GEOL50130 Mini Project

CIS Username: bgtt65

1 Data Exploration

The Durham daily temperature record (from 1901 to 2019) is shown as a heatmap in Fig. 1. Obviously, the temperature data perform a strongly seasonal characteristic. For the left plot in Fig. 2, as the daily temperature is varied due to other weather conditions, it shows more unstable curves for each year. Therefore, the dataset was simplified to monthly temperature (see the right plot in Fig. 2) which is with periodic curves. As the monthly dataset is more predictable, it is a good start to deal with this short period data at the beginning and then extend to the daily temperature forecast.

The approach of this project is to use a simplified dataset of monthly temperature to generate a series of procedures for estimating the 2020 monthly temperature by classic models firstly, then follow the same procedures to predict the 2020 daily temperature using the chosen model and find whether there is any limitation. After that, attempt to improve the chosen model.

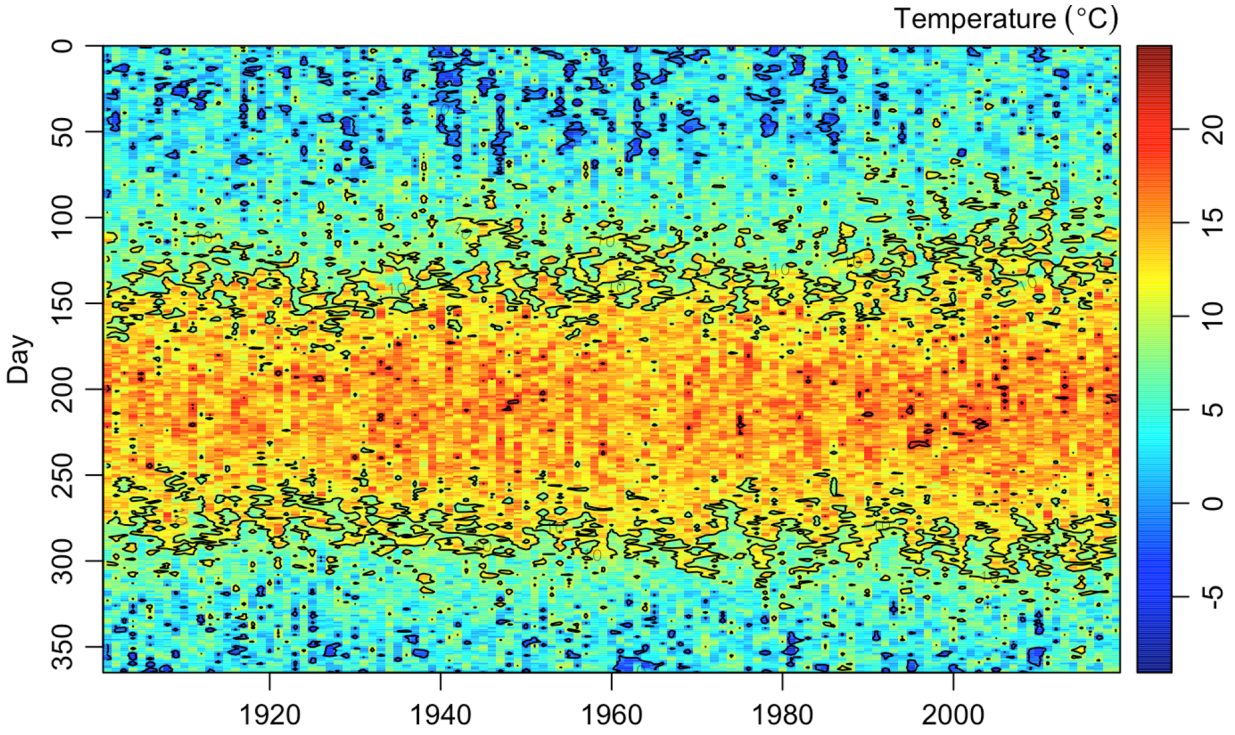

Fig. 1: Heatmap for Durham temperature

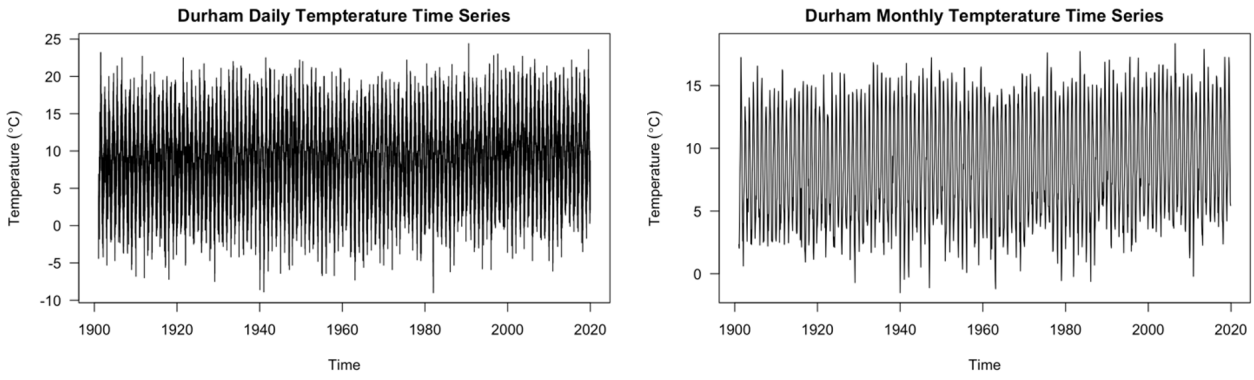

Fig. 2: Time series for Durham daily (left) and monthly(right) temperature

2 Monthly Temperature Forecast

Two strategies were considered to prove the credibility of the comparison. In the train/test split strategy, the whole dataset was split into a training set (80% data, from 1901 to 1995) and a test set (20% data, from 1996 to 2019) for model fitting and prediction. For the cross-validation (CV) strategy, to reduce the complicated computation, the observations from 2015 to 2019 and 6-step were chosen.

To evaluate the reliability of the training models, mean, naïve, seasonal naïve, and draft methods were applied for setting a baseline. The accuracy and results for both strategies are shown in Table 1, Fig. 3, and Fig. 4, all the values in seasonal naïve are the lowest compared to other methods. It is no doubt that the seasonal naïve is the best method here to make a standard for following training models.

Method	**Root mean squared error (RMSE)**				
Strategy:	Train/test split				Cross-validation
Mean	Training	4.452	Test	4.427	4.741
Naïve	Training	2.745	Test	8.806	6.440
Seasonal naïve	Training	1.800	Test	1.689	2.739
Draft	Training	2.745	Test	8.866	7.368

