Weather and climate forecasting using long short-term memory models

Alejandro Rigg Franco, 19 January 2022

Introduction

The aim of the project was to see if machine learning techniques could be applied to weather and climate forecasting. The models designed were recurrent neural networks, using long short-term memory layers. The data used was daily data from weather stations around the world, all members of the Global Historical Climatology Network. As an extension to the project, the data from three neighbouring stations was incorporated and relevant parameters were varied to see if weather forecasts could be improved upon.

Background

In recurrent neural networks (RNN), the state of a system at a particular time is dependent on its former states. They are thus useful for sequential data problems and have been used for many different applications such as speech recognition [1]. and traffic forecasting [2]. In the network design of an RNN, there is a hidden layer or layers where neurons are sequentially connected. One example of such a layer is long short-term memory (LSTM) first described in 1997 [3]. Each layer comprises an array of gated cells that can make binary choices about the flow of information. Each cell receives input about the current state and the output of the former state as well as a longer-term memory input. The cell determines which part of the long-term memory to forget and how much of the output of the former state to add to it. From this modified memory state, the cell then decides what to output. One of the advantages of LSTM layers is that they allow a simple back propagation through the memory inputs; therefore, they are efficient to run. One disadvantage of RNNs is that outputs can accumulate, and cost functions explode rather than converge. It is important to normalise data to protect against this. Key parameters in data preparation for RNN networks are the offset and the time shift. The offset is how far forward the model makes its prediction and the time shift gives the number of previous periods that are considered in making this prediction.

The data used was from the Global Historical Climatology Network (GHCN), a database of daily weather data from over 100,000 weather stations across the globe [4]. For each station, a range of different measurements are recorded. Common ones are: maximum temperature (TMAX), minimum temperature (TMIN), precipitation (PRCP), snow (SNOW) and snow depth (SNWD). These measurements are referred to as “elements”.

Method

The provided python code was adapted so that the *Pandas* library could be used to manipulate and analyse the data. The procedure to read the data into *Pandas* was adapted from [5] to reference the data maintained by UCL and to add a data frame for countries. It was decided to focus on the 991 stations that make up GSN (GCOS Surface Network). The latter is a subset of GHCN and gives “a fairly uniform spatial coverage from places where there is a good length and quality of data record” [6]. A methodology for finding those with the longest, best quality data with a broad number of elements was devised. For each station, extra fields were added: length, completeness and quality of the five main elements (maximum temperature, minimum temperature, precipitation, snow and snow depth). In addition, the start and end date of each file was recorded, alongside the total number of elements present. For example, windspeeds were thought to be interesting additional features to add to the models to enhance predictability. This allowed the stations to be filtered to a shortlist of 26 stations.

Filter	**Criteria**
End date	After 30 October 2020
Snow data	Included in dataset
Snow depth data	Included in dataset
Completeness of maximum temperature data	>0.995
Quality of maximum temperature data	>0.999
Completeness of snow data	>0.990

Table 1 - filters for selecting shortlist of stations

From this shortlist, a preferred station was selected for climate and weather prediction. While the stations had been filtered for completeness, a few gaps existed and needed to be identified. *Seaborn* library heatmaps [7] helped achieve this, and also aided the selection of other elements for testing. Gaps were analysed and, if small, were forward filled. On two occasions there were small gaps in the date ranges, both 4 consecutive days. The missing dates were added with forward filled values for the relevant elements. Statistics were produced for each element to show its mean, standard deviation, minimum and maximum values. This helped identify any anomalous data. On one occasion an anomaly was found, and it was replaced with the average value of that element two days before, and after it. In experiments where data from more than one station was used, the columns were renamed in order to be able to join the datasets along a common date index. All elements were normalised, by dividing by the maximum value of that element. Strictly speaking, temperature data in Celsius could be negative, but any effect was deemed likely to be small as minimum temperatures were never very negative. All other elements used were always positive.

A function was designed to create the number of lookback periods for the data as well as how far forward a prediction was to be made. For climate predictions, monthly data was used with a 120-month lookback and a prediction forward of 12 months. For weather predictions, daily data was used with initially a 365-day lookback and a prediction forward of 1 day, i.e., tomorrow. In an extension project, different lookbacks were tested.

The resulting data was then split into a training set (60% of data), validation set (20% of data) and testing set (20% of data). This was then trained on a model adapted from [8]. This contained 2 LSTM layers: the first with 64 neurons, the second with 32. This was followed by a dense neural network layer of 32 neurons and completed by an output layer to output the number of elements in the dataset. Different models were tested in the extension project; this initial model was named "Alpha".

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 120, 64)	17920
lstm_1 (LSTM)	(None, 32)	12416
dense (Dense)	(None, 32)	1056
dense_1 (Dense)	(None, 5)	165

```
Total params: 31,557
Trainable params: 31,557
Non-trainable params: 0
```

Table 2 – Alpha model summary. Output shape of dense_1 layer would vary with number of elements (in this case five).

The model was compiled with a mean squared error loss function using an ADAM optimiser. Models were then run. Plots were made of the test data compared to predicted data and R-squared statistics calculated.

Results –station and element selection

The selected station for climate and weather predictions was Grand Junction Walker Field, Colorado, USA. It had the longest data (since 1900), with a perfect quality score for all of the five main elements. The completeness of each of these was 0.9998 or above. A second selection was Fairbanks International Airport, Arkansas, USA. It has a long data set from 1929 with nearly perfect quality and completeness scores. As it is fairly nearby, it was considered that might be useful in combination with the Colorado station to improve weather predictions there.

Using a data visualisation method adapted from [7], the completeness of the data could be checked.

