404 Weather Prediction Project

Caleb Dame, Daniel Mortenson

April 2021

Abstract

Weather prediction is a canonical problem in algorithmic forecasting. This study aims to showcase the performance of predictive weather models created daily coordinate data of temperature, precipitation, relative humidity and air pressure. Both ARMA and Convolutional Neural Networks are trained to make predictions about near-future weather. These models' performances are tested and compared against each other and baseline accuracy. Data limited to daily observations prove to be insufficient to train a neural network (either CNN or RNN) to make accurate predictions and accounting for multiple variables at each significant location prove to be to prohibitive for ARMA models.

1 Introduction

1.1 Problem

Forecasting has long been one of the most difficult and popular areas of data modeling, and the classic forecasting problem of predicting the weather has long been a notorious one. Weather is a phenomenon that is well understood mechanically and its changes have long been explainable by science, but even today, when we ask if it is likely to rain tomorrow, we are more often than not only answered in terms of probability.

For our project, we attempt to accurately model such weather statistics such as temperature and precipitation in various regions of the United States by leveraging decades of time series data. Using various modeling techniques such as Auto Regressive Moving Average (ARMA) and Convolutional Neural Network (CNN) Models, we hope to minimize the Mean Squared Error (MSE) of the predictions.

1.2 Motivation

Most weather forecasting is done by climatologists and meteorologists with a large knowledge of weather mechanics and domain expertise, at institutions with a capacity to collect more frequent data, and with the computational infrastructure to build extremely complex, realistic, and costly models. Recent papers employ Convolutional Neural Networks that improve upon the standard industry predictions, but are trained on the entire globe using hourly data and incorporating many additional features such as altitude, UV radiation, barometric pressure, and relative humidity [Weyn 2020]. Our motivation is to answer the following questions:

- **Can a simple model effectively forecast weather phenomena?**

  Simpler models often lend themselves to easier interpretation and greater transparency. The benefit of using a simple model in predicting the weather is that, after building and training the model, one could gain greater intuition of how the mechanisms of weather work and how independent features in past periods contribute to future events. Effective simple models, especially in the physical and life sciences, tend to inform future research that estimate causal and mathematical relationships.

- **Can an extremely flexible model accurately forecast weather phenomena when only trained on a limited number of features only collected daily?**

  For more complicated models (like Neural Networks), like those used in the professional field of weather forecasting, make up in performance what is lost in transparency. We would like to push the limits and see accurately weather can be forecasted when only a handful of features measured daily are accessible (Precipitation, Max Temperature, Min Temperature, etc.), but large amounts of geographic and decades long longitudinal data are used in estimation.

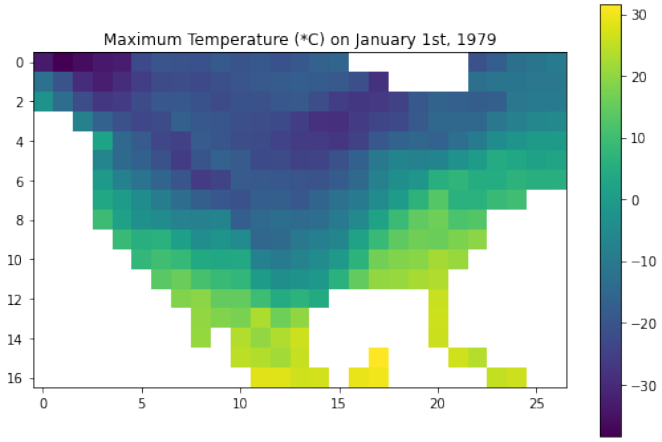
2 Data

2.1 Data Origins and Composition

One of the benefits of the modern era is the abundance of data over the past several decades. Since we are concerned with building an accurate model that depends on a few features and the geographical connections between those features, we determined that we would incorporate the maximum and minimum temperatures, relative humidity, air pressure, and precipitation levels on a latitude-longitude grid across the United States. Fortunately, the Physical Sciences Laboratory of the National Oceanic and Atmospheric Association of the United States of America, has compiled free-to-use databases that suit the needs of this study perfectly. The databases we accessed recorded daily data from Jan 1st, 1979 to December 31, 2020, which gives us a total of 15341 days to work with [Xie and Arkin 1997].

