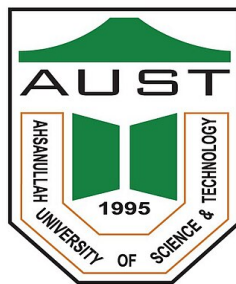


Application of machine learning tools for the analysis and prediction of wind and solar power generation

By

Tarek Ahmed 200108144
Md. Sabbir Hossain 200108148

A Thesis Submitted to the Department of
Mechanical and Production Engineering,
Ahsanullah University of Science and Technology
in Partial Fulfillment of
the requirements for the Degree of
BACHELOR OF SCIENCE IN MECHANICAL ENGINEERING



DEPARTMENT OF MECHANICAL AND PRODUCTION ENGINEERING (MPE)
AHSANULLAH UNIVERSITY OF SCIENCE AND TECHNOLOGY (AUST)
DHAKA-1208, BANGLADESH

CERTIFICATE OF ACCEPTANCE

The thesis entitled as “**Application of machine learning tools for the analysis and prediction of wind and solar power generation**” submitted by the following students have been accepted for partial fulfillment of the requirement for the degree of B.Sc. in Mechanical Engineering.

First author 20.01.08.144

Second author 20.01.08.148

Supervisor name

Dr. Md. Kharshiduzzaman

Associate professor

Department of Mechanical and Production Engineering

Ahsanullah University of Science and Technology, Dhaka.

External name

Dr. Fazlar Rahman

Associate Professor

Department of Mechanical and Production Engineering,

Ahsanullah University of Science and Technology, Dhaka

DECLARATION OF CANDIDATE

It is hereby declared that this thesis or any part of it has not been submitted elsewhere for the award of any degree or diploma.

First Author **20.01.08.144** **Signature:** _____

Second Author **20.01.08.148** **Signature:** _____

**This work is dedicated to
Our Loving Parents**

ACKNOWLEDGEMENT

I express deep thanks to Prof. Dr. Md. Kharshiduzzaman for granting me the opportunity to delve into the project "Application of machine learning tools for the analysis and prediction of wind and solar power generation." His guidance and support fueled extensive research, unveiling new perspectives. This enriching experience not only broadened my knowledge but also inspired personal and academic growth. I am genuinely thankful for the valuable insights gained under his mentorship.

ABSTRACT

The transition to renewable energy is crucial for mitigating climate change and reducing reliance on fossil fuels, but the intermittent nature of wind and solar power generation poses challenges for grid stability, resource optimization, and operational planning. Accurate forecasting is vital to address these issues, enabling grid operators to anticipate fluctuations, allocate resources effectively, and maintain stability. This thesis explores the application of machine learning techniques—Linear Regression, Decision Tree Regression, and Random Forest Regression—for analyzing and predicting wind and solar power generation using meteorological data and historical energy records. Among these models, Random Forest Regression demonstrated superior performance, capturing complex patterns and providing reliable short- and medium-term forecasts. Rigorous preprocessing, feature selection, and validation underscored the importance of data quality in achieving accurate predictions. While challenges such as limited data availability, computational demands, and model interpretability remain, the findings highlight the transformative potential of machine learning in improving renewable energy integration. Future work could explore hybrid models, integrate additional meteorological variables, and enhance model scalability to address these limitations. By advancing renewable energy forecasting, this study contributes to improving grid stability, reducing operational costs, and supporting the global transition to sustainable, low-carbon energy systems.

Contents

Certificate of Acceptance	ii
Declaraton of Candidate	iii
Acknowledgement	v
Abstract	vi
List of Figures	x
List of Table	xi
Nomenclature	1
1 Introduction	1
1.1 Background	1
1.2 Problem Statement	2
1.3 Objectives	2
1.4 Initial Idea of an ML algorithms	2
1.5 Limitations	3
1.6 Scopes of this chapter	3
2 Literature review	4
2.1 Forecasting Horizon and Temporal Resolution	4
2.2 Uncertainty Quantification and Probabilistic Forecasting	5
2.3 Predicting the wind and solar generation using linear regression	5
2.4 R-squared (R2) score	6
2.5 Forecast Evaluation Metrics and Model Validation	6
2.6 Forecasting for Energy Markets and Economic Analysis	7
2.7 Case Studies and Practical Applications	8
2.8 Social, Environmental, and Policy Implications	8
2.9 Features/Guidelines	9
2.9.1 Features	9
2.9.2 Guidelines	9
2.10 Testing/Certification	10
2.11 Validation and Verification	10
2.12 The Development and Algorithm Methodology	10
2.13 Research Gaps	11

3	The Methodology	13
3.1	Step 1: The statement of the problem	13
3.2	Step 2: The determination of objectives	14
3.3	Step 3: The process of reviewing the literature	14
3.4	Step 4: The formulation of methodology	14
3.5	Step 5: The process of Development	15
3.6	Step 6: The Reporting	18
4	The Process of Development	19
4.1	Refine Problem Definition and Constraints	19
4.1.1	Problem Definition	19
4.1.2	Constraints	20
4.2	Research and Investigate	21
4.2.1	Development Variables	21
4.2.2	Algorithms	22
4.3	Data Collection and Preparation	26
4.4	Solar Power Generation Prediction Using Weather Data	28
4.5	Wind Power Generation Prediction Using Weather Data	30
4.6	Solar Power Generation Forecasting	33
4.7	Wind Power Generation Forecasting	36
5	Results & Optimization	40
5.1	Solar Power Generation	40
5.1.1	Linear Regression	40
5.1.2	Decision Tree Regression	41
5.1.3	Random Forest Regression	43
5.2	Wind Power Generation	46
5.2.1	Linear Regression	46
5.2.2	Decision Tree Regression	47
5.2.3	Random Forest Regression	49
5.3	Solar Power Generation Forecasting	52
5.3.1	Linear Regression	52
5.3.2	Random Forest Regression	57
5.4	Wind Power Generation Forecasting	65
5.4.1	Linear Regression	65
5.4.2	Random Forest Regression	69
6	Conclusions	77
A	GANTT Chart for Spring 2023	80
B	GANTT Chart for Fall 2023	81

List of Figures

1.1	Initial idea of the Training process	2
1.2	Initial idea of the Prediction process	3
2.1	Forecasting Horizon	4
2.2	Uncertainty Quantification	5
2.3	Linear function of the input equation	5
2.4	Mean squared error	6
2.5	Coefficient of determination R^2	6
2.6	Forecast Evaluation of Wind Generation	7
2.7	Forecast Evaluation of Solar Generation	7
2.8	Forecasting for Energy Markets	8
2.9	Flow Chart of development Methodology	11
3.1	Methodology of the Final Year Project	13
3.2	An Exemplary Development Journey	16
4.1	Final output of Data cleaning process	26
4.2	The Generation of Wind power (MWh) vs. Weather data features	27
4.3	The Generation of Solar power (MWh) vs. Weather data features	27
4.4	Correlation Matrix Heatmap	27
5.1	Performance Summary of Regression Model	41
5.2	Performance Summary of Regression Model	42
5.3	Performance Summary of Regression Model	44
5.4	Performance Summary of Regression Model	47
5.5	Performance Summary of Decision Tree Regression	48
5.6	Performance Summary of Random Forest Regression	50
5.7	Performance of Linear Regression for Solar Power Generation at Different Time Steps	53
5.8	Performance of Linear Regression for Solar Power Generation at Different Time Steps	55
5.9	Performance of Linear Regression for Solar Power Generation at Different Time Steps	56
5.10	Performance of Linear Regression for Solar Power Generation at Different Time Steps	57
5.11	Performance of Random Forest Regression for Solar Power Generation at Different Time Steps	59
5.12	Performance of Random Forest Regression for Solar Power Generation at Different Time Steps	60

5.13	Performance of Random Forest Regression for Solar Power Generation at Different Time Steps	61
5.14	Performance of Random Forest Regression for Solar Power Generation at Different Time Steps	63
5.15	R^2 (Test) Comparison Between Models at Different Time Steps	64
5.16	Performance of Linear Regression for Wind Power Generation at Different Time Steps	66
5.17	Performance of Linear Regression for Wind Power Generation at Different Time Steps	67
5.18	Performance of Linear Regression for Wind Power Generation at Different Time Steps	68
5.19	Performance of Linear Regression for Wind Power Generation at Different Time Steps	69
5.20	Performance of Random Forest Regression for Wind Power Generation at Different Time Steps	71
5.21	Performance of Random Forest Regression for Wind Power Generation at Different Time Steps	72
5.22	Performance of Random Forest Regression for Wind Power Generation at Different Time Steps	73
5.23	Performance of Random Forest Regression for Wind Power Generation at Different Time Steps	75
5.24	R^2 (Test) Comparison Between Models at t+1h and t+6h	75

List of Tables

5.1	Performance of Linear Regression	41
5.2	Performance of Decision Tree Regression	42
5.3	Performance of Random Forest Regression	43
5.4	Performance of Random Forest Regression with Varying Trees	45
5.5	Feature Importance with Random Column	45
5.6	Enhanced Predictive Accuracy with Multi-Feature Model	46
5.7	Enhanced Predictive Accuracy with Single Feature Model	46
5.8	Performance of Linear Regression	47
5.9	Performance of Decision Tree Regression	48
5.10	Performance of Random Forest Regression	49
5.11	Performance of Random Forest Regression with Varying Trees	51
5.12	Feature Importance Using Random Test	51
5.13	Enhanced Predictive Accuracy with Multi-Feature Model	51
5.14	Enhanced Predictive Accuracy with Single Feature Model	52
5.15	Performance of Linear Regression for Solar Power Generation	53
5.16	Performance of Linear Regression for Solar Power Generation	54
5.17	Performance of Linear Regression for Solar Power Generation	56
5.18	Performance of Linear Regression for Wind Power Generation	57
5.19	Performance of Random Forest Regression for Solar Power Generation . .	58
5.20	Performance of Random Forest Regression for Solar Power Generation . .	60
5.21	Performance of Random Forest Regression for Solar Power Generation . .	61
5.22	Performance of Random Forest Regression for Solar Power Generation . .	62
5.23	Performance Comparison of Models	63
5.24	Performance of Linear Regression for Wind Power Generation	65
5.25	Performance of Linear Regression for Wind Power Generation	67
5.26	Performance of Linear Regression for Wind Power Generation	68
5.27	Performance of Linear Regression for Wind Power Generation	69
5.28	Performance of Random Forest Regression for Wind Power Generation .	70
5.29	Performance of Random Forest Regression for Wind Power Generation .	72
5.30	Performance of Random Forest Regression for Wind Power Generation .	73
5.31	Performance of Random Forest Regression for Wind Power Generation .	74
5.32	R^2 (Test) Comparison Between Models at t+1h and t+6h	75

Chapter 1

Introduction

1.1 Background

The transition towards renewable energy sources has become a global imperative to mitigate climate change and reduce dependency on fossil fuels. In this context, renewable energy adoption has seen significant progress, with renewables constituting approximately 46 percent of total net electricity generation by 2019 ¹. Wind energy emerged as a dominant contributor at 25 percent, surpassing brown coal and nuclear energy.

However, the intermittent nature of wind and solar power generation presents significant challenges for grid operators ². Unlike conventional power sources, wind and solar energy production fluctuates with meteorological conditions, making grid stability and supply-demand balance more complex. The variability in output can lead to voltage fluctuations, grid instability, and operational challenges.

Forecasting wind and solar power generation has thus become essential for effective grid management and optimizing energy resources ³. Accurate predictions enable grid operators to anticipate fluctuations, plan for contingencies, and optimize resource allocation. Traditional forecasting methods often rely solely on historical data or meteorological models, which may not capture the intricate dynamics of renewable energy systems accurately.

Machine learning (ML) techniques offer a promising avenue for improving the accuracy of wind and solar power forecasts ⁴. By leveraging vast amounts of data, including meteorological variables, historical power generation records, and grid data, ML algorithms can uncover complex patterns and relationships to make more precise predictions. These advanced forecasting models can significantly enhance grid stability, increase renewable energy integration, and reduce operational costs, contributing to the ongoing energy transition and sustainability efforts.

¹According to data from time-series data on wind and solar power production, retrieved on NASA MERRA-2

²Grid operators face challenges related to the intermittent nature of wind and solar power generation, leading to issues such as voltage fluctuations and grid instability.

³Effective forecasting of wind and solar power generation is crucial for grid management and resource optimization

⁴Machine learning techniques offer promising capabilities for improving the accuracy of renewable energy forecasts by leveraging diverse datasets and advanced algorithms.

1.2 Problem Statement

The integration of intermittent renewable energy sources like wind and solar into the power grid presents significant challenges, including grid instability and difficulty in balancing supply and demand. Accurate forecasting of wind and solar power generation is crucial for efficient grid management, optimal resource allocation, and cost reduction. Traditional forecasting methods often fall short in capturing the complexities of renewable energy systems. Hence, there is a pressing need for advanced forecasting models that can effectively leverage meteorological data and historical power generation records to predict renewable energy output accurately.

1.3 Objectives

1. Develop machine learning algorithms to predict wind and solar power generation from meteorological data.
2. Forecast wind and solar power generation at different time horizons, ranging from $t+1h$ to $t+6h$.
3. Improve grid stability and efficiency by providing accurate and timely forecasts of renewable energy output.
4. Reduce operational costs and optimize resource allocation in the power sector through effective utilization of renewable energy forecasts.

1.4 Initial Idea of an ML algorithms

For structural models, we propose utilizing numerical weather prediction (NWP) data to forecast wind and solar power generation. Support Vector Machines (SVM) and Random Forests are promising algorithms for this task, leveraging NWP variables as input features to predict renewable energy output.

Time-series models aim to forecast power generation based on historical data. Techniques such as Autoregressive Integrated Moving Average (ARIMA) or Long Short-Term Memory (LSTM) networks are suitable for endogenous forecasting, utilizing past power generation records as input features.

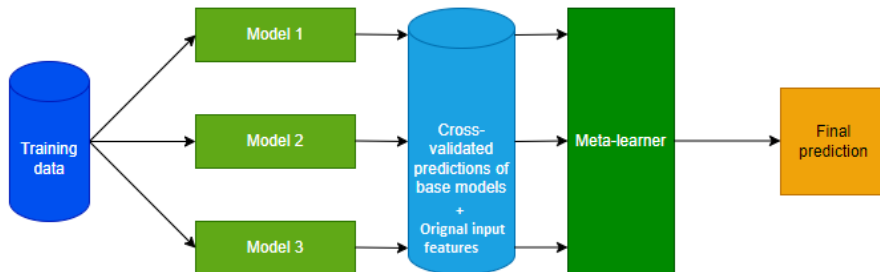


Figure 1.1: Initial idea of the Training process

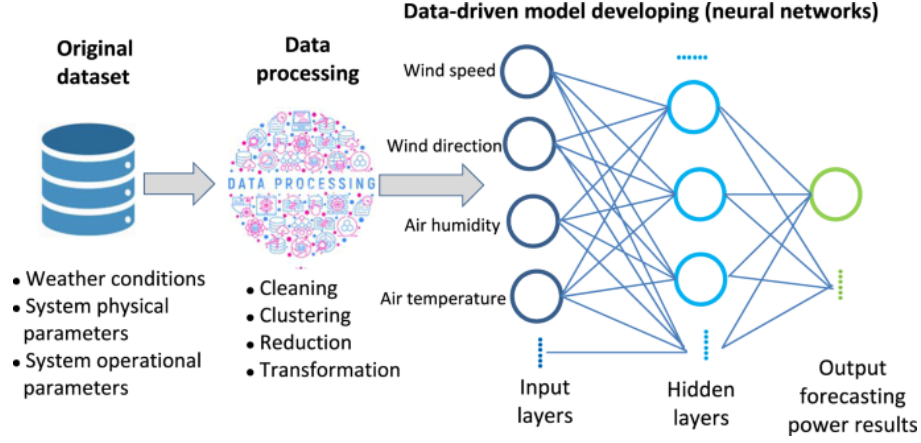


Figure 1.2: Initial idea of the Prediction process

Hybrid models combine NWP data with historical power generation records to enhance forecasting accuracy. By incorporating both exogenous and endogenous variables, algorithms like Gradient Boosting Machines (GBM) or Recurrent Neural Networks (RNNs) can provide more robust predictions, considering the complex interplay between meteorological conditions and historical energy production. These approaches offer diverse strategies for accurately forecasting wind and solar power generation, crucial for effective grid management and renewable energy integration.

1.5 Limitations

Limitations of this project include potential inaccuracies in weather forecasting data, which may affect the reliability of predictions. Moreover, the complex interplay between meteorological conditions and renewable energy generation poses challenges for model accuracy. Additionally, the dynamic nature of grid operations and unforeseen events could impact the effectiveness of forecasting algorithms. Furthermore, the availability and quality of historical data may vary, affecting the robustness of the models. Lastly, regulatory constraints and policy changes could influence the implementation and scalability of the proposed forecasting solutions.

1.6 Scopes of this chapter

The scope of this chapter encompasses a comprehensive review of relevant literature pertaining to wind and solar power forecasting. It will explore various methodologies, including traditional statistical approaches and machine learning techniques, used for renewable energy prediction. Additionally, the chapter will discuss challenges and limitations encountered in forecasting renewable energy output, such as data availability, model complexity, and grid integration issues. Furthermore, it will highlight recent advancements and emerging trends in the field, providing a foundation for the development and evaluation of forecasting models in subsequent chapters of the thesis.

Chapter 2

Literature review

2.1 Forecasting Horizon and Temporal Resolution

Determining the optimal forecasting horizon and temporal resolution is pivotal for accurate wind and solar power prediction. Short-term forecasts, spanning hours to days, are indispensable for facilitating immediate decision-making by grid operators. Meanwhile, medium and long-term forecasts, encompassing weeks to months, play a crucial role in strategic energy planning and market operations. Achieving a balance between accuracy and computational complexity across these varied horizons is imperative for enhancing the performance and practical utility of forecasting models in real-world scenarios.[1]

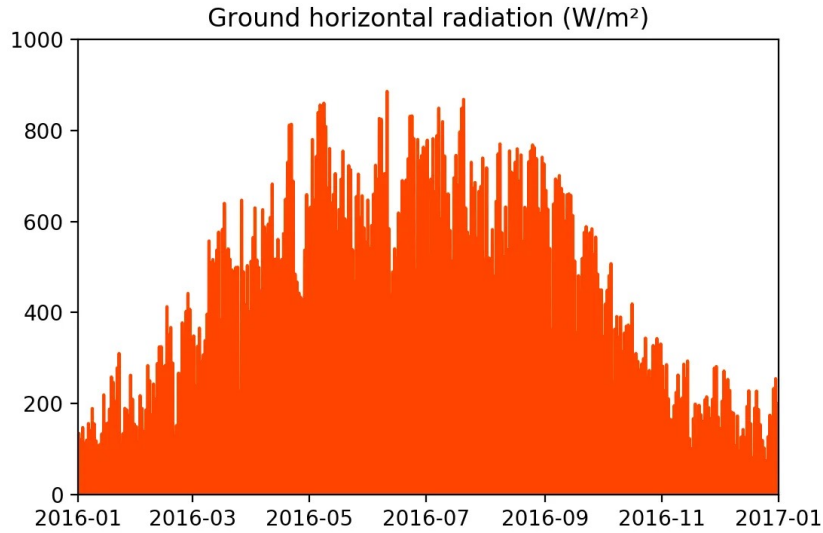


Figure 2.1: Forecasting Horizon

Determining the ideal forecasting horizon and temporal resolution involves weighing the specific needs of grid operators, energy planners, and market participants.[2] Short-term forecasts offer real-time insights for immediate operational adjustments, while medium and long-term forecasts inform strategic planning and investment decisions. By optimizing forecasting models to accommodate these diverse requirements, stakeholders can effectively harness the potential of renewable energy sources and facilitate the transition towards a sustainable energy future.[3]

2.2 Uncertainty Quantification and Probabilistic Forecasting

Quantifying uncertainty in wind and solar power forecasts is paramount for informed decision-making and risk management in energy systems. Probabilistic forecasting techniques, such as ensemble methods and Bayesian inference, provide valuable insights by generating probability distributions of future energy generation.[4]

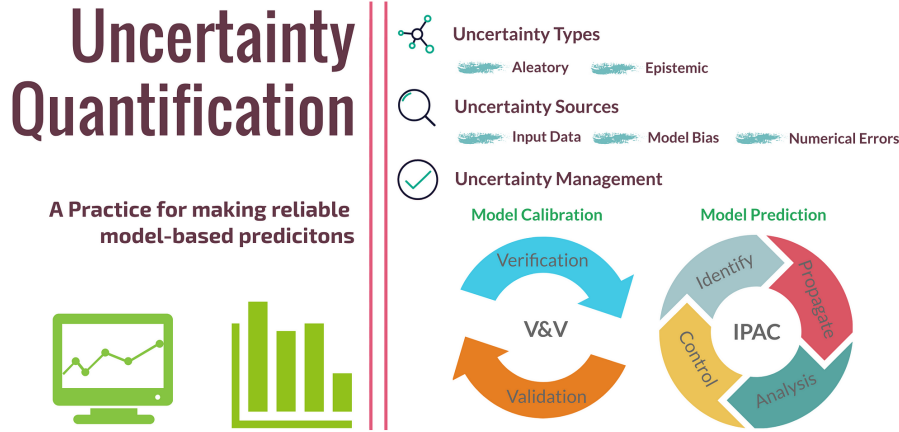


Figure 2.2: Uncertainty Quantification

By incorporating uncertainty estimates into forecasting models, stakeholders can better assess the reliability of predictions and plan for contingencies, ultimately enhancing grid reliability and resilience in the face of variability and uncertainty inherent in renewable energy sources.[5]

2.3 Predicting the wind and solar generation using linear regression

In the realm of renewable energy forecasting, linear regression models have garnered attention for predicting wind and solar power generation. These models leverage historical data of meteorological variables, such as wind speed and solar irradiance, to estimate future energy output. Linear regression offers simplicity and interpretability, making it an appealing choice for forecasting tasks. Studies have explored various adaptations of linear regression, including multivariate regression models that incorporate additional predictors to enhance prediction accuracy.[6] Despite its straightforward approach, linear regression may face limitations in capturing nonlinear relationships and complex interactions inherent in renewable energy systems.

$$f : \mathbb{R}^n \rightarrow \mathbb{R}, \quad \hat{y} \equiv f(\mathbf{x}) = \boldsymbol{\beta}^\top \mathbf{x} + \beta_0$$

Figure 2.3: Linear function of the input equation

Researchers have addressed these challenges by refining model features, incorporating advanced data preprocessing techniques, and exploring ensemble approaches to improve

$$\underset{\beta, \beta_0}{\operatorname{argmin}} \frac{1}{N} \sum_{i=1}^N (y_i - \beta^\top \mathbf{x}_i - \beta_0)^2$$

Figure 2.4: Mean squared error

forecasting performance. While linear regression serves as a foundational technique in wind and solar power forecasting, ongoing research aims to augment its capabilities through integration with other machine learning algorithms and advanced statistical methods, thereby advancing the accuracy and reliability of renewable energy predictions.[7]

2.4 R-squared (R2) score

In the landscape of renewable energy forecasting, the evaluation metric known as the R-squared (R2) score holds significance in assessing the efficacy of predictive models, including those employing linear regression. This metric quantifies the proportion of variability in the dependent variable that is explained by the independent variables. In the context of wind and solar power generation prediction, a higher R2 score indicates a stronger correlation between the forecasted and observed energy outputs, thereby reflecting the model's predictive accuracy.[8]

$$R^2 = 1 - \frac{\sum_i (y_i - f(\mathbf{x}_i))^2}{\sum_i (y_i - \bar{y})^2}$$

Figure 2.5: Coefficient of determination R²

Literature exploring the use of linear regression for wind and solar power forecasting often includes discussions on optimizing R2 scores through feature engineering, model refinement, and data preprocessing techniques. However, despite its utility, linear regression's reliance on linear relationships may limit its ability to capture the complexities inherent in renewable energy systems. Researchers strive to address this challenge by integrating advanced modeling techniques and ensemble approaches to enhance predictive performance and achieve higher R2 scores.[9] Consequently, the R2 score serves as a crucial benchmark for evaluating the effectiveness and reliability of linear regression models in renewable energy forecasting.

