**WEATHER PREDICTION SYSTEM**

**A PROJECT REPORT**

*for*

**DATA MINING TECHNIQUES (SWE2009)**

*in*

**M.Tech (Integrated) Software Engineering**

*by*

**OHM PRAKASANATHAM S (19MIS0154)**

**SURENDHAR A S (19MIS0172)**

*Under the Guidance of*

**Dr. SENTHILKUMAR N C**

Associate Professor, SITE



**School of Information Technology and Engineering**

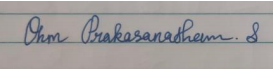
April, 2023

**DECLARATION BY THE CANDIDATE**

I hereby declare that the project report entitled **“WEATHER PREDICTION SYSTEM”** submitted by me to Vellore Institute of Technology; Vellore in partial fulfillment of the requirement for the award of the course **Data Mining Techniques (SWE2009)** is a record of bonafide project work carried out by us under the guidance of **Dr. Senthilkumar N C.** I further declare that the work reported in this project has not been submitted and will not be submitted, either in part or in full, for the award of any other course.

Place:Vellore

Date: 10.04.2023 Signature





**School of Information Technology & Engineering [SITE]**

**CERTIFICATE**

This is to certify that the project report entitled **“WEATHER PREDICTION SYSTEM”** submitted by **SRINIVAS S (19MIS0276)** to Vellore Institute of Technology, Vellore in partial fulfillment of the requirement for the award of the course **Data Mining Techniques (SWE2009)** is a record of bonafide work carried out by them under my guidance.

**Dr. Senthilkumar N C**

**Associate Professor, SITE**

**Weather Prediction System**

**Abstract**

Weather forecasting is the application of science and technology to predict the state of the atmosphere for a given location. Ancient weather forecasting methods usually relied on observed patterns of events, also termed pattern recognition. For example, it might be observed that if the sunset was particularly red, the following day often brought fair weather. However, not all of these predictions prove reliable. Here this system will predict weather based on parameters such as temperature, humidity and wind. User will enter current temperature; humidity and wind, System will take this parameter and will predict weather(rainfall in inches) from previous data in database(dataset). Weather forecasting system takes parameters such as temperature, humidity, and wind and will forecast weather based on previous record therefore this prediction will prove reliable. This system can be used in Air Traffic, Marine, Agriculture, Forestry, Military, and Navy etc….

**Keywords** – XGBoost , ,Pattern recognition, temperature, humidity and wind

1. **INTRODUCTION**

Weather prediction is a complex task that requires accurate analysis and modeling of various meteorological data. Machine learning algorithms such as Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Extreme Boosting (XGBoost) have been widely used for weather prediction tasks due to their ability to handle high-dimensional and nonlinear data. In this comparison, we will evaluate and compare the performance of these algorithms for weather prediction. Specifically, we will analyze their accuracy, computational efficiency, and suitability for real-time weather prediction. The goal of this comparison is to provide insights into the strengths and weaknesses of these algorithms and help researchers and practitioners choose the most appropriate algorithm for their specific weather prediction task.

1. **BACKGROUND**

SVM, KNN, and XGBoost are three popular machine learning algorithms that can be used for weather prediction. Here is a brief explanation of each algorithm and its suitability for weather prediction:

**Support Vector Machines (SVM):**

SVM is a powerful algorithm for classification and regression tasks. It works by finding a hyperplane that separates the data into different classes, with a margin that maximizes the distance between the hyperplane and the closest points from each class. In weather prediction, SVM can be used to classify weather conditions into different categories (e.g., sunny, cloudy, rainy, etc.) based on input features such as temperature, humidity, wind speed, and pressure. However, SVM may not be as suitable for predicting continuous variables such as temperature or precipitation levels.

**K-Nearest Neighbors (KNN):**

KNN is a simple yet effective algorithm for classification and regression tasks. It works by finding the k-nearest neighbors of a given data point and using their labels (in classification) or values (in regression) to predict the label or value of the new data point. In weather prediction, KNN can be used to predict weather conditions or continuous variables based on the similarity of input features with previously observed data points. However, KNN can be computationally expensive for large datasets, and the choice of k can have a significant impact on the accuracy of the model.

**XGBoost:**

XGBoost is a powerful ensemble learning algorithm that can be used for classification and regression tasks. It works by building an ensemble of decision trees, where each tree is trained to predict the residuals of the previous tree. XGBoost is particularly well-suited for weather prediction due to its ability to handle complex nonlinear relationships in the data, as well as its ability to handle missing data and outliers. XGBoost can be used to predict weather conditions or continuous variables based on input features such as temperature, humidity, wind speed, and pressure.

Overall, the choice of algorithm for weather prediction depends on the specific task at hand, the size and complexity of the dataset, and the desired level of accuracy and interpretability.

**Literature Survey**

From all these Literature surveys , researchers have used Machine learning and Deep learning for weather prediction and they train their system with the past data set of traditional weather prediction systems based on pattern of events .

**[1].** This paper discusses the Prediction of Total Cloud Cover (TCDC) from numerical weather simulation models, such as Global Forecast System (GFS), can aid renewable energy engineers in monitoring and forecasting solar photovoltaic power generation. A major challenge is the systematic bias in TCDC simulations induced by the errors in the numerical model parameterization stages. Correction of GFS-derived cloud forecasts at multiple time steps can improve energy forecasts in electricity grids to bring better grid stability or certainty in the supply of solar energy. We propose a new kernel ridge regression (KRR) model to reduce bias in TCDC simulations for medium-term prediction at the inter-daily, e.g., 2–8 day-ahead predicted TCDC values .

**[2]**. This study proposes an effective model for enhancing the short-term wind speed forecasting performance by considering the effect of multiple meteorological factors. (a) The filter-wrapper non-dominated sorting differential evolution algorithm incorporating K-medoid clustering (FWNSDEC) is designed to select key meteorological factors and generate multiple feature subsets. For each feature subset, the hybrid deep learning model is designed: (b) singular spectrum analysis (SSA) is used to decompose the meteorological factors and construct the three-dimensional input structure; (c) convolutional long short-term memory (ConvLSTM) network is then adopted to process the sample set of three-dimensional sequence, and the final forecasting result is the average prediction of all the built ConvLSTMs.

**[3]**. Existing weather forecasting models are based on physics and use supercomputers to evolve the atmosphere into the future. Better physics-based forecasts require improved atmospheric models, which can be difficult to discover and develop, or increasing the resolution underlying the simulation, which can be computationally prohibitive. An emerging class of weather models based on neural networks overcome these limitations by learning the required transformations from data instead of relying on hand-coded physics and by running efficiently in parallel. Here we present a neural network capable of predicting precipitation at a high resolution up to 12 h ahead. The model predicts raw precipitation targets and outperforms for up to 12 h of lead time state-of-the-art physics-based models currently operating in the Continental United States. The results represent a substantial step towards validating the new class of neural weather models.

