Predicting Next-day Rain in Australia

Muhammad Arkan Alireza, Pandya Athallah Erlambang, Salama A Mostafa

Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn Malaysia,
Parit Raja 86400, Batu Pahat, Johor, Malaysia
JI210015@siswa.uthm.edu.my, JI210024@siswa.uthm.edu.my,
salama@uthm.edu.my

Abstract. This study aims to predict rain in Australia with a machine learning classification approach. Precise and accurate rain prediction is very important for planning and management of water resources, flood warning, construction activities, aviation operations, transportation activities, agricultural jobs, and many others. By finding hidden patterns from available aspects of past meteorological data, data mining algorithms can efficiently predict rainfall. When testing the weather data on four different models, this study found that random forest is the best algorithm for predicting next-day rain while logistic regression is the least suited algorithm, while multilayer perceptron and decision tree performs well enough. For future work we encouraged others to try different algorithms such as LightGBM, Catboost, and XGBoost.

Keywords — rain, prediction, Australia, logistic regression, decision tree, random forest, multilayer perceptron

1 Introduction

Australia is a continent and a country made up of several states. The climate in Australia's eight states varies greatly. Most of Australia has four seasons namely summer, winter, autumn and spring. Rainfall in Australia varies widely, as the continent has a diverse climate, and northern Australia has a tropical climate. Then, the southwest and south coast have a subtropical climate (Climate of the World: Australia | weatheronline.co.uk, 2022). Western Australia is full of deserts, the climate is dry, very hot during the day but very cold at night. Finally, the east coast of Australia has a marine climate, so the area has fairly high rainfall throughout the year.

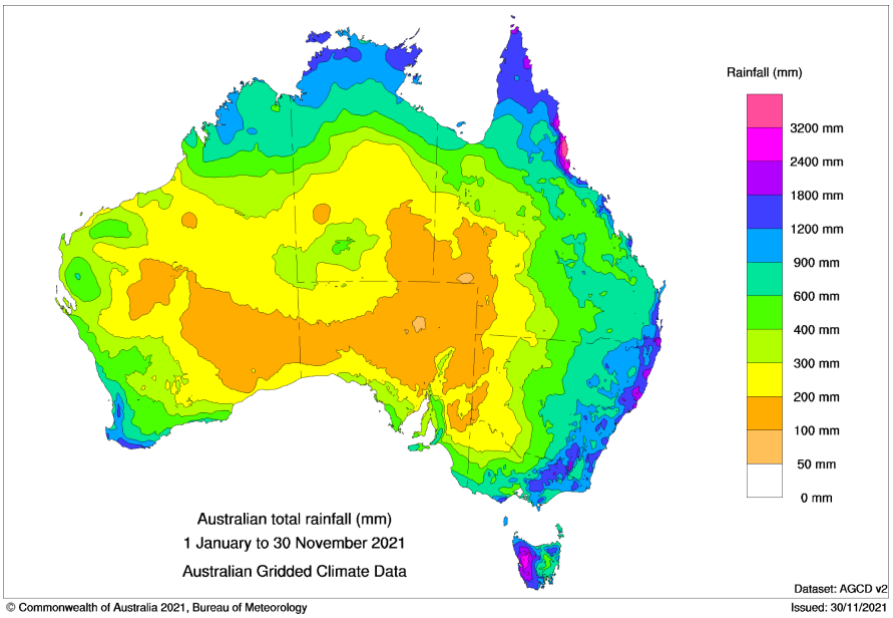

Fig 1. Australian total rainfall (mm) from 1 January to 30 November 2021

Accurate rain prediction is one of the most challenging and important tasks in the world today, and Australia is no exception because it has a diverse climate. What makes it challenging is that rain is dynamic in nature from climatic phenomena and random fluctuations involved in physical processes. Usually rain predictions are made for several time periods which include weekly, monthly and seasonal predictions. Precise and accurate rain prediction is very important for planning and management of water resources, flood warning, construction activities and aviation operations and others. This means that rain predictions could greatly affect economics (Thirumalai et al., 2017). In order for the results of rain prediction to be optimal, various complexities need to be addressed, such as weather statistical data which has many features including humidity, pressure, wind speed, pollutants, concentration, and others.

Data mining is the process of identifying interesting, surprising, or profitable structures in huge datasets (Hand, 2007). It brings together concepts, tools, and methodologies from computer science, machine learning, database technology, and other data analytics techniques. To perform our Rain Prediction, we propose four methods: Logistic Regression, Decision Tree, Random Forest, and Multi Layer Perceptron. We evaluate the performance of each algorithms by using five parameters : Accuracy, Precision, Recall, F1 Score, and ROC AUC.

