SYMBOLIC REGRESSION IN ENERGY ENGINEERING

CHRISTIAN CHINWEIKE UKACHUKWU

MSc. Sustainable Energy Engineering

Department of Energy Science

Faculty of Engineering

Lund University

Supervisor(s)

RIXIN YU

Submitted in partial fulfilment of the requirements for the Award of Master of Science in Sustainable Energy Engineering.

June 4, 2024

ABSTRACT

The global call for more sustainable energy development and natural resource management hinges on both the technical ability and social capacity to harness the potentials from these resources.

Symbolic Regression in Energy Engineering explores leveraging machine learning to solve renewable energy challenges arising from the notorious volatility of resources. Symbolic regression, a machine learning technique, uncovers mathematical models from data without predefined structures, thus providing interpretable and accurate models. This thesis investigates symbolic regression's applications in energy engineering, particularly in predicting renewable energy outputs such as wind speed against power output, which are highly variable and unpredictable. The study utilizes genetic programming to evolve symbolic expressions that model complex relationships within wind energy systems. The methodology includes collecting and preprocessing data, training symbolic regression algorithms, and evaluating models using various metrics. The results demonstrate symbolic regression's effectiveness in creating predictive models that outperform traditional regression methods in both accuracy and interpretability. By capturing intrinsic data patterns, symbolic regression offers a promising approach to enhancing the reliability and efficiency of renewable energy systems. The discussion highlights the advantages of symbolic regression over traditional methods, including better model interpretability and reduced human bias, and suggests future research directions to further improve this technique's applicability in energy engineering.

This abstract captures the essence of the thesis, highlighting the importance of symbolic regression in addressing renewable energy challenges, the methodology employed, and the significance of the results obtained.

ACKNOWLEDGEMENT

I acknowledge God for being God and Guide over me in all; the Swedish Institute for the lifetime opportunity to be part of this program through their sponsorship and training; My family, for their support and prayers; My Fiancée, Pempherani Namacha for her encouragement throughout this journey; My teachers, Jens, Per, Hesam and my supervisor Rixin, for showing me clearly that energy is beyond a career; And my friends, Noah, Kiran, Emmanuel, Ajibola, Shadrach, Mirembe and Kelly for cheering me in success and lifting me in downtimes.

Table of Contents

Table of Figures

List of Tables

1.0 Introduction and Theoretical Background

1.1 Background of Symbolic Regression

People's perceptions of reality have been shaped by their observation of the world around them, which most times does not reflect the realities of a different space. Therefore, research seeks for realities that are widely applicable in multiple spaces. Its a common knowledge that some renewable energy sources like wind and solar, have been notorious for being volatile and highly unpredictable, hence, several assumptions have been made while describing some parameters necessary for harnessing the energy therein which warrants research. Research follows a well-defined path, often called the scientific method. This method involves three key phases:

- **Observation:** This is where scientists gather information about the world around them. They collect data through experiments, measurements, or even simple observations (Sobh et al, 2006).
- **Hypothesis Generation:** Based on their observations, researchers propose a possible explanation for the data they collected. This explanation, called a hypothesis, should be clear and testable. A good hypothesis often allows for predictions about future observations of the same system (Chua W et al, 2019).
- **Hypothesis Validation:** This phase is crucial! Scientists design experiments or gather new data to see if the hypothesis holds true (Miller et al, 2002). Does the proposed explanation actually match reality?

1.2 Symbolic Regression: Unveiling the Equation Behind the Data

One technique used in hypothesis generation as mentioned above is called Symbolic Regression (SR). Here, researchers aim to discover a mathematical formula, often written as an equation, that best describes the relationship within a given set of data (Keren et al 2023). Imagine a natural phenomenon influenced by specific factors (measurements or features). Symbolic Regression attempts to uncover the mathematical equation that connects these measurements to the observed outcome.

Eschewing a priori model specification, symbolic regression circumvents the introduction of human bias or limitations in domain knowledge. It endeavours to unveil the inherent relationships within the dataset by allowing the data's intrinsic patterns to dictate the appropriate models, rather than imposing human-centric, mathematically convenient structures. The fitness function guiding the model evolution incorporates not only error metrics (ensuring accurate data prediction) but also specialized complexity measures (Weng, B. et al. 2020). This ensures the resulting models capture the underlying data structure in a human-interpretable manner. This facilitates the reasoning process and bolsters the probability of gleaning insights into the system generating the data. Furthermore, it enhances generalizability and extrapolation behaviour by mitigating overfitting. Notably, accuracy and simplicity can be addressed as distinct regression objectives, resulting in optimal solutions forming a Pareto front. Alternatively, they can be combined into a single objective through a model selection principle like minimum description length.

It is a method of regression analysis that searches the space of mathematical expressions to find the best model for a given dataset, both in terms of accuracy and simplicity. Unlike

traditional regression methods, SR doesn't start with a predefined model. Instead, it explores a vast space of mathematical expressions to find the most accurate and interpretable representation of the data. It is a non-standard method that does not require a pre-specified model structure, but instead infers the model from the data. Symbolic regression has been applied to renewable energy, specifically in the context of photovoltaic (PV) power forecasting.

Recent research has also explored the integration of symbolic regression with other techniques, such as machine learning and data mining, to improve its performance and applicability. For example, a study published in Nature Communications used symbolic regression to guide the design of new oxide perovskite catalysts with improved oxygen evolution reaction (OER) activities (Weng, B. et al.,2020)

Liron Simon explained symbolic regression as a powerful tool for understanding complex relationships between variables, and it is particularly useful in situations where the underlying dynamics are hard to model from physical principles or simplified models are needed. It can be applied to various fields, including physics, where it is used to formulate accurate symbolic expressions even from data with high noise levels (Liron et al, 2023). The method is also useful in physics-informed modelling, where it aids in formulating an accurate symbolic expression and offers a novel a-priori feature selection process to test different hypotheses efficiently.

1.3 How Symbolic Regression works.

It builds a set of random formulars to represent the relationship between variables in data. These variables are some independent variables and their resultant dependent variables. These formulars are meant to predict the initial set of data as a generalized correlation between the dependent and independent variables. It carries out this by following a sequence of genetic selection. Symbolic regression utilizes the principles of genetic programming to autonomously discover model structures and their associated parameters. This approach commences with a meticulously chosen collection of mathematical operators, functions, variables, and constants that act as the fundamental building blocks. These elements are then subjected to a process of random combination and recombination, echoing biological evolution, and repeated over numerous iterations. The objective is to evolve a set of equations that effectively represent the inherent dynamics of the system. A fitness function serves as the guiding force, selectively retaining the most successful solutions – those that demonstrate the most faithful representation of the data as measured by a designated error metric – for subsequent "reproduction" and "mutation" steps. Conversely, solutions deemed inadequate are discarded. This iterative process continues until a predetermined level of accuracy is achieved. Symbolic regression, empowered by genetic programming, has emerged as a significant tool within the domain of industrial empirical modelling (P. Valsaraj et al, 2019).

1.3.1 Symbolic expression

Symbolic regression powered by genetic programming leverages LISP programming methos of expressing the relationship between variables in a simple correlation. The fundamental building blocks are symbolic expressions, or s-expressions for short. These s-expressions serve a dual purpose, representing both the program itself and the data it manipulates. An s-expression can take one of two forms: an atom or a list. Atoms, the basic syntactic units of Lisp, encompass both numeric values and symbols. These symbols can be constructed from letters, numbers, and even non-alphanumeric characters.

For instance, looking at the equation below.

$$Z = \frac{Y^2 + 7W - 3}{X} \ldots.(1)$$

It can be written in a form that reflect the operations that connects the independent variables W, X, Y, and the integers to produce the dependent variable, Z. Using the LISP programming s-expression, the integers and dependent variables are referred to as the atoms. The atoms are arranged in a way that it connects to their operations.

The following can be written in the form as below:

$$Z = (\div\ (-\ (+\ (\times\ Y\ Y)\ (\times\ 7\ W)\ 3)\ X) \ldots.(2)$$

For each set of the atoms, some operations are used to explain the interrelationship usually placed at the left corner, immediately before the set of atoms. For $(\times\ 7\ W)$ This multiplies y y itself while for $(\times\ Y\ Y)$ This multiplies w by 7 and finally both set of atoms are connected by a "+" operation.

Graphically, the following can be represented in a generational tree as follows:

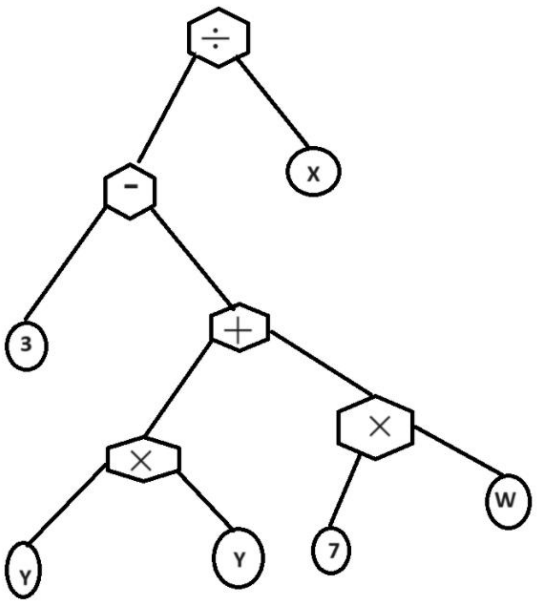

Figure 1 A graphical representation of gene expression.

The tree graph known as Syntax tree consists of several nodes and leaves. The nodes are the functions and operations that connects the variables and constants while the leaves or terminal nodes are the independent variables and integers that make up the correlation.

1.3.2 Evaluating Performance in Genetic Programming for Symbolic Regression

Within the domain of genetic programming (GP) applied to symbolic regression, the concept of fitness serves as a crucial metric for assessing the effectiveness of an evolved program (Fleck P. et al, 2024). Similar to other machine learning paradigms, GP necessitates the establishment of a well-defined objective function, which dictates whether maximizing or minimizing the value leads to an optimal solution, specific to the problem at hand (Fleck P. et al, 2024).

Regression Problems:

Mean Squared Error (MSE): This metric quantifies the average squared difference between predicted and actual values (M, Padhma. 2023).

Root Mean Squared Error (RMSE): The RMSE is derived from the MSE by taking its square root, offering a more interpretable measure of prediction error in the same units as the target variable (M, Padhma. 2023).

Classification Problems:

Logarithmic Loss: This function measures the performance of a classification model by penalizing the model for assigning low probabilities to correct classifications and high probabilities to incorrect classifications (Anushruthika, 2023)

Binary Cross-Entropy Loss: This metric is another common choice for evaluating binary classification models, specifically designed for problems with two possible outcomes (Anushruthika, 2023).

1.3.3 Initialization of the parameters

The tabulation below outlines pertinent parameters along with their respective descriptions important to the initialization phase of Genetic Programming. These parameters are fundamental in configuring the symbolic operation:

Table 1 A list of symbolic regression parameters and their functions

Parameter	Description
population_size	Denotes the quantity of programs partaking in the inaugural generation and each subsequent generation thereafter.
function_set	Signifies the ensemble of mathematical functions available for utilization within the operation.
generations	Specifies the maximum number of iterations until termination of the programs.
stopping_criteria	Establishes a predefined criterion, often a numerical threshold representing an optimal score, for halting the program.
p_crossover	A percentage parameter dictating the selection of a random subtree from the victorious program in a tournament, to be replaced in ensuing generations through crossover.
p_subtree_mutation	Represents a percentage parameter facilitating the reintroduction of

	defunct functions and operators into the population, fostering diversity.
p_hoist_mutation	A percentage parameter responsible for excising genetic material from tournament victors.
p_point_mutation	A percentage parameter selecting random nodes from the tournament champion to be replaced, contributing to variation.
max_samples	Augments the subsampling endeavours on data, thus enhancing the diversity of perspectives on individual programs.
parsimony_coefficient	Governs the penalty applied to the fitness measure during selection, thereby regulating program complexity.

1.4 The difference between symbolic regression and other forms of statistical regression

Traditional regression methods focus on optimizing the parameters within a pre-defined model structure. In contrast, symbolic regression employs an inductive approach, inferring the model itself from the data. This entails the simultaneous discovery of both the model structure and its parameters.

This data-driven approach comes with a significant challenge: an exponentially larger search space. Symbolic regression not only contends with an infinite space of potential expressions, but also the possibility of infinitely many models perfectly fitting a finite dataset (assuming no constraints on model complexity). Consequently, symbolic regression algorithms may require substantially more computational resources compared to traditional regression techniques to identify suitable models and parameterizations. This computational burden can be mitigated by restricting the algorithm's building blocks based on prior knowledge of the underlying system that generated the data. However, the decision to employ symbolic regression ultimately hinges on the extent of this domain knowledge.

Despite these challenges, the very characteristic that presents a vast search space also offers advantages. Evolutionary algorithms, commonly used in symbolic regression, rely on diversity within the population to effectively explore this space. This often results in the identification of a collection of high-performing models (along with their respective parameter sets). Analysing this ensemble can provide deeper insights into the underlying process, allowing the user to select an approximation that best balances accuracy and interpretability based on their specific needs.

1.5 Importance of Symbolic Regression in Energy Engineering

Symbolic regression is a powerful tool in energy engineering, as it can help to identify the relationships between various factors and energy consumption or production. This technique is particularly useful in time series analysis, where it can be used to model and predict energy trends over time. It is based on the principles of genetic programming, which allows it to spot complex relationships and patterns in data that may not be immediately apparent. This makes it a valuable tool for understanding the behaviour of energy systems and predicting future trends.

In energy engineering, symbolic regression has been used to model the behaviour of various systems, including renewable energy systems, energy efficiency systems, and power

generation systems. For example, it has been used to predict electricity consumption based on factors such as temperature, humidity, and time of day (Lei Gan a et al.,2022).

One of the key advantages of symbolic regression is its ability to provide interpretable results. This means that it can help to identify the factors that are driving energy consumption or production, which can be useful for improving the efficiency of energy systems and reducing their environmental impact.

Symbolic regression is a valuable tool in energy engineering, as it can help to identify complex relationships in energy data and provide interpretable results. Its use in time series analysis and integration with other techniques has the potential to further improve its performance and applicability in this field.

1.6 Application of symbolic regression

The scope of symbolic regression in energy engineering includes its application in various aspects of energy engineering, such as renewable energy, energy efficiency, and power generation. It has been used to guide the design of new materials with improved activities in energy engineering, develop model predictive control systems for HVAC scheduling, and model the behaviour of various energy systems.

- Modelling complex relationships: Symbolic regression can be used to model complex relationships between various factors and energy consumption or production. This is particularly useful in time series analysis, where it can be used to predict energy trends over time.
- Improving efficiency: By integrating physical laws and mathematical models into data-driven approaches, symbolic regression can improve the accuracy and efficiency of energy systems. This can help to reduce the computational burden and improve the overall performance of energy systems.
- Explaining relationships: Symbolic regression can provide interpretable results, which can help to explain the relationships between various factors and energy consumption or production. This can be useful for improving the efficiency of energy systems and reducing their environmental impact.
- Integrating with other techniques: Recent research has explored the integration of symbolic regression with other techniques, such as machine learning and data mining, to improve its performance and applicability. For example, physics-based symbolic regression has been used to improve the accuracy of power flow modeling and analysis.

Generally, symbolic regression is a powerful tool in energy engineering, as it can help to identify complex relationships in energy data and provide interpretable results. Its use in time series analysis and integration with other techniques has the potential to further improve its performance and applicability in this field.

2.0 Research Questions and Objectives

Some renewable energy sources like wind and solar energy are considered volatile because of the high seasonality and unpredictable nature (Hoen, B.D, 2022), hence, the needs to leverage artificial intelligence in modelling their expected performance and power output. These models will be necessary starting from the planning phase of the energy development process through the consumption of energy.