2.5 Forecast Evaluation Metrics and Model Validation

Evaluating the performance of wind and solar power forecasting models requires robust metrics and validation techniques.[10]

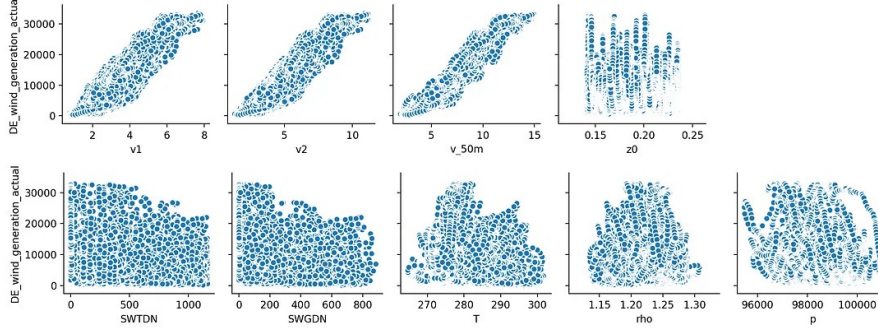


Figure 2.6: Forecast Evaluation of Wind Generation

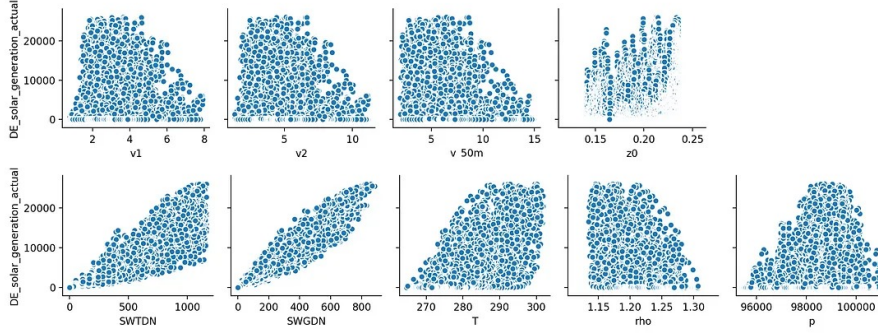


Figure 2.7: Forecast Evaluation of Solar Generation

Common evaluation metrics include Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Forecast Skill Score (FSS), which quantify the accuracy and skill of predictions against observed data. Model validation involves testing the forecasting model's performance on independent datasets and assessing its reliability under different environmental conditions and operational scenarios.[11] Rigorous evaluation ensures the trustworthiness and effectiveness of forecasting models, guiding stakeholders in selecting suitable models for practical applications.

2.6 Forecasting for Energy Markets and Economic Analysis

Forecasting wind and solar power generation is crucial for energy market operations, pricing mechanisms, and economic analysis in the renewable energy sector. Accurate forecasts enable market participants to optimize energy trading strategies, manage risk exposure, and enhance revenue generation from renewable energy assets.[12]

Economic analysis tools, such as levelized cost of energy (LCOE) calculations and financial modeling, rely on reliable forecasts to assess the profitability and viability of renewable energy projects.[13] By providing insights into future energy supply and demand dynamics, forecasting supports informed decision-making and investment planning in energy markets.



Figure 2.8: Forecasting for Energy Markets

2.7 Case Studies and Practical Applications

Analyzing case studies and real-world applications of wind and solar power forecasting offers valuable insights into the performance and impact of forecasting models across diverse contexts. These case studies provide tangible examples of the practical challenges, implementation strategies, and outcomes of forecasting initiatives in various geographical regions, climates, and grid infrastructures. By examining successful applications and learning from past experiences, stakeholders can identify best practices and evaluate the effectiveness of forecasting models. These insights enable stakeholders to refine renewable energy management strategies and optimize grid integration efforts. Additionally, the lessons learned from case studies empower stakeholders to make informed decisions, mitigate risks, and enhance the overall efficiency and reliability of renewable energy systems. Overall, case studies serve as invaluable resources for advancing the field of wind and solar power forecasting and driving the transition towards a more sustainable energy future.

2.8 Social, Environmental, and Policy Implications

Exploring the social, environmental, and policy implications of wind and solar power forecasting sheds light on its broader societal relevance and significance. Forecasting plays a critical role in facilitating the transition to renewable energy, reducing greenhouse gas emissions, and mitigating climate change impacts.[14] Social and environmental considerations, such as community acceptance, land use planning, and biodiversity conservation, influence renewable energy deployment decisions and forecasting priorities.[15] Policy frameworks, regulations, and incentives shape the development and adoption of forecasting technologies, influencing their effectiveness and adoption in real-world contexts. Understanding these implications is essential for fostering informed decision-making and sustainable energy transitions.

2.9 Features/Guidelines

2.9.1 Features

- **Solar power generation (MWh):** Total electricity generated from solar energy sources, measured in megawatt-hours (MWh), reflecting the output of solar panels or solar power plants over a given period.
- **Wind power generation (MWh):** Total electricity generated from wind energy sources, measured in megawatt-hours (MWh), indicating the output of wind turbines or wind farms during a specific timeframe.
- **Solar installed capacity (MW):** Total capacity of solar energy systems installed and available for electricity generation, measured in megawatts (MW), representing the maximum output potential under optimal conditions. Improve grid stability and efficiency by providing accurate and timely forecasts of renewable energy output.
- **Wind installed capacity (MW) :** Total capacity of wind energy systems installed and operational, measured in megawatts (MW), indicating the maximum power output achievable from wind turbines or wind farms.
- **Surface solar irradiance (W/m²):** Solar energy flux received per unit area at the Earth's surface, measured in watts per square meter (W/m²), indicating the intensity of sunlight available for solar power generation.
- **Air temperature 2 meters above ground (°C) :** Temperature of the air measured at a height of 2 meters above ground level, expressed in degrees Celsius (°C), affecting the efficiency of both solar and wind energy systems.
- **Air density at ground level (kg/m³):** Density of air at ground level, measured in kilograms per cubic meter (kg/m³), influencing the performance and aerodynamics of wind turbines.
- **Precipitation (mm/hour):** Rate of water falling from the atmosphere to the ground, measured in millimeters per hour (mm/hour), impacting solar power generation and potentially affecting wind turbine operation.
- **Snowfall (mm/hour):** Rate of snow accumulation on the ground, measured in millimeters per hour (mm/hour), potentially affecting the performance and efficiency of both solar and wind energy systems.
- **Cloud cover fraction ([0, 1] scale):** Fractional measure representing the extent of sky covered by clouds, ranging from 0 (clear sky) to 1 (completely overcast), influencing the availability of sunlight for solar power generation and impacting solar irradiance levels.

2.9.2 Guidelines

1. **Guidelines for Data Collection:** Follow established guidelines for collecting meteorological and energy generation data, ensuring consistency, accuracy, and reliability in data acquisition processes.

2. **Guidelines for Model Development:** Adhere to guidelines for developing forecasting models, including best practices for feature selection, model selection, and parameter tuning, to ensure robust and effective model performance.
3. **Guidelines for Model Evaluation:** Utilize guidelines for evaluating forecasting models, incorporating standard metrics and methodologies for assessing predictive accuracy, reliability, and uncertainty, to ensure rigorous evaluation and comparison of model performance.
4. **Guidelines for Reporting Results:** Follow guidelines for reporting research results, including clear and transparent documentation of methodologies, procedures, and findings, ensuring reproducibility and transparency in research outcomes.
5. **Guidelines for Ethical Conduct:** Adhere to ethical guidelines for research conduct, including principles of integrity, honesty, and respect for intellectual property rights, to ensure the highest standards of professionalism and ethical conduct in the thesis project.

2.10 Testing/Certification

Testing and certification processes for wind and solar power forecasting models ensure their accuracy, reliability, and performance under diverse environmental conditions and operational scenarios. Certification bodies may conduct rigorous testing procedures, including validation against observed data, benchmarking against industry standards, and evaluation of forecasting accuracy and skill metrics.[16] Certified forecasting models provide stakeholders with confidence in their suitability for real-world applications, such as grid management, energy trading, and renewable energy investment decisions, fostering trust and credibility in forecasting providers and supporting the widespread adoption of forecasting technologies.[17]

2.11 Validation and Verification

Validation involves assessing the ability of models to accurately represent real-world wind and solar power generation dynamics. This entails comparing model predictions with observed data from historical records or real-time measurements. Techniques like cross-validation help gauge the generalization performance of models across diverse datasets, ensuring they capture underlying patterns effectively.[18]

On the other hand, verification focuses on confirming the correctness of model implementations and algorithms. This process involves rigorous testing to verify that the code and algorithms function as intended, producing accurate and reliable results.[19] Conducting comprehensive validation and verification procedures bolsters the credibility of forecasting models, empowering stakeholders to make well-informed decisions in renewable energy management and grid integration strategies.

2.12 The Development and Algorithm Methodology

The Development and implementation of experimental methodologies in wind and solar power forecasting research are fundamental for producing credible and reproducible

outcomes. Researchers meticulously craft experimental protocols, encompassing data collection procedures, model selection criteria, and evaluation metrics, to ensure scientific rigor and reliability.[20] Transparent documentation of experimental processes, including the delineation of data sources and analysis techniques, enables peer review and result reproducibility, fostering a culture of accountability and trust within the scientific community.

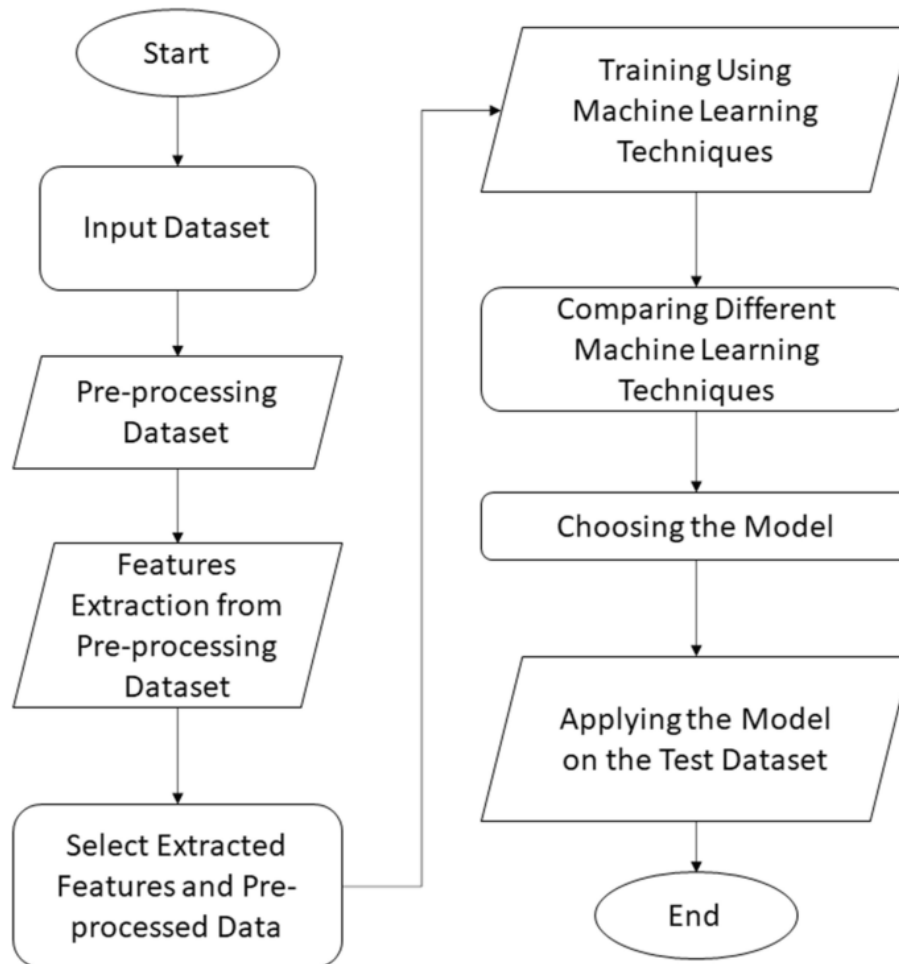


Figure 2.9: Flow Chart of development Methodology

Such robust experimental methodologies not only enhance the credibility and impact of forecasting research but also catalyze technological advancements in renewable energy. By adhering to rigorous experimental standards, researchers not only validate their findings but also pave the way for innovation, driving improvements in modeling approaches, data analysis methodologies, and forecasting algorithms.[21] This commitment to methodological rigor lays a solid foundation for the ongoing development and refinement of renewable energy forecasting techniques, facilitating the transition to a sustainable and resilient energy future.

2.13 Research Gaps

Identifying research gaps in wind and solar power forecasting is essential for advancing the state of the art and addressing critical challenges and uncertainties in the field. Research

gaps may encompass areas where existing forecasting models exhibit limitations or fail to adequately capture complex phenomena, such as extreme weather events, spatial variability, and uncertainty quantification.[22] By pinpointing gaps in knowledge and methodologies, researchers can prioritize research efforts, allocate resources effectively, and develop innovative solutions to overcome barriers and drive progress in renewable energy forecasting. Addressing research gaps is key to enhancing forecasting accuracy, reliability, and applicability in real-world scenarios, supporting the transition to a sustainable energy future.

Chapter 3

The Methodology

Utilizing a structured development methodology is paramount when employing machine learning tools for the analysis and prediction of wind and solar power generation. This informed approach offers a systematic framework, guiding efforts from defining requirements to model development and validation. Adhering to these established procedures directs focus towards comprehensively understanding the operational context, facilitating the creation of robust predictive models within real-world constraints. Employing proven methodologies ensures rigor in this data-driven engineering endeavor. Figure 3.1 illustrates the overall application process through a detailed flowchart.



Figure 3.1: Methodology of the Final Year Project

3.1 Step 1: The statement of the problem

Many conventional methods for analyzing and predicting wind and solar power generation lack precision due to their reliance on simplistic models and limited data inputs, hindering

their effectiveness in real-world applications. There exists a pressing demand for advanced machine learning algorithms capable of accurately forecasting renewable energy generation patterns with minimal computational resources and without compromising prediction accuracy. This project seeks to bridge this gap by developing and implementing machine learning models tailored specifically for wind and solar power prediction. By leveraging state-of-the-art algorithms and optimizing them for embedded systems, we aim to create a scalable and efficient solution that can be deployed across diverse renewable energy installations. The detailed findings and insights derived from this endeavor are outlined in Section 1.2.

3.2 Step 2: The determination of objectives

This thesis endeavors to construct and assess a machine learning-based system for predicting wind and solar power generation utilizing cost-effective components. Essential metrics such as prediction accuracy, computational efficiency, and adaptability to varying environmental conditions will be meticulously evaluated. Once the system meets predetermined performance benchmarks, practical demonstrations will highlight its utility and potential applications in the renewable energy sector. Further elaboration on the findings and implications of this research is provided in Section 1.3.

3.3 Step 3: The process of reviewing the literature

The research will commence by exploring the fundamental development of the system architecture for analyzing and predicting wind and solar power generation. This will involve examining key components such as data collection sensors and processing units, focusing on their integration and functionality. Subsequently, a comprehensive review of machine learning algorithms suitable for predictive modeling in renewable energy systems will be conducted. Special attention will be given to optimizing algorithm performance for resource-constrained environments. The objective is to contribute to advancing the field of renewable energy forecasting by addressing current research gaps. The comprehensive findings and insights gleaned from this endeavor are elaborated upon in Section 2.

3.4 Step 4: The formulation of methodology

The field of renewable energy analysis and prediction has witnessed significant progress, notably with the integration of machine learning tools. A prominent challenge within this domain revolves around accurately forecasting wind and solar power generation. This research endeavors to address this challenge by crafting a comprehensive methodology tailored for the analysis and prediction of renewable energy outputs.

The methodology encompasses several pivotal stages, commencing with the meticulous development and assembly of specialized hardware components optimized for efficient data collection and processing. Each phase is meticulously planned and executed to ensure the final product is not only capable of precise prediction but also operates with efficiency and reliability.

The initial phase of the methodology prioritizes the optimization of the hardware setup, encompassing sensors and data acquisition modules, to facilitate the collection

of real-time data crucial for predictive modeling. Subsequent stages revolve around the development and implementation of machine learning algorithms specifically tailored to forecast renewable energy production.

These algorithms leverage the collected data to discern intricate patterns and trends, empowering precise predictions regarding wind and solar power generation. Rigorous testing and validation protocols are integrated throughout the process to guarantee the accuracy and reliability of the predictive models.

By harnessing the potential of machine learning tools and adhering to a systematic methodology, this research aims to contribute significantly to the advancement of renewable energy analysis and prediction, ultimately setting new benchmarks within the field.

Chapter 3 delves into the topic in detail, while Section 2.12 explores the associated findings.

3.5 Step 5: The process of Development

The Development process for harnessing machine learning tools in the analysis and prediction of wind and solar power generation involves several key steps. Firstly, a comprehensive problem definition is established to delineate the scope and objectives of the study, identifying the specific aspects of wind and solar power generation to be addressed. Subsequently, an extensive literature review is conducted to explore existing research on machine learning applications in renewable energy analysis, uncovering relevant methodologies and research gaps. Following this, data collection and preparation ensue, involving the gathering of pertinent data sets on wind and solar power generation and ensuring their quality through cleaning and preprocessing. The algorithm selection phase entails evaluating various machine learning algorithms suitable for analyzing and predicting renewable energy outputs.

Model development proceeds by constructing and training machine learning models using the selected algorithms and prepared data sets. Validation and testing are then performed to assess the models' performance and reliability. Finally, performance evaluation and interpretation of results conclude the Development process, providing insights into the effectiveness of the applied machine learning tools and suggesting future research directions.

Refine Problem Definition and Constraints

In the initial phase of the research, refining the problem definition and constraints emerges as a pivotal step. It involves precisely defining the scope and objectives of the study, including identifying the specific aspects of wind and solar power generation to be analyzed and predicted through machine learning tools. Establishing a clear understanding of the research goals and the specific challenges to be addressed is crucial. This clarity enables focused efforts and facilitates the formulation of effective strategies to tackle the complexities inherent in forecasting renewable energy outputs, thereby laying a solid foundation for the research endeavor.

Research and Investigate

Exploring and investigating through a comprehensive literature review aids in comprehending the current research landscape concerning machine learning applications in renewable

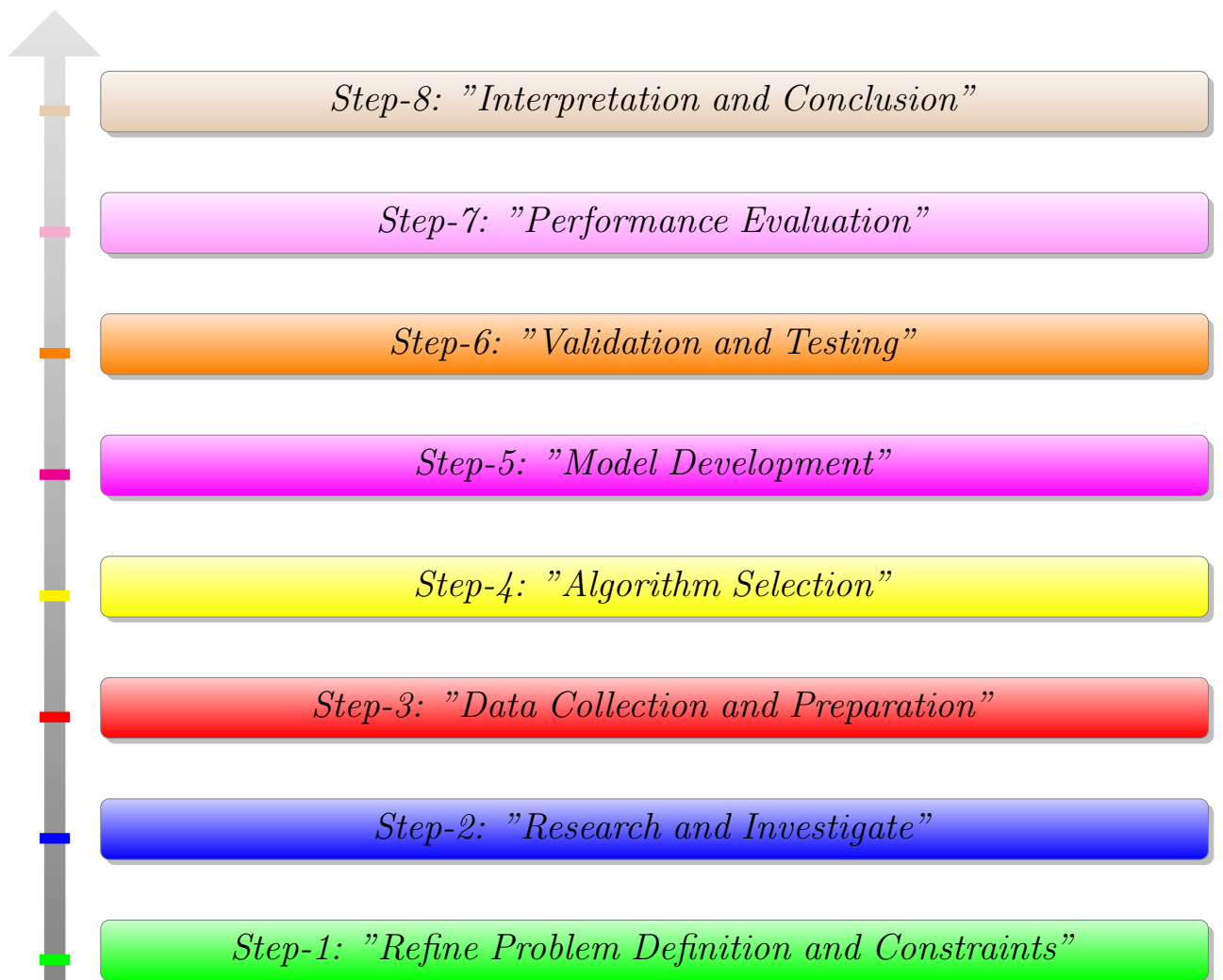


Figure 3.2: An Exemplary Development Journey

energy analysis. This process entails identifying pertinent methodologies, algorithms, and research gaps to augment existing knowledge and offer fresh insights to the field. By delving into the existing literature, researchers can gain valuable insights into the current state of machine learning applications in renewable energy analysis, identify areas for improvement or further investigation, and contribute to advancing the field's understanding and capabilities. This exploration serves as a foundation for subsequent research efforts, guiding the direction and focus of the study.

Data Collection and Preparation

Gathering relevant data sets on wind and solar power generation is essential for training and testing machine learning models. This involves collecting historical weather data, solar irradiance measurements, wind speed records, etc. The data must then be cleaned and preprocessed to ensure its quality and suitability for machine learning analysis.

Algorithm Selection

Critical to the research process is the evaluation of diverse machine learning algorithms to identify the most fitting ones for analyzing and predicting wind and solar power generation. Key factors including model complexity, interpretability, accuracy, and computational efficiency must be meticulously weighed during the algorithm selection process. By conducting thorough evaluations, researchers can discern which algorithms align best with the research objectives and constraints. This decision-making process ensures that the selected algorithms not only meet the analytical requirements but also offer practical viability and computational efficiency. Ultimately, the judicious selection of machine learning algorithms forms the backbone of the predictive modeling framework, laying the groundwork for accurate and effective analysis of renewable energy generation.

Model Development

Developing and training machine learning models encompasses utilizing chosen algorithms and curated datasets. This stage entails experimentation with diverse model architectures, hyperparameters, and feature engineering techniques to enhance model performance and ensure precise prediction. By exploring various configurations and adjustments, researchers can optimize the models' accuracy and efficiency in analyzing and predicting renewable energy generation. This iterative process allows for refinement and enhancement, ultimately leading to the development of robust and effective machine learning models for renewable energy analysis.

Validation and Testing

Validating the developed models using appropriate techniques, such as cross-validation or holdout validation, is necessary to ensure their reliability and generalization performance. Testing the models on unseen data sets helps assess their predictive accuracy and identify any potential issues or biases.

Performance Evaluation

Evaluating the performance of the developed models involves assessing predefined metrics such as prediction accuracy, precision, recall, RMSE, MAE, etc. This step helps in quantifying the effectiveness of the applied machine learning tools for wind and solar power generation analysis and prediction.