Table 1: Accuracy for basic methods in model fitting

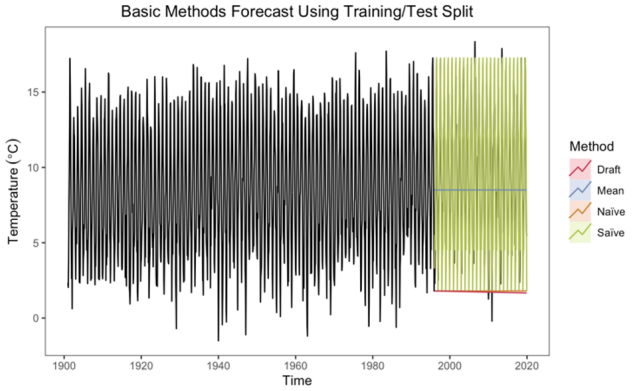

Fig. 3: Train/test split results for basic methods

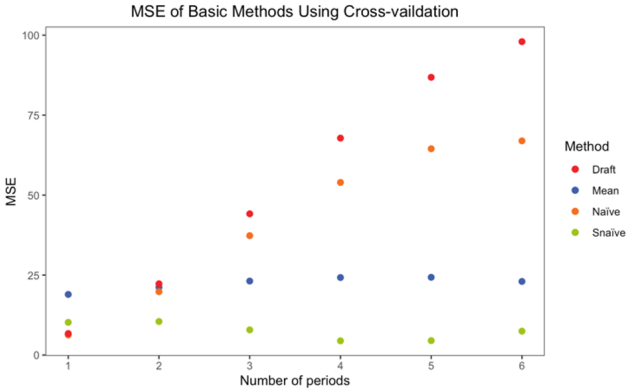

Fig. 4: MSE results for basic methods using CV

For the training models, some classic methods for predicting time series were chosen, including linear regression, ETS (Error, Trend, Seasonal) model in exponential smoothing, ARIMA, STL (Seasonal and Trend decomposition using Loess) decomposition and neural network. The STL model combines ETS or ARIMA models in order to forecast the seasonally adjusted component.

The fitting results for baseline and training models are shown in Fig. 5. As for Fig. 6, the regression and both two STL models remain a steady result, whereas the MSE of other typical methods using CV goes up with each increase in the period. As for Table 2, all the RMSE values of training models using train/test split are smaller than the baseline. The neural network performs well on training set but gets a bad result on test set which means the model is overfitting. After comparison, the STL + ETS model is utilized to predict the 2020 monthly temperature (see Fig. 7) as it has the smallest RMSE on test set.

Method	**Root mean squared error (RMSE)**				
Strategy:	Train/test split				Cross-validation
Seasonal naïve (baseline)	Training	1.800	Test	1.689	2.739
Linear regression	Training	1.279	Test	1.271	1.629
ETS	Training	1.276	Test	1.267	5.520
ARIMA	Training	1.436	Test	1.306	4.520
STL + ETS	Training	1.161	Test	1.236	1.456
STL + ARIMA	Training	1.133	Test	1.333	1.332
Neural Network	Training	0.657	Test	1.646	3.715