**[4]**. Precipitation in any form—such as rain, snow, and hail—can affect day-to-day outdoor activities. Rainfall prediction is one of the challenging tasks in weather forecasting process. Accurate rainfall prediction is now more difficult than before due to the extreme climate variations. Machine learning techniques can predict rainfall by extracting hidden patterns from historical weather data. Selection of an appropriate classification technique for prediction is a difficult job. This research proposes a novel real-time rainfall prediction system for smart cities using a machine learning fusion technique. The proposed framework uses four widely used supervised machine learning techniques, decision tree, Naïve Bayes, K-nearest neighbors, and support vector machines. For effective prediction of rainfall, the technique of fuzzy logic is incorporated in the framework to integrate the predictive accuracies of the machine learning techniques, also known as fusion.

**[5].** Forecasting wind speed near the surface with high-spatial resolution is beneficial in agricultural management. There is a discrepancy between the wind speed information required for agricultural management and that produced by weather agencies. To improve crop yield and increase farmers’ incomes, wind speed prediction systems must be developed that are customized for agricultural needs. The current study developed a high-resolution wind speed forecast system for agricultural purposes in South Korea. The system produces a wind speed forecast at 3 m aboveground with 100-m spatial resolution across South Korea. Logarithmic wind profile, power law, random forests, support vector regression, and extreme learning machine were tested as candidate methods for the downscaling wind speed data. The wind speed forecast system developed in this study provides good performance, particularly in inland areas.

**[6]**. Weather forecasting is an important application in meteorology and has been one of the most scientifically and technologically challenging problems around the world. As the drastic effects of climate change continue to unfold, localised short term weather prediction with high accuracy has become more important than ever. In this paper, a collaborative machine learning-based real-time weather forecasting system has been proposed whereby data from several locations are used to predict the weather for a specific location. In this work, five machine learning algorithms have been used and tests have been performed in four different locations in Mauritius to predict weather parameters such as Temperature, Wind Speed, Wind Direction, Pressure, Humidity, and Cloudiness.

**[7].** As one of the most concerned issues in modern society, air quality has received extensive attentions from the public and the government, which promotes the continuous development and progress of air quality forecasting technology. In this study, an automated air quality forecasting system based on machine learning has been developed and applied for daily forecasts of six common pollutants (PM2.5, PM10, SO2, NO2, O3, and CO) and pollution levels, which can automatically find the best “Model + Hyperparameters” without human intervention. Five machine learning models and an ensemble model (Stacked Generalization) were integrated into the system, supported by a knowledge base containing the meteorological observed data, pollutant concentrations, pollutant emissions, and model reanalysis data.

**[8]**. This Article discusses the question of whether it is possible to completely replace the current numerical weather models and data assimilation systems with DL approaches. This discussion entails a review of state-of-the-art machine learning concepts and their applicability to weather data with its pertinent statistical properties. We think that it is not inconceivable that numerical weather models may one day become obsolete, but a number of fundamental breakthroughs are needed before this goal comes into reach.This article is part of the theme issue ‘Machine learning for weather and climate modelling’.

**[9]**. In this article, they demonstrate that the proposed lightweight deep model can be utilised for weather forecasting up to 12 h for 10 surface weather parameters. The model outperformed the state-of-the-art WRF model for up to 12 h. The proposed model could run on a standalone computer, and it could easily be deployed in a selected geographical region for fne-grained short to medium-term weather prediction. Furthermore, the proposed model is able to overcome some challenges within the WRF model, such as the understanding of the model and its installation, as well as its execution and portability. There are three types of results, namely: (1) a comparison of various machine learning techniques, statistical forecasting approaches, and a dynamic ensemble method with the proposed approach for weather forecasting, (2) performance of short-term weather forecasting, and (3) performance of long-term weather forecasting using the proposed model.

**[10]**. This paper proposes that conducted with the AROME-Arctic regional mesoscale numerical weather prediction system, using as lateral boundary conditions (LBCs) observing-system experiments performed at the European Centre for Medium-Range Forecasts (ECMWF) with the global forecasting system. This allows the assessment of the relative impacts of observations on forecast skill through regional data assimilation (DA), through LBCs, and the total impact due to the denial of observations in both the regional and global forecasting systems.

**[11]**. The Red Sea provides 90% of the Kingdom’s potable water by desalinization, supporting tourism, shipping, aquaculture, and fishing industries, which together contribute about 10%–20% of the country’s GDP. All these activities, and those elsewhere in the Red Sea region, critically depend on oceanic and atmospheric conditions. At a time of mega-development projects along the Red Sea coast, and global warming, authorities are working on optimizing the harnessing of environmental resources, including renewable energy and rainwater harvesting. All these require high-resolution weather and climate information. Toward this end, we have undertaken a multipronged research and development activity in which we are developing an integrated data-driven regional coupled modeling system.

**[12]**. This unpredictable nature can make it especially challenging for emergency responders, infrastructure managers, and power utilities to be able to prepare and react to these types of events when they occur. Predictive analytical methods could be used to help power utilities adapt to these types of storms, but there are uncertainties inherent in the predictability of convective storms that pose a challenge to the accurate prediction of storm-related outages.

**[13]**. This paper explores whether the many integrations with different versions of the model physics can be used to obtain more accurate and more reliable probability distributions for the model parameters. Some model parameters have a continuous range of possible values. Other parameters are categorical and act as switches between different parameterizations. In an evolutionary algorithm, the member configurations that contribute most to the quality of the ensemble are duplicated, while adding a small perturbation, at the expense of configurations that perform poorly.

**[14]**. Proper specification, which is determined by the forecast system set-up, is often required. Previous studies have investigated its relevance in various global and regional numerical weather prediction (NWP) systems; however, very few have explored it in tropical NWP systems. Here, we present and evaluate the structures of the background error covariance matrix for a tropical convective-scale NWP system. A total of 12 background error covariance matrices are modelled using differences between pairs of forecasts of different lengths but valid at the same time, based on the application of the vertical-first and horizontal-first transform order formulations on six permutations of the training data.

**[15].** In this paper, the scientific and technological progress of NWP in China since 1949 is summarized. The current status and recent progress of the domestically developed NWP system—GRAPES (Global/Regional Assimilation and PrEdiction System) are presented. Through independent research and development in the past 10 years, the operational GRAPES system has been established, which includes both regional and global deterministic and ensemble prediction models, with resolutions of 3–10 km for regional and 25–50 km for global forecasts.

**[16].** This paper implements the real time weather prediction system that can be used in number of applications like homes, industries, agriculture, stadiums etc. for predicting the weather information. The system utilizes a temperature and humidity sensor i.e. DHT11 and a light intensity sensor i.e. LDR. The sensed data from the sensors are uploaded to a ThingSpeak cloud server using NodeMCU and ESP8266-01 module. The data is also displayed on a customized HTML webpage for monitoring the real time values. A logistic regression model is used for setting up the machine learning environment. This model is trained using the pre- recorded values of sensor data.

