyperparameters and trying new ones. A great asset of Optuna is that it also implements pruning, meaning that it has the ability to stop unpromising trials early. [45]

It is the *sampler* that decides what hyperparameters to try next, and different samplers employ different algorithms. [45] In this paper, the `TPESampler` is used, which uses a Tree-structured Parzen Estimator algorithm. [44]

3.5.2 Tree-structured Parzen Estimator Algorithm

Unlike Gaussian processes, that model the Expected Improvement (EI) as $p(y|x)$, the Tree-structured Parzen Estimator (TPE) models EI as $p(x|y)$ and $p(y)$ instead. Intuitively, this means that instead of trying to model the output given a certain set of hyperparameters, the TPE models what hyperparameters often result in a good output. The TPE defines $p(x|y)$ as:

$$p(x|y) = \begin{cases} l(x), & \text{if } y < y^* \\ g(x), & \text{if } y \geq y^* \end{cases} \quad (15)$$

where y^* is a threshold value, $l(x)$ is the distribution of good hyperparameters, and $g(x)$ is the distribution of bad hyperparameters. The TPE selects y^* as a quantile γ of observed values y , so that $p(y < y^*) = \gamma$. A small γ will therefore focus more on exploitation, while a bigger γ will focus more on exploration. It can be shown that:

$$EI_{y^*}(x) \propto \frac{l(x)}{g(x)} \quad (16)$$

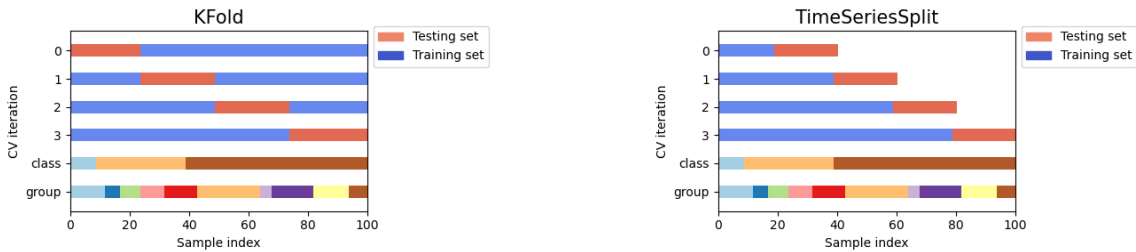
In other words, to maximise improvement of the output, the hyperparameters x should score a high value under $l(x)$, and a low value under $g(x)$. This means that the hyperparameters should be similar to known good hyperparameters, while also differing from known bad hyperparameters. [46]

3.5.3 Time Series Cross-Validation

When tuning hyperparameters, the performance needs to be evaluated according to some metric in order for the optimisation algorithm to decide if a trial's hyperparameters are good or bad. Instead of evaluating the performance on the training data - which often results in overly optimistic results, leading to overfitting - cross-validation can be used. [7]

The most basic idea of cross-validation is called k -fold validation. In this approach, the training data is divided into k subsets called "folds". For each of the folds, a model is trained using $k - 1$ of the folds as training data, and evaluated on the remaining fold. This procedure is repeated over all k folds, and the average performance is then recorded as the performance measure. Figure 7a shows the idea of k -fold cross validation. [47]

A time series cross-validation maintains the chronological order of the data (`shuffle=False`), enabling the model to learn the structure of a real time series. [9] Figure 7b shows the idea of time series validation - as seen, the testing data always corresponds to the latest timestamps, which stops the model from "learning from the future" - thereby reducing overfitting. [47]



(a) Standard k -fold cross-validation.

(b) Time series split cross-validation.

Figure 7: k -fold cross validation vs. time series split cross-validation. [47]

3.6 Scaling

Standardisation of features is common practice in machine learning, as it ensures that no single feature dominates the model simply because it was measured on a larger scale than the others.

This thesis uses `Scikit-learn`'s `StandardScaler()` to scale data. `StandardScaler()` standardises features by removing the mean and scaling to unit variance:

$$z = \frac{x - u}{s} \quad (17)$$

where z is the standard score, x is the value of the sample, u is the mean of the training samples and s is the standard deviation. [48]

3.7 Evaluation Metrics

The performance of trained models can be evaluated using different metrics. This paper focuses on four different evaluation metrics, namely Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Symmetric Mean Absolute Percentage Error (sMAPE) and Normalised Root Mean Square Error (nRMSE). The formulas for these metrics can be written as:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=0}^{n-1} (y_t - \hat{y}_t)^2} [7] \quad (18)$$

$$MAE = \frac{1}{n} \sum_{i=0}^{n-1} |y_t - \hat{y}_t| [49] \quad (19)$$

$$sMAPE = \frac{100\%}{n} \sum_{t=1}^n \frac{|y_t - \hat{y}_t|}{(|y_t| + |\hat{y}_t|)/2} [50] \quad (20)$$

$$nRMSE = \frac{RMSE}{y_{average}} [51] \quad (21)$$

RMSE is a risk metric corresponding to the square-root of the ℓ_2 -norm loss, where outliers are given more weight. RMSE has the advantage of being expressed in the same unit as the target variable, which facilitates interpretation. MAE measures the ℓ_1 -norm loss, which means that it gives equal weight to all errors and does not penalise outliers as heavily as RMSE, since errors are not squared. [7] sMAPE is scale-independent and is stable with zero or near-zero denominators [50], which is why it was chosen for this paper as PV data contains a lot of values equal to zero due to nighttime. nRMSE is the RMSE normalised to the average of the observed values. [51] Since this paper covers several datasets, the nRMSE is useful as it is scale-independent, making comparison between datasets easier.

4 Data Collection

Data was collected through open-source data. Since RQ1 entails evaluating the performance of forecasting models across PV sites from *different climatic regions*, the Köppen-Geiger Climate Classification was used as a framework for selecting datasets representative of distinct climate zones. In total, four different datasets were collected, as presented in Table 2.

Table 2: Overview of datasets used in the study.

Location	Time Span	Time Resolution	KG Zone	KG Zone Description
Alice Springs, NT, Australia [52]	2017–2025	5 min	BWh	Subtropical desert
Kersey, CO, USA [53]	2017–2023	5 min	BSk	Mid-latitude steppe
Lisbon, Portugal [54]	2019–2022	1 h	Csa	Mediterranean
Calgary, AB, Canada [55]	2015–2023	1 h	Dfb	Humid continental

A more detailed description of each climate zone is as follows:

- **BWh - Subtropical desert:** average temperature is *more* than 18°C [36], and the world’s heat records are generally found in these hot deserts. [56] While evaporation exceeds precipitation, the actual evaporation remains *below half* of the potential evaporation. No (or infrequent) frost. [36]
- **BSk - Mid-latitude steppe:** average temperature is *less* than 18°C. While evaporation exceeds precipitation, the actual evaporation remains *below* the potential evaporation. [36] Winters are cold and possibly freezing, with snowfall not being uncommon. [57]
- **Csa - Mediterranean:** average temperature during the *warmest month* is *more* than 22°C. At least four months have average temperatures exceeding 10°C. The precipitation during the wettest winter months is at least three times as much than during the driest summer month. Frost may occur during winter. [36]
- **Dfb - Humid continental:** average temperature during the *warmest month* is *less* than 22°C. The average temperature of the *coldest month* is generally well below -3°C. Frost-free periods usually last 3-5 months [58], and winters are considered severe. [36]

While the Australian and Portuguese dataset already contained local weather-data, the American and Canadian did not. Hence, additional data was downloaded from *Photovoltaic Geographical Information System* (PVGIS), which is a free and open access tool on photovoltaic system performance developed by *Joint Research Centre* (JRC), the EU Commission’s research body. PVGIS can be accessed here: [59]. Data was downloaded from “Hourly data”.

When using PVGIS, a few details about the PV panels need to be filled out, namely: the coordinates, mounting type, slope, azimuth, PV technology, installed peak power [kWp] and system loss [%]. A solar radiation database also needs to be chosen. Table 3 shows the input parameters for the data collection of each site. Even though the Australian and Portuguese dataset already contained local weather-data, the JRC data would still serve as useful, as seen later on.

Some assumptions had to be made about the PV panels in order to fill out all the details, and these are marked with * in Table 3. The PV technology for the Australian dataset is marked with ** because the PV technology was actually CIGS, but since this was not an option on PVGIS, CIS was chosen as a close substitute. For the two unknown PV technologies (Portugal and Canada), Crystalline Silicon was assumed as this is the default in PVGIS.

The assumptions about the slope and azimuth were mainly based on satellite pictures from *Google Earth*. Using the given coordinates, the solar panels could be located and the azimuth determined by measuring the direction they were facing relative to South. The slope was approximated using *Google Earth*’s 3D-feature, and measuring the angle of the panels compared to the horizontal plane.

Table 3: Parameters for PVGIS for each dataset. Assumptions are marked with *.

Dataset	Coordinates	Mounting type	Slope	Azimuth	Technology	kWp	Loss [%]	Database
Australia	-23.454429 133.522834	Fixed	20°	180° *	CIS **	5.2	15% *	ERA5
USA	40.3864 -104.5512	Inclined axis *	0° *	-	CdTe	4738.0	15% *	ERA5
Portugal	38.833 -9.191	Fixed *	20° *	18° *	Chrystalline Silicon *	16.32	15% *	SARAH3
Canada	51.029705 -114.117287	Fixed *	-27° *	25° *	Chrystalline Silicon *	10.5 *	15% *	ERA5

As satellite pictures revealed that the PV panels from Portugal and Canada were mounted on small-scale rooftop installations, the mounting type was assumed to be fixed. The American satellite pictures, however, suggested that the dataset came from a utility-scale solar farm (as also implied by the high installed peak power, as shown in Table 3). Therefore, the mounting type was assumed to be *inclined axis*, as large-scale solar farms typically prioritize energy output over simplicity. This means that the modules are mounted on a structure rotating east-west throughout the day, ensuring that the angle to the sun always is as small as possible.

The system loss was overall assumed to be 15%. PVGIS uses 14% as a standard system loss, but this was rounded up to 15% as a more conservative assumption. The installed peak power in the Canadian dataset was assumed to be 10.5 kWp based on plots of the dataset.

5 Data Preprocessing

All data processing, modelling, and evaluation was carried out in Python. The analysis relied primarily on `pandas` for data manipulation, `scikit-learn` for ML-models, and `TensorFlow` for the LSTM-model.

Each dataset was preprocessed thoroughly before any feature selection could take place. Data preprocessing includes, but is not limited by, outlier handling, interpolation, taking care of missing timestamps and missing values, merging datasets with the corresponding JRC dataset, and feature engineering.

5.1 Australia Site



Figure 8: Satellite picture of Australia solar site. The blue arrow indicates the PV system analysed in this report.

Initially, the Australian dataset included 14 columns and 811308 rows. Three columns were dropped straight away due to their irrelevance for this study: `Performance_Ratio`, `Current_Phase_Average` and `Active_Energy_Delivered_Received`.

The remaining columns were the following:

- `timestamp` - timestamp in ACST (UTC+9:30) [5 minute resolution]
- `Active_Power` - instantaneous active power [kW]
- `Wind_Speed` - wind speed [m/s]
- `Weather_Temperature_Celsius` - temperature [°C]
- `Weather_Relative_Humidity` - relative humidity [%]
- `Global_Horizontal_Radiation` - global horizontal radiation [W/m^2]
- `Diffuse_Horizontal_Radiation` - diffuse horizontal radiation [W/m^2]
- `Wind_Direction` - wind direction [°]
- `Weather_Daily_Rainfall` - daily rainfall [mm]

- `Radiation_Global_Tilted` - total tilted irradiance [W/m^2]
- `Radiation_Diffuse_Tilted` - diffuse tilted irradiance [W/m^2]

The `timestamp` column was converted to datetime format, and was renamed to `time`. `Active_Power` was renamed to `kW` and the two columns were plotted against each other as an initial visual inspection, as seen in Figure 9.

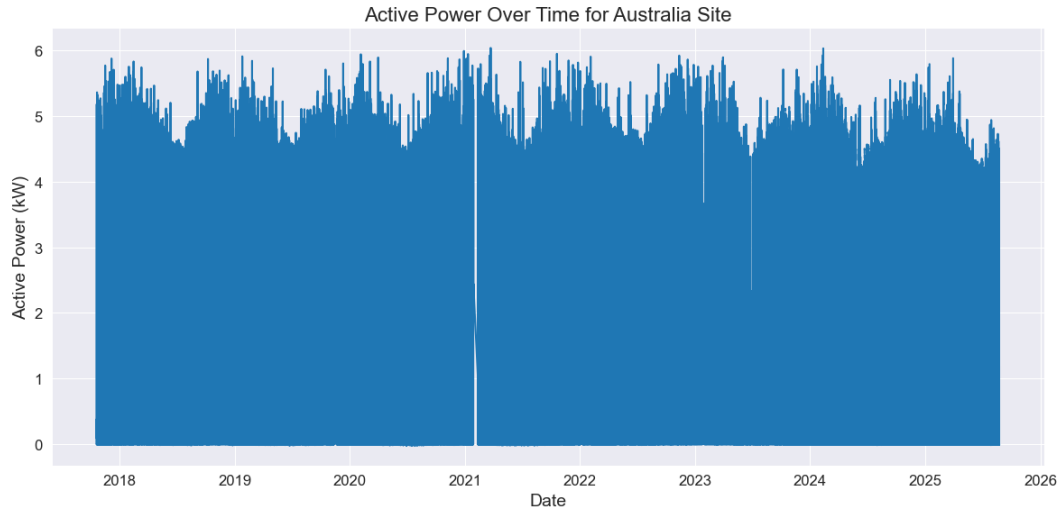


Figure 9: Active power over time for Australia site.

All negative values of `kW` were then set to zero, under the assumption that they were due to measurement noise.

5.1.1 Missing Timestamps and Values

Knowing that the data should have a 5 minute resolution, a simple algorithm was implemented to calculate how many timestamps were missing. This turned out to be 13 584. After some further investigation, missing timestamps turned out to occur significantly more frequently from 2023 onwards. Given the increased irregularity, all observations from 1 January 2023 were dropped. Although imputation techniques could have been applied to recover missing timestamps, the high frequency and systematic nature of the missingness from 2023 onwards suggested a potential underlying data quality issue rather than random omissions. Dropping the affected period was therefore considered more appropriate than risking the introduction of imputed values into a structurally unreliable portion of the dataset. The remaining dataset covered more than five years (October 2017 to December 2022), and was therefore considered sufficient for the analysis.

Before adding the rest of the missing timestamps, NaN values were addressed. The column `Wind_Speed` was dropped as it turned out that it *only* contained NaN-values, making the column useless. All other columns, except for `Radiation_Global_Tilted` and `Radiation_Diffuse_Tilted`, were interpolated as the NaN values corresponded to a negligible proportion of the data. The columns `Radiation_Global_Tilted` and `Radiation_Diffuse_Tilted` were left unaltered; the NaN values corresponded to 12% of the data, which was a too large portion to interpolate reliably, yet too small to justify discarding the data entirely.

After the missing values were added, the missing timestamps were added as well, with NaN entries in all other columns.

5.1.2 Exploration of Local Weather Variables

As the Australian dataset already contained many interesting weather variables, these were plotted and investigated.

Figure 10a shows `Weather_Temperature_Celsius` plotted over time. As seen, two different types of outliers seemed to be present in the dataset; both extreme spikes (such as -40°C or $+60^{\circ}\text{C}$), and sudden drops to exactly 0°C .

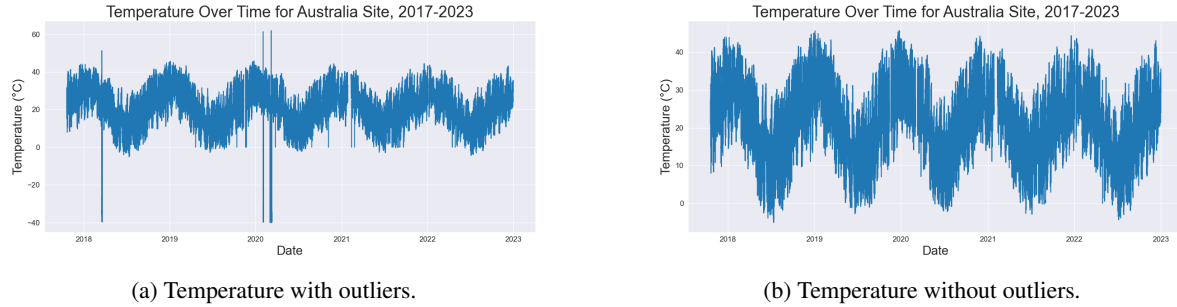


Figure 10: Temperature data before and after outlier handling.

Starting with the extreme spikes, these seemed to cover longer periods (hours or days) and were probably due to measurement errors, see Figure 11a. These periods were removed entirely, where the temperature values were replaced with NaNs. The sudden drops to 0°C , on the other hand, were isolated single-point anomalies (see Figure 11b) and were therefore treated via interpolation. Figure 10b shows the temperature over time after the adjustments.

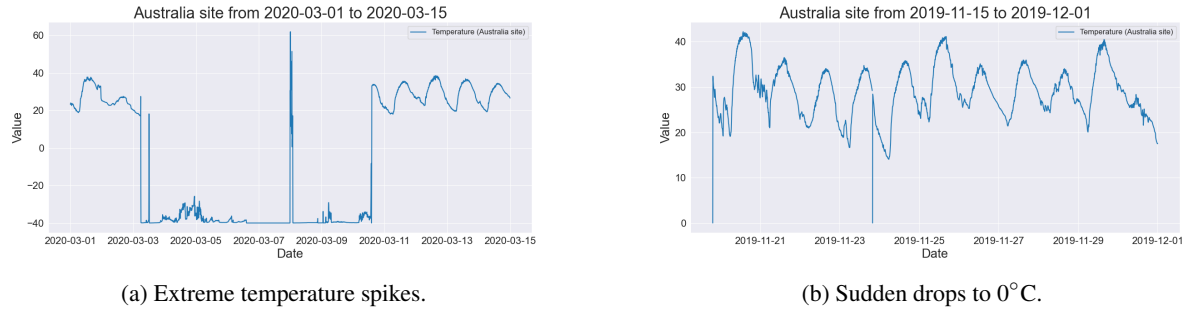
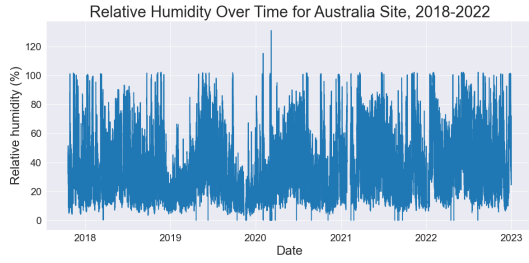
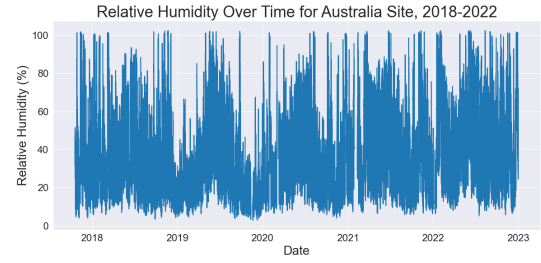


Figure 11: Examples of outliers in temperature data.

`Weather_Relative_Humidity` was treated in a similar way; longer periods of obvious sensor faults were removed, and isolated anomalies were interpolated. As seen in Figure 12a, this included sudden drops to 0%, and a couple of spikes well above 100%. Even though 100% is the theoretical maximum value of relative humidity, some observations slightly exceeded this threshold (up to approximately 102%) after the outlier handling, see Figure 12b. These values were retained, as the deviations were small and are likely attributable to minor sensor calibration offsets or measurement uncertainty rather than true physical inconsistencies.



(a) Original data.



(b) Outliers removed.

Figure 12: Relative humidity data before and after outlier handling.

The rest of the weather variables were plotted but no changes were made, except for the removal of one additional outlier in `Global_Horizontal_Radiation` through interpolation.

5.1.3 Importing Data from JRC

The DataFrame from JRC initially contained the following columns:

- `time` - timestamp in UTC [1 hour resolution]
- `P` - estimated power output [W]
- `Gb(i)` - beam (direct) irradiance on the inclined plane [W/m^2]
- `Gd(i)` - diffuse irradiance on the inclined plane [W/m^2]
- `Gr(i)` - reflected irradiance on the inclined plane [W/m^2]
- `Hsun` - sun height [$^\circ$]
- `T2m` - 2 meter air temperature [$^\circ\text{C}$]
- `WS10m` - 10 meter total wind speed [m/s]
- `Int` - 1 means solar radiation values are reconstructed (0 otherwise)

To begin with, `time` was converted to datetime format and the time zone was changed to ACST. Then, the DataFrame was resampled to a 5 minute resolution using linear interpolation. Furthermore, `P` was converted to kW instead of W, and a new feature `G` was created, where $G = G_b(i) + G_d(i) + G_r(i)$. This feature represents the total irradiance on the tilted PV surface. `Int` was also dropped, as it only contained the value 0, making the feature useless.

5.1.4 Merging Data

The data frame from the Australia site was merged with the JRC data frame, using `time` as index. Even though the Australian dataset already included many weather features, the JRC dataset could be served as a complement - for example, since the column `Wind_Speed` had to be dropped, `WS10m` could be used instead.

As observed, the JRC dataset included a feature denoted `P`. This represents an estimated power output based on a photovoltaic system with similar characteristics to the real system, as modelled by JRC. Thus, it does not correspond to directly measured data. Still, a comparison between this estimated feature and the real data was conducted to evaluate its accuracy.

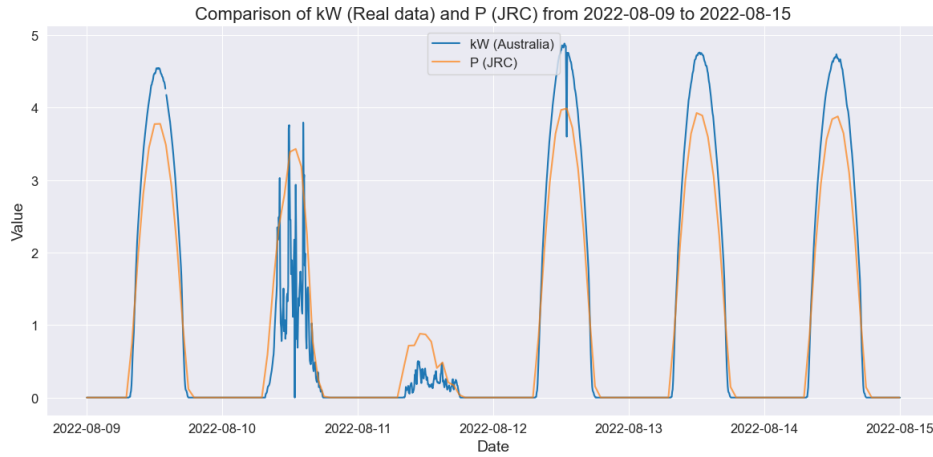
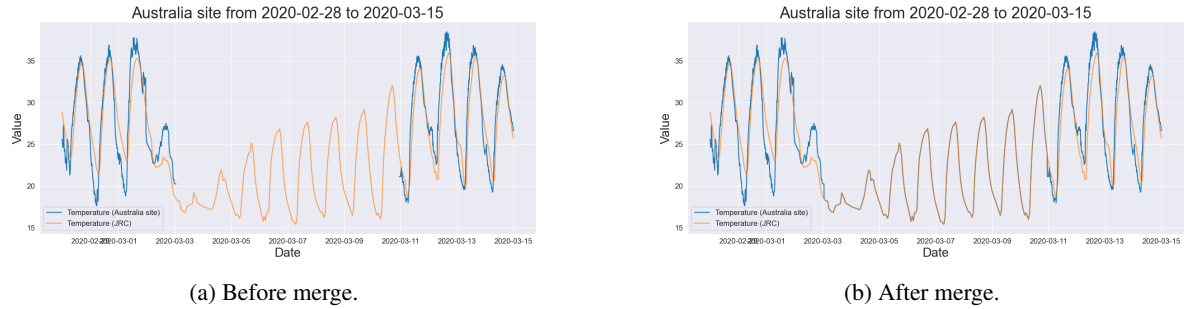


Figure 13: Real measured power vs estimated power through JRC.

As seen in Figure 13, there seems to be a scaling disparity between the estimated and measured data on clear days. In an ideal case, these should align perfectly. However, JRC's P still captures the general pattern of the measured data, even though it is evident that the datasets originally had different time resolutions.

Recall that several time periods had their temperature data removed when handling outliers. To address these gaps, NaN values in `Weather_Temperature_Celsius` were replaced with JRC's estimated temperature data T_{2m} . See Figure 14.



(a) Before merge.

(b) After merge.

Figure 14: Temperature data (measured and JRC) before and after merge.

5.1.5 Feature Engineering and Resampling

Using the Python package `pvl`, additional features could be extracted for the exact coordinates of the location, namely:

- `solar_elevation` - the sun's angle above the horizon
- `solar_azimuth` - the sun's direction relative to North
- `ghi_clear` - Global Horizontal Irradiance under clear-sky conditions
- `dni_clear` - Direct Normal Irradiance under clear-sky conditions
- `dhi_clear` - Diffuse Horizontal Irradiance under clear-sky conditions

Using `Global_Horizontal_Radiation` and `ghi_clear`, the feature `CSI` could be created:

$$CSI = \frac{GHI}{GHI_{clear}} \quad (22)$$

A high CSI would indicate a clear day, while a low value would indicate overcast conditions, as `ghi_clear` represents the theoretical maximum GHI under clear-sky conditions.

Lastly, a few temporal features were created as well: `year`, `month`, `hour`, `dayofyear`, `minute_of_day`, `tod_sin`, `tod_cos`, `doy_sin` and `doy_cos`. The last four were created using the following formulas:

$$tod_sin = \sin\left(\frac{2\pi \cdot minute_of_day}{1440}\right) \quad (23)$$

$$tod_cos = \cos\left(\frac{2\pi \cdot minute_of_day}{1440}\right) \quad (24)$$

$$doy_sin = \sin\left(\frac{2\pi \cdot dayofyear}{365}\right) \quad (25)$$

$$doy_cos = \cos\left(\frac{2\pi \cdot dayofyear}{365}\right) \quad (26)$$

The reason why these features were introduced is because temporal patterns are cyclical. For example, the timestamp 23:59 is very similar to the timestamp 00:00 the following day. But by only using linear temporal features, such as `minute_of_day`, the features suggest that they are very far from each other: 23:59 corresponds to minute 1440, and 00:00 corresponds to minute 0. By introducing cyclical features using `sin()` and `cos()`, the features are able to better capture the similarities between these two points. The exact same logic can be applied to Dec 31st versus Jan 1st.