2 Related Work

On the basis of monthly and seasonal timeframes, Singh (2017) constructed a model for forecasting Indian Summer Monsoon Rainfall. A time series dataset

spanning 1871 to 2014 was utilized to forecast. The dataset was divided into two parts: (1) training data from 1871 to 1960, and (2) test data from 1961 to 2014. The dynamic nature of monsoon rainfall was shown by statistical analysis, which could not be predicted efficiently using mathematical and statistical models. As a result, the authors of this study suggested three methods for this type of prediction: fuzzy sets, entropy, and artificial neural networks. To deal with the dynamic character of the ISMR, a forecasting model is constructed employing these three strategies. Fuzzy set theory is used in the suggested model to deal with inherited uncertainty in a dataset. In this paradigm, the entropy computational notion was updated and input was provided as a degree of membership in the entropy function. Fuzzy Information-Gain was the name given to the entropy function (FIG). The ANN was then used to defuzzify each fuzzified rule. Each fuzzy-FIG set's value was then used as an input into ANN. Because it combines fuzzy sets, entropy, and ANN, the suggested model was dubbed "Fuzzy-Entropy-Neuro Based Expert System for ISMR Forecasting." Standard Deviations (SDs), Correlation Coefficient (CC), Root Mean Square Error (RMSE), and Performance Parameter were used to evaluate the performance of the suggested model (PP). In comparison to other current models, the proposed model is effective and efficient, according to the findings.

Cramer et al. (2017) compared the predictive performance of the "Markov chain extended with rainfall prediction" method to the predictive performance of other widely used machine learning techniques such as Support Vector Regression, Genetic Programming, M5 Rules, M5 Model trees, Radial Basis Neural Networks, and k-Nearest Neighbours. Daily rainfall data was collected from 42 cities across two continents with a wide range of weather conditions. Twenty cities from Europe and 22 from the United States were chosen. The experiment was conducted on two continents for two reasons: first, to run the experiment in distinct climates with varying weather, and second, since the selected cities were geographically separated from one another. The final goal was to avoid biasing the experiment to a specific climate or geographic region. According to the findings, accumulated rainfall levels can produce better outcomes than forecasting based on daily rainfall data. Support Vector Regression, Radial Basis Functions, and Genetic Programming all performed well when using the accumulated data, however Radial Basis Functions outperformed the contemporary "Markov chain" approach. Each approach used the same parameters for all of the datasets, hence the best feasible set of parameters for all of the techniques was not guaranteed. The researchers discovered a link between predicted accuracy and meteorological parameters such as the volatile nature of rainfall, the amount of maximum rainfall, and the interquartile range of rainfall during the trial. Furthermore, no substantial differences in algorithm prediction error were found between cities on both continents (USA and Europe). The problem of rainfall data discontinuity was overcome with the use of accumulated rainfall quantities.

Pour et al. (2016) suggested a hybrid strategy that combines two methods to downscale daily rainfall: 1) Random Forest and 2) Support Vector Machine. RF was chosen for its classification robustness, and it was used to forecast whether it will

rain or not, whereas SVM was chosen for its ability to fit non-linear data, and it was used to predict the amount of rain that would fall if it did. The suggested model was tested by downscaling daily rainfall at three locations on Peninsular Malaysia's east coast: Dungun, Besut, and Kemaman. The Department of Irrigation and Drainage Malaysia provided daily rainfall time series data from 1961 to 2000. The National Centre for Environmental Prediction reanalysis dataset yielded a total of 26 climatic features, which were employed as predictors for downscaling the models. Various quality control operations were conducted to assess the homogeneity of rainfall time series. To reflect the issues, histograms were produced for the dataset, and the Student's t test was performed to find any variance in the means between two segments of the dataset, which were ultimately found to be homogeneous at all three sites. According to the findings, the hybrid technique can downscale rainfall with a Nash-Sutcliff efficiency of 0.90-0.93, which is significantly greater than that of RF and SVM models.

In Victoria, Australia, Bagirov et al. (2017) introduced Clusterwise Linear Regression as a technique for monthly rainfall prediction. The proposed CLR is a clustering and regression approach combined method. CLR incrementally collected subsets from the dataset, which could then be simply estimated one by one using a linear function. The data set for prediction was compiled from eight different weather stations between 1889 and 2014 and included five meteorological variables. Three weather stations were chosen from Victoria's east region, two from its central region, and three from its west region. The ultimate purpose of the geographically separated stations was to test the proposed model's performance in several areas with different atmospheres. Vapor Pressure and Solar Radiation were among the metrological variables employed as predictors. Evaporation, Minimum Temperature, and Maximum Temperature are all factors to consider. SVM Reg, ANNs, CLR with CR-EM, and MLR were all compared to this proposed approach. The model was built using training data for each weather station and each technique, and then tested using test data. The performance of the suggested technique was evaluated using four accuracy parameters: Mean Absolute Scaled Error, Mean Absolute Error, Root Mean Squared Error, and coefficient of efficiency, which were compared to actual and anticipated rainfall data. In most cases, the proposed strategy beat other prediction algorithms, according to the findings.

