Haoyang Tan
CS 472, CRN: 32094
Prof. Thien Huu Nguyen

Final Project Report
Date: 6/11/2023

i. Abstract:

This essay focuses on predicting rain tomorrow based on historical weather observations in Australia. Accurate weather forecasts are crucial for various sectors, and this study aims to compare the performance of two classification algorithms, logistic regression and K-nearest neighbors (KNN), in predicting rain. The provided dataset is preprocessed to handle non-numerical features, missing data, and outliers. The logistic regression algorithm models the relationship between input variables and the probability of rain, while KNN classifies new data points based on their similarity to historical data. Experiments are conducted with parameter tuning to optimize the models' accuracy. The results show that both classifiers achieve similar accuracy rates, with logistic regression performing slightly better. Logistic regression also demonstrates faster training time compared to KNN. The study concludes that with appropriate preprocessing and parameter selection, the models can achieve acceptable accuracy, but further improvements are possible with optimized data processing methods and feature selection.

ii. Introduction:

The problem I am trying to solve is predicting whether it will rain tomorrow based on historical weather observations. This is important because accurate weather forecasts play a crucial role in various sectors such as agriculture, transportation, and outdoor events planning. By accurately predicting rain, we can help individuals and organizations make informed decisions and take appropriate actions to mitigate the potential impact of rain.

To solve this problem, I plan to use two classification algorithms: logistic regression and K-nearest neighbors (KNN). By training and evaluating both logistic regression and KNN models on the provided dataset of daily weather observations from Australia, we can determine which algorithm performs better in predicting rain tomorrow. The selected model can then be used to make accurate predictions on new, unseen data, helping people and organizations make informed decisions based on the likelihood of rain.

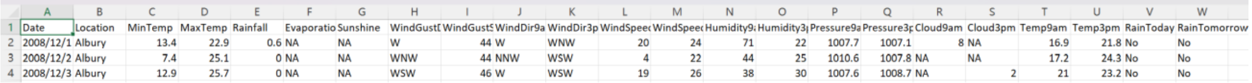
iii. Background:

Logistic regression is a commonly used algorithm for binary classification problems, where we predict a yes/no outcome. It models the relationship between the input variables (weather observations) and the probability of rain tomorrow.

K-nearest neighbors is another popular classification algorithm that can effectively predict the weather. It works by classifying new data points based on their similarity to existing data points. In this case, KNN would compare the current weather observations to historical data and classify whether it is likely to rain tomorrow based on the nearest neighbors.

iv. Methods:

In this section, we will include how to preprocess the data, the feature space we have decided to explore, and the classifiers we will use.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	Date	Location	MinTemp	MaxTemp	Rainfall	Evaporatio	Sunshine	WindGustD	WindGustS	WindDir9a	WindDir3p	WindSpeed	WindSpeed	Humidity9a	Humidity3p	Pressure9a	Pressure3p	Cloud9am	Cloud3pm	Temp9am	Temp3pm	RainToday	RainTomorrow
2	2008/12/1	Albury	13.4	22.9	0.6	NA	NA	W	44	W	WNW	20	24	71	22	1007.7	1007.1	8	NA	16.9	21.8	No	No
3	2008/12/2	Albury	7.4	25.1	0	NA	NA	WNW	44	NNW	WSW	4	22	44	25	1010.6	1007.8	NA	NA	17.2	24.3	No	No
4	2008/12/3	Albury	12.9	25.7	0	NA	NA	WSW	46	W	WSW	19	26	38	30	1007.6	1008.7	NA	2	21	23.2	No	No

As shown in the figure, our data consists of 22 features and one target variable. Features such as Date, Location, WindGustDir, WindDir9am, and WindDir3pm are given as strings, while RainToday and RainTomorrow have two options to choose from. Since these features are not in numerical form, we need to preprocess them and convert them into types that the program can recognize. Let's analyze these features one by one.

For the Date feature, as its distribution is widely dispersed and can lead to overfitting, and it doesn't have a significant impact on the final result, we won't consider this feature and set all its elements to 0.