Interpretation and Conclusion

Interpreting the results of the analysis and drawing conclusions regarding the effectiveness of the applied machine learning tools is essential. Discussing the implications of the findings and suggesting areas for future research and improvement helps in providing insights for further advancements in the field.

3.6 Step 6: The Reporting

In the reporting phase, the thesis extensively chronicles the entire project journey, spanning from the initial research and development phases to the final stages of testing and evaluation. It meticulously outlines problem specifications, research methodologies, chosen solutions, prototyping efforts, testing procedures, results analysis, and any subsequent redesigns. The report emphasizes the project's significant contributions to the field of renewable energy analysis and prediction through the application of machine learning tools. Specifically, it aims to provide a transparent showcase of the rigorous methodology employed and the innovative approaches utilized in successfully developing predictive models for wind and solar power generation.

Chapter 4

The Process of Development

The development process described in the preceding chapter has been applied in this project and is depicted in Figure 4.1. The development process for integrating machine learning tools into the analysis and prediction of wind and solar power generation initiates with a meticulous problem definition, elucidating the research objectives and associated challenges. Subsequently, an exhaustive literature review is conducted to identify pertinent methodologies and bridge existing research gaps. Data collection ensues, involving the procurement of relevant datasets which are then meticulously cleansed and preprocessed.

Algorithm selection follows, entailing the evaluation of diverse options based on criteria such as accuracy and complexity. Model development progresses by iteratively training and refining chosen algorithms using prepared datasets. Rigorous validation and testing procedures are undertaken to ensure model reliability and generalization. Performance evaluation gauges predictive accuracy via predefined metrics. Lastly, the interpretation of results and formulation of conclusions underscore the effectiveness of the applied machine learning tools, with insightful discussions on implications and future research directions. This structured methodology aims to propel advancements in renewable energy analysis and prediction.

4.1 Refine Problem Definition and Constraints

4.1.1 Problem Definition

Unreliable wind and solar forecasts disrupt grid balance and stability. Accurate predictions are crucial for optimal resource allocation and voltage control. Advanced forecasting models and optimized methodologies are needed to address these challenges. To find a solution to this problem, we first need to specify the problems.

- **Forecasting Imbalances:** Inaccurate forecasts of wind and solar power generation can lead to imbalances in the energy grid, affecting supply-demand equilibrium.
- **Grid Stability Challenges:** Intermittent nature of wind and solar energy production poses challenges for maintaining grid stability and reliability.
- **Allocation Hindrance:** Lack of precise predictions for wind and solar power generation hinders optimal energy resource allocation and management.
- **Voltage Fluctuations:** Voltage fluctuations and power quality issues arise due to the non-controllable characteristics of wind and solar production.

- **Uncertain Reserves:** Difficulty in estimating reserves and scheduling power system operations due to uncertain renewable energy outputs.
- **Congestion Management:** Congestion management in the power grid becomes challenging without accurate forecasts of wind and solar power generation.
- **Storage Efficiency:** Effective power storage management relies on precise predictions of renewable energy generation patterns.
- **Trading Inefficiency:** Inefficient trading of produced power in the electricity market due to unreliable forecasts can lead to increased costs.
- **Model Availability:** Limited availability of advanced forecasting models tailored for wind and solar power generation.
- **Understanding Gaps:** Lack of comprehensive understanding of the relationship between weather conditions and renewable energy production.
- **Methodology Optimization:** Absence of optimized methodologies for integrating machine learning techniques into wind and solar power prediction.
- **Data Reliability:** Need for reliable data sources and preprocessing techniques to enhance the accuracy of predictive models for renewable energy generation.

Precise problem specification is essential for efficiency and resource conservation. It ensures a targeted approach, minimizing ambiguity and directing efforts where they are needed most. Beyond defining the issue, it contextualizes its significance, outlining its repercussions and desired outcomes. Ultimately, problem specification serves as a roadmap, guiding the development of effective solutions by illuminating the problem's scope, context, and potential resolutions.

4.1.2 Constraints

- **Data Availability:** Limited availability of high-quality data for training and testing machine learning models poses a significant constraint.
- **Computational Resources:** High computational requirements for training complex machine learning algorithms limit model scalability and efficiency.
- **Model Interpretability:** Complex machine learning models such as random forests may lack interpretability, hindering insights into the underlying factors influencing predictions.
- **Overfitting:** Risk of overfitting exists, especially with decision tree-based models, leading to poor generalization on unseen data.
- **Feature Selection:** Identifying and selecting relevant features from a pool of variables requires domain expertise and may impact model performance.
- **Hyperparameter Tuning:** Optimizing hyperparameters for machine learning algorithms can be time-consuming and resource-intensive.

- **Model Complexity:** Increasing model complexity to improve accuracy may lead to diminished interpretability and longer training times.
- **Data Imbalance:** Imbalanced datasets, where one class is significantly more prevalent than others, can bias model predictions and affect performance.

4.2 Research and Investigate

The presented research conducted in Chapter 2, delves into the critical domain of predicting wind and solar power generation using machine learning methodologies, an area of increasing significance in the renewable energy sector. The study effectively highlights the escalating share of renewable energy, particularly wind and solar, in the energy portfolio, emphasizing the challenges posed by their intermittent nature. By elucidating the necessity of accurate forecasting for grid stability, reserve estimation, and optimal energy management, it underscores the pivotal role of predictive models in mitigating operational complexities in the power sector. Moreover, meticulous details are provided regarding the data collection process, pre-processing techniques, and the application of three distinct machine learning algorithms—Linear Regression, Decision Tree Regression, and Random Forest Regression. The results portray commendable accuracy in predicting solar and wind power generation, validated through performance metrics such as R^2 values. However, the discussion extends beyond mere model performance, delving into feature selection, hyperparameter tuning, and future research directions, thereby offering a comprehensive understanding of the predictive modeling landscape in renewable energy forecasting.

4.2.1 Development Variables

- Solar power generation, in MWh;
- Wind power generation, in MWh;
- Solar installed capacity, in MW;
- Wind installed capacity, in MW;
- Windspeed at 10 meters above ground, in m/s;
- Direct horizontal radiation, in W/m^2 ;
- Diffuse horizontal radiation, in W/m^2 ;
- Top-of-the-atmosphere solar irradiance, in W/m^2 ;
- Surface solar irradiance, in W/m^2 ;
- Air density at ground level, in kg/m^3 ;
- Precipitation, in mm/hour;
- Air temperature 2 meters above ground, in $^{\circ}C$;
- Snowfall, in mm/hour;

- Snow mass, in kg/m²;
- Cloud cover fraction, a $[0, 1]$ scale.

Development variables encompass solar and wind power generation, installed capacity, wind speed, radiation levels, air density, precipitation, temperature, snowfall, snow mass, and cloud cover. These factors are crucial for understanding and predicting renewable energy production dynamics.

4.2.2 Algorithms

Linear Regression

Linear regression is a statistical method used to model and analyze the relationship between two variables: a dependent variable (also called the response or target) and one or more independent variables (also called predictors or features). The goal is to find the best-fitting straight line (or hyperplane in higher dimensions) that represents this relationship.

Key Concepts in Linear Regression

Simple Linear Regression

- Models the relationship between a single independent variable (x) and the dependent variable (y).
- The equation of the regression line is:

$$y = \beta_0 + \beta_1 x + \epsilon$$

where:

- β_0 : Intercept (value of y when $x = 0$).
- β_1 : Slope (rate of change of y with x).
- ϵ : Error term (difference between observed and predicted values).

Multiple Linear Regression

- Extends simple linear regression to include multiple independent variables (x_1, x_2, \dots, x_n).
- The equation is:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n + \epsilon$$

Assumptions

- **Linearity:** The relationship between predictors and the dependent variable is linear.
- **Independence:** Observations are independent of each other.
- **Homoscedasticity:** The variance of residuals is constant across all levels of the independent variable(s).
- **Normality:** Residuals are normally distributed.

Fitting the Model

- The parameters $(\beta_0, \beta_1, \dots, \beta_n)$ are estimated using the **least squares method**, minimizing the sum of squared residuals:

$$\text{RSS} = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

where y_i is the actual value and \hat{y}_i is the predicted value.

Linear regression is foundational in machine learning and statistics, serving as a stepping stone for more complex models like logistic regression and polynomial regression.

Random Forest Regression

Random forest regression is a machine learning algorithm that extends decision tree regression by combining multiple trees (an ensemble) to improve the accuracy and robustness of predictions. It is a non-linear model, well-suited for handling complex datasets with interactions and non-linear relationships.

Key Concepts of Random Forest Regression

Ensemble Learning

- Random forest is an ensemble method that uses multiple decision trees.
- Each tree makes its own prediction, and the final prediction is obtained by averaging (for regression) the predictions of all trees.

Bootstrap Aggregation (Bagging)

- Each tree is trained on a random sample (with replacement) of the dataset.
- This reduces overfitting and variance compared to a single decision tree.

Feature Randomness

- At each split in a tree, a random subset of features is considered.
- This introduces diversity among the trees and reduces correlation between them.

Regression Output

- For a given input, each tree outputs a prediction.
- The final output of the random forest is the mean of the predictions from all trees:

$$\hat{y} = \frac{1}{T} \sum_{t=1}^T \hat{y}_t$$

where T is the total number of trees, and \hat{y}_t is the prediction from the t -th tree.

Steps in Building a Random Forest Regressor

Data Preparation

- Split the dataset into training and testing subsets.
- Normalize or scale features if necessary.

Train the Model

- Specify the number of trees (T), maximum depth, and the number of features to consider at each split.
- Train each tree on a bootstrap sample of the training data.

Make Predictions

- For a given input, pass it through each tree in the forest.
- Average the outputs from all trees to get the final prediction.

Evaluate the Model

- Use metrics like Mean Squared Error (MSE), Root Mean Squared Error (RMSE), or Mean Absolute Error (MAE) to assess performance.

Random forest regression is a versatile and powerful tool, often used when the relationship between input features and the target variable is complex or non-linear.

Decision Tree Regression

Decision Tree Regression is a machine learning algorithm that models the relationship between input features and a target variable using a tree-like structure. It partitions the dataset into smaller subsets based on feature splits, recursively dividing the data until certain stopping criteria are met. The final prediction is made by averaging the target values within each leaf node.

Key Concepts of Decision Tree Regression

Tree Structure

A decision tree consists of:

- **Root Node:** Represents the entire dataset and the starting point for splits.
- **Internal Nodes:** Represent decisions made based on feature values.
- **Leaf Nodes:** Represent the final predictions (average of target values in regression).

Splitting Criterion

The algorithm selects splits based on a metric that minimizes the variance in the target variable within the resulting subsets:

$$\text{Variance Reduction} = \text{Variance}_{\text{parent}} - \sum \left(\frac{\text{size}(\text{child})}{\text{size}(\text{parent})} \cdot \text{Variance}_{\text{child}} \right)$$

Recursive Partitioning

The dataset is split recursively, with each step aiming to reduce the variance in the target variable as much as possible.

Prediction

For a given input, the tree traverses from the root to a leaf node based on feature values, and the predicted value is the average of target values in that leaf node.

Steps in Building a Decision Tree Regressor

Data Preparation

- Split the dataset into training and testing subsets.

Train the Model

- Start at the root node.
- Select the best feature and threshold to split the dataset.
- Recursively split the data until a stopping criterion is met (e.g., maximum depth, minimum samples per leaf).

Make Predictions

- Pass the input through the tree by following the decision rules at each node.
- Output the average target value of the leaf node reached.

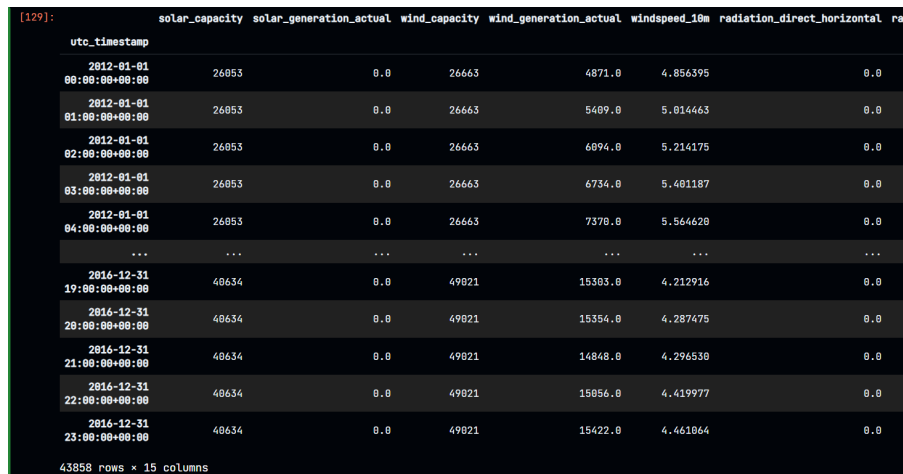
Evaluate the Model

- Use metrics like Mean Squared Error (MSE), Root Mean Squared Error (RMSE), or Mean Absolute Error (MAE) to assess performance.

Decision Tree Regression is a supervised learning algorithm that predicts continuous target variables by recursively splitting the dataset based on feature values to minimize variance within partitions. It is easy to interpret and handles non-linear relationships but can overfit the data if the tree grows too deep.

4.3 Data Collection and Preparation

In the data cleaning process for focusing on the analysis and prediction of wind and solar power generation, several crucial steps are undertaken to ensure the integrity and suitability of the dataset for machine learning analysis. Initially, comprehensive time-series data on wind and solar power production, alongside capacity, are collected for Germany. Additionally, weather data relevant for power system modeling, obtained at hourly resolution and aggregated from NASA MERRA-2 reanalysis, is incorporated. These datasets, encompassing variables such as temperature, precipitation, snowfall, cloud cover, air density, windspeed at 10m height (windspeed 10 m), radiation (both direct and diffuse) at horizontal surfaces, and irradiance at the Earth's surface and top of the atmosphere (irradiance surface and irradiance toa), are combined and prepared for cleaning and subsequent analysis. Another crucial step entails outlier detection and removal to mitigate their potential impact on analysis and modeling outcomes. Through the use of Seaborn, boxplots or scatterplots are generated to visually identify outliers. Subplots are particularly useful for comparing variable distributions before and after outlier removal, facilitating a comprehensive assessment of data integrity. One pivotal aspect of data cleaning involves handling missing values effectively. Here is the Final output of Data cleaning process where 43858 rows \times 15 columns are present and no missing value.



```
[129]:
```

utc_timestamp	solar_capacity	solar_generation_actual	wind_capacity	wind_generation_actual	windspeed_10m	radiation_direct_horizontal	rad
2012-01-01 00:00:00+00:00	26953	0.0	26663	4871.0	4.856395	0.0	
2012-01-01 01:00:00+00:00	26953	0.0	26663	5409.0	5.814463	0.0	
2012-01-01 02:00:00+00:00	26953	0.0	26663	6894.0	5.214175	0.0	
2012-01-01 03:00:00+00:00	26953	0.0	26663	6734.0	5.461187	0.0	
2012-01-01 04:00:00+00:00	26953	0.0	26663	7378.0	5.564628	0.0	
...
2016-12-31 19:00:00+00:00	48634	0.0	49821	15383.0	4.212916	0.0	
2016-12-31 20:00:00+00:00	48634	0.0	49821	15354.0	4.287475	0.0	
2016-12-31 21:00:00+00:00	48634	0.0	49821	14848.0	4.296538	0.0	
2016-12-31 22:00:00+00:00	48634	0.0	49821	15856.0	4.419977	0.0	
2016-12-31 23:00:00+00:00	48634	0.0	49821	15422.0	4.461864	0.0	

43858 rows x 15 columns

Figure 4.1: Final output of Data cleaning process

Techniques such as imputation or deletion are employed to address missing values, ensuring data completeness. The Seaborn library proves invaluable in visualizing the distribution of missing values across various variables, aiding in informed decision-making regarding their treatment.

Another crucial step entails outlier detection and removal to mitigate their potential impact on analysis and modeling outcomes. Through the use of Seaborn, boxplots or scatterplots are generated to visually identify outliers. Subplots are particularly useful for comparing variable distributions before and after outlier removal, facilitating a comprehensive assessment of data integrity.

Lastly, the dataset is split into training and testing sets for model development and evaluation. Subplots enable the visualization of variable distributions in both sets, ensuring they are representative of the overall dataset. By adhering to these steps and leveraging tools like the Seaborn library for visualization and analysis, the data cleaning process

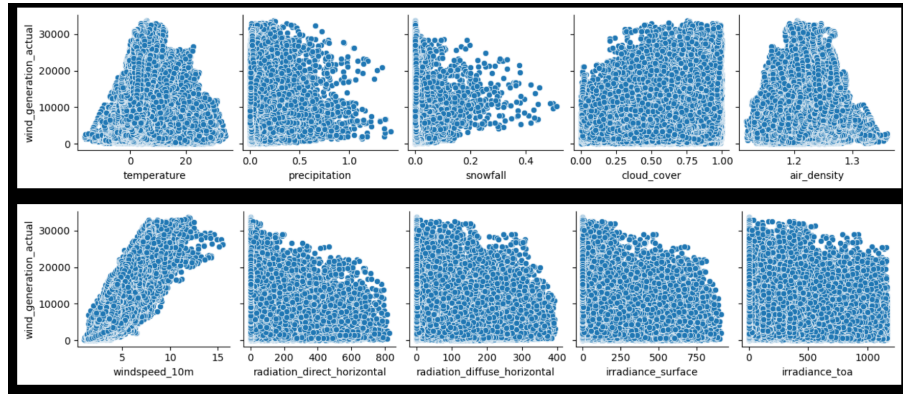


Figure 4.2: The Generation of Wind power (MWh) vs. Weather data features

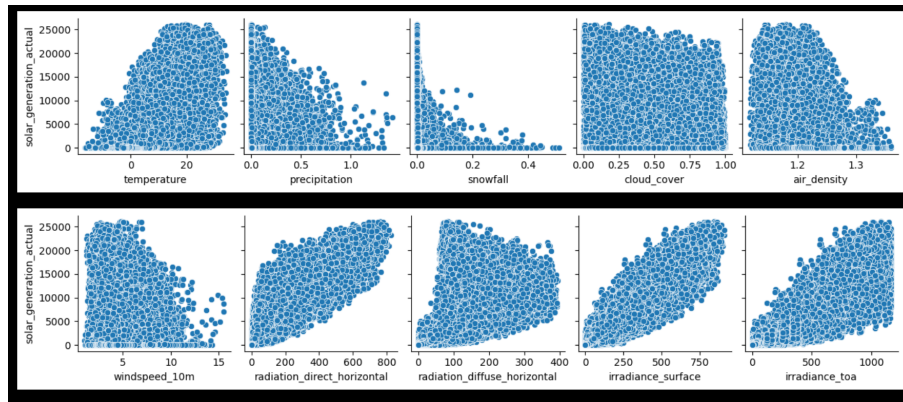


Figure 4.3: The Generation of Solar power (MWh) vs. Weather data features

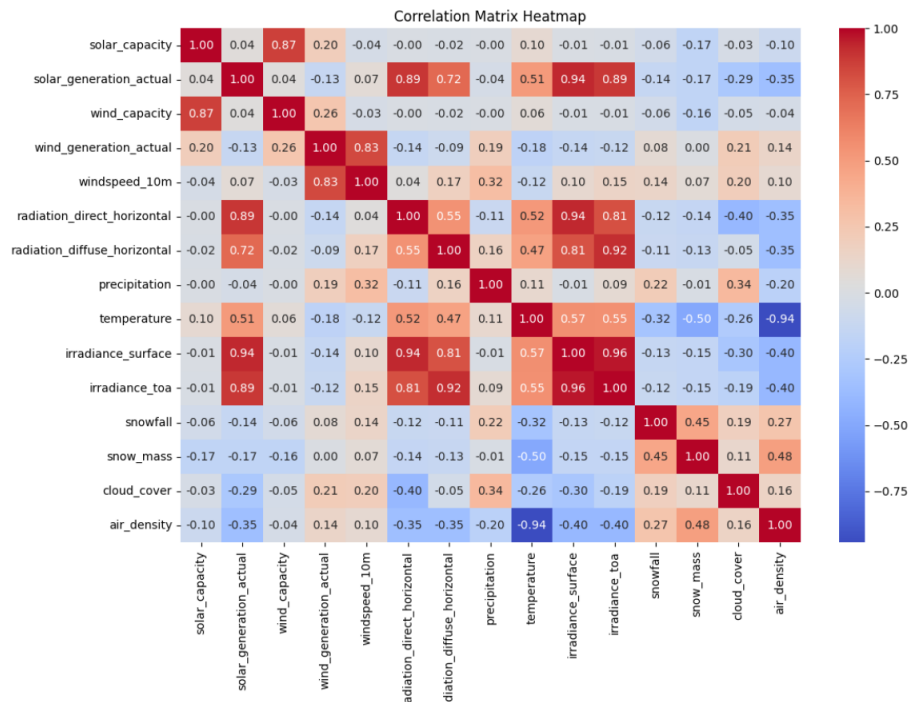


Figure 4.4: Correlation Matrix Heatmap

ensures the dataset is well-prepared for subsequent machine learning analysis and modeling endeavors.

4.4 Solar Power Generation Prediction Using Weather Data

Cross-Validation

To evaluate the performance of an algorithm, we use a technique called cross-validation (commonly abbreviated as CV). In k -fold cross-validation, the dataset is divided into k smaller subsets, known as **folds**. The model is trained on $k - 1$ of these folds, and its performance is evaluated on the remaining fold. This process is repeated k times, with each fold used once as the validation set. The final performance metric is the average of the performance metrics calculated in each iteration.

In the code, the ‘`cross_val_score`’ function from `sklearn.model_selection` is employed, with the number of folds set to $k = 5$. This ensures a robust evaluation of the model.

For the `LinearRegression` model, the default performance metric is the coefficient of determination, R^2 . This metric measures how well the predictions align with the actual values, representing the proportion of variance in the target variable that is explained by the model.

Predicting Solar power generation using features

1.1 Training and validation of the models for all features (n=10)

The following features were selected for training and testing the models:

```
selected_features = ['solar_capacity',
                    'windspeed_10m',
                    'radiation_direct_horizontal',
                    'radiation_diffuse_horizontal',
                    'irradiance_surface',
                    'irradiance_toa',
                    'precipitation',
                    'snowfall',
                    'cloud_cover',
                    'air_density']
X_train = X_train_master[selected_features]
X_test = X_test_master[selected_features]
scores = cross_val_score(X_train, y_train, cv=5)
```

The training set (X_{train}) and test set (X_{test}) are defined using the selected features, and the model is validated with 5-fold cross-validation. The average R^2 score provides insights into the model’s predictive accuracy.

1.2 Training and validation of the models for selected features (n=7)

A reduced set of 7 features is used for training and validation:

```
selected_features = ['solar_capacity',
                    'windspeed_10m',
```

```
'irradiance_surface',  
'precipitation',  
'snowfall',  
'cloud_cover',  
'air_density']
```

This helps analyze the model's performance when fewer features are included, allowing a comparison with the full feature set.

1.3 Training and validation of the models for selected feature (n=1)

In this case, only one feature, 'irradiance surface', is selected for training and validation:

```
selected_features = ['irradiance_surface']
```

The models for predicting solar power generation are trained and validated using different sets of features. In the first case, 10 features are used, with the model validated through 5-fold cross-validation to assess predictive accuracy via the average R^2 score. In the second case, a reduced set of 7 features is used to analyze performance with fewer inputs. Finally, the model is tested with a single feature, "irradiance surface," for comparison.

Predicting solar power generation using the LinearRegression model

```
# First, we train the model for the prediction for solar power  
                                generation based on weather data.  
  
lr = LinearRegression()  
scores = cross_val_score(lr, X_train, y_train, cv=5)  
print(f"The average score for the LinearRegression model (training) is:  
      %0.3f" % np.mean(scores))  
  
# Second, we validate the model for the data.  
lr.fit(X_train, y_train)  
predictions_lr = lr.predict(X_test)  
r2 = r2_score(y_test, predictions_lr)  
print(f"The R2 score of the LinearRegression model (test) is: %0.3f" %  
      r2)
```

The code uses a Linear Regression model to predict a target variable based on selected features from the dataset. First, it evaluates the model's performance with cross-validation on the training data, calculating an average score. Then, it fits the model to the training data and makes predictions on the test data. The R^2 score is computed to assess the model's performance on unseen data. The selected features, including solar capacity, windspeed, radiation, and weather-related variables, are used for training and testing the model.