To support the planting calendar in Soreang, Gunawansyah et al. (2017) presented the Evolving Neural Network for rainfall forecast and anomaly detection. The data was gathered from the Departments of Agriculture and Water Resources between 1999 and 2013. To find the appropriate weights and biases, the suggested ENN uses Artificial Neural Networks and Genetic Algorithms. The proposed framework included several processes, beginning with the acquisition of raw data and continuing through the pre-processing phase, which included the steps of data integration, transformation, reduction, and cleaning. The data was divided into three scenarios: dry season from April to September, wet season from October to March, and entire data from January to December. Each scenario was further separated into training and test data, with training data being 9, 12, 14 years and testing data being

6, 3, 1 years. The suggested framework's learning process utilized integrated methodologies, and the results were then used for rainfall prediction and anomaly identification, with the ultimate result being the expected planting start time. In 2014, the first week of January, April, and October were chosen as the start dates for planting. According to the findings, an accuracy of 84.6 percent was obtained utilizing all data from 1999 to 2013, with 66.02 percent for the dry season and 79.7% for the rainy season.

Hernández et al. (2016) proposed a Deep Learning-based architecture for forecasting the next day's accumulated rainfall. Two approaches are used in the proposed architecture: the Autoencoder Network and the Multilayer Perceptron Network. The feature selection activity was conducted by an unsupervised network, while the classification and prediction duties were allocated to the Multilayer Perceptron Network. The data for the prediction was gathered from the Universidad Nacional de Colombia's Instituto de Estudios Ambientales (IDEA) in Manizales, Colombia. The data set covered the years 2002 to 2013 and included 47 meteorological variables. IDEA collected data from a meteorological station in the city's central location and stored it in an environmental DWH. Pre-processing was not required because ETL steps were performed on data. For the purposes of training, validation, and testing, 2952 data samples were divided into subsets, with 70 percent, 15 percent, and 15 percent, respectively. The data was then normalized to keep the values within the range of 0 to 1 for easier processing. The experiment's results were compared to those of other approaches, including a naive approach that predicts accumulated rainfall of t 1 for t, MLP with optimal parameters for training and validation sets, and several other published techniques. Measurement errors, such as Mean Square Error and Root Mean Square Error, were used to assess performance.

3 Methodology

In predicting next-day rain in Australia, this project uses the Cross-Industry Process for Data Mining (CRISP-DM) methodology. CRISP-DM stands for Cross Industry Standard Process for Data Mining and is a 1996 methodology created to shape Data Mining projects. It consists of 6 steps to conceive a Data Mining project and they can have cycle iterations according to developers’ needs. Those steps are Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment.

Fig. 2 CRISP-DM illustration

I. Business Understanding

The Business Understanding phase focuses on determining the project's goals and requirements. Apart from the third activity, the other three tasks in this phase are standard project management activities that apply to almost all projects:

1. Determine business objectives: Before defining business success criteria, you should "thoroughly grasp, from a business perspective, what the consumer genuinely wants to accomplish."
2. Examine the situation: Determine the availability of resources, project requirements, risk and contingency assessments, and a cost-benefit analysis.
3. Determine your data mining objectives: You should describe what success looks like from a technical data mining standpoint in addition to the business objectives.
4. Make a project plan: For each project phase, choose technology and tools and create thorough blueprints.

While many teams rush through this step, developing a solid business knowledge is like laying a solid foundation for a house — it's vitally necessary.

II. Data Understanding

The next step is to comprehend the data. It strengthens the foundation of Business Understanding by focusing on identifying, collecting, and analyzing data sets that can assist you in meeting project objectives. There are four tasks in this phase:

1. Collect preliminary data: Gather the information you'll need and, if necessary, import it into your analytic program.

2. Describe the information: Examine the data and make a list of its surface attributes, such as data type, number of records, and field names.
3. Investigate the data: Investigate the data in greater depth. It may be queried, visualized, and linkages between the data can be discovered.
4. Examine the data for accuracy: What is the state of the data? Is it clean or dirty? Any faults with quality should be documented.

III. Data Preparation

This phase, sometimes known as "data munging," is responsible for preparing the final data set(s) for modeling. It has five objectives:

1. Data to be chosen: Determine which data sets will be used and why they were included or excluded.
2. Data that is free of errors: This is frequently the most time-consuming task. You'll most likely fall victim to garbage-in, garbage-out if you don't have it. Erroneous values are frequently corrected, imputed, or removed during this task.
3. Construct information: Create new characteristics that will be beneficial. Calculate a person's BMI using their height and weight fields, for example.
4. Integrate information: Combine data from multiple sources to create new data sets.
5. Data should be formatted as follows: As needed, re-format data. You might, for example, convert string values containing integers to numeric values in order to conduct mathematical operations.

