Time Series Temperature Prediction

Tomer Zur

Abstract

The following report will explore predicting the weather each day. This report will review LSTM neural models that I created to predict the daily max temperature in four Australian cities, and answer the question of how well weather patterns over time can be used to predict the weather with LSTMs. After reviewing the performance of these models, this report will reach the conclusion that the models and data being used are well suited for this task, as they achieve a low error in each of the different cities being tested.

Background

For my final project I am predicting the weather using time series modelling. Specifically, I am predicting the max temperature each day in four different Australian cities. I chose this topic because I wanted to get some practice working with time series models and applying machine learning to the task of forecasting time series data. I also thought that weather would be an interesting topic to apply machine learning to.

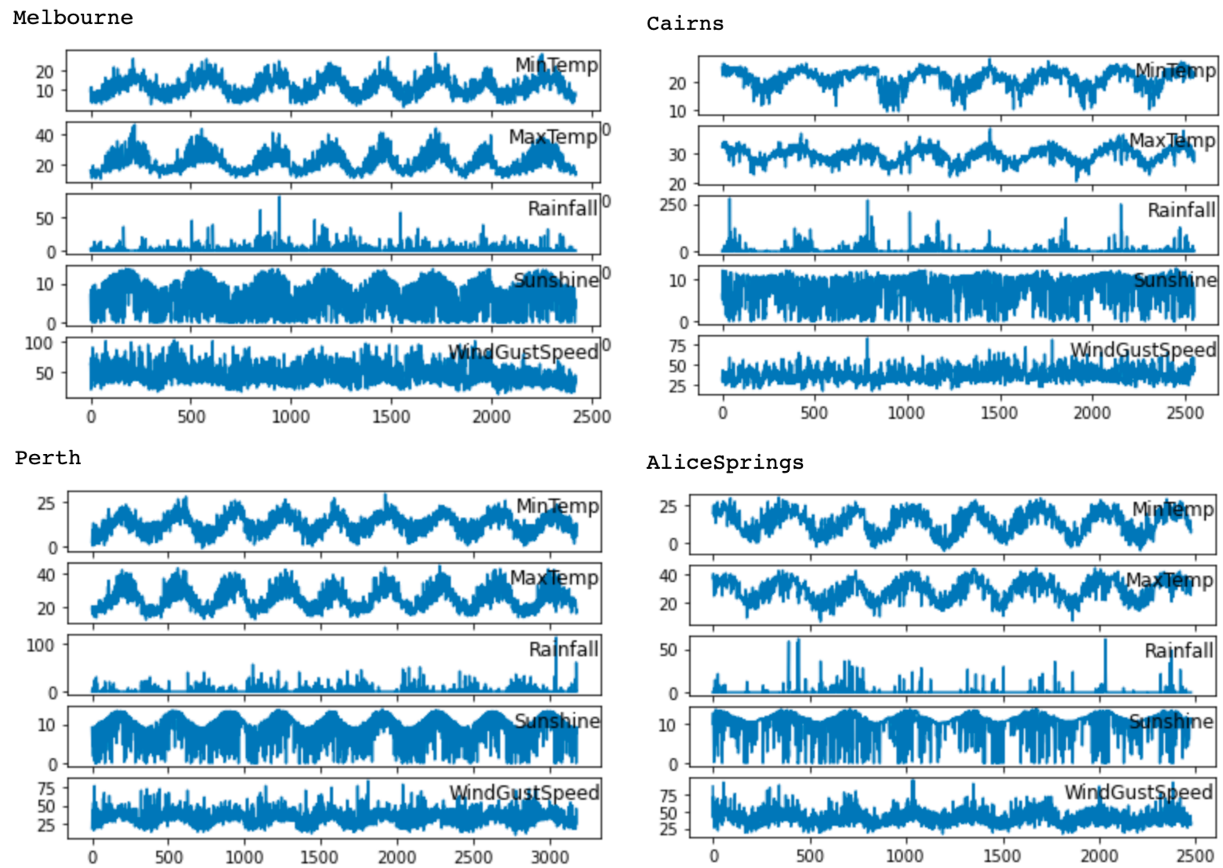
I am predicting the max temperature in four cities which are located far apart geographically. These cities are Melbourne, Cairns, Perth, and Alice Springs. I chose these four cities because they are each located in geographically different areas, and will each have different climates and weather patterns.

Data

The data source I am using is a csv of the weather data from a bunch of different Australian cities. This dataset is publicly available on Kaggle.

The features that I am using from this dataset to make my predictions are windows of the previous week's daily max temperatures, min temperatures, rainfall, sunshine, and wind speed. I will then pass these features in to an LSTM neural model, which will make my predictions.

As we can see in the graphs on the next page, these features all follow seasonal patterns. In all four cities, we can see that each of the features are higher at certain times during the year and lower at other times. Hopefully, these trends get captured by the LSTM.



Daily observations for the five features we will be using in each of the four cities. We can see that each of these features follows a yearly pattern.

