

**T.C.**  
**ANTALYA BİLİM UNIVERSITY**  
**INSTITUTE OF POSTGRADUATE EDUCATION**

**ELECTRICAL AND COMPUTER ENGINEERING**  
**MASTER'S THESIS**

**ADVANCED GENETIC PROGRAMMING MODELS FOR PRECIPITATION  
FORECASTING**

**Nooria SULTANI**

**JUNE 2025**

**ANTALYA**



**T.C.**  
**ANTALYA BILIM UNIVERSITY**  
**INSTITUTE OF POSTGRADUATE EDUCATION**

**ELECTRICAL AND COMPUTER ENGINEERING**  
**MASTER'S THESIS**

**ADVANCED GENETIC PROGRAMMING MODELS FOR PRECIPITATION  
FORECASTING**

**Nooria SULTANI**

**JUNE 2025**

**ANTALYA**

**T.C.  
ANTALYA BILIM UNIVERSITY  
INSTITUTE OF POSTGRADUATE EDUCATION**

**ADVANCED GENETIC PROGRAMMING MODELS FOR PRECIPITATION  
FORECASTING**

**Nooria SULTANI**

This thesis was accepted by the jury (with unanimous vote / majority vote) on the date [30/06/2025](#) in ELECTRICAL AND COMPUTER ENGINEERING PROGRAM of ELECTRICAL AND COMPUTER ENGINEERING DEPARTMENT.

Prof. Dr. Ali DANANDEH MEHR (Supervisor)

Asst. Prof. Dr. Shahram TAHERI

Assoc. Prof. Dr. Rifat TÜR

Director of The Institute

Asst. Prof. Dr. Oğuzhan HERDİ

Thesis Submission Date:

## DECLARATION

Master Thesis of this study named “Advanced Genetic Programming Models for Precipitation Forecasting”, which I presented, I declare that scientific moral principles were followed in the preparation of this study, in case of benefiting from the works of others, reference is made in accordance with scientific norms, no falsification has been made in the data used, and that any part of this study is not presented as another academic study.

... / ... / 20..

Nooria SULTANI



## CONTENTS

ABSTRACT.....	i
ÖZET .....	ii
SYMBOLS AND ABBREVIATIONS .....	iii
LIST OF TABLES .....	v
LIST OF FIGURES .....	vi
PREFACE .....	viii
1. INTRODUCTION .....	1
1.1. Challenges in Traditional Rainfall Prediction Models.....	1
1.2. Motivation to Use Genetic Programming .....	1
1.3. Research Objectives .....	2
1.4. Research Questions .....	2
1.5. Structure of the Thesis .....	2
2. LITERATURE REVIEW .....	3
2.1. Hydrological Forecasting and Rainfall Prediction Models.....	3
2.2. Traditional Methods for Rainfall Forecasting.....	4
2.3. Genetic Programming in Hydrology.....	5
2.4. Signal Decomposition Techniques in Time Series Analysis .....	6
2.5. Hybrid Models in Rainfall Prediction .....	7
2.6. Evaluation Metrics for Forecasting Models.....	8
3. METHOD.....	11
3.1. Data Collection and Preprocessing .....	11
3.1.1. The Area of Study .....	11
3.1.2 Data Description.....	11
3.1.3. Data Cleaning.....	12
3.2. Data Transformation .....	13
3.3. Signal Decomposition with Wavelet Transform.....	14
3.3.1. Rationale for the Use of Wavelet Decomposition.....	15
3.3.2. Selection of Mother Wavelet .....	16
3.3.3. Decomposition Process .....	16
3.4. Genetic Programming For Symbolic Regression.....	16
3.4.1. GPdotNET Tool .....	18
4. FINDINGS .....	20
4.1. GPdotnet Setup.....	20
4.3. Model 2 Hybrid WT-GP .....	26
Figure 4.3.10.Tree Structure of Model-2 .....	34

4.4. Model 3 GP with Lagged Data Features .....	34
4.5. Model 4 Correlation Analysis of Lagged Data not Decomposed Data.....	38
4.6. GP Model 5 Performance with Lagged Features (No Wavelet) .....	42
4.7. Model 6 GP with Wavelet-Based Lag Features .....	46
4.8. Comparative Analysis of All GP Models.....	50
4.9. Key Observations .....	52
5. CONCLUSIONS AND SUGGESTIONS.....	54
5.1. The Summary Points From This Research.....	54
5.2. Scientific Implications .....	54
5.3. Comparison with Previous Studies .....	55
5.4. Limitations of the Study.....	55
5.5. Recommendations for Future Research .....	55
REFERENCES.....	57

## **ABSTRACT**

### **ADVANCED GENETIC PROGRAMMING MODELS FOR PRECIPITATION FORECASTING**

**Nooria SULTANI**

**MSc Thesis in Electrical and Computer Engineering**

**Supervisor: Prof. Dr. Ali DANANDEH MEHR**

**June 2025: 74 pages**

Rainfall forecasting is an intricate task as hydrological processes tend to be nonlinear and dynamic. This thesis This thesis conducts a comparative analysis between a Genetic Programming (GP)-based model and a hybrid model that combines GP with wavelet-based signal decomposition. The meteorological parameters of Antalya, Turkey, were employed here and analyzed between 2016 and 2023, covering variables such as temperature, humidity, sea-level pressure, wind speed, solar radiation, and precipitation. The raw data were initially preprocessed by performing normalization, treating missing values, and eliminating outliers. Next, meteorological variables were used for wavelet decomposition to form the multi-scale components of the raw and lagged data. The description of principles and the development of several GP models were completed. These GP models were tested with different combinations of inputs, lagged data, and wavelet-decomposed data. The model assessment measures were calculated for each model, including the RMSE and NSE scores, to decide which model outperforms the others.

The outcome clearly shows that the GP model integrating temporal features but excluding the wavelet transformation provided the most precise prediction, while the models with wavelet components exhibited lower performance due to the domination of the complexity of the inputs. The model's efficiency was significantly increased via the feature selection based on the correlation analysis approach without any loss of model accuracy. In general, the study underlines the possibility of GP-based models to be used in the case of rainfall forecasting tasks, and the importance of temporal information and feature selection is discussed. Further studies can be conducted to test different hybrid strategies and feature engineering methods that might improve forecasting accuracy.

**KEYWORDS:** Genetic programming, wavelet decomposition, rainfall forecasting, symbolic regression, feature selection, Antalya.

**COMMITTEE:** Prof. Dr. Ali DANANDEH MEHR (Supervisor)

Asst. Prof. Dr. Shahram TAHERI

Assoc. Prof. Dr. Rifat TÜR

## ÖZET

### YAĞIŞ TAHMINİ İÇİN GELİŞMİŞ GENETİK PROGRAMLAMA MODELLERİ

Nooria SULTANI

Yüksek Lisans Tezi, Elektrik ve Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Ali DANANDEH MEHR

Haziran 2025: 74 sayfa

Yağış tahmini, hidrolojik süreçlerin doğrusal olmayan ve dinamik doğası nedeniyle karmaşık bir zorluktur. Bu tez, kısa vadeli yağış tahmini için genetik programlama (GP) ile dalgacık (wavelet) tabanlı sinyal ayrıştırma ve birleştirme geliştirilmiş bir hibrit modelleme yaklaşımı önermektedir. Çalışmada, 2016–2023 yılları arasında Antalya, Türkiye’den alınan sıcaklık, nem, deniz seviyesi basıncı, rüzgar hızı, güneş radyasyonu ve yağış gibi meteorolojik değişkenleri içeren veri seti kullanılmıştır.

Ham veriler, normalizasyon, eksik değerlerin giderilmesi ve aykırı değerlerin düzeltilmesi yoluyla ön işlemden geçirilmiştir. Ardından, çok ölçekli bileşenleri çıkarmak için hem ham hem de gecikmeli meteorolojik değişkenlere dalgacık ayrıştırması uygulanmıştır. Çalışmada, ham girdiler, gecikmeli özellikler ve dalgacıkla dönüştürülmüş verilerin çeşitli kombinasyonlarını kullanan GP modelleri geliştirilmiş ve değerlendirilmiştir. Model performansı, Ortalama Kök Kare Hata (RMSE) ve Nash-Sutcliffe Verimlilik Katsayısı (NSE) ile ölçülmüştür.

Bulgular, dalgacık ayrıştırması kullanılmadan gecikmeli meteorolojik değişkenlerle eğitilen GP modelinin en iyi tahmin performansını gösterdiğini ortaya koymaktadır. Buna karşılık, dalgacık bileşenlerini entegre eden modeller, artan girdi karmaşıklığı nedeniyle daha düşük doğruluk sergilemiştir. Korelasyon analizine dayalı değişken seçimi ise model verimliliğini, doğruluktan ödün vermeden artırmıştır. Genel olarak, çalışma GP tabanlı modellerin yağış tahmininde güçlü bir potansiyele sahip olduğunu ve zamansal bilgilerin ve değişken seçiminin önemini vurgulamaktadır. Gelecek araştırmalarda, alternatif hibrit stratejiler ve gelişmiş değişken mühendisliği yöntemlerinin incelenmesi önerilmektedir.

**ANAHTAR KELİMELER:** Genetik Programlama, Dalgacık Ayrıştırması, Yağış Tahmini, Sembolik Regresyon, Değişken Seçimi, Antalya.

**JÜRİ:** Prof. Dr. Ali DANANDEH MEHR (Danışman)

Dr. Öğr. Üyesi. Shahram TAHERI

Doç. Dr. Rıfat TÜR

## SYMBOLS AND ABBREVIATIONS

RMSE	: Root Mean Squared Error – a measure of the average magnitude of the errors
NSE	: Nash–Sutcliffe Efficiency – a normalized statistic indicating model predictive power
GP	: Genetic Programming
GPM	: Global Precipitation Measurement (satellite mission)
GPdotNET	: Genetic Programming software platform based on .NET
WT	: Wavelet Transform
DWT	: Discrete Wavelet Transform
ANN	: Artificial Neural Network
ARIMA	: Autoregressive Integrated Moving Average
CWT	: Continuous Wavelet Transform
EMD	: Empirical Mode Decomposition
SVM	: Support Vector Machine
NWP	: Numerical Weather Prediction
MAE	: Mean Absolute Error
MSE	: Mean Squared Error
MAPE	: Mean Absolute Percentage Error
KGE	: Kling–Gupta Efficiency
PBIAS	: Percent Bias
SWAT	: Soil and Water Assessment Tool
WRF	: Weather Research and Forecasting model
HSPF	: Hydrological Simulation Program Fortran
ML	: Machine Learning
ECMWF	: European Centre for Medium-Range Weather Forecasts
PCA	: Principal Component Analysis

PRISM : Parameter-elevation Regressions on Independent Slopes Model  
STFT : Short-Time Fourier Transform  
IMF : Intrinsic Mode Function  
IQR : Interquartile Range  
SSA : Singular Spectrum Analysis  
LSTM : Long Short-Term Memory  
XGBoost : Extreme Gradient Boosting



## LIST OF TABLES

<b>Table 3.1.</b> Summary of Dataset Variables.....	12
<b>Table 3.3.1.</b> Common Wavelet Families Types.....	15
<b>Table 4.1.</b> Genetic Programming Parameters Used in GPdotNET.....	20
<b>Table 4.2.1.</b> The Evaluation of Raw Data Metrics.....	22
<b>Table 4.3.2.</b> Results Of Wavelet-Decomposed Input Features. ....	30
<b>Table 4.4.1.</b> Matrices of GP Model with Lagged Data Features.....	34
<b>Table 4.5.1.</b> Matrices of Correlation Analysis of Lagged Data.....	38
<b>Table 4.6.1.</b> Performance of the GP Model Using Selected Lagged Meetrics.....	42
<b>Table 4.7.1.</b> Matrices of the GP Model.....	46
<b>Table 4.8.1.</b> Comparative of All GP Models.....	50

## LIST OF FIGURES

<b>Figure 3.1.1.</b> Antalya Map.....	11
<b>Figure 3.2.1.</b> Tree Structure of GP.....	17
<b>Figure 4.2.1.</b> Training of Raw Data GP Model .....	22
<b>Figure 4.2.2.</b> Testing Raw Data GP Model .....	23
<b>Figure 4.2.3.</b> Raw Data GP Prediction.....	23
<b>Figure 4.2.4.</b> Raw Data GP Model Simulation.....	24
<b>Figure 4.2.5.</b> Raw Data GP Fitness Simulation.....	24
<b>Figure 4.2.6.</b> Tree Structure of Model-1.....	25
<b>Figure 4.3.1.</b> Dew point Decomposed.....	26
<b>Figure 4.3.2.</b> Humidity Decomposed.....	27
<b>Figure 4.3.3.</b> Tmax Decomposed.....	28
<b>Figure 4.3.4.</b> Wind Speed Decomposed.....	29
<b>Figure 4.3.5.</b> Wavelet Decomposition Training Data.....	30
<b>Figure 4.3.6.</b> Wavelet Decomposition Testing Data.....	31
<b>Figure 4.3.7.</b> Wavelet Decomposition Prediction Data.....	31
<b>Figure 4.3.8.</b> Wavelet Decomposition of GP Model Simulation.....	32
<b>Figure 4.3.9.</b> Wavelet Decomposition of GP Fitness Simulation.....	32
<b>Figure 4.3.10.</b> Tree Structure of Model-2.....	33
<b>Figure 4.4.1.</b> Lagged of Training Data.....	34
<b>Figure 4.4.2.</b> Lagged of Testing Data.....	35
<b>Figure 4.4.3.</b> Lagged of Data Prediction.....	35
<b>Figure 4.4.4.</b> GP Model Simulation.....	36
<b>Figure 4.4.5.</b> GP Fitness Simulation.....	36
<b>Figure 4.4.6.</b> Tree Structure of Model3.....	37

<b>Figure 4.5.1.</b> Training Data (Correlation-based Lagged Features).....	38
<b>Figure 4.5.2.</b> Testing Data (Correlation-based Lagged Features).....	39
<b>Figure 4.5.3.</b> Prediction (Correlation-based Lagged Features).....	39
<b>Figure 4.5.4.</b> GP Model Simulation (Correlation-based Lagged Features).....	40
<b>Figure 4.5.5.</b> GP Fitness Simulation (Correlation-based Lagged Features).....	40
<b>Figure 4.5.6.</b> Tree structure of Model4.....	41
<b>Figure 4.6.1.</b> Training Data (Selected Lagged Features).....	42
<b>Figure 4.6.2.</b> Testing Data (Selected Lagged Features).....	43
<b>Figure 4.6.3.</b> Prediction (Selected Lagged Features).....	43
<b>Figure 4.6.4.</b> GP Model Simulation (Selected Lagged Features).....	44
<b>Figure 4.6.5.</b> GP Fitness Simulation (Selected Lagged Features).....	44
<b>Figure 4.6.6.</b> Tree Structure of Model-5.....	45
<b>Figure 4.7.1.</b> Training Data (Lagged with Wavelet-Decomposed).....	46
<b>Figure 4.7.2.</b> Testing Data (Lagged with Wavelet-Decomposed).....	47
<b>Figure 4.7.3.</b> Prediction (Lagged with Wavelet-Decomposed).....	47
<b>Figure 4.7.4.</b> GP Model Simulation (Lagged with Wavelet-Decomposed).....	48
<b>Figure 4.7.5.</b> GP Fitness Simulation (Lagged with Wavelet-Decomposed).....	48
<b>Figure 4.7.6.</b> Tree Structure of Model-6.....	49

## PREFACE

This master's thesis titled "Advanced Genetic Programming Models for Precipitation Forecasting" represents a significant milestone in my academic journey at Antalya Bilim University, in the Department of Electrical and Computer Engineering. Completing this thesis has been both a challenging and rewarding experience. I would like to express my deepest gratitude to my supervisor, Prof. Dr. Ali Danandeh Mehr, whose invaluable guidance, encouragement, and constructive feedback shaped this work. I also extend my heartfelt thanks to the faculty members, colleagues, and friends who provided support, insightful discussions, and motivation throughout this process. Lastly, I am profoundly grateful to my family for their unwavering love, patience, and encouragement, which gave me the strength to complete this research successfully. It is my sincere hope that the findings of this study will contribute meaningfully to the field of hydrological forecasting and inspire further research in this important domain.



## 1. INTRODUCTION

A precise evaluation of precipitation plays a crucial role in guiding water resource management and agricultural planning globally. It is crucial in tackling floods and droughts and in the proper distribution of water resources among people. The ability to forecast rainfall accurately is essential for the sustainable development of water-stressed areas, and the reduction of the risks of water-related problems also helps prevent desertification and promotes efficient water use. Precipitation forecasting is a very complex and non-linear process. Therefore, precision is the biggest challenge of all. Traditional mathematical models often fail to capture the complexity of rainfall patterns. With this in mind, it is essential to carry out this process through contemporary computer methods, which will succeed in copying and analyzing complex precipitation patterns (K. Rodríguez-Vázquez, 2012).

### 1.1.Challenges in Traditional Rainfall Prediction Models

Researchers have continued applying three main forecasting models, which include statistical regression as well as autoregressive integrated moving average (ARIMA) and numerical weather prediction (NWP) models for precipitation estimation. The nonlinear nature of rainfall along with its unpredictable variability makes these prediction techniques yield subpar results (Eralda Gjika, 2019).

Statistical models presume linearity and stationarity, which are often not fulfilled in hydrological data that have been observed in reality. NWP models, on the other hand, require enormous computing resources and are extremely sensitive to precise atmospheric input data, making them less feasible for short-term rainfall prediction (Arpit Kapoor, 2023).

Machine learning models like artificial neural networks (ANNs) and support vector machines (SVMs) have been investigated as promising alternatives. Although these models are promising, they are prone to overfitting, poor interpretability, and hyperparameter sensitivity. To address these drawbacks, hybrid models combining various approaches have been suggested as viable alternatives for rainfall prediction (Dongsheng Li, 2023).

### 1.2.Motivation to Use Genetic Programming

Genetic programming (GP) is an evolutionary algorithm inspired by Darwinian natural selection. GP's automatic mathematical expression or symbolic regression model development is highly fitting to observed data and well-suited for hydrological applications. Unlike traditional black-box models, GP produces interpretable mathematical expressions that researchers can analyze directly (D. Klotz, 2017).

In recent years, the use of GP in precipitation forecasting has attracted significant interest. Studies on GP models indicate that they can outperform traditional statistical and machine learning (ML) methods in capturing complex rainfall patterns (Babovic, 2019). One of the strengths of GP is its ability to generate symbolic equations, which makes the method highly flexible. This flexibility reduces the need for extensive feature

engineering, making GP a suitable tool for complex hydrological modeling (Danandeh Mehr, 2020).

Given these advantages, this thesis proposes a new hybrid approach that combines GP and decomposition techniques applied to signals, aimed at maximizing the potential of the model for developing patterns in rainfall time series data for an improved quality of short-term forecasting for rainfall.

### **1.3. Research Objectives**

This research is mainly focused on improving the short-term rainfall forecasting performance by introducing a new hybrid mechanism that incorporates genetic programming and signal decomposition. The specific goals are as follows:

- To gather and pre-process the historical rainfall data in Antalya City to make sure data quality is enough for model training.
- To use an emerging signal decomposition method to reveal features of interest in the rainfall time series.
- Development of a GP-based symbolic regression model using decomposed rainfall.
- Analyzing the model performance with Nash-Sutcliffe Coefficient of Efficiency (NSE) and Root Mean Squared Error (RMSE) metrics

### **1.4. Research Questions**

The following questions shall form the contents of discussion with the emergent findings of this study.

1. How does the performance of GP compare to traditional methods in daily precipitation forecasting?
2. Does wavelet decomposition improve or review of the GP performance?
3. What is the effect of lagged data on the GP performance?

### **1.5. Structure of the Thesis**

The rest of this thesis is structured as follows: Chapter 2 Literature Review: Summarizes previous research in rainfall predictions, signal decomposition approaches, and genetic programming in hydrology. Chapter 3 Methodology outlines the data collection process, preprocessing methods, model development, and performance evaluation techniques. Chapter 4 results and discussion Provide the results of the study, a comparison of the proposed model with existing methods, and discussions. Conclusion of this work and direction of future research are covered in Chapter 5.

## 2. LITERATURE REVIEW

The precise forecasting of rainfall is of great importance in hydrological and meteorological applications, which encompass flood risk management, water resource planning, and agricultural productivity. During the advancement of the years, many technologies were conceived for the prediction of the rains and included traditional statistical models, machine learning techniques, and hybrid models. Using genetic programming (GP), an evolutionary computing method, is one of the latest techniques for making complex, non-linear relationships in rainfall time series data visible. This article is a review of the background materials on traditional as well as modern rainfall modeling (Danandeh Mehr, 2020).

### 2.1. Hydrological Forecasting and Rainfall Prediction Models

Accurate predictions of rainfall and hydrological processes are scaling up the planning and decision-making in agriculture, urban development, and environmental conservation; thus, they are indispensable components of water resource management, flood forecasting, and disaster mitigation. Hydrological forecasting and rainfall prediction are the core elements of these disciplines. The accuracy and reliability of the models have improved significantly due to the development of various models over the years (Zhang, Liu, Zhang, & Li, 2025).

Hydrological forecasting involves the use of mathematical models to simulate the flow and distribution of water within a catchment area the course and distribution of water within a catchment area. These models are generally classified as either deterministic or stochastic. SWAT and HSPF are physically based models that simulate hydrological processes using governing equations (Escamilla-Rivera, 2022). Such models require comprehensive input data—including topography, soil characteristics, and land use—and are typically used for long-term simulations and scenario analysis.

There's an alternative type of model, which is a stochastic model that uses statistical methods and is designed to describe the incalculable variability and risk. They are autoregressive integrated moving averages (ARIMA) and Markov chain models. They are commonly utilized in the short-term forecast; in this case, they are extremely useful because physical processes are not well mechanized and there is an absence of data (Katimon et al., 2018).

Precipitation-predicting models are part of the hydrological prognosis system that focuses mainly on the forecast of the scale, timing, and distribution of precipitation. The classification of these models falls into the categories of numerical weather prediction (NWP) models, statistical models, and machine learning models. The NWP model is a water research and forecast model that's based on the equations governing the numerical solution of atmospheric dynamics and physics, which is the Weather Research and Forecasting (WRF) model (Han & Hong, 2018). They necessitate colossal computation and the use of vast input data, but the projections provided are comprehensively detailed and distributed.

Data-based regression techniques and time series analysis are some of the statistical models employed to find the trend and link between independent and dependent variables. These equations are quite simple and marginally computational, but they might not accurately visualize the complex nonlinear relationships of a given weather phenomenon (Chavan & Momin, 2019). Machine learning frameworks, for example, using artificial neural networks (ANNs) (Haykin, 1999), support vector machines (SVMs) and random forests (Breiman, 2001), have been the most favorable models in recent times owing to the ability to depict complex nonlinear relationships and the capability to handle a diverse set of data. These models are often complemented by other means of improving data quality, such as data preprocessing and feature selection (Yaseen, 2019).