To reduce the number of data points, the DataFrame was lastly resampled to a 15 minute resolution. The reason why 15 minutes was chosen was because this is the resolution of the MTU, as previously discussed.

5.2 USA Site

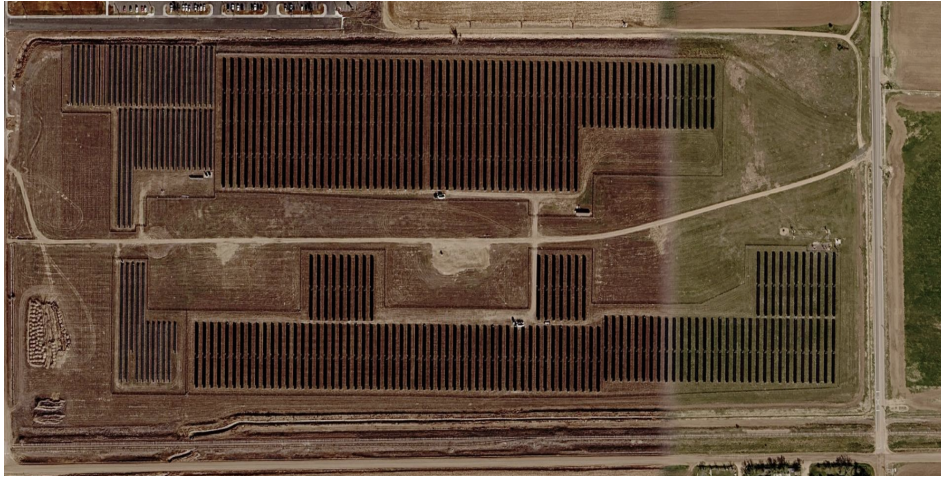


Figure 15: Satellite picture of USA solar site.

Initially, the American dataset contained 17 columns and 655434 rows. Despite the vast amount of columns, all of them were deemed to be irrelevant except for two, and were therefore dropped.

The remaining columns were:

- `measured_on` - timestamp in MT (UTC -7/UTC-6) [5 minute resolution]
- `meter_ac_power_(kw)_meter_150150` - instantaneous active power [kW]

The column `measured_on` was converted to datetime format and was renamed to `time`, and the column `meter_ac_power_(kw)_meter_150150` was renamed to `kW`. The two columns were then plotted against each other for an initial inspection, as seen in Figure 16.

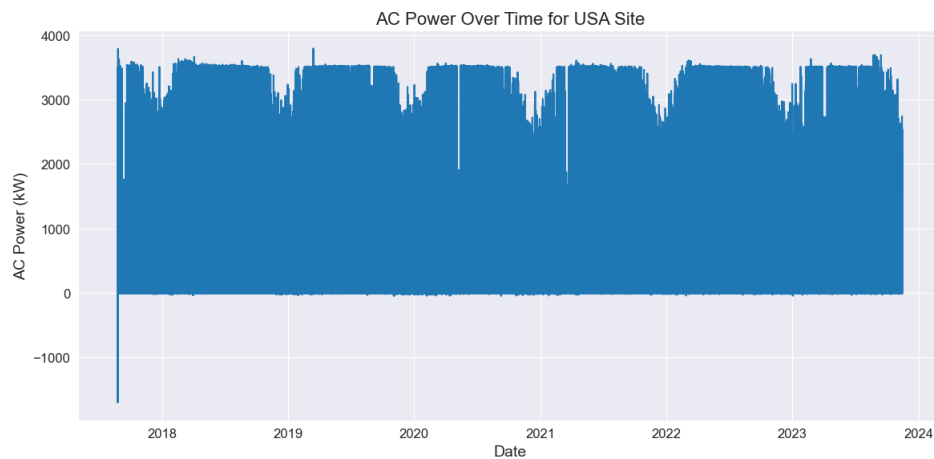


Figure 16: Active power over time for USA site.

As seen, there was an obvious outlier in the very beginning of the time series. This was removed by only including data from November 1st 2017 onwards.

5.2.1 Daylight Saving Time

As Colorado uses daylight saving, these timestamps were looked at more closely. Colorado switches *to* daylight saving the second Sunday in March each year [60], meaning that the 02:00-03:00 hour does not exist. Surprisingly, this non-existent hour was however not missing from the dataset. Upon further inspection, it was found that all surrounding timestamps had slightly negative values for kW , while the twelve timestamps (60 minutes in a 5 minute resolution) for the missing hour each year consistently had exactly $kW=0$. The theory was therefore formed that these data points were added in afterwards.

Colorado switches *from* daylight saving the first Sunday in November each year [60], making the hour 01:00-02:00 repeat itself. Duplicates were therefore expected in the time series data, but similarly as before, no duplicate was found. The assumption was that one of the duplicated hours was removed afterwards - perhaps the second hour overwriting the first one in the database.

The reason why it was important to check the hours affected by the DST, was because the dataset was to be later merged with the JRC data. The JRC data is always in UTC format - which has no DST - and it is crucial to ensure that the datasets are fully aligned, as a failure to account for the DST offset would result in the two datasets being shifted by one hour relative to each other for half of the year.

5.2.2 Missing Timestamps and Values

The data contained no missing timestamps nor NaN-values. However, it contained many negative values of kW . These were set to be equal to 0 as they were assumed to be due to measurement noise.

5.2.3 Importing Data from JRC and Merging Data

Data from JRC was imported and engineered the exact same way as described in the corresponding chapter for the Australia Site. The datasets were then merged.

Using the JRC data, a few outliers could be detected and removed from the original data. These outliers significantly differed from both the surrounding data, but also JRC's P , which can be served as a guideline as it is represented an estimated power output of a PV system with similar characteristics.

Additionally, 26 days with zero generation throughout the entire day were excluded from the dataset, as these are assumed to represent system downtime rather than weather-driven variation. A peak of exactly 0 kW is implausible given the system's typical output of 3500+ kW on clear days. This could also be verified by comparing with P .

After the merge, timestamps affected by DST were looked at again. As previously discussed, the JRC data needed to be converted from UTC. As a consequence of this, NaN-values were *now* present in the data between 02:00-03:00 the second Sunday of each March, and the same with duplicates in November. It was decided that the NaN-rows were to be dropped, and that only one of each duplicate was to be kept.

P and kW were plotted against *time*, as seen in Figure 17. A benefit of plotting these together is that any temporal misalignment between the datasets would be immediately visible as an offset between the two curves.

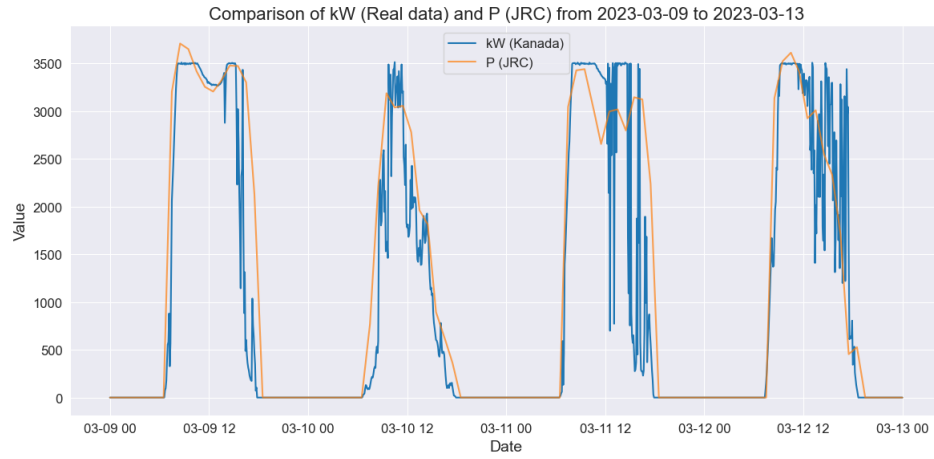


Figure 17: Real measured power vs estimated power through JRC.

5.2.4 Feature Engineering and Resampling

Additional features, such as features from `pvl` and temporal features, were added in the exact same way as described in the corresponding chapter for the Australia site. The only difference was that `CSI` was created using `G` instead of `Global_Horizontal_Radiation`. The USA dataset was also resampled to a 15 minute resolution instead of a 5 minute one.

5.3 Portugal Site



Figure 18: Satellite picture of Portugal solar site.

The Portuguese dataset originally contained 4 columns and 35063 rows. Two columns were deemed irrelevant and were therefore dropped straight away: `Specific Energy (kWh/kWp)` and `CO2 Avoided (tons)`. The remaining two columns were:

- `Date` - timestamp in WET/WEST (UTC+0/UTC+1) [1 hour resolution]
- `Produced Energy (kWh)` - produced energy [kWh]

Since the temporal resolution was 1 hour, the power would have the same numerical value as `Produced Energy (kWh)`:

$$P = \frac{E}{t} \implies kW = \frac{kWh}{h} \quad (27)$$

Therefore, the column was renamed to `kW` without any prior alterations. The column `Date` was renamed `time` and converted to datetime format. As usual, the two columns were then plotted against each other, see Figure 19.

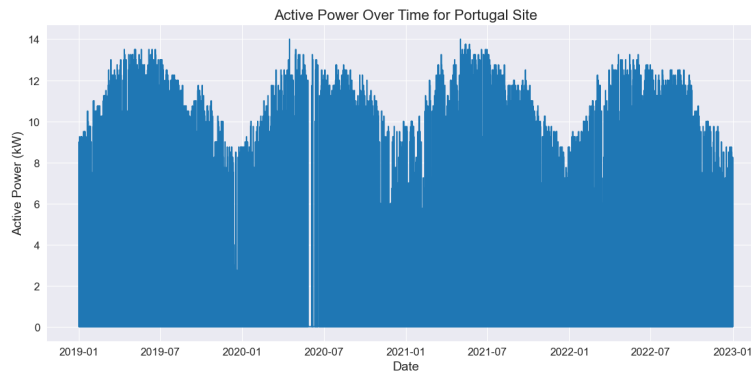


Figure 19: Active power over time for Portugal site.

5.3.1 Missing Timestamps and Values

The dataset contained no negative kW values, nor did it contain any NaN-values. There were only five missing timestamps; four of which were expected, as they all represented the non-existent hour when Portugal switches to DST in March. The fifth one was 2022-06-01 00:00, and as this was in the middle of the night, the timestamp was simply added with kW=0.

28 days contained kW=0 during daytime (between 11:00-15:00). As previously discussed, obtaining the exact value kW=0 is quite unlikely, even for an overcast day. These days were marked as suspicious and would be revisited later.

5.3.2 Exploration of Local Weather Variables

The Portuguese dataset included local weather variables that needed to be imported separately. The available columns were:

- time - timestamp in WET/WEST (UTC+0/UTC+1) [1 hour resolution]
- temperature_2m (°C) - temperature [°C]
- relative_humidity_2m (%) - relative humidity [%]
- dew_point_2m (°C) - dew point temperature [°C]
- apparent_temperature (°C) - perceived temperature [°C]
- cloud_cover (%) - cloud cover [%]
- wind_speed_10m (km/h) - wind speed [km/h]
- wind_direction_10m (°) - wind direction [°]
- shortwave_radiation (W/m²) - shortwave solar radiation [W/m²]

time was converted to datetime and the DataFrame was merged with the original data. All weather variables were plotted and inspected, but as opposed to the Australian dataset, no obvious outliers could be detected. See Figure 20.

5.3.3 Importing Data from JRC and Merging Data

Data from JRC was imported and engineered the exact same way as described in the corresponding chapter for the Australia Site, except that it was not resampled to a 5 minute resolution. In fact, the JRC data for the Portugal site had an hourly sampling interval offset by 10 minutes (i.e. recordings at HH:10 rather than HH:00). Since the offset was consistent across all timestamps, the JRC data was shifted back by 10 minutes to align with the top of each hour, enabling straightforward merging with the other dataset. The datasets were then merged in the same way as before. Figure 21 shows P and kW plotted against time.

Additionally, a few weather variables were plotted against time, where the local weather data was compared to the corresponding JRC data (such as temperature_2m (°C) and T2m, or wind_speed_10m (km/h and WS10m). This was mainly to confirm that the local weather data was both accurate and temporally aligned with the JRC reference data.

Recall that 28 days were previously identified as suspicious. By plotting these days and comparing to P as a reference point, they were examined more closely and then removed if the data appeared evidently faulty.

5.3.4 Daylight Saving Time

As stated, the non-existent timestamps when Portugal switches to DST were already missing from the dataset from the start. Consequently, no NaN-values were created when the dataset was merged with JRC, and no further action was required.

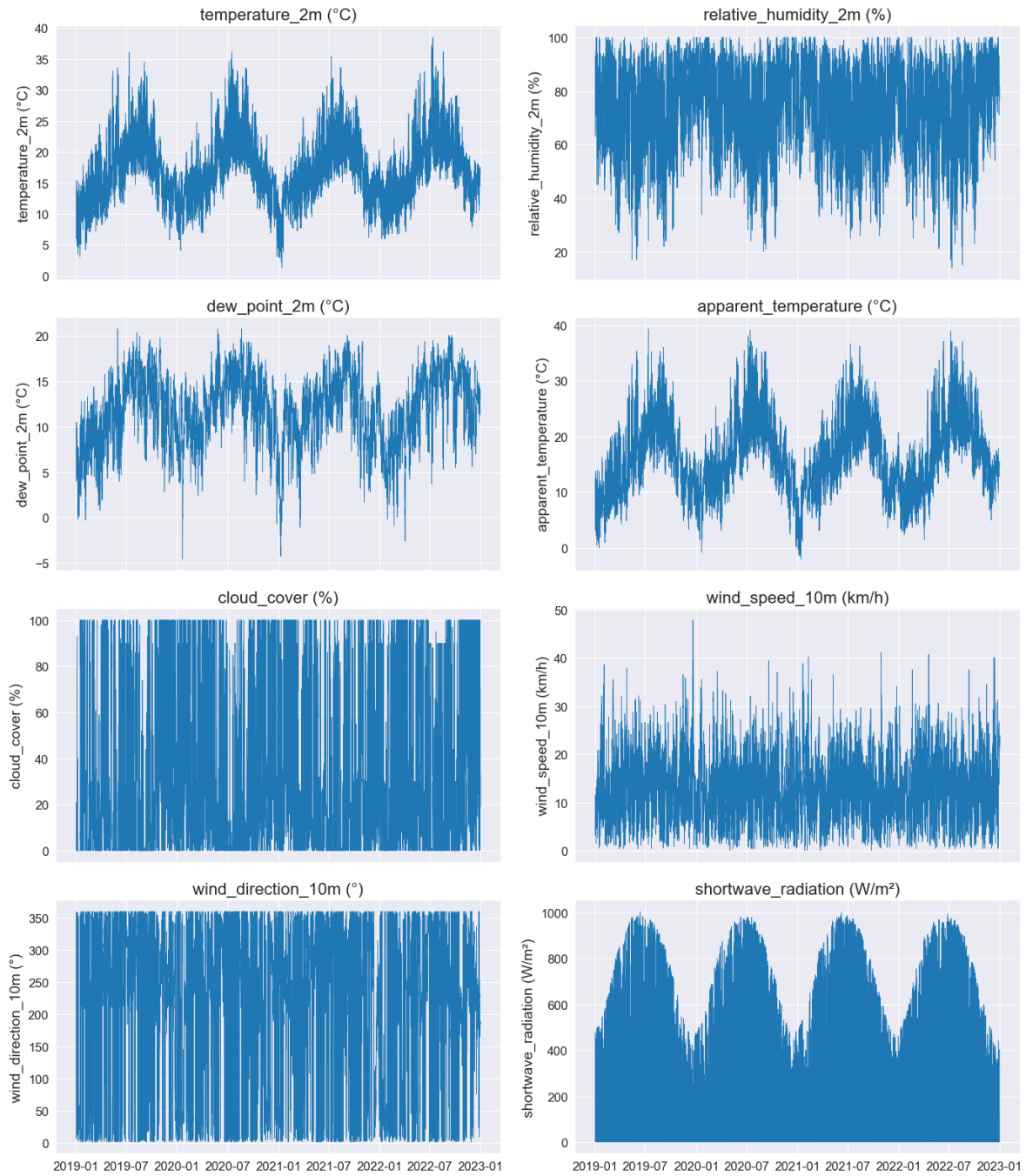


Figure 20: Local weather variables for Portugal site.

Duplicates existed in the dataset for the timestamps when Portugal switches from DST, which was also expected. Only one of each duplicate was kept, and the rest were dropped.

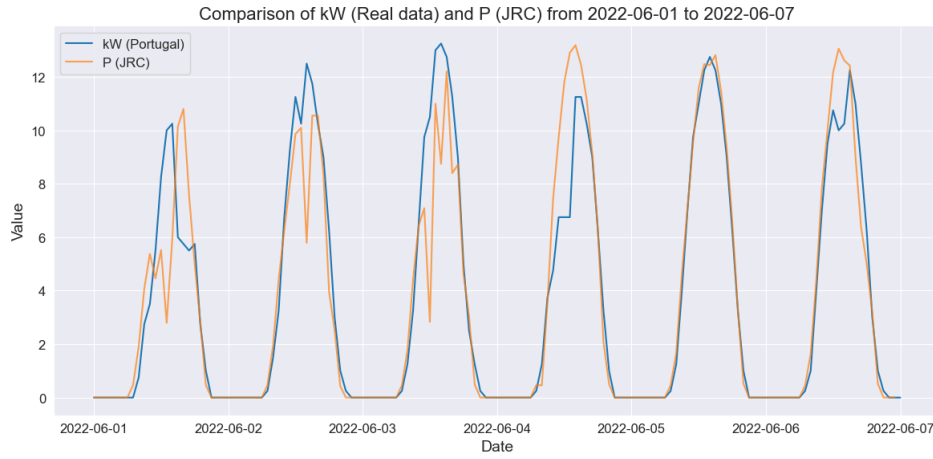


Figure 21: Real measured power vs estimated power through JRC.

5.3.5 Feature Engineering

Additional features, such as features from `pvl` and temporal features, were added in the exact same way as described in the corresponding chapter for the Australia site. `CSI` was created using `shortwave_radiation` as numerator. No resampling was required, as the 1-hour resolution, while coarser than the 15-minute resolution used elsewhere, could not be improved upon without introducing artificial data. Thus, 1-hour resolution was retained for this dataset.

5.4 Canada Site



Figure 22: Satellite picture of Canada solar site.

The Canadian dataset initially contained 8 columns and 258423 rows. All columns were dropped except for the following:

- name - PV site
- date - timestamp in MT (UTC -7/UTC-6) [1 hour resolution]
- kWh - produced energy [kWh]

kWh was renamed kW (using the same logic as discussed in the corresponding section for the Portuguese dataset), and date was renamed time and converted to datetime format. The Canadian dataset contained 11 different PV sites, all located in the city of Calgary (Alberta), distinguished by the column name. kW was plotted against time for each site, and eventually Richmond – Knob Hill Community Hall was chosen as it contained the most consistent data, with no obvious anomalies or data gaps detected.

The rows for all other sites were dropped, and the column name was dropped thereafter. Figure 23 shows kW plotted against time.

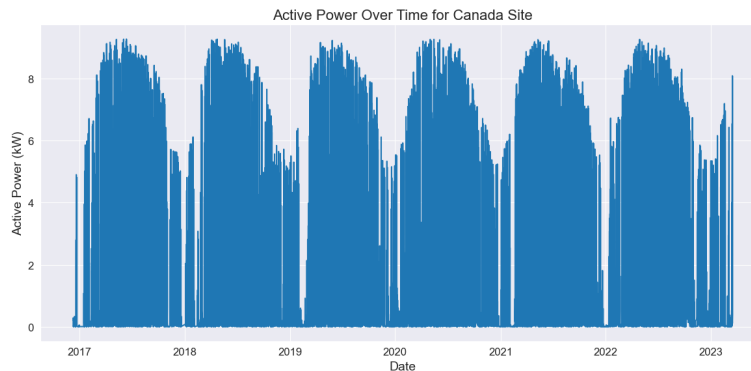


Figure 23: Active power over time for Canada site.

5.4.1 Missing Timestamps and Values

The column `kW` contained no negative values, nor any NaN-values. However, the dataset contained 28298 missing timestamps, which was found to be due to the systematic exclusion of nighttime. All missing timestamps were added with `kW=NaN`, and would be taken care of later on.

5.4.2 Importing Data from JRC and Merging Data

Data from JRC was imported and processed in the same manner as described for the Portugal site, except that it was shifted *forward* 30 minutes instead of *back* 10 minutes. The recordings for the Canadian JRC data were in the HH:30 format instead of HH:10. The forward shift was chosen under the assumption that each HH:30 recording represents the average of the interval leading up to the following hour. The datasets were then merged and plotted, see Figure 24.

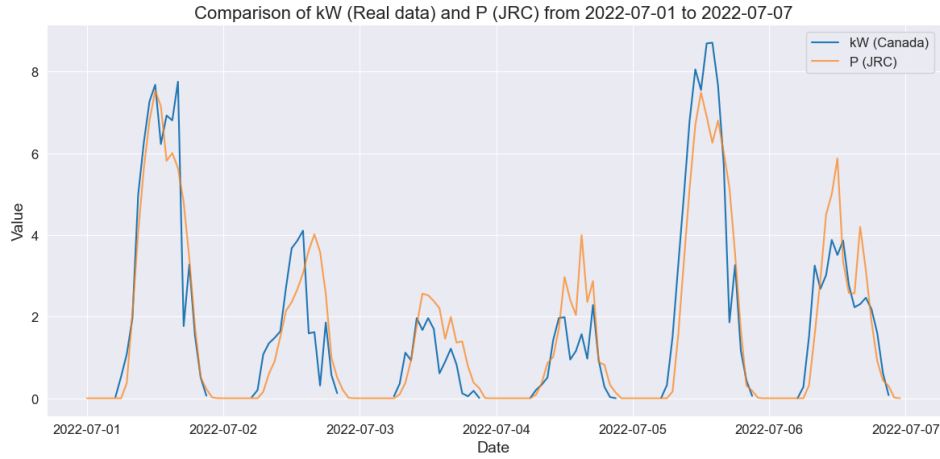


Figure 24: Real measured power vs estimated power through JRC.

The missing timestamps during night time in the original data can be clearly seen in Figure 24. Using the newly added feature `G`, all `kW=NaN` were replaced with 0 in rows where `G=0` was true. This was a convenient way to fill in the missing nighttime values, as `G=0` naturally indicates when it is nighttime - removing the need to manually specify a fixed nighttime interval. Furthermore, isolated NaN-values (i.e. single missing values surrounded by valid data) were linearly interpolated. Figure 25 shows the difference in NaN-hours before and after these alterations. The remaining rows with `kW=NaN` were dropped.

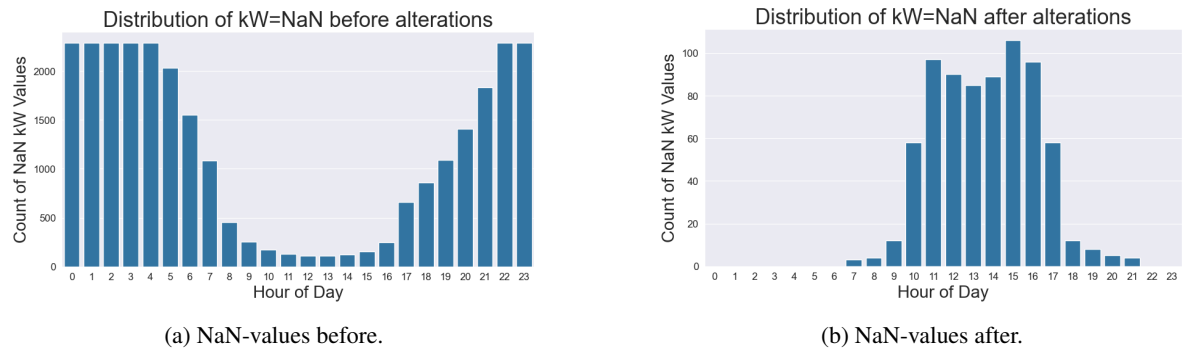


Figure 25: NaN-values (from missing timestamps) before and after alterations.

5.4.3 Daylight Saving Time

Timestamps affected by the Alberta daylight saving transition in March could be identified by locating rows where $P=\text{NaN}$. These rows were dropped. Similarly, DST transitions in November were identified by finding duplicated timestamps in the dataset. Only one of each duplicate was kept.

5.4.4 Feature Engineering

Additional features, such as features from `pvl` and temporal features, were added in the exact same way as stated in the corresponding chapter for the USA site. Like the Portuguese dataset, the Canadian dataset was not resampled.

5.5 Feature Selection

As RQ1 entails comparing the performance of forecasting models in different climate zones, an immediate problem was introduced when it came to feature selection. As features are the input parameters for ML models, *the performance of a forecasting model will be directly affected by the quality of its features*. Since the Australian and Portuguese datasets contained locally measured weather variables, the performance of an ML model using these features might get an immediate advantage compared to the ML models using JRC's modelled features. Furthermore, the higher temporal resolution of the Australian and American datasets - compared to the hourly resolution of the Portuguese and Canadian datasets - may also influence model performance.

With this logic, *no* local weather variables should be allowed as features, and *all* datasets should have a 1-hour temporal resolution, in order to allow for a completely fair comparison between performances. However, discarding this information would forfeit the opportunity to explore the full potential of each site - that is, how well a model can truly perform when given the best possible conditions.

Therefore, it was decided that two different versions of each dataset were to be created:

- **Enhanced version:**
 - Highest possible (max 15 min) temporal resolution
 - Optimal site specific feature selection
- **Standardised version:**
 - 1 hour temporal resolution
 - Standardised feature selection

Features were selected by generating Pearson correlation heatmaps and pairplots for all possible combinations of features. Identifying correlations within the data is a critical step in statistical modelling, as it enables the selection of strong predictors of the target variable [24]. Figures 26 and 27 shows examples of one of the Pearson correlation heatmaps and pairplots produced in the feature selection of the Enhanced Australian dataset.