In this research, we looked at two properties of wind energy and addresses some questions thereof:

- Power Output: A very important aspect of planning wind energy development is to observe the wind profile in the area in terms of wind speed, wind density, and the corresponding wind power. This had been for long estimated with the formular:

$$P_{air} = 0.5\rho_{air}AU^3 \ldots.(3)$$

Where P_{air} is the Power available in the air, ρ_{air} is the average air density which is usually 1.225kg/m^3 between the heights of 0 and 100m above the ground. A is the sweep area in m^2 and U is the average instantaneous wind speed in m/s.
A is calculated from the rotor radius/diameter as in the equation $A = \pi R^2$, where R is the distance from the centre of the hub to the tip of the blade.
The figure below:

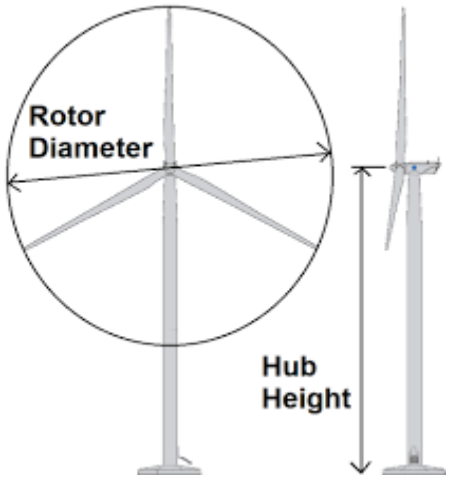

Figure 2 An illustration of the dimensions of a wind turbine.

The equation for power in the air made from clearly measurable parameters and the air density is also generalized regardless of the temperature, elevation, topography, and humidity of the area, hence does not reflect the reality on the terrain.

- Noise estimation: One of the major concerns of developing a wind energy project is the noise that comes from the turbine. To tackle this, several governments and local authorities have instituted measures to limit the noise to the barest minimum. In Europe, the noise expected to be heard from a wind turbine is 35-45dB when measured from a distance of 300m (Chiu, CH., 2021). Compared to other noise

sources, as in the figure below, this noise limit is meant to be very bearable and would not cause any health challenges.

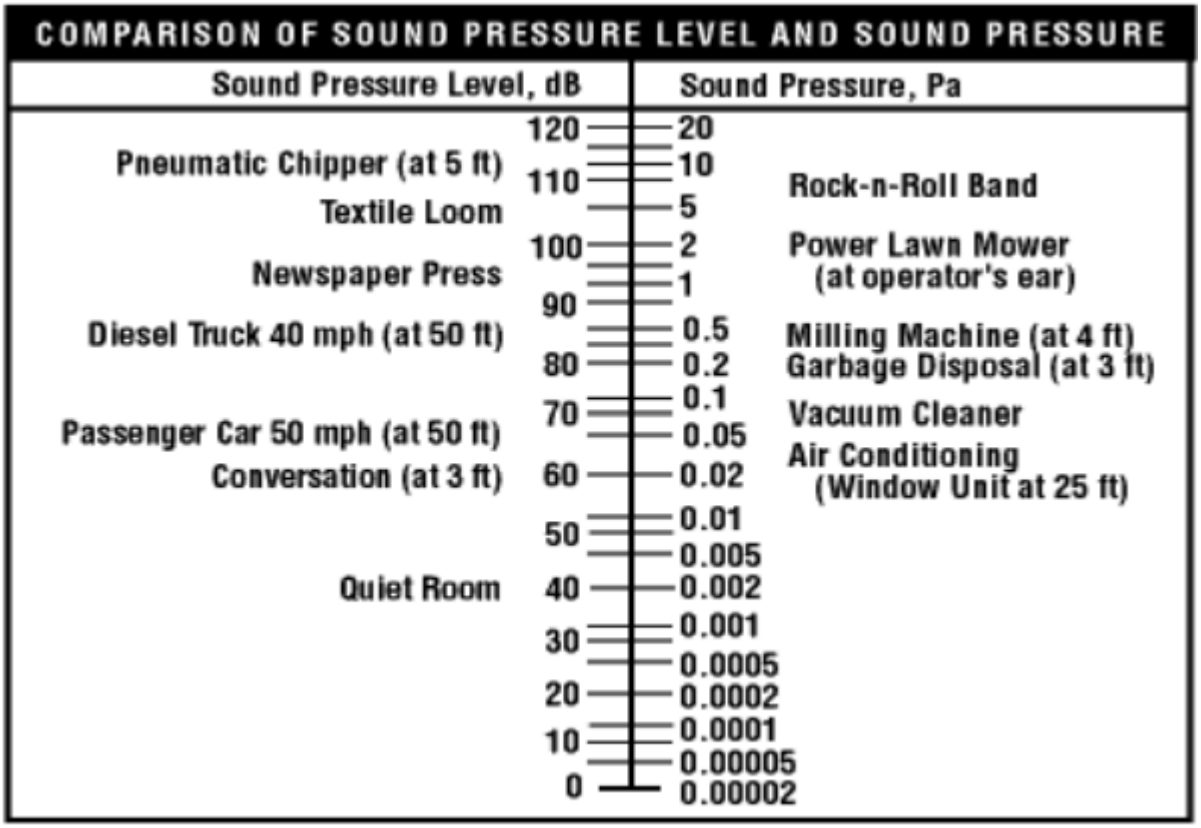

Figure 3 A comparative chart of Sound Power of wind turbines and other acoustic systems.

To achieve this, the designers of the wind turbine will have to optimize the sound dampening systems of the turbine based on the location of the turbine and the distance between the turbine and residential areas.

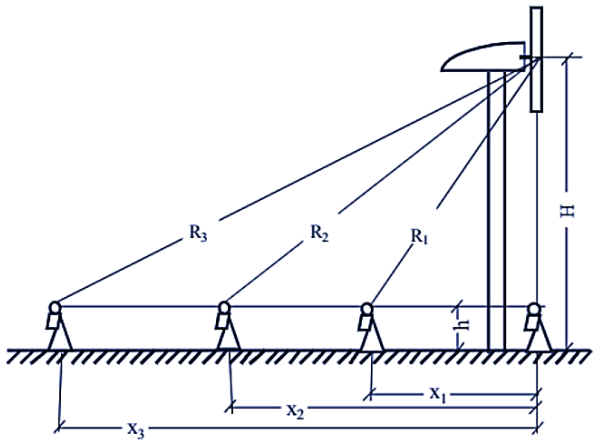

Figure 4 An illustration of measurement patterns for estimating wind turbine sound power effects on measuring tools.

The model below has been used extensively to estimate the surrounding noise based on the distance between the turbine and the measuring tool, the height of the turbine and the sound emitted by the turbine (Nick Jenkins et al, 2017):

$$L_w = L_p + 10log_{10}(2\pi R^2) + 0.005R \text{.....} (4)$$

Where:

L_w is the Sound Power at the wind turbine.

L_p is the sound power at the point of measurement.

R is the distance between the hub of the turbine and the point of measurement. This can also be measured from the centre of one blade at an angle of deviation between the reference point of measurement and the foot of the tower, usually denoted as r. To calculate R as follows$R = \sqrt{H^2 + D^2}$. Where D is the distance between the foot of the tower and the point of measurement and H is the distance height of the hub from the ground.

In trying to make this estimate, the two main parameters of concern are the Sound Power emitted from the turbine and the distance from the point of measurement. This is putting optimization into consideration where the expected measurement will be the average range in regulation (35 to 45dB) which is 40dB and the average height of wind turbine is 90m. Hence the equation for the base truth becomes:

$$L_w = 40 + 10log_{10}(2\pi(90^2 + D^2)) + 0.005\sqrt{90^2 + D^2} \text{.....} (5)$$

The key objective of this research is to design and train a machine learning model, Symbolic Regression to generate explainable equations for highly dynamic data emanating from energy systems. At the end of the research, explainable models were generated for estimating the power output of a turbine from the wind speed and the optimal sound level of a wind turbine based on the distance between the turbine and the point of measurement. In achieving these objectives, the following questions were addressed:

i. What are the parameters important to train the symbolic regression model to predict the relationship between variables in data to be applicable to real wind field measurements?
ii. What are mathematical explainable models and correlations which can be derived from a field data that can predict wind power output for that field over a range of wind speed and the optimal design sound power of a wind turbine over a range of distances?
iii. How does the prediction outputs from symbolic regression model compare to some other popular machine learning models?
iv. What are the limitations to applying symbolic regression effectively in energy engineering and potential improvements that can be made?

2.1 Scopes and Limitations

The process of creating a generalizable and interpretable mathematical expression underlying a set of data stems from the availability of reliable and well cleaned datasets. Symbolic regression has a very good ability in evaluating dataset with consistent trends and for this, several outliers may be ignored in the process of fitting the model into the dataset.

The research was carried out using different data sources with different measurement inconsistencies hence, the generated equations will be particular to each dataset and their respective sources.

3.0. Methodology

3.1 Data Collection and Preprocessing

The data for this research was sourced from two processes. These processes required to develop symbolic regression models and mathematical expressions are data-intensive and requires a lot of historic, simulated, or orchestrated data to function effectively. Google cloud GPU was used through ***Colab*** to carry out the training of the model.

3.1.1 Existing equations

For the Turbine Sound Power- Distance measurements, a range of possible distances between the turbines and measurement point (usually in the residential area or location of concerns) is simulated to calculate the proximate design sound power emitted from a wind turbine. This is also used to estimate the sound effect of one turbine to another in a case of cluster.

$$L_w = 40 + 10log_{10}(2\pi(90^2 + D^2)) + 0.005\sqrt{90^2 + D^2}$$

3.1.2 Field Data details and source

The field data used for this research is a 2023 full year hourly log of wind turbine Texas AW3000/77 cleaned and managed at the National Renewable Energy Laboratory (NREL) in Texas, United States. It was made public under license derived from the American Wind Energy Association. The specification of the turbine is as tabulated below:

Parameters	Value
Manufacturer	Acciona Windpower
Wind Class	IEC/NVN IIA
Name	AW3000/77
Rotor diameter	111m
Rated output	3600KW
Hub Height	80m
Location	Texas U.S
Cut-in wind speed	3.0m/s
Cut-off wind speed	25m/s
Maximum rotor speed	17.15m/s

The data was collected and pre-processed in a manner that it exhibits pristine data completeness and is free from any extraneous noise, factors that typically impede forecasting endeavours with actual datasets and detract from the overarching research objective. After creating and training functional symbolic regression algorithm with simulated data from exciting equations, the field data was therefore fed into the system to produce an interpretable mathematical formular for future uses in that field for estimating the energy use.

3.2 Symbolic Regression Algorithms

Symbolic regression, harnessed within the *gplearn* framework, leverages the power of genetic programming (GP) paradigms to unveil mathematical expressions that optimally

represent the inherent relationships encapsulated within a dataset. This approach hinges on the iterative evolution of symbolic expressions, akin to mathematical formulae, across successive generations. The guiding principles for this evolution are derived from natural selection and genetic recombination, mirroring the very processes that drive biological adaptation.

I. Initialization: The process commences with the establishment of a population comprised of random symbolic expressions. These expressions are typically depicted as tree structures, wherein each node embodies an operator (e.g., addition, multiplication) or an operand (e.g., variable, constant). In each case as mentioned in the section above, two types of

II. Evaluation: Each symbolic expression within the population is meticulously evaluated against the provided dataset. This evaluation quantifies the expression's fitness, often measured by a pre-defined error metric such as mean squared error (MSE) or mean absolute error (MAE). Essentially, this step gauges how well a particular expression aligns with the observed data.

III. Selection: The algorithm strategically selects symbolic expressions for procreation, exhibiting a propensity towards those boasting superior fitness scores. This process emulates natural selection, where individuals better suited to their environment (those with higher fitness) are more likely to pass on their traits.

IV. Genetic Operators: The chosen symbolic expressions undergo a series of genetic operations, mirroring the mechanisms observed in biological reproduction. These operations include:

- Crossover: This process entails the exchange of subtrees between parent expressions, akin to the exchange of genetic material during sexual reproduction. By fostering the exchange of building blocks, crossover facilitates the exploration of novel and potentially superior expressions.

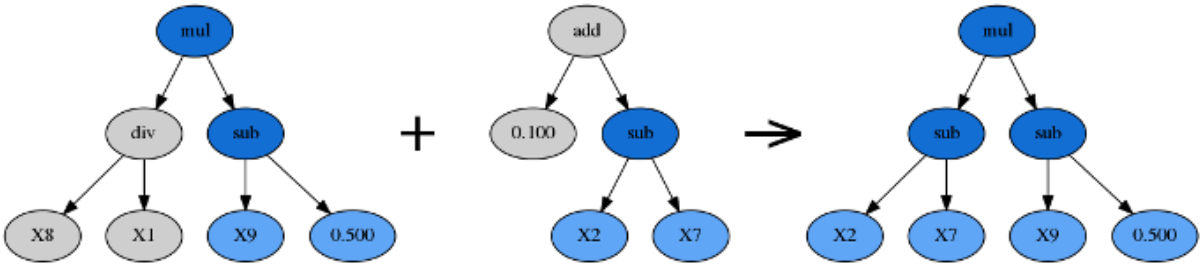

Figure 5 Cross over illustration.

- Mutation: This operation introduces deliberate yet controlled alterations to the structure or parameters of individual expressions. Mutation acts as a catalyst for

diversification, preventing the population from stagnating at a local optimum and potentially leading to the discovery of improved solutions.

In gplearn symbolic regression, there are basically three types of mutation that can be available:

i. **Subtree Mutation (Aggressive Restructuring):** This mutation strategy, controlled by the parameter p_subtree_mutation, is characterized by its substantial impact on individual genomes. It injects a high degree of novelty by replacing random subtrees within the winner's program with entirely new genetic material. This process can potentially reintroduce functionalities or operators that may have been lost during prior evolutionary cycles, thereby promoting genetic diversity within the population. In essence, a subtree is chosen randomly from the champion program and substituted with a novel subtree generated de novo.

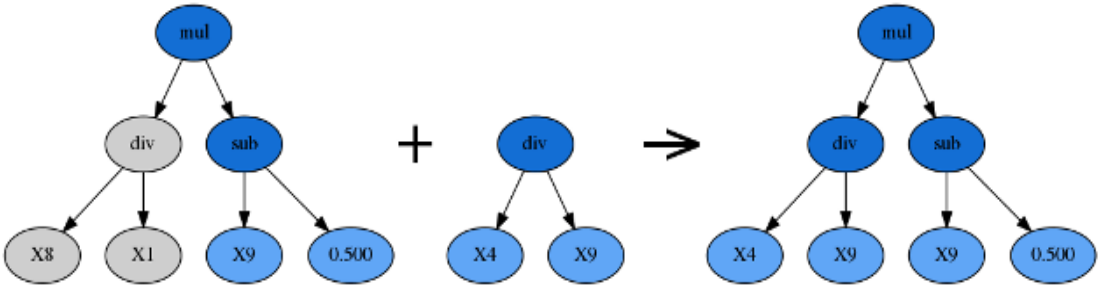

Figure 6 Subtree Mutation

ii. **Hoist Mutation (Parsimony Enforcement):** This mutation technique, governed by the p_hoist_mutation parameter, serves the primary function of mitigating program bloat. Bloat refers to the uncontrolled growth of program size, potentially leading to inefficiencies. Hoist mutation achieves bloat control by strategically removing genetic material from the winner's program. It selects a subtree from the champion program and subsequently chooses a subsubtree within it. This subsubtree is then "hoisted" to replace the original subtree, resulting in a more parsimonious offspring for the next generation.

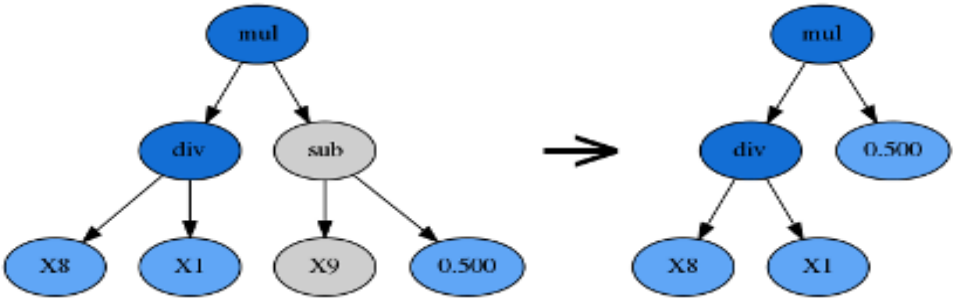

Figure 7 Hoist Mutation

iii. **Point Mutation (Fine-Grained Modification):** Point mutation is likely the most prevalent mutation operator employed in genetic programming. Similar to subtree

mutation, it has the potential to reintroduce lost functionalities or operators into the population, thus maintaining genetic diversity. This mutation strategy involves selecting random nodes from the winner's program and replacing them. Terminals are substituted with other terminals, and functions are replaced with functionally equivalent functions possessing the same arity (number of arguments) as the original node. The resultant program constitutes the offspring for the subsequent generation.