To prepare our data for our models, we down-scaled our higher-resolution data to match our lower-resolution data. Although the maximum and minimum temperature and precipitation data was sampled at every .5 degree latitude and longitude, the relative humidity and air pressure data was sampled ever 2.5 degrees latitude and longitude, so we averaged each 5x5 chunk of the higher resolution data so that the shape of each data set would be the same. We also restricted our latitude and longitudes to the continental United States, although any region of the world with accurate data would function the same. One day of Maximum Temperature data is visualized in Figure 1.

Figure 1: Down-scaled Sample, Maximum Temperature



2.2 Data Cleaning and Feature Engineering

Because the PSL dataset we are using for this study is so extensive, there are many missing data that we needed to handle. For the temperature data, relative humidity, and air pressure features, we replaced missing values by averaging the nearest available data in time, with the majority of cases being isolated, single-day events. In the case of multiple missing datapoints in a row, we performed a simple linear interpolation using the non-missing datapoints on either end of the missing section. Luckily, there were no gaps in the data larger than 5 days, so we are not concerned that these replacements affected the overall quality of the dataset. For the precipitation dataset, we replaced all missing data with 0 since no precipitation is more likely than some precipitation on a given day across the United States. Similarly, there were very few missing datapoints, and we are confident in the reliability of these data.

3 Methodology

It this paper, we will limit our modeling techniques to various adaptations of the canonical Auto Regressive Moving Average (ARMA) model, a Convolutional Neural Network (CNN), and a Recurrent Neural Network (RNN). Our ARMA implementation will be evaluated on its ability to predict one day in the future at a time, and the CNN will evaluated in its ability to predict multiple days into the future.

3.1 ARMA Models

To evaluate the various implementations of ARMA, each implementation will be tasked with predicting, one day at a time, the change in weather for the entirety of the year 2020 in Provo, Utah (United States).

In our research will will evaluate and compare the following to model variations in forecasting both the daily minimum and maximum temperatures

1. AR(1 day) with data from only the coordinate of interest
2. ARMA(7 days, 1 noise term) with data from only the coordinate of interest
3. ARMA(14 days, 2 noise terms) with data from only the coordinate of interest
4. ARMA(1 day *(nine locations)*,1) with data from eight neighboring zones
5. ARMA(3 days *(nine locations)*, 2) with data from eight neighboring zones

We have found that, computationally, training an ARMA model that estimates more than 30 auto-regressive coefficients can take prohibitive amounts of time, so we will be unable to explore (within the context of ARMA) the benefits of considering the weather in neighboring areas and their auto-regressive coefficients beyond 3 days and 8 neighboring zones.

Each different ARMA model will be evaluated identically. To avoid regression to the mean, an occurrence in ARMA models when future predictions are made using past predictions, we will forecast only a single day in advance and retrain the model each new day with only true weather data.

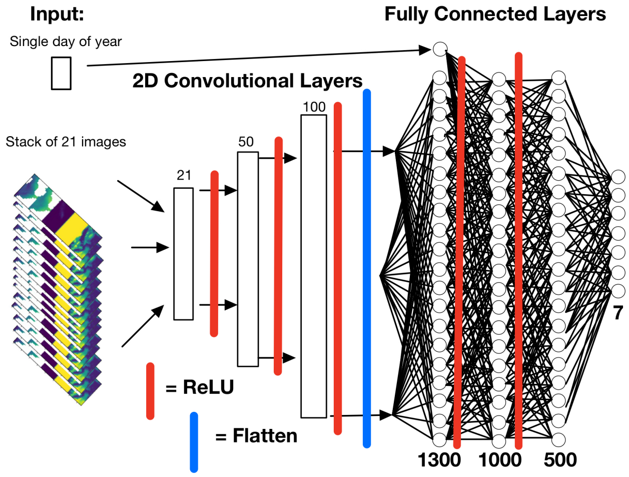
To avoid bias in selecting a sample over which we evaluate the model, we will evaluate each set of model parameters as it predicts the temperature for each day over the course of the year 2020. Also, even after de-trending and de-seasonalizing the data, there still appear to be fluctuations in the variance and speed of oscillation of the residuals, indicating possible seasonal variation in the optimal ARMA model. To avoid residual weather patterns in the distant past from figuring heavily in the training of a model that predicts the next day, the model will be trained on a moving window of only 28 days in order to prevent the predictions from being overly past-biased.