Even with the current progress made in weather forecasting and rain forecasting models, quite a few challenges still persist. These comprise the necessity for accurate input data, the intricacy of the models, and the unpredictability connected with the forecasts. The subsequent studies must concentrate on the more accurate and reliable model development, the rise in data availability and quality, and the incorporation of different modeling methods to address these issues (Kratzert et al., 2018).

## **2.2. Traditional Methods for Rainfall Forecasting**

One of the most frequently used traditional ways to forecast rainfall is the empirical approach to weather forecasting, which is the oldest model existing today. These methods could be loosely binned into three categories, which are empirical, statistical, and very physical models of different types. Empirical models are the ones that, to some extent, are based on the relationship that is there between rainfall and other meteorological elements like temperature, humidity, and wind speed. These models are usually simple and quite simple, too, but they might disregard the complex impacts of the different variables (Kumar et al., 2024).

Statistical models, the likes of regression analysis, time series analysis, and stochastic models, based mainly on historical data, are used to discover the patterns and trends in rainfall. Some regression models, such as linear and multiple regressions, find the relations between rainfall and the other elements (Wilks, 2011). Time series such as autoregressive integrated moving average (ARIMA) and seasonal decomposition are the most common methods that have been used to predict and forecast the historical data of rainfall (Benziane, 2024).

Physical models, including the numerical weather prediction (NWP) models, utilize the fundamental equations that govern the atmospheric processes. These models, among them the Weather Research and Forecasting (WRF) model and the European Centre for Medium-Range Weather Forecasts (ECMWF) model, are computationally heavy and request significant amounts of input data. However, although they give accurate and high-resolution weather forecast data, the processes are very complex, and the data are often of very high resolution (Kalnay, 2002).

NWP models are the usual draws of both long-range prediction and very short-term predictions and possess the unique ability to foretell extreme weather scenarios (Kalnay, 2002). The classical methods of rainfall forecasting, although being

widely used, still come with several drawbacks. Empirical models that may not be able to uncover the entire web of interactions between individual variables and their correct answers may be prevented by the lack of quality data (Wilks, 2011). Statistical models that could be limited may have difficulties in finding the intersections that are not linear between the rainfall processes, and their performance could deteriorate when they are applied to other regions or time periods (Box et al., 2015). Physical models are more influential; they not only take a lot of time to input and need high-quality data but could still be unavailable (Kalnay, 2002).

Recently, there is an increasing fascination with integrating conventional methods with contemporary tactics such as machine learning and data assimilation to boost the precision and constancy of rainfall forecasts all over the world. For instance, the applications of machine learning models, such as artificial neural networks (ANNs) and support vector machines (SVMs), have been successful in doing a great job by holding the historical and present relation between them (Yaseen et al., 2016). Data assimilation techniques, like the Kalman filter and ensemble Kalman filter, which are the kinds of techniques of data assimilation that have been in use, have been recently employed to model a weather forecast ingesting observational data and to forecast the weather over the entire area of interest in much more detail than by classical data assimilation (Kalnay, 2003). To sum up, the older techniques of rainfall prediction have gained an indispensable place in the development of modern forecasting techniques. However, irrespective of this, the methods are still being made use of, and their findings have quite important implications for the understanding of meteorological phenomena. The upcoming studies need to center on the association of traditional methods alongside advanced techniques for the enhancement of forecasting capabilities (Yaseen et al., 2016).

### **2.3. Genetic Programming in Hydrology**

The hydrological community has turned to genetic programming (GP), an evolutionary computation algorithm, due to the numerous capabilities it provides, the accurate depiction of nonlinear relationships and the necessary adaptability to various datasets. GP is a genetic algorithm that is inspired by the concepts of natural selection and genetic evolution, whereby through many generations, the candidate solutions become the fittest one (Koza, 1992).

While the use of genetic programming in hydrology has been mostly confined to applications like rainfall-runoff, streamflow forecasting, and prediction of water quality, there have been occasional instances of its application to other areas such as fuzzy logic, model parameter estimation, and input optimization (Muttill & Lee, 2005; Savic et al., 1999). Rainfall runoff, streamflow forecasting, and water quality prediction are just some of its applications. However, it is possible to construct GP models that will adequately capture the complex relationships by adopting a genetic algorithm. GP models, for example, are significantly broader in handling different types of data, like time series, spatial, or categorical data, so the applications of GP models in hydrology are also very broad, such as filtering, erosion, and sedimentation (Giustolisi & Savic, 2006).

One of the main advantages of genetic programming that we might point to is the ability to automatically choose the relevant variables that need to be fed to the model, as

well as the model structures among many possible alternatives, and its direct consequences are a decrease in the man-hour need and the need to have an expert (Muttill & Lee, 2005). In the field of hydrology, there are noises and missing data that are almost always existent in the data that is collected, but GP models are still robust enough to not only cope with such problems but also to extract useful information from such noise. The equations of GP models are also concise and easy to explain simply because GP models produce explicit mathematical expressions that, in detail, describe the dependence of the input and output variables (Sivapragasam et al., 2007). The only reason for this is that GP models can be computationally intensive, especially for large datasets and complex problems. GP models tend to be computationally expensive in the case of large datasets and complex problems. The performance of GP models may be severely affected as a result of the choice of parameters like population size, mutation rate, and crossover rate that may need tuning. The GP models can also suffer from undesirable effects from overfitting, especially when the training data are limited (Žegklitz & Pošík, 2015). Furthermore, GP models may display overfitting problems mainly due to a small number of training samples that may also be considered noisy.

Recently, people started to get more and more into the subject of including GP with such methods, like data preprocessing, feature selection, and ensemble modeling, to increase its success. Normalization and outlier detection are some of the commonly used data preprocessing techniques to enhance the quality of the input data. Feature selection is also a technique, such as principal component analysis (PCA) and mutual information, used to decrease the input data dimensionality and enhance the GP modeler's interpretability. Ensemble modeling methods, for example, bagging and boosting, have been practiced to raise the stability and accuracy of GP models (Virgolin, 2021).

In short, GP is a good tool for explaining complex non-linear relationships in different aspects of hydrology. Even so, it should be noted that the hydrologic case is one of the situations that may complicate its optimization. However, it has been shown that even rigorous hydrological problems can be effectively solved using GP, and it brings very encouraging results. In the future, it is necessary to address the limitations in GP and offer novel ways for the method to be applied in hydrologic sciences (Parasuraman & Elshorbagy, 2008).

#### **2.4. Signal Decomposition Techniques in Time Series Analysis**

Signal decomposition techniques are commonly used in time series analysis to divide a complicated signal into its parts that can be assessed independently. These methods are especially beneficial in the analysis of time series that are not stationary and have nonlinear behavior, which are commonly found in hydrological and meteorological databases (Huang et al., 1998; Labat, 2005). Among the most typical signal decomposition methods is the Fourier Transform, which splits a signal into its frequency elements. The Fourier Transform is most useful when the signals are periodical in nature; however, signals that are non-stationary with the frequency components that are changing with time should not apply it (Mallat, 1999). To solve this, the Short-Time Fourier Transform (STFT) was created, which, as the name suggests, cuts the signal into short segments in time and then uses the Fourier Transform for each segment (Cordero & Rodino, 2008).

Nevertheless, STFT has a constant time-frequency resolution, which may not be enough for the signals with a frequency that is not stable in the time domain.

The wavelet transform is a method that breaks the signal down into the form of components, and time goals are different by providing variable time-frequency resolution. The wavelet transform decomposes a signal into wavelets, which are localized in (time-time and time-frequency) domains. This makes the wavelet transform particularly useful for analyzing non-stationary signals, where frequency tone varies by time (Daubechies, 1990). This has been the main application in hydrology for a variety of situations, for example, rainfall-runoff modeling, streamflow forecasting, and drought analysis (Adamowski, 2008; Labat, 2005).

Empirical Mode Decomposition (EMD) is a unique signal decomposition technique that is capable of performing signal decomposition by decomposing into Intrinsic Mode Functions (IMFs), which are in the range of the signal. One asset of the EMD technique is that it can quickly and accurately provide information on nonlinear and non-stationary signals, which do not contain any further assumptions regarding the signals in its code. One of the achievements gained from the application of EMD in hydrology is in the environment area. The EMD method has become well-known for rainfall analysis, streamflow forecasting, and water quality prediction (Huang et al., 1998).

There has been an increasing trend towards signal decomposition methods being fastened with the other methods, such as machine learning and statistical modeling, to increase the correctness and reduce inaccuracies of the time series analysis. For instance, artificial neural networks (ANNs) combined with the wavelet transform have been able to enhance the performance of rainfall-runoff models (Adamowski, 2008). Similarly, EMD has been used in tandem with support vector machines (SVMs) to improve the accuracy of streamflow forecasting (Katipoğlu et al., 2023). These dual schemes have turned out to be very promising and can bring a new level of quality to time series analysis in hydrology.

Summing up, the process of signal decomposition enhances the efficiency and capabilities of non-stationary feature and hydrologic component studies. It is notable that these methods have been used to solve various tasks and have shown promising results. The future direction should concern the testing of new signal decomposition techniques as well as the application of them together with other methods to significantly improve the accuracy and reliability of time series analysis in hydrology, which has been reported to increase the periodicity of daily precipitation time series or variance PRISM estimates through decomposition (Labat, 2005).

## **2.5. Hybrid Models in Rainfall Prediction**

Model hybrids of rainfall prediction use techniques from different models to refine the precision and dependability of the rainfall projection. Along with the traditional ways of statistical and physical models, these models typically reflect the new methods recently introduced, like machine learning and signal decomposition. To illustrate the complex difficulties that arise during a rainfall situation, traditional methods, such as statistical and

physical models, are often used with contemporary techniques, such as machine learning and signal decomposition.

The most common hybrid approach is the combination of numerical weather prediction (NWP) models with machine learning models. NWP models, like the Weather Research and Forecasting (WRF) model, exclusively base their detailed and spatially distributed predictions on the physical laws that govern the processes taking place in the atmosphere. Nonetheless, the models are demanding in terms of computation and may still fail to depict the intricate and extremely complex nonlinear relationships existing within the rainfall processes. ML models, such as artificial neural networks (ANNs) and support vector machines (SVMs), are used to strengthen the performance of NWP models by recognizing these intricate relationships. ANNs, for instance, have been used to post-process the outputs of NWP models for an accurate rainfall forecast (Waqas et al., 2024).

Hybrid models are a common approach that is the use of statistical models and machine learning models together. Statistical models approaching autoregressive integrated moving average (ARIMA) and regression models are borrowed from the past and are used to identify patterns and trends in rainfall. However, these models may not take into account the complex nonlinear relationships inherent in rainfall processes. Machine learning models, such as random forests and gradient boosting machines, are the ones that take over the performance of statistical models by catching such complex relationships. For instance, random forests have been employed to enhance the accuracy of rainfall forecasts by uniting the predictions of multiple statistical models (Katimon et al., 2018).

Several hybrid models are used for rainfall prediction in combination with the signal decomposition techniques of the wavelet transform and empirical mode decomposition (EMD). EMD and wavelet transform work on the principle of breaking a complex rainfall signal associated with the climatic parameters into simpler parts that can be analyzed separately. For example, the addition of wavelet transform to ANNs has helped in the rainfall-runoff model's better performance (Katipoğlu et al., 2023).

Lately, there has been incredible growth in the use of the developed hybrid models to improve the precision and truth of the forecasts of the rainy environment. For example, NWP coordination has been calculated on the use of machine learning synchronous with signal decomposition techniques to refine precipitation predictions. Such complex hybrid modes have quite a potential and hence can be employed for enhancing the performance of models being designed to predict rainfall accurately (Waqas et al., 2024).

Hybrid models, which are a combination of different forecasting approaches, can be the basis for the best rainfall forecast algorithm; they are capable of increasing the forecast accuracy and reliability. They have been increasingly adopted for numerous applications and have displayed promising results. Forthcoming research should primarily look at introducing new hybrid systems and applying them with other methods to improve the precision and reliability of rainfall forecasts (Laleh Parviz et al., 2023).

## **2.6. Evaluation Metrics for Forecasting Models**

Evaluation metrics are necessary tools to gauge the efficiency of forecasting models and spot the differences among specials. These metrics offer numbers such as

precision, error rate, and robustness, all of which are vital for decision-making and the selection of the model (Willmott & Matsuura, 2005).

The most often used evaluation metric is the Mean Absolute Error (MAE), which calculates the average absolute deviation between the forecast and the actual values. The MAE is simple to understand and shows the insensitivity of the model to outliers, yet it does not offer any kind of detail regarding the errors (Chai & Draxler, 2014). On the other hand, the mean squared error (MSE) is also one of the ways to judge the precision of the forecaster that manifests itself as the bias squared. MSE is the square of the mean of the squared differences between the forecasted and observed values. It reads as the average squared deviation of the forecast from the actual data. MSE has a problem with outliers because the model assigns less weight to smaller errors. However, the MSE is more powerful in the sense that it gives greater priority to larger errors, and therefore, it is valuable in the detection of models that yield large errors.

The Root Mean Squared Error (RMSE) is the square root of MSE, and it is employed quite often as the simplest method of describing the error in terms of the same units as the observed values. Moreover, RMSE is a rugged error metric that is challenged for upside, and it is concerned with larger errors (Legates & McCabe Jr, 1999). One more efficiency measure is the Mean Absolute Percentage Error (MAPE); it shows you the average absolute percentage difference between the forecasted and the observed values. MAPE can be easily expressed and is suitable for comparing two or more models if data is available, but the selective behavior it shows is underestimating the result (Makridakis, 1993).

The coefficient of determination ( $R^2$ ) is a measure of the relationship between the original values and the predicted values. The greater the value of  $R^2$  is, the better the description the model provides of the data. Despite this,  $R^2$  can be unreliable in the case of model overfitting and the non-linear relationship between the predicted and observed values (Krause et al., 2005).

Models that were also used in the evaluation of hydrology, like Nash-Sutcliffe Efficiency (NSE), Kling-Gupta Efficiency (KGE), and Percent Bias (PBIAS), are popularly used. NSE tests the model's ability to perform considerably better in predicting observed values than the mean of the observations. The index ranges from  $-\infty$  to 1; the higher the index, the better the model performs (Nash & Sutcliffe, 1970).

KGE also combines correlation, bias, and variability for the complete assessment of model performance, and it also varies from  $-\infty$  to 1 (Gupta et al., 2009). Lately, there's been a lot of interest in the concept of using various evaluation metrics at the same time for model performance assessment. This method gives a more comprehensive assessment and helps to identify the strengths and weaknesses of every model. The model is a model; one example could be of low MAE yet high PBIAS, which means accuracy is not an issue but bias might be one of its sides. In the same way, a model might display a high  $R^2$  but a low NSE; in other words, it is good at explaining the variance but not as good at predicting the actual data (Chai & Draxler, 2014).

As a result, evaluation metrics are the main point of evaluating and comparing forecasting models. They are not just about the accuracy of models but also include the reliability and robustness that are essential for decision-making and model selection. Further research would be anticipated to introduce new evaluation metrics while simultaneously prioritizing the use of several metrics for a more comprehensive performance assessment (Gupta et al., 2009; Willmott & Matsuura, 2005).



### 3. METHOD

This study compares several modeling approaches for precipitation forecasting, with a primary focus on the application of Genetic Programming (GP). The method starts with separating the raw meteorological time series into several sub-signals. Each of these is then subjected to symbolic regression via GP. The last part of the forecast is a combination of all the sub-models' forecasts. The above-mentioned technique has the intention of extracting in the most optimal way both linear and nonlinear patterns that are present in the data set.

#### 3.1. Data Collection and Preprocessing

##### 3.1.1. The Area of Study

Antalya, located on the southwest coast of Turkey, has a Mediterranean climate with hot, dry summers and mild, wet winters. The weather in the city is subject to considerable changes in the rainy season, making the town an ideal study case for advanced rain prediction models. The geographical and climate conditions are such that it is dependent on the provision of accurate and reliable forecasting models regarding water resources, agriculture, and disaster prevention .



**Figure 3.1.1.** Antalya Map

**Source :** <https://nationalparksofturkey.com/antalya-maps/>

##### 3.1.2 Data Description

The dataset used in this research are daily meteorological records that was collected between January 1, 2016, and December 31, 2023. The dataset was retrieved from the NASA Global Precipitation Measurement (GPM) website (<https://gpm.nasa.gov/>), a satellite mission with global coverage that provides high-resolution, near-real-time precipitation data, widely used in hydrology. The data was

validated for accuracy to guarantee that no missing information or data issues exist by employing quality checks for NASA GPM data. It was certain that no missing values or data gaps were present after the thorough investigation. The whole data were tested with great care to ensure they were consistent, and no gaps were found. A few words about the structure of the dataset are shown in Table 3.1.

**Table 3.1.** Summary of Dataset Variables

Data	Average	Maximum	Minimum	Mode	Skewness	Standard Diviation
Maximum daily temperature(°C)	23,96	41,48	2,85	28,88	0,01	9,01
Minimum daily temperature(°C)	12,91	27,66	-3,94	23,48	-0,14	7,38
Average daily temperature(°C)	18,44	33,26	-0,32	19,27	-0,02	8,06
Dew point(°C)	8,85	20,32	-10,33	10,77	-0,60	5,50
Atmospheric pressure at sea level (hPa)	96,05	97,57	94,52	95,97	0,32	0,50
Relative humidity(%)	60,59	97,89	24,87	41,63	0,09	16,25
Wind speed(m/s)	2,11	6,86	0,70	1,81	1,86	0,84
Incoming solar radiation(W/m <sup>2</sup> )	8,10	14,07	0,68	13,47	-0,09	3,54
Daily precipitation (target) Mm	1,90	86,69	0,00	0,00	5,78	5,82

### 3.1.3. Data Cleaning

Before the analysis, the dataset went through a thorough cleaning process that dealt with usual problems such as missing variables, extreme values, and inconsistencies.

**Handling Missing Values:** The incomplete data points were recognized and changed using the proper imputation techniques. For instance, when temperature values were not indicated, they were filled using linear interpolation, while mode imputation

helped us solve the missing categorical data issue. Here, we can find that the imputation approach taken was in line with the practices used for meteorological data preprocessing.

**Outlier Detection and Treatment:** outlier detection was performed by using the interquartile range (IQR) method to distinguish significant observation values. Information lying 1.5 times lower than the first quartile or higher than the third quartile was regarded as an outlier, and the values were corrected or deleted, as required by the situation and the plausibility of the values. A case in point is the unavailability of an unexpectedly small precipitation value of 0.5 mm, which was perceived as an outlier using the IQR method and thus was taken away from the dataset to ensure the consistency of the data.

**Consistency Checks:** The dataset was checked for consistency between the units and the formats. For example, it was ascertained that the temperature readings were in degrees Celsius, and a mechanism was developed to eliminate monetary differences that were consistent with the uniformity across the dataset.

### 3.2. Data Transformation

A preliminary data-messaging set of operations was applied to the raw data to get them into the correct format for the modeling stage.

**Normalization:** The numerical information was scaled to a range of 0 to 1 using the Min-Max scaling method. It is a known fact that genetic programming algorithms are dependent on the input data scales; thus, normalization becomes a key pre-processing task to enhance the model's convergence ability.

**Encoding Categorical Variables:** If any categorical variables were present, they were transformed into binary vectors using the one-hot encoding method. This technique ensures that the model interprets categorical inputs as distinct classes without assuming any ordinal relationship among them, which is essential for preserving the true structure of the data.

**Temporal Feature Engineering:** Several additional variables were created in order to capture the different aspects of a time-based situation, for example, a time series (1) with periodic properties. For example, the features of day-of-year and month were introduced to the model to learn the periodical occurrence of rainfalls during the year.

The data collection and pre-processing procedures followed in this study are essential to make sure of the reliability and accuracy of the proposed hybrid model forecasting. Through proper methodology, such as addressing the issues of the incomplete cases, outlier observations, and data incompatibility, and performing the necessary changes, the dataset was finally set for advanced modeling techniques like wavelet decomposition and GP. Therefore, these preprocessing steps not only improved data quality but also enhanced the robustness and generalizability of the hybrid forecasting model.

### 3.3. Signal Decomposition with Wavelet Transform

Wavelet transform is a strong mathematical tool for the analysis of non-stationary or time-varying signals. In contrast to other signal processing approaches like the Fourier Transform, which describe the signal in sine and cosine waves and output information solely in the frequency domain, wavelet transform enables a signal to be expressed both in the time domain and in the frequency domain. Wavelet analysis is hence well-suited for investigating complicated, real-world signals that contain transient characteristics or localized structures in the time domain. The principal concept of the wavelet transform is the separation of a signal into a series of basis functions referred to as wavelets, which are the result of translating and scaling a prototype wavelet, or the "mother wavelet." This allows for the examination of signal characteristics at various resolutions, both the low-frequency trends and the high-frequency detail. By adjusting the transform, it is able to zoom in on short-duration, high-frequency parts or zoom out to analyze long-duration, low-frequency parts, depending on the purpose of the analysis.

The wavelet transform is typically divided into two broad categories: There exist two broad types of wavelet transform: Continuous Wavelet Transform (CWT) and Discrete Wavelet Transform (DWT). The former provides a highly redundant representation and finds application in signal visualization and feature extraction, while the latter gives a compact and more efficient representation suitable for signal compression and numerical modeling. Discrete Wavelet Transform is applied more in practice as a popular choice, considering its low cost of computation and suitability for applications in algorithms.

The mother wavelet selection is a critical part of wavelet analysis. Every family of wavelets, including Haar, Daubechies, Symlets, Coiflets, Biorthogonal, Morlet, and Mexican Hat, possesses different mathematical characteristics to suit particular applications. For example, the Haar wavelet, being the most elementary, proves to be superior in the detection of abrupt changes in the signal. Daubechies wavelets are smoother and more regular in their basis functions, a property appropriate for common signal denoising. Coiflets and symlets are a compromise between symmetry and smoothness and are best suited for exact signal reconstruction and compression. Conversely, continuous wavelets, including Morlet and Mexican Hat, are well utilized in physics and geoscience for event detection and time-frequency analysis.