Predicting solar power generation using the DecisionTreeRegression model

```
# First, we train the model for the prediction for solar power  
                                generation based on weather data.  
  
dt = DecisionTreeRegressor()
```



```

scores = cross_val_score(dt, X_train, y_train, cv=5)
print(f"The average score for the DecisionTreeRegression model (
                                training) is: %0.3f" % np.mean(
                                scores))

# Second, we validate the model for the data.
dt.fit(X_train, y_train)
predictions_dt = dt.predict(X_test)
r2 = r2_score(y_test, predictions_dt)
print(f"The R2 score of the DecisionTreeRegression model (test) is: %0.
3f" % r2)

```

The code uses a `DecisionTreeRegressor` and performs 5-fold cross-validation on the training data to evaluate model performance. The average score from cross-validation is printed, reflecting the model's ability to generalize. The model is then trained on the full training data (`dt.fit`), and predictions are made on the test set. The R^2 score of the model on the test data is calculated using `r2_score`, indicating the proportion of variance explained by the model on unseen data.

Predicting solar power generation using the Random Forest Regression model

```

# Frist, we train the model for the prediction for solar power
                                generation based on weather data.

rf = RandomForestRegressor()
scores = cross_val_score(rf, X_train, y_train, cv=5)
print(f"The average score for the RandomForestRegression model (
                                training) is: %0.3f" % np.mean(
                                scores))

# Second, we validate the model for the data.
rf.fit(X_train, y_train)
predictions_rf = rf.predict(X_test)
r2 = r2_score(y_test, predictions_rf)
print(f"The R2 score of the RandomForestRegression model (test) is: %0.
3f" % r2)

```

The code uses a `RandomForestRegressor` to train and evaluate the model. It performs 5-fold cross-validation on the training data, printing the average score to assess model performance during training. The model is then fitted to the entire training set using `rf.fit`, and predictions are made on the test set with `rf.predict`. The R^2 score is calculated using `r2_score` to measure the model's performance on the test data, indicating how well it generalizes to unseen data.

4.5 Wind Power Generation Prediction Using Weather Data

Predicting wind power generation using features

1. Training and validation of the models for all features (n=10)

The following features were selected for training and testing the models:

```

selected_features = ['wind_capacity',
                    'windspeed_10m',
                    'radiation_direct_horizontal',
                    'radiation_diffuse_horizontal',
                    'irradiance_surface',
                    'irradiance_toa',
                    'precipitation',
                    'snowfall',
                    'cloud_cover',
                    'air_density']
X_train = X_train_master[selected_features]
X_test = X_test_master[selected_features]
scores = cross_val_score(X_train, y_train, cv=5)

```

The training (`X_train`) and test (`X_test`) sets are created using the selected features, and the model is evaluated through 5-fold cross-validation. The average R^2 score reflects the model's predictive performance.

2. Training and validation of the models for selected features (n=7)

A reduced set of 7 features is used for training and validation:

```

selected_features = ['wind_capacity',
                    'windspeed_10m',
                    'irradiance_surface',
                    'precipitation',
                    'snowfall',
                    'cloud_cover',
                    'air_density']

```

This helps analyze the model's performance with fewer features, enabling a comparison with the full feature set.

3. Training and validation of the models for selected feature (n=1)

In this case, only one feature, 'windspeed 10m', is selected for training and validation:

```

selected_features = ['windspeed_10m']

```

The models for predicting wind power generation are trained and validated using different feature sets. In the first scenario, the model is trained with 10 features and validated through 5-fold cross-validation to evaluate its predictive accuracy based on the average R^2 score. In the second scenario, a reduced set of 7 features is used to assess performance with fewer inputs. Finally, the model is tested with only a single feature, "windspeed 10m," to facilitate comparison.

Predicting wind power generation using the LinearRegression model

```

# Frist, we train the model for the prediction for wind power
                                generation based on weather data.
lr = LinearRegression()
scores = cross_val_score(lr, X_train, y_train, cv=5)
print(f"The average score for the LinearRegression model (training) is:
                                %0.3f" % np.mean(scores))

# Second, we validate the model for the data.

```

```

lr.fit(X_train, y_train)
predictions_lr = lr.predict(X_test)
r2 = r2_score(y_test, predictions_lr)
print(f"The R2 score of the LinearRegression model (test) is: %0.3f" %
      r2)

```

First, the model is trained to predict wind power generation using weather data. A `LinearRegression` model is used, and its performance is evaluated through 5-fold cross-validation on the training data. The average score is printed to assess the model's accuracy during training. Next, the model is validated on the test data by fitting it with `lr.fit` and making predictions using `lr.predict`. The R^2 score is calculated to evaluate the model's predictive accuracy on data.

Predicting wind power generation using the `DecisionTreeRegression` model

```

# Frist, we train the model for the prediction for wind power
# generation based on weather data.
dt = DecisionTreeRegressor()
scores = cross_val_score(dt, X_train, y_train, cv=5)
print(f"The average score for the DecisionTreeRegression model (
      training) is: %0.3f" % np.mean(
      scores))

# Second, we validate the model for the data.
dt.fit(X_train, y_train)
predictions_dt = dt.predict(X_test)
r2 = r2_score(y_test, predictions_dt)
print(f"The R2 score of the DecisionTreeRegression model (test) is: %0.
      3f" % r2)

```

Initially, a `DecisionTreeRegressor` model is trained to predict wind power generation based on weather data. The model's performance is assessed using 5-fold cross-validation on the training set, and the average score is displayed to indicate its effectiveness during training. Subsequently, the model is fitted to the training data with `dt.fit` and predictions are made on the test set using `dt.predict`. The model's performance on unseen data is then evaluated by calculating the R^2 score.

Predicting wind power generation using the `Random Forest Regression` model

```

# Frist, we train the model for the prediction for wind power
# generation based on weather data.
rf = RandomForestRegressor()
scores = cross_val_score(rf, X_train, y_train, cv=5)
print(f"The average score for the RandomForestRegression model (
      training) is: %0.3f" % np.mean(
      scores))

# Second, we validate the model for the data for.
rf.fit(X_train, y_train)
predictions_rf = rf.predict(X_test)
r2 = r2_score(y_test, predictions_rf)

```

```
print(f"The R2 score of the RandomForestRegression model (test) is: %0.3f" % r2)
```

First, a `RandomForestRegressor` model is trained to predict wind power generation based on weather data. To evaluate the model's training performance, 5-fold cross-validation is applied, and the average score is computed. This provides an indication of how well the model performs on different subsets of the training data. Next, the model is fitted to the training data using `rf.fit`, and predictions are generated for the test data. The model's accuracy on the test set is assessed by calculating the R^2 score.

4.6 Solar Power Generation Forecasting

```
solar['solar_generation_t+1'] = solar['solar_generation_actual'].shift(
    periods=1)
solar['solar_generation_t+2'] = solar['solar_generation_actual'].shift(
    periods=2)
solar['solar_generation_t+3'] = solar['solar_generation_actual'].shift(
    periods=3)
solar['solar_generation_t+4'] = solar['solar_generation_actual'].shift(
    periods=4)
solar['solar_generation_t+5'] = solar['solar_generation_actual'].shift(
    periods=5)
solar['solar_generation_t+6'] = solar['solar_generation_actual'].shift(
    periods=6)
```

The code creates columns in the `solar` DataFrame to store lagged values of `solar_generation_actual`, shifted forward by 1 to 6 periods, for forecasting future solar generation at $t + 1$ through $t + 6$.

Split dataset into training and test

```
from sklearn.model_selection import train_test_split
train, test = train_test_split(solar, test_size=.2, random_state=3)

X_train = train[['solar_generation_actual', 'solar_capacity', '
                windspeed_10m', '
                radiation_direct_horizontal', '
                radiation_diffuse_horizontal', '
                irradiance_surface', '
                irradiance_toa', 'precipitation', '
                snowfall', 'cloud_cover', '
                air_density']]

X_test = test[['solar_generation_actual', 'solar_capacity', '
               windspeed_10m', '
               radiation_direct_horizontal', '
               radiation_diffuse_horizontal', '
               irradiance_surface', '
               irradiance_toa', 'precipitation', '
               snowfall', 'cloud_cover', '
               air_density']]
```

The code splits the `solar` DataFrame into training and testing datasets, using an 80-20 split. It selects specific features (e.g., solar generation, windspeed, radiation, etc.)

for both training (`X_train`) and testing (`X_test`) datasets. The target variable (`y_train` and `y_test`) depends on the forecast horizon, such as $t + 1$, $t + 2$, etc.

Training and validation of the models for selected features

1. Training and validation of the models for all features (n=11)

The following features were selected for training and testing the models:

```
X_train = train[['solar_generation_actual', 'solar_capacity', '
                windspeed_10m', '
                radiation_direct_horizontal', '
                radiation_diffuse_horizontal', '
                irradiance_surface', '
                irradiance_toa', 'precipitation', '
                snowfall', 'cloud_cover', '
                air_density']]

# y_train will depend on the duration considered for the forecast (see
# below). It could be: t+1h, t+2h, t+
# 3h, etc.

X_test = test[['solar_generation_actual', 'solar_capacity', '
                windspeed_10m', '
                radiation_direct_horizontal', '
                radiation_diffuse_horizontal', '
                irradiance_surface', '
                irradiance_toa', 'precipitation', '
                snowfall', 'cloud_cover', '
                air_density']]

# y_test will also depend on the duration considered for the forecast (
# see below). It could be: t+1h, t+2h
# , t+3h, etc.
```

The code selects features such as solar generation, capacity, windspeed, radiation, precipitation, and air density from the `train` and `test` datasets to create `X_train` and `X_test` for training and testing machine learning models. The target variables, `y_train` and `y_test`, represent the solar generation forecasts for future time horizons such as $t + 1$, $t + 2$, $t + 3$, etc., depending on the prediction duration. This setup prepares the data for forecasting solar power generation based on historical and environmental factors.

2. Training and validation of the models for selected features (n=8)

A reduced set of 8 features is used for training and validation:

```
X_train = train[['solar_generation_actual', 'solar_capacity', '
                windspeed_10m', 'irradiance_surface',
                'precipitation', 'snowfall', '
                cloud_cover', 'air_density']]

X_test = test[['solar_generation_actual', 'solar_capacity', '
                windspeed_10m', 'irradiance_surface',
                'precipitation', 'snowfall', '
                cloud_cover', 'air_density']]
```

This helps analyze the model's performance with fewer features, enabling a comparison with the full feature set.

3. Training and validation of the models for selected feature (n=2)

```
X_train = train[['solar_generation_actual', 'irradiance_surface']]

X_test = test[['solar_generation_actual', 'irradiance_surface']]
```

The code creates `X_train` and `X_test` by selecting only two features: `solar_generation_actual` (actual solar generation) and `irradiance_surface` (surface solar irradiance) from the `train` and `test` datasets. These features are used for training and testing machine learning models focused on solar power forecasting.

Using the LinearRegression model to forecast solar generation.

```
# we train and validate the LinearRegression for all forecast durations
#                               (1 to 6h)
lr = LinearRegression()
for i in range(len(y_list)):
    # training of the model via cross-validation
    scores_solar = cross_val_score(lr, X_train, train[y_list[i]], cv=5)
    print(f"\nThe average score linear regression for t+{i+1}h is: %0.3f" % np.mean(scores_solar))

    # validation of the model
    lr.fit(X_train, train[y_list[i]])
    predictions_lr = lr.predict(X_test)
    r2 = r2_score(test[y_list[i]], predictions_lr)
    print(f"The R2 score of the linear regression model for t+{i+1}h is
          r2 = %0.3f" % r2)
```

The code trains and validates a `LinearRegression` model for forecast durations ranging from 1 to 6 hours. Using cross-validation with 5 folds, it evaluates the model's performance during training and prints the average cross-validation score. After training, the model predicts solar generation on the test set, calculates the R^2 score for each forecast duration, and displays the results.

Using the RandomForestRegressor model to forecast solar generation.

```
rf = RandomForestRegressor()
for i in range(len(y_list)):
    # training of the model via cross-validation
    scores_solar = cross_val_score(rf, X_train, train[y_list[i]], cv=5)
    print(f"\nThe average score for random forest regression (100
          decisions trees) and t+{i+1}h
          is: %0.3f" % np.mean(
          scores_solar))

    # validation of the model
    rf.fit(X_train, train[y_list[i]])
    predictions_rf = rf.predict(X_test)
    r2 = r2_score(test[y_list[i]], predictions_rf)
    print(f"The R2 score of the random forest regression (with 100
          decision trees) for t+{i+1}h is
          r2 = %0.3f" % r2)
```

The code trains and validates a `RandomForestRegressor` model with 100 decision trees for forecast durations from 1 to 6 hours. Cross-validation with 5 folds evaluates the model's performance during training, and the average cross-validation score is printed. After training, the model predicts solar generation on the test set, calculates the R^2 score for each forecast duration, and displays the results.

4.7 Wind Power Generation Forecasting

```
wind['wind_generation_t+1'] = wind['wind_generation_actual'].shift(
    periods=1)
wind['wind_generation_t+2'] = wind['wind_generation_actual'].shift(
    periods=2)
wind['wind_generation_t+3'] = wind['wind_generation_actual'].shift(
    periods=3)
wind['wind_generation_t+4'] = wind['wind_generation_actual'].shift(
    periods=4)
wind['wind_generation_t+5'] = wind['wind_generation_actual'].shift(
    periods=5)
wind['wind_generation_t+6'] = wind['wind_generation_actual'].shift(
    periods=6)
```

The code creates columns in the `wind` DataFrame to store lagged values of `wind_generation_actual`, shifted forward by 1 to 6 periods, for forecasting future wind generation at $t + 1$ through $t + 6$.

Split dataset into training and test

```
from sklearn.model_selection import train_test_split
train, test = train_test_split(wind, test_size=.2, random_state=3)

X_train = train[['wind_generation_actual', 'wind_capacity', '
                windspeed_10m', '
                radiation_direct_horizontal', '
                radiation_diffuse_horizontal', '
                irradiance_surface', '
                irradiance_toa', 'precipitation', '
                snowfall', 'cloud_cover', '
                air_density']]
# y_train will depend on the duration considered for the forecast (see
# below). It could be: t+1h, t+2h, t+
# 3h, etc.

X_test = test[['wind_generation_actual', 'wind_capacity', 'windspeed_10m',
               'radiation_direct_horizontal', '
               radiation_diffuse_horizontal', '
               irradiance_surface', '
               irradiance_toa', 'precipitation', '
               snowfall', 'cloud_cover', '
               air_density']]
# y_test will also depend on the duration considered for the forecast (
# see below). It could be: t+1h, t+2h
# , t+3h, etc.
```

The code splits the `wind` DataFrame into training and testing datasets with an 80-20 split. It selects relevant features, such as wind generation, windspeed, radiation, etc., for both the training (`X_train`) and testing (`X_test`) datasets. The target variable (`y_train` and `y_test`) is based on the forecast horizon, such as $t + 1$, $t + 2$, etc.

Training and validation of the models for selected features

1. Training and validation of the models for all features (n=11)

The following features were selected for training and testing the models:

```
X_train = train[['wind_generation_actual', 'wind_capacity', 'windspeed_10m', 'radiation_direct_horizontal', 'radiation_diffuse_horizontal', 'irradiance_surface', 'irradiance_toa', 'precipitation', 'snowfall', 'cloud_cover', 'air_density']]

X_test = test[['wind_generation_actual', 'wind_capacity', 'windspeed_10m', 'radiation_direct_horizontal', 'radiation_diffuse_horizontal', 'irradiance_surface', 'irradiance_toa', 'precipitation', 'snowfall', 'cloud_cover', 'air_density']]
```

The code extracts features such as wind generation, capacity, windspeed, radiation, precipitation, and air density from the `train` and `test` datasets to form `X_train` and `X_test` for training and testing machine learning models. The target variables, `y_train` and `y_test`, represent wind generation predictions for future time periods such as $t + 1$, $t + 2$, $t + 3$, etc., depending on the forecast horizon. This configuration prepares the data for predicting wind power generation based on historical and environmental conditions.

2. Training and validation of the models for selected features (n=8)

A reduced set of 8 features is used for training and validation:

```
X_train = train[['wind_generation_actual', 'wind_capacity', 'windspeed_10m', 'irradiance_surface', 'precipitation', 'snowfall', 'cloud_cover', 'air_density']]

X_test = test[['wind_generation_actual', 'wind_capacity', 'windspeed_10m', 'irradiance_surface', 'precipitation', 'snowfall', 'cloud_cover', 'air_density']]
```

This allows for the evaluation of the model's performance with a reduced set of features, facilitating a comparison with the model's performance using the complete feature set.

3. Training and validation of the models for selected feature (n=2)

```
X_train = train[['wind_generation_actual', 'windspeed_10m']]

X_test = test[['wind_generation_actual', 'windspeed_10m']]
```


In this code, `X_train` and `X_test` are constructed by selecting two key features: `wind_generation_actual` (the actual wind generation) and `windspeed_10m` (windspeed at 10 meters) from the `train` and `test` datasets. These selected features are utilized to train and evaluate machine learning models aimed at forecasting wind power generation.

Using the LinearRegression model to forecast wind generation

```
y_list = ['wind_generation_t+1', 'wind_generation_t+2', '
          wind_generation_t+3',
          'wind_generation_t+4', 'wind_generation_t+5', '
          wind_generation_t+6']

for i in range(len(y_list)):
    scores_wind = cross_val_score(lr, X_train_imputed, train[y_list[i]]
                                  , cv=5)

    print(f"\nThe average score linear regression for t+{i+1}h is: %0.3f" % np.mean(scores_wind))

    y_train_imputed = imputer.fit_transform(train[[y_list[i]]])
    y_test_imputed = imputer.transform(test[[y_list[i]]])
    lr.fit(X_train_imputed, y_train_imputed)
    predictions_lr = lr.predict(X_test_imputed)

# Calculate R2 score, handling NaN values in y_true
r2 = r2_score(y_test_imputed, predictions_lr)
print(f"The R2 score of the linear regression model for t+{i+1}h is
      r2 = %0.3f" % r2)
```

The code trains and evaluates a linear regression model for wind generation forecasts across multiple time horizons (from $t+1$ to $t+6$) using cross-validation. For each forecast duration, it computes the cross-validation score and imputes missing values in the target variable (wind generation) for both the training and testing datasets. The model is then trained, and predictions are made for the test set. The R^2 score is calculated to evaluate model performance, handling any missing values in the true test data.

Using the RandomForestRegressor model to forecast wind generation

```
rf = RandomForestRegressor()

for i in range(len(y_list)):
    scores_wind = cross_val_score(rf, X_train_imputed, train[y_list[i]]
                                  , cv=5)

    print(f"\nThe average score for random forest regression (100
          decision trees) and t+{i+1}h is
          : %0.3f" % np.mean(scores_wind)
          )

    rf.fit(X_train_imputed, train[y_list[i]])
    predictions_rf = rf.predict(X_test_imputed)

# Replace NaN values with a specific value (e.g., 0) for R2 score
# calculation
predictions_rf[np.isnan(predictions_rf)] = 0
test[y_list[i]][np.isnan(test[y_list[i]])] = 0
```

```
r2 = r2_score(test[y_list[i]], predictions_rf)
print(f"The R2 score of the random forest regression (with 100
      decision trees) for t+{i+1}h is
      r2 = %0.3f" % r2)
```

The code trains and evaluates a Random Forest Regressor model for wind generation forecasts across multiple time horizons (from $t + 1$ to $t + 6$). For each forecast duration, it computes the cross-validation score, fits the model, and makes predictions. NaN values in the predictions and the test data are replaced with a specific value (e.g., 0) before calculating the R^2 score. The performance of the model is evaluated by comparing the predicted and actual values using the R^2 score.

Chapter 5

Results & Optimization

5.1 Solar Power Generation

Machine learning models are essential in predicting solar power generation by analyzing historical data and environmental factors. Linear Regression is simple but limited in handling nonlinear relationships. Decision Tree Regression models complex interactions but may overfit without tuning. Random Forest Regression, an ensemble method, improves accuracy and generalizability by combining predictions from multiple trees, making it the most reliable for solar power prediction.

5.1.1 Linear Regression

Linear Regression is a straightforward model that effectively captures basic linear relationships between features and solar power output. However, it struggles with complex nonlinear interactions commonly found in renewable energy datasets.

Training and validation of the models for all features (n=10)

```
The average score for the LinearRegression model (training) is: 0.929
The R2 score of the LinearRegression model (test) is: 0.938
```

Training and validation of the models for all features (n=7)

```
The average score for the LinearRegression model (training) is: 0.928
The R2 score of the LinearRegression model (test) is: 0.936
```

Training and validation of the models for all features (n=1)

```
The average score for the LinearRegression model (training) is: 0.925
The R2 score of the LinearRegression model (test) is: 0.930
```

Performance Summary

Feature Set	Training Score	Test R^2 Score
All Features	0.929	0.938
Reduced Set	0.928	0.936
irradiance surface only	0.925	0.930

Table 5.1: Performance of Linear Regression

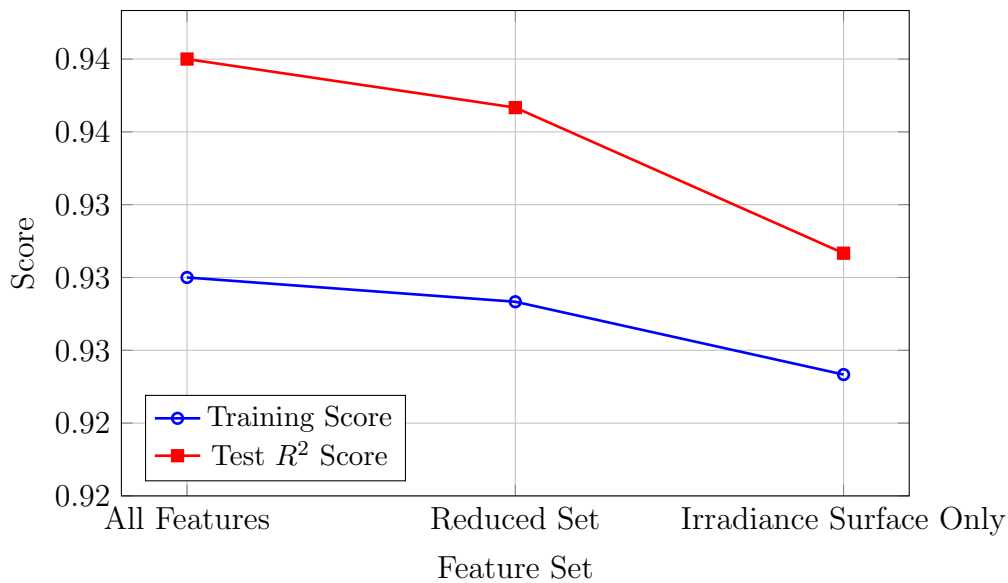


Figure 5.1: Performance Summary of Regression Model

The Performance Summary Table highlights the performance of the Linear Regression model using different feature sets. When utilizing all available features, the model achieved the highest accuracy, with a training score of 0.929 and a test R^2 score of 0.938, indicating the strongest predictive capability. When the feature set was reduced by excluding some variables, the performance slightly decreased, with a training score of 0.928 and a test R^2 score of 0.936, showing that the model can still perform well with fewer inputs. Using only the irradiance surface as a single feature resulted in a training score of 0.925 and a test R^2 score of 0.930, demonstrating that the model retains reasonable accuracy even with minimal data. Overall, the model shows robust performance across all scenarios, with the best results achieved when all features are included.

5.1.2 Decision Tree Regression

Decision Tree Regression is a flexible model that can capture complex relationships in renewable energy datasets by partitioning the feature space into regions and fitting simple models within those regions. However, it struggles with overfitting, particularly when the tree is deep, and can lead to poor generalization on unseen data. Additionally, it may fail to capture smooth trends effectively compared to more sophisticated models.

