Investigating rain prediction in locations across Australia using Deep Learning

Ben Jordan (70091361)

Introduction

For my project, I tackled the problem of weather prediction from the perspective of a weather forecasting enthusiast. In particular, I looked at the binary classification of the event 'Will it rain tomorrow?' based on the data available from the last 7 days. My time spent in Ljubljana has piqued my interest in this question, particularly with how unreliable the weather forecast has been - so how hard can it be? It is also important to gain a degree of understanding of what features are important and where the prediction comes from. To give this problem a crack myself, I used a dataset on Kaggle, Rain in Australia[1] which contains years of daily weather data for numerous locations across Australia. I redefined the designated problem given on Kaggle from a regular classification problem to a time-series classification problem, this would allow me to leverage more data to inform the prediction.

Methods

Data Description

The data set author states that the data contains 'about 10 years of daily weather observations data' for Australia. In reality, it contains many years of weather data for 49 locations across Australia with some missing sections and some locations only having data for the last few years of records. As well as the target variable, *RainTomorrow* each entry contains additional explanatory variables such as *MaxTemp*, *Pressure9am*, *Humidity9am*, *Cloud3pm* and *RainToday* and more.

Adapting to time-series problem

Since I re-purposed this dataset for a time-series classification analysis, several steps and considerations had to be taken to ensure the integrity of the data for a time-series analysis. Firstly, I changed my target variable to *RainToday* and removed *RainTomorrow* as I will be using the 7 days previous to predict the next entry. I also had to ensure that my time-series data was continuous, at least concerning the 7-day window, which was a challenge as I have data across numerous locations and there are numerous temporal gaps in the data for each location. Finally, it was important to decide how I will fairly split my data across training, testing and validation - care had to be taken to consider the different locations and gaps in the data. I was only looking at a 7-day window so I was not too concerned about seasonal variations and trends but I did want to allocate at least 1 year's worth of data for testing and also 1 year's worth for validation for all locations to verify the model's performance year-round. The previous years would be used for training, which gave a split of 74% - 13% - 13% for the training, validation and test sets.

This immediately raises the concern of Concept Drift, where the data is no longer relevant to the problem, for example over time the characteristics of *RainToday* may have changed. I monitored and investigated this by comparing the performance on the validation set to the test set.

Exploratory Data Analysis

Missing Values

As I mentioned earlier, missing data poses a big problem for time-series data in both the typical way but also with the additional issue of maintaining the continuity of the time-series samples.

Firstly, I looked at the entries over time for each location. In Figure 1, we can see that the coverage for each location varies and there are gaps for many of the locations. This problem can be mitigated by carefully preparing the samples (each sample is 7 days of data) of each of these continuous sections for each location and then re-combining them for training.

Secondly, I looked at the more traditional missing values: in the entries that are observed, what is not present? In Figure 2, we can see that the data is far from complete in this sense as well. The amount of missing data and what is missing varies by location. Some features are entirely missing for locations such as *Evaporation* and *Sunshine* for Newcastle. Removing observations is not an option as this will lead to more gaps in the time series. Removing columns is possible but I wanted to avoid this as well. Therefore, I assumed my data was Missing Completely at Random and imputed the missing values. I used a simple imputation based on the training set: median for continuous variables, to be more robust to outliers and mode for categorical variables. Note that for some observations the target had to be imputed as well to maintain the continuity of the data.

Data Preparation

After resolving the missing values, I also carefully encoded categorical variables and standardised the continuous variables in preparation for training. I created dense samples from my data using a sliding window of 7 days with stride 1 and combined and split up into appropriate training, validation and test sets as described.

Models

I used an implementation of the Transformer-based multivariate time series model, TSTPlus [1] in the tsai package [2] and experimented with the data used for training and the hyperparameters to optimise performance. I experimented with dropout rates (on both the fully-connected component and the

[1] Rain in Australia Dataset: https://www.kaggle.com/datasets/jsphyg/weather-dataset-rattle-package