In both hydrological and meteorological applications, the wavelet transform has been a powerful tool in time series analysis. Climate and meteorology are subject to non-stationarity and non-linear effects from the intricate interaction between atmospheric processes. Such complexity may not be adequately accounted for by conventional statistical models. Wavelet decomposition, in its ability to decompose the initial time series into different sub-signals at different scales, extracts concealed structures and temporal patterns that cannot be discerned elsewhere. Wavelet-based techniques are, therefore, well-suited for precipitation prediction, drought detection, river discharge modeling, and interpreting environmental information.

continuous wavelet transform (CWT) and discrete wavelet transform (DWT). The former provides a highly redundant representation and is utilized in signal visualization and extraction of features, while the latter provides a compact and more efficient

representation, which is applicable in signal compression and numerical modeling. The discrete wavelet transform is, in fact, more often used in practice since it is less computationally costly and feasible in algorithmic applications.

**Table 3.3.1.** Common wavelet families types

Wavelet Family	Type	Features	Common Applications
Haar	Discrete	Simple, fast, orthogonal, captures sudden changes	Abrupt signal changes, image compression
Daubechies (dbN)	Discrete	Compact support, orthogonal, smooth basis functions	Signal processing, denoising
Symlets (symN)	Discrete	Symmetric, similar to Daubechies, improved phase	Data compression, forecasting
Coiflets	Discrete	Near symmetry, more vanishing moments	Feature extraction, denoising
Biorthogonal	Discrete	Symmetry, linear phase, perfect reconstruction	Medical imaging, signal reconstruction
Morlet	Continuous	Complex wavelet, good time-frequency localization	Feature detection in non-stationary data
Mexican Hat	Continuous	Second derivative of Gaussian, smooth and localized	Event detection, feature analysis

In summary, wavelet transform is more than a mathematical curiosity; it is rather a powerful and highly useful tool for modern data analysis. Its versatility for application in any domain, scalability in coordination with computation, and capacity for uncovering obscured information serve to make it a key to solving modern, complicated, real-world issues.

### 3.3.1. Rationale for the Use of Wavelet Decomposition

Precipitation time series usually display non-linear features and are not stationary due to the interaction of different atmospheric phenomena. Due to these complexities, the choice of a decomposition method that can capture the information at a very local scale in a two-dimensional space is justified. Wavelet application comes to satisfy the above requirement through the theorem that the signal can be expressed by the components being a function of the frequency and the time.

### 3.3.2. Selection of Mother Wavelet

The mother wavelet that is chosen plays a mental role as it sets the characteristics of the decomposed components. Among wavelets that were commonly used for hydrological purposes, Daubechies, Coiflets, and Symlets were the most popular. However, this study goes with the choice of Daubechies wavelet (db4) because of its orthogonality and the ability to be used for both smooth and sharp variations in the precipitation data.

### 3.3.3. Decomposition Process

The wavelet decomposition process is defined in the following steps: Signal Decomposition: The series of original precipitation times is sent through several high-pass as well as low-pass filters to split the high-frequency (detail) and the low-frequency (approximation) parts.

**Downsampling:** Once the filtering process is complete, the data becomes half in size and is downsized by a factor of two to reduce instead of doubling the information.

**Recursive Decomposition:** The next step is to again break down the approximation signals recursively to obtain several levels of detail and approximation coefficients.

Mathematically, the representation of the decomposition is by:

$$x(t) = A_n(t) + \sum_{i=1}^n D_n(t)$$

that  $x(t)$  is the original signal,  $A_n(t)$  is the approximation at level  $n$ ,  $D_n(t)$  is the detail at level  $i$ .

## 3.4. Genetic Programming For Symbolic Regression

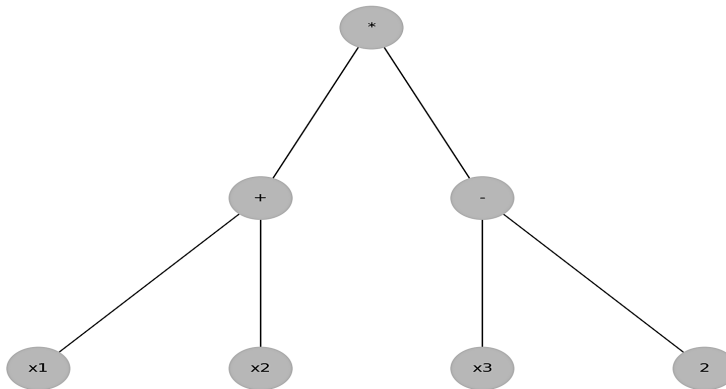
Genetic programming (GP) is an evolutionary algorithm mimicking natural selection and genetic features. It creates computer programs that solve problems by imitating the natural process of evolution. If we talk about symbolic regression, GP looks for the most accurate mathematical formulas for a given data set, which makes it especially useful for modeling the complex, nonlinear relationships in environmental data like precipitation (Koza, 1992).

The main function of GP is to improve a group of potential solutions step by step. The population that comprises the GP system is a group of models, each of which is discovered by the use of a mathematical model as a template, and the model is represented with a tree of functions (e.g., +, -, ×, ÷, sin(·), cos(·), etc.) and terminals (input variables and numbers). The model's performance is subsequently tested for the fitness function of the algorithm, which is typically centered on prediction error, and later, so-called genetic operations like crossover, mutation, and selection are used to manipulate the generation. By repeatedly using these operations, models are created that closely match the target variable or even daily rainfall amounts in general.

This study made use of the open-source GPdotNET as the principal tool for modeling. GPdotNET is a user-friendly graphical interface that allows symbolic regression with different configurations. The software also allows for the modification of key variables, such as population size, mutation rate, selection mechanism, and the number of generations, which are crucial to the problem at hand. The entire setup was carefully designed to optimize these parameters for the best possible accuracy while still being reasonably general, allowing them to be followed by another researcher. To be more specific, one of the critical features of GPdotNET is that it lets you see the process as the algorithm is executed, a very important point that the scientific community can appreciate because the evolution step by step can be observed; thus, changes in both model structure and model fitness during all that execution of the algorithm can be understood.

The modeling process began by inputting cleaned and normalized meteorological data, including variables such as temperature, humidity, atmospheric pressure, and wind speed, into the GP system. The software then initiated a symbolic regression approach and evolved mathematical models that mapped these input variables to the target output, which was daily precipitation. During the optimization phase, multiple runs were performed to ensure the stability and repeatability of the results. The models were evaluated using standard performance metrics such as root mean square error (RMSE) and Nash-Sutcliffe efficiency (NSE), ensuring that only the most accurate and least costly models were selected.

One of the distinguishing features of GP is the use of tree-based model structures. These trees represent symbolic mathematical formulas in a hierarchical manner. For example, a simple example of a model generated by GP could be as follows:



**Figure 3.2.1.** Tree Structure in GP

$$f(x_1, x_2, x_3) = (x_1 + x_2) * (x_3 - 2)$$

The corresponding tree structure for this expression would look like this:

The root node is the multiplication operator \*.

The left child of the root is the addition operation (x1 + x2).

The right child of the root is the subtraction operation ( $x3 - 2$ ).

This representation allows GP to manipulate the expressions flexibly during evolution. Each subtree can be interpreted as an independent component of the model and can be replaced or modified through genetic operations.

This type of symbolic output is a major strength of GP, offering interpretability and transparency, unlike black-box models such as neural networks. In scientific fields such as hydrology, this transparency is essential for validating the model behavior and understanding the influence of the variable.

In this study, the final GP models were obtained by training on wavelet-decomposed subsignals of the precipitation series. Each subseries was independently modeled using GP, and the results were combined to form the overall precipitation forecast. This combined approach significantly improved the accuracy and robustness of the forecast by capturing both local and global signal characteristics.

As a result, genetic programming for symbolic regression provides a powerful and flexible modeling framework capable of capturing the complex dynamics of precipitation time series. Its evolutionary nature, combined with the ability to interpret symbolic expressions and compatibility with mixed modeling, makes it an excellent choice for environmental forecasting tasks.

### 3.4.1. GPdotNET Tool

GPdotNET was developed using the Microsoft .NET platform and incorporates various evolutionary computing algorithms, including genetic programming (GP), artificial neural networks (ANN), and differential evolution. The software is known for its simple and graphical user interface, as well as the ability to display the modeling process, fitness function changes, and tree structure during the evolution process.

**Graphical User Interface (GUI):** The users can change the GP parameters with the help of the provided GUI.

**Customizable Function and Terminal Sets:** The sets of mathematical functions and input variables that the users need in the GP process can be defined by the competent users.

The visualization tools provide a clear depiction of model evolution and allow for effective tracking of the population's fitness progression across generations.

**Export Options:** After the evolution, the models can be used for various further analyses or different applications. These features embedded in GPdotNET have made it the ultimate choice for GP researchers in a myriad of fields. They can now run experiments on real-world problems without being forced to put their hands in the conundrum of the algorithm implementation.

In this research, GPdotNET was utilized for the generation of predictive models for the daily precipitation in Antalya Province, Turkey. The execution of this process has been implemented through the following steps:

**Data Preparation:** Data on the climate, which was handled first, includes data like temperature, humidity, and wind speed that were first cleaned and then imported into GPdotNET.

**Configuration of GP Parameters:** The most important settings in Genetic Programming (GP), including population size, number of generations, mutation and crossover rates, and the selection method, were adjusted.

**Definition of Function and Terminal Sets:** The functions (e.g., addition, subtraction, multiplication, and division) and the input variables were indicated as the mathematical functions and input variables that would be used.

**Execution of GP Algorithm:** The process of GP was started, and the software was given permission to create, maintain, and improve models in each generation.

**Model Selection and Validation:** The best models arrived at through the fitness metrics—RMSE and NSE—were the ones selected to carry out the remaining analysis.

The whole training and optimization process of the model is highly essential in constructing predictive models that are highly accurate and dependable. In the case of rain prediction, the innate non-linearity and intricacy of the atmospheric data require the use of methods for robust modeling. One of the techniques, Genetic Programming (GP), is an evolutionary algorithm-based approach that provides a natural means for modelers of such systems. This work has adopted GPdotNET as its engine, which is a GP, Linux-based framework that incorporates symbolic regression principles for daily precipitation forecasting.

## 4. FINDINGS

This part discloses the results of the genetic programming (GP) models used for the short-term precipitation forecasting. The original meteorological features (e.g., maximum temperature, minimum temperature, humidity, wind speed, sea level pressure, dew point, solar radiation, and precipitation) were included without any preprocessing or signal decomposition. Unprocessed features were first used in GPdotNET to evaluate the baseline performance. After that, to get the influence of past information on the weather conditions of a specific day, lagged versions of the input features were created by introducing time delays (e.g., Tmax-2, Tmax-1, Tmax). The second model was then developed using these delayed features to investigate the role of earlier data in the forecasting process. To further improve the forecasting ability, the wavelet decomposition has been used to address the challenges posed by time lags in weather data. Each input's four quarters—three details and one approximation—were prepared by a one-level wavelet transform (DWT). We then built a new GP model using the wavelet-transformed data. Other than that, the relationships among the features were found through the correlation analysis. This was done with and without wavelet decomposition. Finally, another experiment was conducted to evaluate whether the model trained with only the original (non-lagged) features was still effective after lagged features were removed, and, thus, this allowed an in-depth survey of different feature engineering strategies. By using performance metrics such as root mean squared error (RMSE) and Nash-Sutcliffe efficiency (NSE) the results of all modeling experiments are then analyzed and compared, giving a full exposition of the models' strengths and weaknesses.

### 4.1. GPdotnet Setup

The original meteorological variables were used as direct inputs for the genetic programming (GP) model without any preprocessing or transformation at the first step of the modeling process. These variables were maximum temperature, minimum temperature, average temperature, humidity, wind speed, sea level pressure, dew point, solar radiation, and precipitation. The principal aim of the initial modeling phase was to find the performance baseline of precipitation forecasting using the raw data.

**Table 4.1.** Genetic Programming Parameters Used in GPdotNET

Parameters	Value/Setting
Population Size	500
Number of Generations	50
Selection Method	Rank selection
Crossover Rate	0.9
Rank value	0.8

**Table 4.1.** continue

Mutation Rate	0.2
Reproduction Rate	0.2
Max InitialTree Depth	5
Max Operation Depth	6
Brood Size	2
Random Constants Range	From 0 to 1, Count:5
Function Set	+, -, ×, ÷, sin, cos, log, sqrt,x <sup>2</sup> ,x <sup>3</sup>
Terminal Set	Input variables (Tmax, Tmin, etc.)
Fitness Function	RMSE (Root Mean Squared Error)
Procossor Setting	Multi Core

The above settings were applied uniformly across all models generated using the genetic programming (GP) algorithm in this study. These parameters were selected after initial experiments to create a balance between exploration and convergence. The use of a relatively high population size (500) along with the ranked selection method and the elitism mechanism resulted in the stability of the evolutionary process. Additionally, by setting a relatively limited depth for the GP trees, the model's complexity was controlled to prevent overfitting.

#### 4.2. Model 1 GP Result

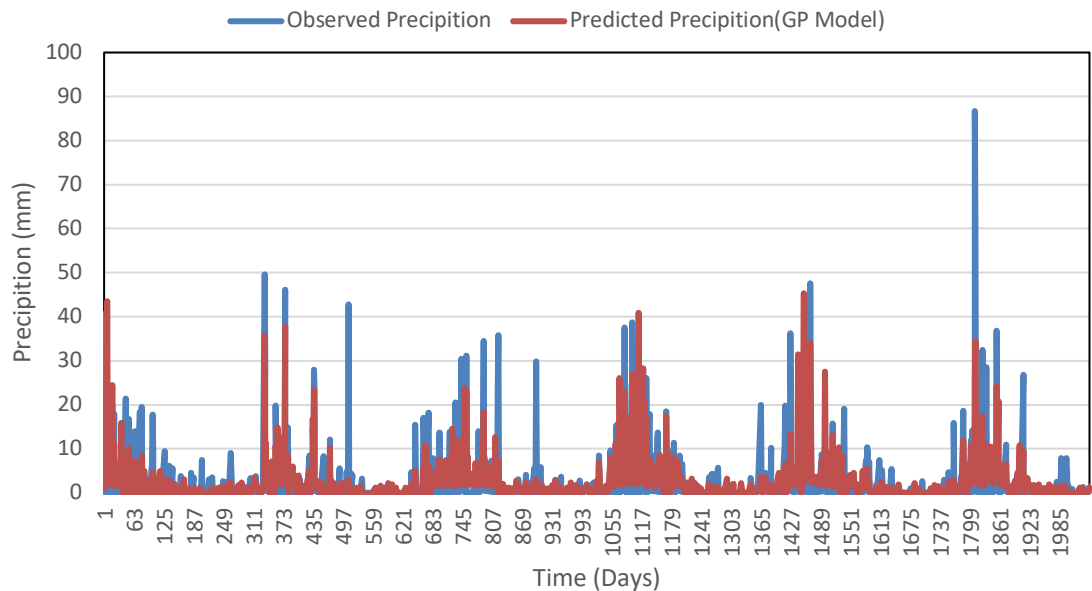
The data set that covered the period from 2016 to 2023 for the Antalya region was divided into training and testing sets: 70% of the data were assigned to the training set, whereas the remaining 30% comprised the testing set. The data split provided the model with the ability to learn from the historical data and then validate its performance on the new data. The GP model was performed via GPdotNET, with the aim of minimizing the Root Mean Squared Error (RMSE) in the course of training. Once the best-performing model had been determined through several iterations and optimization processes, it was selected based on its performance in the testing set.

Evaluation metrics derived from the model are provided in Table 4.2.1 In the case of the training set, RMSE and Nash-Sutcliffe Efficiency (NSE) were found to be 3.639 and 0.567, respectively. When it comes to the testing set, the RMSE value rose a little but remained within 3.717, while the NSE value was 0.667. These findings demonstrate the GP model's ability to catch some hidden rainfall patterns without using other features as training inputs. At the same time, the results of these performance metrics also point to the fact that the model has the potential to be improved, namely, its precision in making accurate predictions and its capability of capturing the inherent dynamics of the precipitation data.

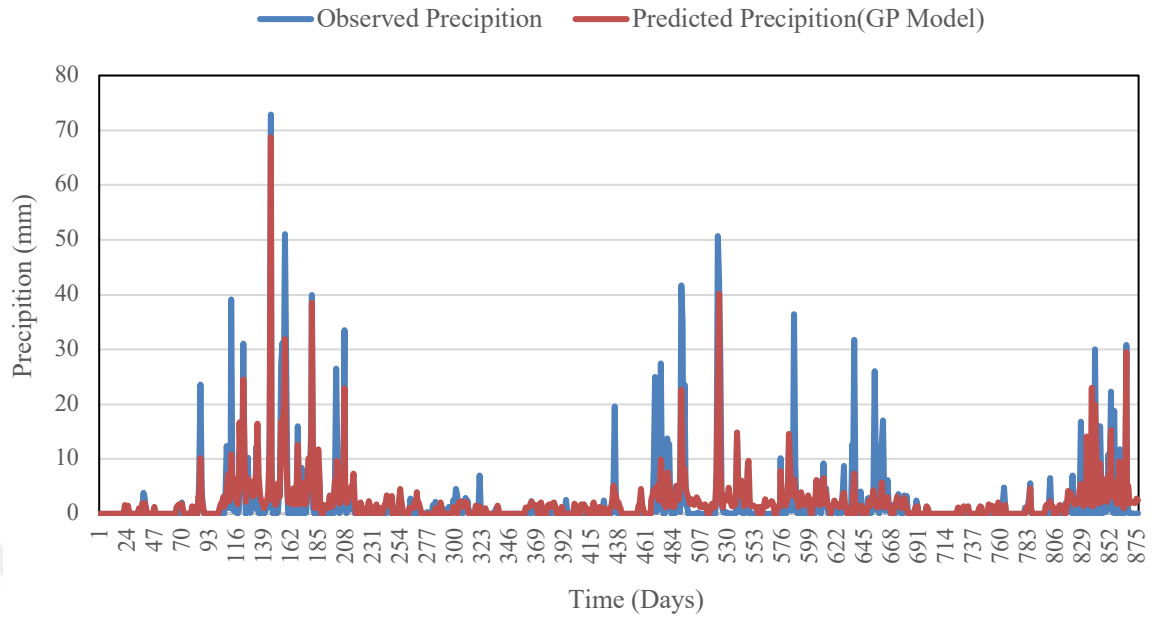
**Table 4.2.1.** The Evaluation of Raw Data Metrics

Metrics	Training	Testing
RMSE	3.639	3.717
NSE	0.567	0.667

The figure 4.2.1 shows the observed and predicted precipitation values coming from the training set. A good match of the values demonstrates how brilliantly the GP model was able to learn the rainfall periods from the training phase. However, occasional discrepancies indicate that the model can be further trained. Thus, this figure is a significant source to help in assessing the model's initial learning performance.

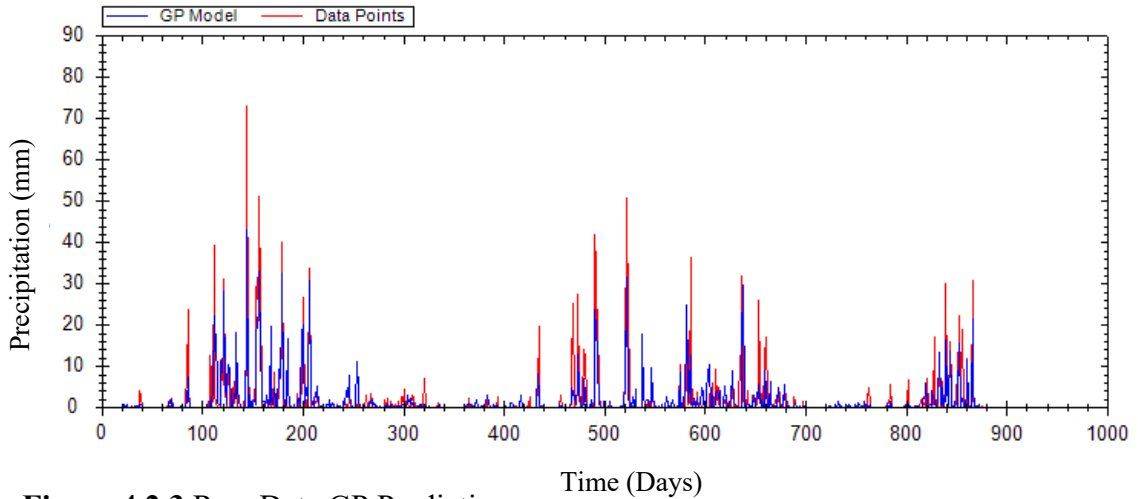
**Figure 4.2.1.** Training of Raw Data GP Model

The figure 4.2.2 visualizes the situation in which the model was tested on 30% of the total data. The figure indicates that the GP model has done a good job in terms of learning from the initial data to make good predictions on the future, unseen data. The predicted values match the observed rainfall patterns, for the most part, but certain differences, the most noticeable during heavy rainfall intervals, show that there is a necessity for more sophisticated modeling methods.



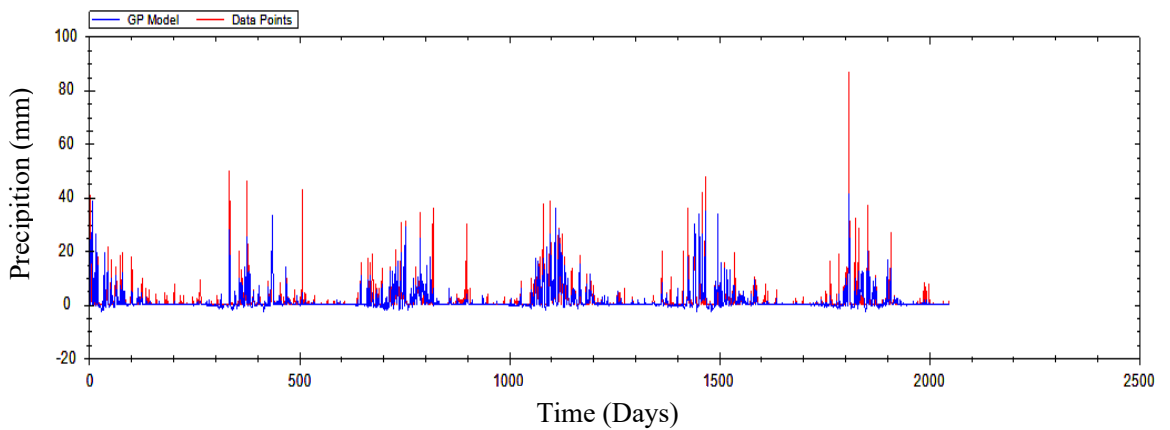
**Figure 4.2.2.** Testing Raw Data GP Model

Figure 4.2.3. The prediction figure shows the comparison of observed and predicted precipitation data (test period) for evaluating the GP model.