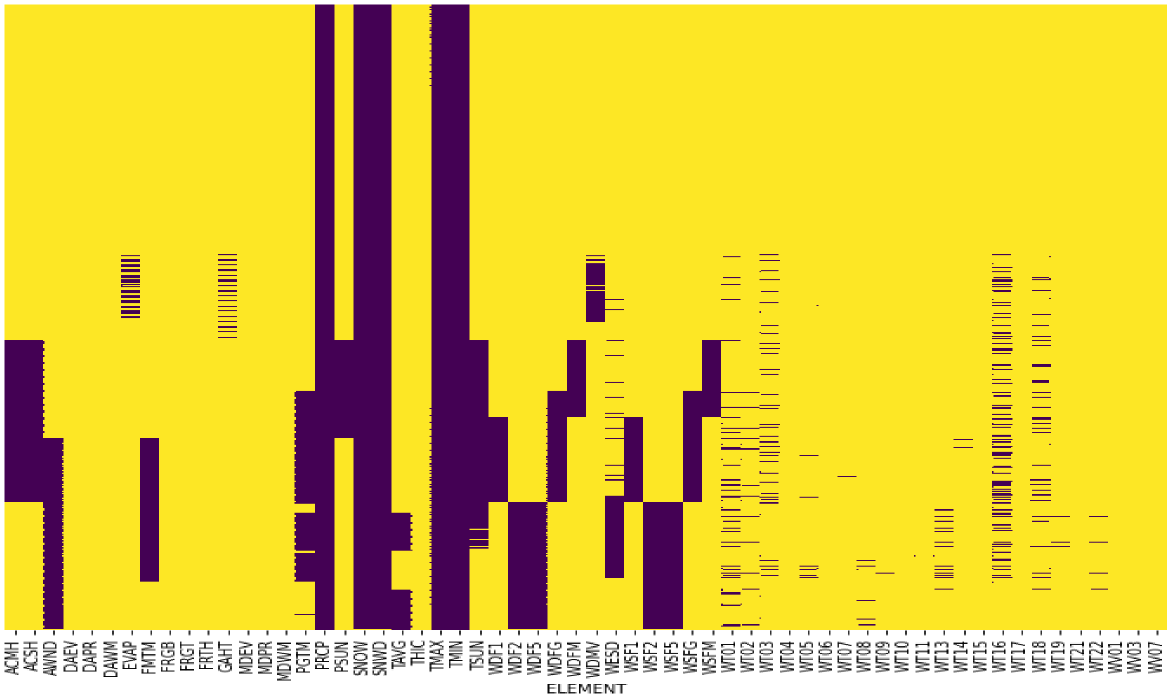

Figure 1. Grand Walker Junction Field, Colorado, USA Heatmap showing data in purple and gaps in data in yellow. NB dates not shown on y-axis for clarity.

Figure 1 shows good data from the start to the end for the five main elements. Good completeness is seen for data for AWND, WDF2, WDF5, WSF2, W5F5, albeit over a shorter period. These elements respectively are average daily wind speed (tenths of meters per second), direction of fastest 2-minute wind (degrees), direction of fastest 5-second wind (degrees), fastest 2-minute wind speed (tenths of meters per second) and fastest 5-second wind speed (tenths of meters per second).

Results - Climate Prediction Model

After some testing using just the element TMAX, it was decided to proceed with the five main elements to train a climate prediction model. First, the data needed to be checked for any null values. The least populated element was TMIN with just 9 values missing out of 44,122 entries (i.e. 0.02%). These do not even show in Figure 1 and it was decided to fill all gaps for all other elements using a forward fill methodology. At the end of the training, the training set loss was 0.0118 and the validation set loss 0.0108; a good convergence had been achieved.

The following figures show the comparison between the predicted results and the test results with the R-squared statistic displayed. This gives the proportion of the variance in the test results explained by the predicted results: 1 means a perfect relationship, 0 indicates no relationship.