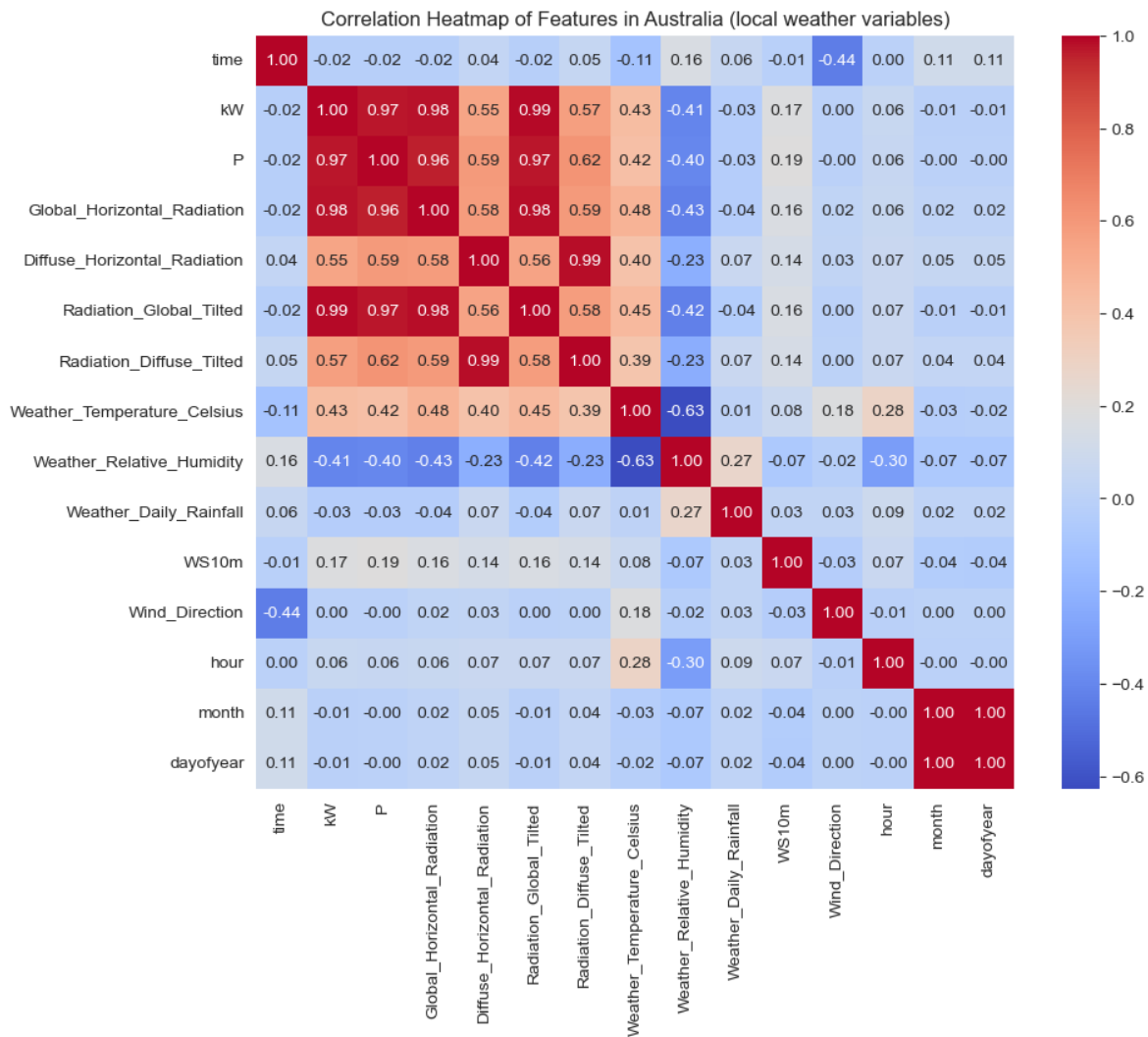


Figure 26: Example of Pearson correlation heatmap. More heatmaps can be found in Appendix B.

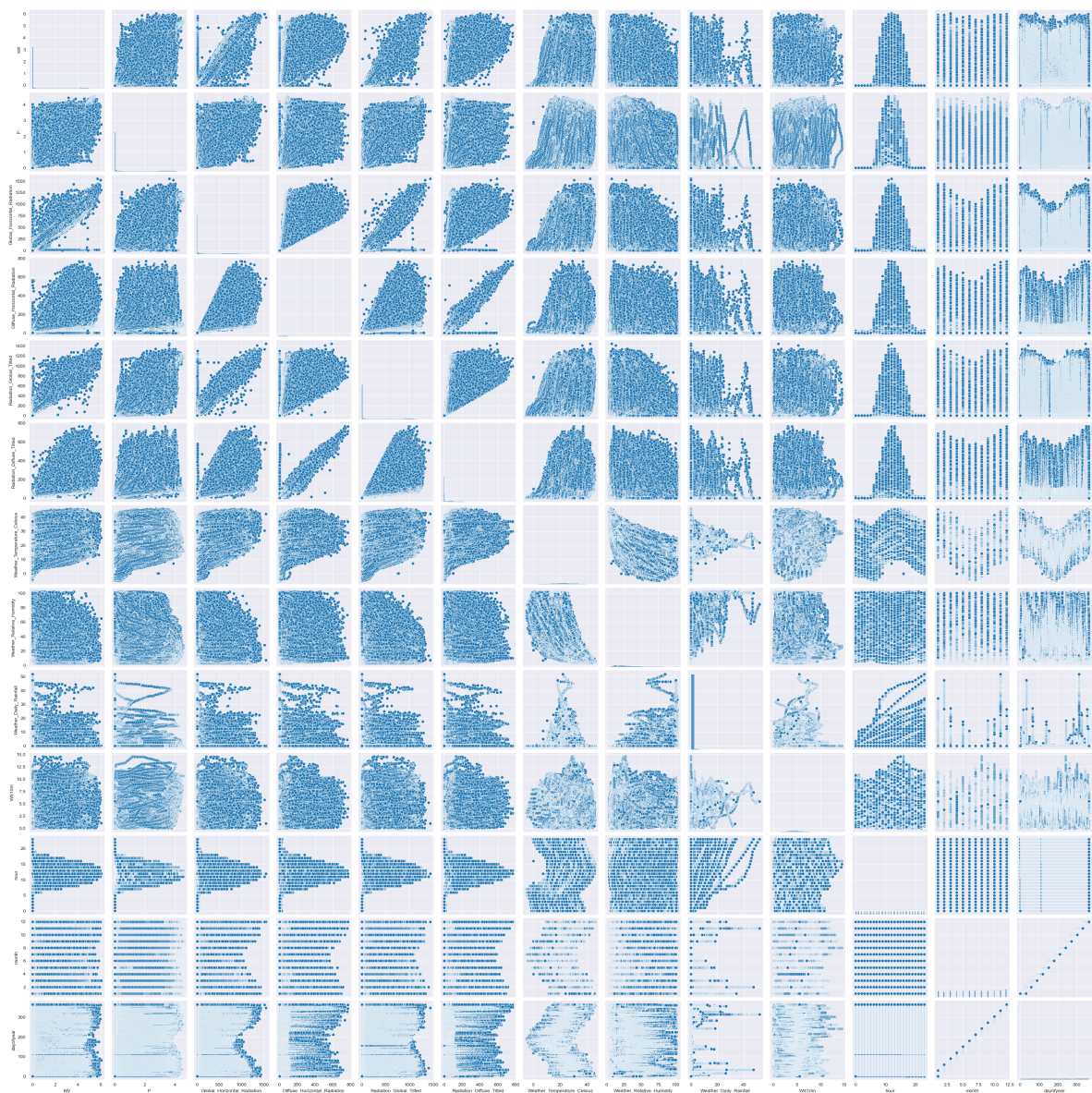


Figure 27: Example of pairplot. More pairplots can be found in Appendix C.

When selecting features for the Standardised datasets, it was important to choose features that were suitable for *all* sites. Fortunately, this proved relatively straightforward, as the predictive strength of the features did not vary substantially across sites. Ultimately, 13 features were identified as sufficient for use across all datasets. From now on, these are referred to as the *Standardised features*, and consist of the following:

- `Gb(i)`
- `Gd(i)`
- `Gr(i)`
- `solar_elevation`
- `solar_azimuth`
- `ghi_clear`
- `CSI`
- `WS10m`
- `T2m`
- `tod_sin`
- `tod_cos`
- `doy_sin`
- `doy_cos`

`CSI` in this case refers to `G/ghi_clear`. The reason why *both* sine and cosine components are included for each temporal variable is that using either alone would produce an ambiguous encoding. For example, 06:00 and 18:00 both yield identical sine values, making them indistinguishable to the model. Together, the two components uniquely identify every point within a cycle, ensuring a correct circular encoding of time.

It turned out that the Standardised features were optimal for the Enhanced American and Canadian dataset as well, which might not be too surprising considering they lacked local weather variables. However, a slightly different feature selection was made for the Enhanced Australian and Portuguese datasets, as the local weather variables were generally preferred for several reasons. First, they offered a broader set of variables to choose from, such as humidity. Second, local weather variables are *measured* rather than modelled, which may yield greater accuracy. For the Australian dataset specifically, a third advantage was resolution: the JRC features were originally provided at an hourly resolution and subsequently linearly interpolated, whereas the local weather variables were available at a native 15-minute resolution, meaning no interpolation was required. Table 4 shows a compilation of the temporal resolution and selected features of each dataset.

Table 4: Overview of selected features for each site and dataset version

Site	Version	Temporal Resolution	Selected Features
Australia	Enhanced	15 min	Global.Horizontal.Radiation Diffuse.Horizontal.Radiation solar.elevation solar.azimuth ghi.clear CSI WS10m Weather.Temperature.Celsius Weather.Relative.Humidity tod.sin tod.cos doy.sin doy.cos
	Standardised	1 h	Standardised features
USA	Enhanced	15 min	Standardised features
	Standardised	1 h	Standardised features
Portugal	Enhanced	1 h	shortwave_radiation (W/m ²) Gb(i) Gd(i) solar.elevation solar.azimuth ghi.clear CSI temperature.2m (°C) relative_humidity.2m (%) cloud.cover (%) wind.speed.10m (km/h) tod.sin tod.cos doy.sin doy.cos
	Standardised	1 h	Standardised features
Canada	Enhanced	1 h	Standardised features
	Standardised	1 h	Standardised features

6 Forecasting Performance Across Climate Zones

6.1 Methodology

RQ1 is intentionally formulated as an open-ended question, imposing no a priori bias toward any particular class of models. To ensure that the selection of ML models was not arbitrary, a systematic approach was adopted for identifying which models to include in this study. Five different articles were read thoroughly, namely: [3], [7], [6], [42] and [8]. What these articles have in common is that they all compare the performance of different machine learning models on PV data and rank them based on their performance. By compiling the results from these articles and assigning internal scores to each model based on its ranking position in each respective article - where a first-place ranking awarded five points, second place four points, and so on - the four ML models with the highest cumulative scores were selected for this study. These ML models were: Random Forest, XGBoost, Ridge Regressor and Support Vector Regressor.

Apart from the ML models, the Naïve Persistence Model was chosen as a baseline model. Furthermore, DL models were to be chosen in the same manner as the ML models; LSTM and RNN rank highly in [3] and [6], and [42] compares a Bi-LSTM, CNN-LSTM and a Stacked LSTM (among other models), where the Stacked LSTM performs on top. Since the LSTM model was reoccurring, in either its true or hybrid form, it was chosen as a first DL model for the study. While the initial ambition was to include a greater number of deep learning models, or even hybrid models, time constraints ultimately limited the selection to only one. It should be noted that DL models technically are ML models as well; this study chooses to distinguish them in order to reflect the conventional separation used in the literature.

6.1.1 Naïve Persistence

The Naïve Persistence model was very simple to implement. Firstly, all NaN-rows were dropped. Then, the predictions were equal to the actual value shifted one time step forward, as Equation 2 suggests. Figure 28 shows three days of the Naïve Persistence model plotted for the Enhanced Australian dataset.

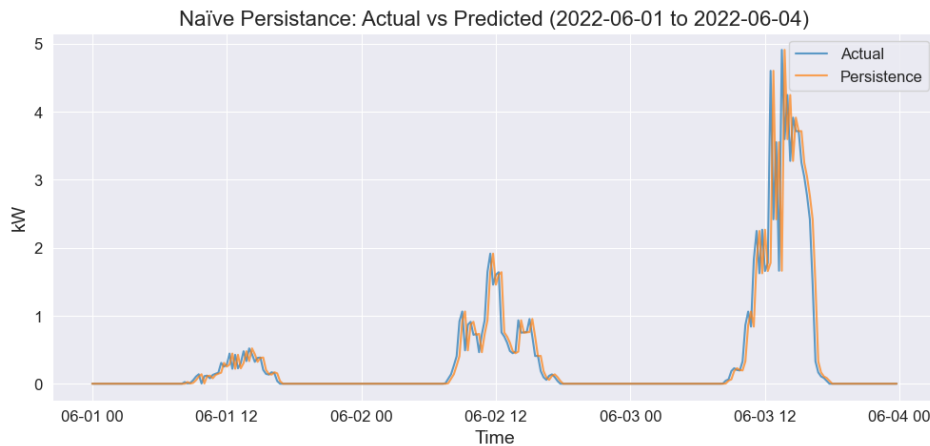


Figure 28: Naïve Persistence, Enhanced Australian dataset.

6.1.2 ML Models

All ML models were developed and trained in the same way. Firstly, two DataFrames were created, X containing the selected features, and y containing the target column kW. The data was then split into training and testing subsets, with an 80/20 ratio. X was scaled using a `StandardScaler()`, and NaN-rows were dropped.

Next, Optuna was used to systematically optimise the hyperparameters of each model, ensuring that performance differences between models reflect their true predictive capabilities rather than arbitrary hyperparameter choices. When creating an Optuna objective function, two things need to be defined: the objective metric and the hyperparameter bounds. RMSE was used as the objective metric to minimise, with $TSCV=5$ employed for cross-validation. [7], [42] and [8] report the hyperparameters used in their respective studies, which served as a starting point for defining the bounds in this study. The bounds were then refined through an iterative process; if Optuna consistently converged towards an optimal solution at one of the boundary extremes, the bounds were adjusted to cover a wider range. Convergence could be identified by examining a Slice Plot (see Figure 29) or a Parallel Coordinate plot (see Figure 30).

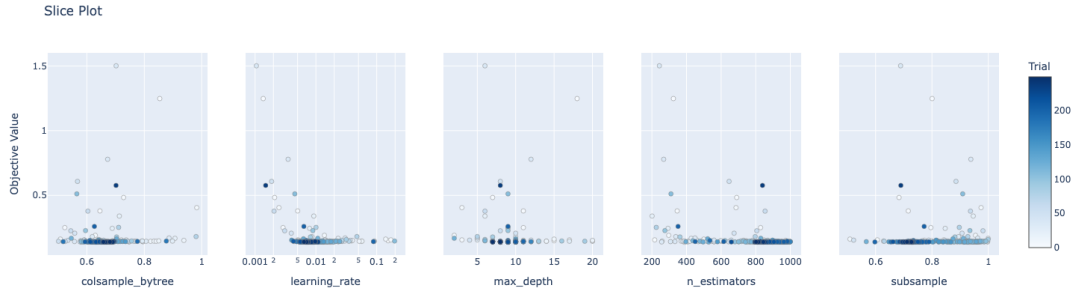


Figure 29: Example of Slice Plot.

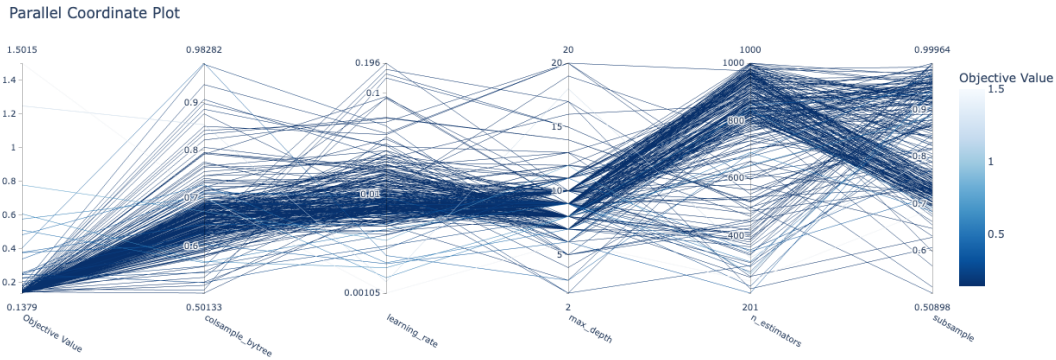


Figure 30: Example of Parallel Coordinate Plot.

The finalised hyperparameter bounds are shown in Table 5. The same hyperparameter bounds were used for all datasets.

All models were optimised using 100 trials, except for the SVR which only employed 15 trials. The reason why SVR was limited to 15 trials was because of time constraints; while most forecasting models could go through 100 trials in less than 30 minutes, 15 trials of SVR took up to 53 hours.

Most studies would converge after a few trials, with the remaining trials largely serving for fine-tuning rather than yielding any substantial improvements. An example of this is shown in Figure 31.

Table 5: Hyperparameter bounds used for ML models.

Model	Hyperparameter	Range / Options
Random Forest	n_estimators	20 – 400
	criterion	squared_error, friedman_mse
	max_depth	5 – 50
	min_samples_split	2 – 20
	min_samples_leaf	1 – 20
	max_features	None, sqrt, log2
XGBoost	n_estimators	200 – 1000
	learning_rate	0.001 – 0.2 (log)
	max_depth	2 – 20
	subsample	0.5 – 1.0
	colsample_bytree	0.5 – 1.0
Ridge Regressor	alpha	0.01 – 1000 (log)
	solver	auto, svd, cholesky, lsqr
SVR	C	1 – 100 (log)
	gamma	0.0001 – 1 (log)
	epsilon	0.001 – 1 (log)
	kernel	rbf (fixed)

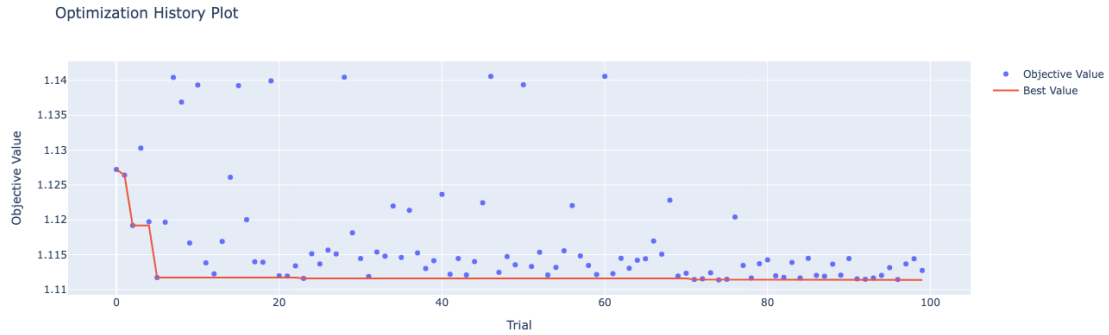


Figure 31: Example of Optimization History Plot. As seen, the objective value converges after a few trials.

The models were then trained on the training data using the optimal hyperparameters identified by the respective Optuna study. Predictions were subsequently generated by applying the trained models to the test data.

6.1.3 LSTM (DL)

LSTM is commonly implemented using PyTorch (developed by Facebook AI Research) or TensorFlow (developed by Google). While PyTorch allows for greater flexibility in model experimentation, TensorFlow was shown by [9] to provide more stable and reliable predictions, and was therefore chosen for this study.

In contrast to the ML models previously discussed, LSTM uses *sequences* as input rather than individual feature vectors:

$$X = (X_{t-L+1}, X_{t-L+2}, \dots, X_{t-1}, X_t) \quad (28)$$

where L represents the sequence length, t denotes the current timestamp, and any X_t is composed of the selected

features for the given timestamp. [3]

The process of creating sequences for x and y was meticulous. Since some of the datasets contained missing timestamps, some contained missing values, and some contained both, a robust approach needed to be implemented in order to generalise the process. The problem with simply ignoring missing timestamps and/or dropping rows containing NaNs, is that the look-back window will be unaware of this fact. For example, suppose that look-back window (sequence length) is equal to 24 (meaning that the LSTM model will look at the past 24 hours to make the next prediction), and that there is a gap of a few hours, or even a few days, in the data. Since the look-back window is ignorant to the gap, it will form a sequence where the final observations before the gap and the first observations after are treated as adjacent timesteps, despite representing non-consecutive points in time. This may cause the model to learn erroneous temporal patterns.

Keeping this in mind, the following process was implemented for all datasets:

- Missing timestamps were imputed, ensuring a continuous temporal range.
- Data was split into training (64%), validation (16 %) and test (20%) sets, maintaining chronological order. The reason why validation data was used rather than `TSCV` was because LSTM models require a fixed validation set during training to monitor performance and apply early stopping.
- `sequence_length` was set to 24 for datasets with a 1 hour resolution and 96 for datasets with a 15 minute resolution, making the look-back window correspond to 24 hours regardless of dataset.
- Sequences were first constructed across the full dataset. Subsequently, any sequence containing NaN values - whether from imputed timestamps or missing feature values - was discarded, retaining only sequences consisting solely of consecutive, valid data points.
- `X_train_seq`, `X_val_seq` and `X_test_seq` were transformed using `StandardScaler()`, but were only fitted on the training data to prevent data leakage.

Next, the LSTM model was defined using the optimiser `Adam` and the activation functions `tanh` (in the LSTM-layer) and `relu` (in the dense layer). The objective function was defined in the same way as before, and 30 trials per study were used for each dataset. The hyperparameter bounds for Optuna are shown in Table 6.

Table 6: Hyperparameter bounds used for LSTM model.

Model	Hyperparameter	Range / Options
LSTM	<code>n_lstm_units</code>	32 – 128
	<code>dropout_rate</code>	0.1 – 0.5
	<code>n_dense_units</code>	16 – 64
	<code>learning_rate</code>	0.0001 – 0.01 (log)
	<code>batch_size</code>	16 – 64 (step=16)

As previously, the LSTM model was finally trained using the optimal hyperparameters, and predictions were subsequently generated. The number of epochs in each Optuna trial was 50, while the number of epochs for the final training was 100. The reason was to reduce computational cost during hyperparameter optimisation, while allowing the final model to train for longer and potentially achieve better performance.

6.2 BWh - Australia

The performance of the forecasting models on both versions of the Australian data is presented in Table 7.

Table 7: Forecasting performance metrics for Australian datasets. Bold indicates the best-performing model for each metric; red shading indicates performance worse than the Naïve Persistence baseline.

Model	RMSE [kW]	MAE [kW]	sMAPE [%]	nRMSE [%]
<i>Enhanced dataset</i>				
Naïve Persistence	0.406	0.168	18.613	33.827
Random Forest	0.090	0.037	6.301	7.473
SVR	0.101	0.042	7.400	8.427
XGBoost	0.090	0.037	6.425	7.529
Ridge Regressor	0.248	0.194	17.608	20.635
LSTM	0.314	0.125	18.133	26.174
<i>Standardised dataset</i>				
Naïve Persistence	0.749	0.414	41.752	62.351
Random Forest	0.439	0.174	22.676	36.524
SVR	0.464	0.200	26.121	38.597
XGBoost	0.440	0.181	24.151	36.630
Ridge Regressor	0.454	0.219	26.550	37.734
LSTM	0.464	0.222	27.568	38.663

As seen, Random Forest outperforms the other models across all metrics, with XGBoost being a close follow-up. Figure 32 illustrates Table 7 in two histograms. It is clear that RF, SVR and XGBoost are significantly better than the other two models in the Enhanced dataset, while all models perform about the same in the Standardised dataset.

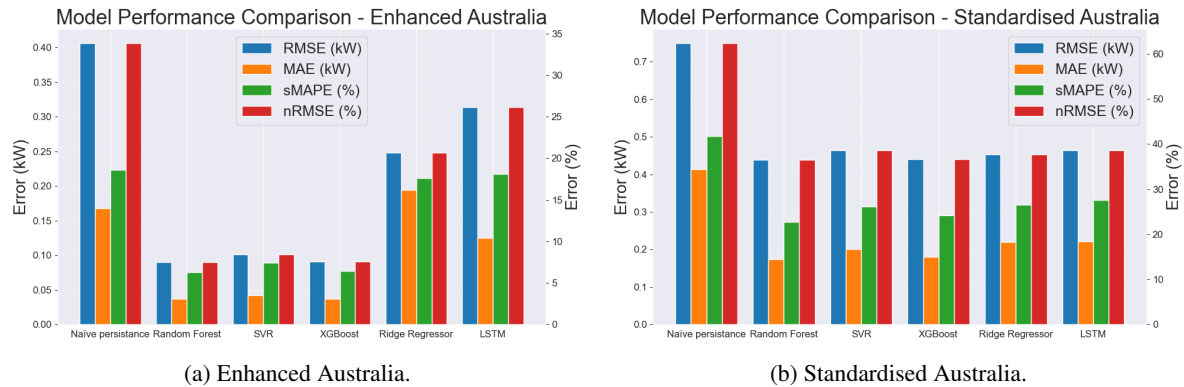


Figure 32: Forecasting performance metrics for Australian datasets.

Figure 33 shows the actual and predicted power production across five days for both the Enhanced and Standardised Australian datasets. The Ridge Regressor in the Enhanced version was the only model performing worse than the baseline model in one of the metrics, and Figure 33a presents a possible explanation as to why; the prediction never fully reaches zero during night time, and is the only model not being able to capture the full peak on clear days. Interestingly though, the Ridge Regressor is able to capture the clear days in the Standardised dataset. However, all models perform much worse in this dataset, where several models are unable to

remain at zero during nighttime, and all models struggle with overcast days.

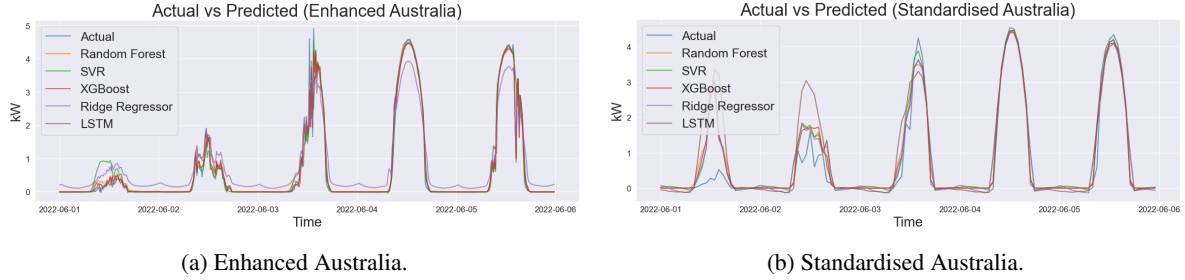


Figure 33: Five days plotted for Australian datasets.

6.3 BSk - USA

The performance of the forecasting models on both versions of the American data is presented in Table 8.

Table 8: Forecasting performance metrics for American datasets. Bold indicates the best-performing model for each metric; red shading indicates performance worse than the Naïve Persistence baseline.