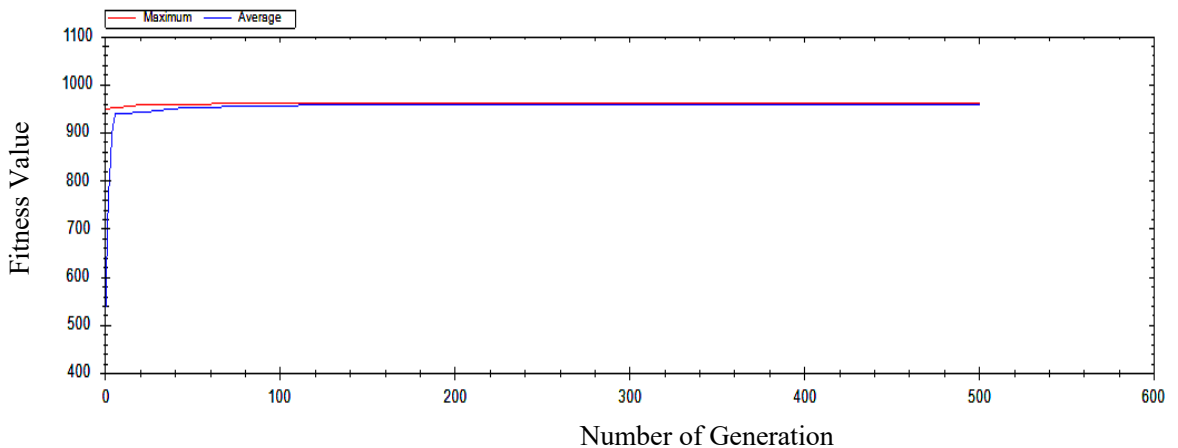
**Figure 4.2.3** Raw Data GP Prediction

The Figure 4.4 presents the results of the GP model performance on the whole dataset that includes both the training and testing parts. This visual is not only a good indicator of the stability of the model but also lets one have a holistic view of the model's performance over time. The wildly varying experiments of observed and synthetic data show exactly how hard it is to keep the accuracy of the forecast in different weather conditions, which gives this figure a crucial role in the model robustness evaluation.



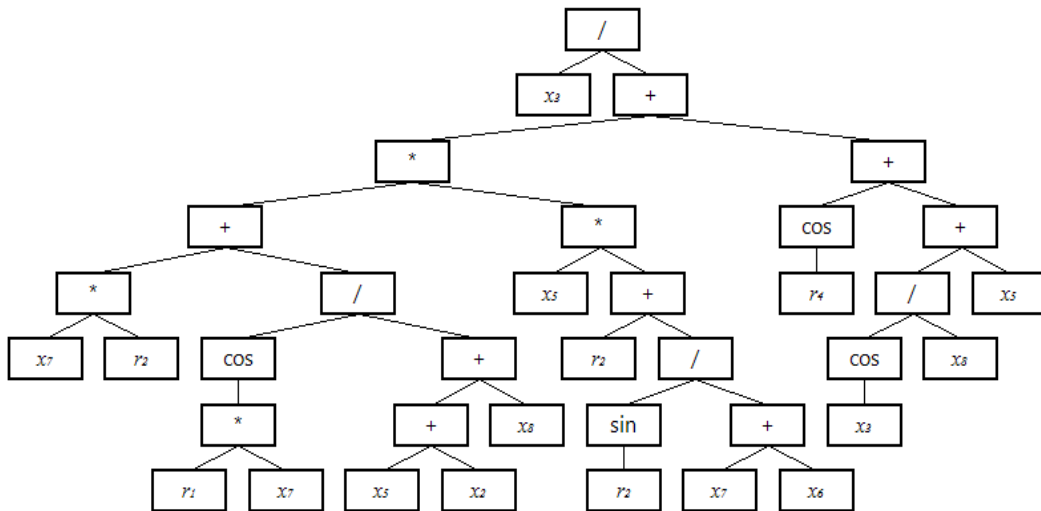
**Figure 4.2.4.** Raw Data GP Model Simulation

Figure 4.2.5 is meant to reflect the GP algorithm’s fitness over the training process. The almost consistent downward path and the ultimate stabilization of fitness value are affirmations of the model’s efficiency in parameter optimization through the generations. A line that converges smoothly implies the algorithm has found a feasible solution, whereas irregularities might mean some more parameter tuning or means of additional iterations are required. The topic of the graph is indispensable before someone could be able to get an impression of the optimization and GP approach reliability.



**Figure 4.2.5.** RawDataGPFitnessSimulation

Overall, this experiment established a baseline for later feature engineering operations, including the inclusion of lagged variables and wavelet decomposition, which are covered in the following sections, and gave insightful first approximations of the performance of the GP model.



**Figure 4.2.6.** Tree Structure of Model-1

Figure 4.2.6 shows the tree-structure that was created for Model -1. This model forecasts the target variable (e.g.,precipitation) with an intricate mathematical formula consisting of a group of input variables and a combination of math operators like addition, multiplication, and division; trigonometric functions (cosine, sine); and inner workings. Such various operations exemplify the model's ability to present nonlinear associations. The mathematical formula of the tree structure of model-1 is as follow :

$$\text{predicted Output} = \frac{x_3}{\left[ (x_7 \times r_2) + \frac{\cos(r_1 \times x_7)}{(x_5 + x_2) + x_8} \right] \times \left[ x_5 \times \left( r_2 + \frac{\sin(r_2)}{x_7 + x_6} \right) \right] + \left[ \cos(r_4) + \left( \frac{\cos(x_3)}{x_8} + x_5 \right) \right]} \quad (4.2.1)$$

As seen in the above formula, the model makes significant use of variables such as  $x_3$ ,  $x_7$ ,  $x_5$ ,  $x_2$ , and  $x_8$  which are meteorological input parameters employed for rainfall forecasting (including of maximum temperature, minimum, average temperature, humidity, wind speed, dew point, sea level pressure and solar radiation).

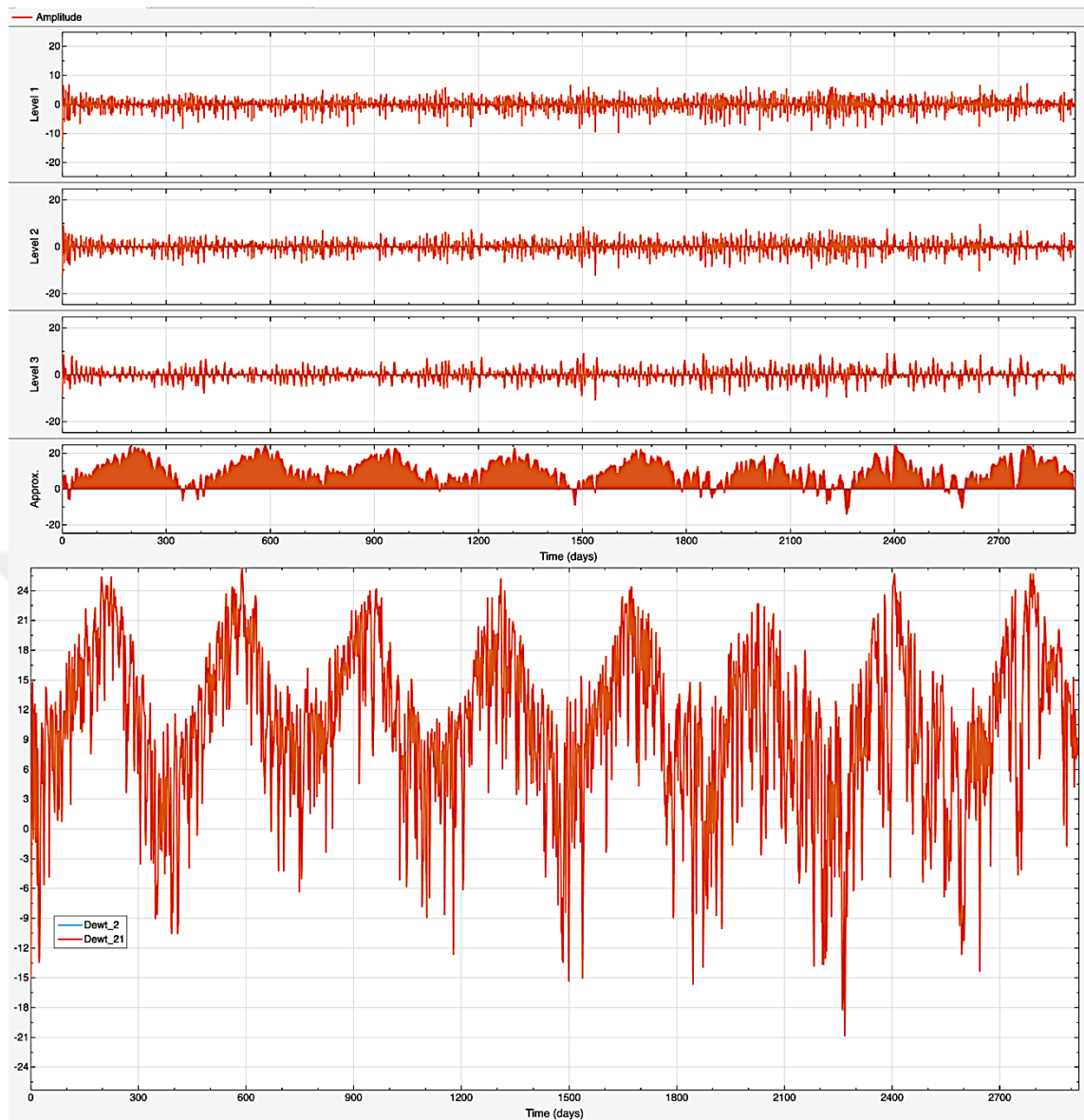
In addition to meteorological input variables, genetic programming (GP) models use values called random numbers, represented by the symbol  $r$  such as  $r_1$ ,  $r_2$  and  $r_4$ . These random numbers are numerical constants generated by the GP algorithm itself. In other words, these values are neither taken from the meteorological input data nor are they given to the model in advance. Rather, at the time of model construction, the GP algorithm randomly generates a few constants so that it can use them in the mathematical equations of the model. These random numbers, like other parts of the model, are optimized during the model training process. That is, in each generation, the GP algorithm examines what numerical values for these constants make the model perform best in forecasting. The main role of the random numbers  $r$  is to increase the flexibility of the model. The presence of these values allows the model to build more complex and accurate mathematical relationships and to better identify nonlinear patterns and sudden changes in meteorological data. These numbers are placed in various parts of the model formula, such as multiplication by variables or inside mathematical functions such as sine, or cosine, to allow the model to express more complex behaviors. The important point is

that the range of these random numbers is adjustable. In this way, the user can determine how many random numbers the algorithm will generate and in what range these numbers should fall.

### 4.3. Model 2 Hybrid WT-GP

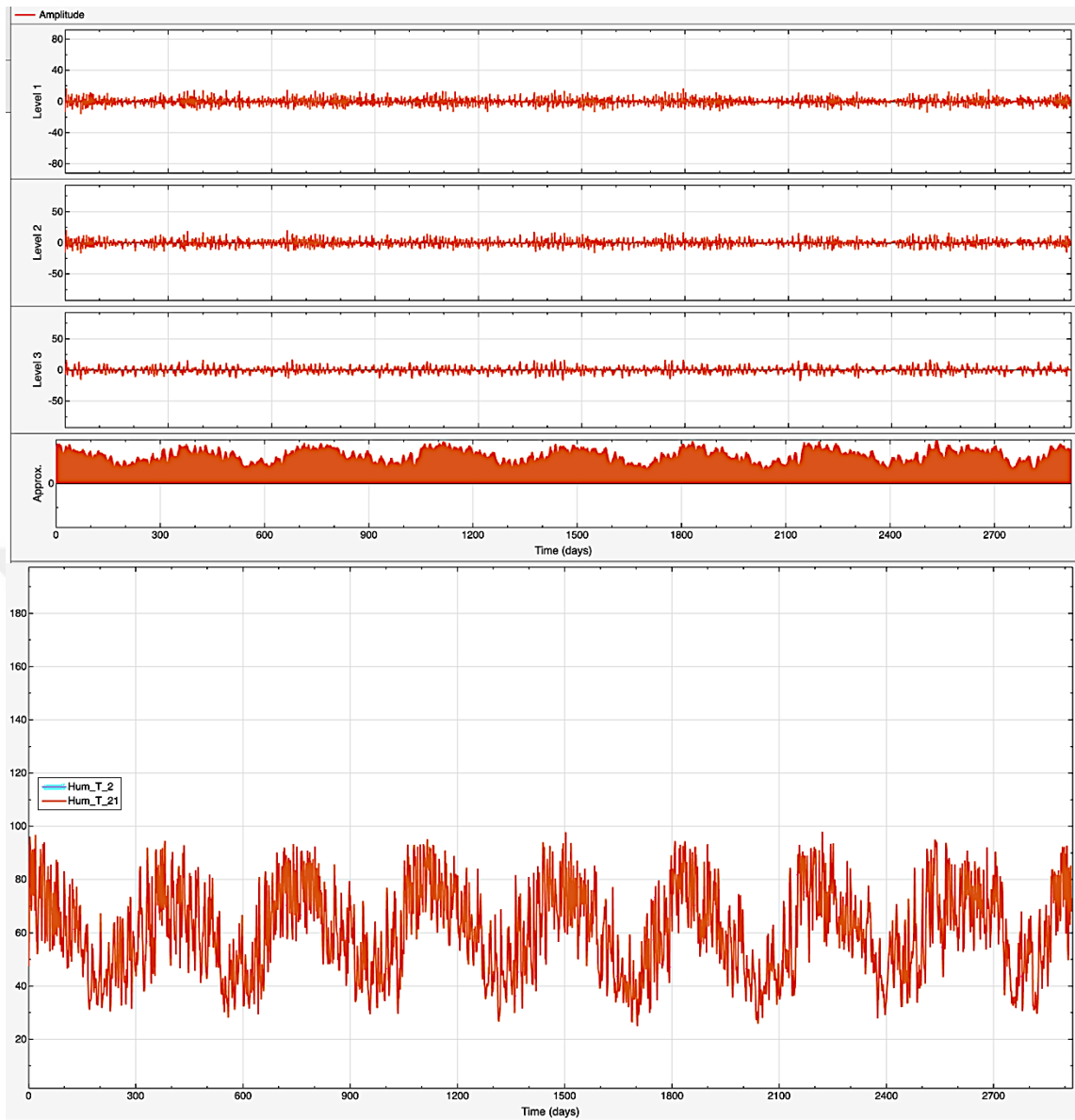
In the second step of the model-building process, the raw meteorological dataset was decomposed using wavelets, and subsequently, the decomposed data were put into the GP model. All the primary features, including sea level pressure, wind speed, maximum and minimum temperature, mean temperature, dew point, solar radiation, and relative humidity, were decomposed into three partial components (D1, D2, D3) and one approximate component (A3) using discrete wavelet transform (DWT) at one level. Although all meteorological variables were decomposed, only a few of them, such as dew point, relative humidity, maximum temperature, and wind speed, were selected as examples for display in the figures.

Figures 4.3.1 through 4.3.4 describe the results of wavelet decomposition of these variables, which were performed in the MATLAB environment. Each figure contains partial and approximate levels of the original signal. These decomposed signals were given as input to the GP model in the next steps to investigate whether the latent patterns in the data, extracted through frequency decomposition, can help improve precipitation forecasting. Figure 4.3.1 shows the wavelet decomposition of the dew point. This decomposition allows for the examination of short-term fluctuations in the water vapor condensation potential.



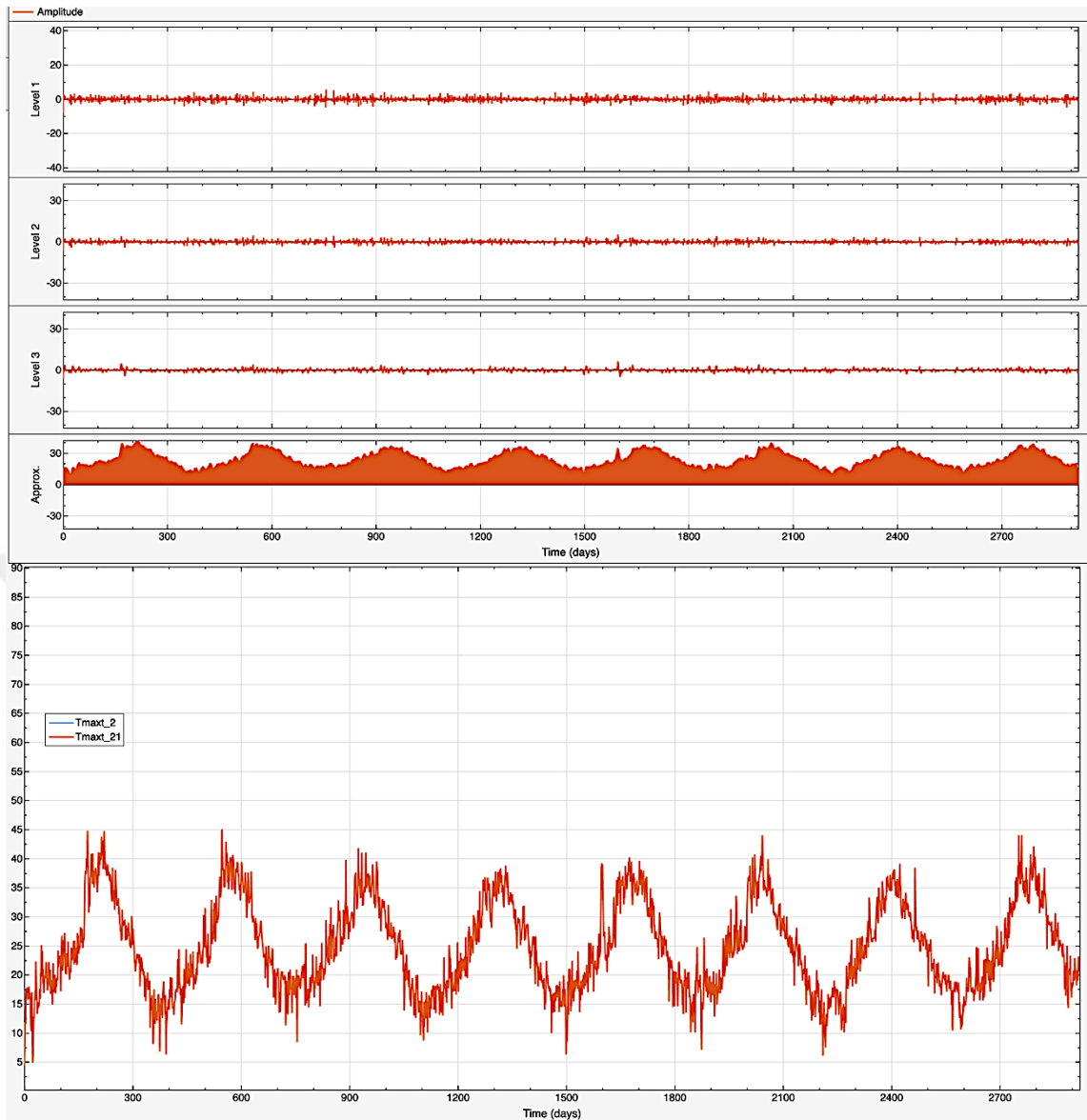
**Figure 4.3.1.** Dew point Decomposed

Figure 4.3.2 shows the decomposition of relative humidity, allowing for the detection of rapid fluctuations in humidity that can affect sudden precipitation events.



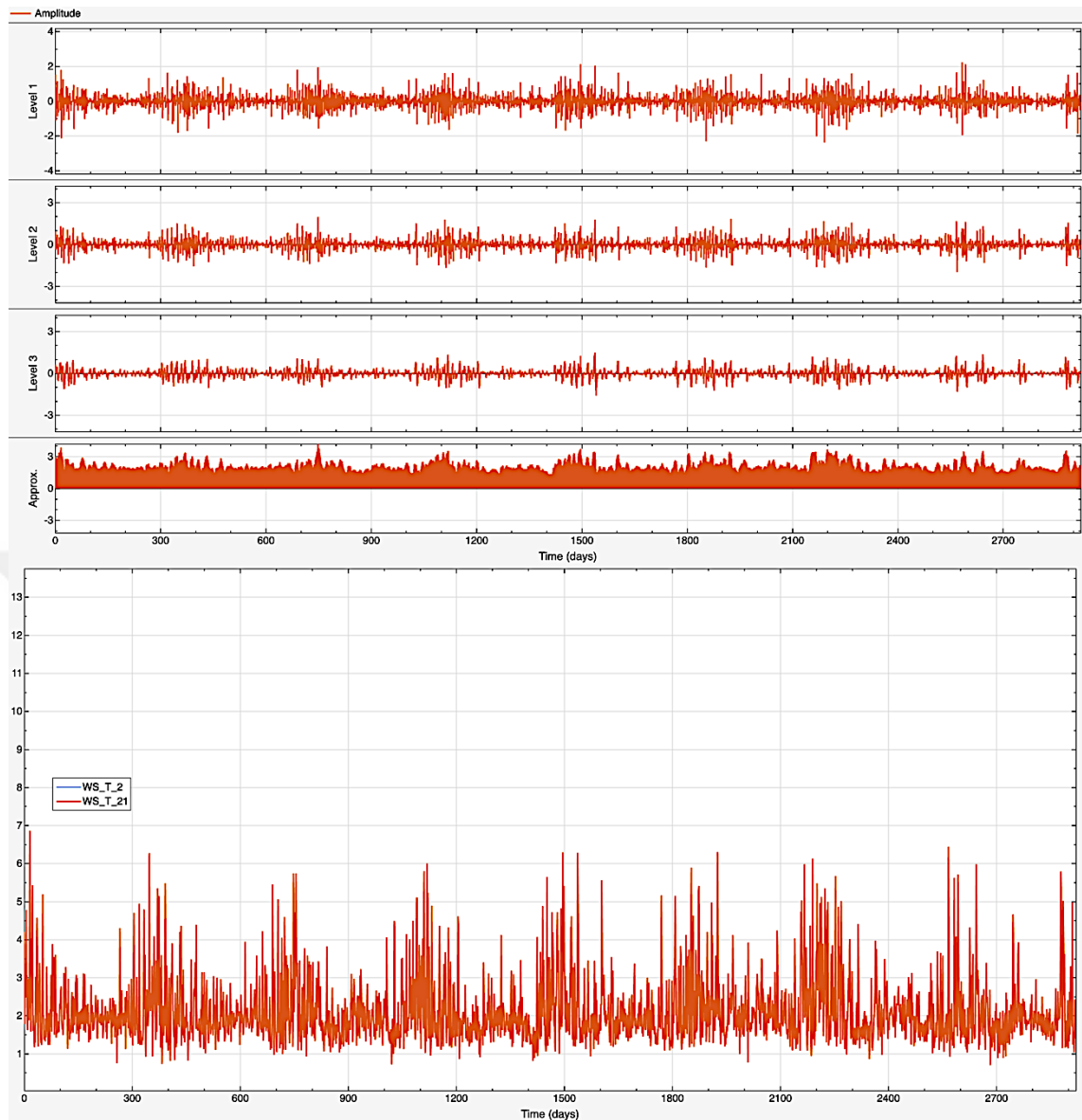
**Figure 4.3.2.** Humidity Decomposed

Figure 4.3.3 divides the maximum temperature ( $T_{max}$ ) into different frequency levels, revealing thermal patterns associated with precipitation events.