Methods

I am using an LSTM neural model to predict the future temperatures. I chose this model because it is generally the preferred model for time series predictions. This is because the LSTM model has feedback connections that help it remembering information over time.

The features that the model is using to make predictions are the minimum temperature, maximum temperature, amount of rainfall, hours of sunshine, and maximum wind speed. For each of these five features, the model will receive the observed values of that feature on the 7 days prior to the current day. Here is a table showing the data that the LSTM will receive in order to make a prediction on one day (with the current day at time t):

Day	Min Temp. (°C)	Max Temp. (°C)	Rainfall (mm)	Sunshine (hrs)	Max Wind Speed (km/h)
t - 7					
t - 6					
t - 5					
t - 4					
t - 3					
t - 2					
t - 1					

My LSTM model is pretty simple. It contains an input LSTM layer, and an output layer. The LSTM layer has 50 neurons, and output layer has 1 neuron (for the single prediction being made). The other hyperparameters that I have in my model are:

- The loss function – Mean Absolute Error
- The optimizer – Adam
- Number of Epochs – 50
- Batch Size – 72

I will be making predictions for up to 7 days in the future. This will provide us with valuable information about if, and how much, the model's performance drops as it predicts further and further into the future.

I will also be using a baseline model to measure my model's error against. This baseline model will simply predict the current day's temperature as the future temperature. For example, if the max temperature in Melbourne today is 20 °C, and the baseline is predicting the temperature in the future, its prediction would then be 20 °C. This same prediction would be made regardless of how many days in the future the baseline model is predicting.

Two more things I did with my data before fitting the model are scaling the data and splitting it into a training set, a validation set, and a test set. I scaled the data using a MinMax scaler, as I needed to make sure each feature had a range of values from 0 to 1. I also split my data into training, validation, and test data using two cutoff dates. These cutoff dates are the same for each of my four cities, so the train/validation/test data was split in this way:

Train	Up to 9/30/2013
Validation	10/1/2013 – 4/25/2015
Test	4/26/2015 and after

I chose these cutoff dates because I wanted to make sure I had at least one year's worth of data in each of my validation and test sets. Since some cities had missing days, those were the most recent cutoff dates that I could choose that would still allow all four validation and all four test sets to each contain at least a year of data.

Results

Here are my final results on my test data. For each city, I will compare the performance of the baseline to the LSTM model. Then, I will show a graph of how the accuracy of the LSTM changes as we predict further into the future. For these results, I will be calculating my accuracy metrics using my test data. If you want to see how my model and baseline performed on the train and validation data sets, you can see those results on my project's Github.

These numbers vary slightly with each run, but these are the numbers I got after my most recent run of my Jupyter Notebook:

Melbourne

# of days ahead we're predicting		**1**	**2**	**3**	**4**	**5**	**6**	**7**
Mean Squared Error (°C)	Baseline	20.57	34.88	36.30	34.33	32.68	32.54	35.46
	LSTM	12.95	19.13	19.91	19.30	19.30	19.64	20.59
Root Mean Squared Error (°C)	Baseline	4.54	5.91	6.02	5.86	5.72	5.70	5.95
	LSTM	3.60	4.37	4.46	4.40	4.39	4.43	4.54
Mean Absolute Error (°C)	Baseline	3.19	4.33	4.42	4.37	4.17	4.19	4.35
	LSTM	2.72	3.33	3.32	3.28	3.22	3.22	3.27

Cairns

# of days ahead we're predicting		1	2	3	4	5	6	7
Mean Squared Error (°C)	Baseline	1.82	2.66	2.99	3.06	3.28	3.48	3.92
	LSTM	1.45	1.94	2.15	2.25	2.40	2.50	2.53
Root Mean Squared Error (°C)	Baseline	1.35	1.63	1.73	1.75	1.81	1.87	1.98
	LSTM	1.20	1.39	1.46	1.50	1.55	1.58	1.59
Mean Absolute Error (°C)	Baseline	0.99	1.26	1.32	1.34	1.38	1.44	1.50
	LSTM	0.90	1.02	1.09	1.13	1.18	1.21	1.21