Model	RMSE [kW]	MAE [kW]	sMAPE [%]	nRMSE [%]
<i>Enhanced dataset</i>				
Naïve Persistence	367.765	135.962	17.219	41.074
Random Forest	576.861	291.126	42.520	64.427
SVR	608.879	318.688	46.714	68.003
XGBoost	577.996	298.854	44.238	64.554
Ridge Regressor	596.045	340.867	46.956	66.570
LSTM	563.940	282.359	44.154	62.984
<i>Standardised dataset</i>				
Naïve Persistence	690.725	317.943	37.536	76.906
Random Forest	574.498	290.441	42.404	63.965
SVR	608.124	279.411	42.796	67.709
XGBoost	573.273	293.556	43.019	63.829
Ridge Regressor	593.698	342.342	46.200	66.103
LSTM	585.952	302.111	46.000	65.241

Interestingly, no model outperforms the Naïve Persistence model in the Enhanced dataset. Since the Enhanced and Standardised versions of the American data both use the *exact same features*, this suggests that data resolution has an influence on model performance.

Another interesting observation is that all models perform worse than the baseline model when using the sMAPE-metric on the Standardised dataset. A possible explanation to this is that `StandardScaler()` centres the data around zero, resulting in negative values. This can cause the denominator in the sMAPE formula, $|y_t| + |\hat{y}_t|$, to approach zero when predictions and actual values have opposite signs, artificially inflating the metric. The Naïve Persistence model is less susceptible to this issue, as its predictions are by definition close to the actual values, keeping the denominator more stable. With this being said, the sMAPE metric might not be suitable to use in combination with data scaled by `StandardScaler()`.

Figure 34 illustrates Table 8 in two histograms. As seen, no model significantly outperforms the others in either dataset.

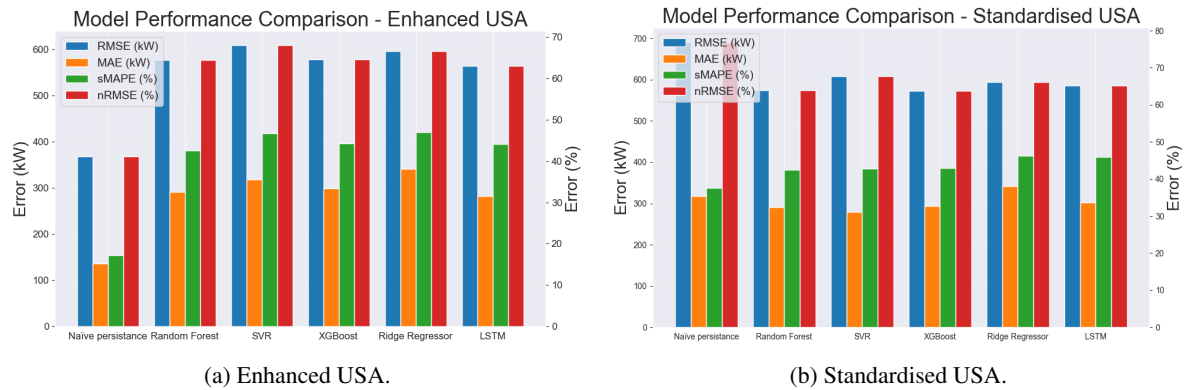


Figure 34: Forecasting performance metrics for American datasets.

Figure 35 shows the actual and predicted power production across five days for both the Enhanced and Standardised American datasets. Once again, the Ridge Regressor struggles with remaining zero during nighttime. Remarkably, the difference in time resolution is visually apparent from the figure. In the Enhanced dataset, the actual power output fluctuates rapidly with high variance - sharp peaks and troughs occurring within minutes. However, the models fail to capture these rapid changes, which is expected given that the features were originally recorded at 1-hour resolution and subsequently linearly interpolated. The mismatch between the high-frequency target variable and the lower-frequency features likely explains why all models perform worse than the Naïve Persistence baseline in the Enhanced dataset. In the Standardised dataset, both the target variable and the features share the same 1-hour resolution, resulting in smoother dynamics that the models are better equipped to handle.

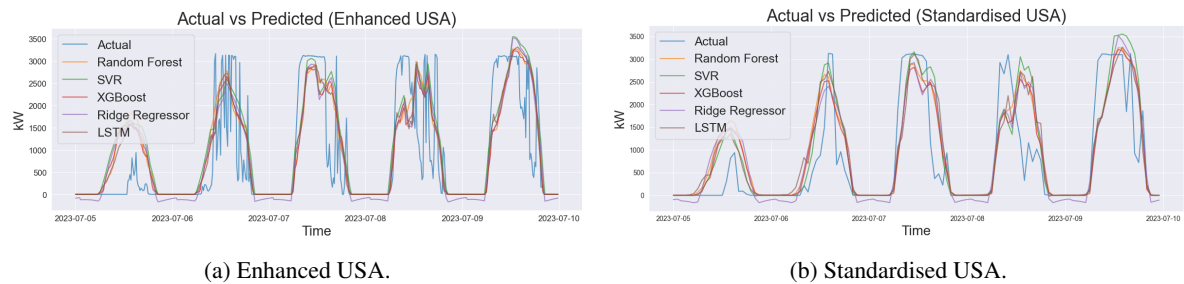


Figure 35: Five days plotted for American datasets.

6.4 Csa - Portugal

The performance of the forecasting models on both versions of the Portuguese data is presented in Table 9.

Table 9: Forecasting performance metrics for Portuguese datasets. Bold indicates the best-performing model for each metric; red shading indicates performance worse than the Naïve Persistence baseline.

Model	RMSE [kW]	MAE [kW]	sMAPE [%]	nRMSE [%]
<i>Enhanced dataset</i>				
Naïve Persistence	1.469	0.896	40.526	51.857
Random Forest	1.077	0.483	23.427	38.016
SVR	1.144	0.500	26.750	40.367
XGBoost	1.072	0.505	23.563	37.833
Ridge Regressor	1.212	0.776	33.906	42.764
LSTM	1.071	0.506	26.078	37.797
<i>Standardised dataset</i>				
Naïve Persistence	1.469	0.896	40.526	51.857
Random Forest	1.102	0.494	23.702	38.882
SVR	1.181	0.512	27.179	41.676
XGBoost	1.099	0.520	23.817	38.795
Ridge Regressor	1.203	0.726	33.325	42.448
LSTM	1.073	0.516	30.096	37.875

LSTM outperforms all other models across the RMSE and nRMSE metrics, while RF performs the best across the MAE and sMAPE metrics. Interestingly, there is not a major difference in the results overall between the two datasets, with the Enhanced version only performing slightly better across each respective metric. Furthermore, all models perform better than the Naïve Persistence model, which stands in stark contrast to the results observed for the American dataset. Figure 36 illustrates Table 9 in two histograms.

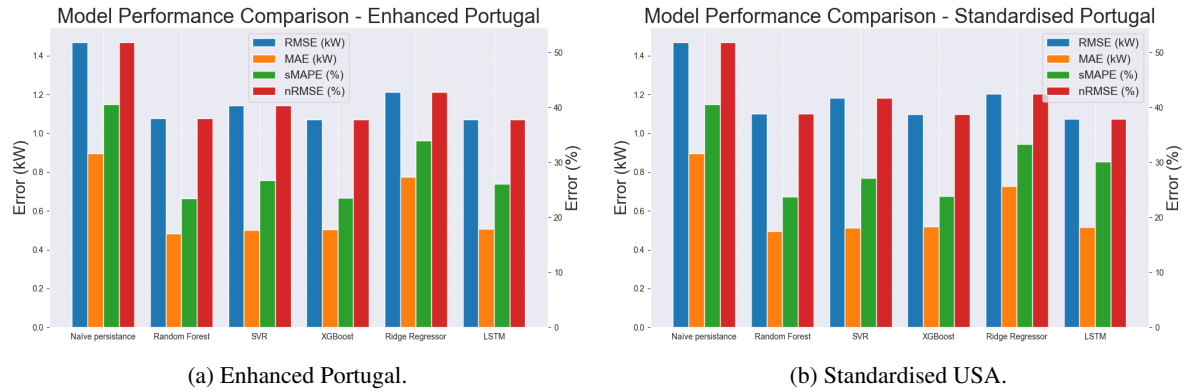


Figure 36: Forecasting performance metrics for Portuguese datasets.

Figure 37 shows the actual and predicted power production across five days for both the Enhanced and Standardised Portuguese datasets, and as seen, the Ridge Regressor struggles during nighttime once more. The most notable difference between the two subplots is that the LSTM model seems to capture the overcast nature of the fourth day (19/10-2022) better than the other models in the Enhanced version - perhaps indicating the LSTM model is using a local weather feature that the other models fail to fully exploit.

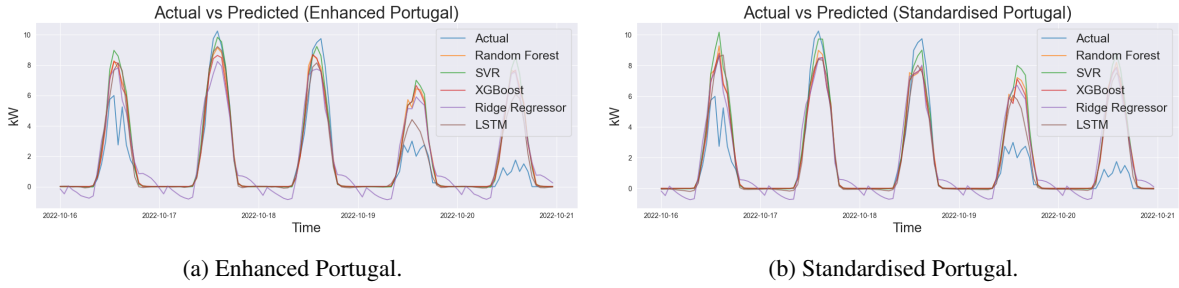


Figure 37: Five days plotted for Portuguese datasets.

6.5 Dfb - Canada

As there is no difference between the Enhanced and Standardised versions of the Canadian data, only one version was created referred to as the *Canadian dataset*. The performance of the forecasting models on the Canadian dataset is presented in Table 10.

Table 10: Forecasting performance metrics for the Canadian dataset. Bold indicates the best-performing model for each metric; red shading indicates performance worse than the Naïve Persistence baseline.

Model	RMSE [kW]	MAE [kW]	sMAPE [%]	nRMSE [%]
Naïve Persistence	0.919	0.474	47.818	70.118
Random Forest	0.843	0.379	38.062	64.301
SVR	0.858	0.373	42.244	65.453
XGBoost	0.842	0.388	39.638	64.221
Ridge Regressor	0.945	0.575	47.300	72.049
LSTM	0.937	0.457	43.946	71.481

Interestingly, the best performing forecasting model varies across the different metrics. Furthermore, the LSTM and the Ridge Regressor perform worse than the Naïve Persistence model in some of the metrics, but not in all. Figure 38 illustrates Table 10 in a histogram.

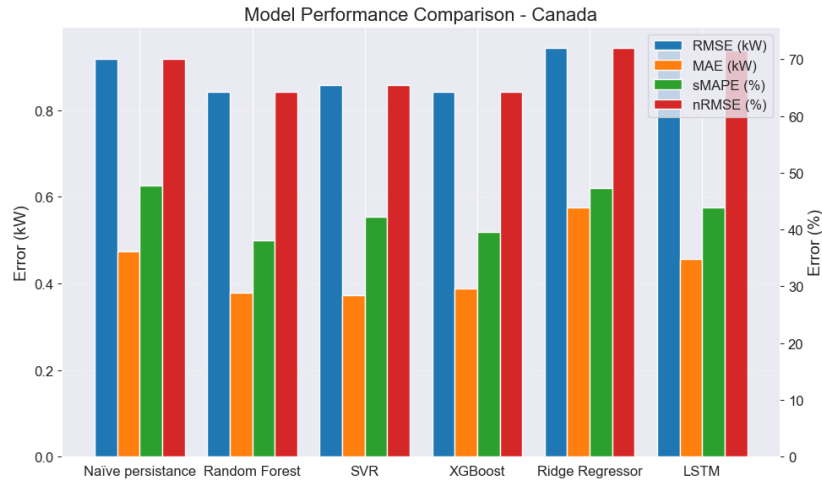


Figure 38: Forecasting performance metrics for the Canadian dataset.

Figure 39 shows the actual and predicted power production across five days for the Canadian dataset. Once again, the Ridge Regressor struggles during nighttime.

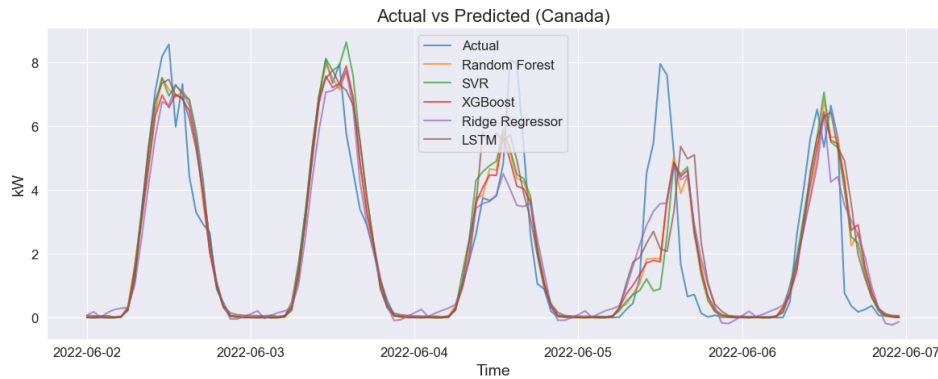


Figure 39: Five days plotted for the Canadian dataset.

Recall that Dfb is known for having severe winters. A very important aspect in PV production is that *snow blocks solar panels* when it settles on their surface, effectively reducing or completely halting energy generation. To see how the forecasting models react to this, an experiment was conducted; by taking a week where the temperature never exceeded 0°C in Calgary (16/12/2022-22/12/2022), JRC's prediction $\mathbb{P}$ was plotted together with the actual production kW , as shown in Figure 40.

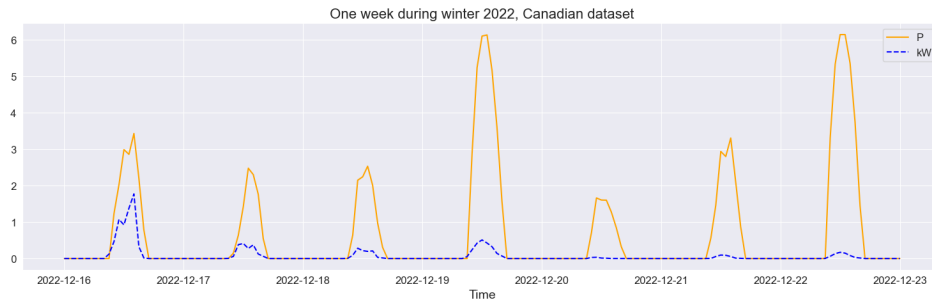


Figure 40: JRC's prediction vs actual production during one week in winter for the Canadian dataset.

Strikingly, there is a major difference between the prediction and the actual production - a difference which is not as apparent when plotting summer weeks. While it cannot be confirmed with certainty that snow accumulation on the panels is the cause, the discrepancy is strongly suggestive. This raises a further question: if JRC's irradiance-based prediction $\mathbb{P}$ overestimates production during such conditions, could the forecasting models suffer from the same bias? Given the near-perfect linear relationship between $\mathbb{P}$ and the irradiance features $\text{Gr}(i)$ and $\text{Gb}(i)$ used by the models, any systematic overestimation embedded in these features may propagate into the model predictions as well. To investigate whether the models accurately capture the reduced production during winter conditions - or whether they too overestimate output - the same week is plotted for each forecasting model below.

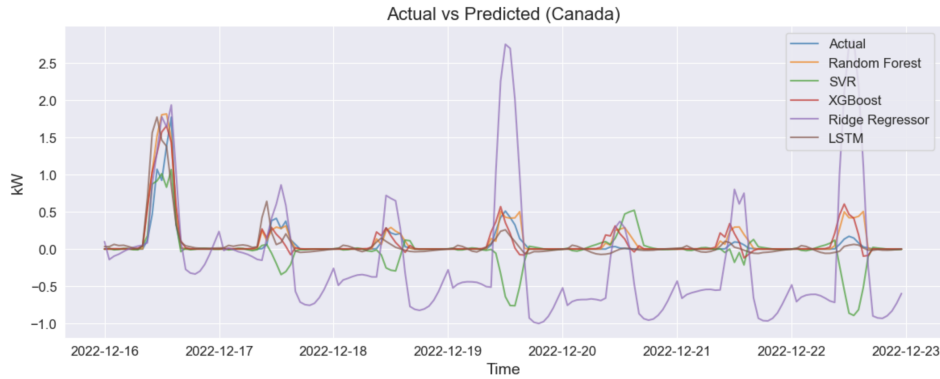


Figure 41: One week in winter plotted for the Canadian dataset.

A few observations can be made from Figure 41. Firstly, the Ridge Regressor has a *very* similar pattern as P in Figure 40, indicating that $G_T(i)$ and/or $G_b(i)$ are dominant features. Secondly, the SVR model actually predicts *negative* power production during daytime, which of course is unreasonable. Interestingly though, RF, XGBoost and LSTM produce predictions close to the actual production, suggesting that these models have learned seasonal patterns and implicitly account for the reduced output expected during winter months.

6.6 Comparative Analysis

6.6.1 Cross-Country Performance

To address RQ1, the performance of each forecasting model is now compared across the four climate zones. To ensure a fair comparison, only the Standardised datasets are considered.

While four metrics have been reported throughout this study, the following analysis relies solely on nRMSE, given its scale-independent nature and thus greater comparability across datasets. It should be noted that sMAPE also is scale-independent, but as discussed in Section 6.3, it might not be a suitable metric to use here.

Table 11: Cross-country comparison of nRMSE [%] across climate zones using Standardised datasets. Bold indicates the climate with the best performance for each model.

Model	BWh	BSk	Csa	Dfb
Naïve Persistence	62.372	76.906	51.857	70.118
Random Forest	36.524	63.965	38.882	64.301
SVR	38.597	67.709	41.676	65.453
XGBoost	36.630	63.829	38.795	64.221
Ridge Regressor	37.734	66.103	42.448	72.049
LSTM	38.663	65.241	37.875	71.481

As shown in Table 11, all models achieve their best performance in the BWh climate zone (Australian dataset), except for the baseline model and the LSTM, which both perform the best in the Csa climate zone (Portuguese dataset). Meanwhile, the BSk and Dfb climate zones (American and Canadian datasets) yield considerably worse performance, yet remain comparable to one another. Before drawing further conclusions from this, an additional analysis was conducted to better understand the underlying reasons for these differences.

Inspired by the experiment conducted in Section 6.5, where one week in winter was analysed for the Canadian dataset, the monthly nRSME was plotted for each model and climate zone, as seen in Figure 42. Since different

climate zones exhibit distinct seasonal patterns, evaluating model performance on a monthly basis allows for a more complete assessment of how well each model captures these variations throughout the year.

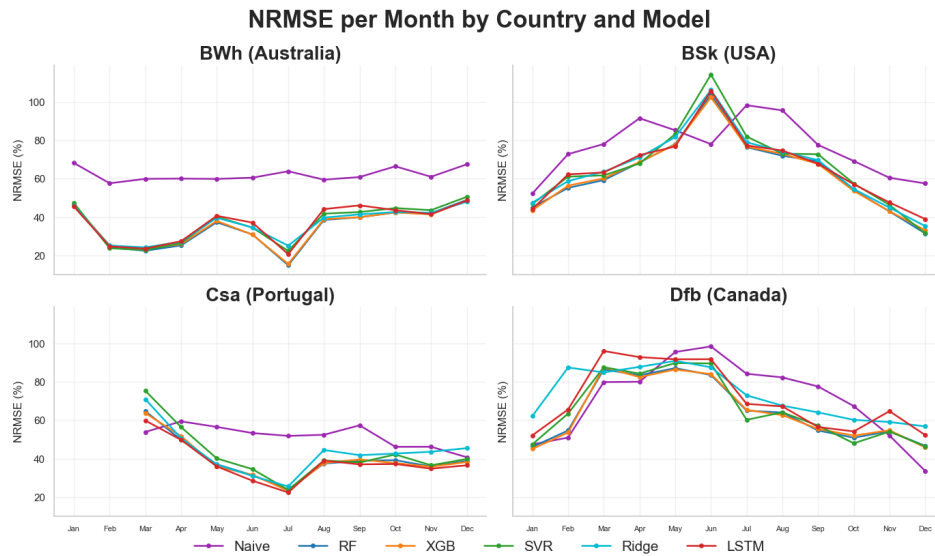


Figure 42: nRMSE per month for different climate zones.

In addition, two tables were created to summarise the best and worst monthly nRMSE across all climate zones for each model, where the cell colour indicates the climate zone in which the prediction was made. Together, these tables provide an overview of both model performance and climate zone characteristics, making it possible to identify which models and climate zones consistently achieve the lowest or highest errors across different times of the year. See Tables 12 and 13.

Table 12: Best nRMSE (%) per month and model. Cell colour indicates climate zone: BWh (Australia) , BSk (USA) , Csa (Portugal) , Dfb(Canada) .

Month	Naive	RF	XGB	SVR	Ridge	LSTM
Jan	47.4	45.0	43.5	46.8	45.8	44.2
Feb	50.9	24.1	25.3	23.8	25.1	24.7
Mar	53.8	22.4	23.4	22.8	24.2	23.5
Apr	59.3	25.2	25.8	26.4	27.1	27.4
May	56.4	37.0	35.8	40.1	36.9	35.9
Jun	53.3	30.9	30.8	34.5	30.9	28.4
Jul	51.8	14.9	15.6	22.4	25.1	20.6
Aug	52.4	37.4	37.7	39.0	39.7	38.9
Sep	57.3	38.8	39.5	37.9	41.4	37.0
Oct	46.1	39.1	37.6	42.1	42.6	37.2
Nov	46.1	35.8	35.8	36.5	42.2	34.7
Dec	33.5	31.5	33.1	31.5	35.3	36.5

Table 13: Worst nRMSE (%) per month and model. Cell colour indicates climate zone: BWh (Australia) , BSk (USA) , Csa (Portugal) , Dfb(Canada) .

Month	Naive	RF	XGB	SVR	Ridge	LSTM
Jan	68.1	46.7	46.4	47.4	62.1	52.1
Feb	72.7	55.2	56.3	63.4	87.4	65.5
Mar	79.8	86.4	87.5	87.5	84.8	96.0
Apr	91.4	83.3	82.4	84.2	87.7	92.8
May	95.4	87.0	86.3	89.7	90.8	91.7
Jun	98.3	103.7	102.2	114.1	106.1	105.7
Jul	98.2	76.3	76.6	81.9	78.9	77.2
Aug	95.5	71.9	72.9	73.0	74.1	74.6
Sep	77.5	69.0	67.6	72.7	69.6	67.7
Oct	69.1	54.0	53.6	57.4	60.1	57.0
Nov	61.0	54.2	54.7	54.0	59.0	64.7
Dec	67.4	48.9	48.5	50.5	56.7	52.2

A discussion about each climate zone follows below:

- **BWh (Australia):** As seen in Figure 42, all models consistently perform significantly better than the Naïve Persistence baseline. This provides strong incentive for the implementation of forecasting models in this climate zone, as even the simplest model - Ridge Regression - achieved considerably lower nRMSE values across all months, suggesting that meaningful improvements over a baseline forecast can be attained with relatively little modelling effort. Notably, the Australian dataset is featured in both Table 12 and 13, making it both a best-performing and worst-performing climate for certain models and months. The worst performances are observed during the Australian summer months (November-January), which may seem counterintuitive given that solar irradiance is at its peak during this period. However, these months also coincide with the wettest season in Alice Springs, highlighting the disruptive effect that cloud cover and precipitation can have on solar energy generation and, consequently, on forecast accuracy. The best performances are observed during the Australian fall and winter months, which are notably the driest and clearest months of the year in Alice Springs, with predominantly cloud-free skies [61]. This results in more stable and predictable solar irradiance patterns, which in turn facilitates more accurate forecasting across all models.
- **BSk (USA):** All models outperformed the Naïve Persistence baseline except during the month of June. It might seem unusual that the nRMSE exceeds 100% for most models during this month; however, this is possible since the nRMSE is normalised by the mean value of the test data. Consequently, an nRMSE above 100% indicates that the RMSE exceeds the mean value of the observed data, implying that the typical prediction error is larger than the average magnitude of the signal itself, which is clearly undesirable. Figure 42 shows a general trend for all models, where the nRMSE is lowest during the Northern hemisphere's winter months (December-January), and then steadily increases throughout the year until a peak is reached in the middle of summer. There could be several explanations to this, with a plausible one being the rapid changes in the PV data, as shown in Figure 35. The rainy period in Kersey lasts from March 17th to October 29th [62], which may indicate that frequent showers and more turbulent weather conditions contribute to the highly fluctuating nature of the load data during these months. Such rapid short-term variations can make the forecasting task more difficult, thereby contributing to the generally higher nRMSE values observed throughout this period. Although Table 13 appears to suggest that this climate zone only performs worst between June and October, this does not necessarily imply that the March-May period exhibits good predictive performance. Rather, it indicates that the Dfb climate zone happened to perform even worse during those particular months.
- **Csa (Portugal):** The most notable detail about Figure 42 might be that the nRMSE is missing for the

months of January and February. The reason is simple: the total timespan of the Portuguese data was only four years, making the test data - corresponding to 20% of the total data - span for less than 10 months. In fact, since the test data starts in the middle of March, the nRMSE should only be analysed from April onwards. As seen in Table 12, the Csa climate zone dominates by achieving the best forecasting performance across virtually all models and months. The only notable exceptions occur in June and July, where the Southern Hemisphere winter conditions yield comparatively stronger performance, as well as in December where the pattern is less consistent. Furthermore, all models substantially outperform the baseline during the summer months, though this margin narrows as the season transitions into late autumn and winter. This coincides with the seasonal patterns of Lisbon's weather; the months June to September mark the clearer, rainless months, while all other months are known as wetter and more overcast. [63] However, the models remain capable of capturing these overcast conditions, albeit with a reduced margin over the naïve baseline.

- **Dfb (Canada):** The Naïve Persistence model outperforms all other models during Calgary's snowy season, lasting from October 13th to May 15th [64], indicating that no model is able to fully capture the dynamics of snow cover and its impact on solar irradiance. Interestingly, however, the overall worst model performance occurs in the middle of the year, coinciding with Calgary's rainy season, which runs approximately from April to October [64]. Much like the patterns observed for the BSk climate zone, the erratic nature of precipitation events likely introduces day-to-day variability that the models struggle to predict. This is further reinforced by the absence of precipitation as a feature, leaving the models unable to account for the influence of rain, snow, and overcast conditions on solar irradiance.

The two superior climate zones in terms of forecasting performance are thus BWh and Csa, while BSk and Dfb yield the worst performance. Notably, both BWh and Csa are characterised by dry, sunny climates with little precipitation, whereas BSk and Dfb both experience snowy seasons and higher precipitation variability. This aligns well with the recurring observation throughout this analysis that drier, clearer seasons yield better forecasting performance than those characterised by precipitation and overcast conditions.