**Figure 4.3.3.** Tmax Decomposed

Figure 4.3.4 shows the decomposition of wind speed, allowing for the separation of instantaneous wind effects from long-term trends.



**Figure 4.3.4.** Wind Speed Decomposed

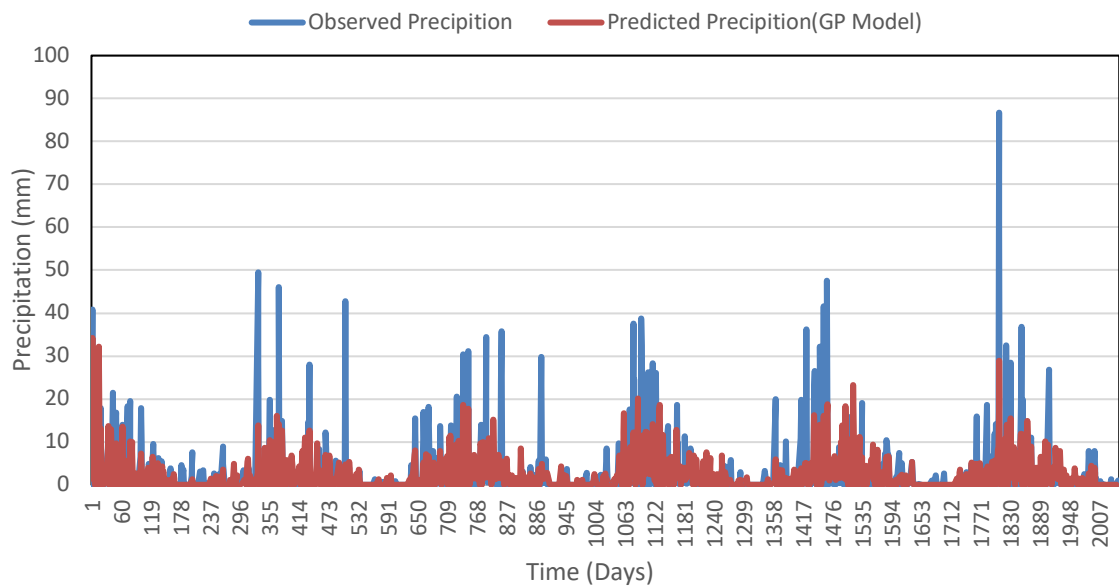
The main goal of this method was to capture different patterns in the data and see if we could gain extra insights from these breakdowns to improve the accuracy of daily precipitation predictions. As in the first experiment, the dataset was partitioned into a training set (70%) and a testing set (30%). The model was then tested against the evaluation metrics of Root Mean Square Error (RMSE) and Nash–Sutcliffe Efficiency (NSE) for performance.

**Table 4.3.2.** Results Of Wavelet-Decomposed Input Festures

Metrics	Training	Testing
RMSE	4.384	4.732
NSE	0.372	0.460

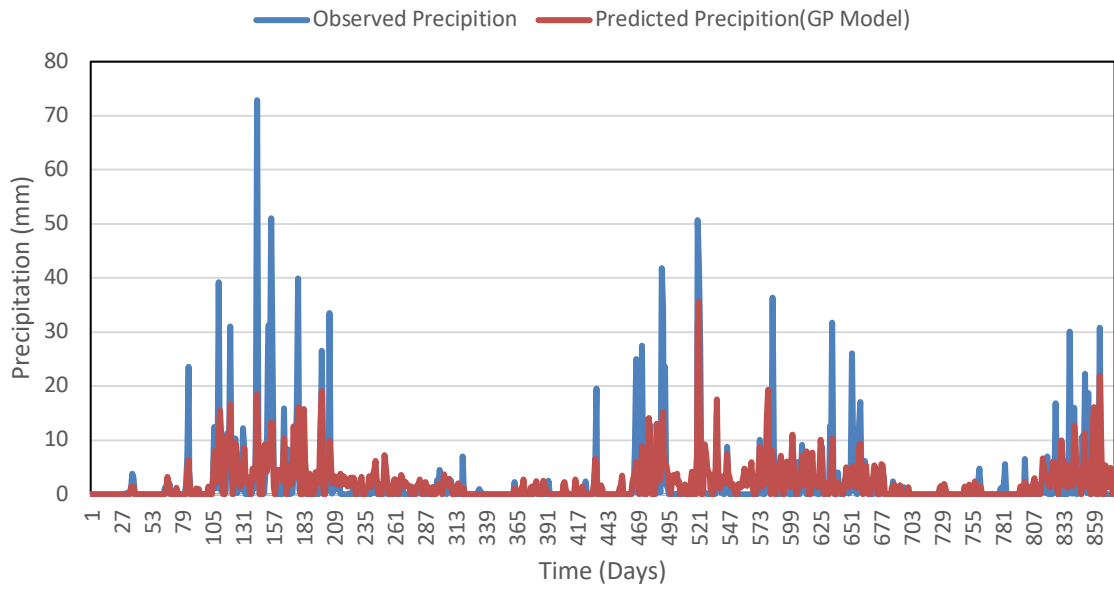
These results show the model’s predictive performance deteriorated significantly, as in the raw data model. In particular, for the training set, the RMSE increased to 4.384, and the NSE decreased to 0.372, indicating a poor predictive performance and model fitting. Likewise, for the testing set, the RMSE was 4.732, while the NSE increased marginally to 0.460. These results indicate that the extra complexity of wavelet-transformed data either significantly overloaded the GP model or added noise, which had bad implications for its performance.

The figure 4.3.1 provides an overview of the expected precipitation values compared with the predicted precipitation data in the training set following wavelet decomposition. The model has identified some underlying patterns. Nevertheless, its performance is not as good as the model trained using raw features, which emphasizes the complexity of employing the wavelet-transformed inputs without the correct model settings.



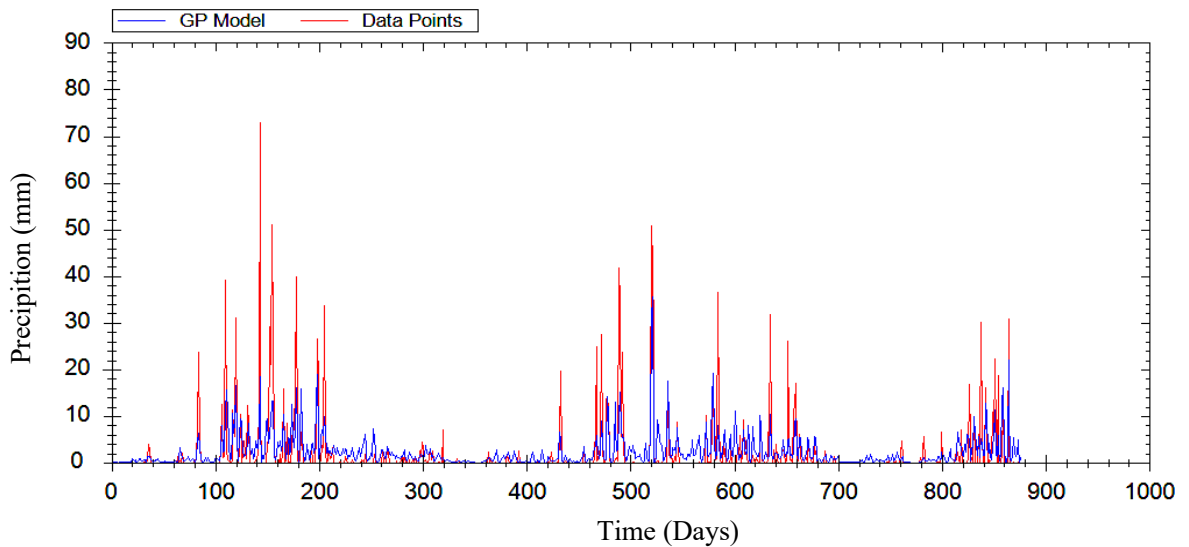
**Figure 4.3.5.** Wavelet Decomposition Training Data

The figure Figure 4.3.2 shows how well its predictions match with the testing data samples. The errors of predicted rainfall values and the observed ones are larger, which implies that the model has only partially generalized when using wavelet-based features.



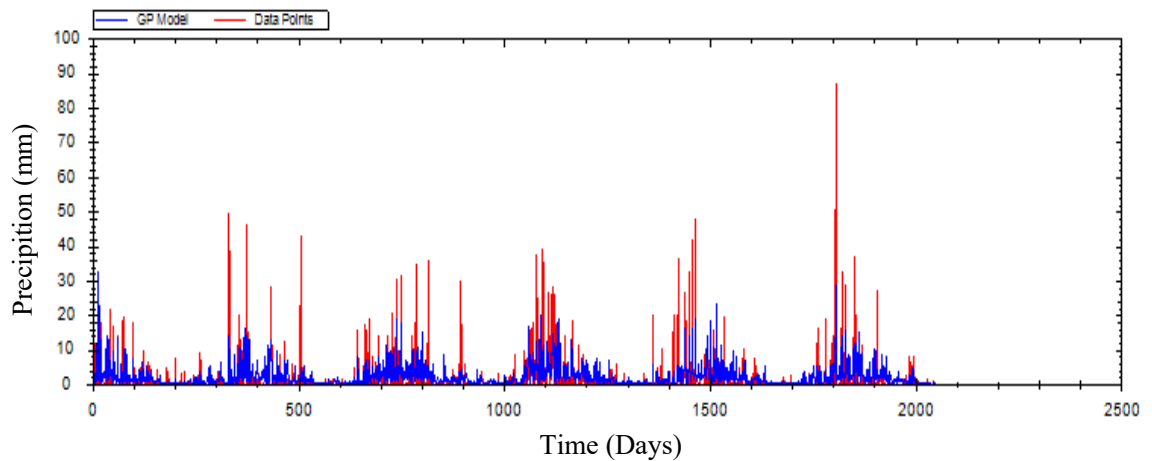
**Figure 4.3.6.** Wavelet Decomposition Testing Data

The values of the precipitation observed and the precipitation predicted during the test period are presented in the figure 4.3.3. The lower predictive power is a result of the increase in the complexity of the approach from processing by wavelet conversion and at the same time points out the necessity for the careful selection of the functionality and its fine regulation.



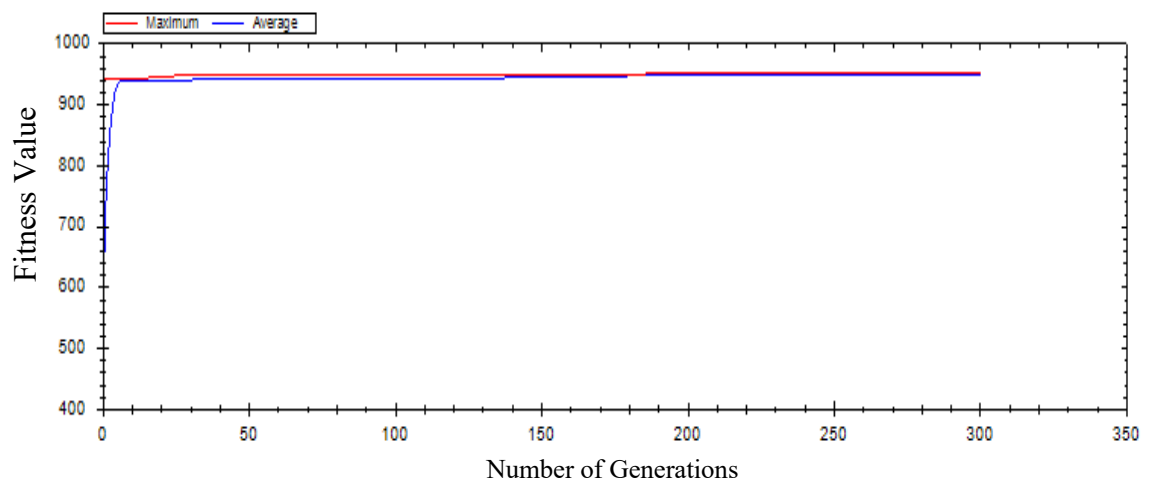
**Figure 4.3.7.** Wavelet Decomposition Prediction Data

The figure 4.3.4 is based on the simulation over the whole set of data, as well as the combined training and testing data. The diagram brings out the overall performance patterns and shows clearly where the model did not perform, in particular cases of heavy rainfall.



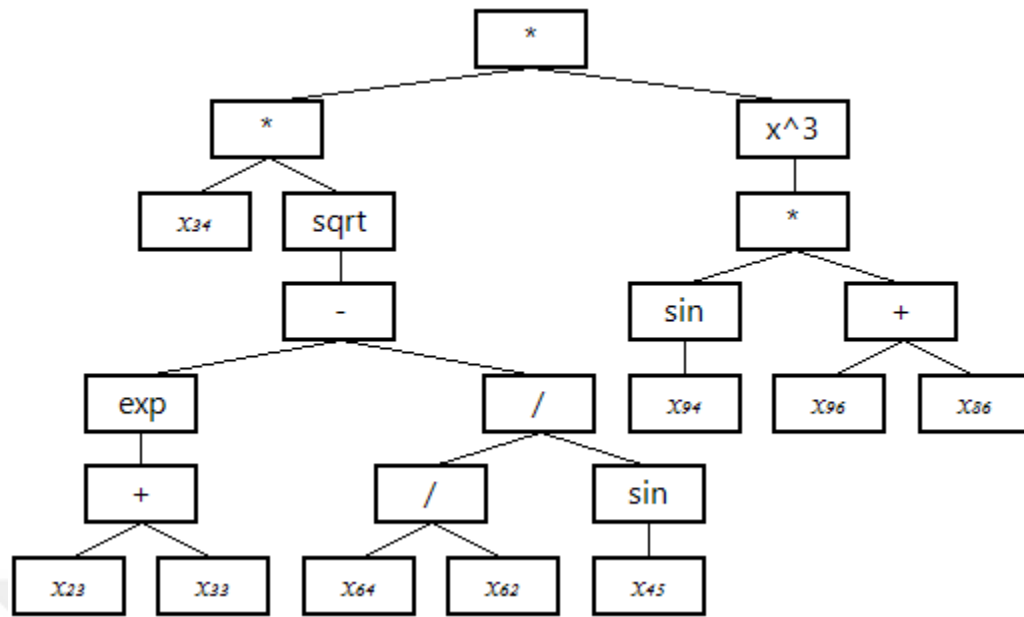
**Figure 4.3.8.** Wavelet Decomposition of GP Model Simulation

This diagram exemplifies the GP algorithm's convergence behavior when applied to wavelet-transformed data. Opposed to the original feature model, the fitness function plot represents a less smooth and more noisy convergence, which, in turn, makes the optimization process also weaker, and difficulties in finding the best solution exist under this configuration.



**Figure 4.3.9.** Wavelet Decomposition of GP Fitness Simulation

The data suggest that the application of wavelet-decomposed data resulted in a model with lower predictive power in contrast to the raw data. The increase in the error and the decrease in NSE depict a negative effect of the added complexity on GP model performance. Thus, going back to feature selection and model tuning is a must for performance improvement.



**Figure 4.3.10.** Tree Structure of Model-2

The tree structure for Model-2, the model that uses very complicated mathematical operations like exponential, square roots, sine, and power functions to predict precipitation. At the same time, the model considers a wide range of variables and makes use of both nonlinear relations and nested calculations to express these relations appropriately. The mathematical formula of this tree structure below show :

$$\text{Predicted Precipitation} = x_{34} \times \sqrt{\exp(x_{23} + x_{33}) - \left(\frac{x_{64}}{\sin(x_{45})}\right) \times [\sin(x_{94}) \times (x_{96} + x_{86})]}^3 \tag{4.3.1}$$

In this model, meteorological variables such as humidity, temperature, wind speed, and other data are first processed by the Wavelet Transform method. In this method, each variable is divided into three detail coefficients and an approximation coefficient to represent different frequency characteristics of the data. For example, humidity variable is split into four levels, D1, D2, D3, and A3, and this is done similarly for all other meteorological variables. This formula show us that the variables denoted by x (for instance, x23, x33, or x94) are the Wavelet decomposition components of the original data. In this formula, besides other things, the exponential and sine functions are present to signify the ability of the model to follow the nonlinear and periodic patterns in meteorological data.

**4.4. Model 3 GP with Lagged Data Features**

To elevate the ability of the model in grasping the time dependencies that occurred in the precipitation behavior, a new experimental setup was built in which the meteorological variables of the past time were brought into the present. The initial features, including Tmax, Tmin, Tave, humidity, wind, dew point, sea level pressure, and solar radiation, were still there, but names with days were also added as input variables

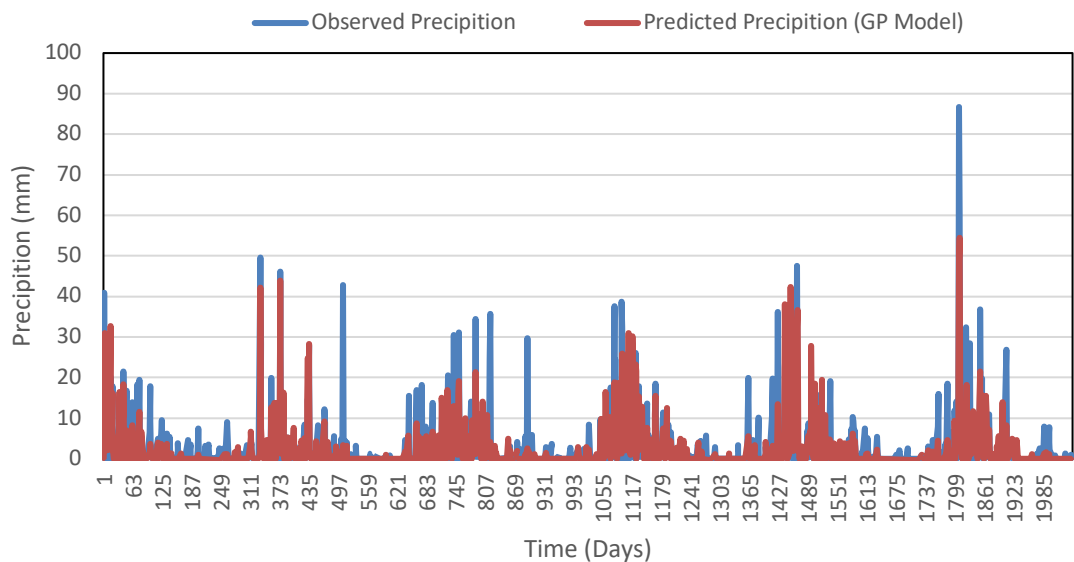
(e.g.,  $T_{max}(t)$ ,  $T_{max}(t-1)$ ,  $T_{max}(t-2)$ ). After it was split, the data was again divided into 70% and 30% for training and testing, respectively, and the Genetic Programming (GP) model was fed by the lag-enhanced feature set. The evaluation was made by utilizing the same measures as in the previous phases, which were Root Mean Square Error (RMSE) and Nash–Sutcliffe Efficiency (NSE).

The new findings can be seen as a substantial increase in the accuracy of the prediction compared with the previous ones. The model on the training dataset gave an RMSE of 3.588 and an NSE of 0.579, whereas the testing dataset turned out to have a higher RMSE, 3.645, and a lower NSE, 0.680, suggesting the model's generalization ability weakened and the matching of precipitation between the predicted ones and observed ones was less good.

**Table 4.4.1.** Matrices of GP Model with Lagged Data Features

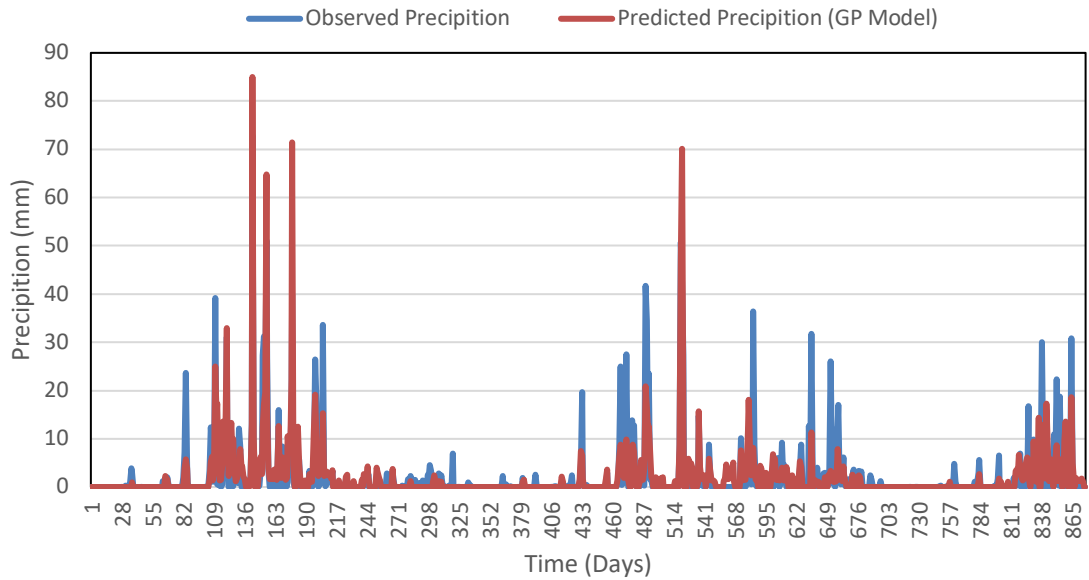
Metrics	Training	Testing
RMSE	3.588	3.645
NSE	0.579	0.680

The figure 4.4.1 demonstrates the resemblance between the actual and the simulated rainfall values in the training dataset with the help of lagged data features. From the figure, it can be seen that the integration of past weather data gave a lot of improvement in the model’s capability to detect time-related relationships; thus, the model was able to forecast more accurately.