**[17].** Air pollution has been a looming issue of the 21st century that has also significantly impacted the surrounding environment and societal health. Recently, previous studies have conducted extensive research on air pollution and air quality monitoring. Despite this, the fields of air pollution and air quality monitoring remain plagued with unsolved problems. In this study, the Pollution Weather Prediction System (PWP) is proposed to perform air pollution prediction for outdoor sites for various pollution parameters.

**[18]**. CASA Prediction System over Dallas Air pollution has been a looming issue of the 21st century that has also significantly impacted the surrounding environment and societal health. Recently, previous studies have conducted extensive research on air pollution and air quality monitoring. Despite this, the fields of air pollution and air quality monitoring remain plagued with unsolved problems. In this study, the Pollution Weather Prediction System (PWP) is proposed to perform air pollution prediction for outdoor sites for various pollution parameters.

**[19]**. Using 1 year of pre-operational service in 2017 and the Fire Weather Index (FWI), here we assess the capability of the system globally and analyse in detail three major events in Chile, Portugal and California. The analysis shows that the skill provided by the ensemble forecast system extends to more than 10 d when compared to the use of mean climate, making a case for extending the forecast range to the sub-seasonal to seasonal timescale. However, accurate FWI prediction does not translate into accuracy in the forecast of fire activity globally. Indeed, when all fires detected in 2017 are considered, including agricultural- and human-induced burning, high FWI values only occur in 50 % of the cases and are limited to the Boreal regions.

**[20]**. In this paper we presented MetNet, a neural weather model for precipitation forecasting. MetNet improves upon the current operational NWP system HRRR for up to 8 hours of lead time. Reaching beyond 8 hours will require ever larger input contexts, rigorous engineering and deeper neural networks. they evaluate the performance of MetNet at various precipitation thresholds and find that MetNet outperforms Numerical Weather Prediction at forecasts of up to 7 to 8 hours on the scale of the continental United States.

**[21]**. In this paper, they presented a technology to utilize machine learning techniques to provide weather forecasts. Machine learning technology can provide intelligent models, which are much simpler than traditional physical models. They are less resource-hungry and can easily be run on almost any computer including mobile devices. Our evaluation results show that these machine learning models can predict weather features accurately enough to compete with traditional models. We also utilize the historical data from surrounding areas to predict weather of a particular area. We show that it is more effective than considering only the area for which weather forecasting.

**[22]**. The activities of many primary sectors depend on the weather for production, e.g. farming. The climate is changing at a drastic rate nowadays, which makes the old weather prediction methods less effective and more hectic. To overcome these difficulties, the improved and reliable weather prediction methods are required. These predictions affect a nation's economy and the lives of people. To develop a weather forecasting system that can be used in remote areas is the main motivation of this work. The data analytics and machine learning algorithms, such as random forest classification, are used to predict weather conditions. In this paper, a low-cost and portable solution for weather prediction is devised.

**[23]**. They have tested the use of neural networks for forecasting the “weather” in a range of simple climate models with different complexity. For this we have used a deep convolutional encoder–decoder architecture that Scher (2018) developed for a very simple general circulation model without seasonal cycle. The network is trained on the model in order to forecast the model state 1 d ahead. This process is then iterated to obtain forecasts at longer lead times. they also performed “climate” runs, where the network is started with a random initial state from the climate model run and then creates a run of daily fields for several decades

**[24].** In this work they have performed experiments and compared data mining algorithms including Naive Bayes, KNN and Decision tree for weather forecast phenomena. The results portrayed that Decision tree more successful in classifying and modelling Data set it also proved its effectiveness in both classification and perdition. The behaviour of KNN algorithm was the weakest among the three algorithms. Naive Bayes which is a simple classifier based on Bayes theorem, is a simple classifier to apply and proves to be efficient in performance against the other two classifiers used as it gives nearly the same results of the Decision tree algorithm. Accuracy for Decision tree, KNN, Naïve biased algorithm are 97.45%, 77.34% and 97.45% respectively Root mean square error for Decision tree, KNN, Naïve biased algorithm are 0.148%, 0.365% and 0.143% respectively . correlation for Decision tree, KNN, Naïve biased algorithm are 0.972%, 0.744% and 0.972% respectively.

**[25]**. In this paper they have investigated the usage of different Arctic observation types in the ECMWF NWP system and the impact of assimilating them on the quality of shortand medium-range forecasts. Our investigation relied on both comprehensive numerical experimentation consisting of both OSEs and FSOI diagnostics. The results demonstrate the importance of both satellite and conventional data in the Arctic region in determining the initial conditions used for NWP. Improvements in the representation of snow in the forecast model are expected in the future, however, such as through the ongoing development of a multilayer snow model.

**[26].** Climate change is shaping extreme heat and rain. To what degree human activity has increased the risk of high impact events is of high public concern and still heavily debated. Recent studies attributed single extreme events to climate change by comparing climate model experiments where the influence of an external driver can be included or artificially suppressed. Many of these results however did not properly account for model errors in simulating the probabilities of extreme event occurrences. Here we show, exploiting advanced correction techniques from the weather forecasting field, that correcting properly for model probabilities alters the attributable risk of extreme events to climate change. This study illustrates the need to correct for this type of model error in order to provide trustworthy assessments of climate change impacts

**[27]**. Weather forecasting is usually solved through numerical weather prediction (NWP), which can sometimes lead to unsatisfactory performance due to inappropriate setting of the initial states. In this paper, we design a data-driven method augmented by an effective information fusion mechanism to learn from historical data that incorporates prior knowledge from NWP. A notable advantage of our proposed method is that it simultaneously implements single-value forecasting and uncertainty quantification, which we refer to as deep uncertainty quantification (DUQ). Efficient deep ensemble strategies are also explored to further improve performance. Experimental results demonstrate that the proposed NLE loss significantly improves generalization compared to mean squared error (MSE) loss and mean absolute error (MAE) loss. Compared with NWP, this approach significantly improves accuracy by 47.76%, which is a state-of-the-art result on this benchmark dataset.

**[28].** India is one of the worst flood-affected countries in the world, with the recent disaster in Kerala in August 2018 being a prime example. A good amount of work has been carried out by employing Internet of Things (IoT) and machine learning (ML) techniques in the past for flood occurrence based on rainfall, humidity, temperature, water flow, water level etc. In addition, a deep learning model is compared with other machine learning models (support vector machine (SVM), K-nearest neighbor (KNN) and Naïve Bayes) in terms of accuracy and error. The results indicate that the deep neural network can be efficiently used for flood forecasting with highest accuracy based on monsoon parameters only before flood occurrence.