This pattern is further supported by the correlation structure of the feature sets across climate zones, as shown in the heatmaps provided in Appendix B. In the Australian and Portuguese datasets, the JRC-derived irradiance variables exhibit notably stronger correlations with the target variable k_W - reaching values of approximately 0.97 and 0.91 respectively - compared to the American and Canadian datasets, where the corresponding correlations are closer to 0.88. This suggests that solar irradiance is a more dominant and consistent driver of PV output in the drier climate zones, whereas the more variable climates introduce additional confounding factors - such as precipitation and cloud cover - that weaken the linear relationship between irradiance and power output. Importantly, since the JRC features are applied consistently across all datasets, the difference in correlation strength likely reflects genuine climatic characteristics rather than differences in data quality.

Overall, no forecasting model can be singled out as significantly better or worse than the others - except for the Naïve Persistence model of course. This suggests that, based on these analyses, there is no strong incentive to favour a more complex model - such as an LSTM - over a well-established alternative such as Random Forest, which achieves comparable performance with considerably less complexity. That said, Ridge Regression can likely be ruled out on practical grounds, as it consistently fails to constrain predictions to zero during nighttime hours, an unphysical behaviour not observed to the same extent in the remaining models.

6.6.2 Impact of Enhanced Resolution and Features

The inclusion of the Enhanced datasets allows for a direct comparison against the Standardised baselines established above, isolating the effects of increased resolution and additional features on model performance.

Starting with the Portuguese dataset, which perhaps yields the smallest difference in performance between the Enhanced and Standardised dataset. While the Enhanced version consistently performs better across all metrics and models, the improvement is marginal, as presented in Table 9. It should be noted that the Enhanced version has the same 1-hour resolution as its Standardised counterpart, meaning that the improvement solely is a product of the feature selection.

Conversely, the Enhanced American dataset has a finer temporal resolution than the Standardised version, while the selection of features remains the same. As already stated in Section 6.3, the mismatch between the high-frequency target variable and the lower-frequency features is a likely explanation why all models perform worse than the naïve baseline. This highlights the importance of using *features with the same original temporal resolution as the data*. Linear interpolation to achieve a higher temporal resolution of features is thus *not* a satisfactory approach in PV forecasting. It should be noted that there is no major difference in performance between the Enhanced and Standardised datasets in terms of scores across metrics and models; instead, the striking difference is that the Naïve Persistence benchmark improves substantially at the finer 15-minute resolution, such that all models now perform worse than this baseline - a contrast to the Standardised datasets where the models held a clear advantage over it.

The most prominent result from an Enhanced dataset is possibly the one from the Australian dataset. As seen in Table 7, Random Forest, SVR and XGBoost not only outperform the remaining models, but also show substantial improvement over their Standardised counterparts. To investigate this further, permutation importance was applied to the best performing model (Random Forest), yielding the following results:

Table 14: Permutation Importance: Random Forest, Enhanced Australian dataset.

Feature	Importance
Global_Horizontal_Radiation	0.3557
CSI	0.0903
solar_elevation	0.0537
ghi_clear	0.0433
tod_cos	0.0175
tod_sin	0.0062
Diffuse_Horizontal_Radiation	0.0039
doy_cos	0.0017
solar_azimuth	0.0012
Weather_Temperature_Celsius	0.0007
doy_sin	0.0002
Weather_Relative_Humidity	0.0002
WS10m	0.0001

As seen, `Global_Horizontal_Radiation` contributes disproportionately to model performance, suggesting that direct solar irradiance is the primary driver of the predicted output. This stands in contrast to the Standardised dataset, where the most prominent feature, `Gb(i)`, achieves a considerably lower importance score of 0.1442. Figure 43 shows the monthly correlation between the most prominent feature for each dataset and the power output kW. As seen, there is a near-perfect correlation between `Global_Horizontal_Radiation` and kW, which is less pronounced for `Gb(i)`, particularly during Alice Springs' wet season in December-January.

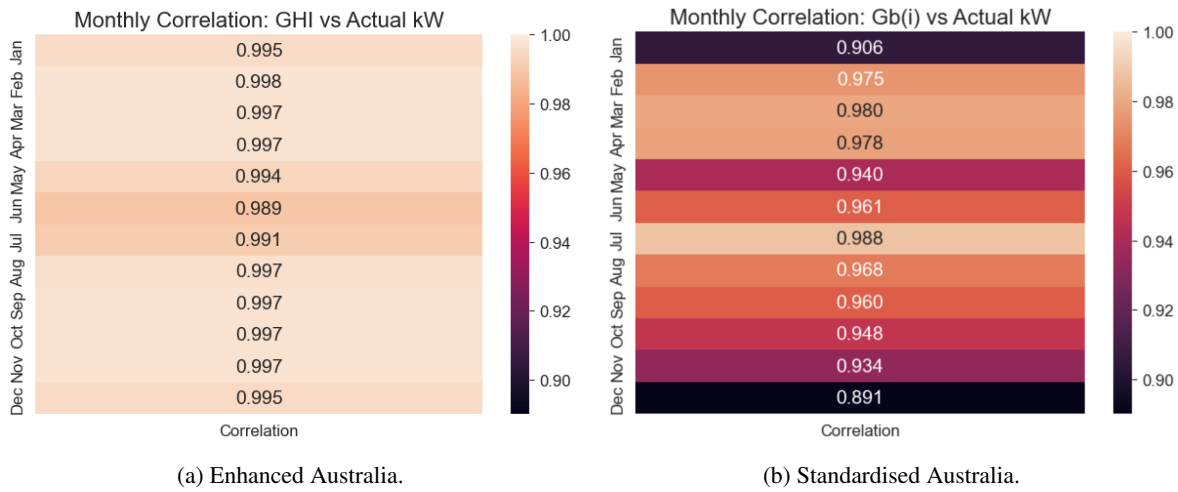


Figure 43: Correlation between kW and most prominent feature from permutation importance.

The dominance of `Global_Horizontal_Radiation` highlights the critical role of feature selection in PV forecasting. Unlike many of the remaining features, which are modelled estimates - such as those derived from the JRC - `Global_Horizontal_Radiation` is a locally measured quantity, which likely explains its near-perfect correlation with the output. In theory, a perfect forecast of GHI would yield an exceptionally accurate prediction of kW. This is, however, somewhat paradoxical: measuring GHI itself requires many of the same meteorological variables and modelled inputs that PV forecasting otherwise depends on - suggesting that the feature's predictive power comes at the cost of the very complexity it appears to circumvent.

7 Forecasting Performance Across Time Horizons

The purpose of RQ2 is to investigate how forecast accuracy varies across different prediction horizons. The biggest difference from the previous section is that the models now will use the selected features to predict the power output of a *future* timestamp instead of the *current* one. Moreover, the models will have access to historical values of the target variable kW , as would be the case in a real-world forecasting setting.

7.1 Exploratory Horizon Experiments

An initial set of experiments was conducted to explore how model performance varies across different prediction horizons. Two horizons were selected: 15 minutes and 24 hours ahead, representing short-term fluctuations and day-ahead planning respectively. These experiments were carried out on a single dataset (Enhanced Australian dataset), and while they were not scaled to all datasets due to time constraints, they provided valuable insights that informed the design of the subsequent Rolling Forecast Strategy.

Two input scenarios were evaluated for each horizon:

- **Scenario A:** Historical and current power consumption (kW), alongside deterministic contextual features - that is, variables known with certainty for any future time step.
- **Scenario B:** Scenario A extended with historical weather variables.

Beyond evaluating model performance across two contrasting prediction horizons, the comparison between Scenario A and Scenario B serves to quantify the added predictive value of historical weather variables.

7.1.1 Methodology

Only the two best-performing ML models for the Enhanced Australian dataset (Random Forest and XGBoost) were selected for this experiment. The LSTM model was also included to represent the deep learning category. While Random Forest and XGBoost previously outperformed LSTM in the Enhanced Australian dataset - as presented in Section 6.2 - this experiment provided an opportunity to compare all three models specifically across different prediction horizons. Additionally, a Persistence model was included as a naïve baseline.

Scenario A

The features in Scenario A were all deterministic features from the previously selected features. These are referred to as `calendar_features`:

- `tod_sin`
- `tod_cos`
- `doy_sin`
- `doy_cos`
- `solar_elevation`
- `solar_azimuth`
- `ghi_clear`

Unlike LSTM, Random Forest and XGBoost do not have the natural ability to look at recent timestamps - therefore, manual lags had to be created. This means that previous values were added as additional features for the models to look at. However, adding features comes with an important trade-off: while previous timestamps can help models capture short-term temporal patterns, an excessive number of lag features risks increasing model complexity and introducing noise, potentially degrading predictive performance.

Since the whole point of `calendar_features` is that they are deterministic, adding manual lags would be redundant. Therefore, manual lags were only added to kW , where the models had access to the most recent 3

hours of power output to make a prediction. For the 24 hour prediction horizon, manual lags were also created to capture kW 23, 24, 25 and 48 hours back, providing information about long-term patterns.

Table 15: Input and output features for the Exploratory Horizon Experiments (Scenario A).

Horizon	Input Features	Output
15 minutes	$kW[t], \dots, kW[t-11]$, tod_sin, tod_cos, doy_sin, doy_cos, solar_elevation, solar_azimuth, ghi_clear	$kW[t+1]$
24 hours	$kW[t], \dots, kW[t-11]$, $kW[t-92]$, $kW[t-96]$, $kW[t-100]$, $kW[t-192]$, tod_sin, tod_cos, doy_sin, doy_cos, solar_elevation, solar_azimuth, ghi_clear	$kW[t+96]$

As the LSTM model handles lagged features naturally in its sequences, no major adjustments were made. The sequence lengths were set to match the lagged features in the corresponding ML models, meaning:

- `sequence_length_15min = 12`
- `sequence_length_24h = 192`

It could be argued that it is unfair that the LSTM model gets access to all of the 192 past timesteps at a time, while the ML models only can see a selected few. However, this reflects a fundamental difference in how the models process temporal information: rather than receiving the full sequence, the ML models are provided with hand-selected lag features intended to capture key historical patterns, partially compensating for their lack of sequential awareness.

As manual lags and sequences were created, the same caution had to be exercised as discussed in Section 6.1.3, ensuring that no non-consecutive temporal patterns were introduced. Therefore, the manual lags/sequences were created first, and NaNs were dropped after. The Enhanced Australian dataset contained no missing timestamps, so this was not an issue.

After the historical features were properly introduced, the data was split. The split ratio was 80% training data and 20% test data for the ML models, while it was 64% training data, 16% validation data and 20% test data for the LSTM model. The data was then scaled using `StandardScaler()` in the same way as before.

Once the datasets were ready, they were optimised using Optuna, with similar hyperparameter bounds as before and 75 trials each. The models were then trained using the hyperparameters from the best respective trial, and predictions were subsequently generated.

The baseline model used for the 15 minute horizon was the same as before (Naïve Persistence), while the baseline model for the 24 hour horizon was changed to the Seasonal Persistence model, as described by Equation 3. Figure 44 illustrates three days of the Seasonal Persistence model.

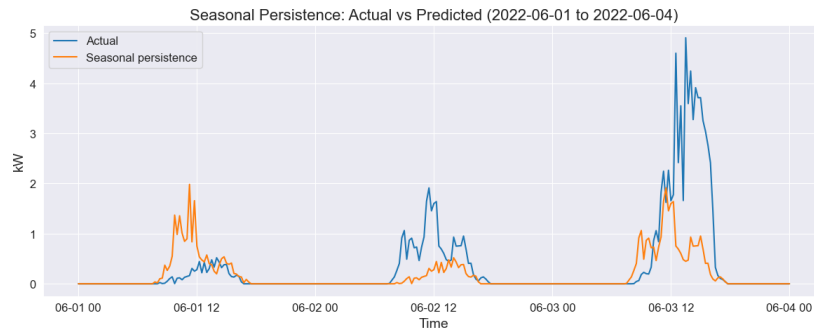


Figure 44: Seasonal Persistence, Enhanced Australian dataset.

Scenario B

Scenario B was implemented in a very similar way to Scenario A. The major difference was the introduction of the remaining features, referred to as `weather_features`:

- `Global_Horizontal_Radiation`
- `Diffuse_Horizontal_Radiation`
- `CSI`
- `WS10m`
- `Weather_Temperature_Celsius`
- `Weather_Relative_Humidity`

As these features are non-deterministic, it would be preferable to have manual lags added for them as well. However, if 12 lags would be created for each feature from `kW` and `weather_features`, $12 \cdot 7 = 84$ new features would have been created. This was deemed excessive, and a compromise was made: the 12 most recent lags of `kW` were retained as in Scenario A, while only 4 lags were added for each `weather_feature`, capturing the evolution over the past hour. Like before, manual lags capturing `kW` 23, 24, 25 and 48 hours prior were also added. Table 16 shows the selected features in both time horizons.

Table 16: Input and output features for the Exploratory Horizon Experiments (Scenario B). *GHI*: Global Horizontal Radiation, *DHI*: Diffuse Horizontal Radiation, *CSI*: Clear Sky Index, *WS*: Wind Speed (10m), *T*: Temperature, *RH*: Relative Humidity.

Horizon	Input Features	Output
15 minutes	<code>tod.sin</code> , <code>tod.cos</code> , <code>doy.sin</code> , <code>doy.cos</code> , <code>solar.elevation</code> , <code>solar.azimuth</code> , <code>ghi.clear</code> , <code>kW[t]</code> , ..., <code>kW[t-11]</code> , <code>GHI[t]</code> , ..., <code>GHI[t-3]</code> , <code>DHI[t]</code> , ..., <code>DHI[t-3]</code> , <code>CSI[t]</code> , ..., <code>CSI[t-3]</code> , <code>WS[t]</code> , ..., <code>WS[t-3]</code> , <code>T[t]</code> , ..., <code>T[t-3]</code> , <code>RH[t]</code> , ..., <code>RH[t-3]</code>	<code>kW[t+1]</code>
24 hours	<code>tod.sin</code> , <code>tod.cos</code> , <code>doy.sin</code> , <code>doy.cos</code> , <code>solar.elevation</code> , <code>solar.azimuth</code> , <code>ghi.clear</code> , <code>kW[t]</code> , ..., <code>kW[t-11]</code> , <code>kW[t-92]</code> , <code>kW[t-96]</code> , <code>kW[t-100]</code> , <code>kW[t-192]</code> , <code>GHI[t]</code> , ..., <code>GHI[t-3]</code> , <code>DHI[t]</code> , ..., <code>DHI[t-3]</code> , <code>CSI[t]</code> , ..., <code>CSI[t-3]</code> , <code>WS[t]</code> , ..., <code>WS[t-3]</code> , <code>T[t]</code> , ..., <code>T[t-3]</code> , <code>RH[t]</code> , ..., <code>RH[t-3]</code>	<code>kW[t+96]</code>

The sequence lengths for LSTM was the same as before, and the rest of the procedure was exactly the same as described in Scenario A.

7.1.2 15-Minute Horizon

Table 17 presents the forecasting performance of all models across Scenario A and Scenario B with a 15-minute horizon. As shown, the inclusion of historical weather variables consistently improves performance across *all models and metrics*, highlighting the value of incorporating contextual features for short-term solar forecasting.

A quite interesting result is that the LSTM model now performs on par with RF and XGBoost, as opposed to before, where it notably underperformed (see Section 6.2). In fact, when comparing Table 17 to Table 7, the LSTM model maintains similar performance, whereas RF and XGBoost perform considerably worse - converging to the level of LSTM. This may suggest that RF and XGBoost are particularly well-suited for nowcasting tasks, where the target variable is predicted at the same timestep as the input features. When the task shifts to genuine forecasting - predicting a future value from historical observations - their advantage

Table 17: Model performance comparison for the 15-minute horizon across Scenario A and Scenario B. Bold indicates the best-performing model and scenario for each metric; red shading indicates performance worse than the Naïve Persistence baseline.

Model	Scenario A				Scenario B			
	RMSE	MAE	sMAPE	nRMSE	RMSE	MAE	sMAPE	nRMSE
Naïve Persistence	0.406	0.168	18.613	33.827	0.406	0.168	18.613	33.827
Random Forest	0.309	0.104	14.998	25.839	0.302	0.104	14.935	25.240
XGBoost	0.313	0.117	16.810	26.212	0.300	0.106	15.210	25.083
LSTM	0.314	0.124	17.561	26.232	0.302	0.118	17.269	25.245

diminishes. If so, the strong results observed in Section 6.2 could be somewhat misleading, as nowcasting rarely reflects real-world forecasting conditions.

Figure 45 shows four days plotted for the LSTM model. While both scenarios are able to predict clear days quite well, it is evident that Scenario B allows for slightly more accurate predictions of overcast days.

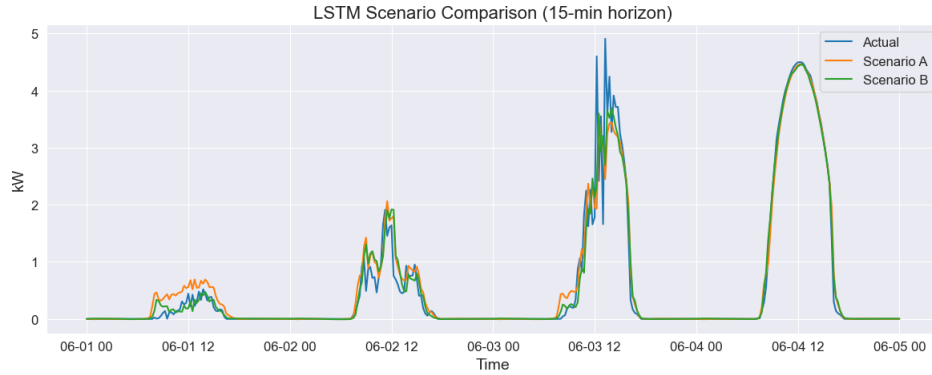


Figure 45: Scenario A vs Scenario B for the LSTM model with a 15-minute horizon.

7.1.3 24-Hour Horizon

Table 18 presents the forecasting performance of all models across Scenario A and Scenario B with a 24-hour horizon. Interestingly, the inclusion of historical weather variables does have the same effect as it did with the 15-minute horizon - in fact, the performance of Scenario B is often slightly worse than Scenario A here. Furthermore, all models consistently perform worse than the Seasonal Persistence baseline across the MAE and sMAPE metrics.

Figure 46 shows five days plotted for Scenario B. As seen, the 24-hour horizon models largely converge towards predicting the general diurnal PV output profile of a clear day, struggling to capture day-to-day variability. There are however instances where the models appear to capture more nuanced behaviour. One such example is visible in the figure, where following a clear day (2022-06-04), the models predict the subsequent day (2022-06-05) with a notably higher peak - closely matching the characteristic curve of a clear-sky day - suggesting that the models do pick up on the persistence of clear-sky conditions to some extent.

Table 18: Model performance comparison for the 15-minute horizon across Scenario A and Scenario B. Bold indicates the best-performing model and scenario for each metric; red shading indicates performance worse than the Naïve Persistence baseline.

Model	Scenario A				Scenario B			
	RMSE	MAE	sMAPE	nRMSE	RMSE	MAE	sMAPE	nRMSE
Seasonal Persistence	0.657	0.227	26.322	54.769	0.657	0.227	26.322	54.769
Random Forest	0.581	0.250	27.397	48.407	0.583	0.247	27.193	48.566
XGBoost	0.579	0.259	28.568	48.264	0.579	0.260	28.523	48.216
LSTM	0.583	0.253	28.974	48.577	0.585	0.268	30.180	48.711

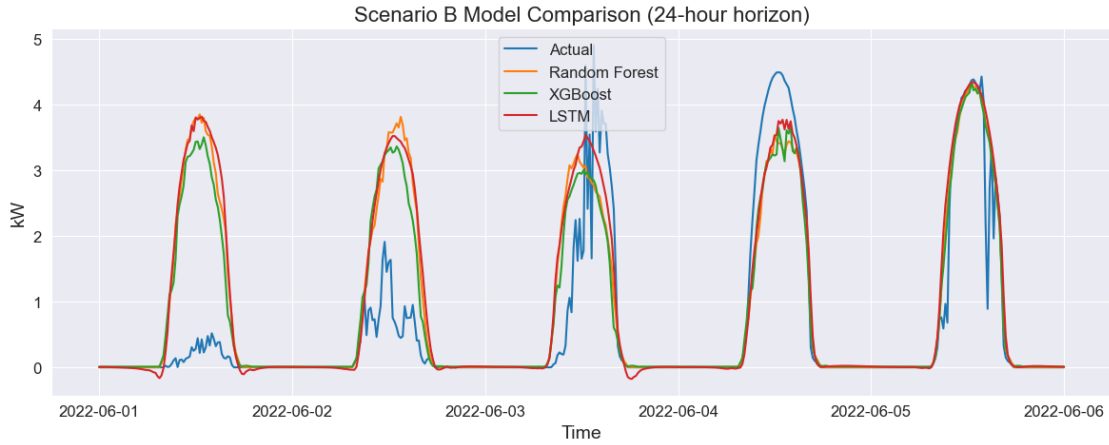


Figure 46: Scenario B for the 24-hour horizon.

7.2 Rolling Forecast Strategy

Building on the insights from the Exploratory Horizon Experiments, this section presents the primary forecasting framework used in this thesis. To ensure the forecasting framework remains relevant in a real-world energy trading context, the selected time horizons were designed with the day-ahead and intraday electricity markets in mind.

- **Day-ahead market:** The day-ahead market opens at 10:00 and closes at 12:00. The assumption in this study is that the bidding takes place at 11:00. With this in mind, the prediction horizon needs to cover 13-37 hours into the future, in order to bid on spot prices for 00:00-23:45 the next day.
- **Intraday market:** The continuous market opens 15:20 the previous day and is open until one hour before delivery. The assumption here is that all biddings on the intraday market take place on the same day as delivery, and at a maximum 6 hours in advance. This may seem arbitrary, but it reflects a balance between capturing short-term price dynamics and allowing for slightly longer temporal patterns to be considered. Furthermore, it is also assumed that all biddings take place during daytime, as PV output is zero during nighttime.

The Exploratory Horizon Experiments showed that historical weather data improves forecast accuracy for *short* prediction horizons, but little to no additional benefit for longer horizons. Moreover, while the 15-minute horizon models were able to capture fine-grained temporal patterns, the 24-horizon largely converged towards predicting the general diurnal PV output profile of a clear day.

The purpose of the Rolling Forecast Strategy is to output vectors of PV predictions for a given delivery times-

tamp. As forecast accuracy typically varies with time horizon, the predicted kW at 14:00 on a given day may differ depending on whether the forecast was made at 11:00 the previous day or at 13:00 on the same day. With this in mind, the LSTM model was selected as the most appropriate architecture, given its ability to produce multi-step ahead predictions in a single forward pass - naturally lending itself to the vector output required by the Rolling Forecast Strategy. Besides, the Exploratory Horizon Experiments showed that the LSTM performed comparably to the ML models across both prediction horizons, strengthening the justification for using LSTM in the Rolling Forecast Strategy.

Only the Enhanced datasets were used for this part of the study. The reason is that a higher temporal resolution is preferable when available, as $MTU = 15$ minutes. Furthermore, unlike the earlier part of the study where the Standardised datasets were used to enable cross-climate comparisons, the purpose here was not to compare performance across datasets, but rather to evaluate how accurately each prediction horizon could be forecasted given the best possible input data.

7.2.1 Methodology

Two different types of predictions were made: *day-ahead forecasting* and *intraday forecasting*. These are based on the time horizons discussed above.

Day-Ahead Forecasting

With the same logic as the previous implementations of LSTM, the sequences were created in the following way:

- First, all missing timestamps were imputed.
- Next sequences were created using `sequence_length = 192` and `selected_features + kW` as input. The target was a vector of kW values spanning $kW[t+1], \dots, kW[t+147]$, corresponding to predictions from 15 minutes to 36 hours and 45 minutes into the future.
- Finally, all sequences containing NaNs were skipped.

Once the sequences were created, all rows were dropped except for the ones corresponding to $t = 11:00$. This was done to significantly reduce computational costs, as the model would only need to be trained on around 1% of the original data. A trade-off of doing so is of course that fewer data points means less accurate predictions. In hindsight, a more robust approach would have been to train the model on the full training data before filtering to 11:00 at inference time - however, this was not feasible within the time constraints of the project.

The data was split into a training, validation and test subset, and then scaled using `StandardScaler()`. When creating the objective function for Optuna, a new metric needed to be defined; as the output was now a vector of predictions, the RMSE was computed as an unweighted average across all forecast horizons by flattening both the targets and predictions. This means that errors at 15 minutes ahead and 37 hours ahead are penalised equally. An alternative approach would have been to assign higher weights to shorter horizons, which could improve short-term accuracy at the cost of longer-term performance. However, this was considered a deliberate design choice - introducing such weights would bias the optimisation towards certain horizons, which was deemed undesirable given the scope of this study.

Each dataset was optimised with 20 trials using Optuna, and the models were then trained using the optimal hyperparameters. As before, the predictions were then generated.

Intraday Forecasting

The Rolling Forecast for intraday forecasting was done in the exact same way as described above, apart from two main differences:

- The horizon length of the output vector was changed from 147 to 24, corresponding to 6 hours.
- Rather than filtering to a single timestamp as with 11:00, the data was filtered to all timestamps between 06:00 and 18:00, corresponding to daytime hours.

7.2.2 Day-Ahead Forecasting

Figure 47 shows the average RMSE per timestep for each climate zone. A couple of observations can be made consistently across all climate zones. Firstly, the RMSE follows a distinct diurnal pattern, reflecting the solar generation cycle. Notably, the first peak in RMSE is lower than the second, suggesting that forecast error is smaller for the same day than for the following day, indicating that time horizon does play a meaningful role even within a 37-hour window. Secondly, short-term forecast accuracy is surprisingly poor; unlike the Exploratory Horizon Experiments where 15-minute forecasts performed excellently, the 15-minute predictions here are considerably worse. A possible explanation to this is the design choice of the objective function, where all time horizons are penalised equally. If the objective function would have used a weighted RMSE instead, short-term forecasting performance might have benefited. Following this theory, the reason why the 15-minute horizon performed so well in the Exploratory Horizon Experiments would be that Optuna only needed to optimize a single point at a time - that is, the model was trained and tuned exclusively for that specific horizon, allowing it to dedicate its full capacity to minimizing error at exactly 15 minutes. This highlights how design choices have substantial implications for performance across different time horizons, and should be carefully considered when developing forecasting systems intended for specific operational use cases.