Regarding the Location feature, since we are analyzing overall precipitation results for Australia, it doesn't have much relevance to the final outcome. Therefore, we won't consider this feature either and set all its elements to 0.

For the WindGustDir, WindDir9am, and WindDir3pm features, their values represent wind directions, and there are 16 possible directions. We will assign the corresponding numerical values from 1 to 16 based on the wind direction table.

The processing of RainToday and RainTomorrow is straightforward. We set No as -1 and Yes as 1.

For missing data, we will replace them with random values already existing in the same column.

To ensure the accuracy of our experiments, we will split the original data into two sets. One set will be the training data, which will comprise 70% of the original data, and the remaining 30% will be used as the testing set.

Classifiers:

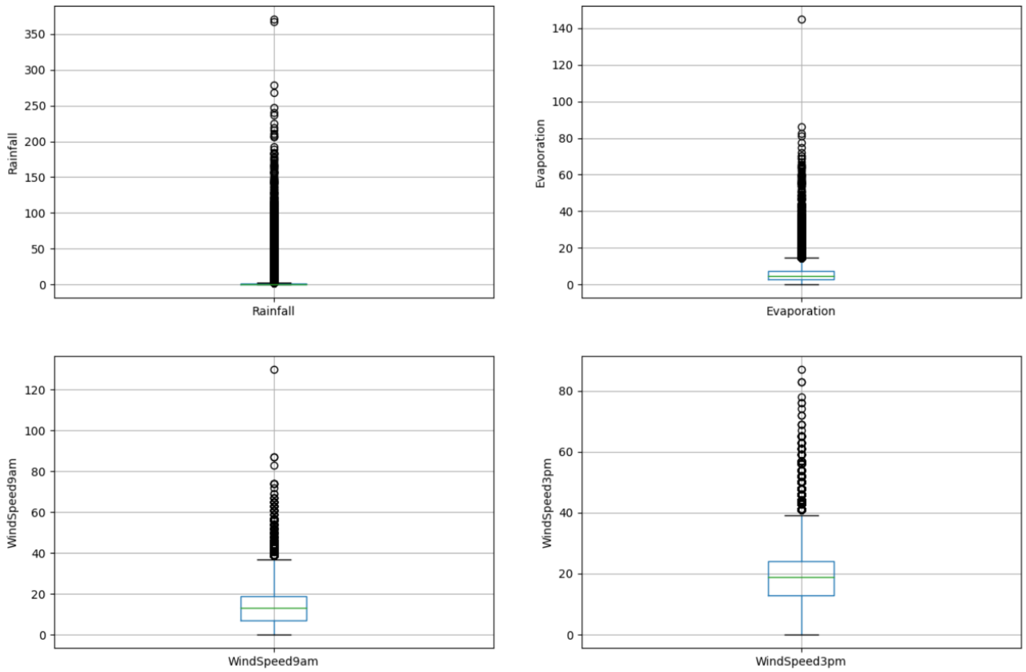
Due to the small size of the dataset and the limited number of features, I have decided to use relatively simple classifiers to avoid overfitting. The two classifiers I have chosen are:

- Logistic Regression: Although it is called regression, it is actually a classification model commonly used for binary classification. Logistic Regression is favored in the industry due to its simplicity, parallelizability, and strong interpretability. The essence of logistic regression is to assume that the data follows a certain distribution and then use maximum likelihood estimation to estimate the parameters.
- KNN Algorithm: Given a training dataset, for a new input instance, we find the K nearest instances in the training dataset and classify the input instance into the majority class among those K instances. This is similar to the idea of "majority rules" in real life.

v. Experiments:

In this section, we will provide a detailed explanation of the experiments conducted, including further data processing, parameter tuning, and result comparisons.