3.2 Convolutional Neural Network

Convolutional Neural Networks have become the most popular form of machine learning because of their ability to find nonlinear relations between large sets of features and target classes. We trained the CNN with large blocks of each available feature in the 5x5 block around our location of interest, along with the day of the year during which the 21-day period started. We hard-coded the neural network model to allow the day of the year to bypass the convolutional layers, so that its value is not lost to matrix operations in the convolution layers. The model outputs a 7-day forecast for the maximum temperature following the 21-day training period.

After testing multiple architectures, we determined that the following would be the best architecture for the weather-prediction Convolutional Neural Network:

Figure 2: Convolutional Neural Network Architecture



3.3 Recurrent Neural Network

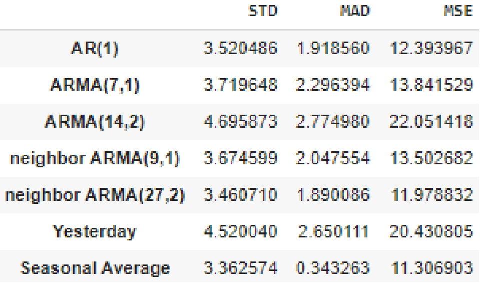
A Recurrent Neural Network (RNN) is an obvious choice for weather forecasting because RNNs handle time-series data effectively. We created an LSTM (Long Short Term Memory) RNN to forecast maximum temperatures for the next 7 days after receiving the previous 21 or 100 days' maximum temperatures. The RNN was composed of a 3-layer LSTM with dropout followed by 3 fully-connected layers. We tested both ingesting and predicting the maximum temperatures directly and ingesting and predicting the changes in temperature from one day to the next. We also tested multiple different rounding schemes for the possible inputs and outputs (increasing and decreasing the size of the model's "vocabulary."

4 Results

4.1 ARMA Models

In the Appendix, Figure 6 is a graph indicating the likelihood of model error exceeding a given threshold. As can be seen, only for very specific ranges of errors do any of the ARMA models provide more accurate predictions than predicting the daily average temperature. Of all the ARMA models, the best performing model is ARMA(3 days (nine locations), 2), the one which takes in its own and 8 adjacent coordinates, computes 3 days of autoregression coefficients for each, and is a two-degree moving average model. The worst performing model is ARMA(14,2), performing worse than using the temperature today as the next day's prediction.

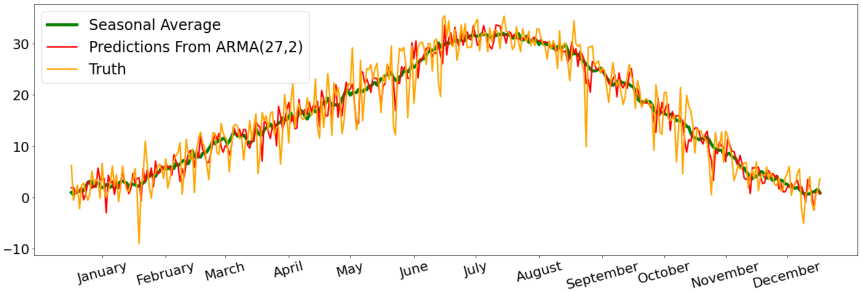
Figure 3: Performance Table



	STD	MAD	MSE
AR(1)	3.520486	1.918560	12.393967
ARMA(7,1)	3.719648	2.296394	13.841529
ARMA(14,2)	4.695873	2.774980	22.051418
neighbor ARMA(9,1)	3.674599	2.047554	13.502682
neighbor ARMA(27,2)	3.460710	1.890086	11.978832
Yesterday	4.520040	2.650111	20.430805
Seasonal Average	3.362574	0.343263	11.306903

The table in Figure 3 shows key statistics for each of the methods, comparing them to rules of thumb like using the daily average temperature or the past day's temperature as the prediction. The MSE refers to the Mean Squared Error, STD is the standard deviation of the error, and the MAD is the median absolute deviation, a robust statistic similar to a standard deviation interpreted as the variability of error.