Table 2: Accuracy for typical methods in model fitting

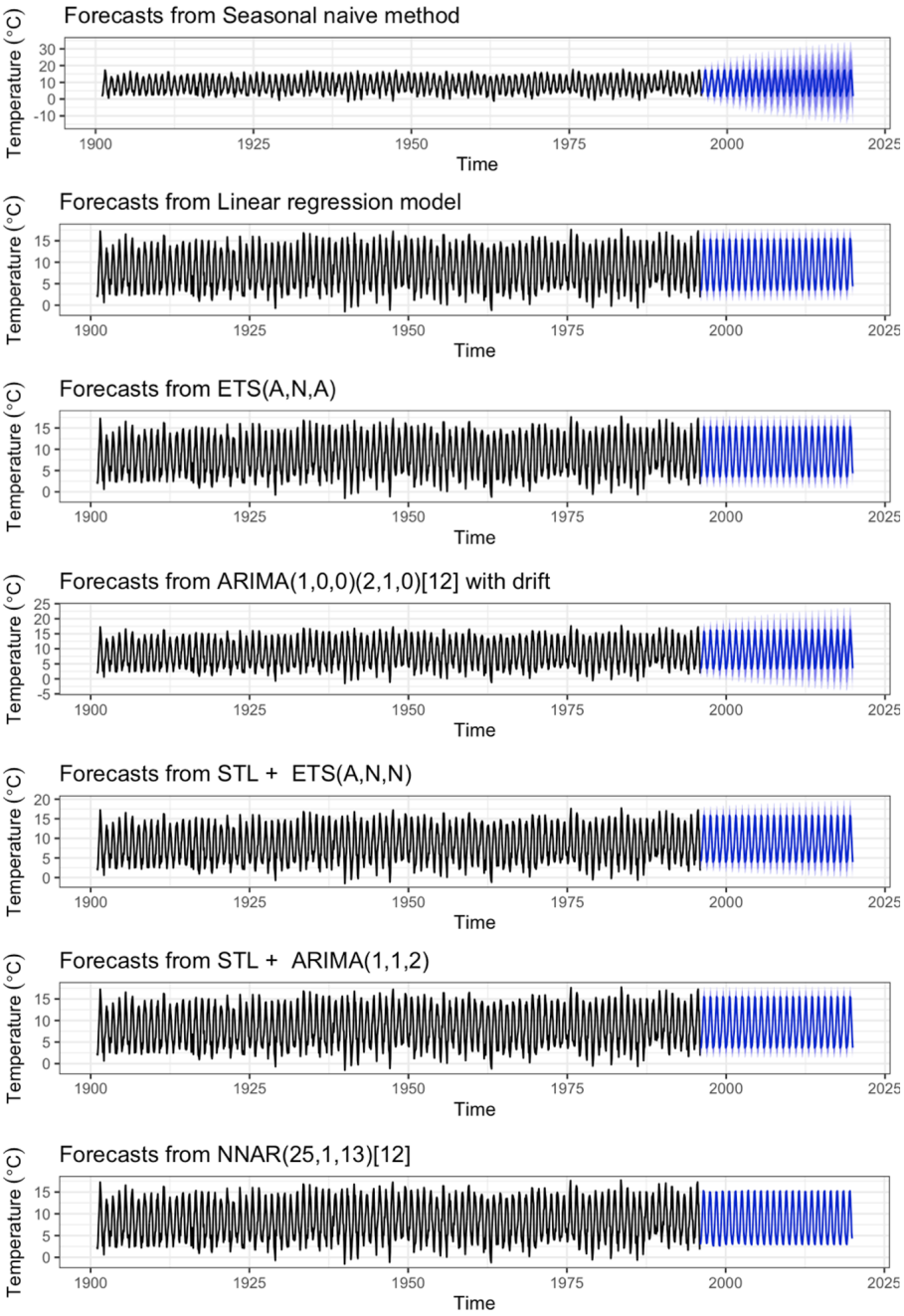

Fig. 5: Train/test split results for baseline and training models

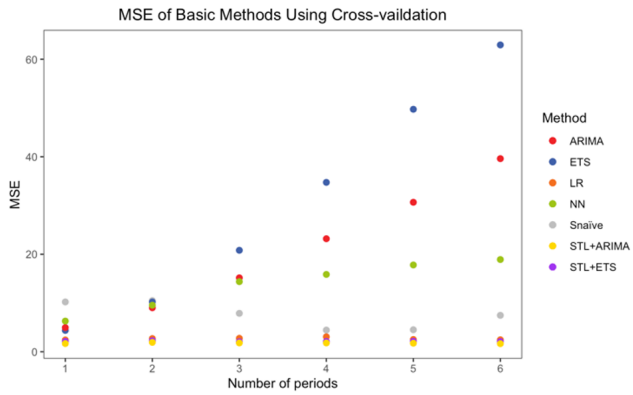

Fig. 6: MSE results for training models using CV

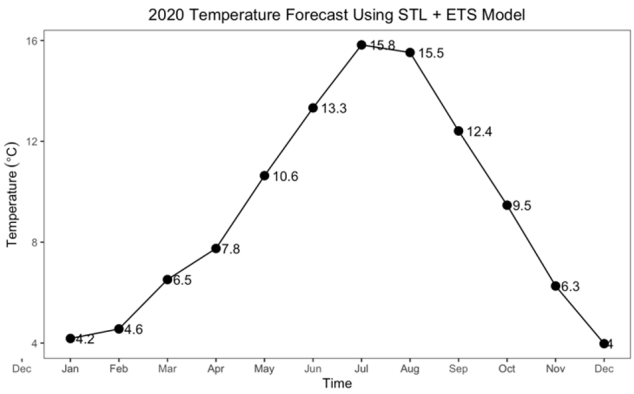

Fig. 7: 2020 monthly temperature forecast

3 Daily Temperature Forecast

Compared to monthly temperature, the period of daily record is over 30 times. There is a limitation to using CV strategy because of the long runtime. Therefore, only the train/test split was considered. Avoiding the complex computation, a training set (from 2010 to 2017) and a test set (from 2018 to 2019) were chosen. From the accuracy and results in Table 3 and Fig. 8, the seasonal naïve is applied for setting a baseline.