IV. Modeling

You'll most likely construct and evaluate a variety of models using a variety of modeling methodologies. There are four tasks in this phase:

1. Modeling techniques to use: Choose which algorithms to test (e.g. regression, neural net).
2. Create a test plan: You may need to divide the data into training, test, and validation sets depending on your modeling technique.
3. Create a model: This might just be a few lines of code like "reg = LinearRegression().fit(X, y)" as glamorous as it seems.
4. Examine the model: Multiple models are typically pitted against one another, and the data scientist must interpret the results based on domain knowledge, pre-defined success criteria, and test design.

Although the CRISP-DM recommends "iterating model construction and assessment until you strongly feel you have identified the best model(s)," in practice, teams should iterate until they identify a "good enough" model, then follow the CRISP-DM lifecycle to improve the model in subsequent iterations.

V. Evaluation

Unlike the Modeling phase's Assess Model job, which focuses on technical model evaluation, the Evaluation phase considers which model best

satisfies the business's needs and what to do next. There are three tasks in this phase:

1. Examine the outcomes: Do the models match the success criteria for a business? Which one(s) should we give the green light to for the company?
2. The review procedure is as follows: Examine the job that has been completed. Was there anything that you missed? Were all of the steps completed correctly? Summarize your results and make any necessary corrections.
3. Determine the following steps: Determine whether to move on with deployment, iterate further, or start new projects based on the results of the previous three activities.

VI. Deployment

A model isn't very useful unless the findings can be accessed by the consumer. This phase's complexity varies greatly. There are four tasks in this final phase:

1. Deployment strategy: Create and document a deployment strategy for the model.
2. Plan for monitoring and upkeep: To avoid problems throughout the operational phase (or post-project phase) of a model, create a detailed monitoring and maintenance strategy.
3. Make a final report: The project team creates a project summary, which may include a final presentation of data mining findings.
4. Project review: Conduct a project review to see what went well, what could have gone better, and how you can improve in the future.

The job of your organization might not finish there. CRISP-DM does not specify what to do after the project is completed (also known as "operations"). However, if the model is going into production, make sure you keep it in production. Constant monitoring and model adjustment are frequently required.

3.1 Dataset

The dataset used in this study is Rain in Australia from the Kaggle: Your Machine Learning and Data Science Community Repository. The dataset is taken from the site Rain in Australia which is data that contains daily weather observations from various Australian weather stations. Where in this dataset the context is predicting whether it will rain tomorrow or not by training a binary classification model on the RainTomorrow target variable. The target variable RainTomorrow means to indicate “Will it rain the next day? Yes or no". The Rain in Australia dataset consists of a rar file containing a dataset file with the name weatherAUS.csv. The form of the file format in the dataset is csv or comma separated values, which means the data in it is separated by commas. The dataset consists of 24 features and 142,193 records.

Table 1. Description of the features

Features	Description
Date	The date of observation
Location	The common name of the location of the weather station
MinTemp	The minimum temperature in degrees celsius
MaxTemp	The maximum temperature in degrees celsius
Rainfall	The amount of rainfall recorded for the day in mm
Evaporation	The so-called Class A pan evaporation (mm) in the 24 hours to 9am
Sunshine	The number of hours of bright sunshine in the day.
WindGustDir	The direction of the strongest wind gust in the 24 hours to midnight
WindGustSpeed	The speed (km/h) of the strongest wind gust in the 24 hours to midnight
WindDir9am	Direction of the wind at 9am
WindDir3pm	Direction of the wind at 3pm
WindSpeed9am	Wind speed (km/hr) averaged over 10 minutes prior to 9am
WindSpeed3pm	Wind speed (km/hr) averaged over 10 minutes prior to 3pm
Humidity9am	Humidity (percent) at 9am
Humidity3pm	Humidity (percent) at 3pm
Pressure9am	Atmospheric pressure (hpa) reduced to mean sea level at 9am
Pressure3pm	Atmospheric pressure (hpa) reduced to mean sea level at 3pm
Cloud9am	Fractions of sky obscured by clouds at 9am. This is measured in "oktas", which are a unit of eighths. It records how many eighths of the sky are obscured by clouds. A 0 measure indicates completely clear sky whilst an 8 indicates that it is completely overcast.
Cloud3pm	Fractions of sky obscured by clouds at 3pm. This is measured in "oktas", which are a unit of eighths. It records how many eighths of the sky are obscured by clouds. A 0 measure indicates completely clear sky whilst an 8 indicates that it is completely overcast.
Temp9am	Temperature (degrees C) at 9am

Temp3pm	Temperature (degrees C) at 3pm
RainToday	Boolean: 1 if precipitation (mm) in the 24 hours to 9am exceeds 1mm, otherwise 0
RiskMM	The amount of next day rain in mm. Used to create a response variable RainTomorrow. A kind of measure of the "risk".
RainTomorrow	Target variable. Prediction of rain tomorrow.