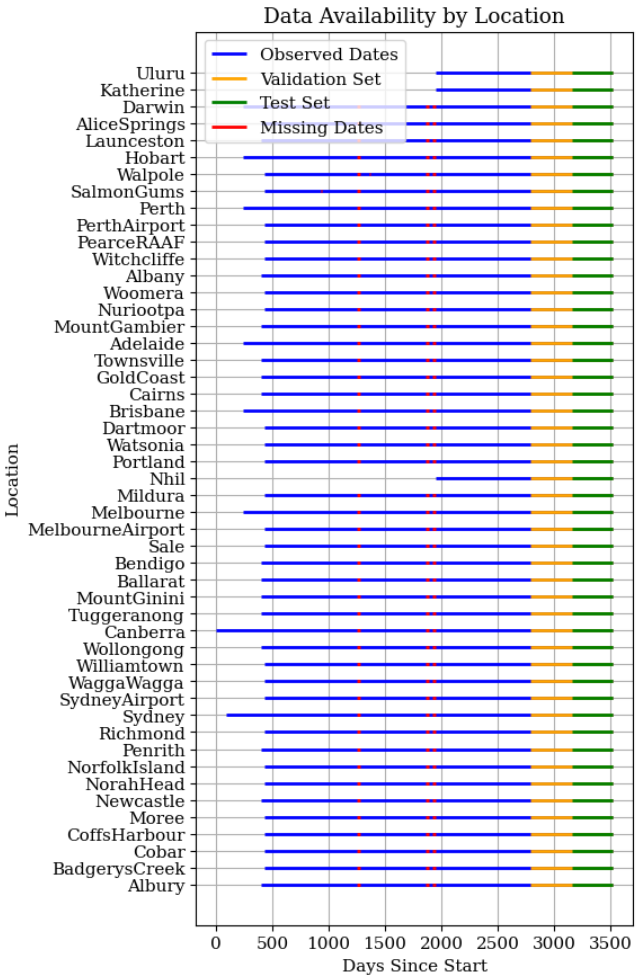

Figure 1: **Temporal gaps** in data by location.

attention components), depth and learning rate, to tune the model.

Baseline

To give a baseline performance I implemented a simple algorithm to predict the probability of rain on the 8th day given the last 7 days as simply the proportion of days that it rained in those 7 days. This gave me an idea of how difficult the problem would be and how much this difficulty varied across locations. This algorithm achieved an AUC of around 0.6 on the training set, leaving plenty of room for improvement.

Univariate time series classification

I started on the univariate problem, only using the target variable, *RainToday* in my model. This should be in effect a smarter implementation of the initial baseline. This will give an indication of how useful the rest of the data provided is for predicting rainfall. The univariate model improved slightly on the simple algorithm achieving an AUC of just over 0.65 on the training set.

Multivariate time series classification

Next, I used the full source of data available to train a model to predict the probability of rain for the next day based on all the data from the last 7 days. I trained the TSTPlus model with default parameters - no regularisation; 3 layers; and model dimension of 128. Out of the box, this achieved good results with an overall AUC of around 0.85 however, the model failed to overfit. To unlock more performance, I increased the model layers to 5 and doubled the dimensions