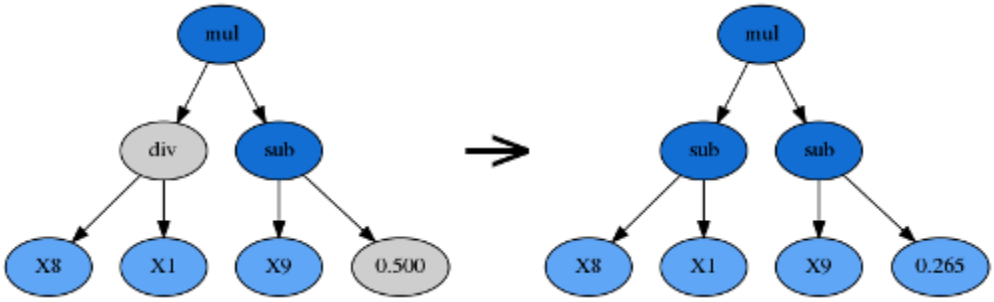

Figure 8 Point Mutation

V. Replacement: The newly generated offspring expressions are strategically introduced into the population, either entirely or partially, based on pre-defined criteria. These criteria might encompass elitism (ensuring the best existing expressions are preserved) or generational turnover (gradually replacing older expressions with fitter offspring).

VI. Termination Criteria: The evolutionary process persists until a pre-defined termination criterion is met. Common termination criteria include reaching a maximum number of generations, achieving a satisfactory fitness threshold (indicating a sufficiently accurate model), or observing convergence within the population (implying a lack of further improvement).

VII. Best Solution Extraction: Upon termination, the algorithm gleans the most optimal model from the final population. This "best" expression, identified based on its superior fitness score, represents the symbolic regression model that most effectively captures the underlying relationships within the dataset.

By meticulously traversing this iterative evolutionary process, *gplearn* strives to unearth concise symbolic expressions that can accurately encapsulate the intricate relationships embedded within a dataset. These expressions not only empower researchers to make predictions but also furnish valuable insights into the functional form of the relationships between input variables and the target variable. The ability to unveil the underlying mathematical structure fosters a deeper understanding of the system under investigation.

3.3 Training Parameter Setting

Each of the fed-in datasets were divided into two portions: Training and Testing dataset. The training dataset is 70% of the data while the testing is 30%. This is to achieve optimal performance and also for in-context evaluation. Building upon the core principles of symbolic regression within ***gplearn,*** we can delve deeper by considering a specific parameter configuration:

Population Size (population_size=5000 to 10000):

The population size dictates the number of candidate symbolic expressions evaluated in each generation. Here, a population size of 10000, usually used for the first test with defined function set, signifies a diverse pool of expressions, fostering a more comprehensive exploration of the solution space. While 5000 signifies a less diverse pool of expressions in the solution space. While larger populations enhance the likelihood of discovering optimal solutions, they also incur increased computational costs.

Function Set ('add', 'sub', 'mul','log','cos','sin'):

This is a collection of operators that form the mathematical expression being generated through symbolic regression. This parameter governs the building blocks available for constructing symbolic expressions. The specific functions included within *function_set* significantly influence the expressiveness of the model and its ability to capture the underlying relationships in the data. The optimal choice for *function_set* depends on the anticipated functional form of the target variable.

Generations (generations=20 to 40):

The number of generations determines the extent of the evolutionary process. Here, 20 generations represents a moderate exploration, striking a balance between achieving a reasonable solution and computational efficiency, while 40 represents a higher level since there is no specific function set being used, hence the need to review more generations. For more complex relationships, increasing generations might be necessary. Conversely, for simpler problems, fewer generations might suffice.

Stopping Criteria (stopping criteria=0.01):

This parameter establishes the fitness threshold that signifies a satisfactory solution. With a stopping criterion of 0.01, the algorithm terminates when the mean squared error (MSE) between the predicted and actual values falls below 0.01. This indicates a high degree of accuracy in the discovered model.

Genetic Operators:

- **Crossover Probability (p_crossover=0.7):** This parameter sets the probability of performing crossover, where subtrees are exchanged between parent expressions. A

value of 0.7 indicates a high likelihood of crossover, promoting the exploration of diverse combinations of building blocks.

- **Subtree Mutation Probability (p_subtree_mutation=0.1):** This value governs the probability of introducing modifications to the structure of individual expressions through subtree mutations. Here, a 0.1 probability suggests a balanced approach, allowing for some variation while maintaining the integrity of the expressions.
- **Hoist Mutation Probability (p_hoist_mutation=0.05):** This parameter controls the likelihood of performing hoist mutations, which involve selecting a random subtree and inserting it elsewhere in the expression. A value of 0.05 signifies a low probability, promoting stability and preventing excessive disruption of promising expressions.
- **Point Mutation Probability (p_point_mutation=0.1):** This value dictates the probability of introducing minor alterations to individual elements within expressions (e.g., changing an operator or operand). A 0.1 probability allows for controlled exploration of nearby solutions in the search space.

Additional Parameters:

- **Max Samples (max_samples=0.9):** This parameter influences the proportion of the training data used for fitting the model in each generation. Here, using 90% of the data (max_samples=0.9) provides a sufficient training set while reserving a portion for potential validation.
- **Verbose (verbose=1):** Setting verbose to 1 enables the algorithm to provide informative messages during the evolutionary process, offering insights into progress and performance.
- **Parsimony Coefficient (parsimony_coefficient=0.01):** This parameter introduces a penalty for overly complex expressions, favoring models with a balance between accuracy and simplicity. A value of 0.01 signifies a slight preference for parsimonious models.
- **Random State (random_state=0):** Setting a fixed random state ensures reproducibility, allowing for consistent results when rerunning the symbolic regression process.

By carefully considering these parameters, researchers can tailor the gplearn symbolic regression algorithm to their specific data and problem at hand. This parameterization not only influences the efficiency of the search process but also impacts the interpretability and complexity of the discovered model.

3.4. Evaluation Metrics

The assessment of symbolic regression models necessitates a comprehensive evaluation strategy encompassing both their predictive accuracy and structural fidelity. This two-pronged approach ensures the discovered expressions not only generate accurate predictions but also unveil the inherent mathematical relationships within the data.

3.4.1 Quantifying Predictive Performance: Numeric Metrics

The cornerstone of numeric metrics lies in gauging the model's ability to replicate the target variable's values. These metrics establish the discrepancy between the actual data points and the corresponding predictions generated by the symbolic expression. Commonly employed numeric metrics include:

- **Mean Squared Error (MSE) and Mean Absolute Error (MAE):** This metric calculates the average of the differences between the predicted and actual values. It can be absolute or squared. A lower MSE and MAE signifies a superior fit, indicating the model's proficiency in approximating the target variable.
- **Prediction Score:** This metric computes the average of the absolute differences between the predicted and actual values. Maximizing prediction translates to a model with enhanced predictive accuracy.
- **R-squared:** This metric quantifies the proportion of variance in the target variable that can be attributed to the model. An R-squared value approaching 1 suggests a strong correlation between the predicted and actual values, signifying a model that effectively captures the underlying trends.

3.4.2 Delving Deeper: Symbolic Metrics

While numeric metrics provide valuable insights into predictive accuracy, symbolic metrics delve a layer deeper. They assess the structural resemblance between the discovered symbolic expression and the true functional form that generated the data. Symbolic metrics transcend mere prediction accuracy, aiming to capture the essence of the mathematical relationship between the input variables and the target variable. Evaluating symbolic similarity presents a unique challenge, and various methodologies have been established:

- **Tree Edit Distance:** This metric calculates the minimum number of edit operations (insertions, deletions, substitutions) required to transform one expression tree into another. A lower edit distance signifies a closer structural resemblance between the discovered expression and the actual underlying function.
- **Normalized Edit Distance:** This metric refines the Tree Edit Distance by incorporating the size and depth of the expression trees, offering a more standardized comparison for expressions of varying complexity.
- **Parsimony:** This metric prioritizes simpler expressions with fewer operators and operands. Less complex expressions are generally considered more interpretable and desirable, facilitating a deeper understanding of the discovered relationships.

3.4.3 Selecting the Optimal Evaluation Strategy

The selection of appropriate evaluation metrics hinges on the specific objectives of the symbolic regression task. If the primary goal is achieving accurate predictions, numeric metrics like MSE or MAE might be sufficient. However, for tasks where interpretability and comprehension of the underlying relationships are paramount, symbolic metrics like normalized edit distance or parsimony become indispensable considerations. By employing a multifaceted evaluation strategy that incorporates both numeric and symbolic metrics, researchers can gain a holistic understanding of the efficacy and interpretability of their symbolic regression models.

4.0 Results and Analysis

In this section, the results, and outputs from several cases as well as their respective analysis were made.

4.1. Predictive Modelling of Sound Power from wind turbine against Distance

In preparing the model for a field scenario, some testings were carried out on simulated data to estimate the performance and understand the best performing values for parameters and their respective effects on the mathematical expressions generated by them.

4.1.1. Data Description

The data for this part of the research was simulated from 1000 samples of the distance between the foot of the wind turbine mast to the point of measurement (Dc) ranging from 100 m to 1000 m. Using equation (5) as described, the value of Turbine Sound Power (Lt) that will theoretically amount to average of 40dB when the turbine is placed at distance, Dc was derived. Plotted below is the Dc-Lt chart.

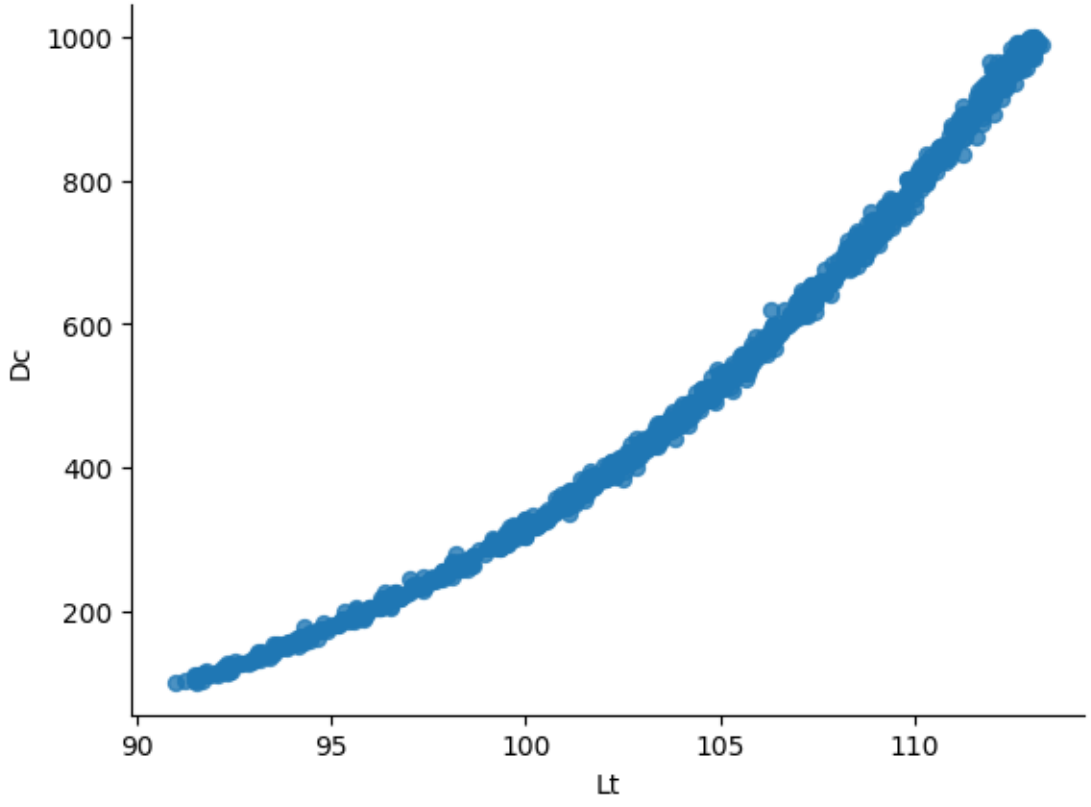

Figure 9: Chart of simulated sound power data - Dc-Lt chart.

Manne Friman (2011) estimated the average sound power of wind turbine of height 90m to be 101.8 dBA which corresponds to approximately 300 m away from residential area according to the data represented in the chart.

4.1.2. Symbolic Regression Models for simulated data

On the training dataset, which is 70% of the entire data, the following values were used to prepare the ***gplearn*** model for predicting and validation using the test data.

Figure 10 Training Parameter for the simulated dataset.

Parameter	**First Value**	**Second Value**
population_size	5000	5000
function_set	'add','sub', 'mul','log','cos','sin'	N/A
generations	40	20
stopping_criteria	0.01	0.01
p_crossover	0.7	0.7
p_subtree_mutation	0.1	0.1
p_hoist_mutation	0.05	0.05
p_point_mutation	0.1	0.1
max_samples	0.9	0.9
verbose	1	1
parsimony_coefficient	0.01	0.01
random_state	0	0

4.1.3. Model analysis with simulated data.

4.1.3.1 First training and testing of simulated data with predefined arithmetic operators.

Using the first parameter, the training was conducted on training data and the following table shows the performance and the mathematical expression as the output.

Table 2 output mathematical expression table for the simulated dataset.

Training time	607.365 Seconds
Output Mathematical expression: where X_0 is the sound, Lt produced by the wind turbine.	$0.0720000000000001X_0(X_0 - 0.54) - 1.732252X_0 + (0.0720000000000001X_0(X_0 + 0.54) - 0.07082341344X_0 - (0.099 - X_0)\cos(0.044X_0) + 0.00348192(X_0 - 0.54)(2X_0 + 0.526) + 0.09672 \cdot (0.18X_0(X_0 - 0.54) + 0.382)\cos(0.044X_0) - 0.09672\sin(X_0) + 0.33755784206944)\cos(0.044X_0) - \log(X_0) - \log(X_0 + 0.526) + \log(\cos(0.044X_0)) - 0.551472948$

In addition to interpretable mathematical expression that comes with training a symbolic regression model, a graphical representation of the output mathematical expression, known as Gene Expression, is possible and it enables the user to visually understand the generational interactions that led to the output. Below is the gene expression for the above mathematical model. In the representation below, the output operators as specified in the ***function_set*** is represented as nodes while the sub operators, variables and integers are the leaves, showing

several generations of iterations needed to produce a reliable mathematical expression for the set of data.