The dataset was last published on December 11, 2020 by Joe and Adam Young in Kaggle from the Bureau of Meteorology, Commonwealth of Australia. In addition, the dataset has a missing value ranging from 1% to 48% which is the largest on several attributes. The way that can be done to handle missing values is to delete the missing value tuples or fill them with constant values based on the average of the values in the data.

Table 2. Attributes of the features

No.	Features	Data Type	Amount of Missing Values	Percentage of the Dataset
1	*Date*	*chr*	*0 Records*	0%
2	*Location*	*numeric*	*0 Records*	0%
3	*MinTemp*	*numeric*	*637 Records*	>0%
4	*MaxTemp*	*numeric*	*322 Records*	>0%
5	*Rainfall*	*numeric*	*1406 Records*	1%
6	*Evaporation*	*numeric*	*60843 Records*	43%
7	*Sunshine*	*numeric*	*67816 Records*	48%
8	*WindGustDir*	*chr*	*9330 Records*	7%
9	*WindGustSpeed*	*numeric*	*9270 Records*	7%
10	*WindDir9am*	*chr*	*10013 Records*	7%
11	*WindDir3pm*	*chr*	*3778 Records*	3%
12	*WindSpeed9am*	*numeric*	*1348 Records*	1%
13	*WindSpeed3pm*	*numeric*	*2630 Records*	2%

14	*Humidity9am*	*numeric*	*1774 Records*	1%
15	*Humidity3pm*	*numeric*	*3610 Records*	3%
16	*Pressure9am*	*numeric*	*14014 Records*	10%
17	*Pressure3pm*	*numeric*	*13891 Records*	10%
18	*Cloud9am*	*numeric*	*53567 Records*	38%
19	*Cloud3pm*	*numeric*	*57094 Records*	40%
20	*Temp9am*	*numeric*	*904 Records*	1%
21	*Temp3pm*	*numeric*	*2726 Records*	2%
22	*RainToday*	*chr*	*1406 Records*	1%
23	*RiskMM*	*numeric*	*0 Records*	0%
24	*RainTomorrow*	*chr*	*0 Records*	0%

3.2 Algorithms

1. Logistic regression (LR)

   The logistic model (or logit model) is used in statistics to model the probability of a specific class or event, such as pass/fail, win/lose, alive/dead, or healthy/sick, existing. This can be used to represent a variety of occurrences, such as determining whether an image contains a cat, dog, lion, or other animal. Each detected object in the image would be assigned a probability ranging from 0 to 1, with a total of one.

   Generalized Linear Models are a bigger class of algorithms that includes logistic regression (glm). Nelder and Wedderburn created this model in 1972 as a way of applying linear regression to issues that were not immediately suitable for it. In fact, they created a class of models (linear regression, ANOVA, Poisson Regression, and so on) that included logistic regression as an exception.

   The fundamental equation of generalized linear model is:

   $$g(E(y)) = \alpha + \beta x1 + \gamma x2$$

   Here, g() is the link function, E(y) is the expectation of the target variable and $\alpha + \beta x1 + \gamma x2$ is the linear predictor (α,β,γ to be predicted). The role of link function is to ‘link’ the expectation of y to linear predictor.

2. Decision tree (DT)

   The Decision Tree method is part of the supervised learning algorithm family. The decision tree approach, unlike other supervised learning algorithms, may also be utilized to solve regression and classification issues.

   By learning simple decision rules inferred from past data, the purpose of employing a Decision Tree is to develop a training model that can be used to predict the class or value of the target variable (training data).

   We start from the root of the tree when using Decision Trees to forecast a class label for a record. The values of the root attribute and the record's attribute are compared. We follow the branch that corresponds to that value and jump to the next node based on the comparison.

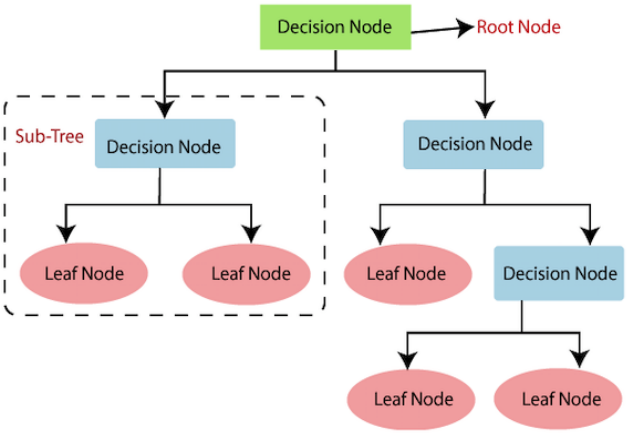

Fig. 3 Decision tree

3. Random forest (RF)

   As the name implies, a random forest is made up of a huge number of individual decision trees that work together as an ensemble. Each tree in the random forest produces a class prediction, and the class with the most votes becomes the prediction of our model.