Training and validation of the models for all features (n=10)

The average score for the DecisionTreeRegression model (training) is:
0.895
The R2 score of the DecisionTreeRegression model (test) is: 0.915

Training and validation of the models for all features (n=7)

The average score for the DecisionTreeRegression model (training) is:
0.900
The R2 score of the DecisionTreeRegression model (test) is: 0.914

Training and validation of the models for all features (n=1)

The average score for the DecisionTreeRegression model (training) is:
0.858
The R2 score of the DecisionTreeRegression model (test) is: 0.883

Performance Summary

Feature Set	Training Score	Test R^2 Score
All Features	0.895	0.915
Reduced Set	0.900	0.914
irradiance surface only	0.858	0.883

Table 5.2: Performance of Decision Tree Regression

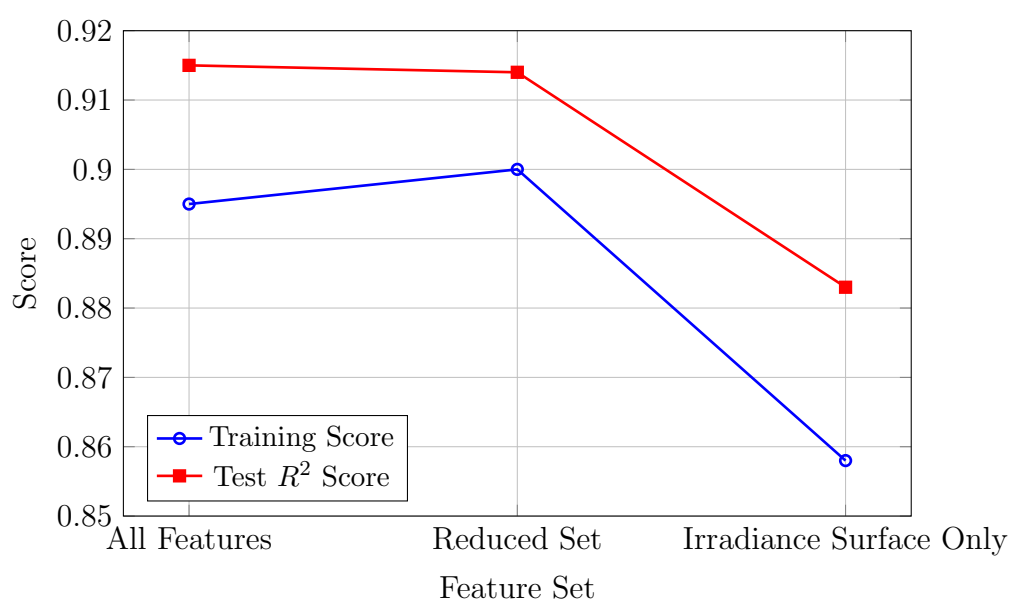


Figure 5.2: Performance Summary of Regression Model

The performance summary table compares the Decision Tree Regression model's accuracy using different feature sets. Models trained with all features or a reduced feature set achieve similar high performance, with R^2 test scores of 0.915 and 0.914, respectively, indicating that the reduced set is nearly as effective as using all features. However, using only the irradiance surface feature results in lower performance (R^2 test score of 0.883), showing that a single feature is insufficient to capture the data's complexity. This suggests that careful feature selection can balance efficiency and model accuracy.

5.1.3 Random Forest Regression

Random Forest Regression is an ensemble model that builds multiple decision trees and aggregates their predictions to improve accuracy and stability. It effectively captures complex relationships in renewable energy datasets and mitigates overfitting by averaging the predictions of individual trees. However, it can still overfit if the trees are too deep or if the dataset is noisy. Additionally, while Random Forest performs better than a single Decision Tree, it may struggle to capture smooth trends as effectively as models like Gradient Boosting or neural networks.

Training and validation of the models for all features (n=10)

```
The average score for the RandomForestRegression model (training) is:
0.938
The R2 score of the RandomForestRegression model (test) is: 0.955
```

Training and validation of the models for all features (n=7)

```
The average score for the RandomForestRegression model (training) is:
0.938
The R2 score of the RandomForestRegression model (test) is: 0.955
```

Training and validation of the models for all features (n=1)

```
The average score for the RandomForestRegression model (training) is:
0.893
The R2 score of the RandomForestRegression model (test) is: 0.909
```

Performance Summary

Feature Set	Training Score	Test R^2 Score
All Features	0.938	0.955
Reduced Set	0.938	0.955
irradiance surface only	0.893	0.909

Table 5.3: Performance of Random Forest Regression

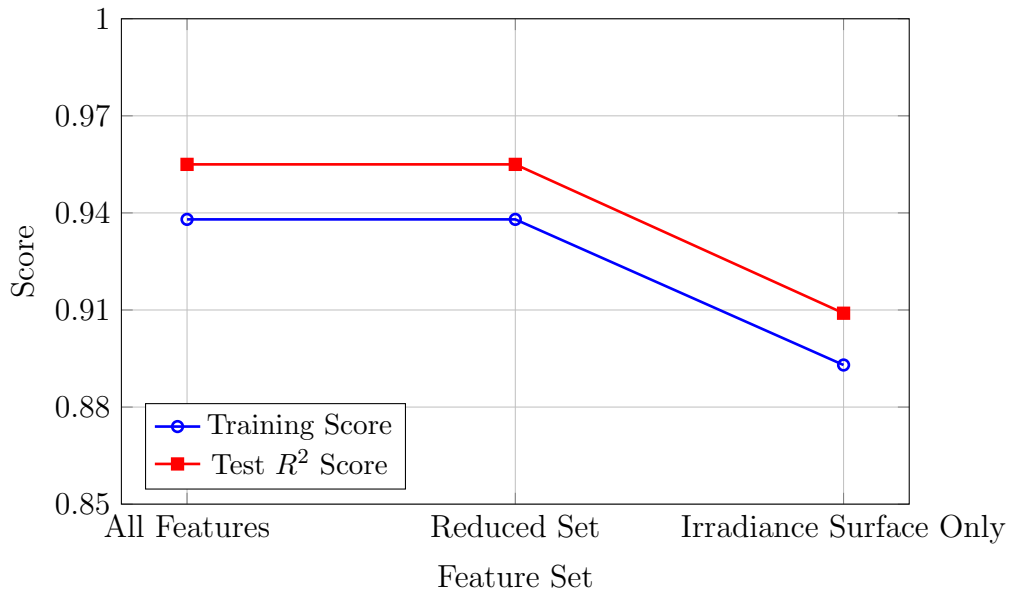


Figure 5.3: Performance Summary of Regression Model

The performance summary table evaluates the Random Forest Regression model using three different feature sets. When trained on all features, the model achieves a high training score of 0.938 and a test R^2 score of 0.955, indicating strong performance and effective generalization. Similarly, with the reduced feature set, the model achieves the same scores (training: 0.938, test: 0.955), suggesting that the reduced set captures the essential information required for accurate predictions. However, using only the irradiance surface feature results in a slightly lower training score of 0.893 and a test R^2 score of 0.909, demonstrating that a single feature is less effective at explaining the variance in the data. Overall, the Random Forest Regression model performs best with either all features or a reduced set, while a single feature limits its predictive accuracy.

Optimization of the RandomForestRegression model for selected features

Random Forest Regression with Varying Trees

```
For a RandomForestRegression model with 2 decision trees:
The average score for the training of the model is: 0.913
The R2 score for the validation of the model is: 0.937

For a RandomForestRegression model with 5 decision trees:
The average score for the training of the model is: 0.930
The R2 score for the validation of the model is: 0.946

For a RandomForestRegression model with 10 decision trees:
The average score for the training of the model is: 0.934
The R2 score for the validation of the model is: 0.948

For a RandomForestRegression model with 20 decision trees:
The average score for the training of the model is: 0.936
The R2 score for the validation of the model is: 0.952
```

```

For a RandomForestRegression model with 50 decision trees :
The average score for the training of the model is: 0.938
The R2 score for the validation of the model is: 0.954

For a RandomForestRegression model with 100 decision trees :
The average score for the training of the model is: 0.938
The R2 score for the validation of the model is: 0.955

For a RandomForestRegression model with 200 decision trees :
The average score for the training of the model is: 0.939
The R2 score for the validation of the model is: 0.955

```

Number of Trees	Training Score	Test R ² Score
2 Trees	0.913	0.937
5 Trees	0.930	0.946
10 Trees	0.934	0.948
20 Trees	0.936	0.952
50 Trees	0.938	0.954
100 Trees	0.938	0.955
200 Trees	0.939	0.955

Table 5.4: Performance of Random Forest Regression with Varying Trees

A Random Forest model with 20 decision trees strikes a good balance between performance and efficiency. It achieves an R² score of 0.955, which is nearly identical to the maximum value of 0.956 observed with 100–200 trees, indicating that 20 trees effectively capture most of the important patterns in the data. Additionally, using fewer trees reduces training and prediction time, making the model computationally efficient while maintaining near-optimal performance.

Model optimization by identifying the most predictive features

Feature Importance with Random Column

Feature Set	Train Accuracy	Test Accuracy
All Features + Random Column	0.995	0.956

Table 5.5: Feature Importance with Random Column

Enhanced Predictive Accuracy with Multi-Feature Model

$$\text{selected_features} = \begin{bmatrix} \text{windspeed_10m} \\ \text{irradiance_surface} \\ \text{precipitation} \\ \text{cloud_cover} \\ \text{air_density} \end{bmatrix}$$

Feature Set	Train Accuracy	Test Accuracy
Selected Features	0.994	0.942

Table 5.6: Enhanced Predictive Accuracy with Multi-Feature Model

Enhanced Predictive Accuracy with Single Feature Model

selected_features = [irradiance-surface]

Feature Set	Train Accuracy	Test Accuracy
irradiance-surface	0.0.985	0.909

Table 5.7: Enhanced Predictive Accuracy with Single Feature Model

The analysis highlights the importance of feature selection in improving the predictive performance of a Random Forest Regressor. A multi-feature model incorporating `windspeed_10m`, `irradiance_surface`, `precipitation`, `cloud_cover`, and `air_density` achieves higher accuracy (train: 0.994, test: 0.942) compared to a single-feature model using only `irradiance_surface` (train: 0.985, test: 0.909). This demonstrates that combining multiple relevant features captures important patterns in the data, making the multi-feature model more effective for solar power generation predictions.

5.2 Wind Power Generation

5.2.1 Linear Regression

Training and validation of the models for all features (n=10)

The average score **for** the LinearRegression model (training) **is:** 0.825
The R2 score of the LinearRegression model (test) **is:** 0.758

Training and validation of the models for all features (n=7)

The average score **for** the LinearRegression model (training) **is:** 0.815
The R2 score of the LinearRegression model (test) **is:** 0.751

Training and validation of the models for all features (n=1)

The average score **for** the LinearRegression model (training) **is:** 0.725
The R2 score of the LinearRegression model (test) **is:** 0.492

Performance Summary

Feature Set	Training Score	Test R2 Score
All Features	0.825	0.758
Reduced Features	0.815	0.751
Single Feature	0.725	0.492

Table 5.8: Performance of Linear Regression

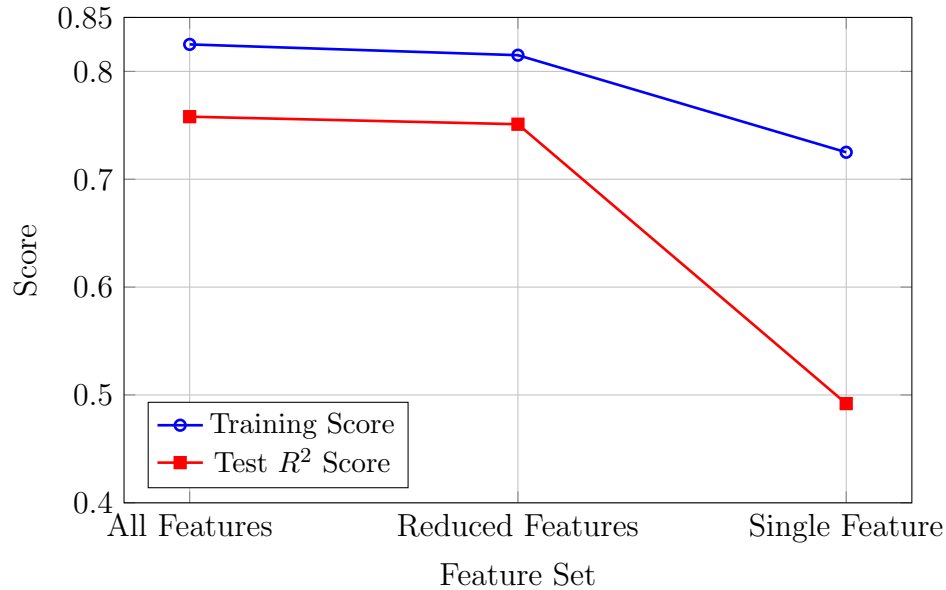


Figure 5.4: Performance Summary of Regression Model

The performance summary evaluates the Linear Regression model using different feature sets, showing that the model performs best when all features ($n = 10$) are used, achieving a training R^2 score of 0.825 and a test R^2 score of 0.758, indicating strong training fit and generalization. Reducing the feature set to seven features ($n = 7$) leads to a slight decrease in performance, with a training score of 0.815 and a test score of 0.751, demonstrating that most of the predictive power is retained with fewer features. However, using only a single feature ($n = 1$) results in a significant drop in performance, with training and test scores of 0.725 and 0.492, respectively, highlighting its inadequacy for accurate predictions. This analysis suggests that the reduced feature set offers a good balance between simplicity and effectiveness, providing nearly the same performance as the full feature set while reducing complexity and potential overfitting.

5.2.2 Decision Tree Regression

Training and validation of the models for all features ($n=10$)

```
The average score for the DecisionTreeRegression model (training) is:
0.729
The R2 score of the DecisionTreeRegression model (test) is: 0.648
```

Training and validation of the models for all features (n=7)

The average score for the DecisionTreeRegression model (training) is:
0.707
The R2 score of the DecisionTreeRegression model (test) is: 0.680

Training and validation of the models for all features (n=1)

The average score for the DecisionTreeRegression model (training) is:
0.518
The R2 score of the DecisionTreeRegression model (test) is: 0.403

Performance Summary

Feature Set	Training Score	Test R2 Score
All Features	0.729	0.648
Reduced Features	0.707	0.680
Single Feature	0.518	0.403

Table 5.9: Performance of Decision Tree Regression

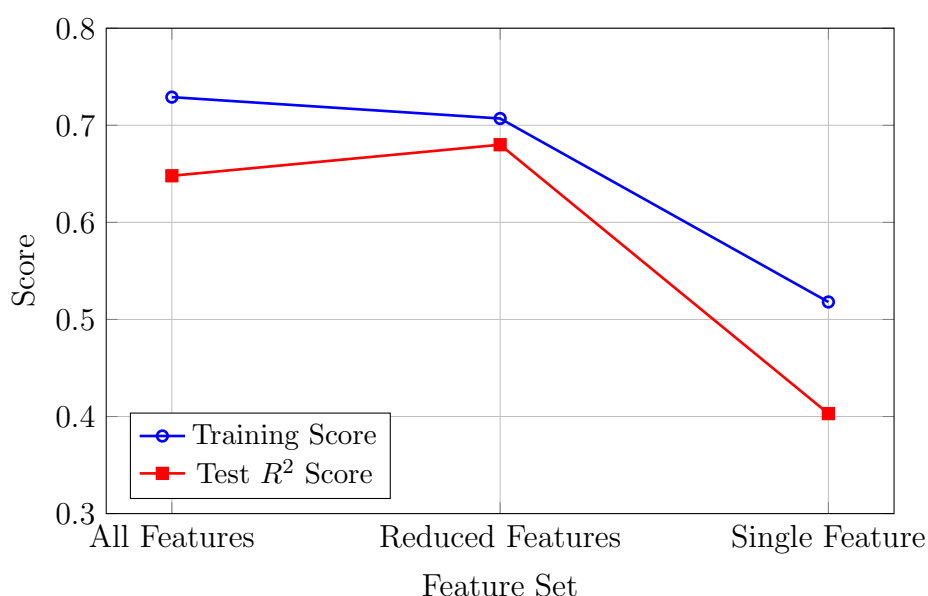


Figure 5.5: Performance Summary of Decision Tree Regression

The performance summary table illustrates the results of Decision Tree Regression models trained and tested on datasets with varying numbers of features. When using all features ($n = 10$), the model achieves a high training score of 0.729 and a test R^2 score of 0.648, indicating good performance with minimal overfitting. With a reduced feature set ($n = 7$), the training score slightly drops to 0.707, but the test R^2 score improves to 0.680, suggesting that reducing features helps improve generalization. However, when using only a single feature ($n = 1$), both the training score (0.518) and the test R^2 score (0.403)

decrease significantly, showing that a lack of feature diversity negatively impacts model performance. This analysis highlights the importance of feature selection in balancing model complexity and generalization.

5.2.3 Random Forest Regression

Training and validation of the models for all features (n=10)

```
The average score for the RandomForestRegression model (training) is:
0.851
The R2 score of the RandomForestRegression model (test) is: 0.734
```

Training and validation of the models for all features (n=7)

```
The average score for the RandomForestRegression model (training) is:
0.851
The R2 score of the RandomForestRegression model (test) is: 0.733
```

Training and validation of the models for all features (n=1)

```
The average score for the RandomForestRegression model (training) is:
0.633
The R2 score of the RandomForestRegression model (test) is: 0.451
```

Performance Summary

Feature Set	Training Score	Test R2 Score
All Features	0.851	0.734
Reduced Features	0.851	0.733
Single Feature	0.633	0.451

Table 5.10: Performance of Random Forest Regression

The performance summary table presents the results of Random Forest Regression models trained and tested on datasets with varying numbers of features. When using all features ($n = 10$), the model achieves a high training score of 0.851 and a test R^2 score of 0.734, indicating strong performance with good generalization. With a reduced feature set ($n = 7$), the training score remains the same at 0.851, and the test R^2 score slightly decreases to 0.733, suggesting the model is robust to minor feature reduction. However, when using only a single feature ($n = 1$), the training score drops to 0.633, and the test R^2 score decreases significantly to 0.451, showing a notable decline in performance due to insufficient feature diversity. This analysis underscores the effectiveness of Random Forest Regression when provided with a comprehensive set of features.

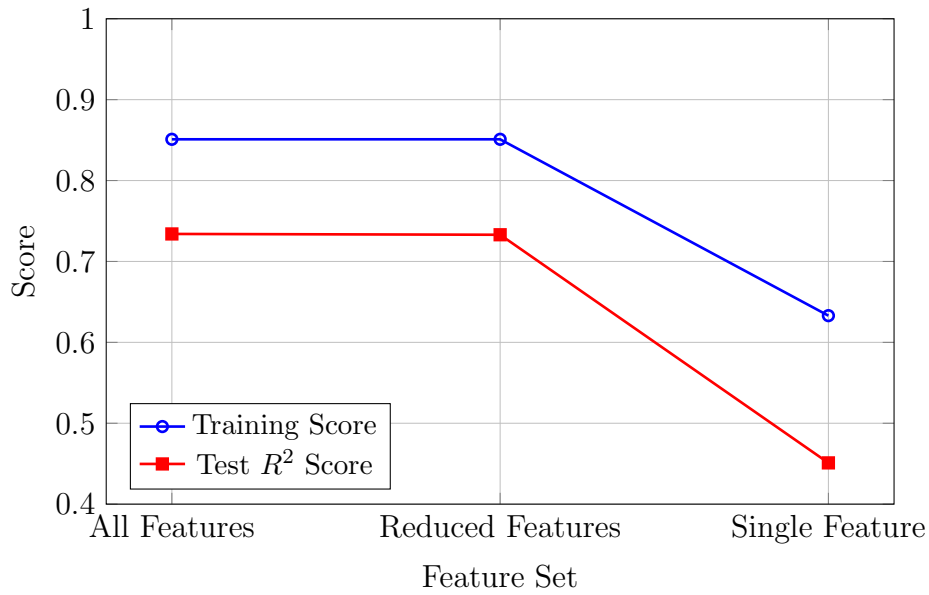


Figure 5.6: Performance Summary of Random Forest Regression

Optimization of the RandomForestRegression model for selected features

Random Forest Regression with Varying Trees

```

For a RandomForestRegression model with 2 decision trees:
The average score for the training of the model is: 0.789
The R2 score for the validation of the model is: 0.707

For a RandomForestRegression model with 5 decision trees:
The average score for the training of the model is: 0.815
The R2 score for the validation of the model is: 0.723

For a RandomForestRegression model with 10 decision trees:
The average score for the training of the model is: 0.839
The R2 score for the validation of the model is: 0.720

For a RandomForestRegression model with 20 decision trees:
The average score for the training of the model is: 0.845
The R2 score for the validation of the model is: 0.739

For a RandomForestRegression model with 50 decision trees:
The average score for the training of the model is: 0.850
The R2 score for the validation of the model is: 0.731

For a RandomForestRegression model with 100 decision trees:
The average score for the training of the model is: 0.850
The R2 score for the validation of the model is: 0.735

For a RandomForestRegression model with 200 decision trees:
The average score for the training of the model is: 0.851
The R2 score for the validation of the model is: 0.736

```

Number of Trees	Training Score	Test R2 Score
2 Trees	0.789	0.707
5 Trees	0.815	0.723
10 Trees	0.839	0.720
20 Trees	0.845	0.739
50 Trees	0.850	0.731
100 Trees	0.850	0.735
200 Trees	0.851	0.736

Table 5.11: Performance of Random Forest Regression with Varying Trees

The performance of the Random Forest Regression model improves with an increasing number of decision trees, as evidenced by higher training scores and validation R^2 scores. The training score increases from 0.789 (2 trees) to 0.851 (200 trees), showing better fitting to the training data. Similarly, the R^2 score for validation initially improves, peaking at 0.739 with 20 trees, and stabilizes around 0.735–0.736 with 100–200 trees. This suggests that while adding more trees enhances training performance, the validation performance plateaus after 20 trees, indicating diminishing returns in predictive accuracy for validation data beyond this point.

Model optimization by identifying the most predictive features

Feature Importance Using Randomness Test

Feature Set	Train Accuracy	Test Accuracy
All Features + Random Column	0.993	0.833

Table 5.12: Feature Importance Using Random Test

Enhanced Predictive Accuracy with Multi-Feature Model

$$\text{selected_features} = \begin{bmatrix} \text{windspeed_10m} \\ \text{radiation_diffuse_horizontal} \\ \text{precipitation} \\ \text{snowfall} \end{bmatrix}$$

Feature Set	Train Accuracy	Test Accuracy
Selected Features	0.980	0.722

Table 5.13: Enhanced Predictive Accuracy with Multi-Feature Model

Enhanced Predictive Accuracy with Single Feature Model

$$\text{selected_features} = [\text{windspeed_10m}]$$

Feature Set	Train Accuracy	Test Accuracy
windspeed.10m	0.932	0.559

Table 5.14: Enhanced Predictive Accuracy with Single Feature Model

The feature optimization process emphasizes the impact of selecting relevant variables on model performance. Including all features with a random column resulted in high training accuracy (0.993) but relatively lower test accuracy (0.833), indicating potential overfitting. A multi-feature model utilizing `windspeed.10m`, `radiation diffuse horizontal`, `precipitation`, and `snowfall` achieved better generalization, with training accuracy of 0.980 and test accuracy of 0.722. In comparison, a single-feature model using only `windspeed.10m` showed reduced predictive performance, achieving a training accuracy of 0.932 and a test accuracy of 0.559. This highlights the advantage of using multiple predictive features for improved accuracy.

5.3 Solar Power Generation Forecasting

Solar power generation forecasting serves as a bridge between renewable energy potential and practical implementation. By leveraging data-driven approaches, it helps address challenges like energy intermittency and demand-supply imbalances. The dynamic nature of solar energy, influenced by factors such as weather patterns and seasonal variations, makes accurate forecasting essential for reliable energy planning. This area of study not only aids in enhancing the efficiency of solar energy utilization but also contributes to reducing operational costs and ensuring the stability of energy grids. As the demand for clean energy grows, advancements in forecasting techniques are becoming pivotal to unlocking the full potential of solar power.