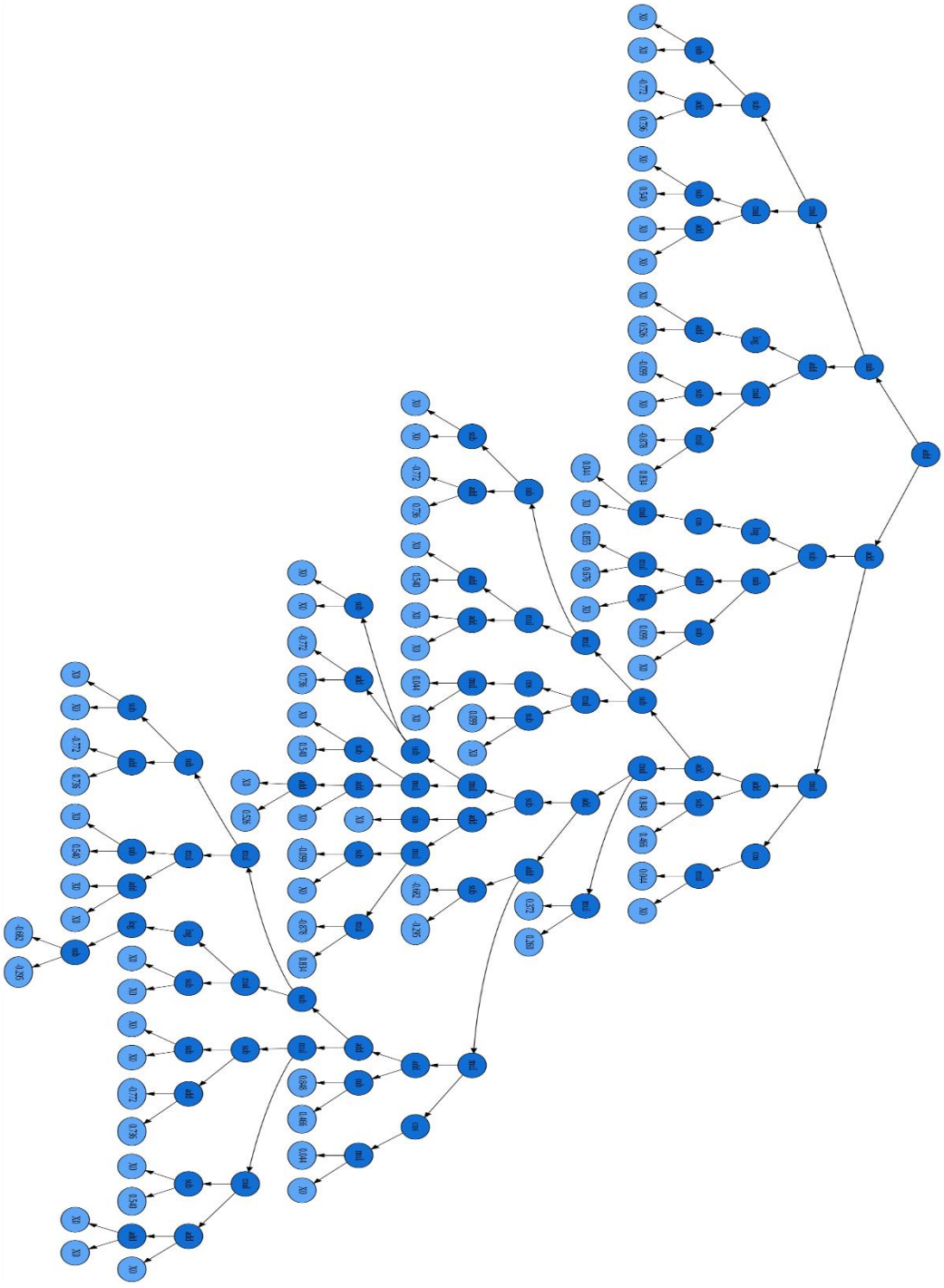

Figure 11: Gene expression of the first training in case 1.

While the output performance score is very high and above the statistical confidence level of certainty, it appears cumbersome and too long, hence the need for next trial of training.

Table 3 Test performance for first testing on simulated data-CASE 1

Prediction time of test data	0.0041408538818359375 Seconds
Performance score	0.9983602524874564

Comparing the ground truth of the test data to the predicted outcomes, a plot was made to show the level of alignment and to justify the performance in the table 3 above. A measure of residual dispersion was also plotted below the main curve to check the extent of deviation of the predicted value from the ground truth.

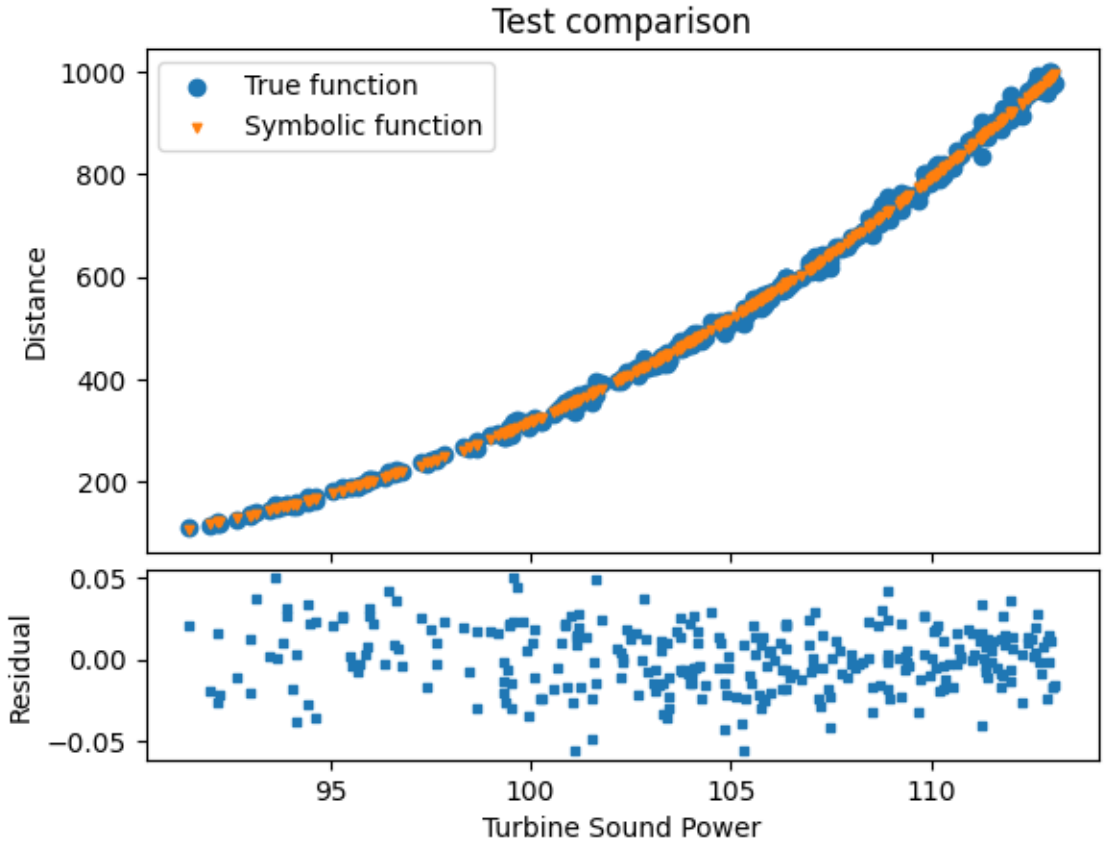

Figure 12 First test comparison for simulated sound power.

4.1.3.2 Second training and testing of simulated data with undefined operators.

Using the parameters in the second value column of the table, there, it is observed that there was no provision of predefined mathematical operators which gives the model an open ability to leverage any operator within the model for execution of training. After 335.956 seconds of training, the following mathematical expression was generated by the symbolic regression model on the training data.

$$-\frac{\left(8.54700854700855X_0+4.85470085470086\right)\left(\frac{X_0^2}{X_0-0.647}+0.862X_0+0.395\right)}{0.874X_0\left(-0.305167-\frac{0.663}{X_0}\right)+\frac{-X_0-0.604}{-0.305167-\frac{0.663}{X_0}}}+$$

$$\frac{0.000207882347076548\left(-X_0-0.042\right)^2\left(X_0-0.946\right)^2\cdot 0.416025\left(X_0+0.328682170542636\right)^2\cdot\left(0.085X_0^2+X_0-2.30193650793651-\frac{0.044619\cdot\left(2.484X_0-\frac{0.945}{X_0}\right)}{X_0\left(-2X_0-1.57868304278922-\frac{0.297}{X_0}\right)}\right)}{X_0^2\left(X_0+0.914\right)^2}$$

To represent the above using gene expression as was done in the first case, the output was as below:

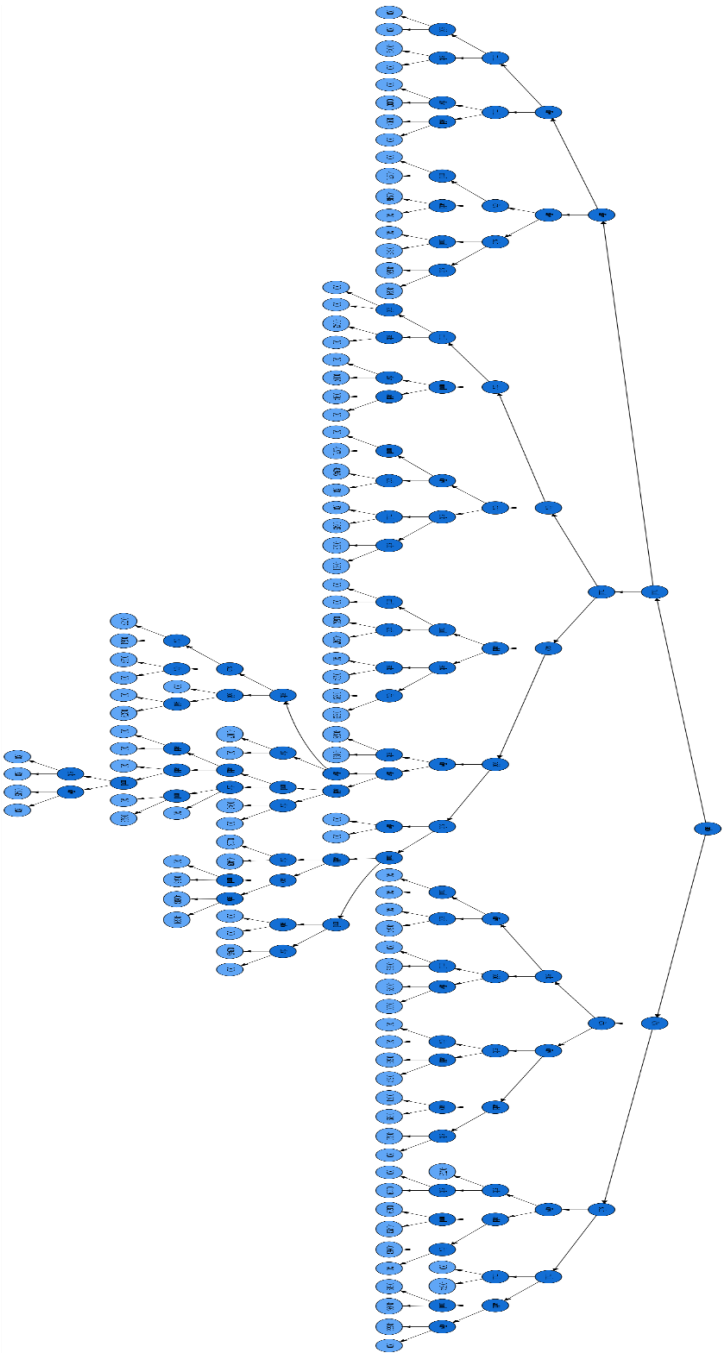

Figure 13 Gene expression for the second test of simulated data in case 1.

At the end of the testing, a clearly different outcomes were observed from the outputs. Below are the results that were observed:

Table 4 Test performance forsecond testing on simulated data

Prediction time of test data	0.006814002990722656 seconds
Performance score	0.9857463519752963

Just like the first case, a plot of the ground truth superimposed by model outputs was plotted as shown below.

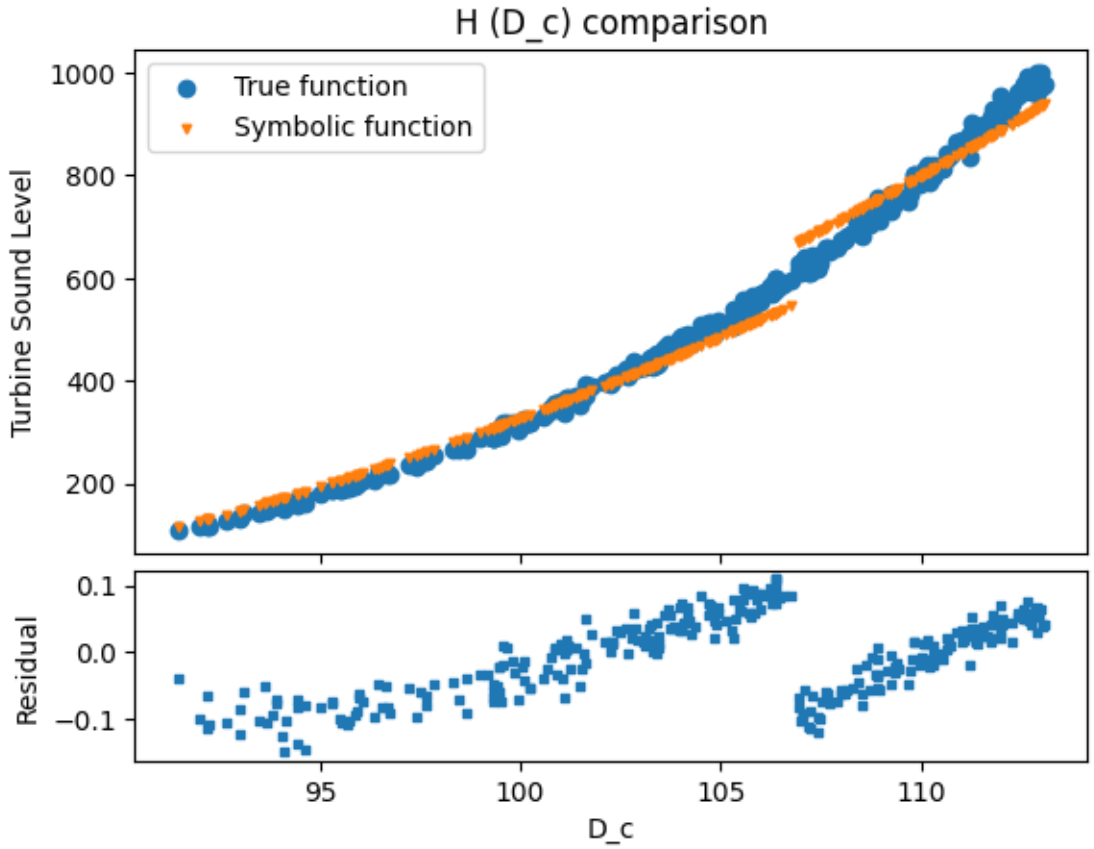

Figure 14 Second test comparison for simulated sound power.

4.1.4 Comparison with other tradition machine learning models

After the symbolic regression, a comparative analysis was conducted with two other traditional regression methods: Decision tree regressor (with max_depth of 5) and random forest regressor (n_estimators=100, max_depth=5). The max_depth value was set at 5 to avoid overfitting. These two regression methods were chosen because of their ability to operate with graphs and near selective probability style. The particulars of the regressors were tabulated below:

Table 5 Test performance for the traditional regression methods

Particulars/Regressor	Decision Tree	Random forest
Max_depth	5	5
N_estimators	N/A	100
Training time	0.004815101623535156 seconds	0.19230151176452637 seconds
Prediction time	0.005401611328125 seconds	0.011875152587890625 seconds
Performance score	0.9971088171211464	0.9978694916455902

The performance of the best trial of the symbolic regression (i.e. with specified mathematical operators), and that of the two traditional regressors were plotted against the ground truth in graphical method.

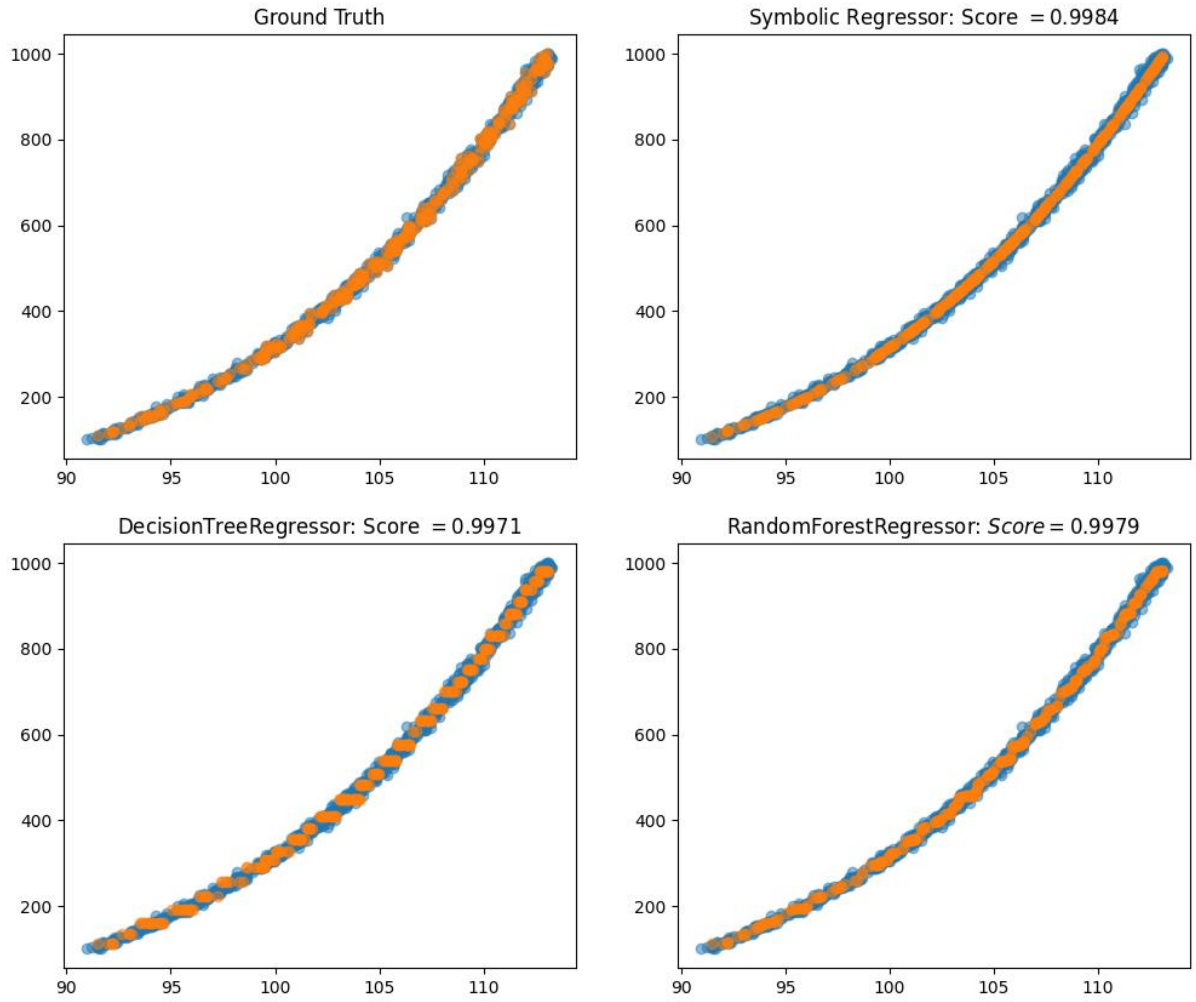

Figure 15 Comparison between symbolic regression and other regressors in case 1.