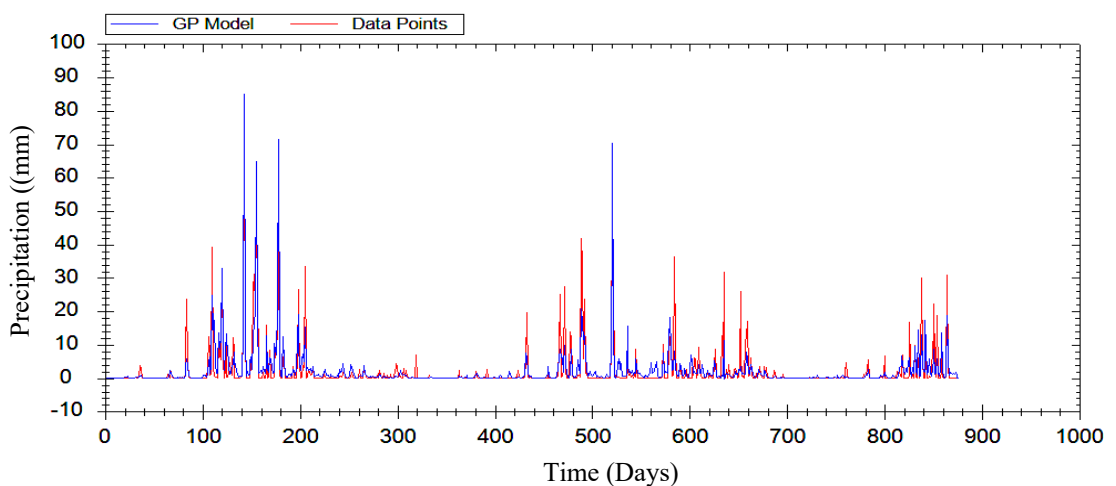
**Figure 4.4.1.** Lagged of Training Data

The figure 4.4.2 illustrates the way the model behaved in the testing dataset after the lagged features were added. As can be seen, the figure confirms that the model is able to predict well the unseen data. However, there are a few times when the forecast is not accurate, especially during high-intensity rainfall events. This not only points to the potential of using past inputs but also to the difficulties in the prediction of rare events.



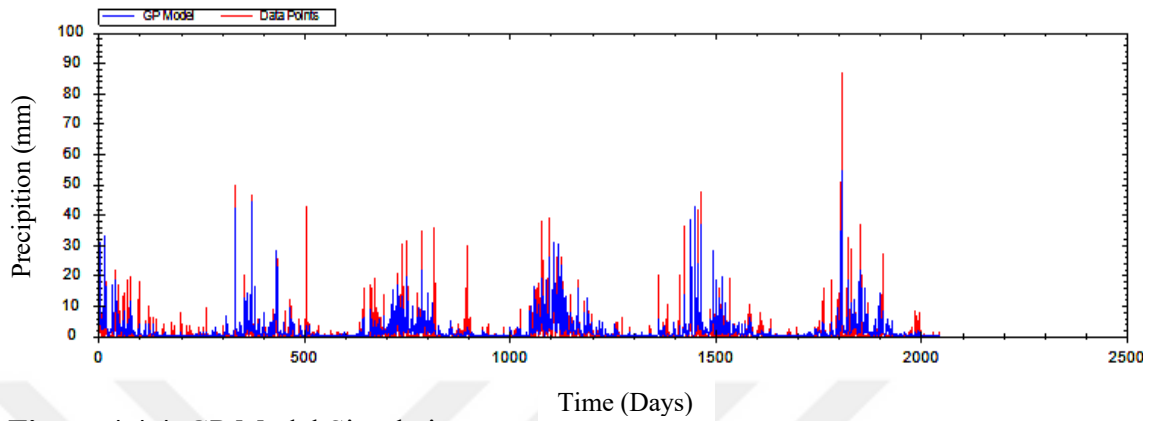
**Figure 4.4.2.** Lagged of Testing Data

The figure 4.4.3 measures the actual and anticipated rainfall quantities of the test set by showing the deviation of the test set from the model. The good accuracy of the graph is just one of the strong points. Indeed, it is much more accurate than the models that did not take into account the effect of the past. The small error indicates that the time reference in rainfall forecasting is of great importance.



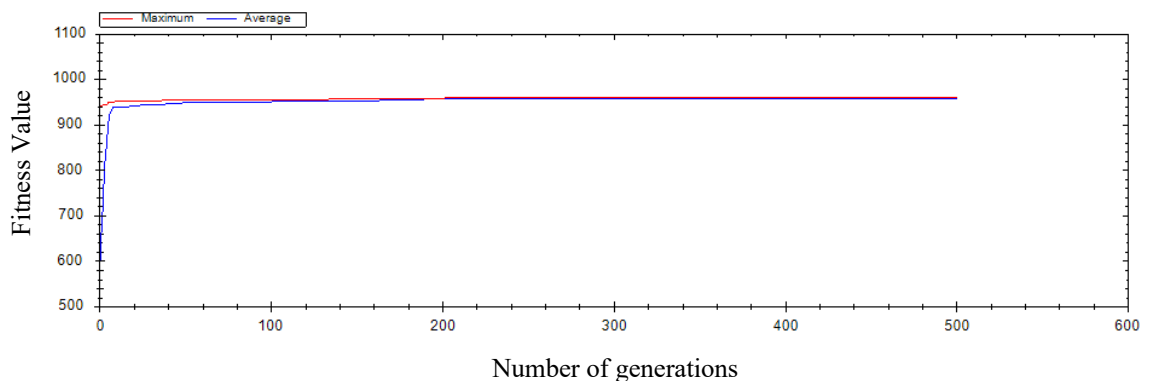
**Figure 4.4.3.** Lagged of data Prediction

The figure 4.4.4 depicts the GP model simulation outcomes on the entire dataset after the training and test phases were combined. The graph shows that incorporating lagged inputs improved model consistency and enhanced the detection of complex rainfall patterns across the dataset.



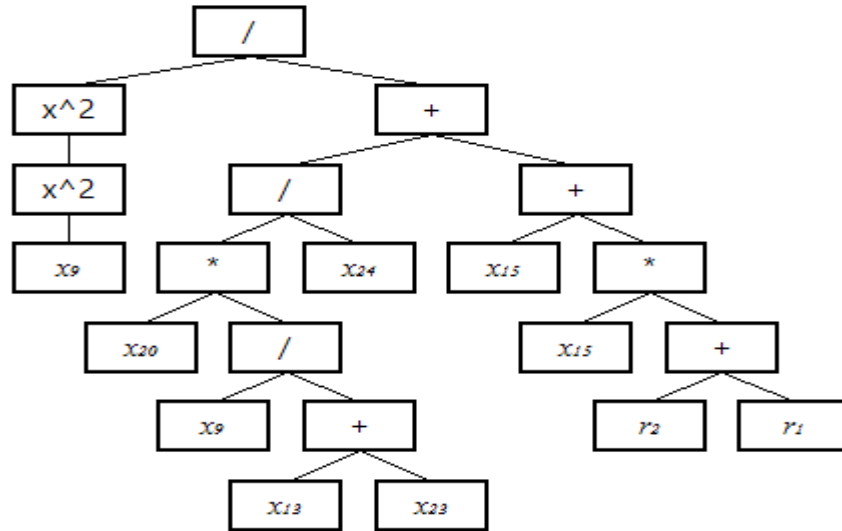
**Figure 4.4.4.** GP Model Simulation

The figure 4.4.5 shows the fitness curve of the GP algorithm during training with lagged features.



**Figure 4.4.5.** GP Fitness Simulation

Lagged meteorological data supplements enhanced the ability of the GP model to capture temporal dependencies in rainfall forecasting. Although predicting performance was enhanced on the calibration set, application on the validation set resulted in small deviations, especially in periods of heavy rainfall. In general, lagged inputs helped in making the model better, giving the respective feature more importance in time series forecasting.



**Figure 4.4.6.** Tree Structure of Model3

Figure 4.4.6 shows the tree structure of model three. In this model, the input variables demonstrated by x, such as x<sub>9</sub>, x<sub>20</sub>, x<sub>13</sub>, x<sub>23</sub>, and x<sub>24</sub>, represent lagged data about meteorological parameters used to predict precipitation. This means that these variables refer to the measured values of meteorological variables at past times and help the model identify time dependencies and lagged effects of atmospheric conditions on future precipitation. The variables r<sub>1</sub>, r<sub>2</sub>, are also random constants in genetic programming that are generated by the GP algorithm and do not already exist in the data. These numbers help the model build more complex and accurate relationships. The value of these numbers is automatically optimized to give the best prediction results. The range of values and the number of these numbers can also be specified in the GP settings table.

The mathematical formula of the model is as follows:

$$predicted\ Precipitation = \frac{(x_9^2)^2}{\left(\frac{x_{20} \times \left(\frac{x_9}{x_{13} + x_{23}}\right)}{x_{24}}\right) + [x_{15} + (x_{15} \times (r_2 + r_1))]} \tag{4.4.1}$$

As can be seen in the formula, the variable x<sub>9</sub> plays a very important role in the model's prediction, as it has been subjected to multiple power operations, indicating its significant impact on the output. The denominator of this fraction includes several variables, such as x<sub>20</sub>, x<sub>24</sub>, and the sum of r<sub>2</sub> and r<sub>1</sub>, indicating the complex relationships between these factors. The use of division and power operators in this model demonstrates its ability to represent linear and nonlinear patterns in precipitation data.

**4.5. Model 4 Correlation Analysis of Lagged Data not Decomposed Data**

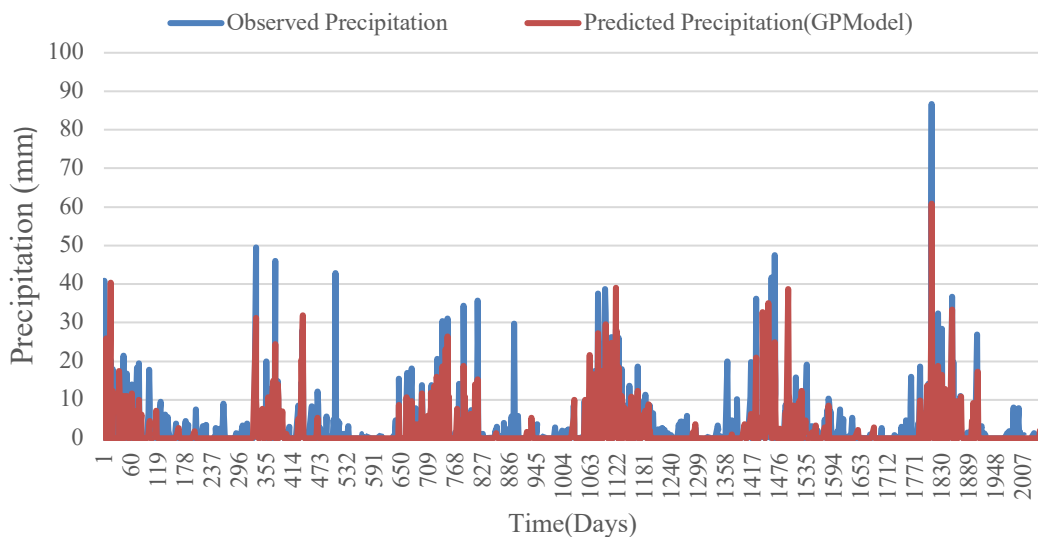
The next modeling step involved reducing model complexity and improving efficiency by selecting statistically relevant lagged variables based on correlation analysis. From the big array of lagged explanatory variables—that comprised, among other things, multiple lagged versions of the Tmax, Tmin, humidity, wind speed, etc.—the correlation matrix was calculated to find the variables that had good statistical bonds with the desired output (precipitation).

Only the most highly correlated lagged variables were used to train the GP model in order to eliminate redundant and poorly informative variables, prevent overfitting, and improve model generalization. The model was trained with 70% of the data and tested with the other 30%. Testing was also done with RMSE and NSE statistics. On the training set, the model produced RMSE = 3.614 and NSE = 0.573. On the test set, the RMSE was 3.695, and the NSE was 0.671. These results show that with fewer input variables, the model produced decently good performance and, in some instances, was comparable, if not superior, to models with the complete lagged set of data.

**Table 4.5.1.** Matrices of Correlation Analysis of Lagged Data

Metrics	Training	Testing
RMSE	3.614	3.695
NSE	0.573	0.671

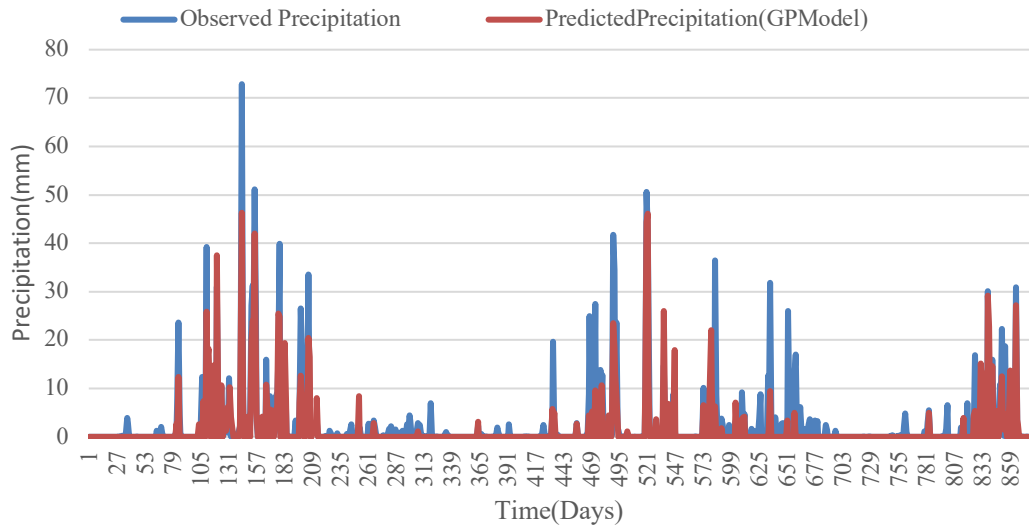
The figure 4.5.1 shows the comparison between the observed and predicted precipitation values in the training dataset after applying correlation-based feature selection. This figure shows that reducing the input dimensionality helped the model maintain good prediction performance during training.



**Figure 4.5.1.** Training Data

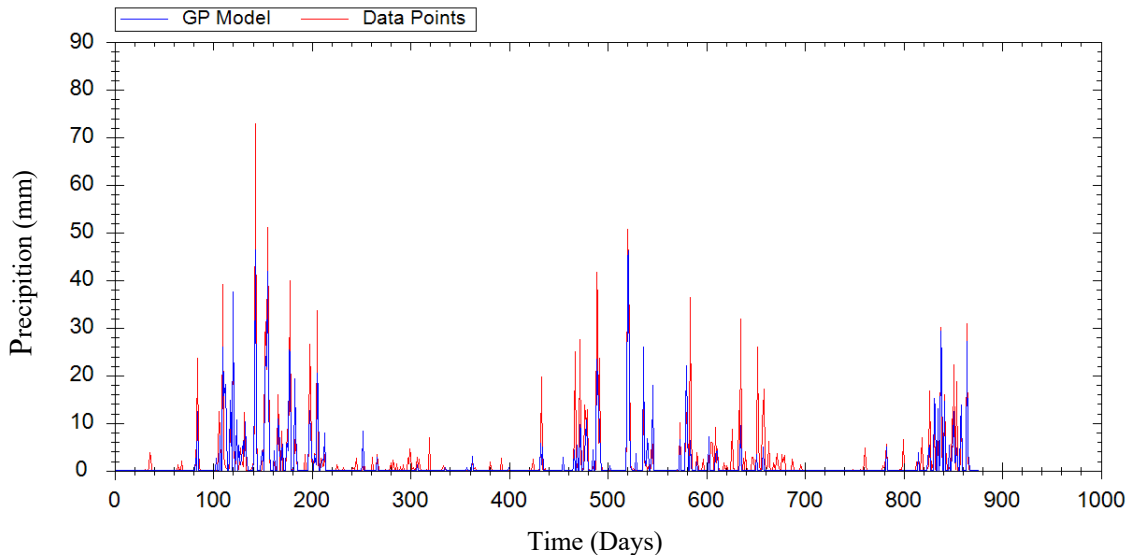
In the figure 4.5.2, the results of the model on the testing dataset are shown. The predictions match the real ob In this figure, the model results are shown on the test data set. The predictions are in good agreement with the actual observations, indicating that the model adapts successfully, even when trained using a smaller selection of lagged

variables. servations quite closely, indicating that the model successfully adapts, even when it has been trained using a smaller selection of lagged variables.



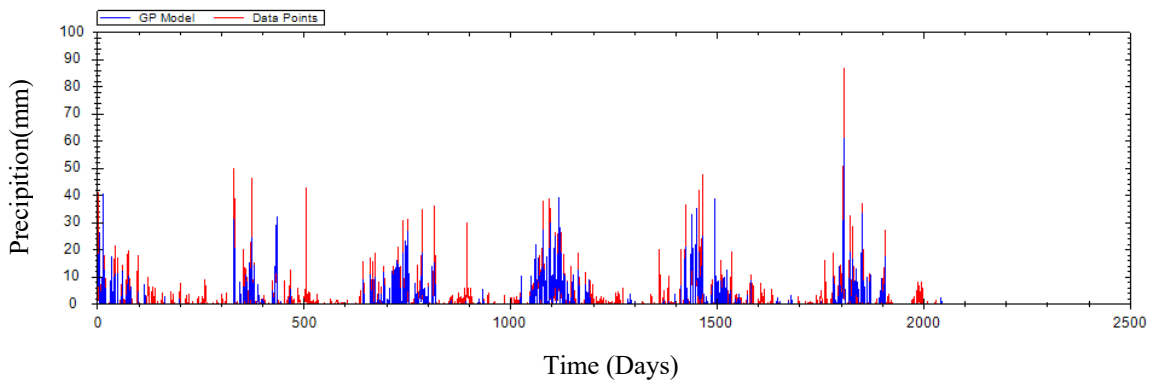
**Figure 4.5.2.** Testing Data

The figure 4.5.3 displays the contrast between the measured and forecasted precipitation, of which the test set was the subject. The corresponding reduced input model not only outperforms that of the reference but also becomes proof that the correlation-based feature selection method, which sacrifices accuracy in favor of speed, may still be used with success.



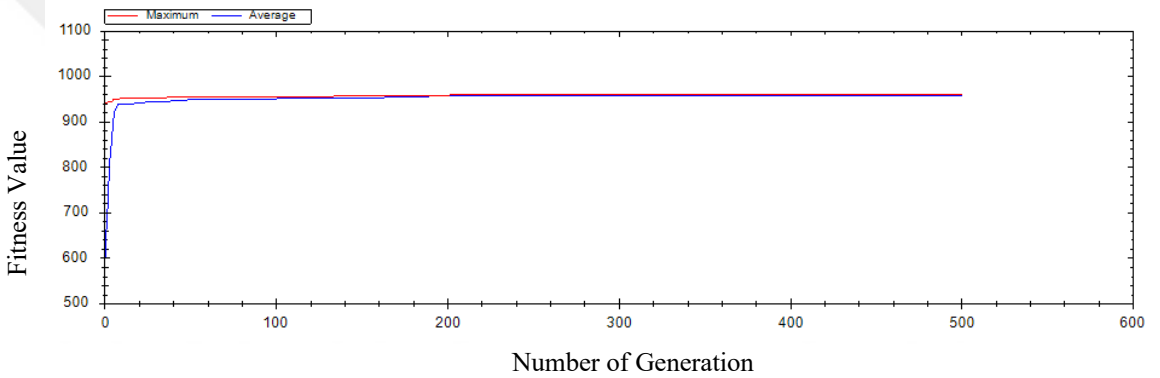
**Figure 4.5.3** Prediction of GP

This figure displays the full simulation results over the entire dataset, showing the model’s consistency and its ability to track rainfall trends after feature reduction.



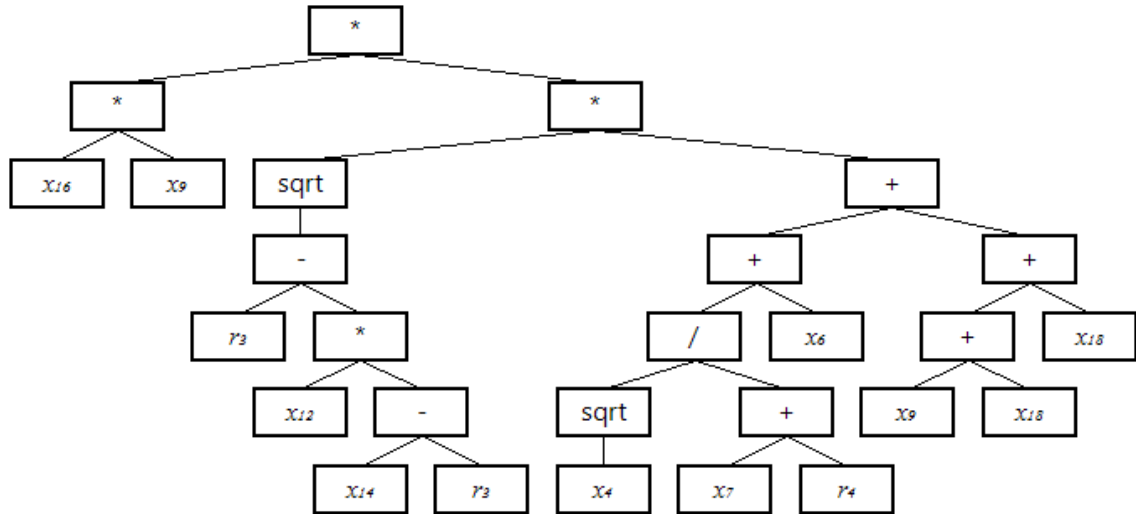
**Figure 4.5.4** GP Model Simulation

The figure 4.5.5 presents the fitness evolution graph of the GP algorithm. The gradual convergence indicates that the model efficiently updated its parameters with a reduced number of inputs, thus gaining steady performance.



**Figure 4.5.5.** GP Fitness Simulation

The training of the GP model with a simplified set of lagged features and no wavelet decomposition led to a decent training performance but, at the same time, poor generalization on the test data. Omission of current-day inputs and multi-scale signals was the reason why it wasn't that predictable anymore. The fitness curve showed smooth convergence; however, it was still not a satisfactory solution. In general, the experiment demonstrated that the temporal context and the complexity of the signal have great significance for credible precipitation forecasting.



**Figure 4.5.6.** Tree structure of Model-4

Figure 4.5.6 shows the tree structure of Model-4, which was testing for rainfall forecasting. In this model, the input variables labeled  $x$ , such as  $x_{16}$ ,  $x_9$ ,  $x_{12}$ ,  $x_{14}$ ,  $x_4$ ,  $x_7$ , and  $x_{18}$ , represent lagged versions of the meteorological parameters used to forecast rainfall. The selection of these lagged variables was based on the correlation coefficients with rainfall, which ensures that only those variables that have meaningful correlations with rainfall patterns are involved in the modelling process. As a result, the model is able to follow the pattern of the changes in meteorological parameters over time and can predict the changes in rainfall as a result. In addition, the variables  $r_3$ ,  $r_4$ , values in the formula are random constants that are automatically generated during the training process of the GP algorithm, which helps to increase the flexibility of the model and improve the forecast accuracy. The range of random constants can also be specified in the GP settings parameters table.

$$\text{Predicted Precipitation} = (x_{16} \times x_9) \times \left[ \sqrt{r_3 - (x_{12} \times (x_{14} - r_3))} \right] \times \left[ \left( \frac{\sqrt{x_4}}{x_7 + r_4} \right) x_6 ((x_9 + x_{18}) + x_{18}) \right] \quad (4.5.1)$$

In summary the tree structure demonstrates the step-by-step manner in which the model uses the input variables with the aid of mathematical operations, including multiplication, division, and addition, to forecast the precipitation. Every branch of the tree is in charge of a separate calculation that, in the end, results in a prediction of the rainfall.