**[29]** The activities of many primary sectors depend on the weather for production, e.g. farming. The climate is changing at a drastic rate nowadays, which makes the old weather prediction methods less effective and more hectic. To overcome these difficulties, the improved and reliable weather prediction methods are required. These predictions affect a nation's economy and the lives of people. To develop a weather forecasting system that can be used in remote areas is the main motivation of this work. The data analytics and machine learning algorithms, such as random forest classification, are used to predict weather conditions. In this paper, a low-cost and portable solution for weather prediction is devised.

**[30]**. This study builds a model that predicts the amounts of solar power generation using weather information provided by weather agencies. This study proposes a two-step modeling process that connects unannounced weather variables with announced weather forecasts. The empirical results show that this approach improves a base approach by wide margins, regardless of types of applied machine learning algorithms. The results also show that the random forest regression algorithm performs the best for this problem, achieving an R-squared value of 70.5% in the test data.

1. **Dataset description & Sample data**

A benchmark dataset for data-driven medium-range weather forecasting, a topic of high scientific interest for atmospheric and computer scientists alike and download for Kaggle dataset Using the Columns :precipitation, temp\_max, temp\_min ,wind and we are going to predict the weather condition column that contain drizzle ,rain, sun, snow and fog.

Dataset link: <https://drive.google.com/drive/folders/1v17jJXh6LlKCmmVtpqBcUgzktwwl9Nto>

1. **PROPOSED ALGORITHM WITH FLOWCHART**

Flow of XGBoost framework:

**Initialize the model:** The XGBoost algorithm starts by initializing the model with a default set of hyperparameters. This includes the number of trees in the ensemble, the learning rate, the maximum depth of each tree, and the regularization parameters.

**Split the data:** The next step is to split the data into training and validation sets. The training set is used to train the model, while the validation set is used to monitor the model's performance and prevent overfitting.

**Build the initial tree:** The XGBoost algorithm builds the first tree in the ensemble by minimizing the loss function. The loss function measures the difference between the predicted and actual values of the target variable.

**Calculate the residuals:** The residuals are the differences between the predicted and actual values of the target variable. The residuals are used to train the next tree in the ensemble.

**Build subsequent trees:** The XGBoost algorithm builds subsequent trees in the ensemble by minimizing the loss function, using the residuals from the previous tree as the target variable.

**Regularize the trees:** To prevent overfitting, the XGBoost algorithm applies regularization techniques to the trees. This includes L1 and L2 regularization, which penalize large coefficients and limit the complexity of the model.

**Combine the trees:** Once all the trees in the ensemble have been built, the XGBoost algorithm combines their predictions to produce a final prediction.

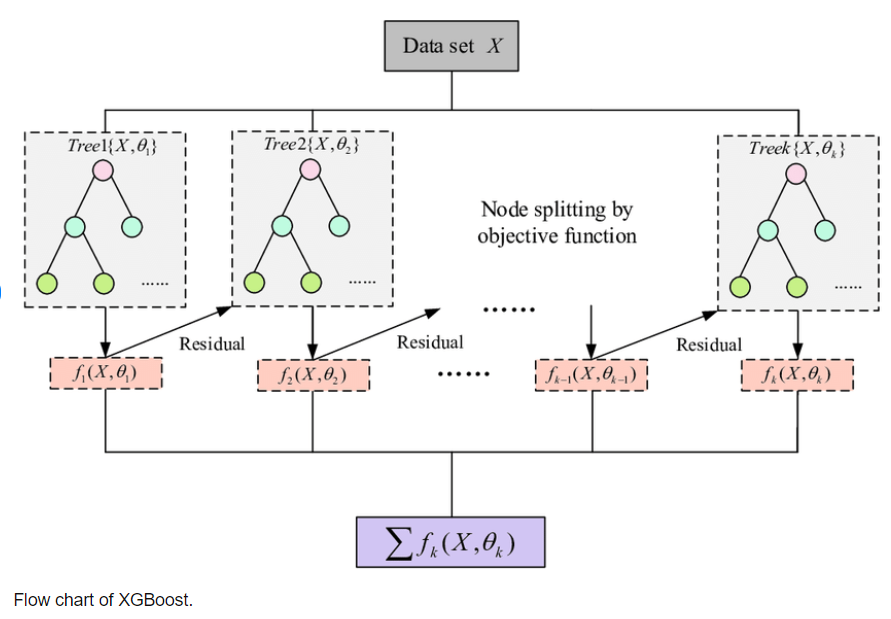
**Evaluate the model:** The XGBoost algorithm evaluates the model's performance on the validation set, using metrics such as the mean squared error or mean absolute error.

**Update the model:** Based on the performance on the validation set, the XGBoost algorithm updates the hyperparameters of the model, such as the learning rate, maximum depth, or regularization parameters. The algorithm then repeats the training process with the updated hyperparameters.

**Make predictions:** Once the model is trained, it can be used to make predictions on new data by passing the data through the ensemble of trees.

This is a simplified flowchart of the XGBoost algorithm, but it gives a good overview of the main steps involved in the training and evaluation process.

Flow chart of XGBoost framework:

****

1. **EXPERIMENTS RESULTS**

In this section, experiments are executed to evaluate the performance of proposed technique using Google colab. In our application, we enter the wind speed ,minimum temperature maximum temperature and precipitation and obtain weather as the output. We predict the weather using the KNN, SVM and XGBoost algorithm. We compared all the three algorithm.

As aresult we got accuracy of 83.06% for XGBoost ,77.42% for SVM and 75.00% for KNN algorithm. XGBoost algorithm for weather prediction, is looks good due to its high accuracy and efficiency. SVM and KNN can also be highly accurate, but may require more computational resources to achieve optimal performance.Overall, XGBoost has demonstrated strong performance for weather prediction tasks.

1. **COMPARATIVE STUDY / RESULTS AND DISCUSSION**

All three algorithms, Extreme Boosting (XGBoost), Support Vector Machine (SVM), and K Nearest Neighbors (KNN), are commonly used for classification problems, we used for weather prediction. Here's a comparison of these three algorithms based on their accuracy:

Extreme Boosting (XGBoost): XGBoost is an ensemble learning algorithm that combines multiple decision trees to improve prediction accuracy. It has been shown to be highly accurate for various classification problems, including weather prediction. XGBoost uses gradient boosting to minimize the loss function and tune model parameters to optimize the model. With careful hyperparameter tuning, XGBoost can achieve high accuracy on weather prediction tasks.

Support Vector Machine (SVM): SVM is a widely used machine learning algorithm for classification tasks. It works by finding a hyperplane that maximally separates the different classes of data points. SVM has been shown to be highly accurate for various classification problems, including weather prediction. However, SVM can be computationally expensive, especially when dealing with large datasets.

K-Nearest Neighbors (KNN): KNN is a simple but effective algorithm that works by finding the k nearest data points to a test point and classifying the test point based on the majority class of the nearest neighbors. KNN can be highly accurate for classification problems, including weather prediction. However, KNN can be sensitive to the choice of k, and can also be computationally expensive, especially when dealing with large datasets.