4.2 Symbolic regression on wind field data of multiple variables

In this section of the report, the research carried out using data from wind turbine at an energy farm in Texas was processed, trained, tested, visualised, and evaluated.

In this field data, investigations were made in addition to the hourly time stamp for a full year which comprises of 8760 records captured. Five scenarios were captured to have a robust data for research. These are Wind Power Output (WP) in KW, Instantaneous wind speed (WS) in m/s, Wind Direction (D), Air Pressure (P) in atm and Temperature (T) in Celsius. Like the other randomised simulated sound power data operation, 70% of the dataset were also used for the training while 30% was used for testing.

4.2.1 Data presentation

The field data was loaded into the symbolic regression program after installing all the dependent packages and it was displayed as the table below:

	WP	WS	D	P	T
0	1766.64	9.926	128	1.000480	18.263
1	1433.83	9.273	135	0.999790	18.363
2	1167.23	8.660	142	0.999592	18.663
3	1524.59	9.461	148	0.998309	18.763
4	1384.28	9.184	150	0.998507	18.963
...	...	...	...	...	...
8755	1234.70	8.848	129	0.998604	19.663
8756	1105.37	8.502	118	1.000090	19.063
8757	1405.71	9.224	117	0.998408	18.463
8758	1221.36	8.799	116	0.998013	18.063
8759	1676.77	9.748	121	1.000380	18.163

The data is therefore explored to verify relationships between the variables. First by checking the general correlations matrix of the variables as below:

Correlation matrix is :

	WP	WS	D	P	T
WP	1.000000	0.954804	-0.066276	-0.181917	0.028492
WS	0.954804	1.000000	-0.096779	-0.177656	0.048702
D	-0.066276	-0.096779	1.000000	-0.074570	-0.119265
P	-0.181917	-0.177656	-0.074570	1.000000	-0.541270
T	0.028492	0.048702	-0.119265	-0.541270	1.000000

The from the correlation matrix, the top 3 performing variables are the Wind Power (WP), Wind Speed (WS) and the atmospheric pressure (P). To verify the correlation matrix, a relationship matrix was plotted graphically to visualise the performance of each of the variable against one another as below:

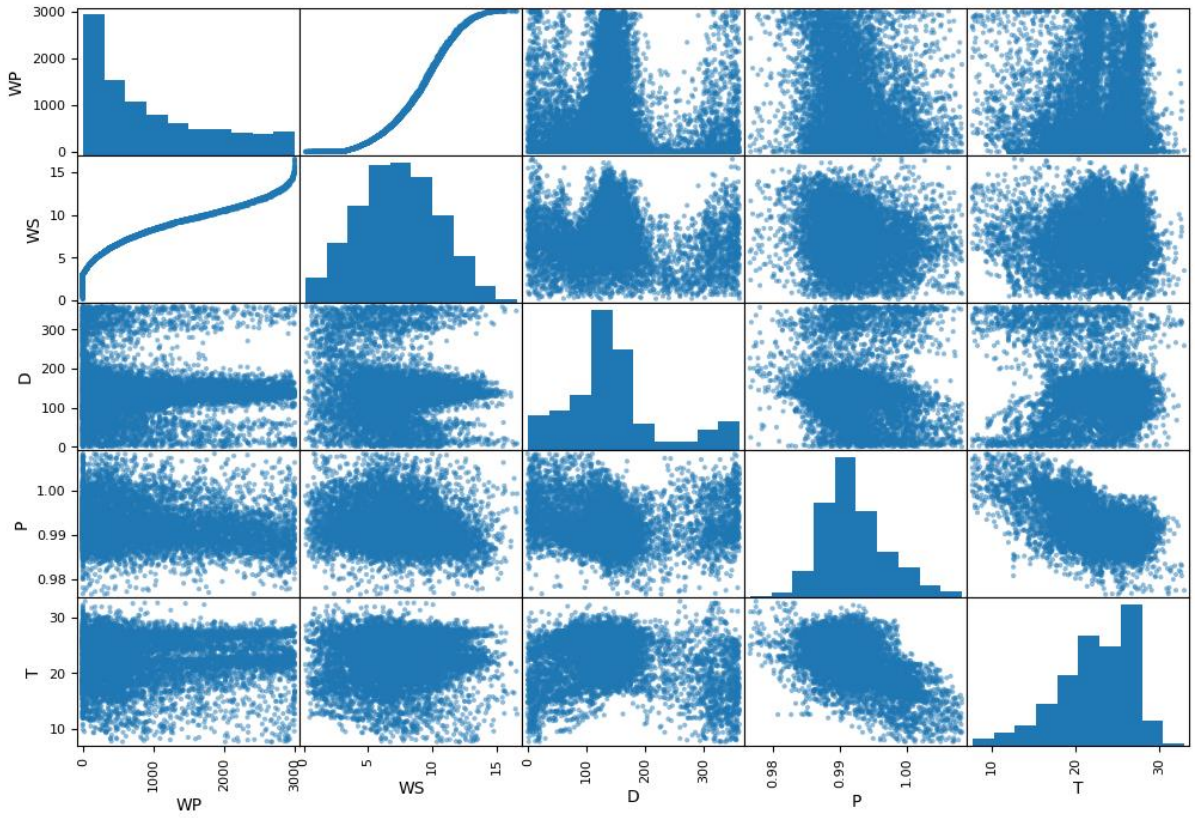

Figure 16 A graphical correlation matrix for the wind field data in case 2.

After confirming the performance of the three variables as mentioned, a 2d and 3d representation was carried out to make sure to verify the dynamics of this relationships as follows:

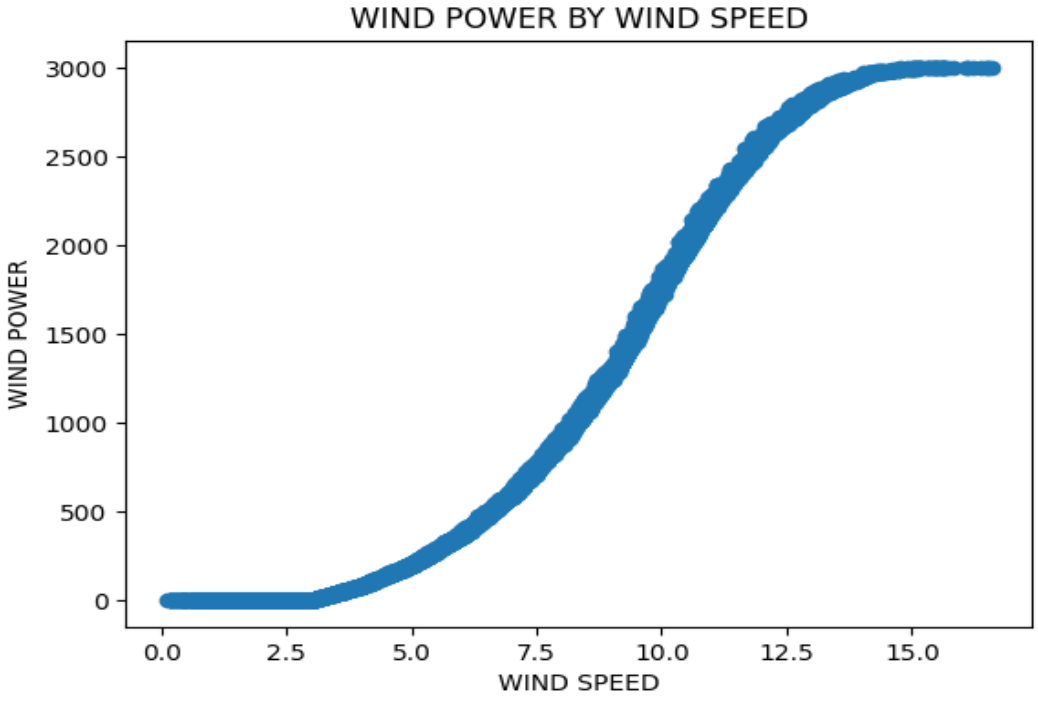

Figure 17 2-D Chart of wind speed against Wind Power output from case 2.

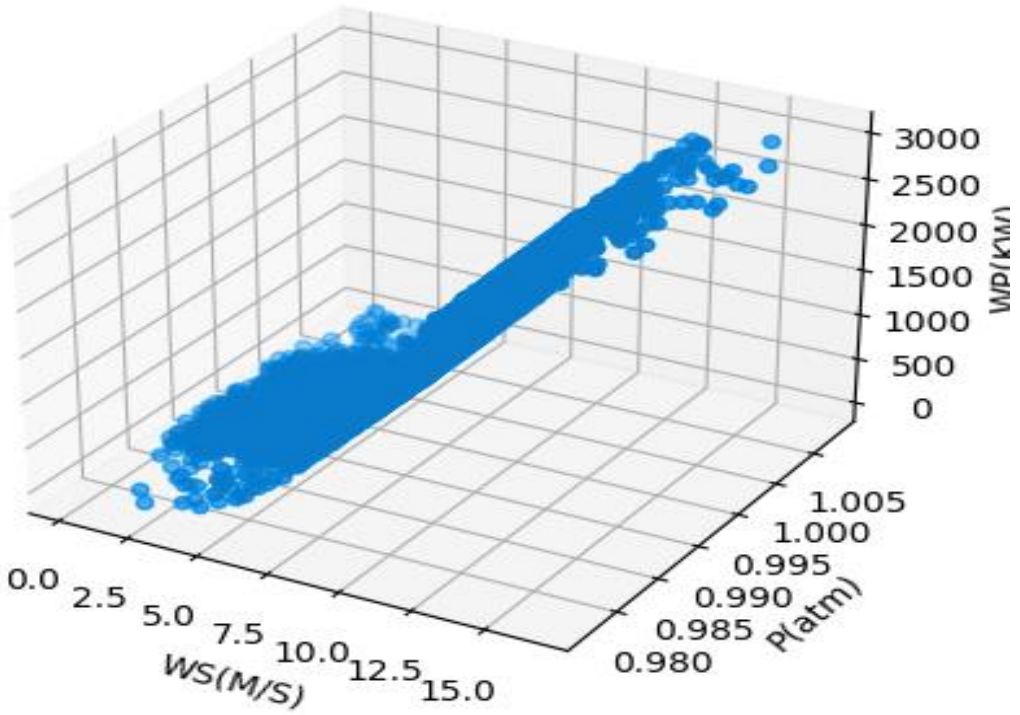

Figure 18 A 3d visualization of the highest correlated variable in case 2.

4.2.2 Symbolic regression model for field data

Having understood the implication of changing each parameter in the training model, the following was used in the case of field data.

Parameter	First Value	Second Value
population_size	10000	5000
function_set	'add','sub','mul','cos','sin','neg','log'	N/A
generations	40	20
stopping_criteria	0.01	0.01
p_crossover	0.7	0.7
p_subtree_mutation	0.1	0.1
p_hoist_mutation	0.05	0.05
p_point_mutation	0.1	0.1
max_samples	0.9	0.9
verbose	1	1
parsimony_coefficient	0.01	0.01
random_state	0	0

4.2.3. Results and Analysis

4.2.3.1 First training and testing of field data with predefined arithmetic operators.

After training the model using field data, the information below was derived.

Time to fit	2695.1628074645996 seconds
Mathematical expression	$-2X_0X_1-X_0-X_1+(1.33136862235806(X_0X_1+0.95)\log(X_{02})-\sin(X_0))(X_0+X_1+\log(\cos(\log(X_1+1.33136862235806(X_0+0.95)\log(X_0X_1\cdot(0.263-X_0))-((0.047-X_0)(\sin(\cos(X_0-0.788))-0.315104)-\sin(X_0-0.644)-\cos(2X_0)-\cos(-\log(X_0)+\sin(X_0-0.644)+\cos(2X_0))+0.105896)\sin(X_1)-\cos(X_1+\log(\cos(X_1))+\cos(-\log(X_0)+\sin(X_0-0.644)+\cos(2X_0))))))-\sin(0.479))\log(X_0)+\log(\log(X_1))+\sin(0.479X_0)-0.687$

Where X_0 is the Wind speed (WS) and X_1 is the atmospheric pressure (P).

This was also plotted as expressions below:

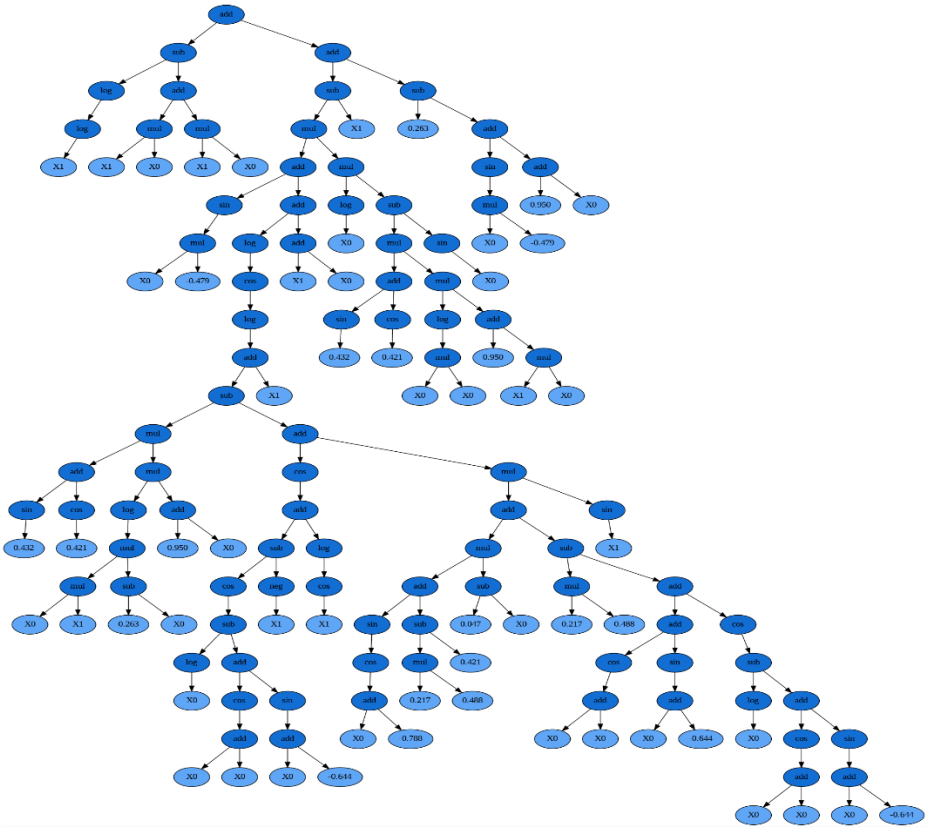

Figure 19: Case 2 first test gene expression.