   The wisdom of crowds is the basic principle behind random forest, and it's a simple yet effective one. The reason the random forest model works so well is that it consists of a huge number of largely uncorrelated models (trees) that work together to outperform any of the individual constituent models.

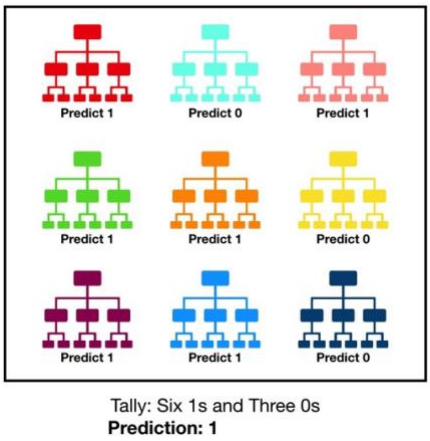

Fig. 4 Random forest

4. Multilayer perceptron (MLP)

   A feedforward artificial neural network called a multilayer perceptron (MLP) is a type of feedforward artificial neural network (ANN). The name MLP is ambiguous; it can be used to refer to any feedforward ANN, or it can refer to networks made up of many layers of perceptrons (with threshold activation). Multilayer perceptrons, especially those with a single hidden layer, are commonly referred to as "vanilla" neural networks.

   There are at least three levels of nodes in an MLP: an input layer, a hidden layer, and an output layer. Each node, with the exception of the input nodes, is a neuron with a nonlinear activation function. Backpropagation is a supervised learning technique used by MLP during training. MLP is distinguished from a linear perceptron by its numerous layers and non-linear activation. It can tell the difference between data that isn't linearly separable.

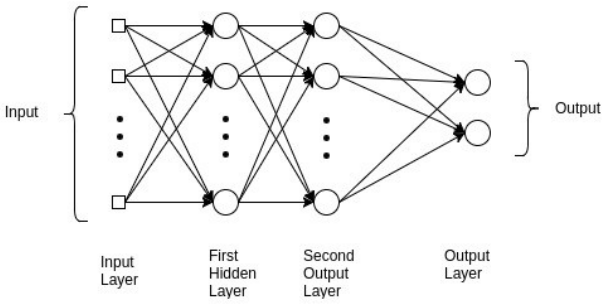

Fig. 5 Multilayer perceptron

3.3 Evaluation Metrics

The observations that are successfully anticipated and hence shown in green are true positives and true negatives. We aim to keep false positives and negatives to a minimum, therefore they're highlighted in red. These terms are a little perplexing. So let's go through each term one by one and make sure we grasp it well.

- True Positives (TP) are accurately predicted positive values, indicating that the value of the actual class and the value of the projected class are both yes. For example, if the actual class value indicates that this passenger survived and the anticipated class also suggests that this passenger survived.
- True Negatives (TN) - These are correctly predicted negative values, implying that the value of the actual class is zero and the value of the predicted class is zero as well. For example, if the real class reports that this passenger did not survive and the anticipated class reports the same.
- False Positives (FP) are when the actual class is not the same as the projected class. For example, if the actual class indicates that this passenger died, but the forecast class indicates that this passenger would live.
- False Negatives (FN) are situations in which the real class is yes but the expected class is no. For example, if the passenger's actual class value reveals that he or she survived while the anticipated class suggests that the passenger will die.

1. Accuracy

   The simplest intuitive performance metric is accuracy, which is just the ratio of properly predicted observations to all observations. One would believe that if our model is accurate, it is the best. Yes, accuracy is a useful statistic, but only when the datasets are symmetric and the values of false positives and false negatives are almost equal. As a result, other parameters must be considered while evaluating the performance of your model.

   Accuracy = TP+TN/TP+FP+FN+TN

2. Precision

   The ratio of accurately predicted positive observations to total expected positive observations is known as precision. The question that this measure answers is how many of the passengers who are identified as having survived actually did. The low false positive rate is related to high precision.

   Precision = TP/TP+FP

3. Recall

   Recall is defined as the proportion of accurately predicted positive observations to all observations in the class. How many of the passengers who genuinely survived were labeled, according to the question?

Recall = TP/TP+FN

4. F1 Score

   The weighted average of Precision and Recall is the F1 Score. As a result, this score considers both false positives and false negatives. Although it is not as intuitive as accuracy, F1 is frequently more useful than accuracy, especially if the class distribution is unequal. When false positives and false negatives have equivalent costs, accuracy works well. It's best to look at both Precision and Recall if the cost of false positives and false negatives is considerably different.