In general, the accuracy of these algorithms depends on the specific data and problem being addressed. However, XGBoost is often the preferred algorithm for classification problems, including weather prediction, due to its high accuracy and efficiency. SVM and KNN can also be highly accurate, but may require more computational resources to achieve optimal performance. As we got accuracy of 83.06% for XGBoost ,77.42% for SVM and 75.00% for KNN algorithm.

1. **CONCLUSION AND FUTURE WORK**

Weather prediction using XGBoost is an active area of research, and there are several potential future works that can be done to improve the accuracy of weather prediction models. Here are some possible avenues of research:

**Incorporating more data sources:** Weather prediction models rely on a wide range of data sources, including historical weather data, satellite imagery, and atmospheric models. Incorporating more data sources can help improve the accuracy of the model. For example, data from ground-based weather stations, ocean buoys, and aircraft can provide valuable information about local weather conditions.

**Tuning hyperparameters:** XGBoost models have several hyperparameters that can be tuned to optimize the model's performance. These include the learning rate, the number of trees in the ensemble, the maximum depth of each tree, and the regularization parameters. Finding the optimal combination of hyperparameters can improve the accuracy of the model.

**Ensemble learning:** Ensemble learning involves combining the predictions of multiple models to produce a final prediction. This can help reduce the impact of individual model errors and improve the overall accuracy of the prediction. XGBoost models are particularly well-suited for ensemble learning due to their ability to handle complex nonlinear relationships in the data.

**Feature engineering:** Feature engineering involves creating new features from the existing data that may be more predictive of the target variable. For example, creating features that capture the interaction between different weather variables (such as temperature and humidity) can help improve the accuracy of the model.

**Using deep learning:** Deep learning models, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have shown promising results in weather prediction tasks. These models can capture complex spatial and temporal patterns in the data, and may outperform XGBoost models in certain scenarios.

Overall, there are several potential avenues of research to improve weather prediction using XGBoost, and it is an exciting area of research with many potential applications

1. **REFERENCES**
2. Ali, M. F. M., Asklany, S. A., El-wahab, M. A., & Hassan, M. A. (2019). Data mining algorithms for weather forecast phenomena: comparative study. International journal of computer science and network security, 19(9), 76-81.
3. Hewage, P., Trovati, M., Pereira, E., & Behera, A. (2021). Deep learning-based effective fine-grained weather forecasting model. Pattern Analysis and Applications, 24(1), 343-366.
4. Wang, B., Lu, J., Yan, Z., Luo, H., Li, T., Zheng, Y., & Zhang, G. (2019, July). Deep uncertainty quantification: A machine learning approach for weather forecasting. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (pp. 2087-2095).
5. Jakaria, A. H. M., Hossain, M. M., & Rahman, M. A. (2020). Smart weather forecasting using machine learning: a case study in tennessee. arXiv preprint arXiv:2008.10789
6. Singh, N., Chaturvedi, S., & Akhter, S. (2019, March). Weather forecasting using machine learning algorithm. In 2019 International Conference on Signal Processing and Communication (ICSC) (pp. 171-174). IEEE.
7. Scher, S., & Messori, G. (2019). Weather and climate forecasting with neural networks: using general circulation models (GCMs) with different complexity as a study ground. Geoscientific Model Development, 12(7), 2797-2809.
8. Sønderby, C. K., Espeholt, L., Heek, J., Dehghani, M., Oliver, A., Salimans, T., ... & Kalchbrenner, N. (2020). Metnet: A neural weather model for precipitation forecasting. arXiv preprint arXiv:2003.12140.
9. Espeholt, L., Agrawal, S., Sønderby, C., Kumar, M., Heek, J., Bromberg, C., ... & Kalchbrenner, N. (2022). Deep learning for twelve hour precipitation forecasts. Nature communications, 13(1), 1-10.
10. Lawrence, H., Bormann, N., Sandu, I., Day, J., Farnan, J., & Bauer, P. (2019). Use and impact of Arctic observations in the ECMWF Numerical Weather Prediction system. Quarterly Journal of the Royal Meteorological Society, 145(725), 3432-3454.
11. Bellprat, O., Guemas, V., Doblas-Reyes, F., & Donat, M. G. (2019). Towards reliable extreme weather and climate event attribution. Nature communications, 10(1), 1-7.
12. G. Verma, P. Mittal and S. Farheen, "Real Time Weather Prediction System Using IOT and Machine Learning," 2020 6th International Conference on Signal Processing and Communication (ICSC), Noida, India, 2020, pp. 322-324
13. Eric P. James, Curtis R. Alexander, David C. Dowell, Stephen S. Weygandt, Stanley G. Benjamin, Geoffrey S. Manikin, John M. Brown, Joseph B. Olson, Ming Hu, Tatiana G. Smirnova, Terra Ladwig, Jaymes S. Kenyon, David D. Turner, The High-Resolution Rapid Refresh (HRRR): An Hourly Updating Convection-Allowing Forecast Model.
14. Pandya, S.; Ghayvat, H.; Sur, A.; Awais, M.; Kotecha, K.; Saxena, S.; Jassal, N.; Pingale, G. Pollution Weather Prediction System: Smart Outdoor Pollution Monitoring and Prediction for Healthy Breathing and Living. Sensors 2020, 20, 5448
15. Radhakrishnan, C., and V. Chandrasekar, 2020: CASA Prediction System over Dallas–Fort Worth Urban Network: Blending of Nowcasting and High-Resolution Numerical Weather Prediction Model. J. Atmos. Oceanic Technol., 37, 211–228
16. Hoteit, I., and Coauthors, 2021: Towards an End-to-End Analysis and Prediction System for Weather, Climate, and Marine Applications in the Red Sea.
17. Di Giuseppe, F., Vitolo, C., Krzeminski, B., Barnard, C., Maciel, P., and San-Miguel, J.: Fire Weather Index: the skill provided by the European Centre for Medium-Range Weather Forecasts ensemble prediction system, Nat. Hazards Earth Syst. Sci., 20, 2365–2378
18. Watson, P.L.; Koukoula, M.; Anagnostou, E. Influence of the Characteristics of Weather Information in a Thunderstorm-Related Power Outage Prediction System. Forecasting 2021, 3, 541-560.
19. David E. Gwyther, Shane R. Keating, Colette Kerry, Moninya Roughan, How does 4DVar data assimilation affect the vertical representation of mesoscale eddies? A case study with observing system simulation experiments (OSSEs) using ROMS v3.9, Geoscientific Model Development, 10.5194/gmd-16-157-2023
20. Houtekamer, P. L., B. He, D. Jacques, R. McTaggart-Cowan, L. Separovic, P. A. Vaillancourt, A. Zadra, and X. Deng, 2021: Use of a Genetic Algorithm to Optimize a Numerical Weather Prediction System. Mon. Wea. Rev., 149, 1089–1104
21. Shen, X., Wang, J., Li, Z. et al. Research and Operational Development of Numerical Weather Prediction in China. J Meteorol Res 34, 675–698 (2020).
22. M. G. Schultz, C. Betancourt, B. Gong, F. Kleinert, M. Langguth, L. H. Leufen, A. Mozaffari and S. Stadtler . Can deep learning beat numerical weather prediction? Bulletin of the American Meteorological Society, 102(2), E228-E245.
23. Suresh Sankaranarayanan, Malavika Prabhakar, Sreesta Satish, Prerna Jain, Anjali Ramprasad, Aiswarya Krishnan; Flood prediction based on weather parameters using deep learning. Journal of Water and Climate Change 1 December 2020; 11 (4): 1766–1783.
24. N. Singh, S. Chaturvedi and S. Akhter, "Weather Forecasting Using Machine Learning Algorithm," 2019 International Conference on Signal Processing and Communication (ICSC), NOIDA, India, 2019, pp. 171-174
25. Kim, S.-G.; Jung, J.-Y.; Sim, M.K. A Two-Step Approach to Solar Power Generation Prediction Based on Weather Data Using Machine Learning. Sustainability 2019, 11, 1501 .
26. Ravinesh C. Deo, A.A. Masrur Ahmed, David Casillas-Pérez, S. Ali Pourmousavi, Gary Segal, Yanshan Yu, Sancho Salcedo-Sanz, Cloud cover bias correction in numerical weather models for solar energy monitoring and forecasting systems with kernel ridge regression, Renewable Energy, Volume 203, 2023, Pages 113-130, ISSN 0960-1481 .
27. Sheng-Xiang Lv, Lin Wang, Multivariate wind speed forecasting based on multi-objective feature selection approach and hybrid deep learning model, Energy, Volume 263, Part E, 2023, 126100, ISSN 0360-5442 .
28. Rahman, A.-u.; Abbas, S.; Gollapalli, M.; Ahmed, R.; Aftab, S.; Ahmad, M.; Khan, M.A.; Mosavi, A. Rainfall Prediction System Using Machine Learning Fusion for Smart Cities. Sensors 2022, 22, 3504.
29. Shin, JY., Min, B. & Kim, K.R. High-resolution wind speed forecast system coupling numerical weather prediction and machine learning for agricultural studies — a case study from South Korea. Int J Biometeorol 66, 1429–1443 (2022).
30. Tulsi Pawan Fowdur, Rosun Mohammad Nassir-Ud-Diin Ibn Nazir, A real-time collaborative machine learning based weather forecasting system with multiple predictor locations,Array,Volume 14,2022,100153,ISSN 2590-0056 .
31. Huabing Ke, Sunling Gong, Jianjun He, Lei Zhang, Bin Cui, Yaqiang Wang, Jingyue Mo, Yike Zhou, Huan Zhang,Development and application of an automated air quality forecasting system based on machine learning,Science of The Total Environment,Volume 806, Part 3,2022,151204,ISSN 0048-9697 .