To check the performance of the model, the expression was tested with the test dataset. The following information was derived:

Time to predict	0.006573915481567383 seconds
Performance score	0.9990568346624557

The output from the test above was plotted in both 2d and 3d dimensions as shown in the figure below:

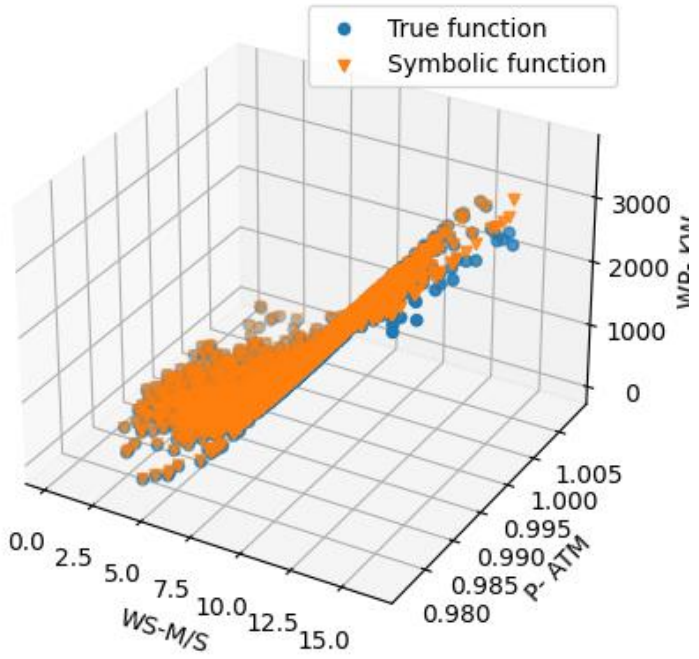

Figure 20: 3-D First test comparison for wind field data in case 2.

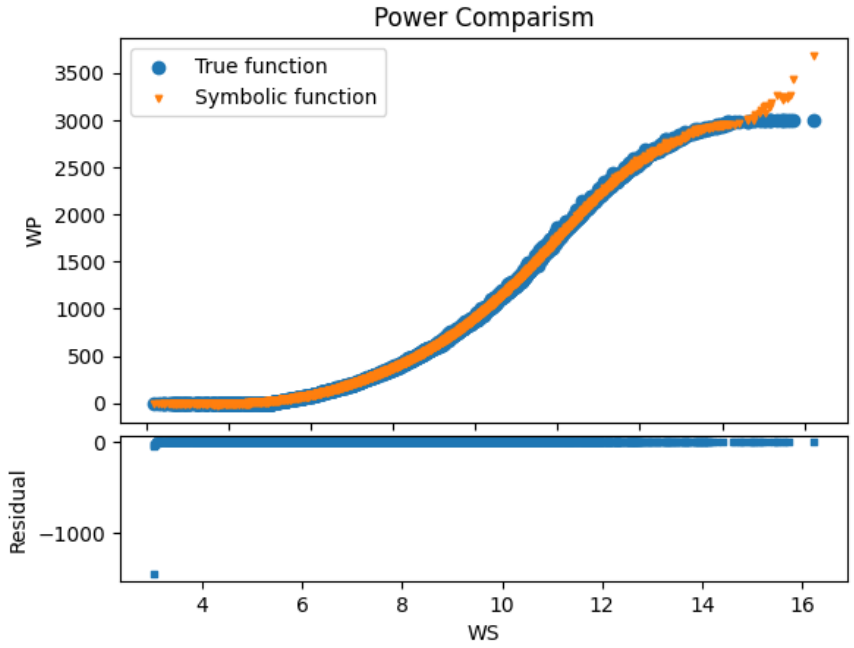

Figure 21: 2-DFirst test comparison for wind field data in case 2.

With a very high performance from the model, another parameter style was tried.

4.2.3.2 Training field data without specified function set.

Using the parameters in the second value column of the table, there, it is observed that there was no provision of predefined mathematical operators which gives the model an open ability to leverage any operator within the model for execution of training. After 268.04965806007385 seconds of training, the following mathematical expression was generated by the symbolic regression model on the training data.

$$(1.825X_0 - 2X_1 - 3.451)\,(14.4838126828742X_0 + \tilde{\infty}X_0 - 16.4838126828742X_1)$$

Just like the first training, the X_0 and X_1 variables are Wind Speed and Atmospheric pressures respectively as the variables upon which the Wind Power depends. Using gene expression, the output of the equation is as shown below:

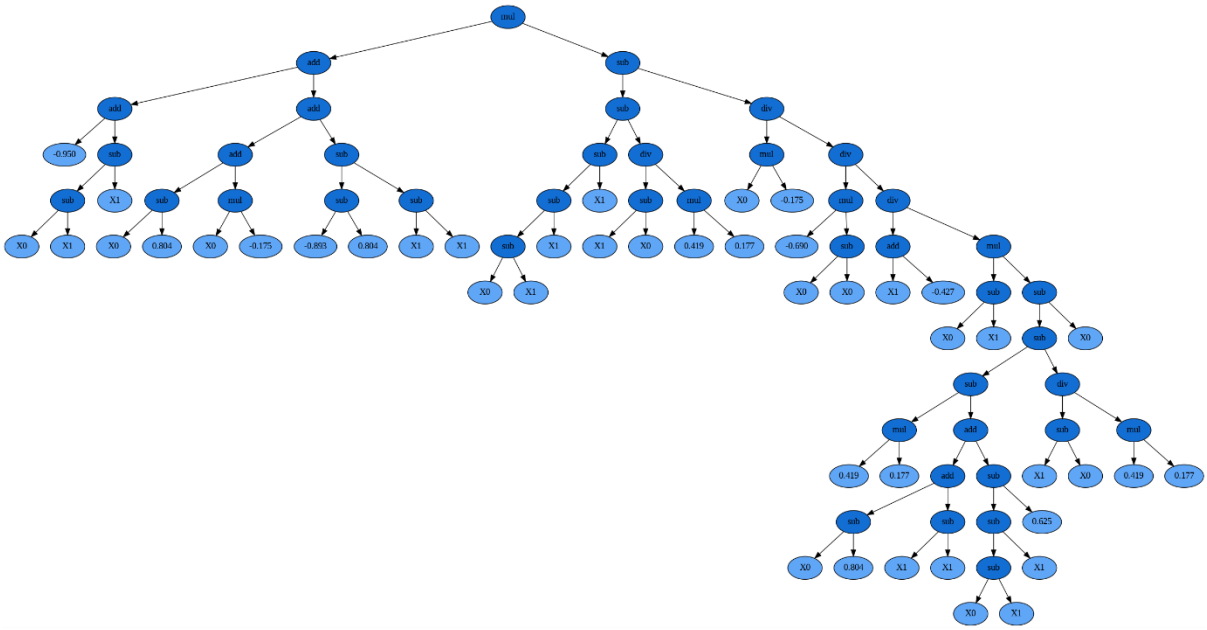

Figure 22: Case 2 second test gene expression.

After testing the function on the test data set, the prediction lasted for 0.0037853 seconds with performance of 0.9530293695138294. The visualization is as shown below:

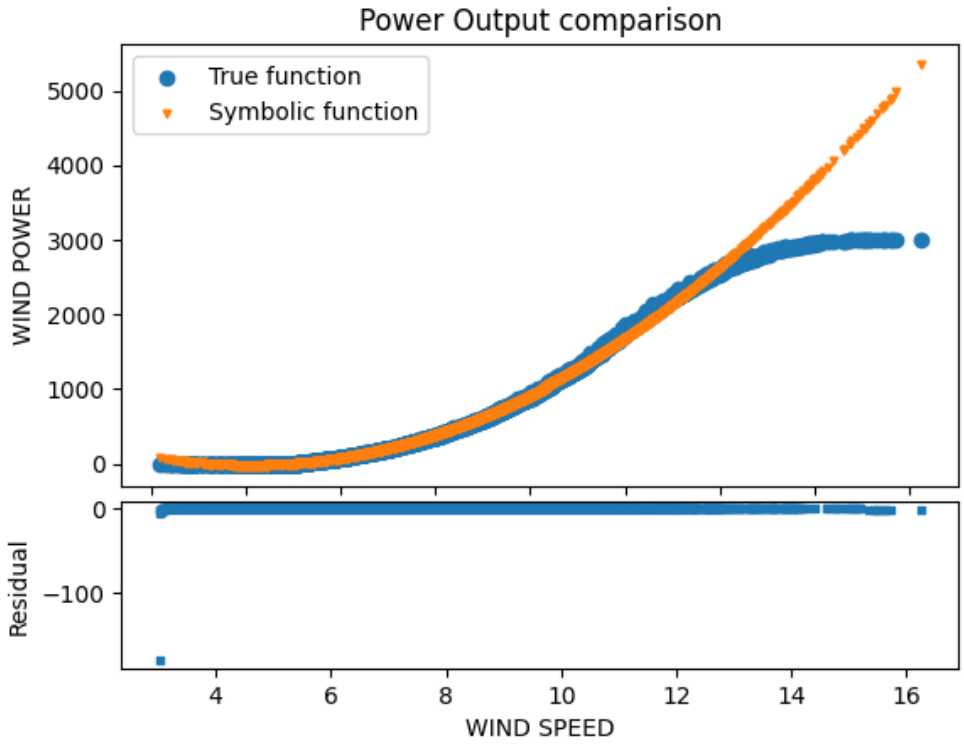

Figure 23: 2-D Second test comparison for wind field data in case 2.

4.2.3.3 Model Improvement

i. Modifying using cubic function

Having tested scenarios with both defined and undefined function set, another operator X^3 which is the 'cube operator' was introduced by means of external definition. Below is an illustration of a cubic function graphically, showing the two-placed ascent of the function.

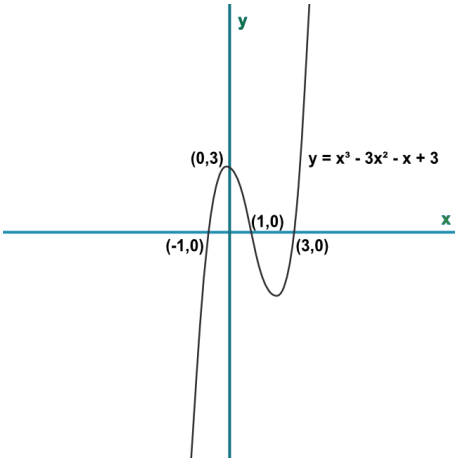

Figure 24 Cubic function illustration.

Knowing that training with defined function set performed better than undefined function set, there was a trial to improve the first by adding further operators to the function set. In the trial, division ('div'), inverse ('inv') and cube (pow_3) was added to the function set. The population size was also increased to the average of the first two trials (7500) and the

generations was placed at 40. After 746.2349247932434 seconds of training, the mathematical function below was generated:

$$(14.668646806097X_0 - 13.4838126828742X_1)\ (X_0X_1 + 0.804X_0 - 2X_1 - 3.647)$$

When tested against the ground truth of the test data, it took 0.002997875213623047 seconds to predict at performance score of 0.9698901576623179. The result is visualized below:

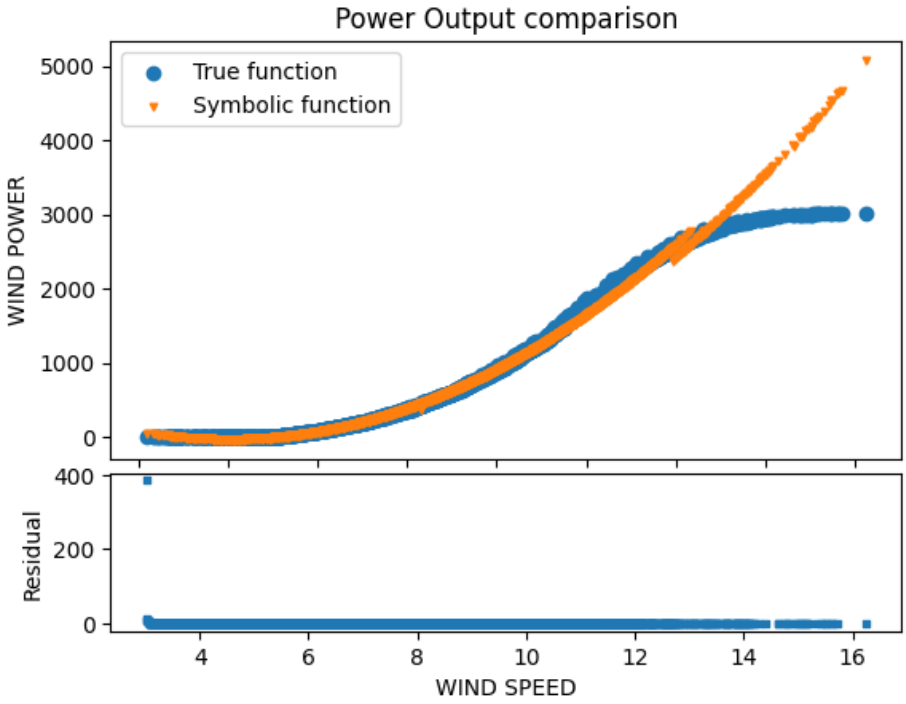

Figure 25: 2-D first improvement test comparison for wind field data in case 2.

ii. Modifying with tangent functions

In trying to further improve the performance of the model at higher wind speed, we looked at operators that has near-same pattern of plot as the ground truth. These operators are “tan ()”, tanh (), “arctan ()” and “arctanh ()”. Where, just like cubic function, **gplearn** does not have arctan, customised models were created to include these operators in the code.

First, looking at Tan () operator curves runs infinitesimally at different radian values, however, at convergence, it maintains same graphical representation as the given trend of data. Tanh () which is the hyperbolic tan () was also investigated and it has almost perfect representation of the trend lines in the graph.

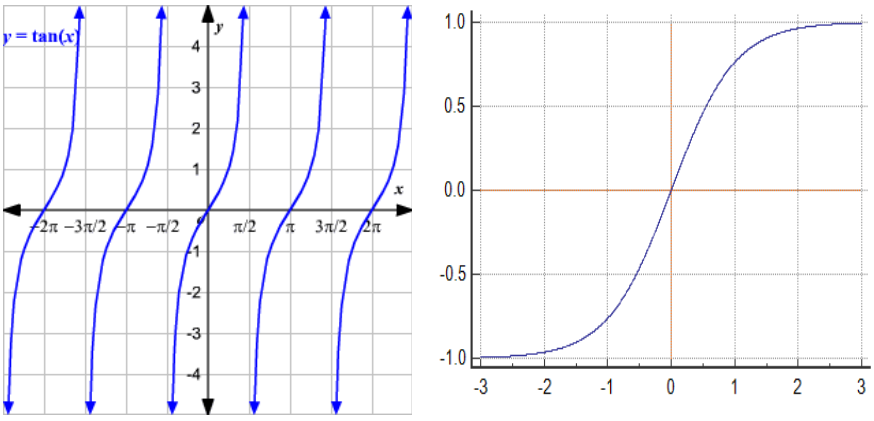

Figure 26 Plot of tan and tanh () function and operator.

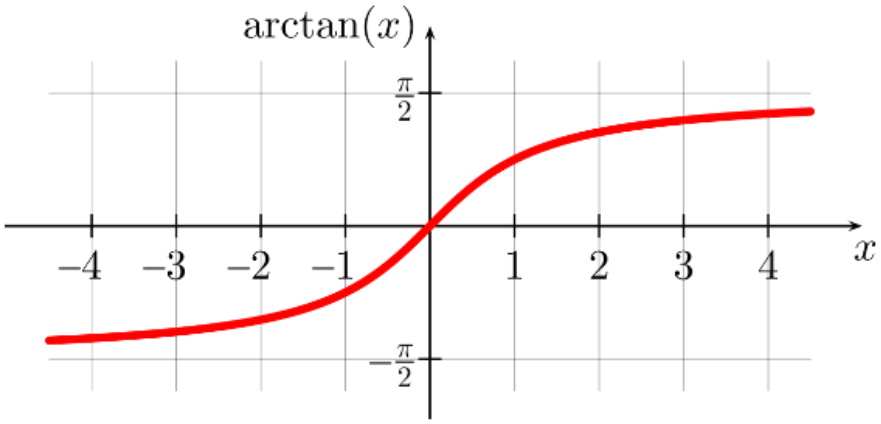

Figure 27 Plot of arctan () function.