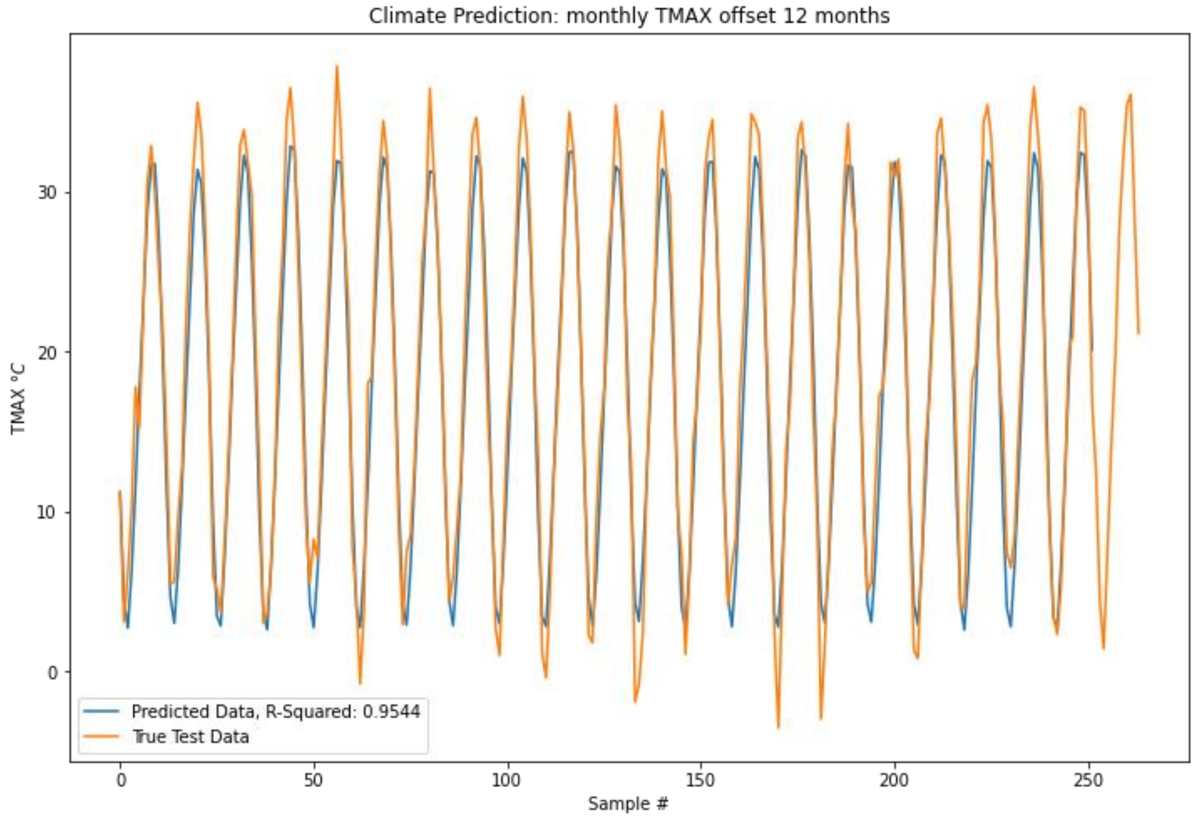

Figure 2. Climate Prediction: Maximum Temperature

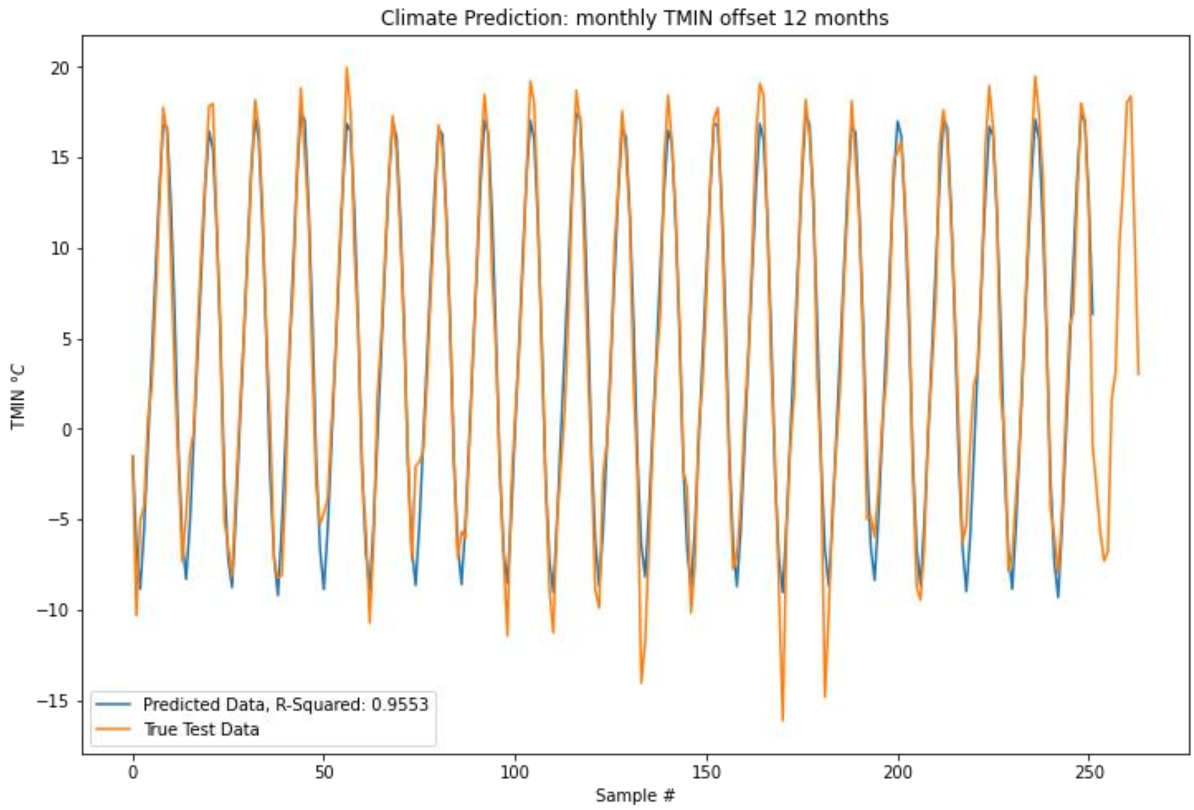

Figure 3. Climate Prediction: Minimum Temperature

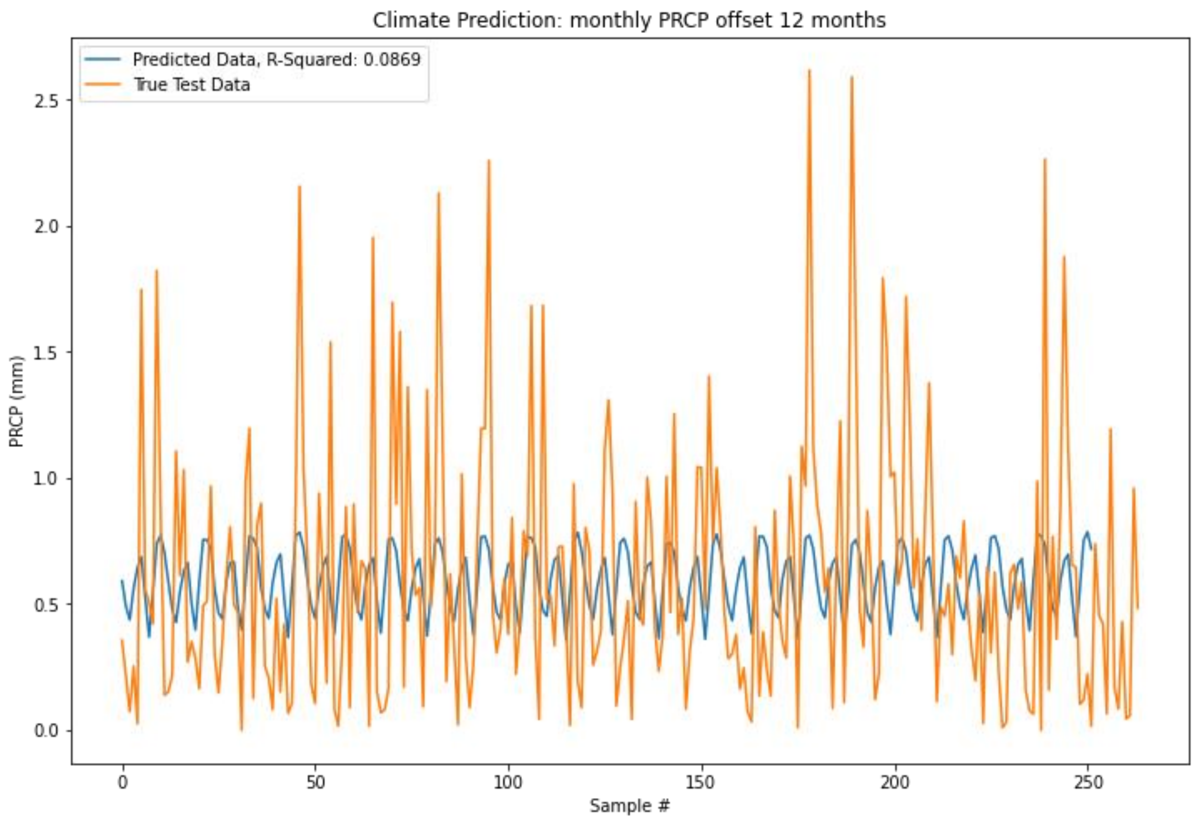

Figure 4. Climate Prediction: Precipitation

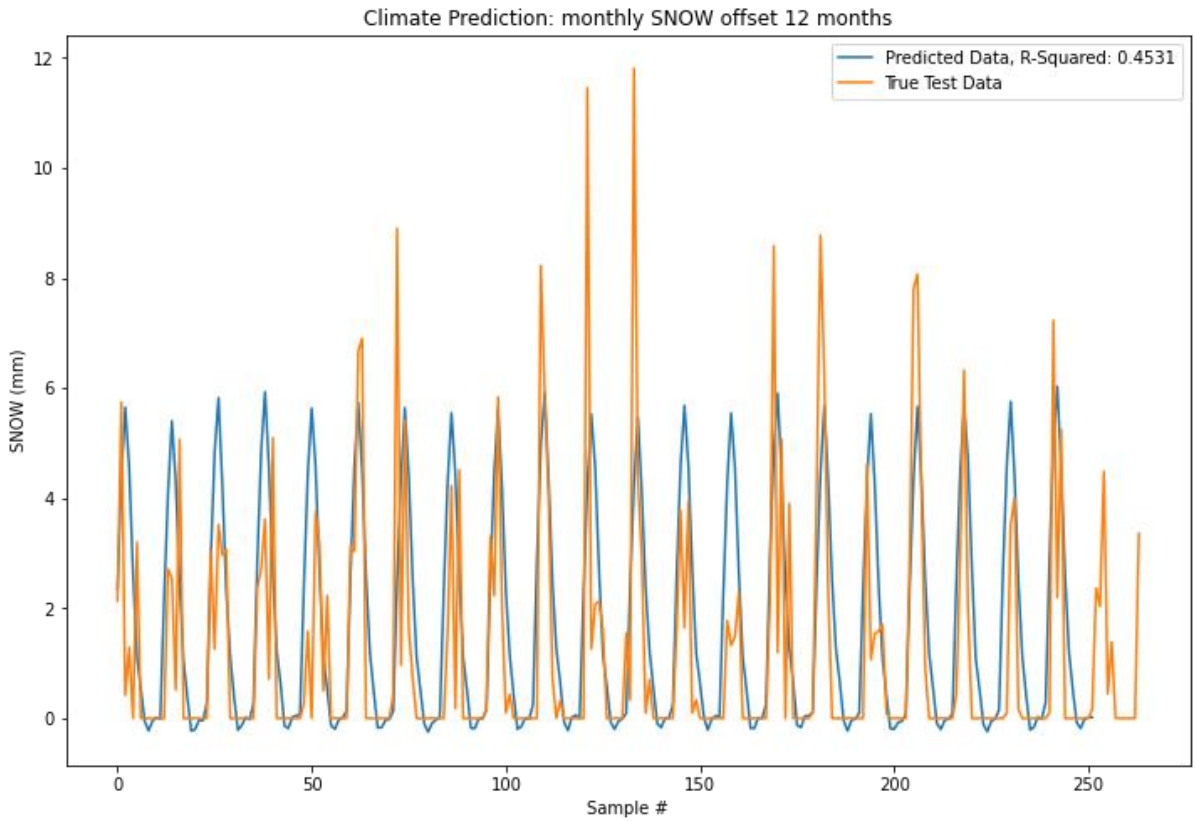

Figure 5. Climate Prediction: Snow

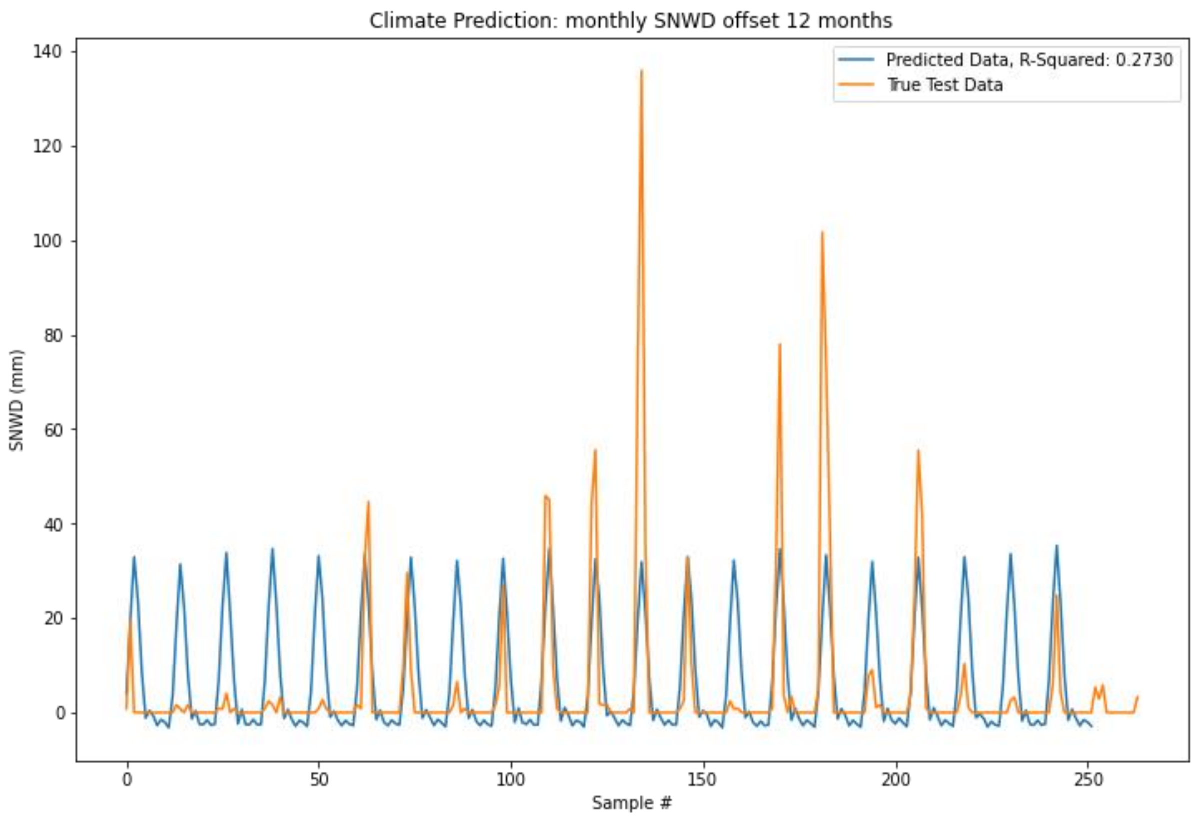

Figure 6. Climate Prediction: Snow Depth

Results - Weather Prediction Model

It was decided to include data from the Arkansas station, but test on the Colorado station. As daily data was used it was decided to use all 10 elements (TMAX, TMIN, PRCP, SNOW, SNWD and the 5 wind elements). The common start date was 1 December 1997. Training converged well with a training loss of 0.0228 and a validation loss of 0.0229. On the following charts of predicted versus actual results, two values of R-squared are presented. The first, compares the predicted data to the true test data. The second compares the true test data to the value of the test data on the previous day. If the former is higher than the later, then the model has predicted the weather better than assuming it would be the same one day as the previous one.