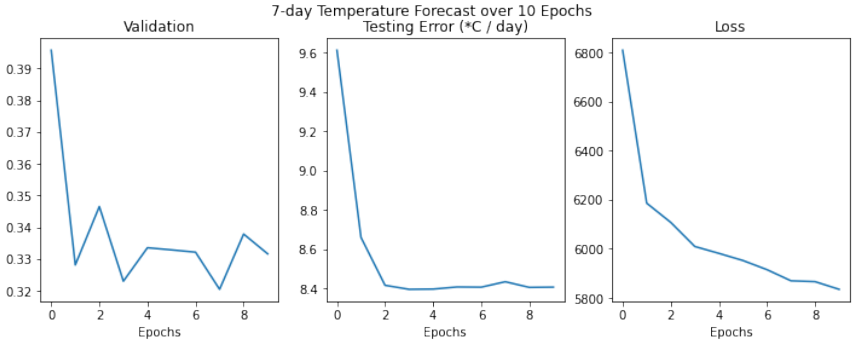
Figure 4: Seasonal Temperatures



4.2 Convolutional Neural Network Performance

Our CNN usually converged fairly rapidly (within 10 epochs). Our main metric for judging the forecast against the actual temperatures is to take the L1-norm of the difference between the forecast and reality, and divide it by 7, which represents the average error (°C) per day.

Figure 5: Training Progress over 10 Epochs



While the testing error does decrease as the network changes, it approaches about 7.5°C average error. This is not necessarily a bad prediction, but as we can see in Figure 7 (See Appendix), the forecast does not generally follow the actual temperature closely.

4.3 Recurrent Neural Network Performance

Our Recurrent Neural Network predicted the 7-day forecast one day at a time, using the test sample for the first day's prediction, and then appending the result to the test sample to make the next prediction. Like the CNN, we measure the error of the RNN by the average degree-Celsius error per day. Our best model achieved an error of 4.5°C on the test set, which outperformed the Convolutional Neural Network (See Appendix, Figure 8).

4.4 Analysis

4.4.1 ARMA Analysis

No matter the complexity of the ARMA model that we used, none were able to out-predict the seasonal average in terms of MSE and MAD. This means that, using the seasonal average as a baseline, the variations of the various ARMA models actually lead each model having a negative R^2 as the suggested deviations from the seasonal average will lead to larger error than if there were no deviations.

This suggests that, while there may be some autoregressive behavior in the data, there is not significant enough of an autoregressive relationship to predict the future changes in seasonality. Future weather is likely correlated more strongly with current and past data at points very far away from a given point of interest. However, the ARMA model is limited in its ability to take into account more than a small number of locations for when k locations are included in an ARMA model with autoregression of order l, there will be kl autoregressive coefficients estimated, and we found that for training data around 300+ observations, training a model of order $kl > 30$ to predict a year's weather takes well over 12 hours and often fails to converge due to model complexity.

4.4.2 CNN Analysis

Of the three models mentioned in this report, the CNN was trained on the most diverse data, including maximum and minimum temperature, relative humidity, surface pressure, precipitation, and day of the year. Unfortunately, it seems that even with all of those data, the model was unable to make a reliable prediction the (average error was 7.5°C). We believe that this is likely due to the temporal scale of our data. Tomorrow's weather is most dependent on today's weather, and depends very little on the weather 2 or 3 weeks ago. Because we only had one data point per feature per location per day, the correlations between the inputs and outputs were weak, and so the network consistently mis-predicted the future maximum temperatures.

4.4.3 RNN Analysis

Similar to the CNN, the RNN we built suffered from weak correlations between the input and output data due to the temporal scale of the data, but unlike the CNN, it was able to use the recurrent relations between the previous and next days' weather to make more accurate predictions. Because the RNN predicts one day at a time, its error on the first day of the forecast was consistently less than the error on the 7th day of forecast, which is better than the CNN's performance, which had high error uniformly across the 7-day forecast.

5 Ethical Considerations

The data used in this study has been graciously provided "as a public service" by the Physical Sciences Laboratory of the National Oceanic and Atmospheric Association.