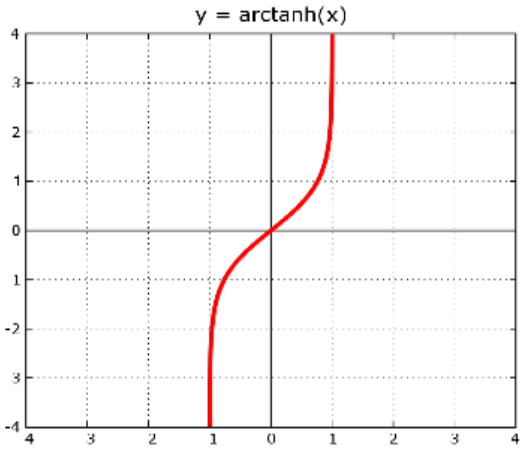

Figure 28 Plot of hyperbolic arctan-arctanh () function.

When only arctan and tan functions were used as part of the operators in the function set, maintaining the 40-generation number and 7500 population size, the particulars are as follows: Training time-1001.499017238617 seconds, Testing time-0.005352973937988281 seconds, prediction accuracy-0.994829961342487. Below is an extract of the expression delivered as the result of the analysis.

$$\left(X_0 + X_1 + \log\left(\tan\left(X_1\right)\right)^3 - \sin\left(0.494X_0 + 0.494\sin\left(\sin\left(0.494X_0 + 0.494\log\left(-\frac{-0.283062X_0 + \frac{\sin\left(0.494X_0 + 0.188726510986745\right)}{\log\left(X_0\right)}}{-0.443502X_0\cos\left(X_1\right) + X_0 + \sin\left(0.494X_0 + 0.494\sin\left(\sin\left(\left(-4.04691248960052X_1 - \frac{\arctan\left(X_0\right)\cos\left(X_1\right)}{\log\left(X_0 - 0.556\right)}\right)\cos\left(\cos\left(\arctan\left(X_1\right)\right)\right)\right)\right)\right)}\right)\right)\right)\right)\right)^3$$

The output was also plotted against the ground truth and the figure below expressed the performance.

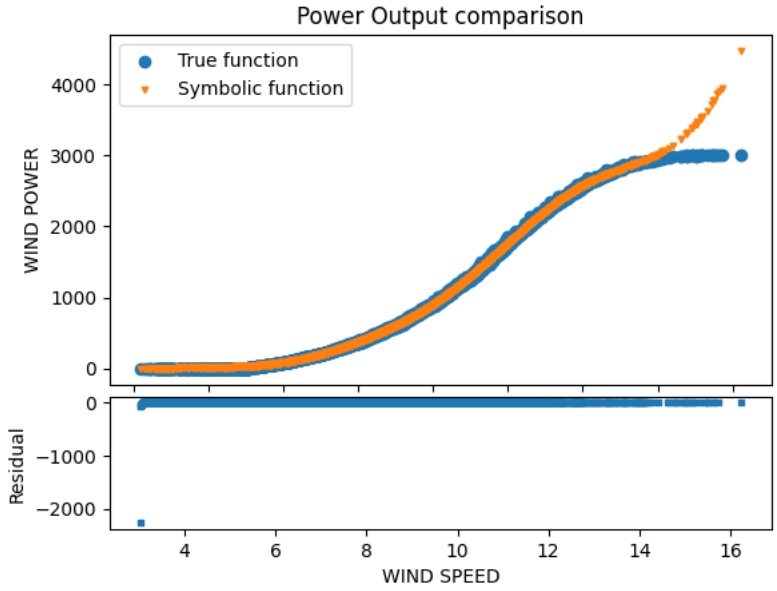

Figure 29 2-D second improvement test comparison for wind field data in case 2.

The next analysis decided to avoid hyperbolic arctan function since the function arctanh does not have closure against negatives in argument vectors, which will make it have adverse search effect. Therefore, all other tangent functions were included in the function_set for another search. The expression output are as follows:

$$X_0^3X_1^3\left(0.0979003342684786 - \sin\left(\tanh\left(\log\left(\log\left(\tanh\left(\log\left(\log\left(\log\left(X_0\right) - \sin\left(\log\left(\left(X_0 + \arctan\left(\left(\left(\sin\left(X_1\right) + \left(4.9330340480727 + i\pi\right)\cos\left(\arctan\left(X_0 + X_1 + 3\sin\left(X_0\right)\right)\right)\right)^3 - \sin\left(\tanh\left(X_0 + X_1 + \arctan\left(X_1 + 0.413\right) + \arctan\left(\arctan\left(X_0\right)\right) + 0.413\right)\right)\right)^3 + 0.4013588925852\right) + \sin\left(2X_1\right) + \sin\left(\arctan\left(X_0\right)\right)\right)^3\right)\right)\right)\right)\right)\right)\right)\right)\right)^3 + X_0^3$$

Maintaining the 40-generation number and 7500 population size, the particulars are as follows: Training time- 1722.1066410541534 seconds, Testing time- 0.005915164947509766 seconds, prediction accuracy-0. 0.9959415926641438. Below is the graphics against ground truth.

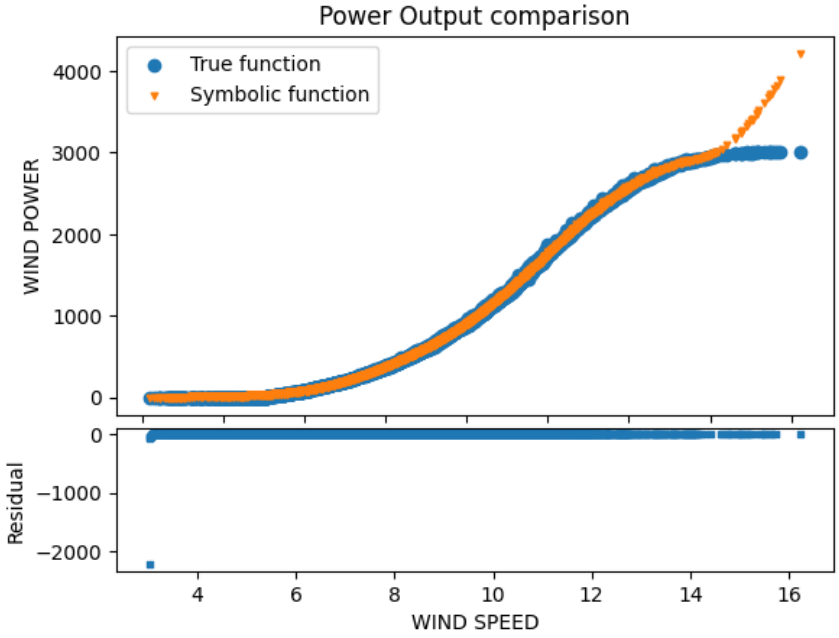

Figure 30 2-d plot of the third improvement attempt.

In addition to the above outputs, a runtime prompt was made between the 15th and 16th generation of selection as shown in the snapshot below.

```
13     54.92      4.31022e+30        30          51.193          52.592      23.95m
14     54.19      7.62567e+58        68          48.3228         52.9592     23.79m
15     56.63      4.06301e+30       101          41.0629         39.8896     24.72m
/usr/local/lib/python3.10/dist-packages/gplearn/functions.py:144: RuntimeWarning: overflow encountered in divide
  return np.where(np.abs(x1) > 0.001, 1. / x1, 0.)
<ipython-input-13-fff4ac54dd2a>:2: RuntimeWarning: overflow encountered in power
  f = x1**3
16     63.81      6.01925e+72       101          40.5637         44.3816     26.15m
17     61.92      3.18178e+34       100          40.2923         46.8231     24.22m
18     73.57     4.92553e+236        71           38.998          39.643     27.60m
19     85.75     1.38382e+194        93          38.6294         44.0711     29.06m
20     85.20     4.50438e+283       119          37.7038         51.0268     28.59m
```

Figure 31Error prompt from runtime due to large number of operators (GPlearn documentation)

Gplearn documentation mentioned that it is dues to very high length of the expressions during that selection stage within the generations. With understanding that the expression is extremely long and complicated, defeating the target of interpretability, the parsimony was increased from 0.01 to 0.05 and other all operators still intact. This resulted to improved expression output, however, the training time was still long (1710.8609819412231 seconds), and part of the performance was also sacrificed to make way for simpler and more interpretable expression as shown below (0.9932227118345662).

$$0.961X_0 \cdot \left(0.961X_0 \tan\left(\frac{0.347}{X_1}\right) + \frac{1}{\arctan\left(-\frac{1}{\sin(X_0)}\right)}\right) + \left(X_0 + X_1 + \frac{\sin\left(0.508463178X_0 + \frac{1}{\arctan^3(X_0)\arctan\left((X_0+0.928)^3\arctan(X_0)\arctan\left(\frac{X_1\left(X_0+(X_0+\sin(X_1)+0.928)^3+0.928\right)\arctan(X_0)}{X_0}\right)\right)}\right)}{\arctan(-\arctan(X_0))}\right)^3$$

The output was also plotted as below.

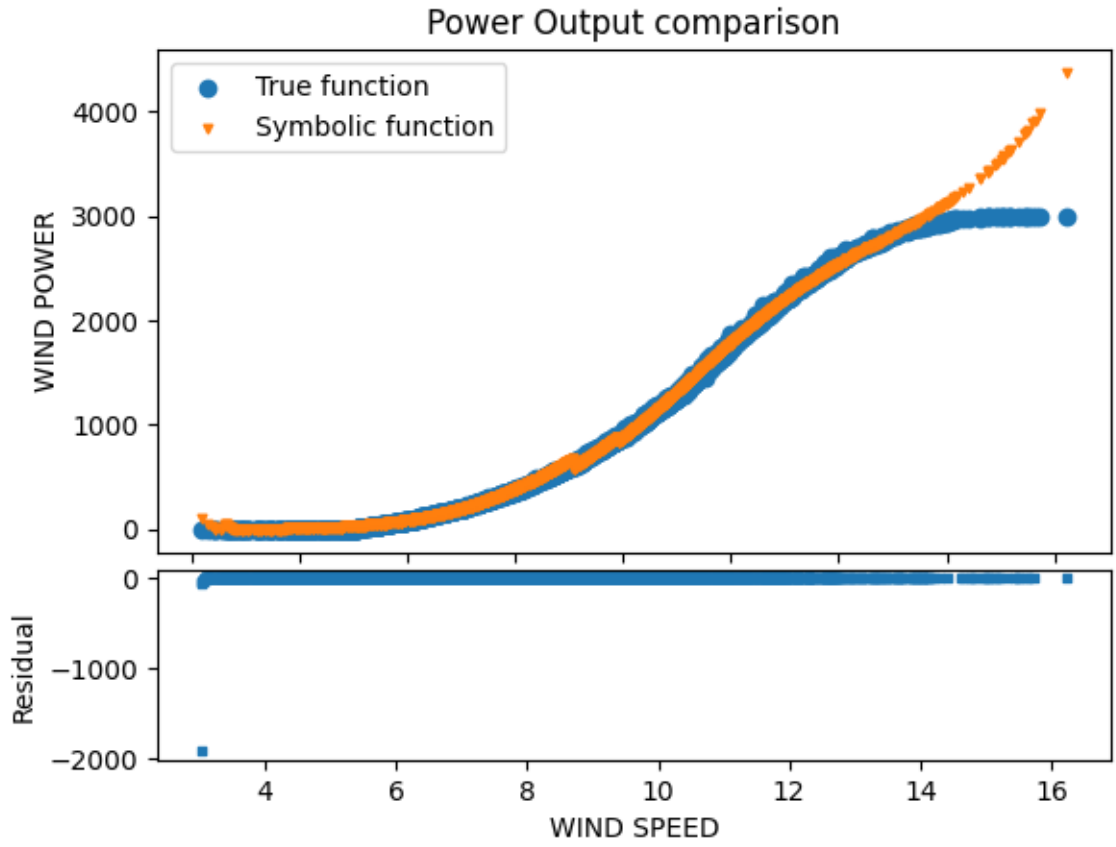

Figure 32 Fourth improvement test on case 2.

4.2.4 Comparison with other tradition machine learning models

Like the simulated data, the wind energy field data was predicted with other traditional regressors. The best performing symbolic regression model was compared to Decision Tree and Random Forest regressors and the result was plotted as follow:

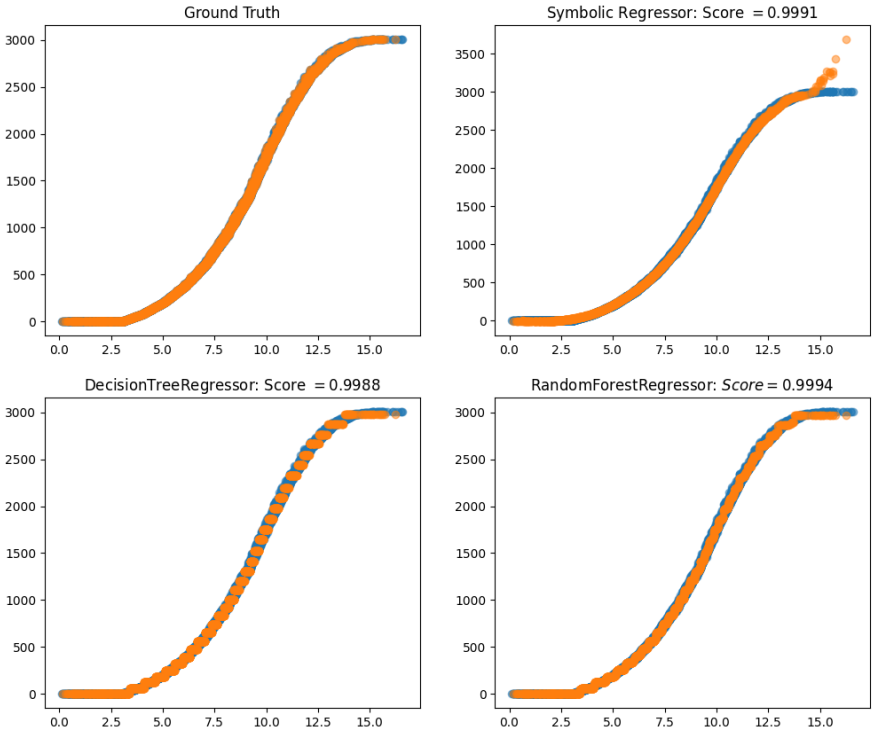

Figure 33: Comparative graphical matrix for SR with other regressors in case 2.

5.0 Discussion

One of the key benefits of leveraging symbolic regression in solving renewable energy challenges is the ability to create a relationship between the parameters that are leading to the known volatility associated with such energy sources. Wind energy is famous for being very volatile and many factors can affect the power output from a wind turbine even at other parameters staying constant. This session describes the deductions made from the results and how they helped to achieve the research objectives.

5.1 Interpretation of Symbolic Regression Models

From the experiment, symbolic regression models anchored on ***gplearn*** leverages multi-level generational selection to discover the best relationship between sets of variables in data. In describing this result, some aspects of the results would be discussed. Different approaches to the training of the model resulted in variation in outputs. In this session, we refer to the simulated turbine sound power data as **Case 1** and the wind field data for power output as **Case 2.**

5.1.1 Performance and Accuracy.

Adopting a model or its outputs is largely dependent on the performance of the model when compared to the deviation from the ground truth. From the result of difference instances of training, the following can be inferred to have significant impact on the performance of the model:

- Size of the population and number of generation parameters: From the model outputs, it will be noticed that variation in the number of population and generation in each set resulted in variation in the performance of the model against same base truth. Looking at the simulated wind sound power data, using the format of (***population_size, generation***), the training with parameters (5000,40) led to a 99.84% accuracy which is higher than the training conducted with less generation (5000, 20) yielding 98.27% accuracy. Considering the field data with more noise and data points, parameter (10000,40) yielded 99.91% accuracy, (7500,40) yielded 97% accuracy while (5000,20) yielded 95.3% accuracy. From the performance at different parameters, it can be inferred that the higher the population size and generation count, the better the performance of the model.
- Operator selections: The main driving forces behind a useful mathematical expression are the operators that make up the function. In symbolic regression driven by gplearn, function_set parameters determine the choice of operators under which the generation search can be conducted. From the Case 1, It can be noticed that the first training performed better than the second training. This is because, the former had defined function set, hence the model operated under limited operators carefully selected in line with the data pattern. On the other hand, the latter was trained without function set, hence left with option of selecting either from the basic arithmetic functions (multiplication, division, addition, and subtraction) without any guide or any mathematical operator it considers necessary, which can be too many to work with.