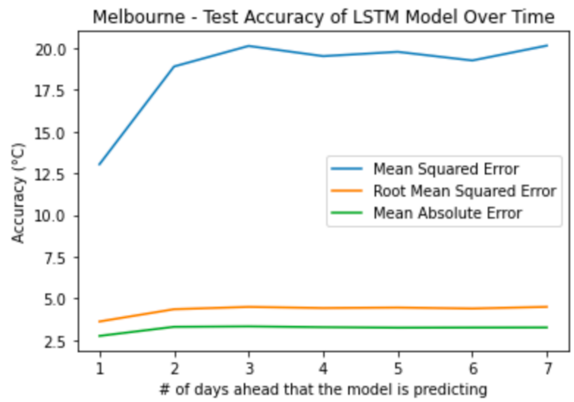
Perth

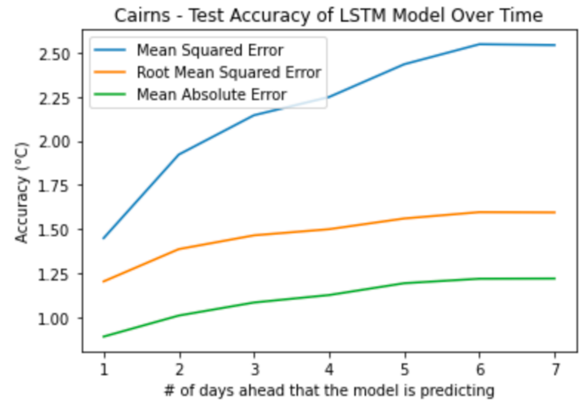
# of days ahead we're predicting		1	2	3	4	5	6	7
Mean Squared Error (°C)	Baseline	13.19	25.96	30.69	30.21	29.54	29.71	30.35
	LSTM	10.16	16.44	17.45	17.99	17.36	17.06	18.20
Root Mean Squared Error (°C)	Baseline	3.63	5.09	5.54	5.50	5.44	5.45	5.51
	LSTM	3.19	4.06	4.18	4.24	4.17	4.13	4.27
Mean Absolute Error (°C)	Baseline	2.64	3.80	4.10	4.09	4.09	4.11	4.20
	LSTM	2.41	3.09	3.19	3.20	3.18	3.15	3.20

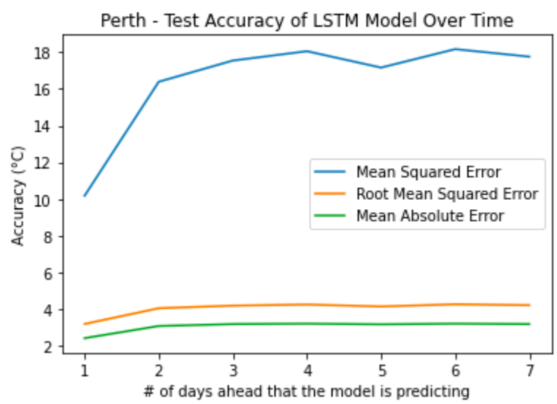
Alice Springs

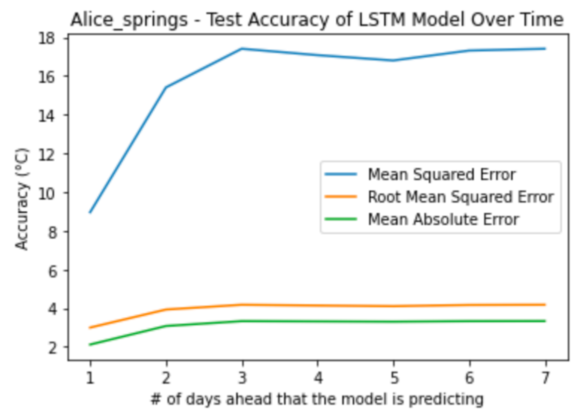
# of days ahead we're predicting		1	2	3	4	5	6	7
Mean Squared Error (°C)	Baseline	10.72	21.62	28.28	30.15	29.31	29.66	28.07
	LSTM	8.68	16.53	17.54	16.94	16.88	17.42	17.30
Root Mean Squared Error (°C)	Baseline	3.27	4.65	5.32	5.49	5.41	5.45	5.30
	LSTM	2.95	4.07	4.19	4.12	4.11	4.17	4.16
Mean Absolute Error (°C)	Baseline	2.42	3.74	4.34	4.40	4.29	4.32	4.18
	LSTM	2.13	3.22	3.34	3.29	3.31	3.33	3.34

Graphs of LSTM Accuracy over Time









Conclusions

From the results presented above, we can see that our LSTM model predicts the daily max temperatures with pretty good accuracy in all four cities. Depending on the city and number of days ahead we are predicting, the model was able to predict the temperature with an error (mean absolute error) of at most 3.34°C, and achieved an error of as low as 0.9°C. These numbers are very good. They are small enough that most people can't notice (or will have a hard time noticing) the difference between the temperatures the model is predicting and the actual temperatures when they go outside.

When we look at the performance of our models in each city, we can see that the model predicts the temperature better in some cities than others. For example, in Cairns, the mean absolute error when predicting the next-day temperature was 0.9°C, while in Melbourne, the mean absolute error for the same prediction was 2.72°C. Most likely, this difference is due to the natural weather patterns in different cities. In Cairns, the temperature stays relatively the same throughout the year, while in Melbourne, the temperature fluctuates a lot more during the year.

Another key takeaway from our results is that the model performs worse the further out we are making predictions. This what we would expect, as there is generally more variance in possible temperatures the further out in time we get from a given day.

One of the most important takeaways is that the model consistently outperformed our baseline. This puts our model's performance into perspective, and lets us know that our low error rates are actually very good error rates to achieve.

Ultimately, this improvement shows that our model was able to derive meaning from our data and able to make informed and generally reliable predictions.

References

The link to the Github repository with my code is: https://github.com/tomerzur/Time-Series-Temperature-Prediction
The Australian weather dataset is available at: https://www.kaggle.com/jsphyg/weather-dataset-rattle-package