**Appendix**

import numpy as np

import pandas as pd

import os

for dirname, \_, filenames in os.walk('/kaggle/input'):

    for filename in filenames:

        print(os.path.join(dirname, filename))

import matplotlib.pyplot as plt

import seaborn as sns

import scipy

import re

import missingno as mso

from scipy import stats

from scipy.stats import ttest\_ind

from scipy.stats import pearsonr

from sklearn.preprocessing import StandardScaler,LabelEncoder

from sklearn.model\_selection import train\_test\_split

from sklearn.neighbors import KNeighborsClassifier

from sklearn.svm import SVC

from sklearn.ensemble import GradientBoostingClassifier

from xgboost import XGBClassifier

from sklearn.metrics import accuracy\_score,confusion\_matrix,classification\_report

data=pd.read\_csv("data/seattle-weather.csv")

data.head()

data.shape

countrain=len(data[data.weather=="rain"])

countsun=len(data[data.weather=="sun"])

countdrizzle=len(data[data.weather=="drizzle"])

countsnow=len(data[data.weather=="snow"])

countfog=len(data[data.weather=="fog"])

print("Percent of Rain:{:2f}%".format((countrain/(len(data.weather))\*100)))

print("Percent of Sun:{:2f}%".format((countsun/(len(data.weather))\*100)))

print("Percent of Drizzle:{:2f}%".format((countdrizzle/(len(data.weather))\*100)))

print("Percent of Snow:{:2f}%".format((countsnow/(len(data.weather))\*100)))

print("Percent of Fog:{:2f}%".format((countfog/(len(data.weather))\*100))

data[["precipitation","temp\_max","temp\_min","wind"]].describe()

sns.set(style="darkgrid")

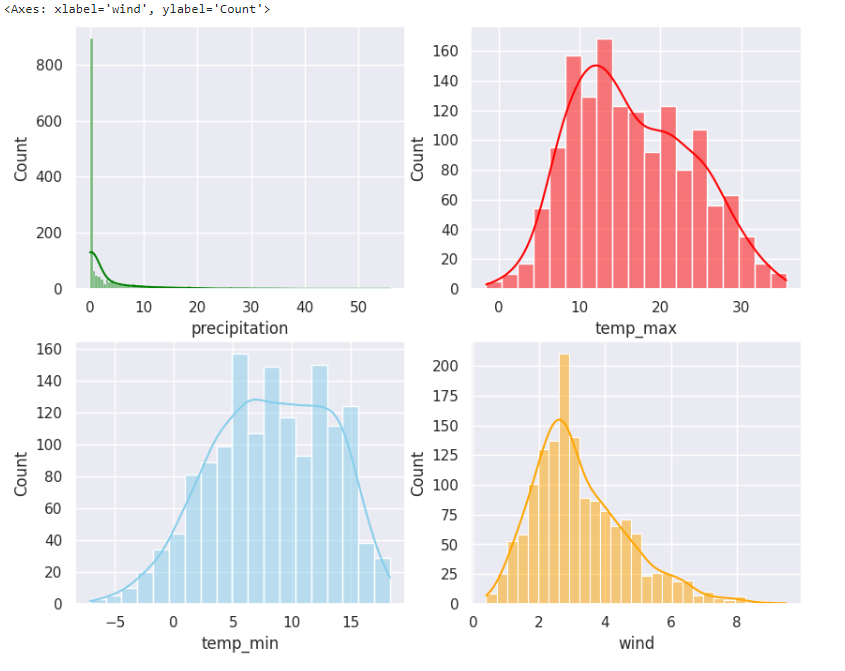
fig,axs=plt.subplots(2,2,figsize=(10,8))

sns.histplot(data=data,x="precipitation",kde=True,ax=axs[0,0],color='green')

sns.histplot(data=data,x="temp\_max",kde=True,ax=axs[0,1],color='red')