6 Conclusion

Weather phenomena are complicated, and therefore difficult to predict. While our models were very instructive, they only outperformed naive methods (like predicting today's temperature or the seasonal average for tomorrow) occasionally. Additionally, they rarely surpassed even moderately more complex rules of thumb such as guessing the yearly average for that day. Based on our research and experience implementing these models, we conclude that a significant part of their failure lies in our choice of dataset.

Modern weather forecasting models use temporally dense data recorded by the minute and hour as well as many more features that we did not include in our dataset, including wind speed, solar radiation levels, cloud cover, and geographic features such as lakes and oceans. Our daily data did not provide enough detail for our models to predict future weather well. To answer the questions we posed in the introduction:

- **Can a simple model effectively forecast weather phenomena?**

  It is likely possible for a simple model to forecast weather phenomena, given the correct data. If our data had been less temporally spread out, we believe that we would have achieved better results.

- **Can an extremely flexible model accurately forecast weather phenomena when only trained on a limited number of features only collected daily?**

  While there may be models that can make better predictions using daily data than ours did, it is clear that more frequent, recent data give the models higher-correlation variables to make predictions using. It's much better to train on 100 data points from the last 2 days than 100 data points over the last 100 days.

7 Appendix

References

[Wey20] Jonathan Weyn. "Improving Data-Driven Global Weather Prediction Using Deep Convolutional Neural Networks on a Cubed Sphere". In: *Journal of Advances in Modeling Earth Systems* 12 (2020).

[XA97] P. Xie and P.A. Arkin. "1997: Global precipitation: A 17-year monthly analysis based on gauge observations, satellite estimates, and numerical model outputs." In: *Bull. Amer. Meteor. Soc.* 78 (1997), pp. 2539–2558.