#### 4.6. GP Model 5 Performance with Lagged Features (No Wavelet)

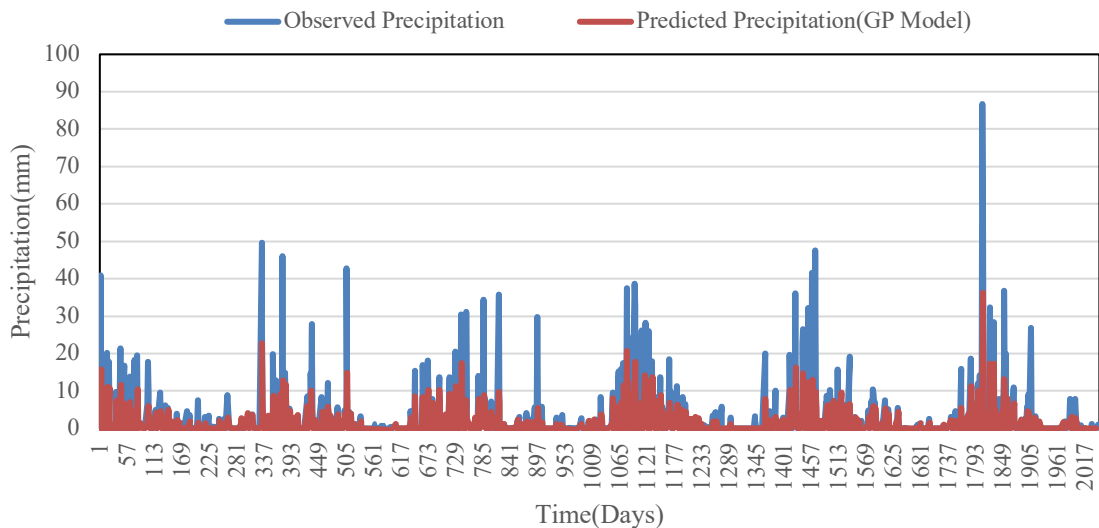
This was done by training the Genetic Programming (GP) model with a subset of the lagged meteorological variables without any wavelet decomposition being applied. It was to see if omitting to include the current day's values ( $T(t)$ ) and basing the model on only a subset of input from the past was enough to achieve satisfactory forecasting accuracy in a less complex modeling setup. The chosen features were identified from

prior correlation analysis and were limited to those with the most statistically significant correlation with precipitation among the lagged inputs. By eliminating both T(t) and the wavelet transformation's added complexity, the model was dramatically simplified. The model was trained on 70% of the data and tested against the other 30%. RMSE and NSE measures were used to evaluate the results. On the training set, RMSE was 5.012 and NSE was 0.179, which reflected a decent fit to the training data. On the test set, RMSE was 5.632, and the NSE fell to 0.235, showing that the model did not generalize very well when starved of both current-day input and multi-scale signal components.

**Table 4.6.1.** Performance of the GP Model Using Selected Lagged Matrics

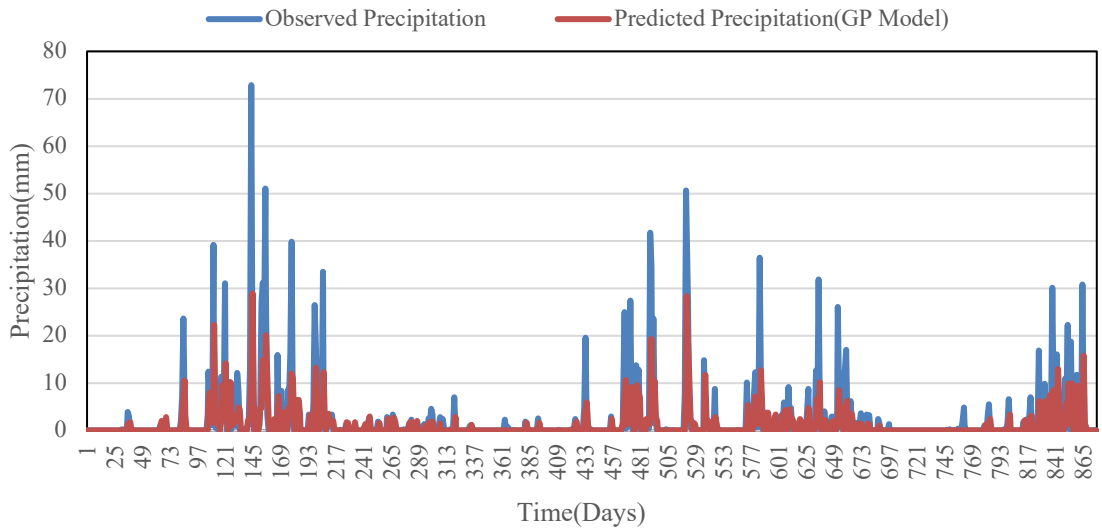
Metrics	Training	Testing
RMSE	5.012	5.632
NSE	0.179	0.235

The figure 4.6.1 represents the correlation of the rainfall values forecasted and observed from the training set using a simplified input dataset, which was without wavelet decomposition. Based on the figure, the model fits well with the training data even after removing the present-day variables and simplifying the input, although it seems that there is still ample space to enhance the model.



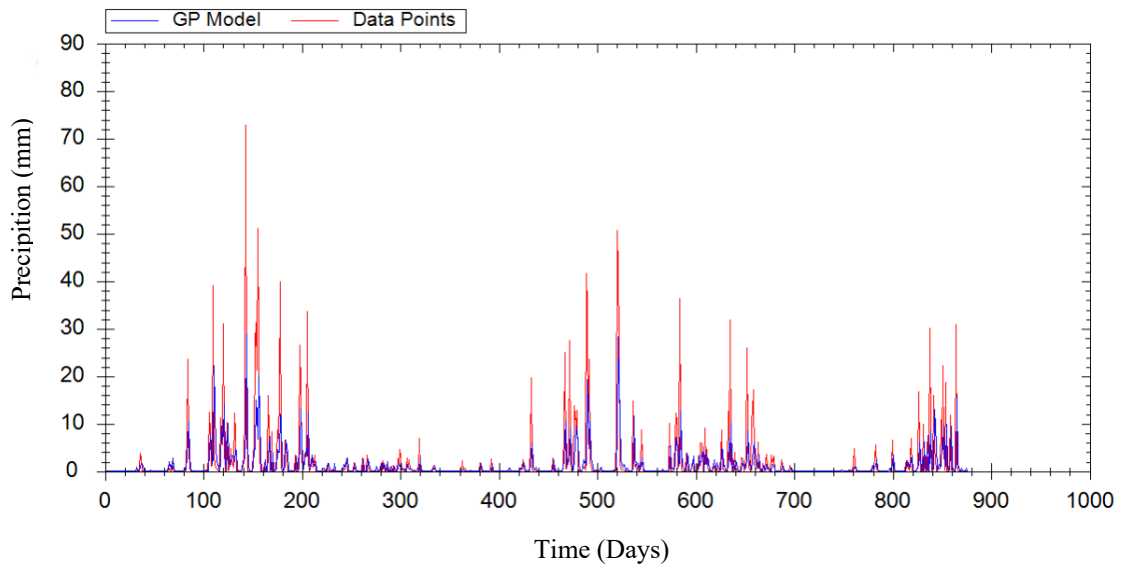
**Figure 4.6.1.** Training Data

The figure 4.6.2 displayed here illustrates the model's accuracy on the testing data set. The errors made are much clearer in this case, which is a sign of little generalization capability if both the info and the features are removed from the aggregates.



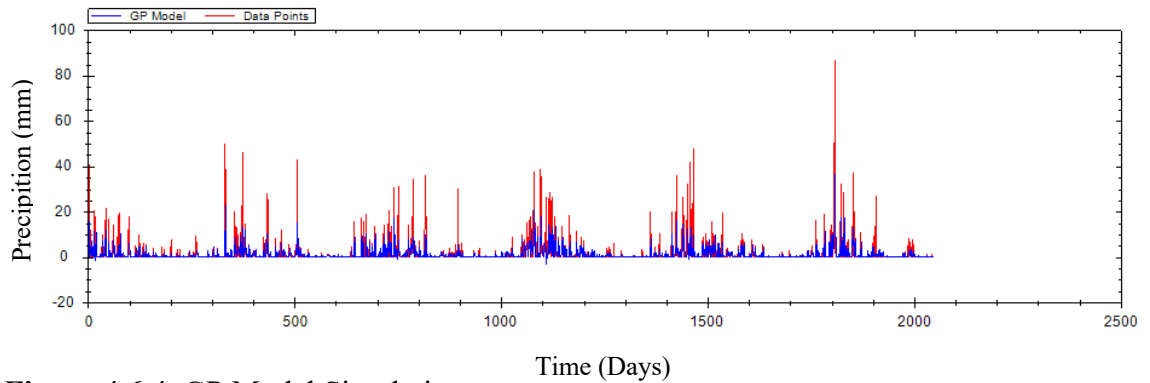
**Figure 4.6.2.** Testing Data

The figure 4.6.3 illustrates a comparison of the real and forecasted precipitation data for the test dataset. The steep contrast between the expected and registered information thus indicates the substantial loss of accuracy in the model, which is led by the absence of some key predictors.



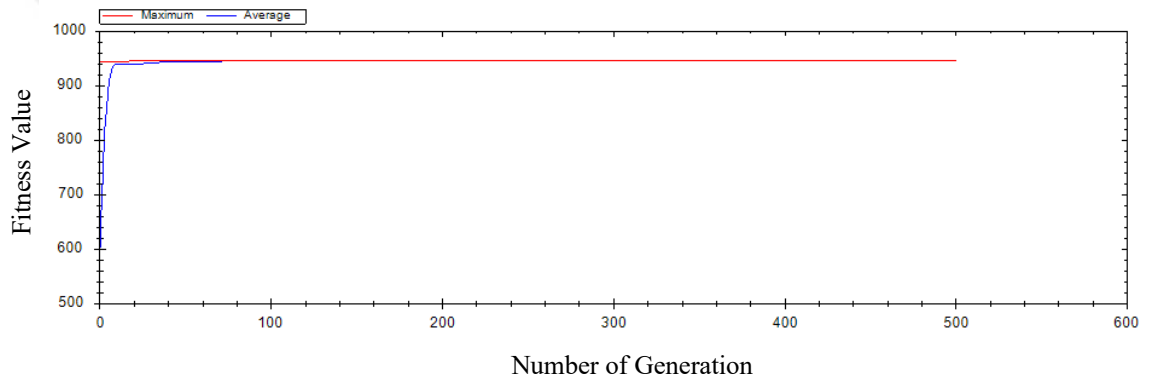
**Figure 4.6.3.** Prediction

The figure 4.6.4 shows the simulation results for the whole dataset, which includes the training and testing phases.



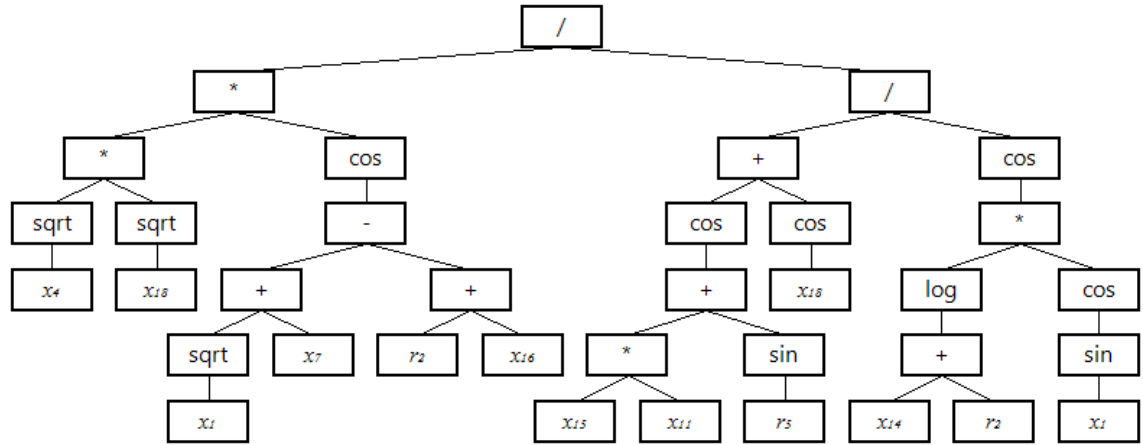
**Figure 4.6.4.** GP Model Simulation

The figure 4.6.5 display the GP method's fitness convergence curve while the algorithm undergoes training. The curve of the graph is stable during the convergence, but it converges to a local minimum, which is just an indication that the model is not capable of getting higher accuracy if it is provided with only a limited set of information.



**Figure 4.6.5.** GP Fitness Simulation

Using a simplified set of time-lagged features without wavelet decomposition to train the GP model displayed a fair training performance and, at the same time, poor generalization performance on the test data. The withdrawal of the multi-scale signals and current-day inputs resulted in an obvious inaccuracy in the predictions. Although the fitness curve suggested uninterrupted stability of the results, the final solution appeared to be less than the optimal one. In general, the test pointed out that, for dependable precipitation prediction, the time context and signal complexity are very important.



**Figure 4.6.6.** Tree Structure of Model-5

Figure 4.6.6 shows the tree structure of Model 5, which is specifically designed for precipitation forecasting and does not use wavelet decomposition. In this model, the input variables represented by the symbol  $x$ , such as  $x_{14}$ ,  $x_{18}$ ,  $x_1$ ,  $x_6$ ,  $x_2$ , and  $x_7$ , represent lagged meteorological parameters. These variables represent the measurements of meteorological data from the previous periods, which, in simple terms, allow the model to account for the time variation in the data and the effect of past weather on future rain. Moreover,  $r_2$ ,  $r_4$ , and  $r_5$  in the equation represent random constants that are chosen automatically in a genetic programming (GP) training process. These constants are used to assemble mathematical expressions with greater flexibility, which then contribute to the model’s better performance while predicting. The range of random constants can also be specified in the GP settings parameters table.

$$\text{predicted precipitation} = \frac{(\sqrt{x_4} \times \sqrt{x_{18}}) \times \cos((\sqrt{x_1} \times x_7) - (r_2 + x_{16}))}{\frac{\cos((x_{15} \times x_{11}) + \sin(r_5)) + \cos(x_{18})}{\cos(\log(x_{14} + r_2) \times \cos(\sin(x_1)))}} \quad (4.6.1)$$

The tree structure of Model-5 demonstrates the model's ability to identify complicated and nonlinear meteorological factors. In conclusion, the structure displays channels with various computation trails that are combined to produce a reliable and accurate precipitation forecast.

**4.7. Model 6 GP with Wavelet-Based Lag Features**

A more refined input feature set was developed in the ultimate modeling exercise by subjecting some of the lagged variables to wavelet decomposition and excluding the precipitation values of the current day (e.g.,  $T(t)$ ) intentionally. This was to mimic a scenario of actual forecasting and assess the performance of the model under a situation where only past and decomposed information is accessible.

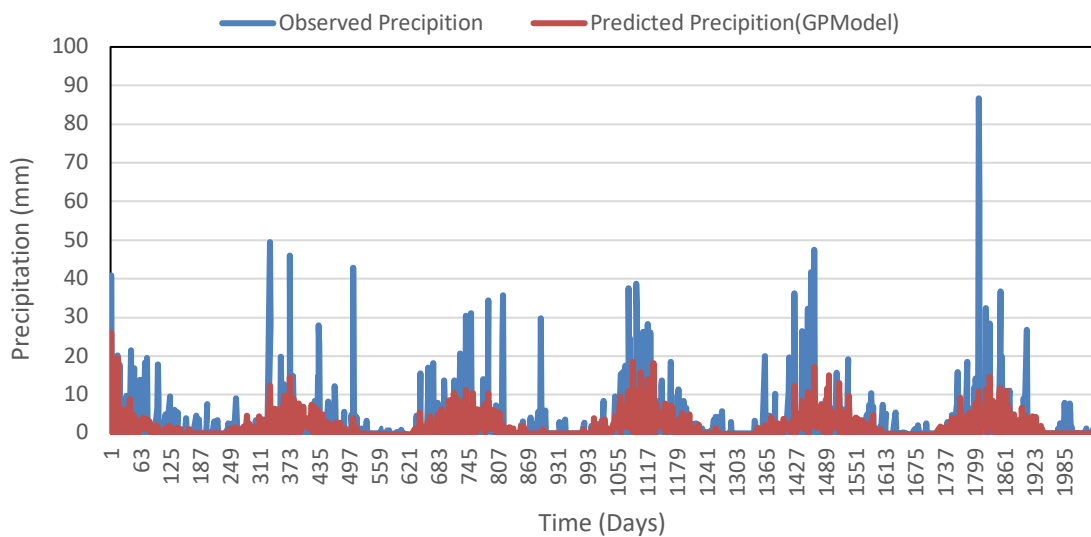
All the selected lagged features—identified through prior correlation studies—were split into several components via discrete wavelet transformation. These sub-series, consisting of detail and approximation levels, were inputs to the Genetic Programming (GP) model. Similar to earlier models, 70% of the data was employed in training and 30% in testing. Model performance, measured in terms of RMSE and NSE, decreased. On the

training set, RMSE was 4.737 and NSE was 0.266. On the test set, RMSE rose to 5.430 and NSE fell to 0.289. These values indicate that without the current value (T(t)), and with the extra complexity of wavelet decomposition added to it, the predictive accuracy of the GP model was substantially decreased.

**Table 4.7.1.** Matrics of the GP Model

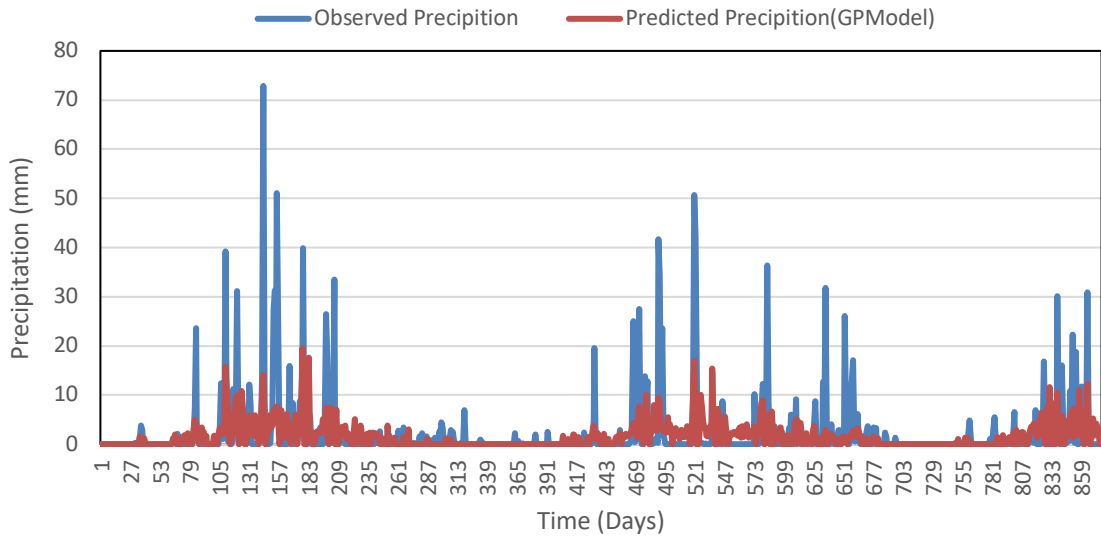
Metrics	Training	Testing
RMSE	4.737	5.430
NSE	0.266	0.289

The figure 4.7.1 displays the actual and forecasted precipitation through the selected lagged features with wavelet decomposition in the training dataset. The figure illustrates the impact of the wavelet decomposition even when the model fit to the training data was largely affected, indicating the difficulty of excluding today’s inputs.



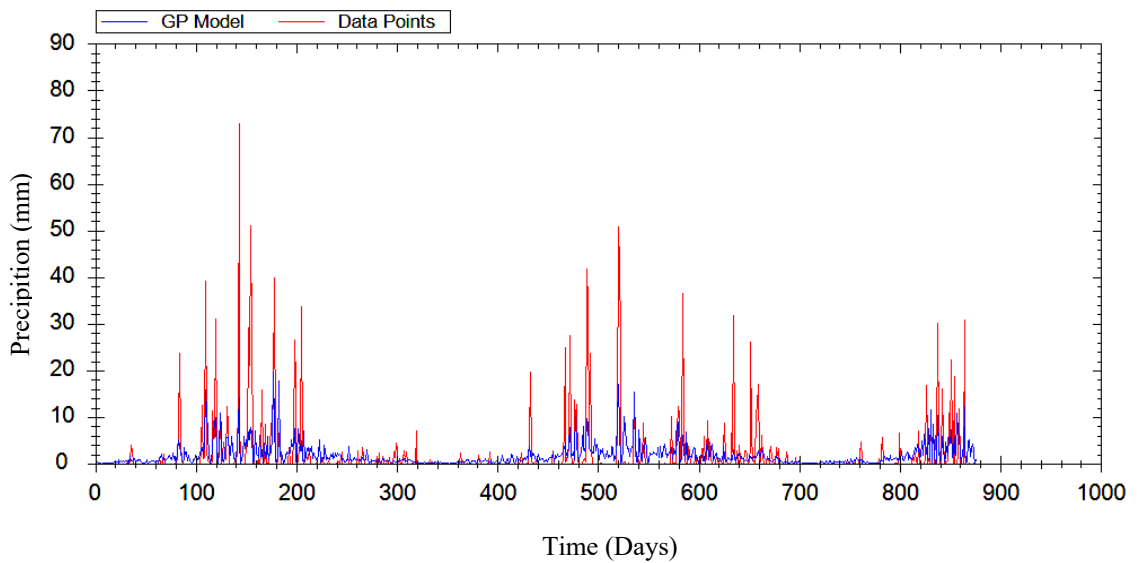
**Figure 4.7.1.** Training Data

This figure 4.7.2 shows the model's predictive performance on the experimental data set. The figure shows significant discrepancies between the predictions and actual observations, indicating limited generalizability when relying on past, decomposed inputs.