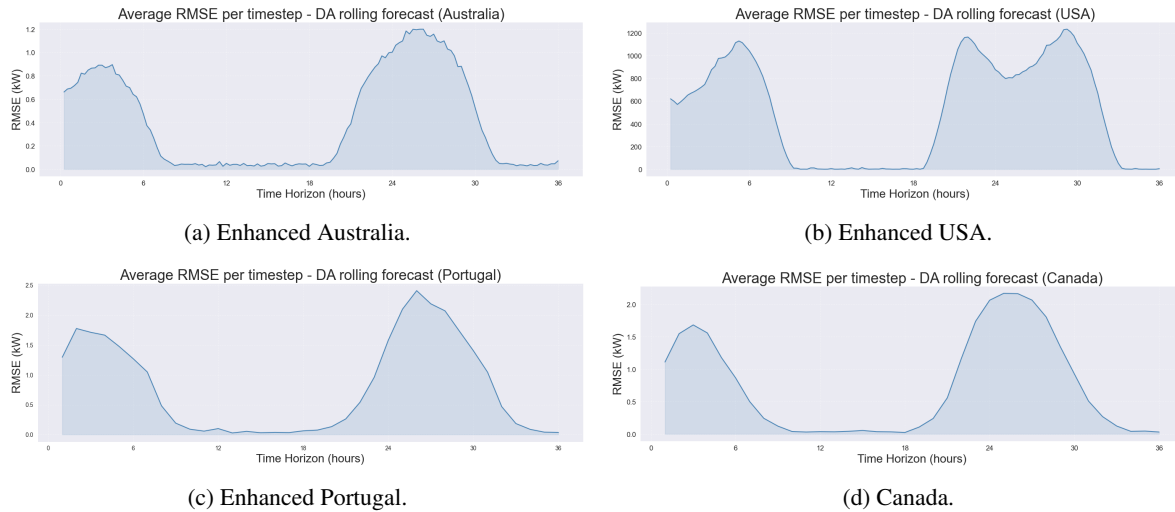


Figure 47: Average RMSE per timestep for day-ahead Rolling Forecast, where the x-axis starts at 11:00 o'clock.

The average RMSE was also compared to the RMSE for an arbitrary choice of two consecutive clear and overcast days, as shown in Figure 48 where Enhanced Australia is plotted. Notably, clear days exhibit lower forecast error than the all-day average, while overcast days exhibit higher error. This illustrates that weather plays an important role in day-ahead forecasting, where clear-sky conditions is more predictable due to its smooth diurnal profile, whereas overcast conditions introduce greater variability driven by cloud intermittency.

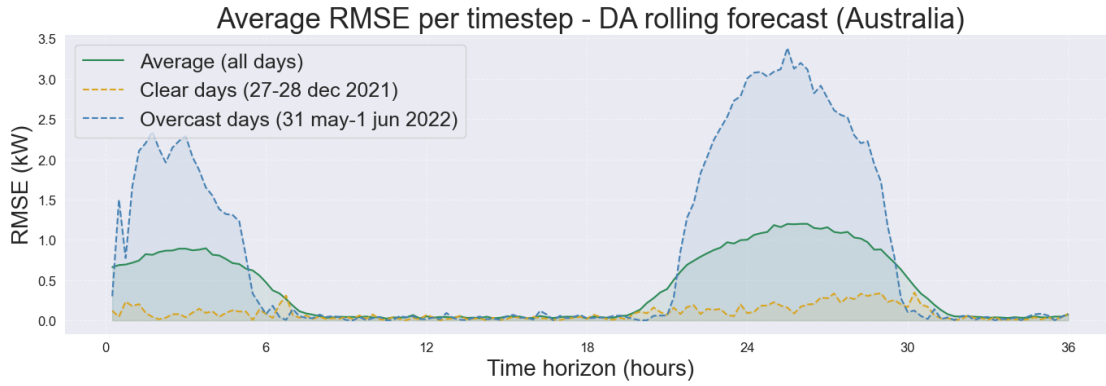


Figure 48: Average RMSE per timestep for day-ahead Rolling Forecast, including clear and overcast days.

7.2.3 Intraday Forecasting

Similarly as before, the average RMSE per timestep was plotted, but this time a subplot was created for each hour the test sequences were initialised at. Figure 49 shows this for the Enhanced Australian dataset, although a similar plot was created for each climate zone, and all climate zones exhibited resembling patterns in their corresponding plots.

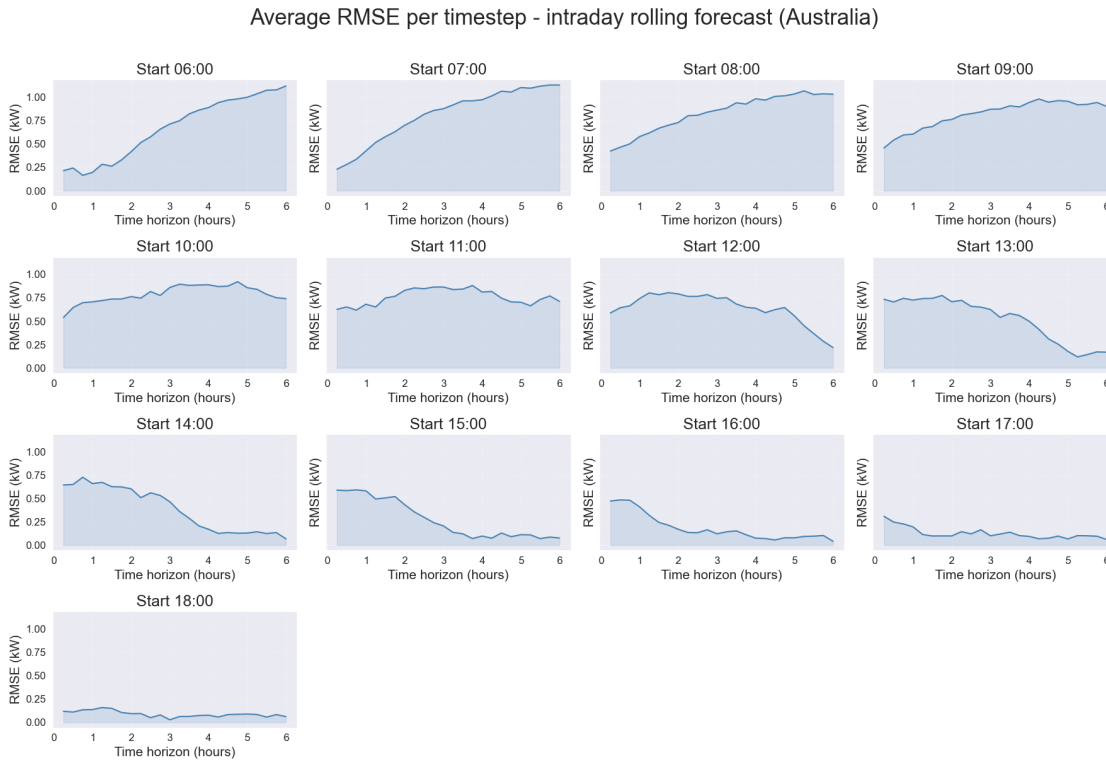


Figure 49: Average RMSE per forecast timestep for the intraday 6-hour Rolling Forecast, stratified by start time.

It might seem like the time horizon doesn't affect the forecasting accuracy at all, and that the error solely

follows the diurnal profile of a clear day. While this is partially true, it is still possible to see that time horizon affects output. For example, take start time 6:00 in Figure 49; the 5-hour RMSE - corresponding to 11:00 - is approximately 1.00 kW. However, at start time 10:00, the 1-hour RMSE - still corresponding to 11:00 - is short of 0.75 kW instead, indicating that error decreases with shorter prediction horizons. However, the driving factor here is undeniably time of day, making the RMSE very low indeed for timesteps corresponding to evening time. Furthermore, the intraday Rolling Forecast has the same problem as the DA Rolling Forecast, where very short prediction horizons (such as 15 minutes) seemingly underperforms. This is likely a consequence of the same objective function design choice discussed previously.

7.2.4 Remarks on Forecast Performance

In general, both the day-ahead and intraday Rolling Forecast Strategies yield disappointing results. The RMSE follows the time-of-day pattern more than anything else, rendering the forecasts somewhat uninformative. This is, however, not surprising: the models have no information about future weather conditions, and in a real-world deployment they would naturally be supplemented with weather predictions to guide the forecast.

Obtaining historical weather forecast data is, however, non-trivial. Beyond availability, such data introduces an additional complication: forecast accuracy itself degrades with time horizon, meaning that a 37-hour weather forecast is inherently less reliable than a 1-hour forecast [65]. Incorporating this effect realistically would require horizon-dependent forecast uncertainty, adding considerable complexity to the pipeline.

To sidestep this issue while still quantifying the potential benefit of weather information, the models are instead evaluated under a *perfect forecast* assumption - that is, future weather features are provided as inputs at inference time, as if a flawless weather prediction were available. This serves as an upper bound on the performance gain that weather information could provide, and motivates the experiments presented in the following section.

7.3 Rolling Forecast Strategy with Perfect Weather Information

Due to time constraints, the following analysis was conducted only using the Enhanced Australian dataset.

7.3.1 Methodology

To train forecasting models with perfect future weather information, the input sequences were constructed with two distinct components: a backward-looking segment containing `selected_features` alongside power consumption (kW) from 48 hours prior up to time t , and a forward-looking segment containing only `selected_features` from $t + 1$ up to $t + h$, where h denotes the forecast horizon. Notably, the model can *not* distinguish between the two parts - instead, all it can see is a long sequence, where kW=0 after a certain point.

In general, the pipeline of both the DA forecasting and the intraday forecasting was essentially the same as described in the previous section. The only other discrepancy, apart from the creation of sequences, was that the data was scaled using two separate `StandardScaler()` - one fitted on the historical part of the sequences (all features including kW) and one fitted on the future part (excluding kW) - both fitted exclusively on training data to prevent data leakage.

The predictions were initially made using the LSTM model. However, the results yielded unpromising, as they in fact were slightly worse than the results presented in Section 7.2. Following an ablation test - where the future values were removed completely and the model re-trained, but using the same Optuna-optimised hyperparameters as before - it was shown that the future values seemingly had no impact on model performance at all, indicating that they were not being used. This is not surprising given the inherent architecture of LSTMs, where sequences are processed strictly left-to-right, meaning that by the time the model reaches the future weather features, its internal representation has already been formed on the basis of historical data alone. Exploiting future information therefore requires an architecture capable of processing the sequence in both directions, motivating the adoption of a Bi-LSTM, which through its backward pass has already observed the future weather values before processing the historical boundary.

However, switching model to the Bi-LSTM raises a concern - if the Bi-LSTM would outperform the LSTM, it would be difficult to ascertain whether the improvement was due to future weather information, or simply because of the switch of model. Therefore, the Bi-LSTM was initially trained using only historical features, with the exact same pipeline as described in Section 7.2. The result was very similar as before, meaning that the Bi-LSTM model and the LSTM model both perform comparatively when it comes to making predictions using solely past features. This allowed for the use of Bi-LSTM moving forward, as any forecasting improvements would be attributable to the future weather features, rather than the model architecture itself.

7.3.2 Day-Ahead Forecasting

Figure 50 shows the average RMSE per timestep for day-ahead forecasting. When comparing to Figure 47a, it can be seen that the RMSE is lower with perfect weather information compared to without; while the peaks previously were around 0.9 kW and 1.2 kW, they are now roughly 0.7 kW and 0.6 kW instead. Interestingly, the two peaks are now more similar in magnitude than in the baseline model, with the second peak being slightly lower than the first. This lacks an obvious explanation, and may simply be a coincidental result.

What is less surprising, however, is that the two peaks are more similar overall; with perfect weather information, forecast uncertainty should in principle be uniformly distributed across the horizon, unlike real-world forecasting where weather forecast uncertainty compounds with temporal distance, naturally producing a higher second peak. However, even with perfect weather information, one would still expect very short-term predictions (e.g. 15 minutes) to be better, as they should benefit from the recent kW history as a strong short-term signal. The fact that this is not observed may suggest that the model does not fully exploit the kW feature for near-term prediction, though this could equally be an artefact of the unweighted RMSE objective discussed previously.

Regardless, the model continues to follow the diurnal pattern, with lower RMSE during early morning and evening hours. This result is unsurprising, as low production during these periods is structurally determined by the solar cycle rather than by weather conditions, making it inherently easier to predict.

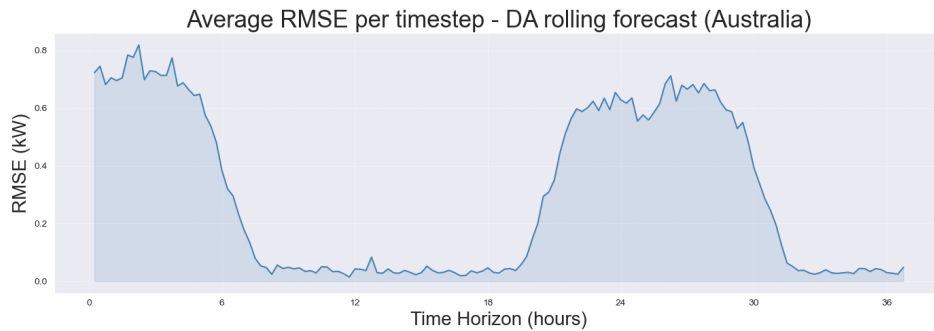


Figure 50: Average RMSE per timestep for day-ahead Rolling Forecast with perfect weather information.

7.3.3 Intraday Forecasting

Future weather information had a clear impact on the intraday forecasting. Figure 51 shows the predictions of an overcast day, where the results from Section 7.2 and 7.3 are compared side-by-side. As seen, the model is able to capture the behaviour of kW much better when future weather information is available.

Furthermore, the average RMSE per timestep was once again calculated and plotted as seen in Figure 52. Whereas the maximum RMSE previously was around 1.1 kW (see Figure 49), the maximum was now 0.6 kW, representing a meaningful improvement. However, a similar pattern to that observed in the day-ahead case emerges. Forecast error appears largely invariant across the prediction horizon, reflecting only the diurnal case.

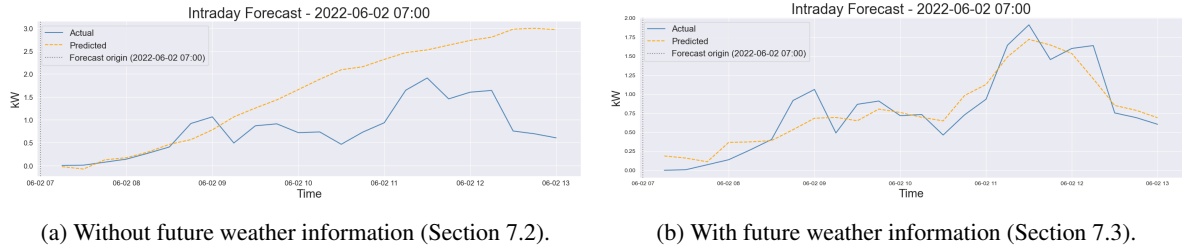


Figure 51: Intraday predictions for a given timestep using pipeline from Section 7.2 and 7.3.

pattern - a result that is consistent with the uniformly distributed forecast uncertainty that perfect weather information provides. Furthermore, very short-term predictions remain unexpectedly poor, again suggesting that the model does not fully leverage the recent kW history as a short-term signal, or alternatively that this is an artefact of the unweighted RMSE objective, as discussed previously.

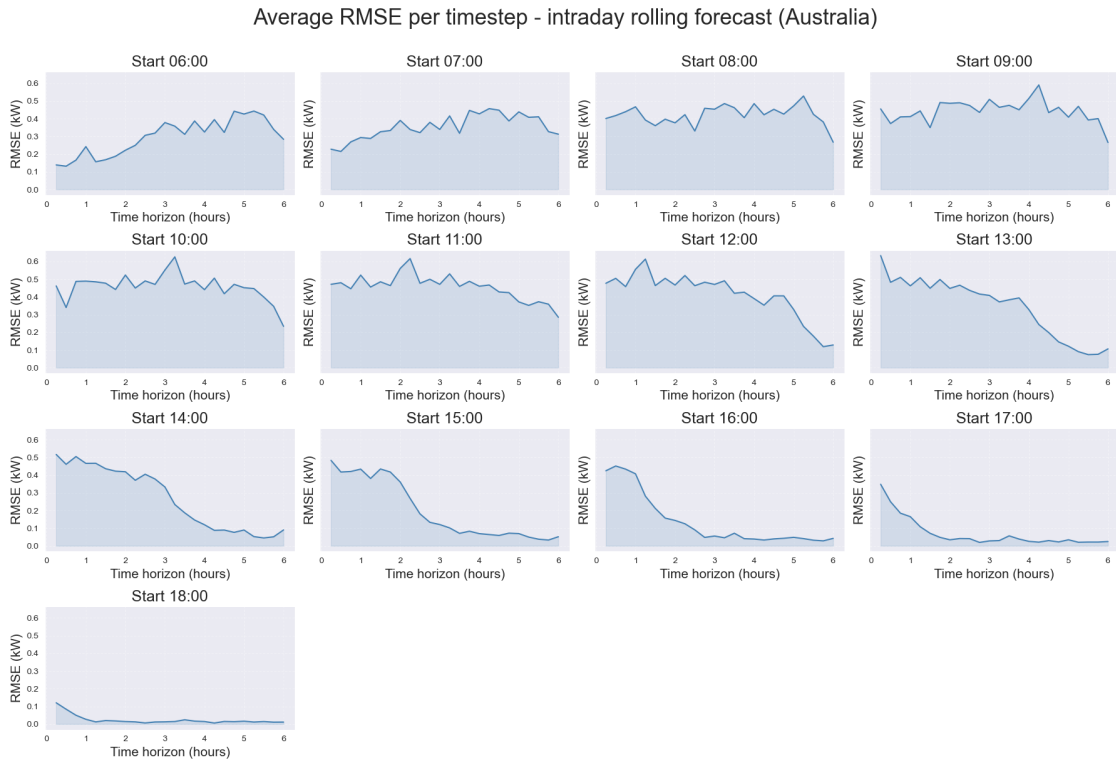


Figure 52: Average RMSE per forecast timestep for the intraday 6-hour Rolling Forecast with perfect weather information, stratified by start time.

All in all, the Rolling Forecast Strategy with Perfect Weather Information provides insight that future weather information makes considerable improvements to PV forecasting, but does not give clear indications that forecasting accuracy is affected by the time horizon rather than simply the time of day. Perhaps if real weather forecasts were used - which are well-known to degrade in accuracy with increasing temporal distance - the outcome would have been different; however, this is of course merely speculation.

8 General Discussion

While RQ1 and RQ2 were investigated separately, the results suggest that the climate in which a PV system operates fundamentally shapes how forecast accuracy degrades as the prediction horizon extends. As noted, the Rolling Forecasts converged heavily towards the diurnal PV output profile of a clear day, especially when no future weather information was available. In climates such as BWh (Australia) and Csa (Portugal), where clear-sky conditions dominate and overcast days are relatively infrequent, this convergence behavior is largely self-correcting - the assumed clear-day profile is simply correct more often. In contrast, BSk (USA) and Dfb (Canada) exhibit greater meteorological variability, meaning that the absence of future weather information is more consequential; the model defaults to a clear-day assumption that is frequently wrong. This suggests that atmospheric variability is itself a fundamental driver of forecasting difficulty - one that operates independently of, and in addition to, the choice of model and forecast horizon.

Of course, in practice, PV forecasting systems rarely operate without any weather forecast input, since the inclusion of future weather information improves forecasting accuracy, as demonstrated in Section 7.3. However, the results from the cross-country comparison in Section 6.6.1 provide independent support for the climate-driven difficulty argument: even when models were evaluated at the shortest possible horizon $h = 0$ (nowcasting) with access to all available features, the sunnier climates (BWh, Csa) consistently yielded lower forecast errors than their more variable counterparts (BSk, Dfb). This suggests that the relative advantage of operating in a stable climate persists regardless of forecast horizon - not merely as an artifact of the Rolling Forecast's convergence behavior. It should be noted, however, that variables such as cloud cover and precipitation were absent from the feature set used in this study, and their inclusion could possibly benefit models in more variable climates disproportionately. That said, weather forecast accuracy itself degrades with increasing horizon length, meaning that the uncertainty compounding effect identified here would likely remain even with a richer set of meteorological inputs.

The differences in climate-driven predictability have tangible implications for how PV operators interact with electricity markets. For example, since the day-ahead market requires predictions with a ~ 12 -36 hour time horizon, it is harder to make accurate bids in more variable climates. However, the intraday market offers an opportunity to correct these deviations as new information becomes available - and this correction mechanism is arguably more valuable in climates like BSk and Dfb, where day-ahead uncertainty is inherently higher. Conversely, in more stable climates such as BWh and Csa, day-ahead forecasts are already sufficiently accurate, potentially reducing the need for intraday adjustments altogether. In other words, the return on investment in advanced forecasting infrastructure is likely to be considerably higher for operators in meteorologically challenging regions than for those in consistently sunny climates.

9 Conclusions

The purpose of this thesis was to examine how PV forecasting accuracy is affected by climate heterogeneity and prediction horizon length - two dimensions that are critical for assessing model generalisability and for informing participation in day-ahead and intraday electricity markets.

Addressing RQ1, forecast accuracy was found to vary substantially across climatic regions, while differences between models were comparatively modest. The sunnier climates, BWh and Csa, consistently yielded lower forecast errors, whereas the more variable climates, BSk and Dfb, produced considerably higher - a pattern that appeared correlated with the frequency of clear-sky conditions. Among the models evaluated, Naïve Persistence stood out as a clear outlier, and Ridge Regression exhibited a systematic weakness in failing to suppress output during nighttime hours. Otherwise, no consistent performance hierarchy emerged across the remaining models, suggesting that increased model complexity does not reliably translate into improved forecast accuracy across the climatic contexts studied here.

With respect to RQ2, forecast accuracy was found to deteriorate consistently with increasing prediction horizon, though the nature of this degradation varied across the three forecasting frameworks employed. The Exploratory Horizon Experiments revealed that short-term forecasts at a 15-minute horizon achieved substantially higher accuracy than those at 24 hours, and further showed that historical weather features meaningfully improved performance at short horizons but provided limited benefit at longer ones. The Rolling Forecast framework, while less suited to capturing short-horizon potential - possibly due to an unweighted objective function that did not prioritise near-term accuracy - also demonstrated that in the absence of future weather information, forecasts converged towards a generic diurnal PV output profile, effectively defaulting to a clear-day assumption. The inclusion of perfect future weather information partially mitigated this degradation, suggesting that meteorological uncertainty is a primary driver of horizon-dependent accuracy loss.

Beyond the research questions, a couple of additional findings emerged from the analysis, specifically regarding feature selection and temporal resolution. The availability of locally measured features in two of the datasets allowed for a more granular assessment of feature importance, revealing that the inclusion of certain variables - most notably locally measured `Global_Horizontal_Radiation` in the Enhanced Australian dataset - substantially improved model performance compared to relying on reanalysis data alone. However, it can be argued that this improvement is somewhat paradoxical, as locally measured GHI - despite its strong predictive power - is itself dependent on the same meteorological complexity that PV forecasting otherwise seeks to bypass. Furthermore, the Enhanced American dataset demonstrated the importance of including features at the same - or finer - temporal resolution as the target variable; linear interpolation of lower-frequency data proved insufficient for this type of analysis, though alternative resampling methods may yield better results.

In summary, this thesis has demonstrated that forecasting difficulty in PV systems is not solely a function of model choice or prediction horizon, but is fundamentally shaped by the climatic context in which a system operates. The results suggest that meteorological variability acts as an independent driver of forecast error - one that persists regardless of horizon length or model complexity. This has tangible implications for electricity market participation; while advanced forecasting infrastructure may offer limited marginal benefit in stable, sunny climates, the return on such investment is likely considerably higher in meteorologically variable regions where day-ahead uncertainty is inherently greater and intraday correction mechanisms become correspondingly more valuable. By evaluating multiple models across four climatically distinct sites, this thesis contributes to a more generalised understanding of PV forecasting performance. Rather than reaching site-specific conclusions, the insights are instead relevant across the diverse climatic conditions in which solar energy systems are increasingly being deployed.

10 Future Work

This thesis examines how forecasting accuracy varies across two different dimensions - climate and time horizon - where both dimensions essentially are examined completely separately from each other. However, as discussed under General Discussion in Section 8, the climate in which a PV system operates shapes how forecast accuracy degrades with an extending prediction. With this said, the *interplay* between climate and time horizon is something that should be further explored in future research, especially in a market relevant context as the relative value of forecast accuracy improvements may differ substantially between climates depending on the horizon at which predictions are made. For instance, it could be investigated whether intraday corrections offer greater benefit in variable climates than in stable ones, and to what extent this interplay influences bidding strategies in day-ahead versus intraday markets.

A limitation of the study is that only four climate zones are examined. Since the Köppen-Geiger climate classification comprises 25 distinct climate zones [36], future research should include a broader range of zones to more fully capture the diversity of climatic conditions under which PV systems operate. Furthermore, this study evaluated only five models beyond Naïve Persistence, and future research should explore a wider range of architectures. Importantly, while this thesis prioritised breadth of comparison over optimisation, subsequent work could focus on developing models specifically tuned for robustness - either across all climatic contexts or tailored to particular climate zones. Finally, the Rolling Forecast framework exhibited unexpectedly poor performance at short horizons, likely due to an unweighted objective function; future work should investigate alternative formulations that better accommodate multi-horizon outputs, particularly where short-term accuracy is of primary interest.