Method	Root mean squared error (RMSE) for train/test split			
Mean	Training	4.931	Test	5.187
Naïve	Training	2.053	Test	8.142
Seasonal naïve	Training	3.832	Test	3.735
Draft	Training	2.053	Test	7.758

Table 3: Accuracy for basic methods in model fitting

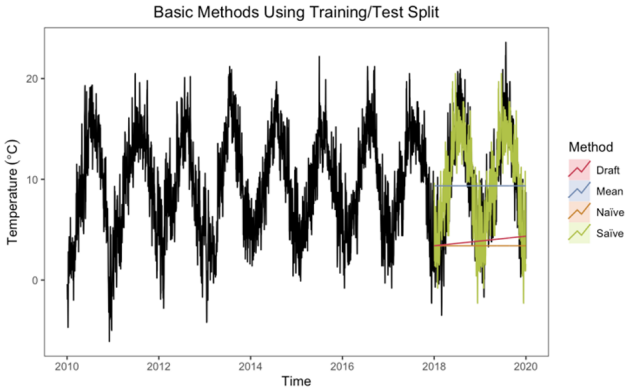

Fig. 8: Train/test split results for basic methods

Following the same step to fit the training models, another limitation of this approach is that the linear regression, ETS model, and neural network cannot work well due to the large frequency of seasonality. The accuracy and results for other models are shown in Fig. 9 and Table 4. The ARIMA model takes a long time to fit and doesn't perform well at its error. Overall, the STL + ARIMA model shows a better performance, and it is used for predicting 2020 daily temperature.

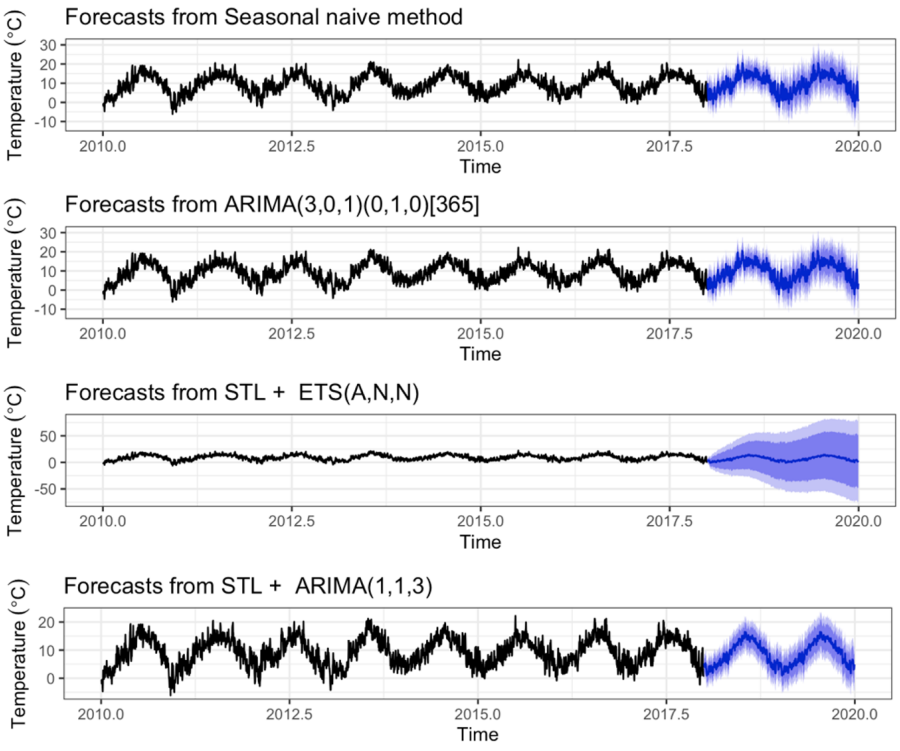

Fig. 9: Train/test split results for baseline and training models

Method		Root mean squared error (RMSE)		
Seasonal naïve (baseline)	Training	3.832	Test	3.735
ARIMA	Training	2.450	Test	3.705
STL + ETS	Training	1.808	Test	3.233
STL + ARIMA	Training	1.685	Test	2.994

