Tanesha Subramaniam

COMPSCIX415.2-013

28 March 2021

Abstract

For this project, the primary research objective was to determine if I am able to predict if it will rain the next day in Australia using evaporation, humidity, pressure, amount of cloud cover and temperature. My hypothesis was that I can accurately predict if it will rain the next day using evaporation, humidity, pressure, amount of cloud cover and temperature. I implemented 5 different classifier algorithms to compare their performance and obtain the best accuracy score. Based on the results, Gradient Boosting Classifier produced the highest accuracy score. However, the accuracy score was not high enough to be considered an accurate prediction. Additionally, the feature importance plot indicated a different set of variables that were significant in predicting if it will rain the next day. Therefore, I concluded that I cannot accurately predict if it will rain the next day in Australia using evaporation, humidity, pressure, amount of cloud cover and temperature.

## BACKGROUND

In this section, I will be covering the research objective, research questions, hypothesis and background of the dataset.

*Research Objective and Questions.* The dataset was posted on Kaggle with the intention of predicting if it will rain the next day in Australia using a particular day's weather condition. The

dataset contains daily measurements of temperature, rainfall, wind, humidity, pressure, evaporation and cloud from several Australian weather stations. Given these variables, the objective of my project is to determine if I am able to predict if it will rain the next day using evaporation, humidity, pressure, amount of cloud cover and temperature. Since I am not knowledgeable about Australia's weather, I would also like to learn about the weather conditions in different cities. With that, my research questions are as follows:

1. Can I accurately predict if it will rain the next day using evaporation, humidity, pressure, amount of cloud cover and temperature?
2. What are the important factors in predicting if it will rain the next day?
3. Which locations in Australia get the most rain?
4. Is it more likely to rain the next day if it rained on a particular day?
5. Which locations in Australia are the hottest?

*Hypothesis.* For the first research question, I hypothesize that I can accurately predict if it will rain the next day using evaporation, humidity, pressure, amount of cloud cover and temperature. I consider an accuracy score greater than 90% an accurate prediction. It will likely rain if the evaporation and humidity are high, pressure and temperature are low and the sky has an overcast (i.e high amount of cloud cover).

Here's a quick summary of how rain is formed. First, clouds are formed through evaporation of water or ice from Earth's surface. When evaporation happens, water in the form of water vapor rises into the atmosphere. When there is a lot of water vapor in the atmosphere, the humidity increases. Water vapor turns into clouds when it cools and condenses, and then turns into water

droplets. This usually happens when the atmospheric pressure is low. As more water condenses, the droplets grow and once it gets too heavy to stay suspended in the cloud, these droplets fall to Earth as rain ("What Makes It Rain?").

*Dataset Background.* The dataset has about 145,000 daily weather observations from over 10 years. The description of each column are as follows:

1. **Date:** Date of observation.
2. **Location:** Name of the location of the weather station.
3. **MinTemp:** Minimum temperature in degrees celsius.
4. **MaxTemp:** Maximum temperature in degrees celsius.
5. **Rainfall:** Amount of rainfall recorded for the day in mm.
6. **Evaporation:** Evaporation (mm) in the 24 hours.
7. **Sunshine:** Number of hours of bright sunshine in the day.
8. **WindGustDir:** Direction of the strongest wind gust in the 24 hours to midnight.
9. **WindGustSpeed:** Speed (km/h) of the strongest wind gust in the 24 hours to midnight.
10. **WindDir9am:** Direction of the wind at 9am.
11. **WindDir3pm:** Direction of the wind at 3pm.
12. **WindSpeed9am:** Wind speed (km/hr) averaged over 10 minutes prior to 9am.
13. **WindSpeed3pm:** Wind speed (km/hr) averaged over 10 minutes prior to 3pm.
14. **Humidity9am:** Humidity (percent) at 9am.
15. **Humidity3pm:** Humidity (percent) at 3pm.
16. **Pressure9am:** Atmospheric pressure (hpa) reduced to mean sea level at 9am.