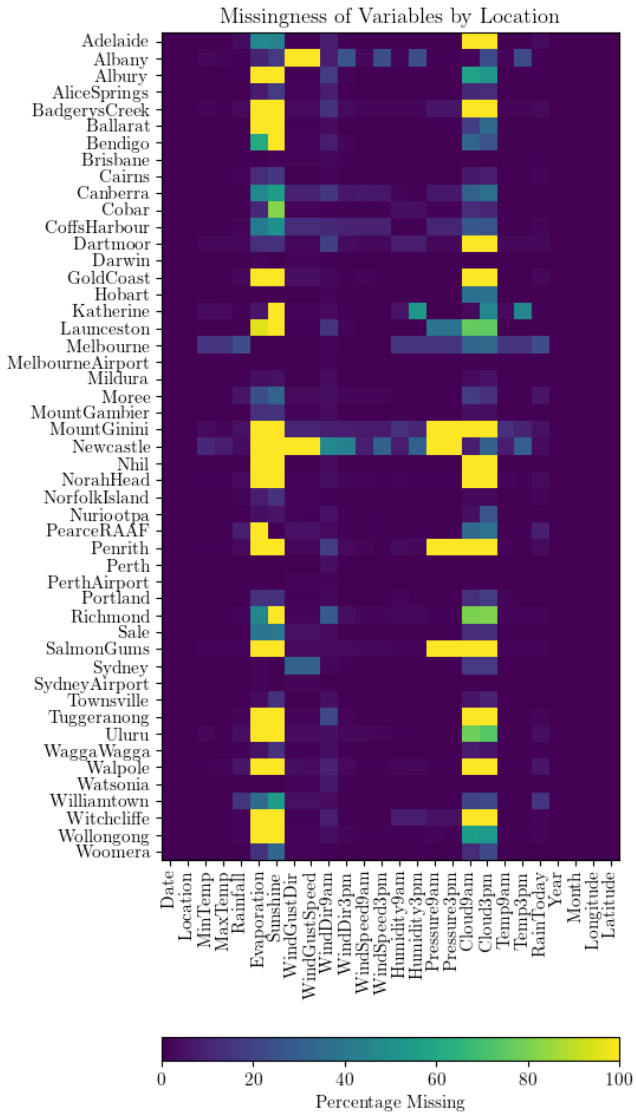

Figure 2: **Missing values** by variable and location.

and attempted to get the model to overfit. The model improved recording an AUC of 0.99 on train and validation sets. It proved to perform best without any regularisation perhaps due to the noise in the data already from the imputation and also the large amount of data used for training.

Results

I then used my model to predict on each location's withheld set of the last year of data. Due to the imbalance in the dataset, I adjusted the threshold for classifying predictions from 0.5 to the one dictated by Youden's statistic - maximising True Positives while minimising False Positives - on the validation set. The overall confusion matrix is shown in Figure 3, it shows that of the 17,000 days 16,500 were predicted correctly and that the model tends to over-predict slightly with more False Positives than False Negatives.

In Table 1, it is clear that the multivariate model substantially outperforms the baseline and the Univariate models across all metrics. Further, the performance is comparable to that on the validation suggesting there is no issue with concept drift and the learned relationships are as relevant on the most recent data. The model achieves very good performance with a very low error rate.

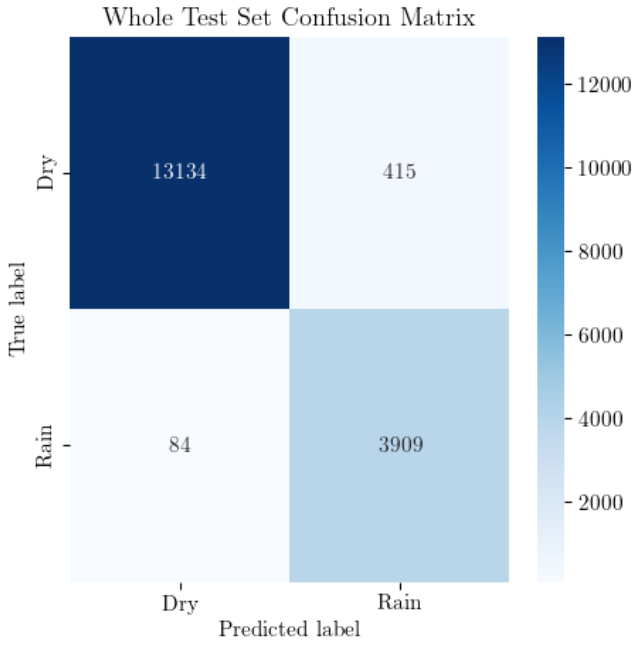

Figure 3: **Confusion matrix** of predictions on entire test set

Model	AUC	F1	Acc.	Precision	Recall
Baseline	0.60	0.31	0.69	0.32	0.35
Univariate	0.65	0.42	0.71	0.37	0.49
Multivariate	**0.99**	**0.94**	**0.97**	**0.91**	**0.98**

Table 1: **Comparison of Model Performance** on the test set aggregated across locations

Spatial variation

Initially, when considering the baseline and the simpler univariate model there is a clear disparity in the predictions. From Figure 5, we can see that inland and northern locations receive better predictions from the models. This suggests that these locations are less volatile and more predictable if these simple models have a reasonable level of performance for these areas. However, many southern and coastal locations do not have good results, suggesting a more volatile and unpredictable weather pattern potentially due to proximity to the ocean. This disparity appears to all but disappear when the final multivariate model is used for prediction, which suggests that with the augmentation of the additional explanatory variables and the more complex model, the rain for even these more volatile locations can be predicted well.