References

- [1] C. Ghenai, “Artificial intelligence based forecasting models for solar pv/hydrogen fuel cell power system: Role of short-term ai-based forecasting for the energy transition,” in *2024 International Conference on Sustainable Energy: Energy Transition and Net-Zero Climate Future (ICUE)*, pp. 1–7, 2024.
- [2] United Nations Development Programme (UNDP), “What is the sustainable energy transition and why is it key to tackling climate change?,” <https://climatepromise.undp.org/news-and-stories/what-sustainable-energy-transition-and-why-it-key-tackling-climate-change>, 2025. Accessed: 2026-01-27.
- [3] X. Huang, Q. Li, Y. Tai, Z. Chen, J. Liu, J. Shi, and W. Liu, “Time series forecasting for hourly photovoltaic power using conditional generative adversarial network and bi-lstm,” *Energy*, vol. 246, p. 123403, 2022.
- [4] N. Alagusundari, A. D. K. E. S. S. S. D. K., and V. R. B., “Ai-based predictive and optimal control for microgrid energy management: Design and implementation,” in *2025 International Conference on Modern Sustainable Systems (CMSS)*, pp. 222–228, 2025.
- [5] D. Yadav, N. Singh, V. S. Bhadoria, V. Vita, G. Fotis, E. G. Tsampasis, and T. I. Maris, “Analysis of the factors influencing the performance of single- and multi-diode pv solar modules,” *IEEE Access*, vol. 11, pp. 95507–95525, 2023.
- [6] L. Jovanovic, N. Bacanin, A. Petrovic, M. Zivkovic, M. Antonijevic, V. Gajic, M. M. Elsayed, and M. Abouhawwash, “Exploring artificial intelligence potential in solar energy production forecasting: Methodology based on modified pso optimized attention augmented recurrent networks,” *Sustainable Computing: Informatics and Systems*, vol. 47, p. 101174, 2025.
- [7] M. Terkes, A. Demirci, I. Dagal, E. Gokalp, and U. Cali, “Comparative analysis of ai-based regression models for solar power forecasting: Techniques, preprocessing, and evaluation,” *Energy Reports*, vol. 15, p. 109071, 2026.
- [8] L. Oumiguil and A. Nejmi, “Short-term pv plant power forecasting model using cnn-gru with xgboost algorithm,” *Scientific African*, vol. 31, p. e03184, 2026.
- [9] Žydrūnas Kavaliauskas, M. Milieška, G. Blažiūnas, G. Gecevičius, and H. Zhairabany, “Neural network approaches for generation forecasting and conceptual design in hybrid renewable energy systems,” *International Journal of Electrical Power Energy Systems*, vol. 172, p. 111241, 2025.
- [10] I. Bevanda *et al.*, “Assessing the impact of solar spectral variability on the performance of photovoltaic technologies across european climates,” *Energies*, vol. 18, no. 14, p. 3868, 2025.
- [11] Svenska kraftnät, “Dagen före-marknaden – fysisk handel med el,” 2025. Accessed: 2026-05-05.
- [12] Svenska kraftnät, “Intradagsmarknaden – justering av dagen före-handel,” 2025. Accessed: 2026-05-05.
- [13] H. F. Carlak, K. Karabanova, and P. Mishra, “Evaluation of machine learning algorithms for weather forecasting in two climatic zones for solar PV modeling,” *International Journal of Photoenergy*, vol. 2026, pp. 1–10, 2026.
- [14] P. Aduama, A. S. Al Sumaiti, and K. H. Al Hossani, “Cross-regional adaptation of deep learning architectures for PV power forecasting under climatic uncertainty,” in *2025 IEEE International Conference on Energy Technologies for Future Grids (ETFG)*, (Wollongong, Australia), IEEE, 2025.
- [15] R. Hasan, T. Siddique, M. S. Rahman, and M. A. Kabir, “Lstm-based deep learning model for solar irradiance prediction across diverse climate zones,” in *2025 IEEE Kansas Power and Energy Conference (KPEC)*, pp. 1–6, 2025.

- [16] M. Pierro, D. Gentili, F. R. Liolli, C. Cornaro, D. Moser, A. Betti, M. Moschella, E. Collino, D. Ronzio, and D. van der Meer, "Progress in regional pv power forecasting: A sensitivity analysis on the italian case study," *Renewable Energy*, vol. 189, pp. 983–996, 2022.
- [17] M. Perera, J. De Hoog, K. Bandara, and S. Halgamuge, "Multi-resolution, multi-horizon distributed solar pv power forecasting with forecast combinations," *Expert Systems with Applications*, vol. 205, p. 117690, 2022.
- [18] P. Herodotou, C. Konstantinou, A. Kyprianou, G. Tziolis, G. Makrides, P. Papanastasiou, and G. E. Georghiou, "Multi-horizon net load forecasting for residential households: evaluating statistical and deep learning models with recursive and direct strategies," *Energy Conversion and Management*, vol. 356, p. 121394, 2026.
- [19] K. Hayawi, H. Maliakkal, N. Venugopal, T. M. Hussain, and G. B. Rajagopalan, "Machine learning - driven solar forecasting in dust-prone regions for sustainable energy systems," *Solar Energy Advances*, vol. 5, p. 100108, 2025.
- [20] M. O. Alhassan, S. Gyamfi, B. Aboagye, F. A. Diawuo, and M. Kwarteng, "In-situ pv generation forecasting of utility-scale solar power plants in ghana: Single and hybrid machine learning approaches," *Solar Energy Advances*, vol. 5, p. 100124, 2025.
- [21] Florida Solar Energy Center (FSEC), "Cells, modules, panels and arrays." <https://energyresearch.ucf.edu/consumer/solar-technologies/solar-electricity-basics/cells-modules-panels-and-arrays/>, 2026. Accessed: 2026-03-19.
- [22] SolAC, "Hur fungerar solceller?." <https://www.solac.se/hur-fungerar-det>, 2026. Accessed: 2026-03-19.
- [23] U.S. Department of Energy, "Solar photovoltaic cell basics." <https://www.energy.gov/eere/solar/solar-photovoltaic-cell-basics>, 2026. Accessed: 2026-03-19.
- [24] C. Ghenai, "Artificial intelligence based forecasting models for solar pv/hydrogen fuel cell power system: Role of short-term ai-based forecasting for the energy transition," in *2024 International Conference on Sustainable Energy: Energy Transition and Net-Zero Climate Future (ICUE)*, pp. 1–7, 2024.
- [25] Alternative Energy Tutorials, "Standard test conditions." <https://www.alternative-energy-tutorials.com/photovoltaics/standard-test-conditions.html>, 2026. Accessed: 2026-03-19.
- [26] K. Nishioka, T. Hatayama, Y. Uraoka, T. Fuyuki, R. Hagihara, and M. Watanabe, "Field-test analysis of pv system output characteristics focusing on module temperature," *Solar Energy Materials and Solar Cells*, vol. 75, no. 3, pp. 665–671, 2003. PVSEC 12, PART III.
- [27] J. Lian, Y. Zhang, C. Ma, Y. Yang, and E. Chaima, "A review on recent sizing methodologies of hybrid renewable energy systems," *Energy Conversion and Management*, vol. 199, p. 112027, 2019.
- [28] U.S. Department of Energy, "Solar integration: Solar energy and storage basics." <https://www.energy.gov/eere/solar/solar-integration-solar-energy-and-storage-basics>, 2026. Accessed: 2026-03-20.
- [29] Svenska kraftnät, "Om elmarknaden," 2025. Accessed: 2026-05-05.
- [30] Nord Pool, "Day-ahead market," n.d. Accessed: 2026-03-31.
- [31] Nord Pool, "Transition to 15-minute market time unit (mtu)," 2025. Accessed: 2026-05-05.
- [32] Nord Pool, "Intraday market," n.d. Accessed: 2026-03-31.
- [33] World Meteorological Organization, "Climate." <https://wmo.int/topics/climate>, 2026. Accessed: 2026-03-20.

- [34] IPCC, Matthews, J.B.R. and V. Möller and R. van Diemen and J.S. Fuglestedt and V. Masson-Delmotte and C. Méndez and S. Semenov and A. Reisinger (eds.), “Annex vii: Glossary,” in *Climate Change 2021: The Physical Science Basis. Contribution of Working Group I to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change* (V. Masson-Delmotte, P. Zhai, A. Pirani, S. Connors, C. Péan, S. Berger, N. Caud, Y. Chen, L. Goldfarb, M. Gomis, M. Huang, K. Leitzell, E. Lonnoy, J. Matthews, T. Maycock, T. Waterfield, O. Yelekçi, R. Yu, and B. Zhou, eds.), pp. 2215–2256, Cambridge, United Kingdom and New York, NY, USA: Cambridge University Press, 2021.
- [35] National Oceanic and Atmospheric Administration (NOAA), “Climate zones — jetstream: Online weather school.” <https://www.noaa.gov/jetstream/global/climate-zones>, 2023. Accessed: 2026-03-20.
- [36] National Oceanic and Atmospheric Administration (NOAA), “Jetstream max: Addition köppen-geiger climate subdivisions.” <https://www.noaa.gov/jetstream/global/climate-zones/jetstream-max-addition-k-p-pen-geiger-climate-subdivisions>, 2023. Accessed: 2026-03-20.
- [37] M. Kottek, J. Grieser, C. Beck, B. Rudolf, and F. Rubel, “World map of the köppen-geiger climate classification updated,” *Meteorologische Zeitschrift*, vol. 15, pp. 259–263, 2006.
- [38] A. Lindholm, N. Wahlström, F. Lindsten, and T. B. Schön, *Machine Learning: A First Course for Engineers and Scientists*. Cambridge University Press, 2022. Pre-publication version, July 8, 2022.
- [39] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York: Springer, 2 ed., 2009.
- [40] IBM, “What is xgboost?,” n.d. Accessed: 2026-04-07.
- [41] GeeksforGeeks, “Artificial neural networks and its applications,” 2025. Last updated: 12 July 2025. Accessed: 9 April 2026.
- [42] P. K. Singh, A. Saraswat, and Y. Gupta, “Deep learning prediction models for short-term solar photovoltaic power generation forecasting,” *Next Energy*, vol. 11, p. 100531, 2026.
- [43] P. Majumdar, “Plant leaf disease detection using xgboost with optuna hyperparameter optimization,” *SN Computer Science*, vol. 6, p. 891, 2025.
- [44] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, “Optuna: A next-generation hyperparameter optimization framework,” in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.
- [45] Synogize, “Hyperparameter tuning for smart people: Why optuna beats gridsearchcv and randomized-searchcv.” <https://www.synogize.io/hyperparameter-tuning-for-smart-people-why-optuna-beats-gridsearchcv-and-randomizedsearchcv>, 2024. Accessed: 2026-04-13.
- [46] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, “Algorithms for hyper-parameter optimization,” in *Advances in Neural Information Processing Systems 24 (NeurIPS 2011)* (J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, eds.), Curran Associates, Inc., 2011.
- [47] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. VanderPlas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and Duchesnay, “Cross-validation: evaluating estimator performance.” https://scikit-learn.org/stable/modules/cross_validation.html, 2024. Accessed: 2026-04-14.
- [48] scikit-learn developers, “StandardScaler – scikit-learn 1.8.0 documentation,” 2025. Accessed: 2026-05-20.

- [49] A. S. Rajawat, O. Mohammed, R. N. Shaw, and A. Ghosh, "Chapter six - renewable energy system for industrial internet of things model using fusion-ai," in *Applications of AI and IOT in Renewable Energy* (R. N. Shaw, A. Ghosh, S. Mekhilef, and V. E. Balas, eds.), pp. 107–128, Academic Press, 2022.
- [50] S. Gawusu and X. Zhang, "Transformer-based deep learning for multi-horizon global power generation forecasting," *Energy Conversion and Management*, vol. 354, p. 121284, 2026.
- [51] M. Dastranj and A. R. Sepaskhah, "Modelling perennial saffron growth and yield under different irrigation water salinities, irrigation water levels and planting methods," *Agricultural Water Management*, vol. 323, p. 110097, 2026.
- [52] Desert Knowledge Australia Solar Centre, "Download data: Solibro, 5.2kw, cigs, fixed." <http://dkasolarcentre.com.au/download>, 2017. Accessed: 23/02/2026.
- [53] OpenEI, "Pvdaq site: Sr.co," n.d.
- [54] E. Sarmas, N. Matias, C. Pereira, and A. R. Antunes, "Photovoltaic power production dataset," 2025.
- [55] I. Lee, "Solar energy production." <https://www.kaggle.com/datasets/ivnlee/solar-energy-production>, 2023. Accessed: 12/02/2026.
- [56] SKYbrary, "Hot desert climate (bwh)." <https://skybrary.aero/articles/hot-desert-climate-bwh>. Accessed: dd/mm/yyyy.
- [57] SKYbrary, "Cold semi-arid climate (bsk)." <https://skybrary.aero/articles/cold-semi-arid-climate-bsk>. Accessed: 27/04/2026.
- [58] SKYbrary, "Warm summer humid continental climate (dfb)." <https://skybrary.aero/articles/warm-summer-humid-continental-climate-dfb>. Accessed: dd/mm/yyyy.
- [59] European Commission Joint Research Centre, "Pvgis - photovoltaic geographical information system." https://re.jrc.ec.europa.eu/pvg_tools/en/. Accessed: dd/mm/yyyy.
- [60] C. E. Clark and L. J. Cunningham, "Daylight saving time (DST)," Tech. Rep. R45208, Congressional Research Service, September 2020. Updated September 30, 2020.
- [61] Weather Spark, "Average weather in alice springs, northern territory, australia year round," 2024. Accessed: 2026.
- [62] Weather Spark, "Climate and average weather year round in kersey," 2024. Accessed: 2026.
- [63] Weather Spark, "Climate and average weather year round in lisbon," 2024. Accessed: 2026.
- [64] Weather Spark, "Climate and average weather year round in calgary," 2024. Accessed: 2026.
- [65] P. Bauer, A. Thorpe, and G. Brunet, "The quiet revolution of numerical weather prediction," *Nature*, vol. 525, pp. 47–55, 09 2015.

A Example of Decision Tree

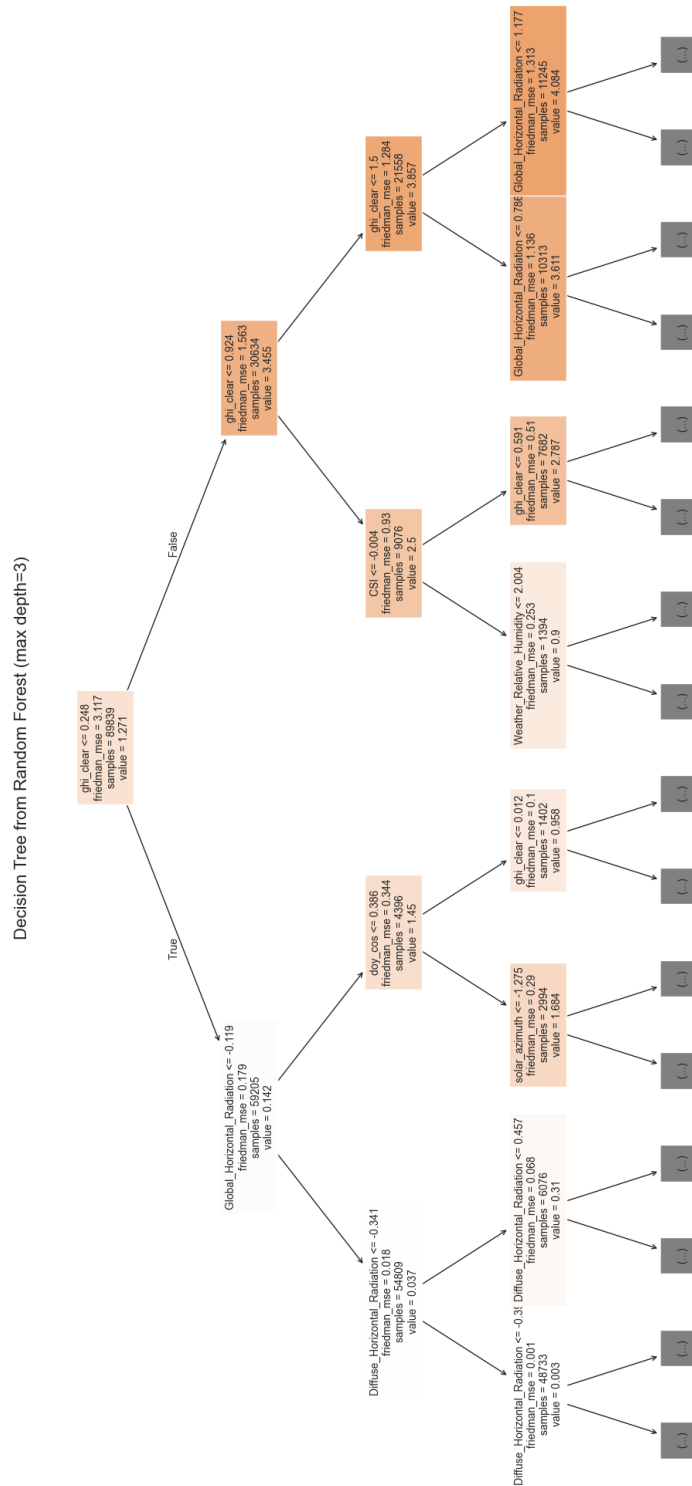


Figure 53: Example of a single decision tree.

B Pearson Correlation Heatmaps

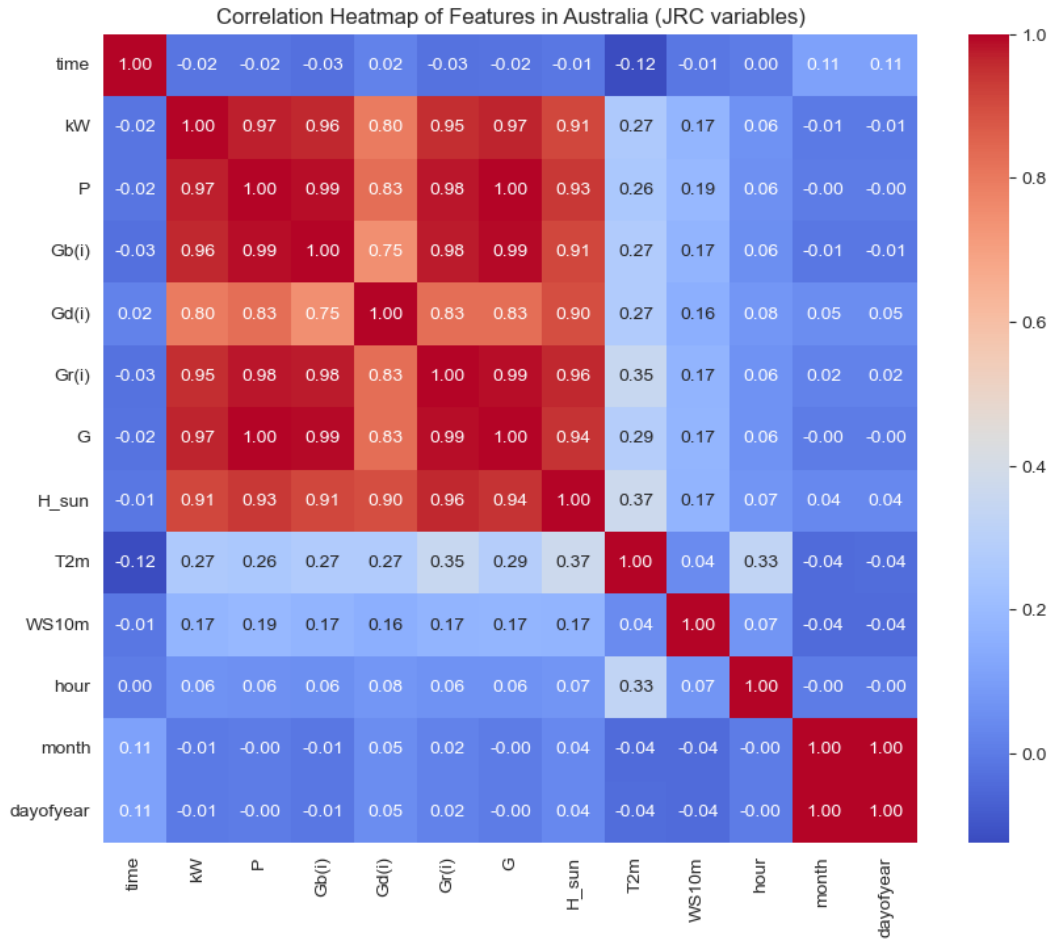


Figure 54: Pearson correlation heatmap of the Australian dataset (JRC variables)

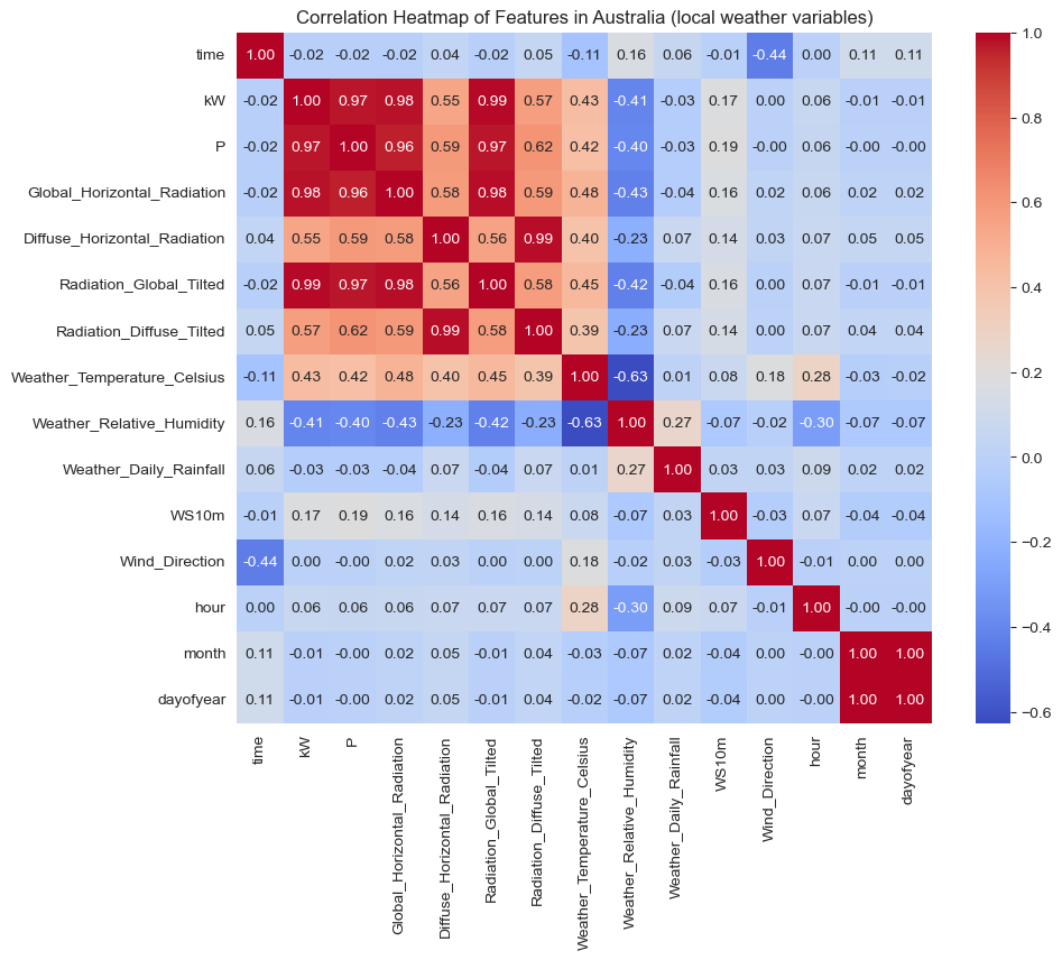


Figure 55: Pearson correlation heatmap of the Australian dataset (local weather variables)

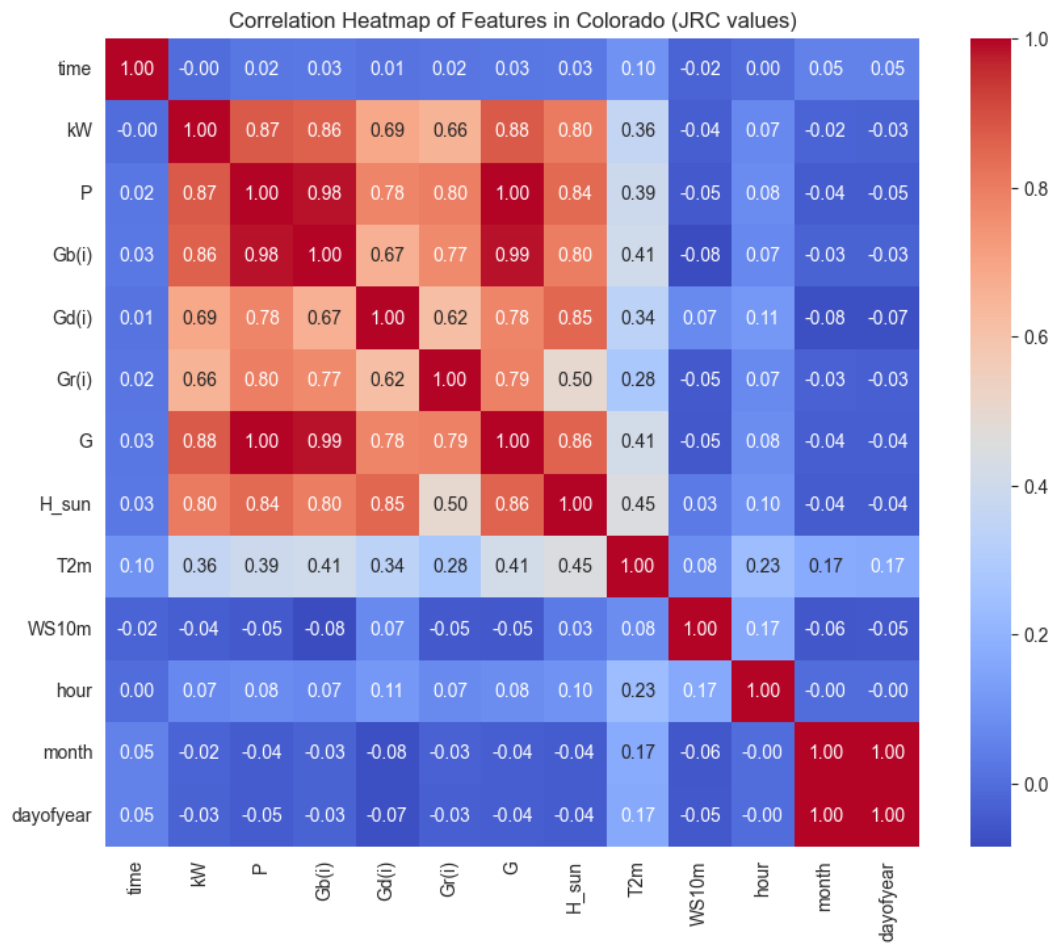


Figure 56: Pearson correlation heatmap of the American dataset (JRC variables)