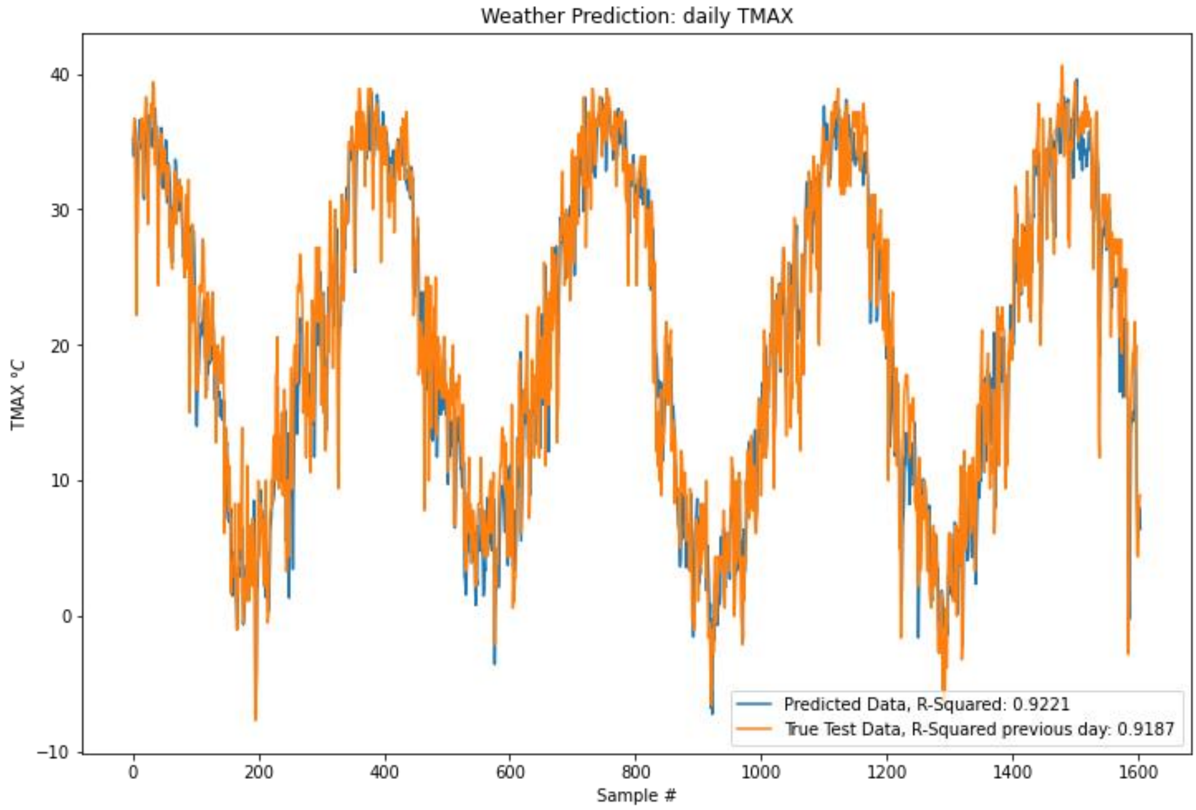

Figure 7. Weather Prediction: Maximum temperature

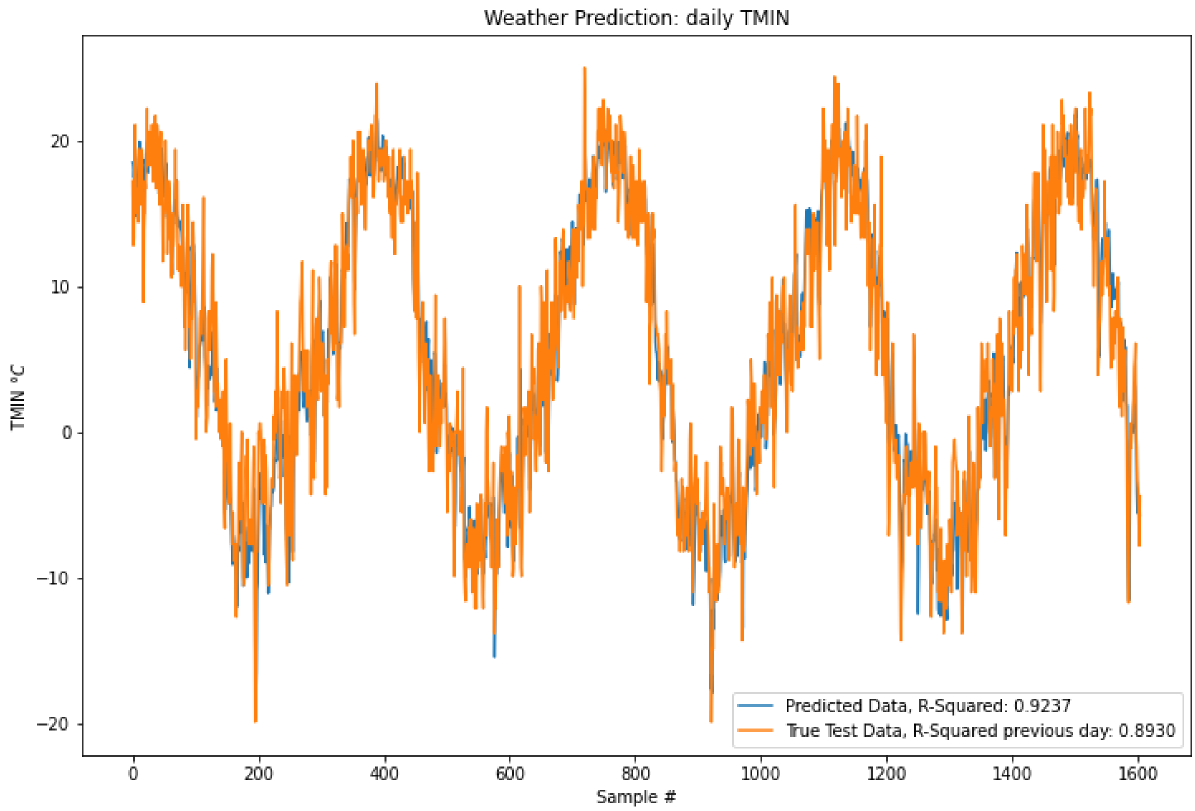

Figure 8. Weather Prediction: Minimum temperature

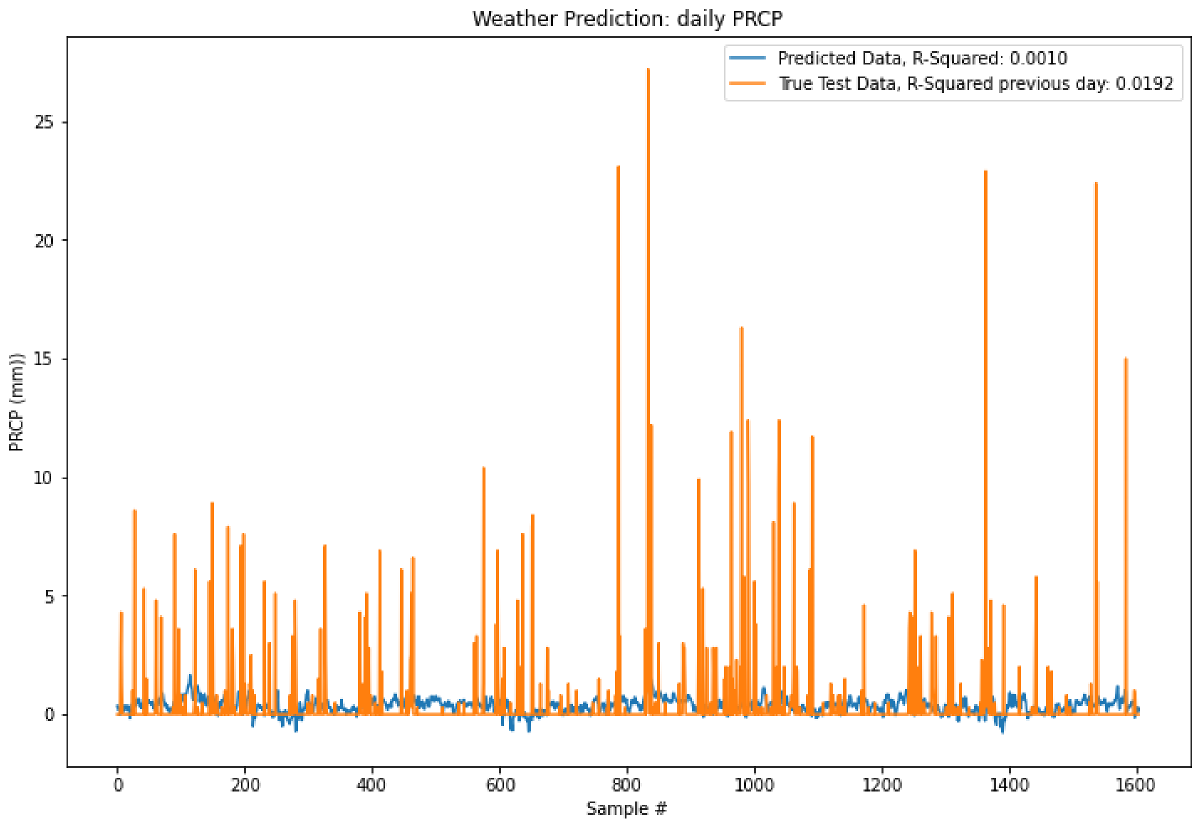

Figure 9. Weather Prediction: Precipitation

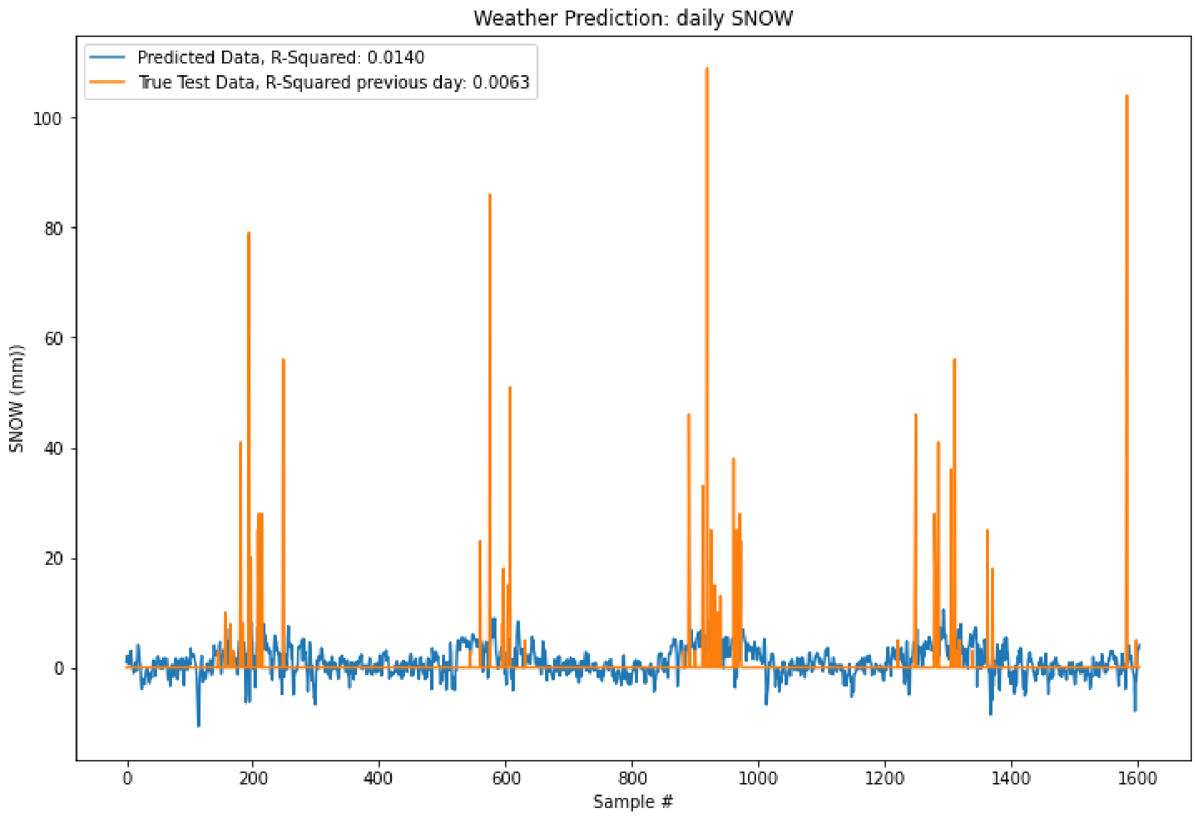

Figure 10. Weather Prediction: Snow

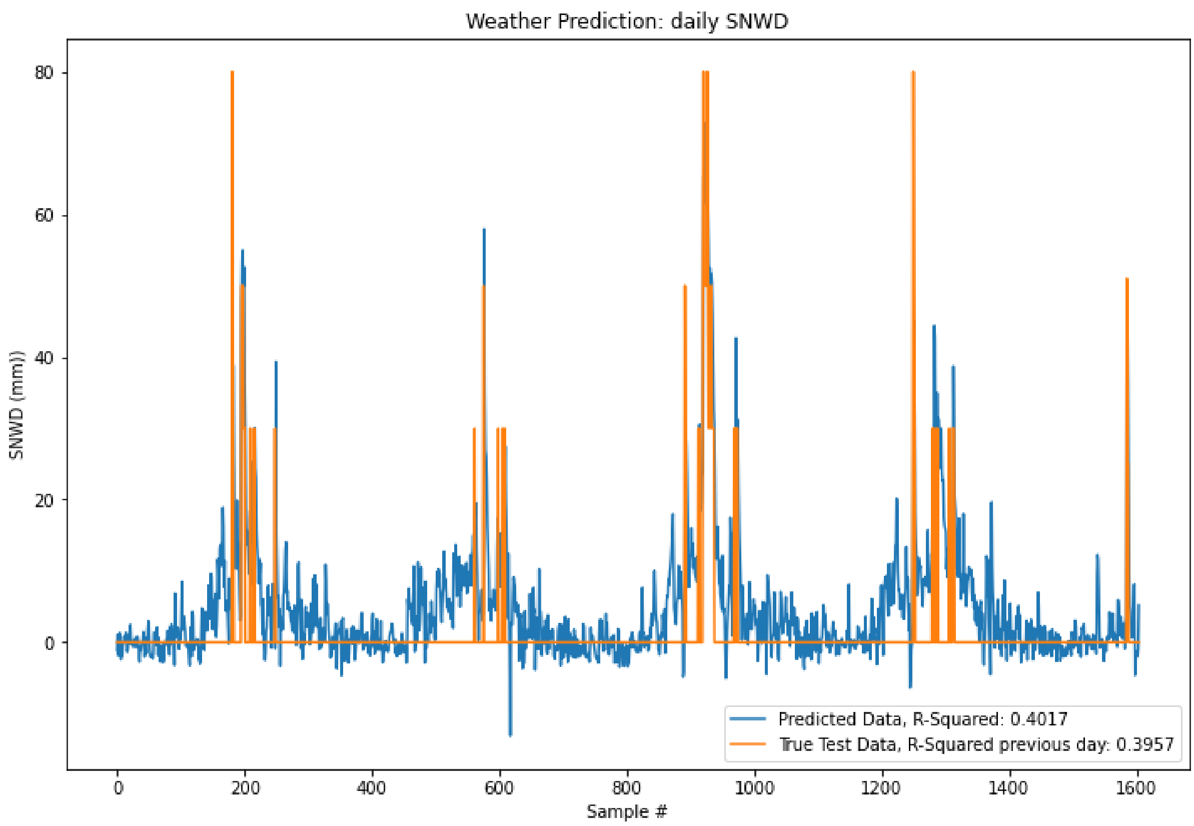

Figure 11. Weather Prediction: Snow Depth

Results – extension project

The extension project considered if the weather predictions for maximum and minimum temperatures at the Colorado station could be improved by adding nearer stations instead of the Arkansas station and by changing other parameters such as the model used, the number of elements included and the time steps. From the short list of 26 stations with good characteristics, three were identified in neighbouring states: Oklahoma City, OK; Albuquerque, NM and North Platte, NE. There were no other stations in Colorado in the short list. Results are shown with just the Colorado station used and then with all four stations. In total 13 runs were made:

No. of stations	No. of Elements	Time steps (days)	Model	TMAX(R-squared)	TMIN(R-squared)	Average	Model improve-ment
			Today predicts tomorrow	92.1%	89.7%	90.9%	
1	2	7	Alpha	92.8%	92.5%	92.6%	1.8%
1	2	30	Alpha	92.9%	92.5%	92.7%	1.8%
1	2	365	Alpha	92.3%	92.1%	92.2%	1.3%
1	10	30	Alpha	94.2%	93.3%	93.7%	2.9%
1	10	365	Alpha	93.3%	93.0%	93.2%	2.3%
1	10	30	Beta	94.4%	93.3%	93.9%	3.0%
4	2	30	Alpha	92.9%	92.9%	92.9%	2.0%
4	10	30	Alpha	93.9%	92.6%	93.2%	2.4%
4	10	365	Alpha	93.2%	92.0%	92.6%	1.8%
4	10	30	Beta	93.8%	92.8%	93.3%	2.4%
4	10	30	Gamma	91.6%	91.1%	91.3%	0.4%
4	10	30	Delta	93.0%	92.0%	92.5%	1.6%
4	10	30	Epsilon	93.6%	92.1%	92.9%	2.0%

Table 3. Extension Project - determining most important parameters for weather forecast

The Alpha model architecture is shown in Table 2. The Beta model was identical except at each hidden layer there were twice the number of neurons: the first LSTM layer had 64 neurons, the second 32 and the dense layer had 64 neurons. The Gamma model was the most complex having even broader LSTM layers and including a convolutional neural network (CNN) layer with a kernel size of 10 and two dropout layers with dropout ratios of 0.2 each.

Layer (type)	Output Shape	Param #
lstm_14 (LSTM)	(None, 30, 256)	304128
conv1d (Conv1D)	(None, 21, 32)	81952
lstm_15 (LSTM)	(None, 256)	295936
dense_14 (Dense)	(None, 64)	16448
dense_15 (Dense)	(None, 40)	2600

Total params: 701,064
Trainable params: 701,064
Non-trainable params: 0

Table 4. Gamma model. Two dropout layers (not shown) were between lstm_15 and dense_14 and then between dense_14 and dense_15.

Delta and epsilon models were identical to gamma except that delta excluded the CNN layer and epsilon excluded the two dropout layers.

Conclusions

In this project, LSTM models were effective in forecasting monthly average temperatures a year in advance. They identified the periodicity of the temperatures as the seasons changed but would often fail to capture closely the extremes in summer and winter months. Monthly average precipitation varied in a more chaotic way and the models were almost completely ineffective (R-squared value of 0.09) in their predictions. Snow and snow depth predictions captured the periodicity of the data, i.e. snow in winter not in summer. However, they were ineffective at capturing the extremes. It was noted that some of the prediction of snow and snow depth were negative. As a refinement, this should be disallowed in any future experiments.

A similar pattern was seen for the weather prediction models. Maximum and minimum temperatures were closely modelled and provided slightly better predictions that just using yesterday’s temperature as a guide for today. The model fits for precipitation, snow and snow depth were poor.

The extension project considered the most important parameters in determining model success for predicting daily temperatures: number of stations used, number of elements included, number of time steps and models used. Today’s temperature is a good predictor of tomorrow’s, but the models all showed improvements over that base case, albeit to different degrees. Of the time steps tested, the best results were achieved for 30 days as opposed to 7 days or 365 days. A large improvement was seen for one station when moving from just the two temperature elements to ten, i.e. adding in precipitation, snow, snow depth and the five wind measurements. Surprisingly, the performance of the model fell when three neighbouring stations were added. Similarly, attempts to add convolution layers and dropout layers worsened the results. Widening each hidden layer by a factor of two (beta model) improved results from the original model (alpha), but not by much. The alpha and beta models were quick to train and this confirmed that LSTM models can run quickly with large amount of data.

The project highlighted the importance and complexity of data manipulation. Investing the time to understand *Pandas* and *Sklearn* libraries proved invaluable in the project. Much more time was spent data wrangling than expected. This was very necessary, and it provides a solid basis for further experimentation. Further experiments could involve trying out even more network architectures, e.g., including other types of layers, than dropout or CNN layers. Also, it would be interesting to consider other stations, variables and predictions with different look-backs or elements.

[1] A. Graves, A.-R. Mohamed and G. Hinton, "Speech recognition with deep recurrent neural networks," *IEEE international conference on acoustics, speech and signal processing*, pp. 6645-6649, 2013.

[2] Z. Zhao, W. Chen, P. C. Chen and J. Liu, "LSTM network: A deep learning approach for short-term traffic forecast," *IER Intelligent Transport Systems*, vol. 11, no. 2, pp. 68-75, 2017.

[3] S. Hochreiter and J. Schmidhuber, "Long term short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735-1780, 1997.

[4] M. J. Menne, I. Durre, R. S. Vose, B. E. Gleason and T. G. Houston, "An overview of the Global Historical Climatology Network-Daily Database," *Journal of Atmospheric and Oceanic Technology*, vol. 29, pp. 897-910, 2012.

[5] N. Haughton, "Pandas extraction of GHCN data," December 2021. [Online]. Available: https://github.com/nedclimaterisk?tab=repositories. [Accessed 20 December 2021].

[6] NOAA, "The GCOS Surface Network (GSN)," 2015 October. [Online]. Available: https://www.ncdc.noaa.gov/gosic/global-climate-observing-system-gcos/gcos-surface-network-gsn-program-overview. [Accessed 5 January 2021].

[7] J. Qazi, "Data Science and Machine Learning using Python - A Bootcamp," [Online]. Available: skillshare.com. [Accessed 20 December 2021]

[8] R. Nicol, "pandas - Python data analysis tutorial - - Optional content," 22 November 2020. [Online]. Available: https://github.com/nichol77/mlForPhysicists/blob/master/Week6/Week6_pandas.ipynb. [Accessed 20 December 2021].