Figure 6: ARMA Performance Curve

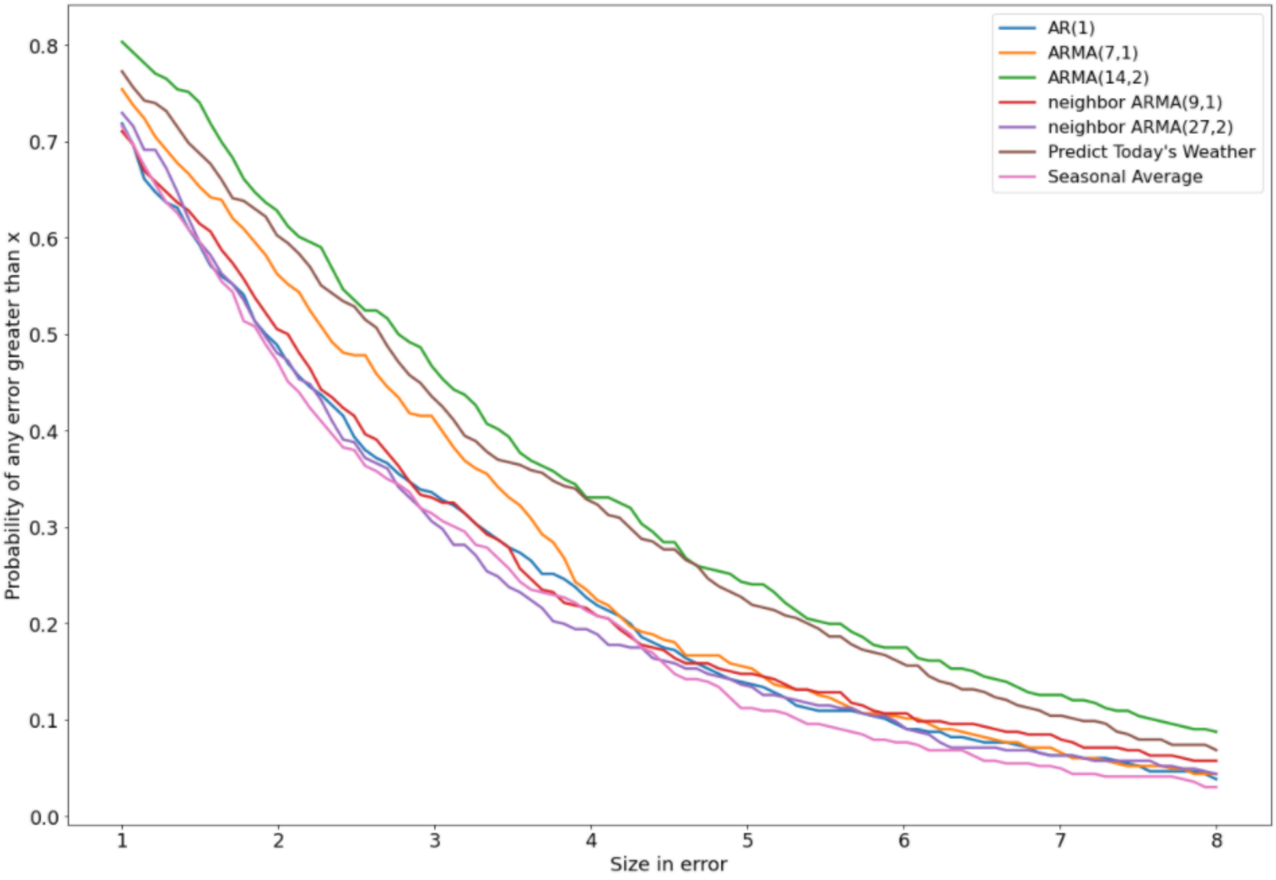


Figure 7: Selected Results (CNN)