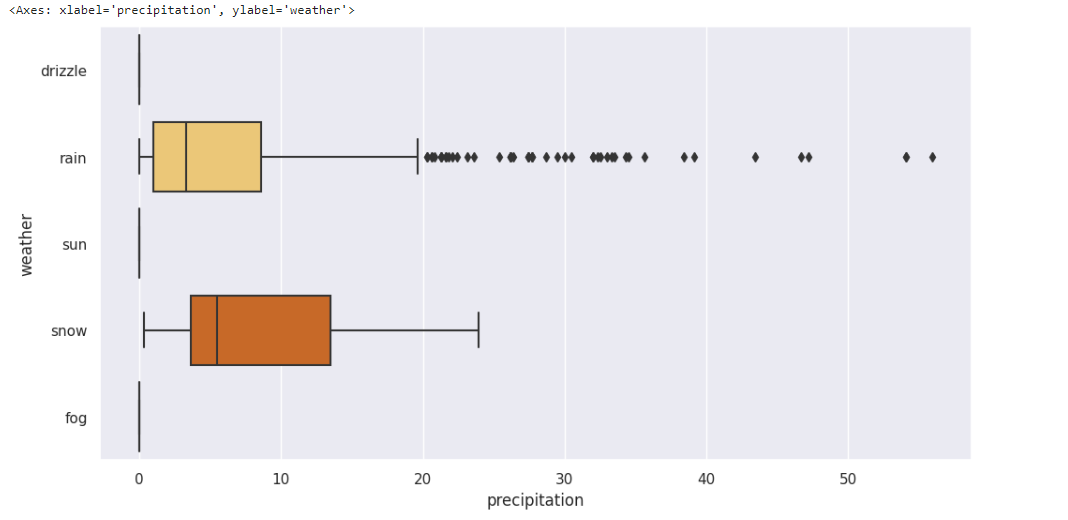
sns.histplot(data=data,x="temp\_min",kde=True,ax=axs[1,0],color='skyblue')

sns.histplot(data=data,x="wind",kde=True,ax=axs[1,1],color='orange')



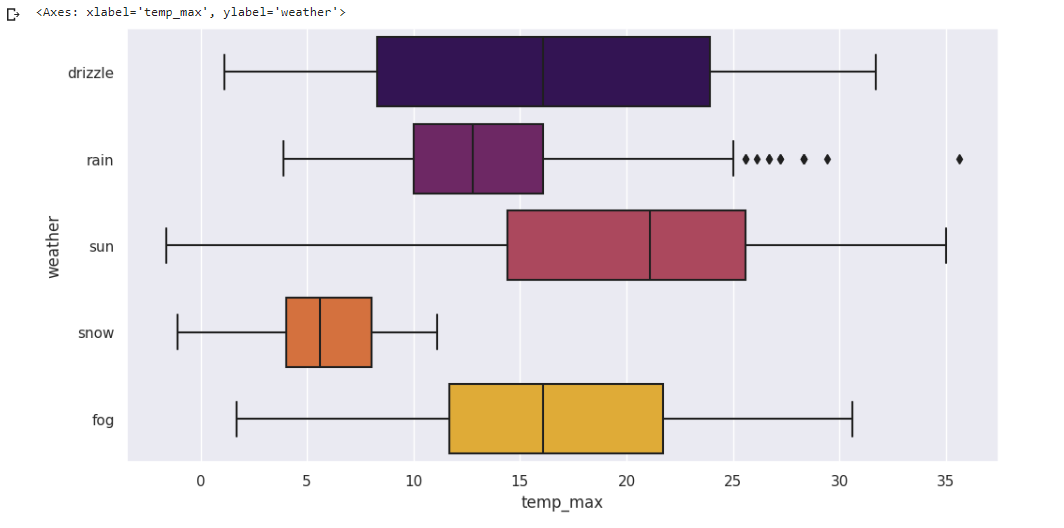
plt.figure(figsize=(12,6))

sns.boxplot(x="precipitation",y="weather",data= data ,palette="YlOrBr")



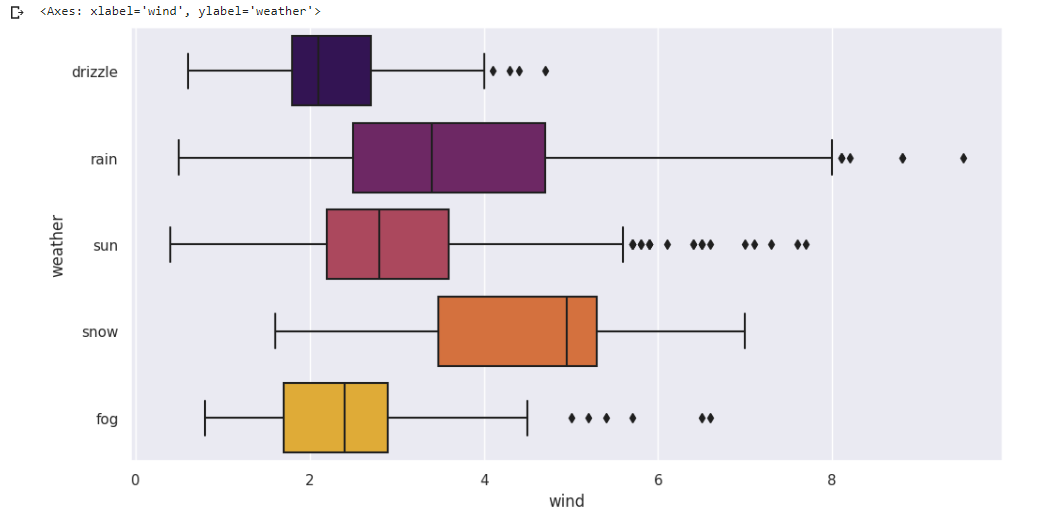
plt.figure(figsize=(12,6))

sns.boxplot(x="temp\_max",y="weather",data=data,palette="inferno")



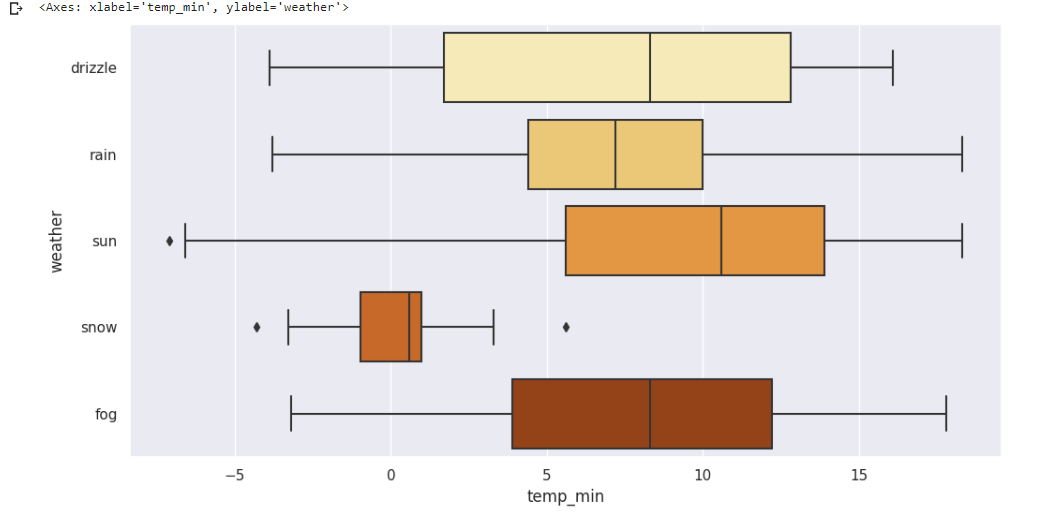
plt.figure(figsize=(12,6))

sns.boxplot(x="wind",y="weather",data=data,palette="inferno")