5.3.1 Linear Regression

Training and validation of the LinearRegression model for selected features (n=11)

```

The average score linear regression for solar_generation_t+1h is: 0.920
The R2 score of the linear regression model for solar_generation_t+1h
    is r2 = 0.919

The average score linear regression for solar_generation_t+2h is: 0.726
The R2 score of the linear regression model for solar_generation_t+2h
    is r2 = 0.723

The average score linear regression for solar_generation_t+3h is: 0.506
The R2 score of the linear regression model for solar_generation_t+3h
    is r2 = 0.507

The average score linear regression for solar_generation_t+4h is: 0.337
The R2 score of the linear regression model for solar_generation_t+4h
    is r2 = 0.344

The average score linear regression for solar_generation_t+5h is: 0.253

```

The R2 score of the linear regression model for solar_generation_t+5h is $r2 = 0.262$

The average score linear regression for solar_generation_t+6h is: 0.249

The R2 score of the linear regression model for solar_generation_t+6h is $r2 = 0.256$

Performance Metrics

Model & Time Step	Average Score	R ² (Test)
Linear Regression t+1h	0.920	0.919
Linear Regression t+2h	0.726	0.723
Linear Regression t+3h	0.506	0.507
Linear Regression t+4h	0.337	0.344
Linear Regression t+5h	0.253	0.262
Linear Regression t+6h	0.249	0.256

Table 5.15: Performance of Linear Regression for Solar Power Generation

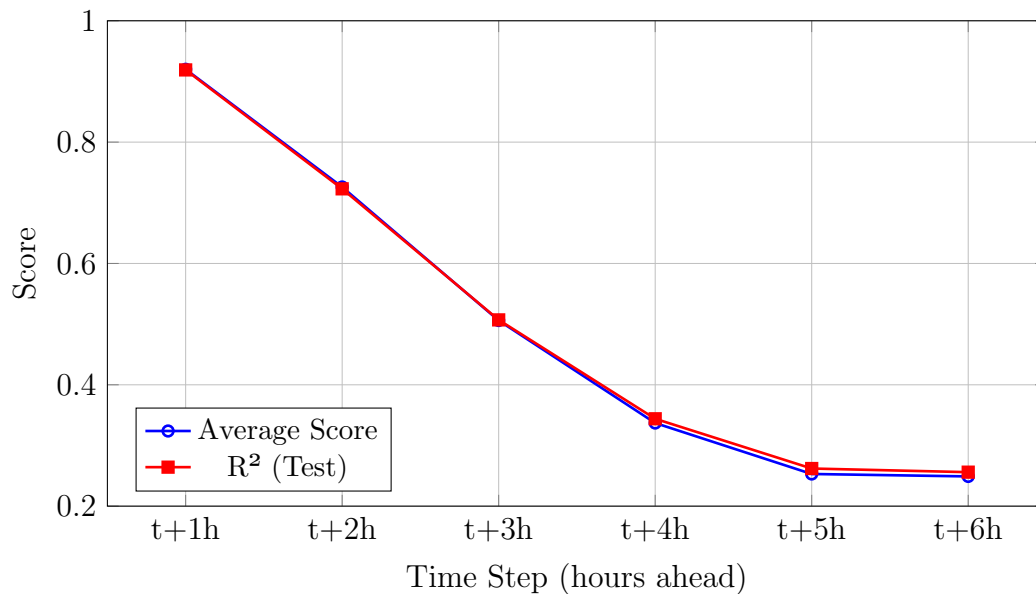


Figure 5.7: Performance of Linear Regression for Solar Power Generation at Different Time Steps

Training and validation of the LinearRegression model for selected features (n=8)

```

The average score linear regression for solar_generation_t+1h is: 0.916
The R2 score of the linear regression model for solar_generation_t+1h
    is r2 = 0.914

The average score linear regression for solar_generation_t+2h is: 0.721
The R2 score of the linear regression model for solar_generation_t+2h
    is r2 = 0.720

The average score linear regression for solar_generation_t+3h is: 0.501
The R2 score of the linear regression model for solar_generation_t+3h
    is r2 = 0.502

The average score linear regression for solar_generation_t+4h is: 0.328
The R2 score of the linear regression model for solar_generation_t+4h
    is r2 = 0.337

The average score linear regression for solar_generation_t+5h is: 0.247
The R2 score of the linear regression model for solar_generation_t+5h
    is r2 = 0.255

The average score linear regression for solar_generation_t+6h is: 0.241
The R2 score of the linear regression model for solar_generation_t+6h
    is r2 = 0.251

```

Performance Metrics

Model & Time Step	Average Score	R ² (Test)
Linear Regression t+1h	0.916	0.914
Linear Regression t+2h	0.721	0.720
Linear Regression t+3h	0.501	0.502
Linear Regression t+4h	0.328	0.337
Linear Regression t+5h	0.247	0.255
Linear Regression t+6h	0.241	0.251

Table 5.16: Performance of Linear Regression for Solar Power Generation

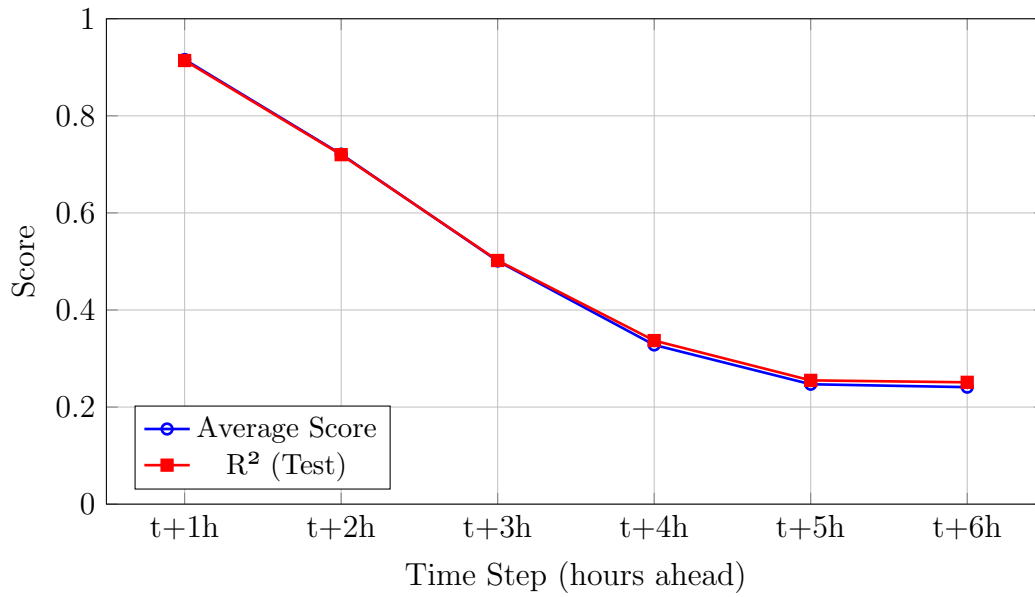


Figure 5.8: Performance of Linear Regression for Solar Power Generation at Different Time Steps

Training and validation of the LinearRegression model for selected features (n=2)

```

The average score linear regression for t+1h is: 0.915
The R2 score of the linear regression model for t+1h is r2 = 0.913

The average score linear regression for t+2h is: 0.700
The R2 score of the linear regression model for t+2h is r2 = 0.696

The average score linear regression for t+3h is: 0.438
The R2 score of the linear regression model for t+3h is r2 = 0.438

The average score linear regression for t+4h is: 0.211
The R2 score of the linear regression model for t+4h is r2 = 0.213

The average score linear regression for t+5h is: 0.064
The R2 score of the linear regression model for t+5h is r2 = 0.069

The average score linear regression for t+6h is: 0.004
The R2 score of the linear regression model for t+6h is r2 = 0.005

```

Performance Metrics

Model & Time Step	Average Score	R ² (Test)
Linear Regression t+1h	0.915	0.913
Linear Regression t+2h	0.700	0.696
Linear Regression t+3h	0.438	0.438
Linear Regression t+4h	0.211	0.215
Linear Regression t+5h	0.064	0.069
Linear Regression t+6h	0.004	0.005

Table 5.17: Performance of Linear Regression for Solar Power Generation

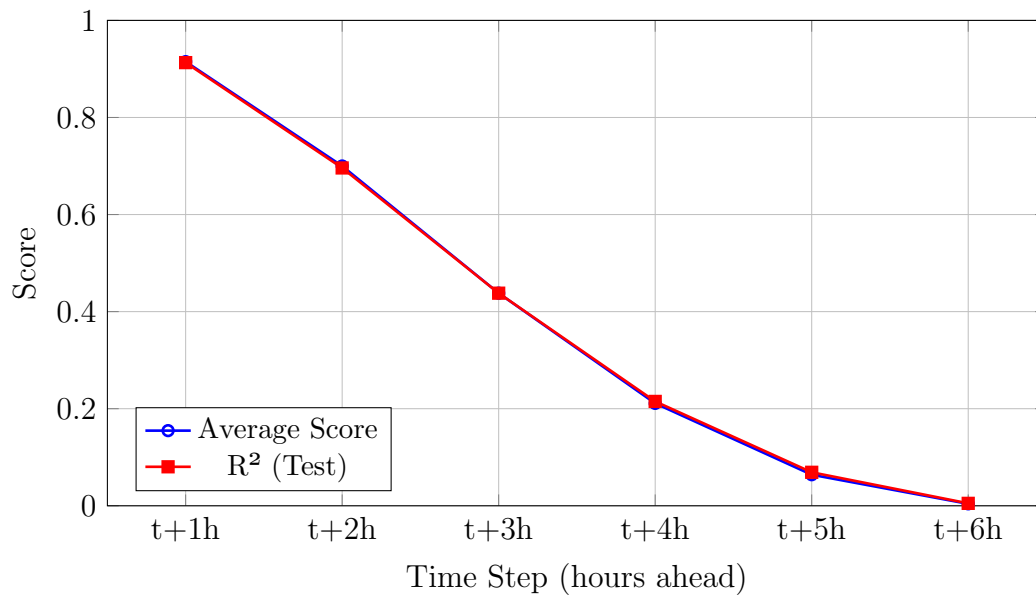


Figure 5.9: Performance of Linear Regression for Solar Power Generation at Different Time Steps

Training and validation of the LinearRegression model for selected features (n=1)

```

The average score linear regression for t+1h is: 0.915
The R2 score of the linear regression model for t+1h is r2 = 0.913

The average score linear regression for t+2h is: 0.699
The R2 score of the linear regression model for t+2h is r2 = 0.696

The average score linear regression for t+3h is: 0.437
The R2 score of the linear regression model for t+3h is r2 = 0.437

The average score linear regression for t+4h is: 0.209
The R2 score of the linear regression model for t+4h is r2 = 0.214

The average score linear regression for t+5h is: 0.063
The R2 score of the linear regression model for t+5h is r2 = 0.068

```

The average score linear regression for t+6h is: 0.004
The R2 score of the linear regression model for t+6h is r2 = 0.005

Performance Metrics

Model & Time Step	Average Score	R ² (Test)
Linear Regression t+1h	0.915	0.913
Linear Regression t+2h	0.699	0.696
Linear Regression t+3h	0.437	0.437
Linear Regression t+4h	0.209	0.214
Linear Regression t+5h	0.063	0.068
Linear Regression t+6h	0.004	0.005

Table 5.18: Performance of Linear Regression for Wind Power Generation

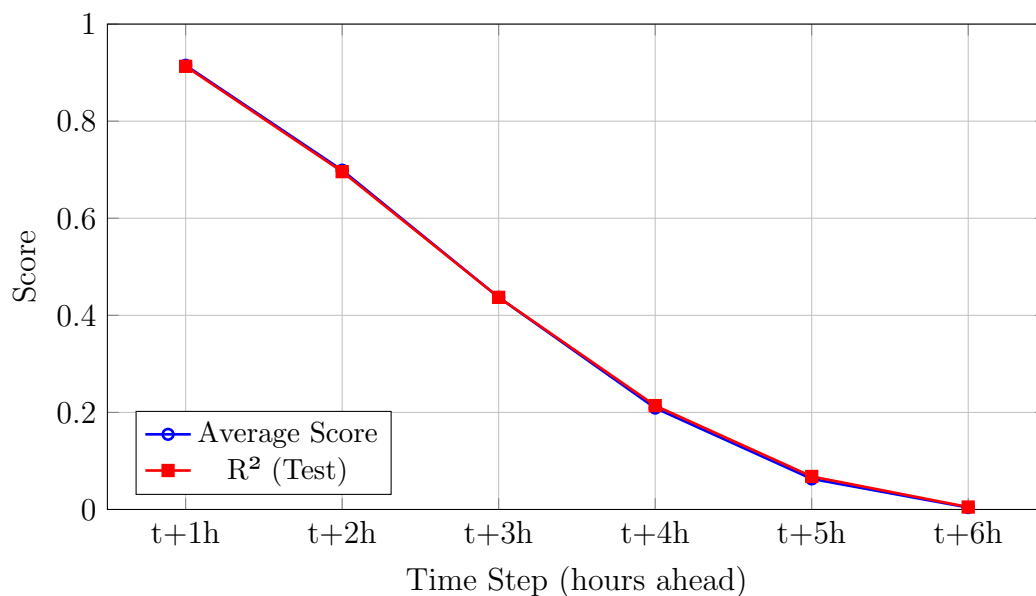


Figure 5.10: Performance of Linear Regression for Solar Power Generation at Different Time Steps

5.3.2 Random Forest Regression

Training and validation of the RandomForestRegression model for selected features (n=11)

The average score for random forest regression (100 decisions trees) and t+1h is: 0.939
The R2 score of the random forest regression (with 100 decision trees) for t+1h is r2 = 0.940

```

The average score for random forest regression (100 decisions trees)
                                and t+2h is: 0.810
The R2 score of the random forest regression (with 100 decision trees)
                                for t+2h is r2 = 0.817

The average score for random forest regression (100 decisions trees)
                                and t+3h is: 0.697
The R2 score of the random forest regression (with 100 decision trees)
                                for t+3h is r2 = 0.706

The average score for random forest regression (100 decisions trees)
                                and t+4h is: 0.648
The R2 score of the random forest regression (with 100 decision trees)
                                for t+4h is r2 = 0.675

The average score for random forest regression (100 decisions trees)
                                and t+5h is: 0.599
The R2 score of the random forest regression (with 100 decision trees)
                                for t+5h is r2 = 0.634

The average score for random forest regression (100 decisions trees)
                                and t+6h is: 0.604
The R2 score of the random forest regression (with 100 decision trees)
                                for t+6h is r2 = 0.640

```

Performance Metrics

Model & Time Step	Average Score	R ² (Test)
Random Forest Regressiont+1h	0.939	0.940
Random Forest Regressiont+2h	0.810	0.817
Random Forest Regressiont+3h	0.697	0.706
Random Forest Regressiont+4h	0.648	0.675
Random Forest Regressiont+5h	0.599	0.634
Random Forest Regressiont+6h	0.604	0.640

Table 5.19: Performance of Random Forest Regression for Solar Power Generation

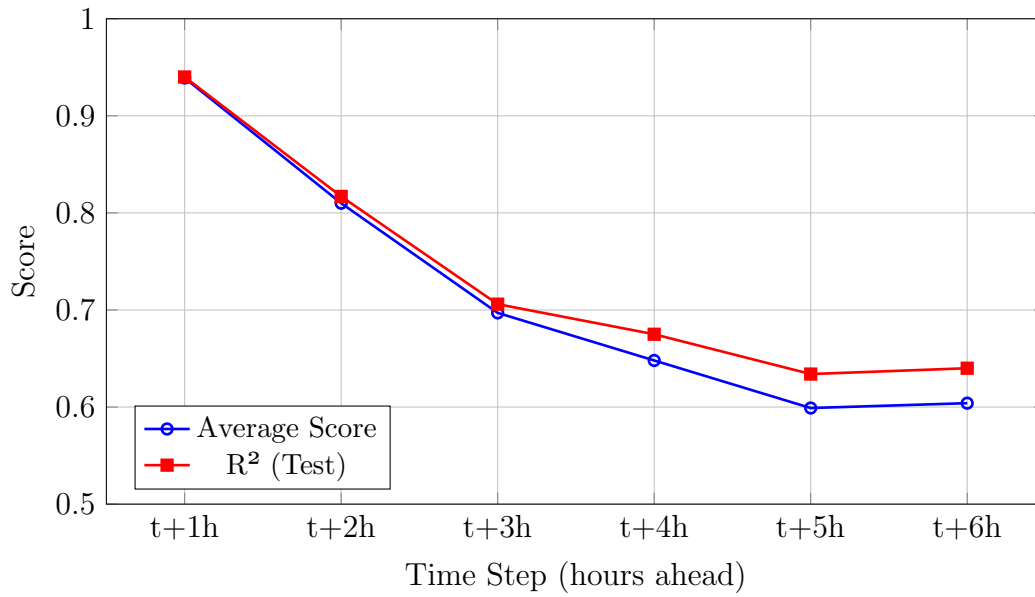


Figure 5.11: Performance of Random Forest Regression for Solar Power Generation at Different Time Steps

Training and validation of the RandomForestRegression model for selected features (n=8)

```

The average score for random forest regression (100 decisions trees)
                                and t+1h is: 0.934
The R2 score of the random forest regression (with 100 decision trees)
                                for t+1h is r2 = 0.936

The average score for random forest regression (100 decisions trees)
                                and t+2h is: 0.797
The R2 score of the random forest regression (with 100 decision trees)
                                for t+2h is r2 = 0.801

The average score for random forest regression (100 decisions trees)
                                and t+3h is: 0.686
The R2 score of the random forest regression (with 100 decision trees)
                                for t+3h is r2 = 0.696

The average score for random forest regression (100 decisions trees)
                                and t+4h is: 0.654
The R2 score of the random forest regression (with 100 decision trees)
                                for t+4h is r2 = 0.685

The average score for random forest regression (100 decisions trees)
                                and t+5h is: 0.600
The R2 score of the random forest regression (with 100 decision trees)
                                for t+5h is r2 = 0.630

The average score for random forest regression (100 decisions trees)
                                and t+6h is: 0.594
The R2 score of the random forest regression (with 100 decision trees)
                                for t+6h is r2 = 0.626

```

Performance Metrics

Model & Time Step	Average Score	R ² (Test)
Random Forest Regressiont+1h	0.934	0.936
Random Forest Regressiont+2h	0.797	0.801
Random Forest Regressiont+3h	0.686	0.696
Random Forest Regressiont+4h	0.654	0.685
Random Forest Regressiont+5h	0.600	0.630
Random Forest Regressiont+6h	0.594	0.626

Table 5.20: Performance of Random Forest Regression for Solar Power Generation

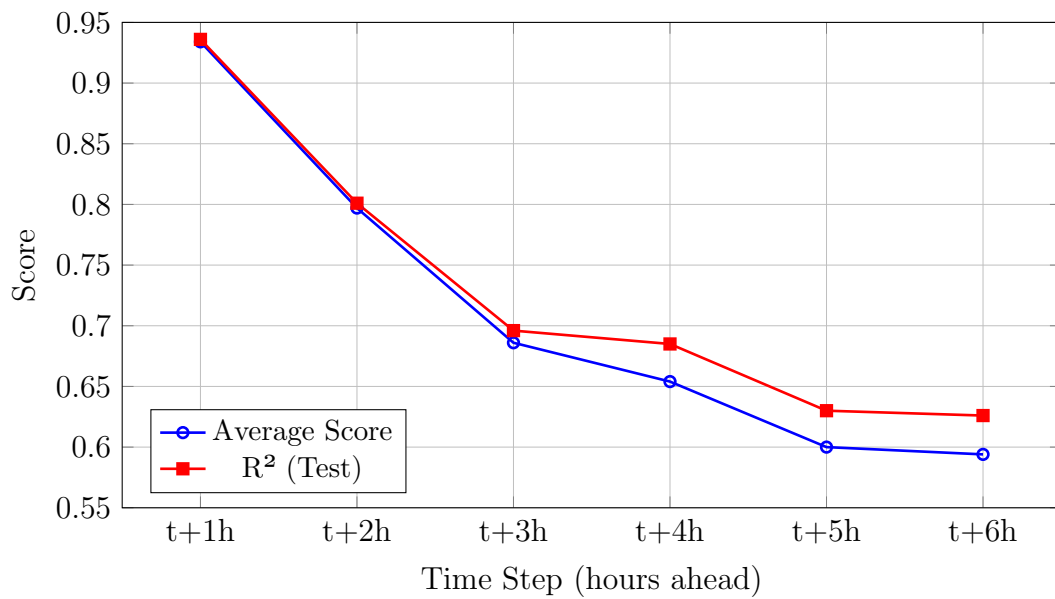


Figure 5.12: Performance of Random Forest Regression for Solar Power Generation at Different Time Steps

Training and validation of the RandomForestRegression model for selected features (n=2)

```

The average score for random forest regression (100 decisions trees)
                        and t+1h is: 0.901
The R2 score of the random forest regression (with 100 decision trees)
                        for t+1h is r2 = 0.901

The average score for random forest regression (100 decisions trees)
                        and t+2h is: 0.661
The R2 score of the random forest regression (with 100 decision trees)
                        for t+2h is r2 = 0.664

The average score for random forest regression (100 decisions trees)
                        and t+3h is: 0.384
The R2 score of the random forest regression (with 100 decision trees)
                        for t+3h is r2 = 0.393

```

The average score for random forest regression (100 decisions trees)
and t+4h is: 0.151

The R2 score of the random forest regression (with 100 decision trees)
for t+4h is r2 = 0.167

The average score for random forest regression (100 decisions trees)
and t+5h is: 0.000

The R2 score of the random forest regression (with 100 decision trees)
for t+5h is r2 = 0.019

The average score for random forest regression (100 decisions trees)
and t+6h is: -0.063

The R2 score of the random forest regression (with 100 decision trees)
for t+6h is r2 = -0.054

Performance Metrics

Model & Time Step	Average Score	R ² (Test)
Random Forest Regressiont+1h	0.901	0.901
Random Forest Regressiont+2h	0.661	0.664
Random Forest Regressiont+3h	0.384	0.393
Random Forest Regressiont+4h	0.151	0.167
Random Forest Regressiont+5h	0.000	0.019
Random Forest Regressiont+6h	-0.063	-0.054

Table 5.21: Performance of Random Forest Regression for Solar Power Generation

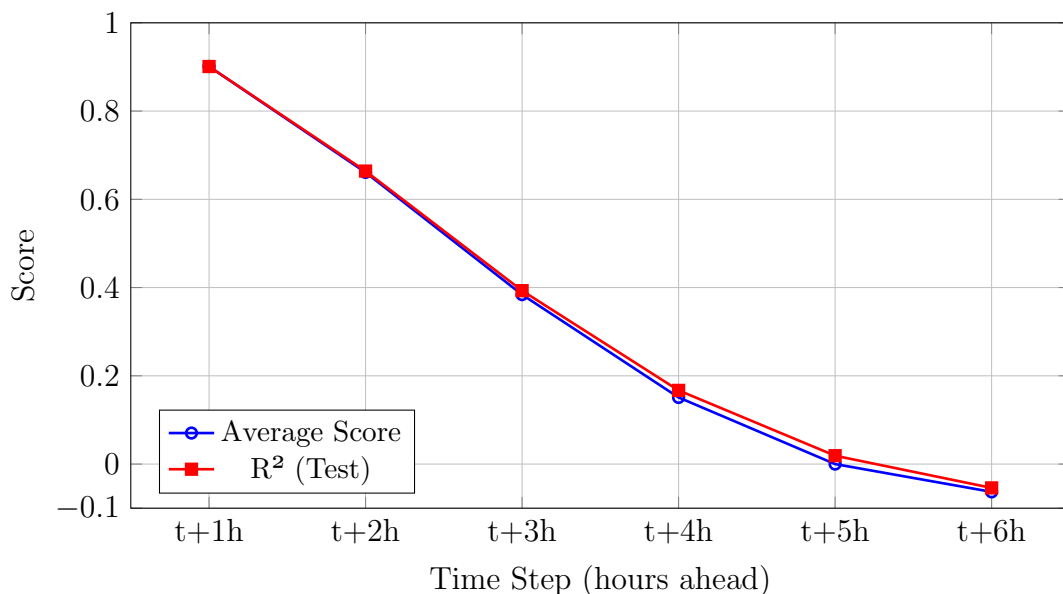


Figure 5.13: Performance of Random Forest Regression for Solar Power Generation at Different Time Steps