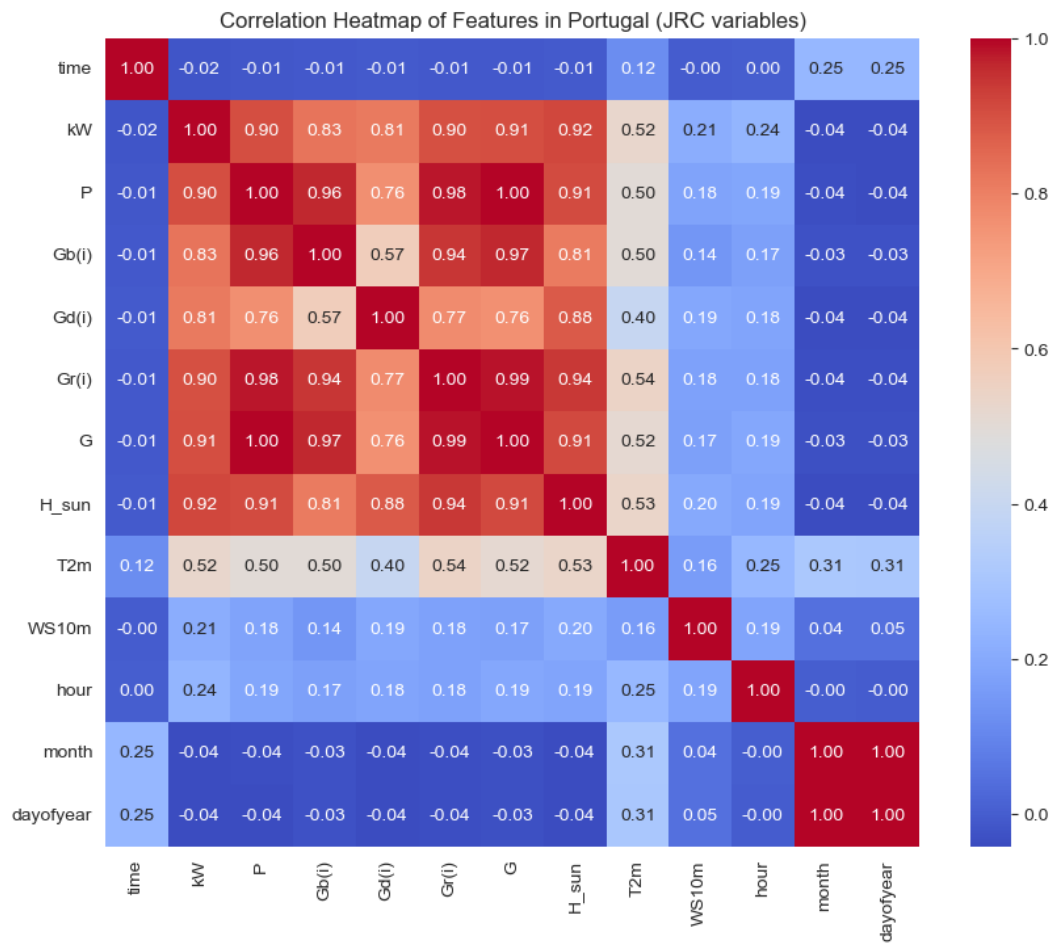


Figure 57: Pearson correlation heatmap of the Portuguese dataset (JRC variables)

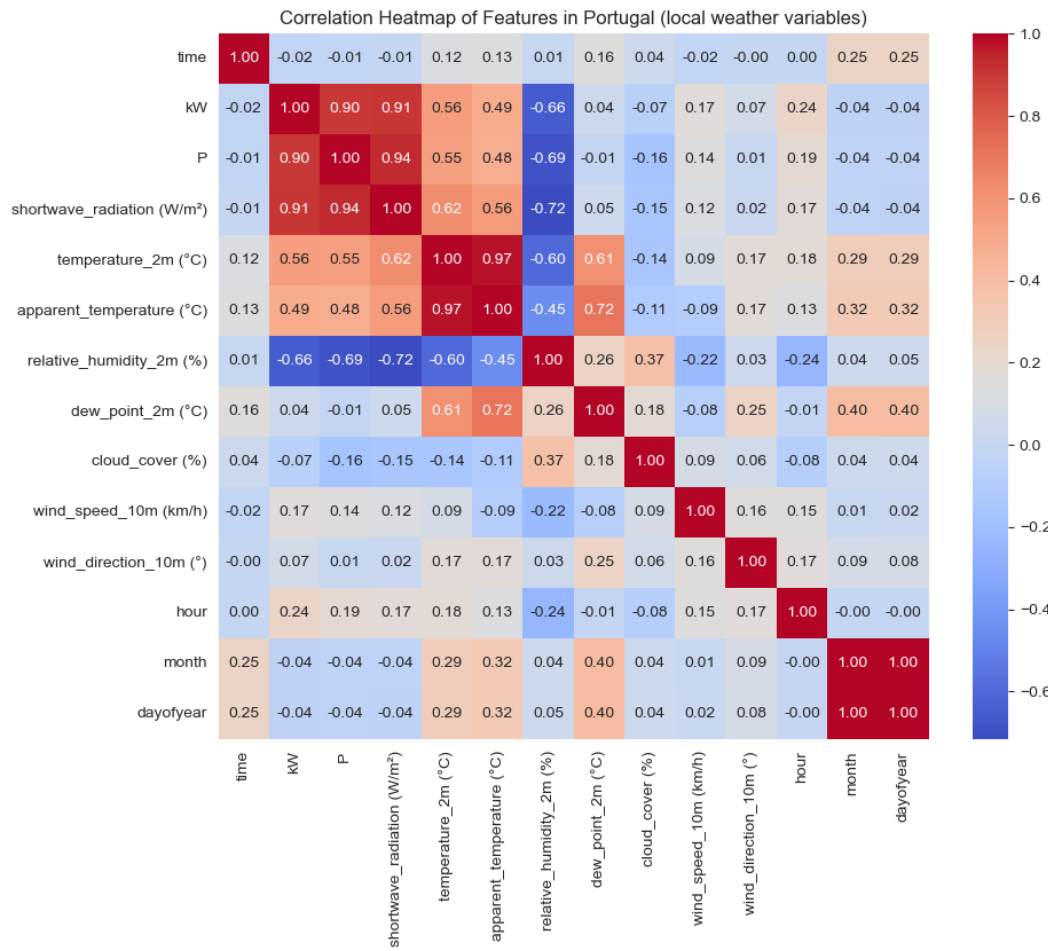


Figure 58: Pearson correlation heatmap of the Portuguese dataset (local weather variables)

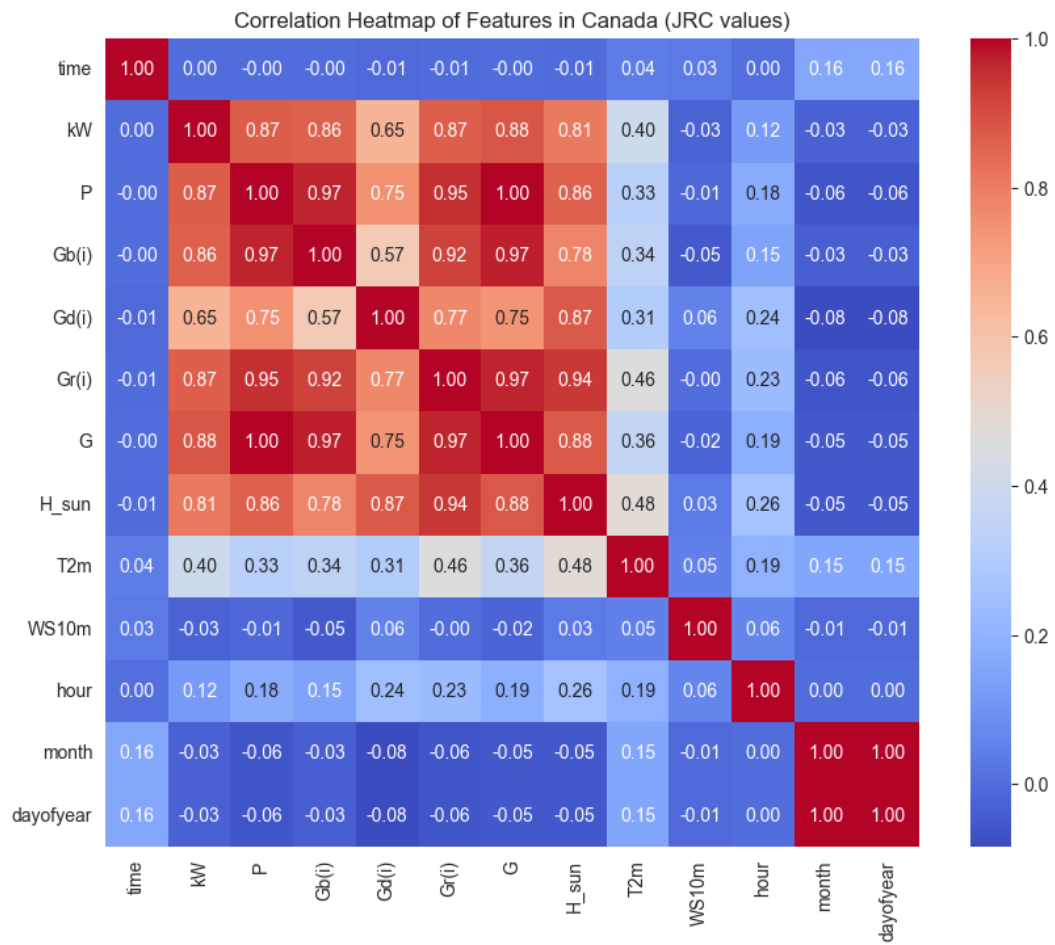


Figure 59: Pearson correlation heatmap of the Canadian dataset (JRC variables)

C Pairplots

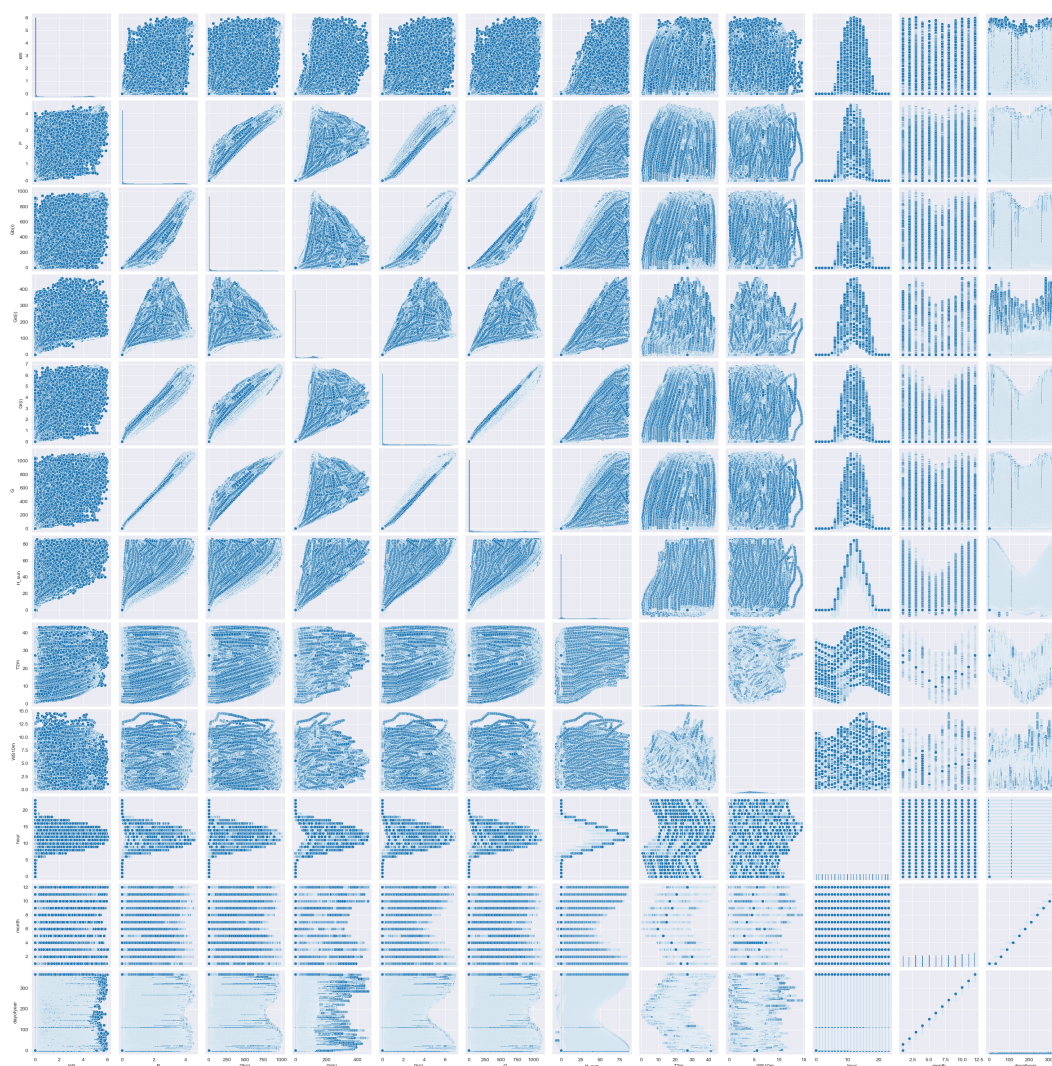


Figure 60: Pairplot of the Australian dataset (JRC variables)

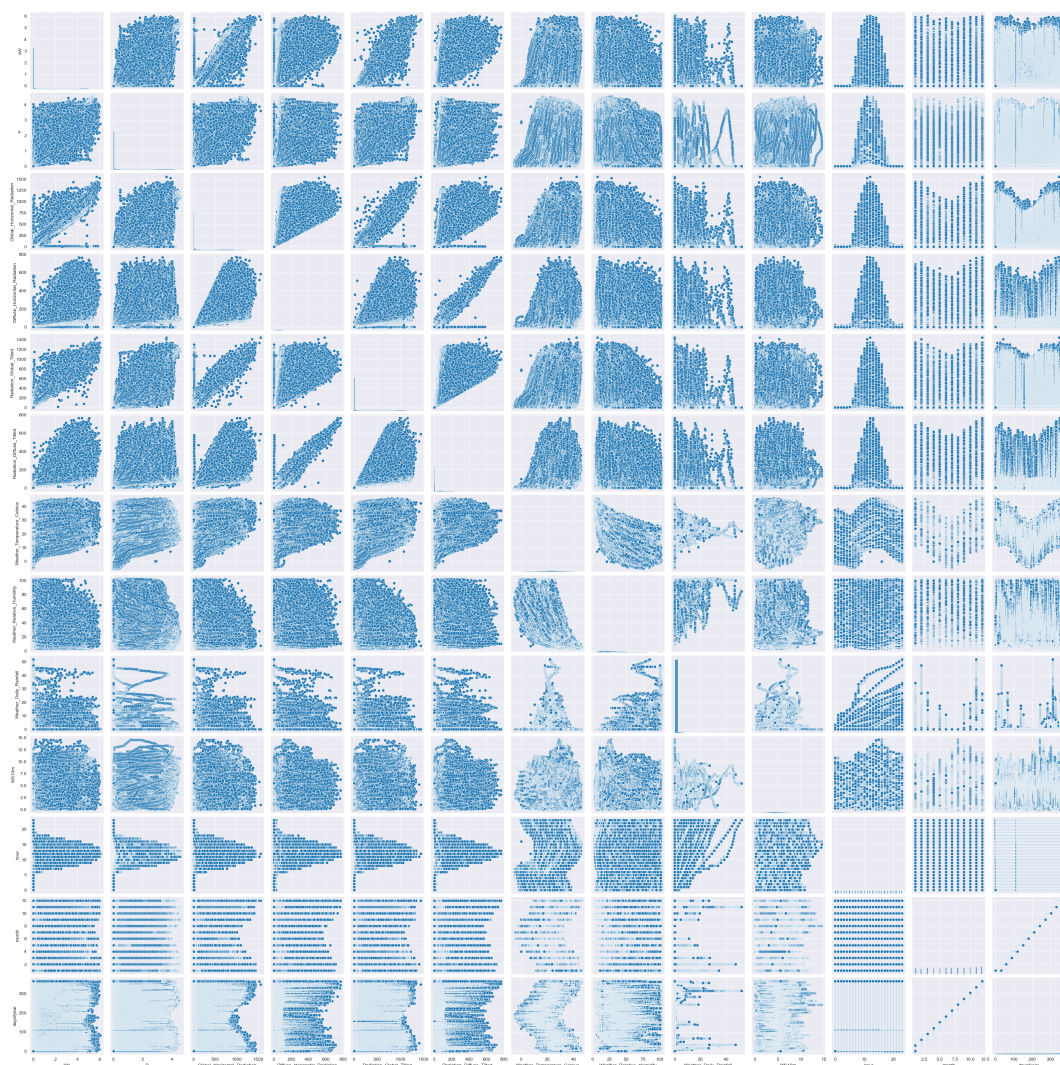


Figure 61: Pairplot of the Australian dataset (local weather variables)



Figure 62: Pairplot of the American dataset (JRC variables)

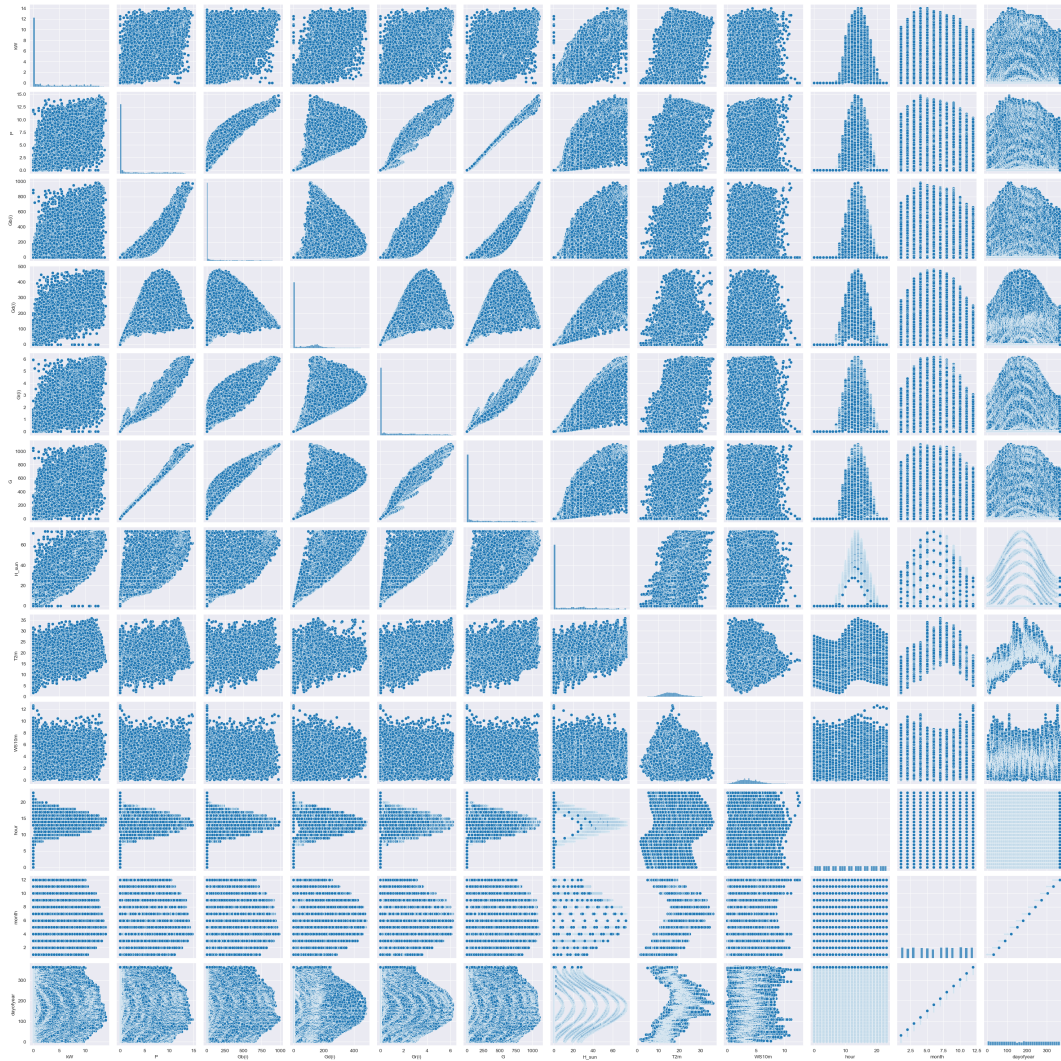


Figure 63: Pairplot of the Portuguese dataset (JRC variables)

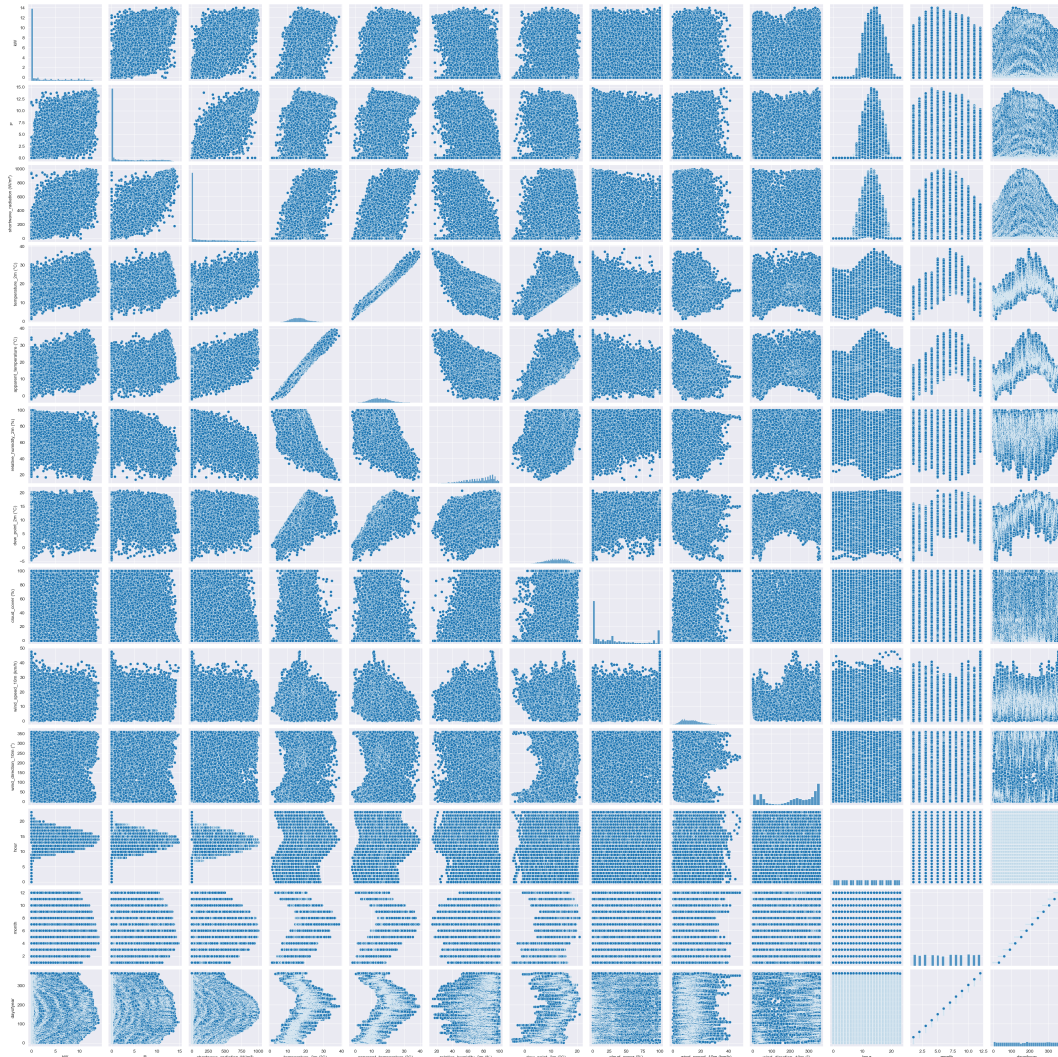


Figure 64: Pairplot of the Portuguese dataset (local weather variables)

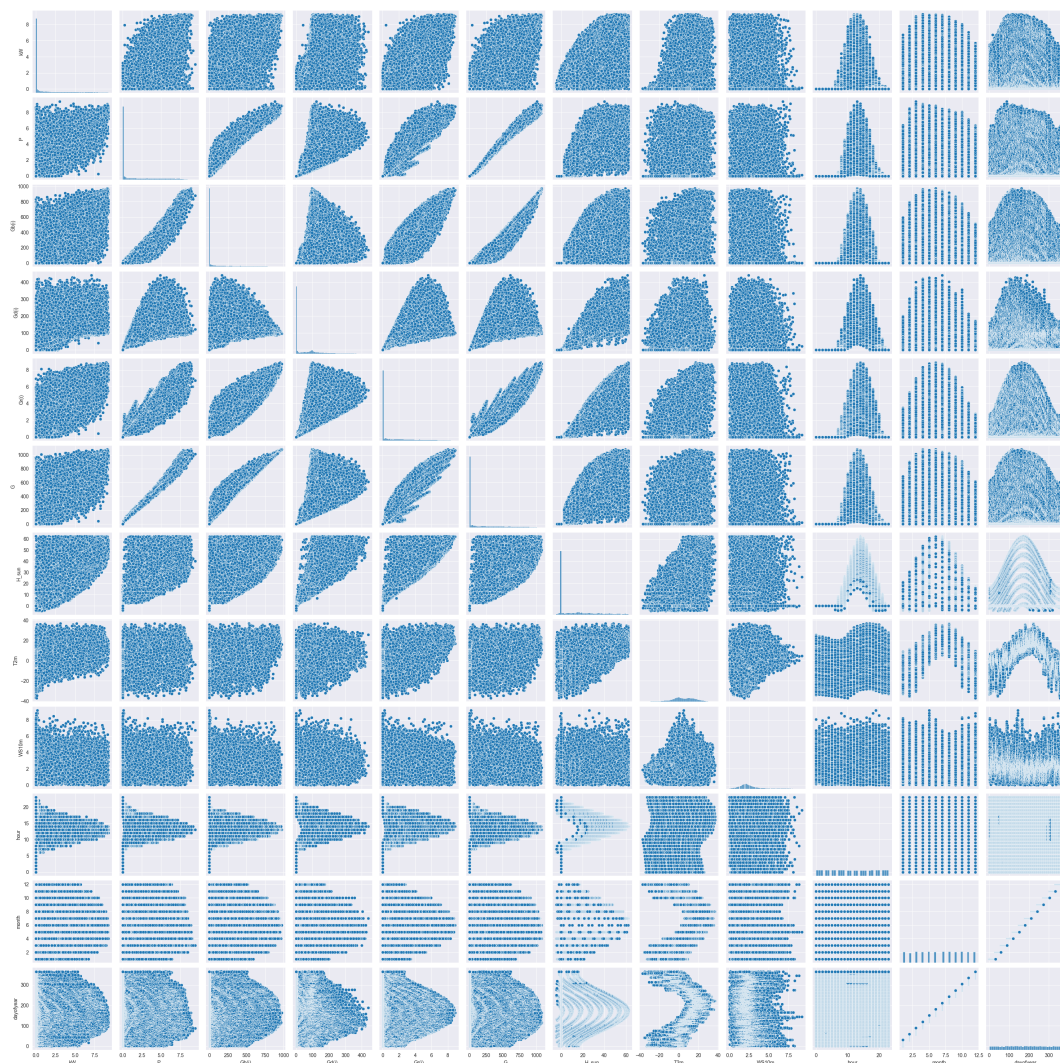


Figure 65: Pairplot of the Canadian dataset (JRC variables)