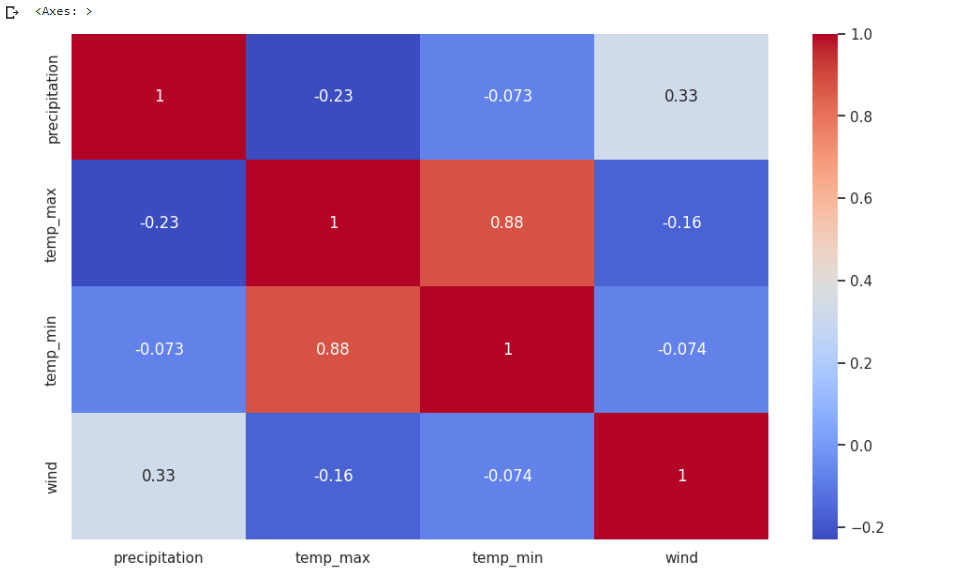
plt.figure(figsize=(12,6))

sns.boxplot(x="temp\_min",y="weather",data=data,palette="YlOrBr")



plt.figure(figsize=(12,7))

sns.heatmap(data.corr(),annot=True,cmap='coolwarm')



df=data.drop(["date"],axis=1)

Q1=df.quantile(0.25)

Q3=df.quantile(0.75)

IQR=Q3-Q1

df=df[~((df<(Q1-1.5\*IQR))|(df>(Q3+1.5\*IQR))).any(axis=1)]

df.precipitation=np.sqrt(df.precipitation)

df.wind=np.sqrt(df.wind)

sns.set(style="darkgrid")

fig,axs=plt.subplots(2,2,figsize=(10,8))

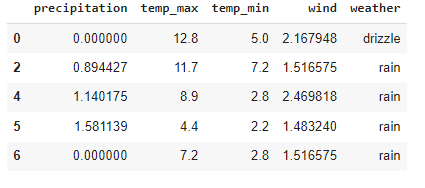
sns.histplot(data=df,x="precipitation",kde=True,ax=axs[0,0],color='green')

sns.histplot(data=df,x="temp\_max",kde=True,ax=axs[0,1],color='red')

sns.histplot(data=df,x="temp\_min",kde=True,ax=axs[1,0],color='skyblue')

sns.histplot(data=df,x="wind",kde=True,ax=axs[1,1],color='orange')

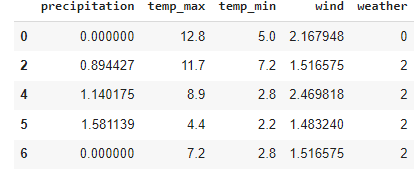
df.head()



lc=LabelEncoder()

df["weather"]=lc.fit\_transform(df["weather"])

df.head()



x=((df.loc[:,df.columns!="weather"]).astype(int)).values[:,0:]

y=df["weather"].values



df.weather.unique()

x\_train,x\_test,y\_train,y\_test=train\_test\_split(x,y,test\_size=0.1,random\_state=2)

knn=KNeighborsClassifier()

knn.fit(x\_train,y\_train)

print("KNN Accuracy:{:.2f}%".format(knn.score(x\_test,y\_test)\*100))



svm=SVC()

svm.fit(x\_train,y\_train)

print("SVM Accuracy:{:.2f}%".format(svm.score(x\_test,y\_test)\*100))



gbc=GradientBoostingClassifier(subsample=0.5,n\_estimators=450,max\_depth=5,max\_leaf\_nodes=25)

gbc.fit(x\_train,y\_train)

print("Gradient Boosting Accuracy:{:.2f}%".format(gbc.score(x\_test,y\_test)\*100))



import warnings

warnings.filterwarnings('ignore')

xgb=XGBClassifier()

xgb.fit(x\_train,y\_train)

print("XGB Accuracy:{:.2f}%".format(xgb.score(x\_test,y\_test)\*100))



input=[[1.140175,8.9,2.8,2.469818]]

ot=xgb.predict(input)

print("The weather is:")

if(ot==0):

    print("Drizzle")

elif(ot==1):

    print("Fog")

elif(ot==2):

    print("Rain")

elif(ot==3):

    print("snow")

else:

    print("Sun")



input=[[0,23.9,16.1,2.8]]

ot=knn.predict(input)

print("The weather is:")

if(ot==0):

    print("Drizzle")

elif(ot==1):

    print("Fog")

elif(ot==2):

    print("Rain")

elif(ot==3):

    print("snow")

else:

    print("Sun")



input=[[0.8,12.8,5.0,4.7]]

ot=xgb.predict(input)

print("The weather is:")

if(ot==0):

    print("Drizzle")

elif(ot==1):

    print("Fog")

elif(ot==2):

    print("Rain")

elif(ot==3):

    print("snow")

else:

    print("Sun")



input=[[0.8,11.7,7.2,2.3]]

ot=svm.predict(input)

print("The weather is:")

if(ot==0):

    print("Drizzle")

elif(ot==1):

    print("Fog")

elif(ot==2):

    print("Rain")

elif(ot==3):

    print("snow")

else:

    print("Sun")