17. **Pressure3pm:** Atmospheric pressure (hpa) reduced to mean sea level at 3pm.
18. **Cloud9am:** Fraction of sky obscured by cloud at 9am, 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 a completely clear sky whilst an 8 indicates that it is completely overcast.
19. **Cloud3pm:** Fraction of sky obscured by cloud at 3pm.
20. **Temp9am:** Temperature (degrees C) at 9am.
21. **Temp3pm:** Temperature (degrees C) at 3pm.
22. **RainToday:** A boolean variable for if it rained on a particular day. It's yes if precipitation (mm) in the 24 hours to 9am exceeds 1mm, otherwise it will be no. In layman's term, it will be yes if it rained and no if it didn't rain.
23. **RainTomorrow:** A boolean variable for if it rained the next day. It's yes if it rained and no if it didn't rain.

Note that since the variables of interest are recorded twice a day, I selected data from 3pm of a particular day to predict if it will rain the next day. Specifically, the variables of interest are Evaporation, Humidity3pm, Pressure3pm, Cloud3pm and Temp3pm and the target variable is RainTomorrow.

## METHODS

To predict the target variable, I used 5 different classifier algorithms as I would like to compare their performances and obtain the best accuracy score. The chosen 5 classifier algorithms are

Decision Trees, Random Forest, Gradient Boosting, k-NN and Naïve Bayes'. The high level steps for all implementations are as follows:

1. Clean and prepare the data.
2. Perform feature selection to determine significant variables in predicting the target variable.
3. Continue to clean and prepare the data, if necessary.
4. Implement algorithm.
5. Attempt to improve the model's accuracy.

*Data Cleaning and Preparation.* For Decision Trees, Random Forest, Gradient Boosting and k-NN, I dropped missing values and outliers as these algorithms are sensitive to extreme values and noisy data. Naïve Bayes' is typically not sensitive to missing values and outliers, but I implemented Gaussian Naïve Bayes'. For Gaussian Naïve Bayes', the data needs to assume a Gaussian distribution, therefore outliers need to be dropped. After dropping values, I scaled the predictor features using StandardScalar as I didn't want features with a higher range value to dominate features with a lower range value in the model. For Tree Classifiers (i.e Decision Trees, Random Forest and Gradient Boosting), I balanced the data using the SMOTENC package, since the target variable is imbalanced. I didn't balance the data for k-NN and Naïve Bayes'.

*Performance Improvement.* For Decision Trees, Random Forest, Gradient Boosting and k-NN, there are several parameters to tune for the model. Therefore, I implemented Grid Search Cross Validation (GridSearchCV) to determine the best parameter values that will optimize the model. For Naïve Bayes', since there are only a couple of parameters, I attempted to improve the model using insights from the feature selection process.