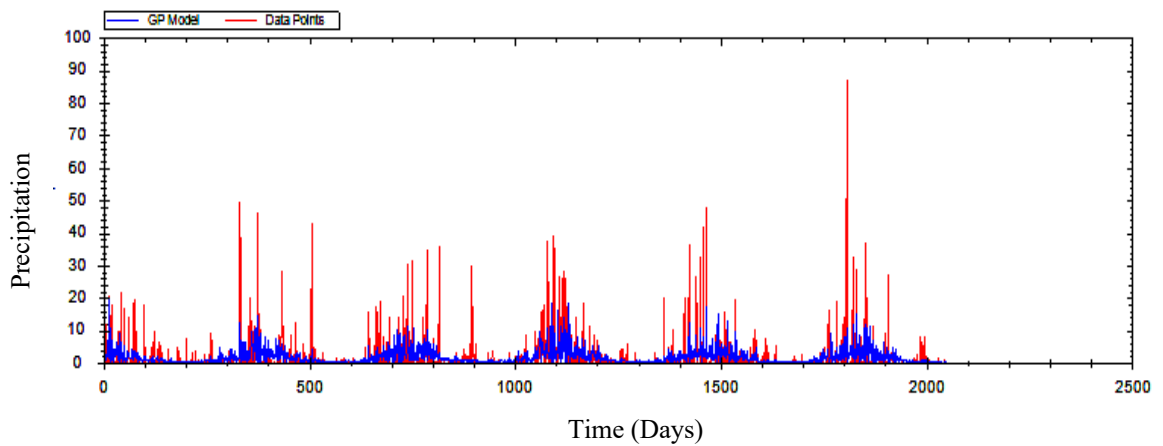
**Figure 4.7.2.** Testing Data

The figure 4.7.3 gives the discrepancy between actual and predicted precipitation values in the test set data. The mismatch of the curves at most of the points emphasizes the great loss of predictive capability, just because the model was deprived of real-time inputs and used the wavelet decomposing procedure.



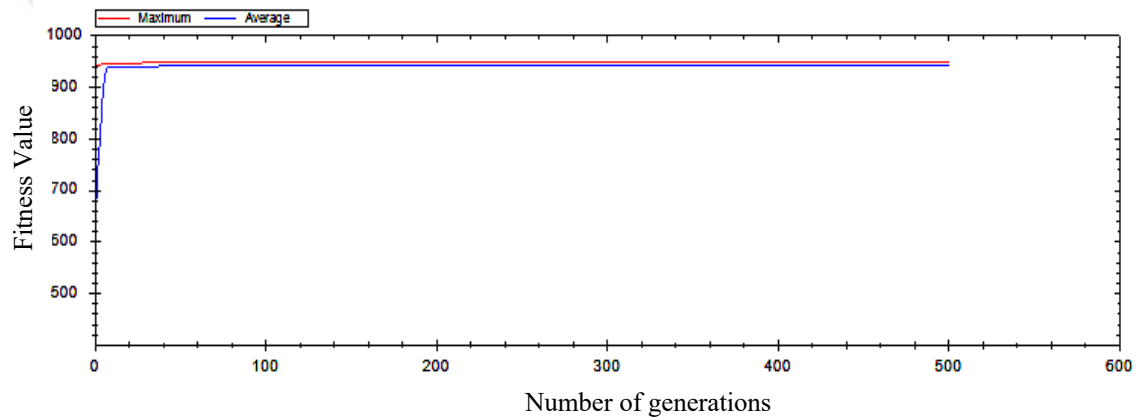
**Figure 4.7.3** Prediction

The figure 4.7.4 shows the simulation results on the entire dataset, combining training and testing phases. It shows the instability and inconsistency of the model predictions under this configuration.



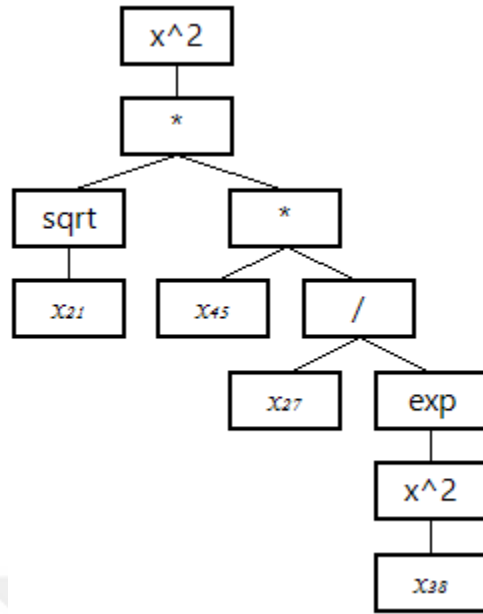
**Figure 4.7.4.** GP Model Simulation

This figure 4.7.5 presents the fitness convergence curve of the GP algorithm during training.



**Figure 4.7.5** GP Fitness Simulation

Using time-delayed selected features along with wavelet decomposition and removal of current day data resulted in a significant decrease in the prediction accuracy of the GP model. Both RMSE increased and NSE decreased in both training and test datasets, indicating a decrease in model performance. The fit curve showed convergence but was at a suboptimal level. Overall, this experiment shows that wavelet decomposition alone cannot compensate for the loss of critical real-time inputs in rainfall prediction.



**Figure 4.7.6.** Tree Structure of Model-6

Figure 4.7.6 the tree structure of Model-6 used for precipitation forecasting. In this model, the input variables,  $x_{21}$ ,  $x_{45}$ ,  $x_{27}$ , and  $x_{38}$ , are explain the wavelet-based lag features taken from meteorological parameters and used for precipitation forecasting . This means that the main meteorological data, such as temperature, humidity, wind speed, solar radiation, and precipitation, are first divided into several components at different frequencies using the wavelet decomposition method. These wavelet components are then considered as lagged data to include information from past times. This process helps the model to identify both short-term fluctuations and long-term trends in precipitation patterns.

$$\text{predicted Precipitation} = \left( \sqrt{x_{21}} \times x_{45} \times \frac{x_{27}}{\exp(x_{38}^2)} \right)^2 \quad (4.7.1)$$

This structure is indicative of the model's capability of identifying nonlinear associations between the input meteorological variables. The introduction of square root, squared power, and exponential functions tells us that the model is prepared for complex interactions and oscillations that are likely to emerge in the rainfall data. In the final analysis, the tree diagram elucidates the way the model combines diverse computational paths to give accurate precipitation predictions.

#### 4.8. Comparative Analysis of All GP Models

This section summarizes and compares the performance of all tested genetic programming (GP) models for daily precipitation forecasting. In this comparison, various combinations of input features, including raw data, lagged data, wavelet decomposition, and feature selection based on statistical correlation, were investigated. Two main indicators were used to evaluate the performance of the models: the root mean square error (RMSE) and the Nash-Sutcliffe efficiency (NSE), both for training and test data.

Table 4.7.1 summarizes the performance of all experiments. The results clearly indicate that the model using only time-lagged features without wavelet decomposition achieved the best forecast accuracy. This model had the lowest RMSE and the highest NSE in the experimental dataset (RMSE = 3.645 and NSE = 0.680), indicating that temporal dependencies play a key role in detecting precipitation patterns.

**Table 4.8.1.** Comparative of All GP Models

Model Description	RMSE (Train)	NSE (Train)	RMSE (Test)	NSE (Test)
Model 1 GP Result	3.639	0.567	3.717	0.667
Model 2 Hybrid WT-GP	4.384	0.372	4.732	0.460
Model 3 GP with Lagged Data Features	3.588	0.579	3.645	0.680
Model 4 Correlation Analysis of Lagged Data not Decomposed Data	3.614	0.573	3.695	0.671
GP Model 5 Performance with Lagged Features (No Wavelet)	5.012	0.179	5.632	0.235
Model 6 GP with Wavelet-Based Lag Features	4.737	0.266	5.430	0.289

In contrast, models that used wavelet-decomposed features, despite their strong theory, performed poorly in extracting multiscale patterns. The added complexity of these inputs reduced model accuracy and increased the risk of overfitting. For example, the model that used time-lagged data and wavelet decomposition (without the  $T(t)$  variable) yielded one of the poorest results (RMSE = 5.430 and NSE = 0.289).

The model using features selected based on correlation also performed relatively well. This shows that statistical feature selection can reduce the complexity of the model while maintaining its accuracy. However, removing same-day variables (such as  $T(t)$ ) had a strong negative impact on the performance of the models.

Overall, the results show that the simplicity of the model, combined with the consideration of temporal memory, outperforms the complex configurations. Therefore, the GP model trained with a limited set of temporal features and not using wavelet decomposition provides the best balance between computational efficiency and prediction accuracy.

## 4.9. Key Observations

### 1. Best Overall Performance

The model that employed the lagged meteorological attributes without wavelet decomposition performed the best overall. It registered the lowest RMSE (3.645) and highest NSE (0.680) on the testing dataset and also had good training statistics. This verifies the efficiency of utilizing temporal attributes in maintaining model accuracy and generalization.

### 2. Underperformance of Wavelet-Only Models

Models that employed wavelet decomposition without the use of lag performed less than optimally both in training and in the test set. For instance, the raw wavelet-decomposed model yielded a high RMSE of 4.732 and NSE of 0.460 in the test set. This suggests that wavelet transformation, with its otherwise tremendous capability in picking up multi-scale patterns, may result in redundancy and noise in symbolic regression models such as GP.

### 3. Combined Wavelet and Lagged Inputs Did Not Enhance Accuracy

The model with both wavelet and lagged indicators (Model 4) did not perform any better than the use of lag alone. RMSE and NSE results were similar to those of the wavelet-only model and indicated that increased model complexity did not increase predictive capacity.

### 4. Impact of Feature Selection

Correlation-based feature selection resulted in a modest decrease in model complexity without much sacrifice in accuracy. Models 5 and 6 produced decent performance with reduced inputs and demonstrate the utility of data-driven feature reduction techniques.

### 5. Negative Impact of Eliminating $T(t)$

Models that did not include the input of the current day ( $T(t)$ ) exhibited considerable degradation in performance. Model 6's exclusion of  $T(t)$  caused NSE to decline to 0.226, and Model 7 (with wavelet decomposition added) saw NSE drop to 0.289. These findings verify that the current day ( $T(t)$ ) is a significant variable in forecasting precipitation over the short term.

Based on the comparative study, it can be concluded that

- Lagged input features perform best for precipitation forecasting where GPs are employed.
- Wavelet decomposition on its own or in conjunction with delayed inputs did not enhance performance and actually decreased it in some instances.

- Feature selection is helpful in reducing the dimensionality in the case of lag-based data.
- Excluding  $T(t)$  substantially reduces model performance and indicates its critical importance to precise forecasting.

Hence, the best model determined in this research is that with the use of lagged meteorological variables without wavelet decomposition that has the best balance between model complexity and performance.



## 5. CONCLUSIONS AND SUGGESTIONS

The main idea of this research was the development of short-term rainfall prediction with the use of the genetic programming (GP) method, which was experimented with using the daily meteorological data of Antalya, Turkey, from 2016 to 2023. The study comprised a structured examination of the predictive performance of models based on different input data types, i.e., non-lagged measurements, lagged variables, wavelet-transformed series, as well as mixed types of these, for the forecasting of the amount of the next day's precipitation.

### 5.1. The Summary Points From This Research

**Model performance across experiments:** The GP model, the use of lagged meteorological variables but without wavelet decomposition, was the most accurate in forecasting, which was observed by the lowest RMSE and the highest NSE scores, and thus more emphasized the significance of temporal patterns in rainfall prediction.

**Wavelet decomposition influence:** The application of wavelet decomposition, with and without lagged variables, contrary to what was expected and what some previous research had shown, didn't have any improvement on the model performance. To put it straightforwardly, the increased complexity and possible data overlap of the wavelet transformation technique also led to a decrease in the accuracy of the solution.

**Feature selection effects:** The introduction of feature selection, which was performed based on the correlation analysis, the computational load and training time of the model, while maintaining high predictive accuracy. These results underline the relevance of dimensionality reduction techniques in data-driven hydrological modeling.

**Exclusion of current-day variables:** When current-day variables ( $T(t)$ ) were deleted, model performance went down dramatically. From this, we can infer that immediate weather-related data is a must to have high-quality rainfall predictions.

**Performance comparison across model types:** When observing the different models' results, the lagged-only GP model was consistently better than the other models. The wavelet versions of the models faced significant challenges in generalization, particularly when they were coupled with lagged features. Among the baseline approaches, the kernel method showed promising initial performance, but it was still the lagged feature model that was clearly ahead. The raw data model without any temporal transformation served as a useful baseline but was easily outperformed by the lagged-only GP model.

### 5.2. Scientific Implications

The findings of this study indicate that in the case of Antalya's climatic and meteorological conditions, the employment of lagged variables for capturing temporal dependencies is more effective for rainfall prediction than the introduction of wavelet transform-based decomposition layers. Although wavelet methods have been effective in

other environmental modeling applications, their efficiency is quite dependent on data characteristics, model configuration, and the variable to be predicted.

This study thus supports the broader understanding in the field of hydrological modeling that smart feature engineering is one of the most important factors in the algorithm selection process, more than the algorithm specifically.

### 5.3. Comparison with Previous Studies

The study results are multipointly related to previous works that emphasize that a time lag plays a significant role in hydrological predictions. The difference with other previous studies, which reported that wavelet-based hybrid models led to better predictions, is that this research indicated a different tendency. Such deviation might be due to

- Differences in data resolution and variability,
- Differences in GP parameter settings
- Region-specific climate behavior
- Or limitations in how wavelet coefficients were used as inputs.

Future studies should aim to determine the specific conditions under which wavelet decomposition enhances machine learning model performance in hydrology.

### 5.4. Limitations of the Study

**Geographic focus:** The study was only restricted to the area of Antalya, resulting in the models not being able to be applied to other regions.

**Computational requirements:** The GP models, particularly those involving numerous input variables or wavelet-transformed data, were computationally demanding.

**Excluded environmental variables:** Such potential factors as soil moisture, evaporation, or remote sensing data were not considered.

**Model validation scope:** Despite the application of train-test splits for model testing, there was no independent data or cross-regional validations.

### 5.5. Recommendations for Future Research

The research and its limitations together suggest what could be done in the future in several areas:

**Hybrid modeling methods:** The potential to merge GP with other decomposition methods, such as empirical mode decomposition (EMD) or singular spectrum analysis (SSA), is examined.

**Algorithmic comparisons:** Compare GP models with best-in-class algorithms like long short-term memory (LSTM) networks, extreme gradient boosting (XGBoost), and random forests.

**Inclusion of broader datasets:** Augment the datasets with other variables such as soil moisture, atmospheric pressure deviations, or satellite-observed indices.

**Cross-regional validation:** Use the models that were developed in several climatic regions to find out if they are fitting and reliable.

**Uncertainty quantification:** Construct a probabilistic approach of GP models that will help in explaining the uncertainty of the prediction to the final users.

**Optimization of computation:** Research about how to decrease the time of running, like using parallel processing or optimizing the GP parameter search.

This document is adding new knowledge to the hydrological forecasting field by presenting the advantages and disadvantages of GP-based rainfall prediction models as per different input configurations. The study suggests that

- Time-lagged meteorological variables play a crucial role in improving the accuracy of precipitation forecasting models.
- Wavelet decomposition is not necessarily a good way to increase performance.
- feature selection is responsible for improving the model's promptness without inducing any loss in the prediction accuracy.

By handling the problems mentioned in this study and exploring the solutions recommended for future development, more reliable and universal forecasting systems can be constructed that contribute to better water resource management and disaster preparedness.

## REFERENCES

- Adamowski, J. F. (2008). River flow forecasting using wavelet and cross-wavelet transform models. *Hydrological Processes: An International Journal*, 22(25), 4877–4891.
- Babovic, V., & Keijzer, M. (2002). Rainfall runoff modelling based on genetic programming. *Hydrology Research*, 33(5), 331–346.
- Box, G. E., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time series analysis: Forecasting and control*. John Wiley & Sons.
- Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5–32. <https://doi.org/10.1023/A:1010933404324>
- Chai, T., & Draxler, R. R. (2014). Root means square error (RMSE) or mean absolute error (MAE)? –Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, 7(3), 1247–1250.
- Chavan, G., & Momin, B. (2019). A Novel Approach for Forecasting the Linear and Nonlinear Weather Data Using Support Vector Regression. In P. K. Sa, S. Bakshi, I. K. Hatzilygeroudis, & M. N. Sahoo (Eds.), *Recent Findings in Intelligent Computing Techniques* (pp. 419–423). Springer Singapore.
- Cordero, E., & Rodino, L. (2008). Short-time Fourier transform analysis of localization operators. *Contemporary Mathematics*, 451, 47.
- Danandeh Mehr, A. (2020). An ensemble genetic programming model for seasonal precipitation forecasting. *SN Applied Sciences*, 2(11), 1821. <https://doi.org/10.1007/s42452-020-03625-x>
- Daubechies, I. (1990). The wavelet transforms, time-frequency localization and signal analysis. *IEEE Transactions on Information Theory*, 36(5), 961–1005.
- Giustolisi, O., & Savic, D. A. (2006). A symbolic data-driven technique based on evolutionary polynomial regression. *Journal of Hydroinformatics*, 8(3), 207–222.
- Gupta, H. V., Kling, H., Yilmaz, K. K., & Martinez, G. F. (2009). Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling. *Journal of Hydrology*, 377(1–2), 80–91.
- Han, J.-Y., & Hong, S.-Y. (2018). Precipitation Forecast Experiments Using the Weather Research and Forecasting (WRF) Model at Gray-Zone Resolutions. *Weather and Forecasting*, 33(6), 1605–1616. <https://doi.org/10.1175/WAF-D-18-0026.1>
- Haykin, S. S. (1999). *Neural Networks: A Comprehensive Foundation*. Prentice Hall. <https://books.google.com.tr/books?id=bX4pAQAAMAAJ>

- Huang, N. E., Shen, Z., Long, S. R., Wu, M. C., Shih, H. H., Zheng, Q., Yen, N.-C., Tung, C. C., & Liu, H. H. (1998). The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 454(1971), 903–995.
- Kalnay, E. (2002). *Atmospheric Modeling, Data Assimilation and Predictability*. Cambridge University Press; Cambridge Core. <https://doi.org/10.1017/CBO9780511802270>
- Kalnay, E. (2003). *Atmospheric modeling, data assimilation and predictability*. Cambridge university press.
- Katimon, A., Shahid, S., & Mohsenipour, M. (2018). Modeling water quality and hydrological variables using ARIMA: a case study of Johor River, Malaysia. *Sustainable Water Resources Management*, 4(4), 991–998. <https://doi.org/10.1007/s40899-017-0202-8>
- Katipoğlu, O. M., Yeşilyurt, S. N., Dalkılıç, H. Y., & Akar, F. (2023). Application of empirical mode decomposition, particle swarm optimization, and support vector machine methods to predict stream flows. *Environmental Monitoring and Assessment*, 195(9), 1108.
- Koza, J. R. (1992). *Evolution of subsumption using genetic programming*. 110–119.
- Kratzert, F., Klotz, D., Brenner, C., Schulz, K., & Herrnegger, M. (2018). Rainfall–runoff modelling using Long Short-Term Memory (LSTM) networks. *Hydrology and Earth System Sciences*, 22(11), 6005–6022. <https://doi.org/10.5194/hess-22-6005-2018>
- Krause, P., Boyle, D., & Bäse, F. (2005). Comparison of different efficiency criteria for hydrological model assessment. *Advances in Geosciences*, 5, 89–97.
- Kumar, R., Goel, R., Sidana, N., Sharma, A., ghai, S., Singh, T., singh, R., Priyadarshi, N., Twala, B., & Ahmad, V. (2024). Enhancing climate forecasting with AI: Current state and future prospect [version 1; peer review: 1 approved]. *F1000Research*, 13(1094). <https://doi.org/10.12688/f1000research.154498>.
- Labat, D. (2005). Recent advances in wavelet analyses: Part 1. A review of concepts. *Journal of Hydrology*, 314(1–4), 275–288.
- Laleh Parviz, Kabir Rasouli, & Ali Torabi Haghighi. (2023). Improving Hybrid Models for Precipitation Forecasting by Combining Nonlinear Machine Learning Methods. *Springer*, 23.
- Legates, D. R., & McCabe Jr, G. J. (1999). Evaluating the use of “goodness-of-fit” measures in hydrologic and hydroclimatic model validation. *Water Resources Research*, 35(1), 233–241.

- Makridakis, S. (1993). Accuracy measures: Theoretical and practical concerns. *International Journal of Forecasting*, 9(4), 527–529.
- Mallat, S. (1999). *A wavelet tour of signal processing*. Elsevier.
- Muttill, N., & Lee, J. H. (2005). Genetic programming for analysis and real-time prediction of coastal algal blooms. *Ecological Modelling*, 189(3–4), 363–376.
- Nash, J. E., & Sutcliffe, J. V. (1970). River flow forecasting through conceptual models part I—A discussion of principles. *Journal of Hydrology*, 10(3), 282–290.
- Parasuraman, K., & Elshorbagy, A. (2008). Toward improving the reliability of hydrologic prediction: Model structure uncertainty and its quantification using ensemble-based genetic programming framework. *Water Resources Research*, 44(12).
- Savic, D. A., Walters, G. A., & Davidson, J. W. (1999). A genetic programming approach to rainfall-runoff modelling. *Water Resources Management*, 13, 219–231.
- Sivapragasam, C., Vincent, P., & Vasudevan, G. (2007). Genetic programming model for forecast of short and noisy data. *Hydrological Processes: An International Journal*, 21(2), 266–272.
- Virgolin, M. (2021). *Genetic programming is naturally suited to evolve bagging ensembles*. 830–839.
- Waqas, M., Humphries, U. W., Chueasa, B., & Wangwongchai, A. (2024). Artificial Intelligence and Numerical Weather Prediction Models: A Technical Survey. *Natural Hazards Research*.
- Wilks, D. S. (2011). Chapter 7—Statistical Forecasting. In D. S. Wilks (Ed.), *International Geophysics* (Vol. 100, pp. 215–300). Academic Press. <https://doi.org/10.1016/B978-0-12-385022-5.00007-5>
- Willmott, C. J., & Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate Research*, 30(1), 79–82.
- Yaseen, Z. M., Jaafar, O., Deo, R. C., Kisi, O., Adamowski, J., Quilty, J., & El-Shafie, A. (2016). Stream-flow forecasting using extreme learning machines: A case study in a semi-arid region in Iraq. *Journal of Hydrology*, 542, 603–614.
- Žegklitz, J., & Pošík, P. (2015). *Model selection and overfitting in genetic programming: Empirical study*. 1527–1528.