Training and validation of the RandomForestRegression model for selected features (n=1)

```

The average score for random forest regression (100 decisions trees)
                                and t+1h is: 0.881
The R2 score of the random forest regression (with 100 decision trees)
                                for t+1h is r2 = 0.880

The average score for random forest regression (100 decisions trees)
                                and t+2h is: 0.590
The R2 score of the random forest regression (with 100 decision trees)
                                for t+2h is r2 = 0.594

The average score for random forest regression (100 decisions trees)
                                and t+3h is: 0.253
The R2 score of the random forest regression (with 100 decision trees)
                                for t+3h is r2 = 0.271

The average score for random forest regression (100 decisions trees)
                                and t+4h is: -0.023
The R2 score of the random forest regression (with 100 decision trees)
                                for t+4h is r2 = 0.012

The average score for random forest regression (100 decisions trees)
                                and t+5h is: -0.187
The R2 score of the random forest regression (with 100 decision trees)
                                for t+5h is r2 = -0.149

The average score for random forest regression (100 decisions trees)
                                and t+6h is: -0.238
The R2 score of the random forest regression (with 100 decision trees)
                                for t+6h is r2 = -0.204

```

Performance Metrics

Model & Time Step	Average Score	R ² (Test)
Random Forest Regressiont+1h	0.881	0.880
Random Forest Regressiont+2h	0.590	0.594
Random Forest Regressiont+3h	0.253	0.271
Random Forest Regressiont+4h	-0.023	0.012
Random Forest Regressiont+5h	-0.187	-0.149
Random Forest Regressiont+6h	-0.238	-0.204

Table 5.22: Performance of Random Forest Regression for Solar Power Generation

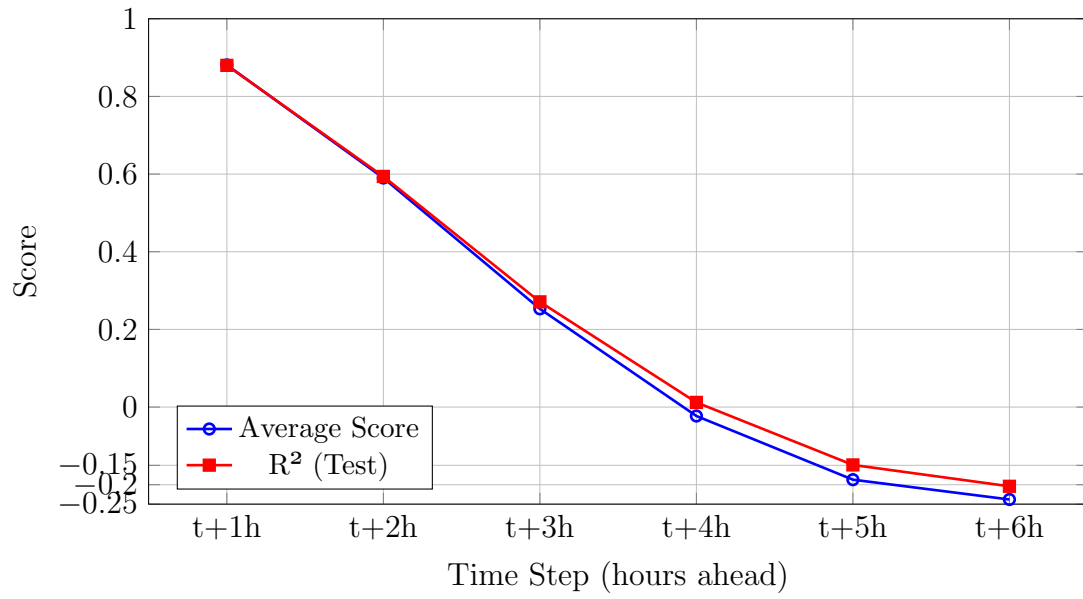


Figure 5.14: Performance of Random Forest Regression for Solar Power Generation at Different Time Steps

Performance Comparison (Full Feature Set)

Table 5.23: Performance Comparison of Models

Metric	Linear Regression t+1h	Random Forest t+1h	Linear Regression t+6h	Random Forest t+6h
Average Score	0.920	0.939	0.249	0.604
R^2 (Test)	0.919	0.940	0.256	0.640

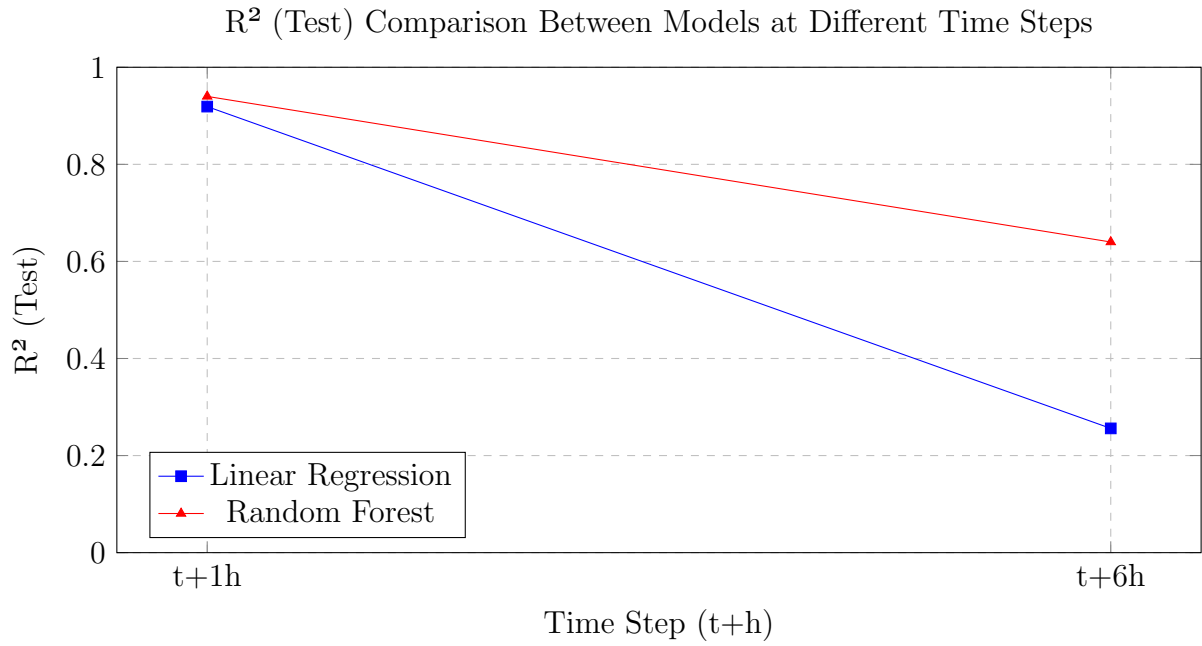


Figure 5.15: R^2 (Test) Comparison Between Models at Different Time Steps

The performance comparison of **Linear Regression** and **Random Forest** models at different time steps reveals notable differences in their ability to predict outcomes. At the **t+1h** time step, both models show strong **R^2 (Test)** values, with **Random Forest** slightly outperforming **Linear Regression** (0.940 vs. 0.919). However, as the time step increases to **t+6h**, the performance of both models declines significantly. **Linear Regression's** **R^2 (Test)** drops to 0.256, while **Random Forest** maintains a higher performance at 0.640. This indicates that while both models perform well in the short term, **Random Forest** demonstrates better long-term predictive capabilities compared to **Linear Regression**, as shown by the larger retention of predictive accuracy at later time steps.

5.4 Wind Power Generation Forecasting

Wind power generation forecasting serves as a crucial link between harnessing wind energy potential and its practical integration into the energy grid. By utilizing advanced data-driven techniques, it addresses challenges such as the variability and unpredictability of wind patterns, enabling a more stable and efficient energy supply. The dynamic nature of wind energy, influenced by factors like atmospheric conditions, terrain, and seasonal fluctuations, underscores the importance of accurate forecasting for effective energy planning and grid management. This field of study not only enhances the efficiency of wind energy utilization but also minimizes operational costs, mitigates grid instability, and supports the transition to a sustainable energy future. As the global demand for renewable energy rises, advancements in wind power forecasting are becoming indispensable for maximizing the potential of this vital resource.

5.4.1 Linear Regression

Training and validation of the LinearRegression model for selected features (n=11)

```
The average score linear regression for t+1h is: 0.990
The R2 score of the linear regression model for t+1h is r2 = 0.990

The average score linear regression for t+2h is: 0.965
The R2 score of the linear regression model for t+2h is r2 = 0.963

The average score linear regression for t+3h is: 0.932
The R2 score of the linear regression model for t+3h is r2 = 0.928

The average score linear regression for t+4h is: 0.893
The R2 score of the linear regression model for t+4h is r2 = 0.890

The average score linear regression for t+5h is: 0.852
The R2 score of the linear regression model for t+5h is r2 = 0.852

The average score linear regression for t+6h is: 0.811
The R2 score of the linear regression model for t+6h is r2 = 0.815
```

Performance Metrics

Model & Time Step	Average Score	R ² (Test)
Linear Regression t+1h	0.990	0.990
Linear Regression t+2h	0.965	0.963
Linear Regression t+3h	0.932	0.928
Linear Regression t+4h	0.893	0.890
Linear Regression t+5h	0.852	0.852
Linear Regression t+6h	0.811	0.815

Table 5.24: Performance of Linear Regression for Wind Power Generation

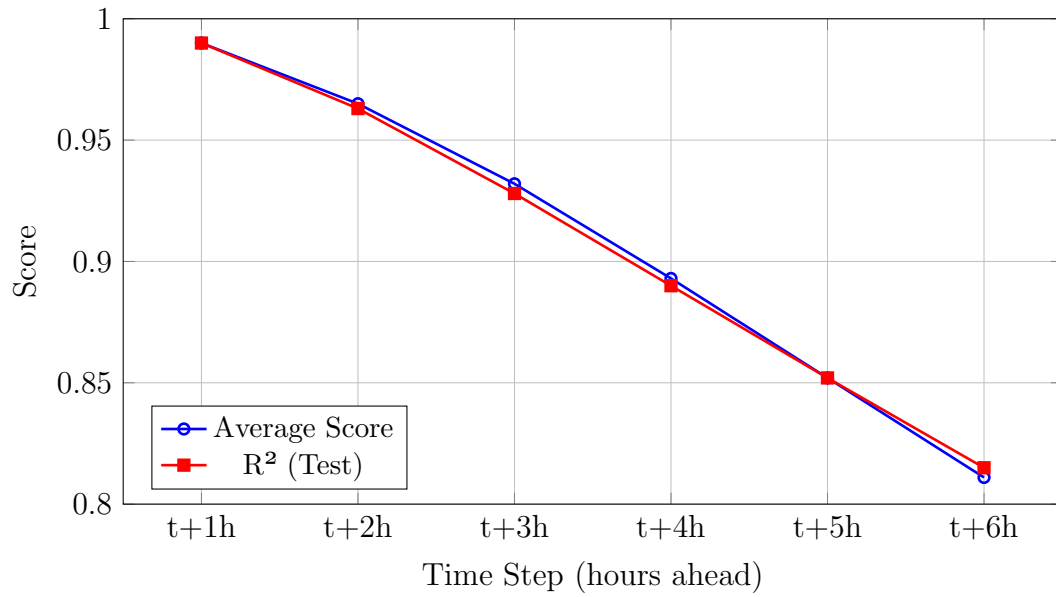


Figure 5.16: Performance of Linear Regression for Wind Power Generation at Different Time Steps

Training and validation of the LinearRegression model for selected features (n=8)

```

The average score linear regression for t+1h is: 0.990
The R2 score of the linear regression model for t+1h is r2 = 0.989

The average score linear regression for t+2h is: 0.965
The R2 score of the linear regression model for t+2h is r2 = 0.963

The average score linear regression for t+3h is: 0.931
The R2 score of the linear regression model for t+3h is r2 = 0.928

The average score linear regression for t+4h is: 0.893
The R2 score of the linear regression model for t+4h is r2 = 0.889

The average score linear regression for t+5h is: 0.852
The R2 score of the linear regression model for t+5h is r2 = 0.851

The average score linear regression for t+6h is: 0.810
The R2 score of the linear regression model for t+6h is r2 = 0.814

```

Performance Metrics

Model & Time Step	Average Score	R ² (Test)
Linear Regression t+1h	0.990	0.989
Linear Regression t+2h	0.965	0.963
Linear Regression t+3h	0.931	0.928
Linear Regression t+4h	0.893	0.889
Linear Regression t+5h	0.852	0.851
Linear Regression t+6h	0.810	0.814

Table 5.25: Performance of Linear Regression for Wind Power Generation

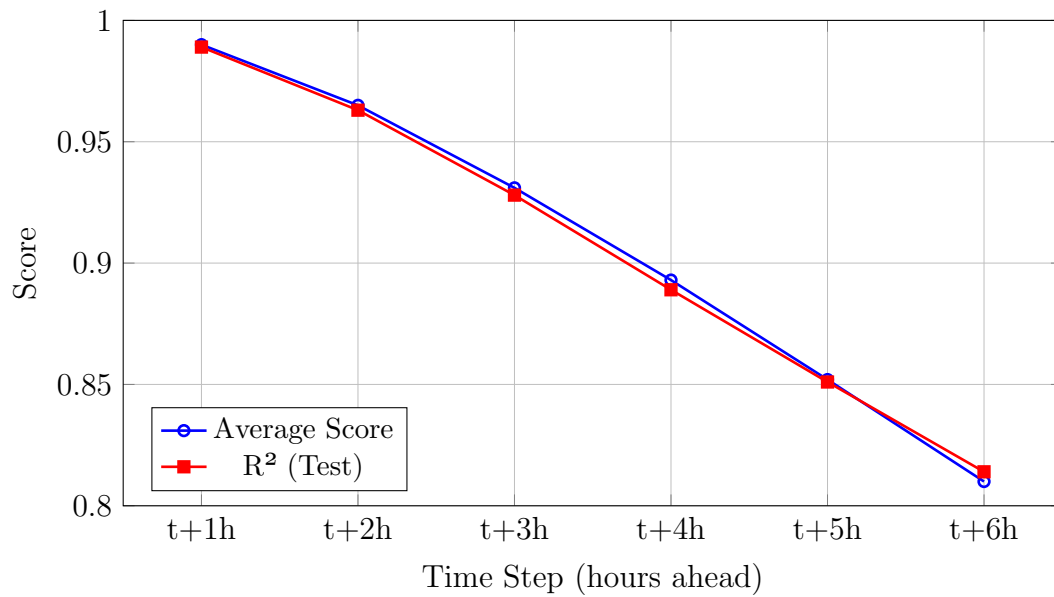


Figure 5.17: Performance of Linear Regression for Wind Power Generation at Different Time Steps

Training and validation of the LinearRegression model for selected features (n=2)

```

The average score linear regression for t+1h is: 0.990
The R2 score of the linear regression model for t+1h is r2 = 0.989

The average score linear regression for t+2h is: 0.964
The R2 score of the linear regression model for t+2h is r2 = 0.963

The average score linear regression for t+3h is: 0.930
The R2 score of the linear regression model for t+3h is r2 = 0.927

The average score linear regression for t+4h is: 0.891
The R2 score of the linear regression model for t+4h is r2 = 0.888

The average score linear regression for t+5h is: 0.850
The R2 score of the linear regression model for t+5h is r2 = 0.850

```

The average score linear regression for t+6h is: 0.808
The R2 score of the linear regression model for t+6h is r2 = 0.812

Performance Metrics

Model & Time Step	Average Score	R ² (Test)
Linear Regression t+1h	0.990	0.989
Linear Regression t+2h	0.964	0.963
Linear Regression t+3h	0.930	0.927
Linear Regression t+4h	0.891	0.888
Linear Regression t+5h	0.850	0.850
Linear Regression t+6h	0.808	0.812

Table 5.26: Performance of Linear Regression for Wind Power Generation

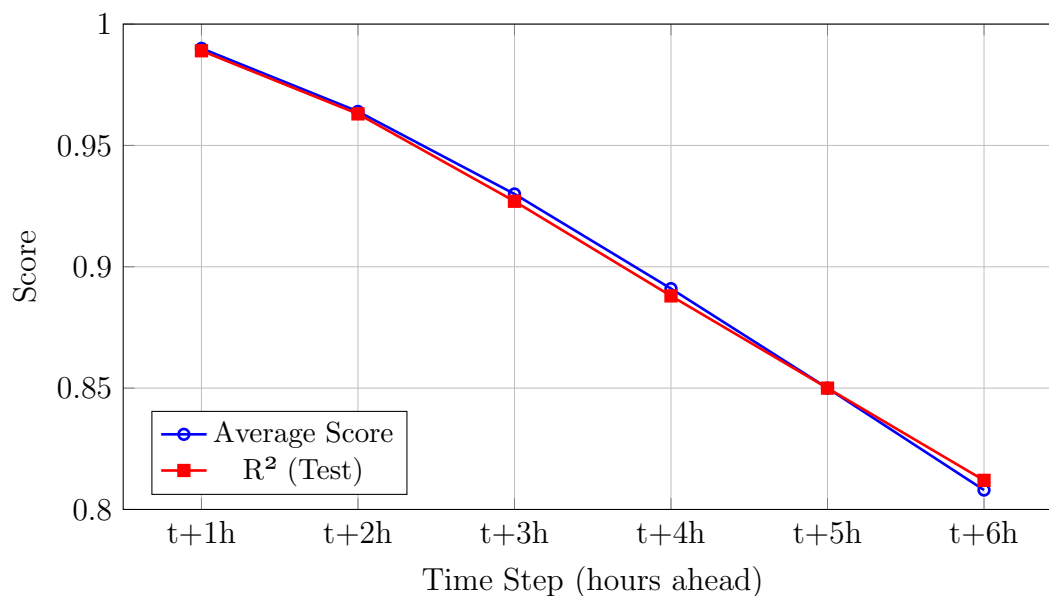


Figure 5.18: Performance of Linear Regression for Wind Power Generation at Different Time Steps

Training and validation of the LinearRegression model for selected features (n=1)

The average score linear regression for t+1h is: 0.990
The R2 score of the linear regression model for t+1h is r2 = 0.989

The average score linear regression for t+2h is: 0.964
The R2 score of the linear regression model for t+2h is r2 = 0.963

The average score linear regression for t+3h is: 0.930
The R2 score of the linear regression model for t+3h is r2 = 0.926

The average score linear regression for t+4h is: 0.891
The R2 score of the linear regression model for t+4h is r2 = 0.888

The average score linear regression for t+5h is: 0.849
The R2 score of the linear regression model for t+5h is r2 = 0.849

The average score linear regression for t+6h is: 0.807
The R2 score of the linear regression model for t+6h is r2 = 0.811

Performance Metrics

Model & Time Step	Average Score	R ² (Test)
Linear Regression t+1h	0.990	0.989
Linear Regression t+2h	0.964	0.963
Linear Regression t+3h	0.930	0.926
Linear Regression t+4h	0.891	0.888
Linear Regression t+5h	0.849	0.849
Linear Regression t+6h	0.807	0.811

Table 5.27: Performance of Linear Regression for Wind Power Generation

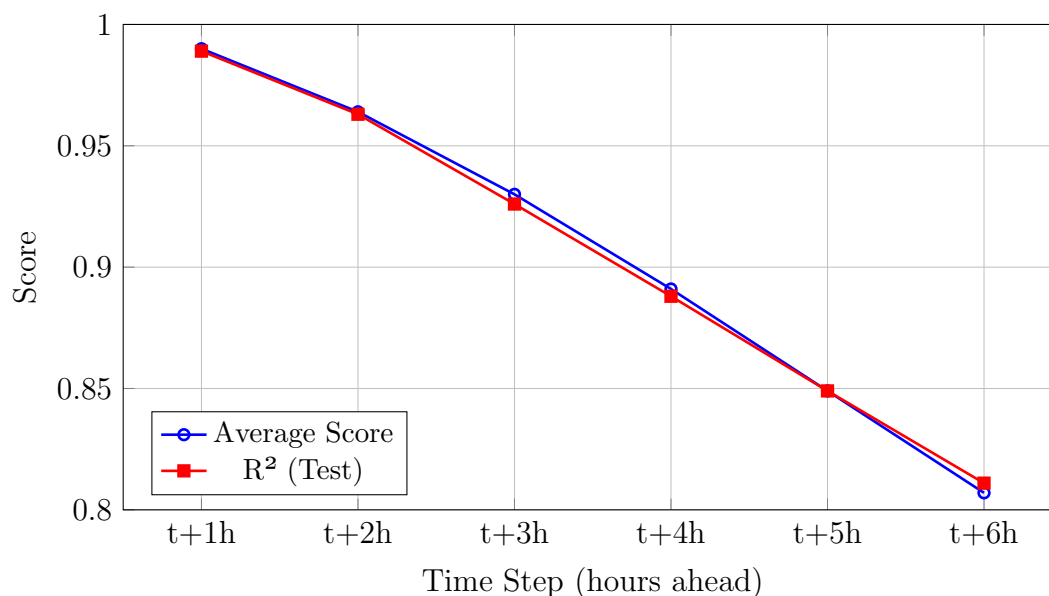


Figure 5.19: Performance of Linear Regression for Wind Power Generation at Different Time Steps

5.4.2 Random Forest Regression

Training and validation of the RandomForestRegression model for selected features (n=11)

The average score for random forest regression (100 decision trees) and
t+1h is: 0.990

The R2 score of the random forest regression (with 100 decision trees)
for t+1h is r2 = 0.990

The average score for random forest regression (100 decision trees) and
t+2h is: 0.968

The R2 score of the random forest regression (with 100 decision trees)
for t+2h is r2 = 0.967

The average score for random forest regression (100 decision trees) and
t+3h is: 0.942

The R2 score of the random forest regression (with 100 decision trees)
for t+3h is r2 = 0.941

The average score for random forest regression (100 decision trees) and
t+4h is: 0.916

The R2 score of the random forest regression (with 100 decision trees)
for t+4h is r2 = 0.918

The average score for random forest regression (100 decision trees) and
t+5h is: 0.891

The R2 score of the random forest regression (with 100 decision trees)
for t+5h is r2 = 0.898

The average score for random forest regression (100 decision trees) and
t+6h is: 0.870

The R2 score of the random forest regression (with 100 decision trees)
for t+6h is r2 = 0.883

Performance Metrics

Model & Time Step	Average Score	R ² (Test)
Random Forest Regression t+1h	0.990	0.990
Random Forest Regression t+2h	0.968	0.967
Random Forest Regression t+3h	0.942	0.941
Random Forest Regression t+4h	0.916	0.918
Random Forest Regression t+5h	0.891	0.898
Random Forest Regression t+6h	0.870	0.883

Table 5.28: Performance of Random Forest Regression for Wind Power Generation

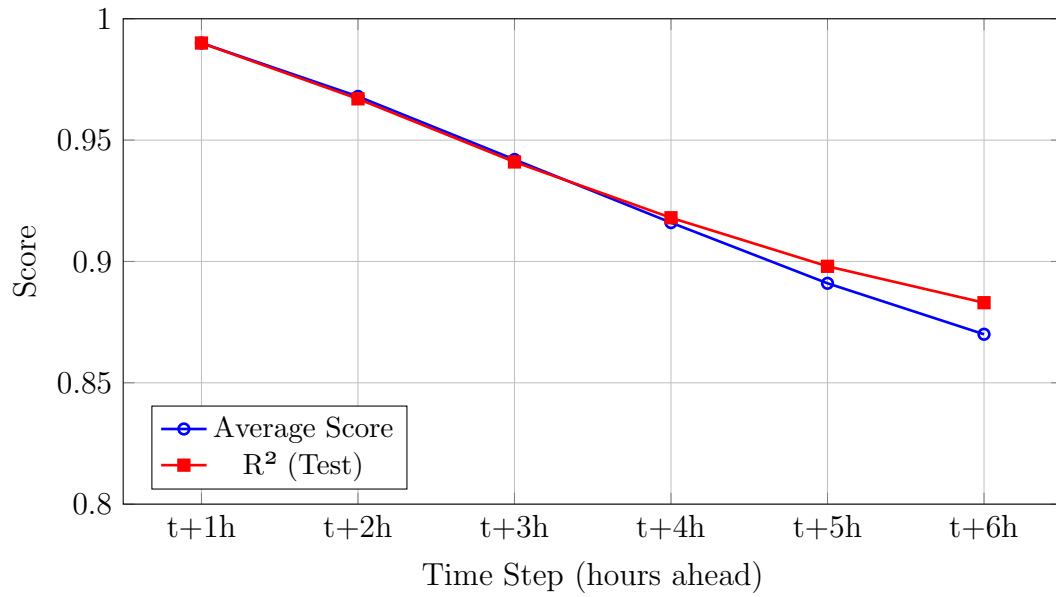


Figure 5.20: Performance of Random Forest Regression for Wind Power Generation at Different Time Steps