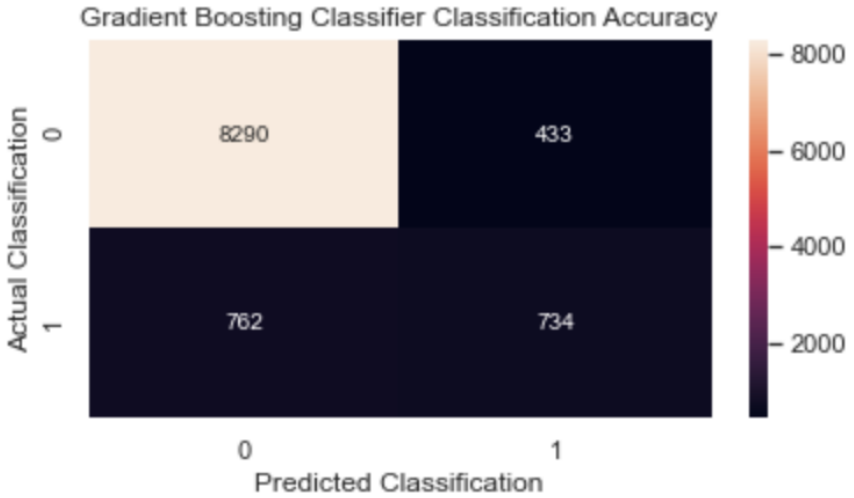
## RESULTS

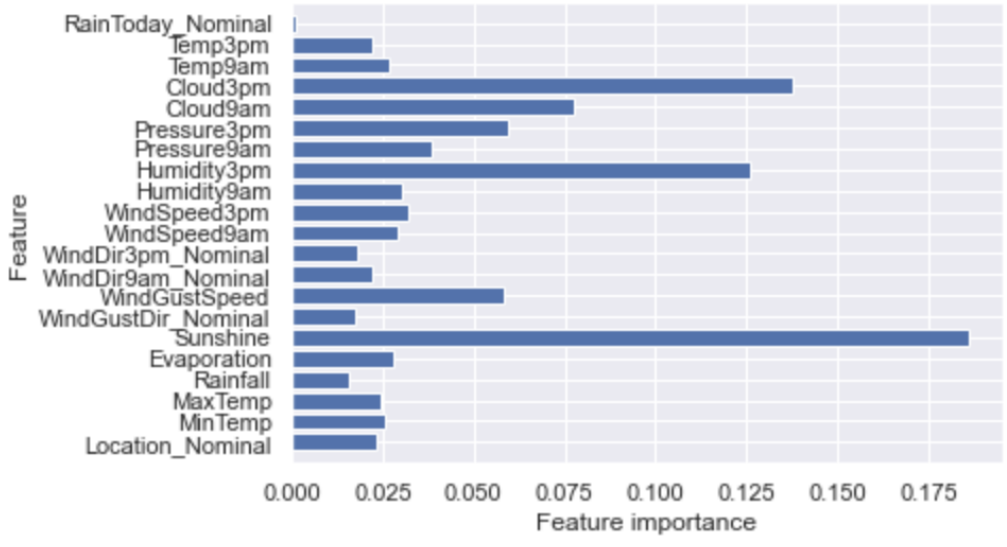
In this section, I will present the best performing model and highlight the accuracy scores for other classifier models. I will also share a couple of descriptive statistics that helped me answer the last 3 research questions.

*Best Model.* The best performing model in predicting target variable, RainTomorrow was the Gradient Boosting model. The parameter values were set to n_estimators = 100, max_depth = 15, learning_rate = 0.075 and max_features =8. Using these parameter values, the model achieved a 93.74% accuracy score on the training set and 88.32% on the test set. Based on the confusion matrix, 762 records were incorrectly classified as class 0, 433 records were incorrectly classified as class 1, 8290 records were correctly classified as class 0 and 734 records were correctly classified as class 1. The feature importance plot indicated that the most important features in predicting the target variable were Sunshine, Cloud3pm and Humidity3pm.

```
#best performing Gradient Boosting model
gb = GradientBoostingClassifier(n_estimators = 100, max_depth = 15, learning_rate = 0.075, max_features = 8, random_sta
gb.fit(X_oversample, np.ravel(y_oversample, order = 'C'))
print("Accuracy on training set: {:.3f}".format(gb.score(X_oversample, np.ravel(y_oversample, order = 'C'))))
print("Accuracy on test set: {:.3f}".format(gb.score(X_test, y_test)))

#cross validation on training set
cv_scores_training = cross_val_score(gb, X_oversample, np.ravel(y_oversample,order='C'), cv=10, n_jobs = -1)

#print each cv score (accuracy) and average them
print(cv_scores_training)
print("CV accuracy on training set:", np.mean(cv_scores_training))

#cross validation on test set
cv_scores_test = cross_val_score(gb, X_test, np.ravel(y_test,order='C'), cv=10)

#print each cv score (accuracy) and average them
print(cv_scores_test)
print("CV accuracy on test set:", np.mean(cv_scores_test))
```

```
Accuracy on training set: 1.000
Accuracy on test set: 0.883
[0.79537639 0.86033626 0.9650363  0.96560948 0.96484524 0.96427207
 0.96197937 0.96847535 0.96694057 0.96158991]
CV accuracy on training set: 0.937446093436753
[0.88454012 0.88747554 0.8816047  0.88356164 0.89334638 0.87279843
 0.88356164 0.87964775 0.88356164 0.88148874]
CV accuracy on test set: 0.8831586583890931
```



*Other Models.* The best performing models for other classifiers are as follows:

1. Random Classifier used default parameter values. Accuracy score on the training set was 93.04% and accuracy score on the test set was 88.29%. The feature importance plot indicated the most important features were Sunshine, Cloud3pm and Humidity3pm.
2. k-NN used all quantitative variables except for Rainfall. Accuracy score on the training set was 87.87% and accuracy score on the test set was 87.79%
3. Gaussian Naïve Bayes' used Pressure3pm and Humidity3pm as predictor features. Accuracy score on the test set was 86.87%.
4. Decision Tree Classifier used criterion = entropy, max_features = 10 and max_depth = 30 as parameter values. Accuracy score on the training set was 85.97% and accuracy score on the test set was 82.66%. The feature importance plot indicated the most important features were Sunshine and Cloud3pm.

*Descriptive Statistics.* Based on the numerical summary of rainfall grouped by location, we can see that Cairns, Darwin and CoffsHarbour have the highest mean and max values. The screenshot below only shows 10 locations with the highest means.

| Location | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| **Cairns** | 2988.0 | 5.742035 | 18.280975 | 0.0 | 0.0 | 0.0 | 2.600 | 278.4 |
| **Darwin** | 3193.0 | 5.092452 | 16.450148 | 0.0 | 0.0 | 0.0 | 1.800 | 367.6 |
| **CoffsHarbour** | 2953.0 | 5.061497 | 17.444480 | 0.0 | 0.0 | 0.0 | 2.000 | 371.0 |
| **GoldCoast** | 2980.0 | 3.769396 | 13.054984 | 0.0 | 0.0 | 0.0 | 1.200 | 183.4 |
| **Wollongong** | 2982.0 | 3.594903 | 11.897181 | 0.0 | 0.0 | 0.0 | 1.000 | 192.0 |
| **Williamtown** | 2553.0 | 3.591108 | 11.757872 | 0.0 | 0.0 | 0.0 | 1.600 | 225.0 |
| **Townsville** | 3033.0 | 3.485592 | 14.985589 | 0.0 | 0.0 | 0.0 | 0.200 | 236.8 |
| **NorahHead** | 2929.0 | 3.387299 | 9.387137 | 0.0 | 0.0 | 0.0 | 1.600 | 126.4 |
| **Sydney** | 3337.0 | 3.324543 | 9.887184 | 0.0 | 0.0 | 0.0 | 1.400 | 119.4 |
| **MountGinini** | 2907.0 | 3.292260 | 8.970322 | 0.0 | 0.0 | 0.0 | 1.800 | 107.6 |

Based on the contingency table below, we can see that if it rained on a particular day, it will likely not rain the next day. RainToday is the rows and if we look at the “No” row, we can see that “No” for RainTomorrow has the highest frequency for that row.

| | NA | No | Yes | rowtotal |
|---|---|---|---|---|
| **NA** | 1855 | 730 | 676 | 3261 |
| **No** | 987 | 92728 | 16604 | 110319 |
| **Yes** | 425 | 16858 | 14597 | 31880 |
| **coltotal** | 3267 | 110316 | 31877 | 145460 |

Based on the pivot table below, we can see that Katherine and Darwin have the highest average maximum temperature and average minimum temperature. The pivot table below was sorted by descending order on MaxTemp, followed by descending order on MinTemp and it only shows the top 10 locations.

| Location | MaxTemp | MinTemp |
|---|---|---|
| **Katherine** | 34.935436 | 20.553564 |
| **Darwin** | 32.540977 | 23.209305 |
| **Uluru** | 30.383195 | 14.466688 |
| **Cairns** | 29.558849 | 21.220467 |
| **Townsville** | 29.367160 | 20.417874 |
| **AliceSprings** | 29.248420 | 13.142284 |
| **Moree** | 26.950548 | 12.905853 |
| **Woomera** | 26.596707 | 13.363727 |
| **Brisbane** | 26.448380 | 16.423807 |
| **PearceRAAF** | 26.051238 | 12.303850 |

## DISCUSSION AND CONCLUSION

*Discussion.* Recall in the previous section, I stated that the best classifier model is the Gradient Boosting Classifier. I would like to discuss the different parameters I tuned to improve the accuracy of the model. First, the max_depth was increased from the default value of 3 to 15.