Table 4: Accuracy for typical methods in model fitting

After the comparison, the baseline and the chosen model are refitted using the whole dataset by train/test split (see Fig. 10). In Table. 5, the RMSE of the STL + ARIMA model is slightly higher than using the partial dataset, but it is better than the baseline. According to the refitting model, the 2020 daily temperature is predicted in Fig. 11 and Fig. 12.

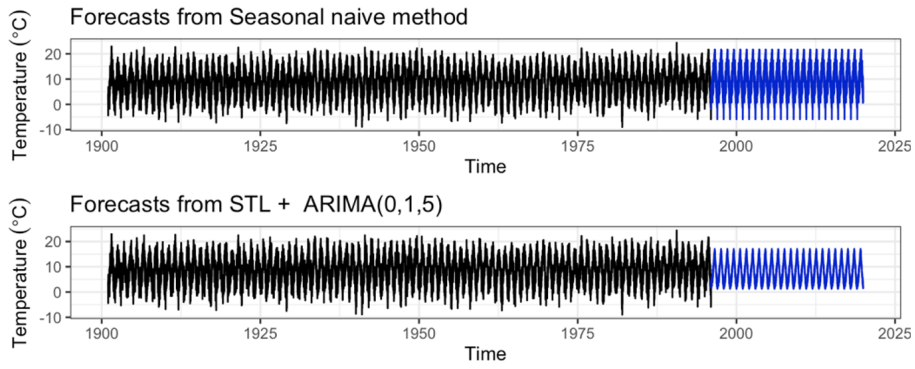

Fig. 10: Train/test split results for baseline and the chosen model

Method		Root mean squared error (RMSE)		
Seasonal naïve (baseline)	Training	3.864	Test	4.095
STL + ARIMA	Training	1.720	Test	3.087

Table 5: Accuracy for baseline and the chosen model in model fitting

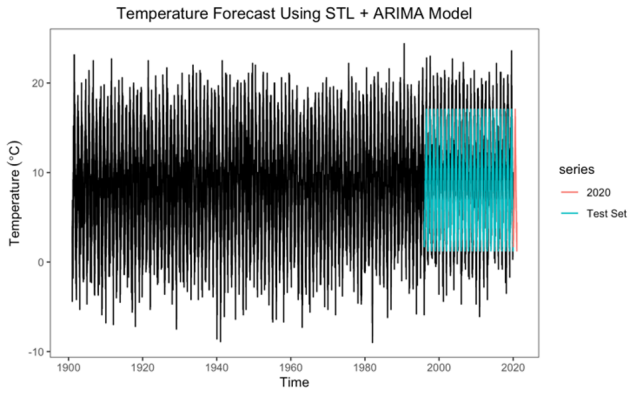

Fig. 11: Refit results for chosen model

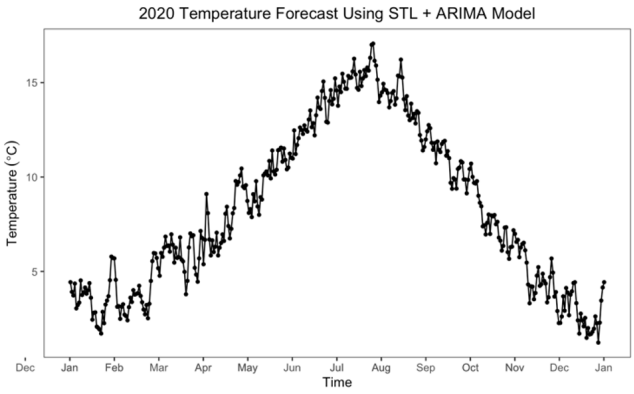

Fig. 12: 2020 daily temperature forecast (STL+ARIMA)

However, in this chosen model, the accuracy is still high, and a clear limitation is its weakness in detecting the trend. To improve this problem, the Facebook Prophet model was introduced. In Fig. 13, the Prophet model recognizes there is a slightly increasing trend in the dataset. Although the RMSE value on training set of the Prophet model is higher than the chosen model, a smaller error on test set is found in Table 6.