   F1 Score = 2*(Recall * Precision) / (Recall + Precision)

5. ROC AUC

   The Area Under the Curve (AUC) - ROC curve is a performance statistic for classification issues at various threshold levels. AUC represents the degree or measure of separability, whereas ROC is a probability curve. It indicates how well the model can distinguish between classes. The AUC indicates how well the model predicts 0 classes as 0 and 1 courses as 1. The higher the AUC, the better the model predicts 0 classes as 0 and 1 classes as 1. By analogy, the higher the AUC, the better the model distinguishes between people who have the condition and those who do not.

   The ROC curve is plotted with TPR against the FPR where TPR is on the y-axis and FPR is on the x-axis.

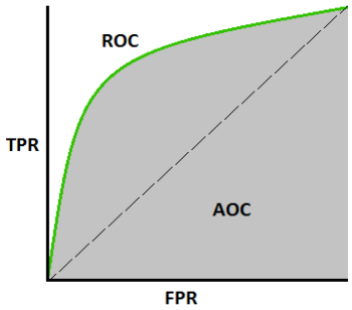

Fig. 5 AUC - ROC Curve

4 Results and Discussion

Table 3. Experimental results

Data split (%)	Algorithm	Accuracy	Precision	Recall	F1 Score	ROC AUC
30-70	LR	0.79541	0.80343	0.84022	0.82141	0.78931

	DT	0.82525	0.85480	0.82869	0.84154	0.82478
	RF	**0.89119**	**0.91011**	**0.89399**	**0.90197**	**0.89081**
	MLP	0.88426	0.89997	0.89251	0.89622	0.88313
40-60	LR	0.79580	0.80342	0.84104	0.82180	0.78965
	DT	0.84446	0.87730	0.83961	0.85804	0.84512
	RF	**0.90422**	**0.92348**	**0.90380**	**0.91354**	**0.90427**
	MLP	0.88698	0.90258	0.89469	0.89862	0.88593
50-50	LR	0.79554	0.80389	0.83961	0.82136	0.78955
	DT	0.85118	0.88147	0.84824	0.86453	0.85158
	RF	**0.91450**	**0.93614**	**0.90930**	**0.92253**	**0.91520**
	MLP	0.88507	0.89971	0.89442	0.89705	0.88380
60-40	LR	0.79586	0.80419	0.83952	0.82148	0.78997
	DT	0.86560	0.89798	0.85714	0.87709	0.86674
	RF	**0.92107**	**0.94359**	**0.91354**	**0.92832**	**0.92209**
	MLP	0.88861	0.90184	0.89872	0.90028	0.88725
70-30	LR	0.79523	0.80455	0.83760	0.82074	0.78949
	DT	0.85410	0.88285	0.85242	0.86737	0.85433
	RF	**0.92659**	**0.95053**	**0.91654**	**0.93323**	**0.92795**
	MLP	0.88954	0.90510	0.89665	0.90085	0.88858

In the 30-70% split, the overall best algorithm is random forest with an accuracy, precision, recall, F1 score and ROC AUC score in the area of 0.89-0.91. While the overall worst algorithm is logistic regression with an accuracy, precision, recall, F1 score and ROC AUC score in the area of 0.78-0.80. While multilayer perceptron is second best after random forest and decision tree third best before logistic regression.

In the 40-60% split, again the overall best algorithm is random forest with an accuracy, precision, recall, F1 score and ROC AUC score in the area of 0.90-0.92. While the overall worst algorithm is again logistic regression with an accuracy, precision, recall, F1 score and ROC AUC score in the area of 0.78-0.84. While multilayer perceptron is second best after random forest and decision tree third best before logistic regression.

In the 50-50% split, again the overall best algorithm is random forest with an accuracy, precision, recall, F1 score and ROC AUC score in the area of 0.91-0.93. While the overall worst algorithm is again logistic regression with an accuracy, precision, recall, F1 score and ROC AUC score in the area of 0.78-0.83. While multilayer perceptron is second best after random forest and decision tree third best before logistic regression.

In the 60-40% split, again the overall best algorithm is random forest with an accuracy, precision, recall, F1 score and ROC AUC score in the area of 0.91-0.94. While the overall worst algorithm is again logistic regression with an accuracy, precision, recall, F1 score and ROC AUC score in the area of 0.78-0.83. While multilayer perceptron is second best after random forest and decision tree third best before logistic regression.

In the 70-30% split, again the overall best algorithm is random forest with an accuracy, precision, recall, F1 score and ROC AUC score in the area of 0.91-0.95. While the overall worst algorithm is again logistic regression with an accuracy, precision, recall, F1 score and ROC AUC score in the area of 0.78-0.83. While multilayer perceptron is second best after random forest and decision tree third best before logistic regression.