The maximum depth limits the number of nodes in a tree. The default value appeared to be too shallow of this dataset. I believe this is expected as the dataset is very large. Next, the learning_rate was reduced from 1.0 to 0.075. Learning rate shrinks the contribution of each tree. Setting a lower value means each tree contributes less and we will need more trees in the ensemble to fit the training data. This will lead to a lower overall variance ("Gradient Boosting Decision Tree Algorithm Explained."). Finally, I significantly reduced the max_features from the default value which is the number of predictor features in the dataset (i.e 21 features) to 8 features. The max_features is the number of features to consider when looking for the best split. It is possible that many of the features were redundant and degraded the model's performance. There were several challenges I faced during the implementation. First, the data was unclean. There were about 145,000 rows in the dataset. However, more than half of the rows had at least one null value. Even though I was left with enough rows to implement the algorithms, having more clean data would have made the models more accurate. Next, almost all of the variables were correlated. This makes sense as weather factors work together to form rain but this made it difficult to implement Naïve Bayes'. Finally, the dataset was too large to implement a more exhaustive GridSearchCV. This implementation wasn't time sensitive, so it was possible to allow the GridSearchCV to run for a while. Unfortunately, this won't always be the case for future implementations.

*Conclusion.* To summarize, Gradient Boosting achieved the highest accuracy score in predicting if it will rain the next day. However, the accuracy score from the model was only 88.32% on the test set. I do not consider this an accurate prediction. Additionally, the feature importance plot

indicated that the most important features in predicting the target variable were Sunshine, Cloud3pm and Humidity3pm. This is different from the variables in my hypothesis which were Evaporation, Humidity3pm, Pressure3pm, Cloud3pm and Temp3pm. Therefore, I conclude that I cannot accurately predict if it will rain the next day using Evaporation, Humidity3pm, Pressure3pm, Cloud3pm and Temp3pm. Before I propose suggestions to further improve the implementation, I would like to restate the last 3 research questions and answer them.

3. **Which locations in Australia get the most rain?** Cairns, Darwin and CoffsHarbour.
4. **Is it more likely to rain the next day if it rained on a particular day?** No, it is less likely to rain the next day if it rained on a particular day.
5. **Which locations in Australia are the hottest?** Katherine and Darwin.

To improve this model in the future, I would obtain more clean data before re-attempting the implementations. As mentioned earlier, more than half of the records had at least one null value. I would also try to predict if it will rain the next day with a different set of variables, perhaps data collected later than 3pm or data collected the next day. I could also work with a domain expert to identify some independent variables that can be used in the Naïve Bayes' algorithm. Finally, rather than using GridSearchCV, I would try using RandomizedSearchCV. RandomizedSearchCV is less exhaustive compared to GridSearchCV as it selects random combinations of hyperparameter values rather than using every combination of the values ("A Comparison of Grid Search and Randomized Search Using Scikit Learn." ). Due to this, RandomizedSearchCV is less computationally intensive but the accuracy score might be lower.

# References

Maklin, Cory. "Gradient Boosting Decision Tree Algorithm Explained." *Medium*, 21 July 2019, towardsdatascience.com/machine-learning-part-18-boosting-algorithms-gradient-boosting-in-python-ef5ae6965be4#:%7E:text=Gradient%20Boosting%20is%20similar%20to,of%20between%208%20and%2032.

"Sklearn.Ensemble.GradientBoostingClassifier — Scikit-Learn 0.24.1 Documentation." *Scikitlearn*, scikit-learn developers (BSD License), scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html. Accessed 30 Mar. 2021.

"What Makes It Rain?" *NOAA SciJinks – All About Weather*, USA.gov, scijinks.gov/rain/#:%7E:text=What%20causes%20rain%3F,fall%20to%20Earth%20as%20rain. Accessed 31 Mar. 2021.

Worcester, Peter. "A Comparison of Grid Search and Randomized Search Using Scikit Learn." *Medium*, 12 June 2019, blog.usejournal.com/a-comparison-of-grid-search-and-randomized-search-using-scikit-learn-29823179bc85.