Compared to the classic models, Prophet model shows its potential to detect even a slight trend. The 2020 daily temperature forecast using this model can be seen in Fig. 14, and the result is selected as the final prediction solution because the accuracy is the standard in this report. But if look at the efficiency, the STL + ARIMA model can fit and predict faster. Therefore, sometimes the model choosing should consider the purpose of the task.

Method		Root mean squared error (RMSE)		
STL + ARIMA	Training	1.720	Test	3.087
Prophet	Training	2.719	Test	2.672

Table 6: Accuracy for STL +ARIMA and Prophet model in model refitting

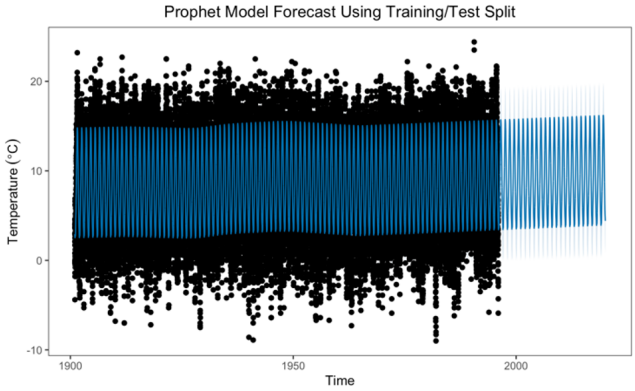

Fig. 13: Fitting results for Prophet model

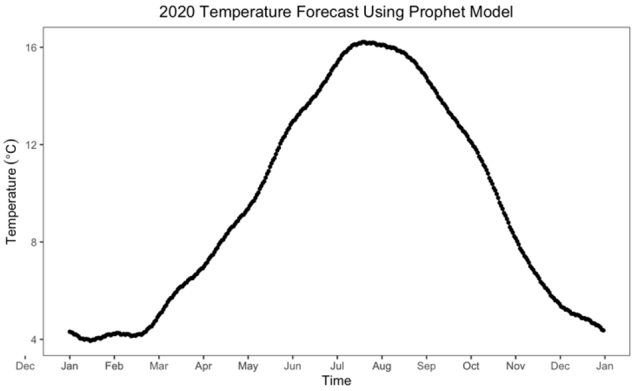

Fig. 14: 2020 daily temperature forecast (Prophet)

4 Conclusion

With the experimental data from the performance of fitting models, the 2020 daily temperature was predicted. For the approach of this report, the limitations of validation strategy and model chosen for long period dataset were found.

In the future, there are three ideas to improve the final prediction. Firstly, the transformation method can be applied to select a particular type of power transform which can remove noise from data. Secondly, to explore the nature of the data comprehensively, individual forecasting models with different characteristics can be combined into a hybrid model. Normally, a hybrid model grabs advantages from classic models and provides a more accurate forecast. In addition, other weather conditions such as wind, precipitation, sunlight, and clouds interact with temperature, it should be useful if these weather conditions can be collected and added to the dataset for model training.

Reference

Hyndman, R.J., & Athanasopoulos, G. (2018) Forecasting: principles and practice, 2nd edition, OTexts: Melbourne, Australia. Accessed on https://otexts.com/fpp2

Coghlan, A. (2018) Using R for Time Series Analysis. Accessed on https://a-little-book-of-r-for-time-series.readthedocs.io/en/latest/src/timeseries.html

Robson, W. (2019) The Math of Prophet. Accessed on https://medium.com/future-vision/the-math-of-prophet-46864fa9c55a