Random forest is the best algorithm in this case because of its impressive versatility, whether you have a regression or classification task, random forest is an applicable model for your needs. It can handle binary features, categorical features, and numerical features. There is very little pre-processing that needs to be done. The data does not need to be rescaled or transformed. They are also great with high dimensionality, random forests are great with high dimensional data since we are working with subsets of data. They are also robust to outliers and non-linear data, random forest handles outliers by essentially binning them. It is also indifferent to non-linear features. They also can handle unbalanced data, it has methods for balancing error in class population unbalanced data sets. Random forest tries to minimize the overall error rate, so when we have an unbalanced data set, the larger class will get a low error rate while the smaller class will have a larger error rate. Finally they have low bias and moderate variance, each decision tree has a high variance, but low bias. But because we average all the trees in random forest, we are averaging the variance as well so that we have a low bias and moderate variance model.

5 Conclusions and Future Work

In conclusion, the most suited algorithm for this study is random forest. Specifically in the 70-30% data split where it reached an astonishing accuracy of 92%, precision of 95%, recall of 92%, F1 score of 93%, and ROC AUC of 92%. The algorithm proved to be the best in all data splits and evaluation metrics because of its impressive versatility, great with high dimensionality, robust to outliers and non-linear data, good in handling unbalanced data, and low bias with moderate variance. While the least suited algorithm for this study is logistic regression, with the algorithm proven to be the worst in all data splits and evaluation metrics. Specifically in the 30-70% data split where it reached a low accuracy of 79%, precision of 80%, recall of 84%, F1 score of 82%, and ROC AUC of 78%.

In future work, we encourage others to try more algorithms on the dataset such as LightGBM, Catboost, XGBoost, etc. Also, we can improve the performance of the

model by using recursive feature elimination, k-fold cross validation, and hyperparameter optimization using GridSearchCV.

Acknowledgement. This research is supported by Universiti Tun Hussein Onn Malaysia.

References

1. Young, J. (2020). Rain in Australia. Kaggle.com. https://www.kaggle.com/jsphyg/weather-dataset-rattle-package
2. Daily Weather Observations. (2022). Bom.gov.au. http://www.bom.gov.au/climate/dwo/
3. Climate Data Online - Map search. (2022). Bom.gov.au. http://www.bom.gov.au/climate/data/
4. Notes to accompany Daily Weather Observations. (2022). Bom.gov.au. http://www.bom.gov.au/climate/dwo/IDCJDW0000.shtml
5. Thirumalai, C., Harsha, K. S., Deepak, M. L., & Krishna, K. C. (2017, May). Heuristic prediction of rainfall using machine learning techniques. In 2017 International Conference on Trends in Electronics and Informatics (ICEI) (pp. 1114-1117). IEEE.
6. Bagirov, A. M., Mahmood, A., & Barton, A. (2017). Prediction of monthly rainfall in Victoria, Australia: Clusterwise linear regression approach. Atmospheric research, 188, 20-29.
7. Bagirov, A. M., & Mahmood, A. (2018). A comparative assessment of models to predict monthly rainfall in Australia. Water resources management, 32(5), 1777-1794.
8. Shabib Aftab, M. A., Hameed, N., Bashir, M. S., Ali, I., & Nawaz, Z. (2018). Rainfall prediction in lahore city using data mining techniques. International journal of advanced computer science and applications, 9(4).
9. Darji, M. P., Dabhi, V. K., & Prajapati, H. B. (2015, March). Rainfall forecasting using neural network: A survey. In 2015 international conference on advances in computer engineering and applications (pp. 706-713). IEEE.
10. Singh, P. (2018). Indian summer monsoon rainfall (ISMR) forecasting using time series data: a fuzzy-entropy-neuro based expert system. Geoscience Frontiers, 9(4), 1243-1257.
11. Cramer, S., Kampouridis, M., Freitas, A. A., & Alexandridis, A. K. (2017). An extensive evaluation of seven machine learning methods for rainfall prediction in weather derivatives. Expert Systems with Applications, 85, 169-181.
12. Pour, S. H., Shahid, S., & Chung, E. S. (2016). A hybrid model for statistical downscaling of daily rainfall. Procedia Engineering, 154, 1424-1430.
13. Liong, T. H. (2017, May). Prediction and anomaly detection of rainfall using evolving neural network to support planting calender in soreang

(Bandung). In 2017 5th International Conference on Information and Communication Technology (ICoIC7) (pp. 1-6). IEEE.

14. Hernández, E., Sanchez-Anguix, V., Julian, V., Palanca, J., & Duque, N. (2016, April). Rainfall prediction: A deep learning approach. In International Conference on Hybrid Artificial Intelligence Systems (pp. 151-162). Springer, Cham.
15. Climate of the World: Australia | weatheronline.co.uk. (2022). Weatheronline.co.uk; Weather.https://www.weatheronline.co.uk/reports/climate/Australia.htm#:~:text=The%20northern%20section%20of%20Australia,and%20cool%2C%20sometimes%20rainy%20winters.
16. Hand, D. J. (2007). Principles of Data Mining. Drug Safety, 30(7), 621–622.