Same pattern can also be noticed in the three trainings in Case 2, where the 2 trainings with defined function_set parameters had better performances than the training without defined function sets.

- Data Cleanliness: While there was no metric to measure the cleanliness of the data, it can be observed that data with less noise visually performed better than that with more noise.

5.1.2 Interpretability of Mathematical Expression

The end goal of symbolic regression model is to come up with an interpretable mathematical expression representing the relationship between elements of data in a dataset. Where parsimony coefficient for all cases and trainings were kept constant (***parsimony_coefficient=0.01),*** it is clearly noticeable that the simpler the expression derived the less the accuracy of the model. Where the first training in Case 1 and Case 2 had high number of operators and complications in their output mathematical expressions, they showed to be relatively performed better their simpler counterpart. The length of the mathematical expression has also been seen to have no effect on the length of the gene expression and trees produced by a model. However, they are more related to the number of operators, number of generations and population size being involved in the model development.

5.1.3 Resource Use (Runtime Requirements)

It is a common knowledge that training a very large machine learning model require enormous amount of computing power, hence the need to make provision for that. In this research, Google Cloud's GPU was used in place of CPU which proved to be very slow and consumes a lot of time. Leveraging local computing also makes it more demanding for energy and hardware cost, therefore the need to involve cloud computing. In this particular ***gplearn*** driven symbolic regression case, the resource use was affected by the following parameters:

- Size of the data: From the results of training the simulated sound power dataset with 700 records out of 1000, the average training time with google GPU cloud computer was 472 seconds, while the wind energy field data with training dataset of 5999 records out of 8571 records took an average of 1236.5 seconds to train. The longer the training time, the more internet time and electrical energy consumed in the model training process. In addition to the number of counts, the memory size of the data involved also affect the computing time consumed in the process of developing the model.
- Shape of the data: Where the data being trained consists of only one variable being used to solve for another variable, it gives rise to a model with one independent variable as in case 1. Case 2 had 2 independent variables (Wind Speed, WS and Atmospheric Pressure, P) used to predict the Wind Power Output from the turbine. From the training time, it can be clearly seen that the computing time required for case 2 is high compared to that of case 1.

- Data Noise: The runtime required to train a dataset with more noise as in case2 is less than the runtime required to train the data with less noise as case 1.

5.2 Comparison with Traditional Regression Approaches

One important aspect of this research is the comparison between symbolic regression and other regressors in machine learning ecosystem. Diagrammatically, these comparisons are shown in figures 14 and 23 for case 1 and case 2 respectively. Symbolic regression anchored on Genetic Programming boasts of some benefits as seen in this research over some of the other traditional regression models like the Decision Tree and Random Forest Regressors.

Performance and Accuracy: From the results of the comparative analysis gotten from case 1, it can be noticed that while all the regressors used performed beyond the statistical confidence level of certainty (95%), the performance of the symbolic regression model exceeded that of the other models significantly. Considering case 2, where the best performing symbolic regression model was compared with other regressors, it performed better than decision tree and while random forest seemed to perform better, the difference was not significant.

Automatic Feature Engineering: Unlike traditional regressors that require manual feature selection and engineering, Symbolic Regression can automatically discover complex relationships and create new features from existing ones through its symbolic representation. This can be particularly beneficial when the underlying relationships between features and the target variable are unknown or not easily captured by traditional feature engineering techniques. In this case of field wind power output, there are unlimited conditions that were not measured or accounted for, but they have significant impact on the power output of the turbine. Symbolic regression made it possible to use constants and coefficients in leveraging the available data to connect the represent the relationship with target variable.

Interpretability: The resulting model from Symbolic Regression is an equation that explicitly shows the relationship between the features and the target variable. This enables researchers and users to understand the logic behind the model's predictions and gain insights into the underlying process it represents. In contrast, traditional regressors like black-box models can be difficult to interpret, making it challenging to understand how they arrive at their predictions.

Flexibility: Symbolic Regression can handle various data types, including symbolic data and continuous data. This flexibility allows it to model a wider range of problems compared to traditional regressors that might be limited to specific data types. While the research was limited to measured numerical data from turbines and simulations, symbolic regression has proven to perform exceptionally with the data type presented in this research.

Discovery of Non-Linear Relationships: Symbolic Regression can effectively capture non-linear relationships between features and the target variable. This is advantageous in scenarios where the data exhibits complex patterns that traditional linear regression models might not be able to capture accurately. As can be seen in the results from both cases in this

research, neither of the equations formed were linear, hence, it eliminated the limitation of finding the best average fit as is common among other regressors.

Potential for Evolving Existing Models: The symbolic representation, gene expression and mathematical expressions of SR models allows them to be further evolved or improved by incorporating domain knowledge or additional data. This can be useful for refining the model over time or adapting it to changing conditions. These outputs can be further modified to predict relationships from future data knowing, having known the basis for the former.

5.3. Challenges and Future Directions

5.3.1 Current Limitations and Issues

In the previous section, Symbolic regression, through the results in the previous chapter showed tremendous benefits over other regressors and some other machine learning models. However, it's important to consider some limitations of Symbolic Regression as well:

Initial Computational Cost: The process of evolving trees can be computationally expensive, especially for large datasets. From the results we got in case 1, while symbolic regression being compared used 607.3658051490784 seconds of run time to train a model, Decision Tree and Random Forest used 0.004815101623535156 seconds and 0.19230151176452637 seconds of run time respectively. In the second case (case 2) with even larger dataset, the best competent symbolic regression model took 2695.1628074645996 seconds to train, while Decision tree and even better performed random forest took a fraction of seconds. While the cost of predicting future data might be lower for symbolic regression since there is an actively verified mathematical expression produced as an outcome, the cost of training the model is enormously greater than other regressor counterparts.

Potential for Overfitting: Due to the flexibility of Symbolic Regression models, there's a risk of overfitting the training data if not properly regularized. In the bid to make sure that a model performs very well for a dataset, the user tends to over-tune the parameters to create room for more intricate searches and to eliminate specific errors observed. This therefore streamlines the model from performing at equal or better standing when exposed to a different dataset. Other regressors leaves an open position for general search and very little option of overfitting.

Readability of Complex Models: While generally interpretable, very complex models generated by symbolic regression through genetic programming can become difficult for humans to understand. As can be noticed in the mathematics expression from the second training of both case 1 and case 2 without defined operators, there was an infinity sign (∞) as part of the interpretable models. This element bound by an operator makes it more complex to explain.

Parsimony Problems: In symbolic regression, there could be sometimes a trade-off between simplicity of the model and accuracy of the model. When it could be easier to look at the second mathematical expressions from case 1 and 2 and conclude that they are simple and easier to explain that the first results from both experiments, selecting them would imply

ignoring the continuity guarantee of the high performing models over future data points. As can be seen in figure 13, where there was a snap in case 1 data continuity, and from case 2's figures 21 and 22 where the model moved away from the data trend midway, it can be inferred that simplest of the models does not always represent the best performing model, even when it can be better explainable.

5.3.2 Potential Improvements in Symbolic Regression Techniques

Having explained different challenges and limitations encountered in symbolic regression, some potential improvements can be done on the conventional symbolic regression techniques.

Firstly, and most importantly, preprocessing and data cleaning ability should be incorporated into the algorithm before feeding in data for training and prediction. Improving the ability of symbolic regression to deal with noisy data or outliers can make the technique more robust and accurate, particularly in industrial and real-world scenarios where clean data is not always available. As was noticed in the wind power data with more noise, it is important to adjust the model to be able to handle noisy data.

Furthermore, efforts should be made towards reducing the run time as much as possible. This can be achieved by increasing the operating power of the computer used in training the model in line with the veracity, variety, and volume of the data being trained. One can choose Graphic Processing Unit (GPU), where Central Processing Unit (CPU) proves sluggish on personal computers. One can also choose cloud computing like amazon web services, google cloud computing (as used in this research) or Azure Cloud Platform to enhance the parallel computing capacity of the processors. Leveraging parallel computing resources to run symbolic regression algorithms can significantly speed up the process by evaluating multiple models simultaneously.

Another approach that would improve the performance of symbolic regression is to employ hybrid methods in feature selection and engineering. Combining symbolic regression with other machine learning techniques, such as neural networks or ensemble methods, might improve model performance, especially on complex datasets. Incorporating automatic feature selection or dimensionality reduction techniques can help in removing irrelevant or redundant variables, making the model simpler and potentially more interpretable. Refining the fitness functions used to evaluate the goodness of fit for generated models can lead to more robust models. This includes the use of multiple criteria, such as simplicity and predictive power, instead of relying solely on error minimization.

Additionally, introducing user-defined constraints can also improve the symbolic regression techniques. This will prevent the user from over-fitting. Allowing users to incorporate domain-specific knowledge through constraints or hints about the expected model form can guide the search process towards more applicable and interpretable models. Improving the efficiency of the underlying algorithms, which can involve advances in tree-based data structure manipulations or the use of more efficient programming languages and libraries. Developing and using metrics to specifically quantify and optimize the interpretability of the

resulting symbolic expressions may help in better aligning with human understanding and facilitate model adoption.

In addition to the above approaches, multi-objective optimization can also improve the efficiency and performance of symbolic regression techniques. Matteo et al considered this approach very useful in developing an effective symbolic regression mode. Incorporating multi-objective optimization techniques that can optimize for conflicting objectives, such as model complexity and accuracy, simultaneously (Matteo et al).

5.3.3 Improvement Issues

In most cases, the need to add more operators arises and sometimes the need to use heavier infrastructure to address the optimal selection problem. However, care should be taken in doing so since in most cases, as we could see, simply results to worse cases. Some cases are digitally inaccessible and any deviation from the expected trend of the operators will mislead the model into selecting its supposed best expression.

As can be seen in section 4.2.3.3 where several improvement measures were employed, which included selecting operators with similar looking trends like tanh (). Apart from putting a very strong strain on resource use, the training took very much longer time and with less accurate result than the first trial of case 1.

5.4 Symbolic Regression as Emerging Trends in Energy Engineering

Symbolic regression is emerging as a powerful technique for data-driven modelling and discovery in the renewable energy sector. The key advantages of symbolic regression are:

Symbolic regression can uncover hidden relationships and elucidating ambiguous connections in energy data without relying on predefined model structures or assumptions (Gajendran,2023). This allows it to capture complex nonlinear phenomena that may not be well-described by traditional engineering models.

By generating interpretable mathematical expressions, symbolic regression provides insights into the underlying physical principles governing energy systems. This can lead to the discovery of new scientific laws and the verification of existing ones.

Symbolic regression has been successfully applied to a variety of energy engineering problems, including wind turbine wake prediction, combined cycle power plant performance estimation (Andelic et al, 2023), and materials discovery for energy applications Angelis, et al., 2023). These studies demonstrate the versatility of the technique across different energy domains.

The combination of symbolic regression with other machine learning methods, such as neural networks, can further enhance the modelling capabilities and interpretability (Nanna, 2021). This hybrid approach leverages the strengths of both techniques.

5.5. Conclusion and Future Study.

In conclusion, the results from this research have shown the wide range of application of symbolic regression in addressing renewable energy challenges of volatility, especially

regarding wind energy power output as studied. Over other regression techniques, it has proven to have tremendous advantages.

Future study would explore several improvement strategies as suggested in section 5.3.2 to come up with more robust and more effective symbolic regression models. It is recommended that more field data be trained to come up with more widely applicable mathematical expression to address key renewable energy challenges.

References

A. Nanna Grytzell (2021) Symbolic Regression Using Genetic Programming Leveraging Neural Information Processing, Lund University. Available at: https://lup.lub.lu.se/luur/download?fileOId=9046102&func=downloadFile&recordOId=9046100

Anđelić, N., Lorencin, I., Mrzljak, V., and Car, Z., (2024). On the application of symbolic regression in the energy sector: Estimation of combined cycle power plant electrical power output using genetic programming algorithm. Engineering Applications of Artificial Intelligence. 133. 1-32. 10.1016/j.engappai.2024.108213.

Angelis, D., Sofos, F. and Karakasidis, T.E. (2023) *Artificial Intelligence in physical sciences: Symbolic regression trends and perspectives - archives of Computational methods in engineering*, *SpringerLink*. Available at: https://link.springer.com/article/10.1007/s11831-023-09922-z (Accessed: 02 May 2024).

Anushruthika (2023) Understanding classification loss functions, Medium. Available at: https://medium.com/@anushruthikae/understanding-classification-loss-functions-7cc13fd6ac97 (Accessed: 29 March 2024).

Bilgili, Mehmet & Alphan, Hakan & Ilhan, Akin. (2022). Potential visibility, growth, and technological innovation in offshore wind turbines installed in Europe. Environmental Science and Pollution Research. 30. 1-19. 10.1007/s11356-022-24142-x.

Chiu, CH., Lung, SC.C., Chen, N. (2021), Effects of low-frequency noise from wind turbines on heart rate variability in healthy individuals. Sci Rep 11, 17817. https://doi.org/10.1038/s41598-021-97107-8

Chua, W. et al. Data-driven discovery and validation of circulating blood-based biomarkers associated with prevalent atrial fibrillation. Eur. Heart J. 40, 1268–1276 (2019)

Echeverri-Londoño, C.A. and González-Fernández, A.E. (2018) Model for the prediction of noise from wind turbines, Revista Facultad de Ingeniería Universidad de Antioquia. Available at: https://www.redalyc.org/journal/430/43057833006/html/ (Accessed: 13 April 2024).

Fleck, P., Werth, B. and Affenzeller, M. (2024) Population dynamics in genetic programming for dynamic symbolic regression, MDPI. Available at: https://www.mdpi.com/2076-3417/14/2/596 (Accessed: 29 March 2024).

Gajendran, M.K., (2023) *Machine learning-based approach to wind turbine wake prediction under yawed conditions*, *MDPI*. Available at: https://www.mdpi.com/2077-1312/11/11/2111 (Accessed: 02 May 2024).

Hoen, B.D., Diffendorfer, J.E., Rand, J.T., Kramer, L.A., Garrity, C.P., and Hunt, H.E., (2018), United States Wind Turbine Database v6.1 (November 28, 2023): U.S. Geological Survey, American Clean Power Association, and Lawrence Berkeley National Laboratory data release, https://doi.org/10.5066/F7TX3DN0.

Keren, L.S., Liberzon, A. and Lazebnik, T. (2023) A computational framework for physics-informed symbolic regression with straightforward integration of domain knowledge, Nature News. Available at: https://www.nature.com/articles/s41598-023-28328-2 (Accessed: 19 March 2024).

Lei Gan a et al. (2022) Integration of symbolic regression and domain knowledge for interpretable modelling of remaining fatigue life under multistep loading, International Journal of Fatigue. Available at: *https://www.sciencedirect.com/science/article/abs/pii/S014211232200161X* (Accessed: 19 March 2024).

M, Padhma. (2023) A comprehensive introduction to evaluating Regression Models, Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2021/10/evaluation-metric-for-regression-models/ (Accessed: 29 March 2024).

Manne Friman (2011) *Directivity of sound from wind turbines*. Available at: http://kth.diva-portal.org/smash/get/diva2:458870/FULLTEXT01.pdf (Accessed: 20 April 2024).

Matteo M., Nicola D., Katsiaryna H., (2024) In-context symbolic regression: Leveraging language models for function discovery. Available at: https://arxiv.org/html/2404.19094v1 (Accessed: 02 May 2024).

Miller, D. C. & Salkind, N. J. Handbook of Research Design and Social Measurement (Sage Publishing, 2002)

P. Valsaraj a b et al. (2019a) Symbolic regression-based improved method for wind speed extrapolation from lower to higher altitudes for wind energy applications, Applied Energy. Available at: https://www.sciencedirect.com/science/article/pii/S0306261919319579 (Accessed: 29 March 2024).

Sobh, R. & Perry, C. (2006), Research design and data analysis in realism research. Eur. J. Mark. 40, 1194–1209.

Wass, R. (2018). Design of Wind Turbine Tower Height and Blade Length: an Optimization Approach. Mechanical Engineering Undergraduate Honors Theses Retrieved from https://scholarworks.uark.edu/meeguht/70

Weng, B. et al. (2020) Simple descriptor derived from symbolic regression accelerating the discovery of new perovskite catalysts, Nature News. Available at: https://www.nature.com/articles/s41467-020-17263-9 (Accessed: 19 March 2024).