There is still some disparity in the predictions as can be seen in the individual confusion matrices in Figure 4, for the best and worst performing locations. Perth airport only reports one False Positive whereas Cairns has 30 days incorrectly predicted, Cairns is the more rainy of the locations potentially making the prediction harder there. However, the AUC is still very high for both locations at 0.99 and 0.97 but the precision and recall are 0.98 to 0.82 and 1 to 0.92 respectively, suggesting that part of this difference is accounted for by the threshold for classification favouring the Perth Airport location more.

Interpretation

To better understand my model and verify the results, I performed a crude feature ablation study, in which one feature

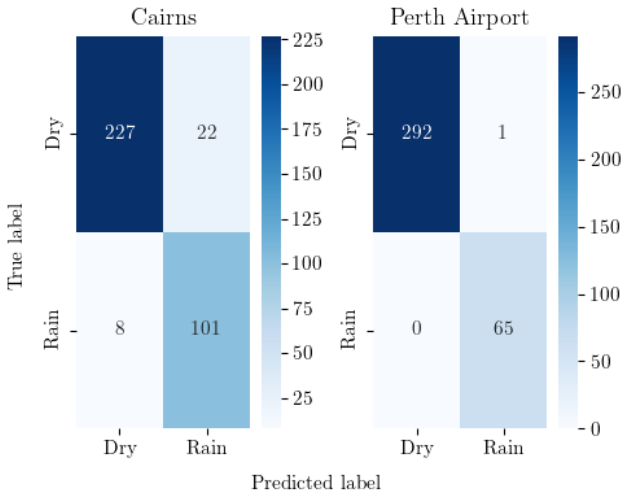

Figure 4: **Best vs Worst location** confusion matrix.

at a time was fixed to its median/mode and then predictions were made on the test set. This gives a rough indication of the importance of the features to the model however, it does not capture interactions between features nor give much interpretation of the variable's effect on the response.

In Table 2, the change in performance metrics when the given feature is fixed to its median is shown. Even with this crude analysis, we can see which features influence the predictions. Under the constraints and shortfalls of this approach we can even tentatively suggest which variables are most important. It appears that pressure at 3pm & 9am; humidity at 3pm; and wind gust speed are the most important factors. A hobbyist weather predictor may agree with us on this point as pressure and humidity are most definitely related to rain and a high wind gust speed might indicate unstable and changing conditions. One surprise is that using this method it appears that *Rainfall* is not one of the most important features. This could be a limitation of this method and potentially a characteristic of the rainfall patterns in Australia.

This approach may raise more questions than it answers but it is a good first step toward understanding the model and also assuring us of its integrity.

Conclusion

The final multivariate time series model achieves good performance on the task of predicting the next days rainfall in locations across Australia based on a 7-day history and improves greatly on a simple baseline. There is some spatial variation in the results, with some locations proving harder to predict than others. The model correctly predicts rainy days with 97% success and dry days with 97% success.

Further, it was possible to infer some level of feature importance from the model through an ablation study which highlights what the important conditions/features to observe are when predicting rainfall.

It must be noted that Australia's climate is dissimilar to that of Slovenia however, I still feel like the weather forecast should be much better.

Extensions

This was my first attempt at tackling the problem of weather forecasting classification and has led to some good first steps however, naturally, there is lots to improve upon and areas to extend this project.

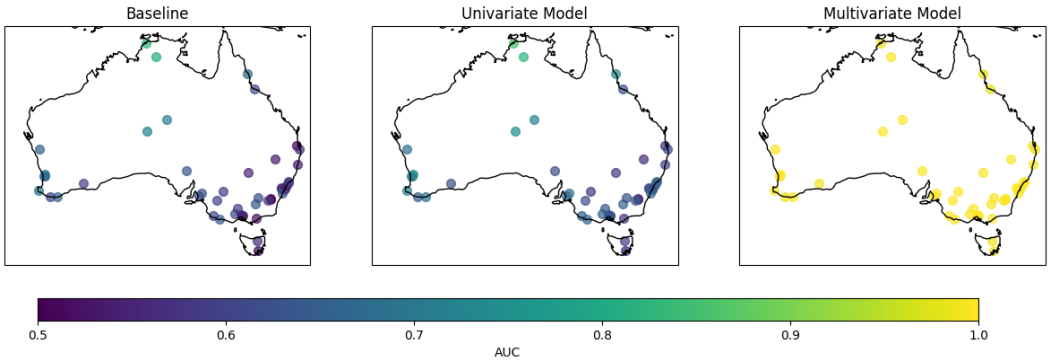

Figure 5: **AUC across Australia** by model.