Training and validation of the RandomForestRegression model for selected features (n=8)

```

The average score for random forest regression (100 decision trees) and
t+1h is: 0.990
The R2 score of the random forest regression (with 100 decision trees)
for t+1h is r2 = 0.990

The average score for random forest regression (100 decision trees) and
t+2h is: 0.968
The R2 score of the random forest regression (with 100 decision trees)
for t+2h is r2 = 0.967

The average score for random forest regression (100 decision trees) and
t+3h is: 0.942
The R2 score of the random forest regression (with 100 decision trees)
for t+3h is r2 = 0.941

The average score for random forest regression (100 decision trees) and
t+4h is: 0.916
The R2 score of the random forest regression (with 100 decision trees)
for t+4h is r2 = 0.918

The average score for random forest regression (100 decision trees) and
t+5h is: 0.892
The R2 score of the random forest regression (with 100 decision trees)
for t+5h is r2 = 0.898

The average score for random forest regression (100 decision trees) and
t+6h is: 0.870
The R2 score of the random forest regression (with 100 decision trees)
for t+6h is r2 = 0.882

```

Performance Metrics

Model & Time Step	Average Score	R ² (Test)
Random Forest Regression t+1h	0.990	0.990
Random Forest Regression t+2h	0.968	0.967
Random Forest Regression t+3h	0.942	0.941
Random Forest Regression t+4h	0.916	0.918
Random Forest Regression t+5h	0.892	0.898
Random Forest Regression t+6h	0.870	0.882

Table 5.29: Performance of Random Forest Regression for Wind Power Generation

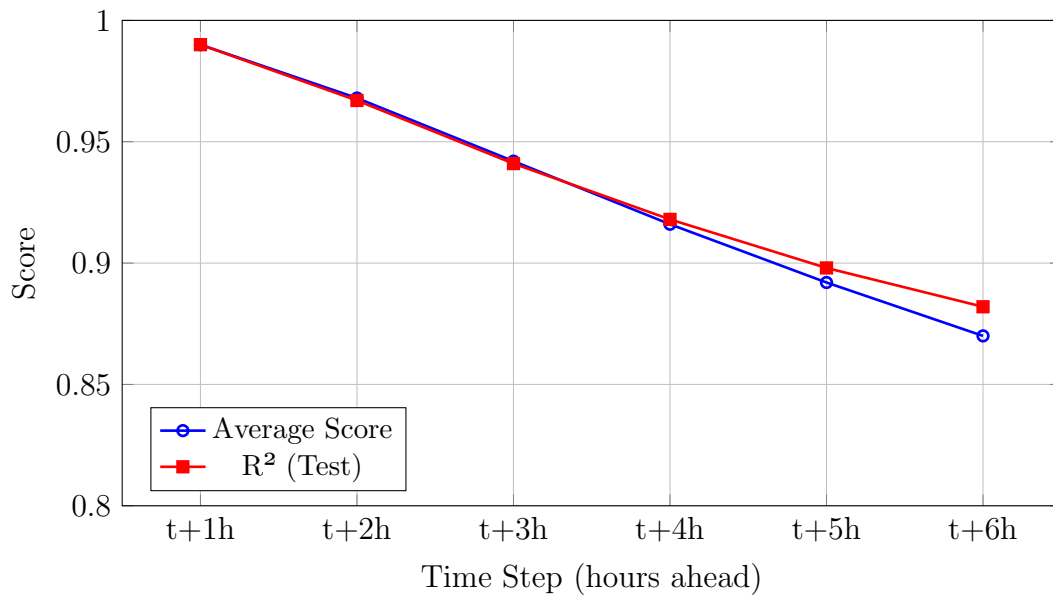


Figure 5.21: Performance of Random Forest Regression for Wind Power Generation at Different Time Steps

Training and validation of the RandomForestRegression model for selected features (n=2)

```

The average score for random forest regression (100 decision trees) and
t+1h is: 0.987
The R2 score of the random forest regression (with 100 decision trees)
for t+1h is r2 = 0.987

The average score for random forest regression (100 decision trees) and
t+2h is: 0.957
The R2 score of the random forest regression (with 100 decision trees)
for t+2h is r2 = 0.956

The average score for random forest regression (100 decision trees) and
t+3h is: 0.918
The R2 score of the random forest regression (with 100 decision trees)
for t+3h is r2 = 0.916

```

The average score for random forest regression (100 decision trees) and t+4h is: 0.872

The R2 score of the random forest regression (with 100 decision trees) for t+4h is r2 = 0.870

The average score for random forest regression (100 decision trees) and t+5h is: 0.825

The R2 score of the random forest regression (with 100 decision trees) for t+5h is r2 = 0.827

The average score for random forest regression (100 decision trees) and t+6h is: 0.777

The R2 score of the random forest regression (with 100 decision trees) for t+6h is r2 = 0.784

Performance Metrics

Model & Time Step	Average Score	R ² (Test)
Random Forest Regression t+1h	0.987	0.987
Random Forest Regression t+2h	0.957	0.956
Random Forest Regression t+3h	0.918	0.916
Random Forest Regression t+4h	0.872	0.870
Random Forest Regression t+5h	0.825	0.827
Random Forest Regression t+6h	0.777	0.784

Table 5.30: Performance of Random Forest Regression for Wind Power Generation

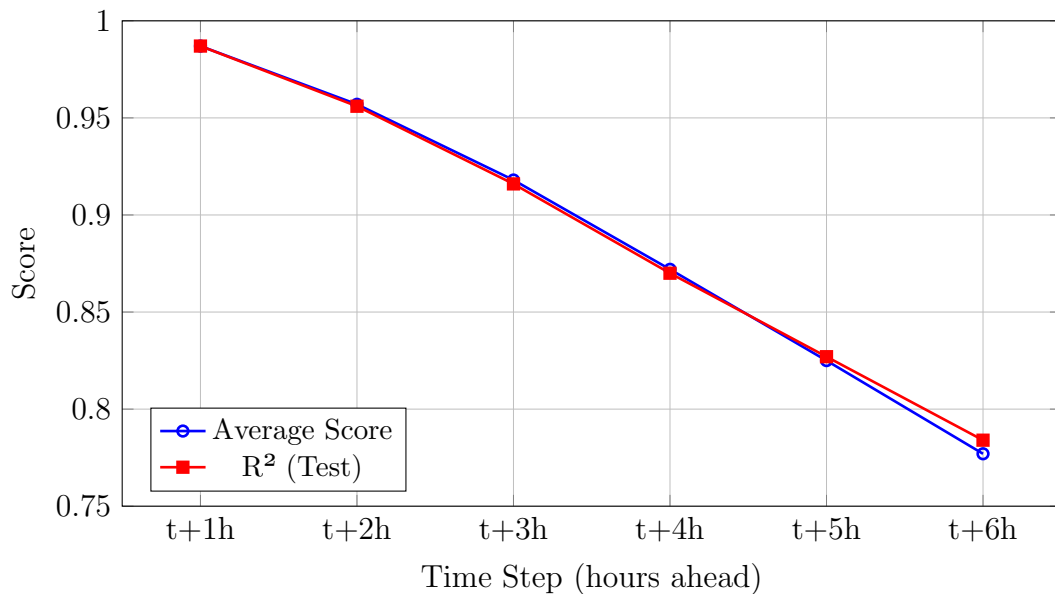


Figure 5.22: Performance of Random Forest Regression for Wind Power Generation at Different Time Steps

Training and validation of the RandomForestRegression model for selected features (n=1)

```

The average score for random forest regression (100 decision trees) and
t+1h is: 0.986
The R2 score of the random forest regression (with 100 decision trees)
for t+1h is r2 = 0.985

The average score for random forest regression (100 decision trees) and
t+2h is: 0.951
The R2 score of the random forest regression (with 100 decision trees)
for t+2h is r2 = 0.949

The average score for random forest regression (100 decision trees) and
t+3h is: 0.904
The R2 score of the random forest regression (with 100 decision trees)
for t+3h is r2 = 0.901

The average score for random forest regression (100 decision trees) and
t+4h is: 0.850
The R2 score of the random forest regression (with 100 decision trees)
for t+4h is r2 = 0.847

The average score for random forest regression (100 decision trees) and
t+5h is: 0.793
The R2 score of the random forest regression (with 100 decision trees)
for t+5h is r2 = 0.793

The average score for random forest regression (100 decision trees) and
t+6h is: 0.734
The R2 score of the random forest regression (with 100 decision trees)
for t+6h is r2 = 0.741

```

Performance Metrics

Model & Time Step	Average Score	R ² (Test)
Random Forest Regression t+1h	0.986	0.985
Random Forest Regression t+2h	0.951	0.949
Random Forest Regression t+3h	0.904	0.901
Random Forest Regression t+4h	0.850	0.847
Random Forest Regression t+5h	0.793	0.793
Random Forest Regression t+6h	0.734	0.741

Table 5.31: Performance of Random Forest Regression for Wind Power Generation

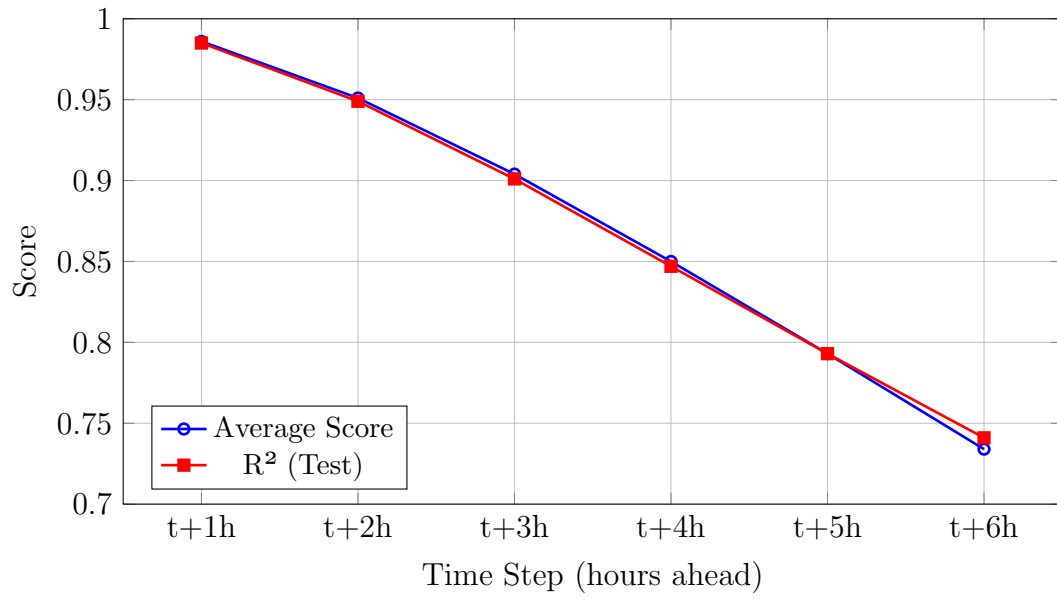


Figure 5.23: Performance of Random Forest Regression for Wind Power Generation at Different Time Steps

Performance Comparison (Full Feature Set)

Table 5.32: R^2 (Test) Comparison Between Models at t+1h and t+6h

Time Horizon	R^2 (Linear Regression)	R^2 (Random Forest)
t+1h	0.990	0.990
t+6h	0.815	0.883

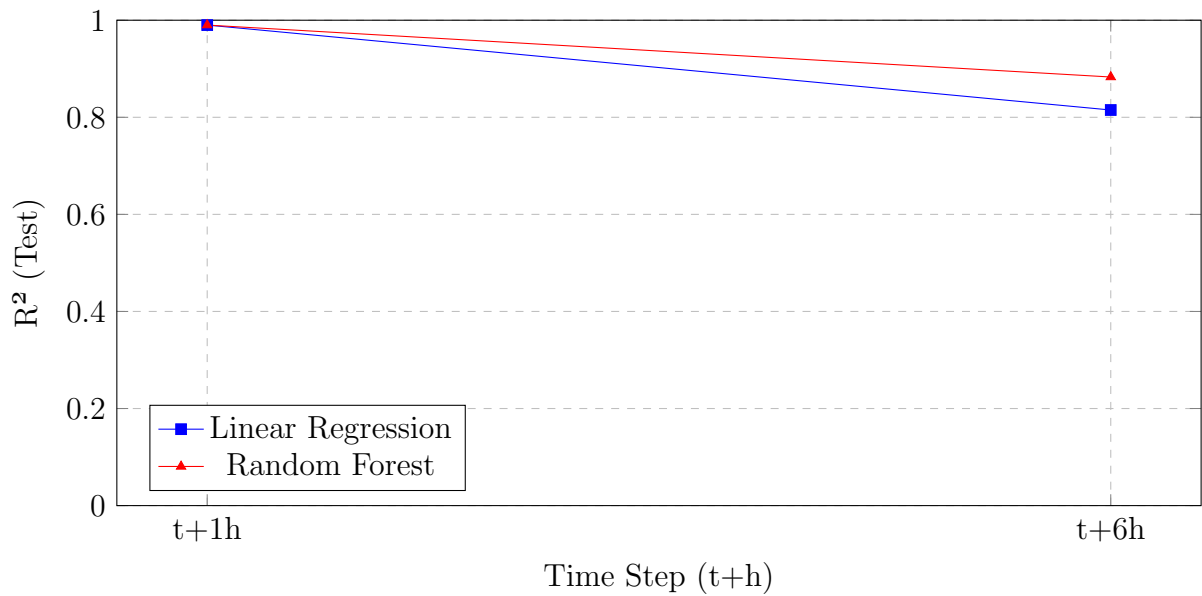


Figure 5.24: R^2 (Test) Comparison Between Models at t+1h and t+6h

The performance of wind power forecasting models, specifically **Linear Regression** and **Random Forest**, shows notable variations in prediction accuracy at different time horizons. At the **t+1h** time step, both models achieve perfect **R² (Test)** values of 0.990, indicating highly accurate short-term predictions. However, as the time horizon extends to **t+6h**, the performance of **Linear Regression** declines significantly, with its **R² (Test)** dropping to 0.815. On the other hand, **Random Forest** demonstrates a more gradual decrease, with its **R² (Test)** value remaining higher at 0.883 at **t+6h**. This suggests that while both models excel in the short term, **Random Forest** offers superior long-term forecasting performance, making it more suitable for applications that require longer time horizon predictions in wind power forecasting.

Chapter 6

Conclusions

Discussion

This thesis explored the application of machine learning tools to analyze and predict wind and solar power generation, addressing the challenges posed by their variability and intermittency. The study employed Linear Regression, Decision Tree Regression, and Random Forest Regression models to forecast renewable energy outputs using meteorological and historical energy data. The results demonstrated the effectiveness of these models, particularly Random Forest Regression, in capturing complex relationships between environmental factors and energy production, resulting in accurate short- and medium-term forecasts. These predictions are crucial for improving grid stability, optimizing resource allocation, and reducing operational costs. The research highlights the importance of robust data preprocessing, feature selection, and validation in developing reliable forecasting models. Despite the promising outcomes, challenges such as data limitations, computational requirements, and model interpretability were identified, underscoring areas for further investigation.

Future Work

Future research should address the limitations identified in this study to improve the scalability, precision, and adaptability of renewable energy forecasting models. Expanding datasets with real-time grid data, higher-resolution meteorological information, and global energy records can enhance prediction accuracy and robustness. Developing hybrid models that integrate machine learning with physical or statistical energy models could further boost performance. Advanced approaches like deep learning and ensemble learning should be explored to better handle complex interactions between environmental variables and energy outputs. Improving interpretability through explainable AI techniques can build trust in predictions, making models more practical. Efforts should also prioritize computational efficiency for large-scale deployment and extend forecasting to long-term horizons for strategic planning. Finally, applying these methods across diverse regions and integrating them with energy policies and grid systems will provide valuable insights into global scalability, supporting a more sustainable energy future.

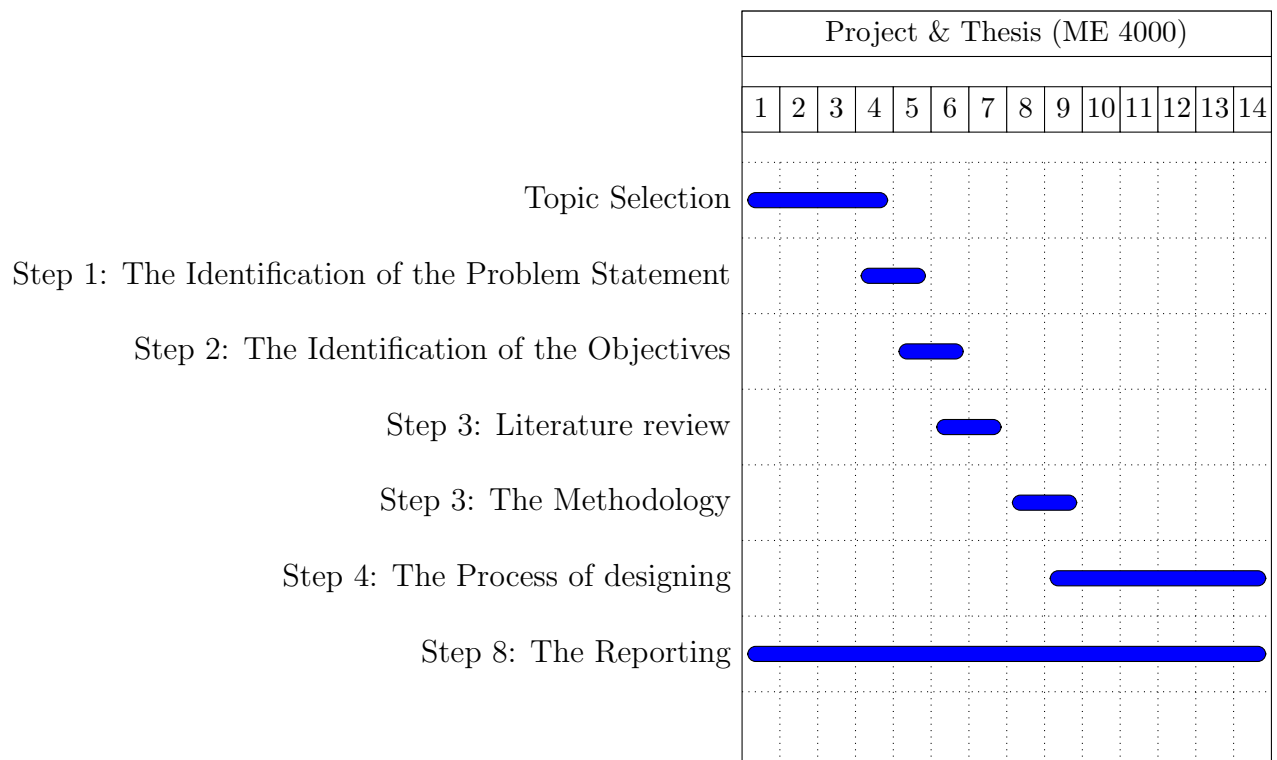
References

- [1] C. Voyant *et al.*, “Machine learning methods for solar radiation forecasting: A review,” in *Renewable energy*, vol. 105, Elsevier, 2017, pp. 569–582.
- [2] O. L. Petchey *et al.*, “The ecological forecast horizon, and examples of its uses and determinants,” *Ecology letters*, vol. 18, no. 7, pp. 597–611, 2015.
- [3] A. Qazi *et al.*, “Towards sustainable energy: A systematic review of renewable energy sources, technologies, and public opinions,” *IEEE access*, vol. 7, pp. 63 837–63 851, 2019.
- [4] V. Lara-Fanego, J. Ruiz-Arias, D. Pozo-Vázquez, F. Santos-Alamillos, and J. Tovar-Pescador, “Evaluation of the wrf model solar irradiance forecasts in andalusia (southern spain),” in *Solar Energy*, 8, vol. 86, Elsevier, 2012, pp. 2200–2217.
- [5] J. Yan, Y. Liu, S. Han, Y. Wang, and S. Feng, “Reviews on uncertainty analysis of wind power forecasting,” *Renewable and Sustainable Energy Reviews*, vol. 52, pp. 1322–1330, 2015.
- [6] S. Aggarwal and L. Saini, “Solar energy prediction using linear and non-linear regularization models: A study on ams (american meteorological society) 2013–14 solar energy prediction contest,” in *Energy*, vol. 78, Elsevier, 2014, pp. 247–256.
- [7] J. He, D. Zhou, and Q. Gu, “Logarithmic regret for reinforcement learning with linear function approximation,” in *International Conference on Machine Learning*, PMLR, 2021, pp. 4171–4180.
- [8] G. Reikard, “Predicting solar radiation at high resolutions: A comparison of time series forecasts,” in *Solar energy*, 3, vol. 83, Elsevier, 2009, pp. 342–349.
- [9] B. Gülmez, “Stock price prediction with optimized deep lstm network with artificial rabbits optimization algorithm,” *Expert Systems with Applications*, vol. 227, p. 120 346, 2023.
- [10] M. Diagne, M. David, P. Lauret, J. Boland, and N. Schmutz, “Review of solar irradiance forecasting methods and a proposition for small-scale insular grids,” in *Renewable and Sustainable Energy Reviews*, vol. 27, Elsevier, 2013, pp. 65–76.
- [11] A. Hammer, D. Heinemann, E. Lorenz, and B. Lückehe, “Short-term forecasting of solar radiation: A statistical approach using satellite data,” in *Solar Energy*, 1-3, vol. 67, Elsevier, 1999, pp. 139–150.
- [12] R. H. Inman, H. T. Pedro, and C. F. Coimbra, “Solar forecasting methods for renewable energy integration,” in *Progress in energy and combustion science*, 6, vol. 39, Elsevier, 2013, pp. 535–576.
- [13] M. Bolinger, R. Wiser, and E. O’Shaughnessy, “Levelized cost-based learning analysis of utility-scale wind and solar in the united states,” *Iscience*, vol. 25, no. 6, 2022.

- [14] A. Dhar, M. A. Naeth, P. D. Jennings, and M. G. El-Din, “Perspectives on environmental impacts and a land reclamation strategy for solar and wind energy systems,” *Science of the total environment*, vol. 718, p. 134 602, 2020.
- [15] M. I. Jordan and T. M. Mitchell, “Machine learning: Trends, perspectives, and prospects,” *Science*, vol. 349, no. 6245, pp. 255–260, 2015.
- [16] A. Ahmed and M. Khalid, “A review on the selected applications of forecasting models in renewable power systems,” *Renewable and Sustainable Energy Reviews*, vol. 100, pp. 9–21, 2019.
- [17] P. Sampaio, P. Saraiva, and A. Guimarães Rodrigues, “Iso 9001 certification forecasting models,” *International Journal of Quality & Reliability Management*, vol. 28, no. 1, pp. 5–26, 2011.
- [18] B. H. Thacker, S. W. Doebeling, F. M. Hemez, M. C. Anderson, J. E. Pepin, and E. A. Rodriguez, “Concepts of model verification and validation,” 2004.
- [19] J. Czocher, G. Stillman, and J. Brown, “Verification and validation: What do we mean?,” *Mathematics Education Research Group of Australasia*, 2018.
- [20] J.-Z. Wang, Y. Wang, and P. Jiang, “The study and application of a novel hybrid forecasting model—a case study of wind speed forecasting in china,” *Applied Energy*, vol. 143, pp. 472–488, 2015.
- [21] A. Y. Barrera-Animas, L. O. Oyedele, M. Bilal, T. D. Akinosho, J. M. D. Delgado, and L. A. Akanbi, “Rainfall prediction: A comparative analysis of modern machine learning algorithms for time-series forecasting,” *Machine Learning with Applications*, vol. 7, p. 100 204, 2022.
- [22] T. G. Dietterich, “Machine-learning research,” in *AI magazine*, 4, vol. 18, 1997, pp. 97–97.

Appendix A

GANTT Chart for Spring 2023



Appendix B

GANTT Chart for Fall 2023