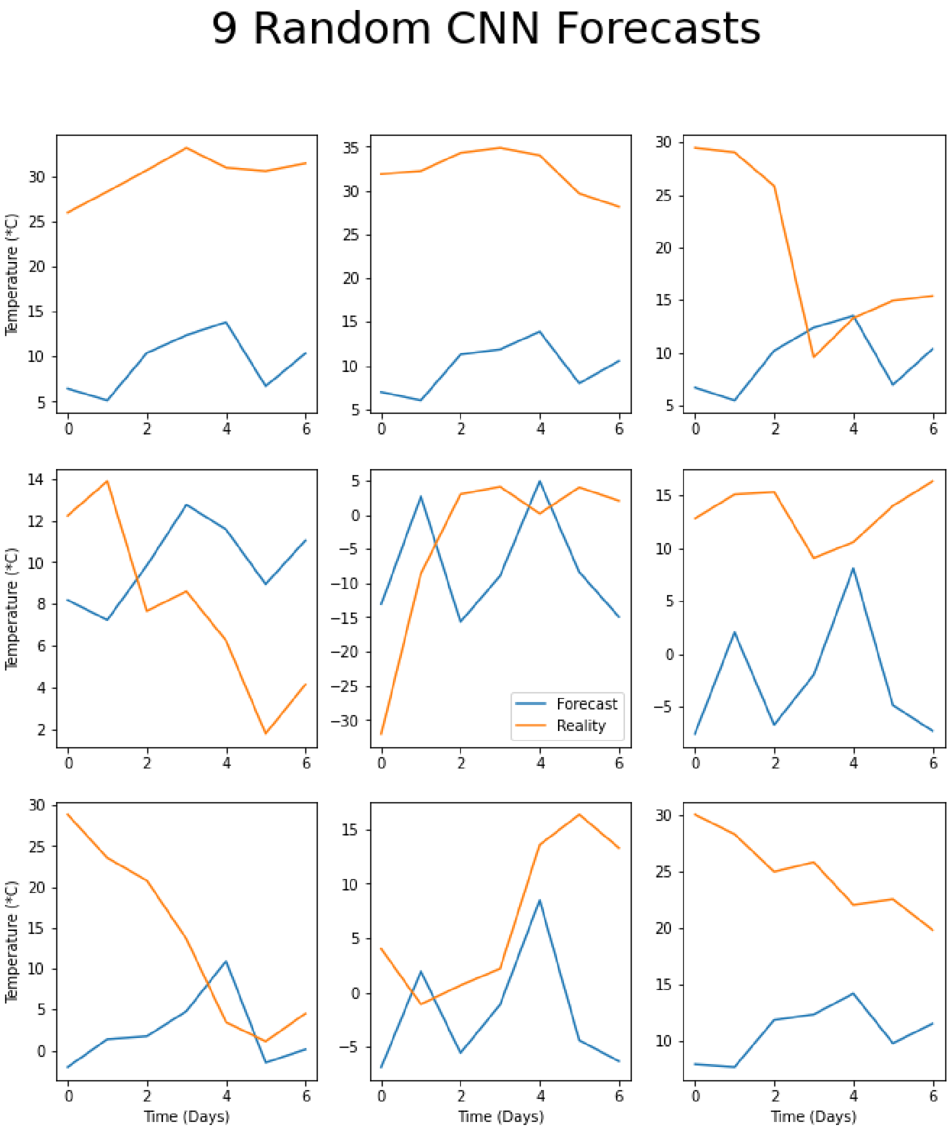
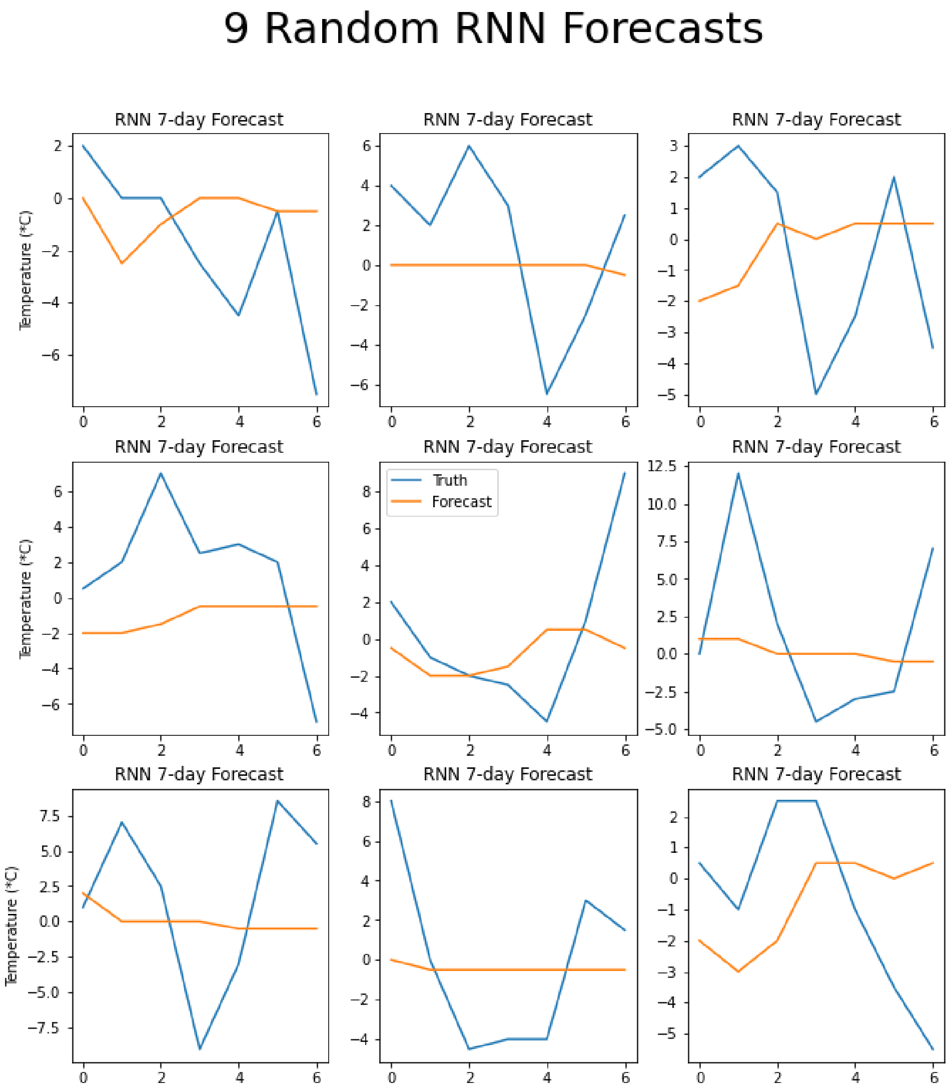


Figure 8: Selected Results (RNN)


9 Random RNN Forecasts
RNN 7-day Forecast
Temperature (*C)
RNN 7-day Forecast
RNN 7-day Forecast
Truth
Forecast