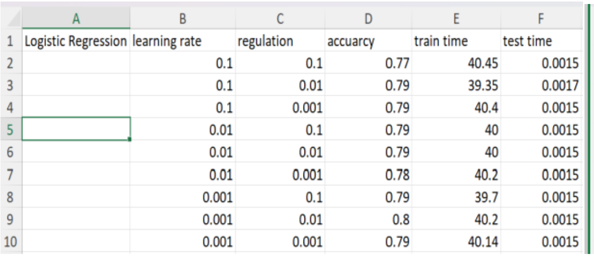
First, we applied the logistic regression method to solve the problem. However, we observed significant fluctuations in the accuracy after each iteration. We suspected that the presence of outliers in the original data might be causing these results. To investigate this, we examined the distribution of the original data.



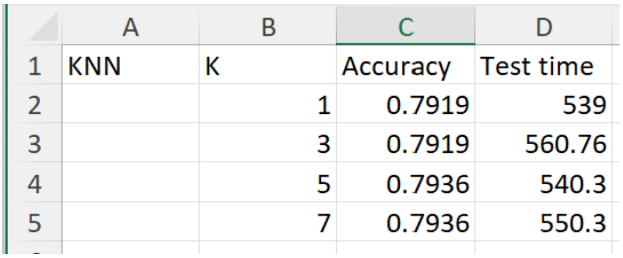
As seen in the figure, our assumption was correct, and there were indeed many outliers in the original data. Therefore, we changed the approach for handling missing data. For missing values, we computed the average of the current column and filled in the missing values accordingly.

Next, we applied two methods to solve the data. For each model, we attempted to adjust the parameters to achieve the optimal values while recording the accuracy, training time, and testing time for each run.

For logistic regression, we made assumptions about the learning rate and regularization weight to approximate the highest accuracy of the model. For KNN, we assumed different values for K {1, 3, 5, 7} to approximate the highest accuracy of the model.



	A	B	C	D	E	F
1	Logistic Regression	learning rate	regulation	accuarcy	train time	test time
2		0.1	0.1	0.77	40.45	0.0015
3		0.1	0.01	0.79	39.35	0.0017
4		0.1	0.001	0.79	40.4	0.0015
5		0.01	0.1	0.79	40	0.0015
6		0.01	0.01	0.79	40	0.0015
7		0.01	0.001	0.78	40.2	0.0015
8		0.001	0.1	0.79	39.7	0.0015
9		0.001	0.01	0.8	40.2	0.0015
10		0.001	0.001	0.79	40.14	0.0015



	A	B	C	D
1	KNN	K	Accuracy	Test time
2		1	0.7919	539
3		3	0.7919	560.76
4		5	0.7936	540.3
5		7	0.7936	550.3

Accuracy Analysis:

We can observe that the accuracy of logistic regression fluctuates between 0.77 and 0.80. As the learning rate increases, the accuracy also increases. In KNN, the highest accuracy is 0.7936, and the lowest is 0.7919. Changing the value of K does not significantly impact the accuracy. Therefore, in terms of accuracy, both classifiers perform similarly. However, logistic regression still achieves higher accuracy when the parameters are appropriately chosen.

Time Analysis:

Most of the time in logistic regression is spent on training the model. The training time does not vary significantly with different parameters. This could be because the model reaches convergence in a similar time frame, leading to the completion of training and exiting the program. On the other hand, KNN does not require training data as it only needs to be stored in an array. However, its drawback is that it takes a long time for testing. For each test sample, it needs to calculate the distance to every sample in the training set, resulting in a significant time cost during testing. Considering the time cost alone, logistic regression is clearly superior to the KNN algorithm.

vi. Conclusion:

In this work, I compared and analyzed the performance of two algorithms in predicting whether it will rain tomorrow in Australia. The results indicate that, with appropriate parameters, the performance stabilizes at an accuracy rate of 80%, which is an acceptable level of accuracy. An important aspect of this task is the handling of the original data, and the experiments demonstrate the necessity of preprocessing.

However, unfortunately, there are still shortcomings in the data processing, and I believe that the reason for reaching the accuracy bottleneck is that the data processing methods and feature selection did not reach their optimal points. My conclusion is that there is further room for development in this research.

vii. References:

prashant111. “Logistic Regression Classifier Tutorial.” *Kaggle*, 13 Mar. 2020, www.kaggle.com/code/prashant111/logistic-regression-classifier-tutorial/input.