Variable	AUC	Acc.	Precision	Recall	F1
Pressure3pm	**-0.16**	-0.15	-0.32	**-0.36**	**-0.34**
Humidity3pm	-0.09	-0.10	**-0.20**	-0.33	-0.27
Pressure9am	-0.08	-0.10	-0.23	-0.18	-0.21
WindGustSpeed	-0.07	-0.09	-0.18	-0.24	-0.21
MaxTemp	-0.07	-0.09	-0.20	-0.22	-0.21
Humidity9am	-0.05	-0.07	-0.16	-0.17	-0.16
MinTemp	-0.04	-0.05	-0.10	-0.14	-0.12
WindSpeed9am	-0.03	-0.05	-0.11	-0.11	-0.11
Temp9am	-0.03	-0.04	-0.08	-0.13	-0.10
Temp3pm	-0.03	-0.05	-0.12	-0.11	-0.12
WindSpeed3pm	-0.02	-0.05	-0.10	-0.10	-0.10
Rainfall	-0.02	-0.03	-0.03	-0.09	-0.06
Cloud9am	-0.01	-0.02	-0.04	-0.04	-0.04
WindDir3pm	-0.01	-0.02	-0.03	-0.06	-0.04
Evaporation	-0.01	-0.01	-0.03	-0.03	-0.03
Sunshine	-0.00	-0.01	-0.02	-0.03	-0.03
WindGustDir	-0.00	-0.01	-0.01	-0.02	-0.02
Cloud3pm	-0.00	-0.01	-0.02	-0.02	-0.02
WindDir9am	-0.00	-0.01	-0.02	-0.01	-0.02
Location	0.00	0.00	0.00	0.00	0.00

Table 2: **Feature ablation** of explanatory variables across performance metrics

Data

An improvement to the data preparation would be smarter imputation such as stratifying the imputation by location and using the medians and modes for those locations and also filling in missing values by using observed values for that date for the spatially closest location with observed data.

Applying to a new location with a different climatology. Australia is for the most part an arid country, it would be interesting to see if what the network learned would apply to a more volatile and unpredictable climate zone such as Ljubljana and if the model could be transferred and fine-tuned on data from such a location. Then to investigate if there are differences in the importance of features for these different climates.

Interpretation

The interpretation is limited at the moment and from an amateur weather predictor's perspective knowing why the weather is changing or why it might rain is particularly important. The next steps would be discerning the importance of features on a given day in the history window and the importance of each day in the window on the prediction. Moreover, using more sophisticated techniques such as SHAP[3] values and LIME[4] would offer more insight into how the features interact with each other and what change in them affects the response.

Models

The current model looks to have some difficulties with locations, in the Ablation Study it shows that the location has no impact on prediction. This is counter-intuitive due to the size of Australia and the different climate zones. Moreover, despite high AUC almost all around, there are some marked differences in the precision and recall for some locations. One solution may be applying individual thresholds for classification on each location or investigating why the location is not being incorporated in the predictions.

Given the very good results, it seems that predicting just the next step for this data is well within the capabilities of the current models. It would be interesting to see how these models can predict multiple days into the future. An extension to the task would be training a model for Multi-step prediction, such as predicting a week of rain from the past 2 weeks and observing if/how the performance degrades further into the horizon. Moreover, looking at the regression task and using *Rainfall* as the target would be another interesting angle.

References

[1] George Zerveas, Srideepika Jayaraman, Dhaval Patel, Anuradha Bhamidipaty, and Carsten Eickhoff. A transformer-based framework for multivariate time series representation learning. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery amp; Data Mining*, KDD '21, page 2114–2124, New York, NY, USA, 2021. Association for Computing Machinery.

[2] Ignacio Oguiza. tsai - a state-of-the-art deep learning library for time series and sequential data. Github, 2023.

[3] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *CoRR*, abs/1705.07874, 2017.

[4] Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. *CoRR*, abs/1602.04938, 2016.