Time Series of Max Temperatures in Four Cities in India

International news in recent years has reported on the record setting high temperatures in India during the summer, with such viral images of pedestrians whose melting shoes stick to the hot pavement. The objective of this analysis is to examine data from previous summers and attempt to predict max temperatures for future summers.

DATA

The data contains daily weather data for fifteen of the most populous cities in India. The daily weather data is only shown for the summer months (April 1 - June 30) for the years 2007 through 2021. There are six temperature-related variables, which are the average, minimum and maximum for both temperature, and 'feelslike.' As I was interested in the record setting high temperatures as of recently, I decided to model the variable for maximum temperature.

In addition to the 'tempmax' variable, the only other variables retained were those indicating date and city. The only variable that could have been retained to be used as an exogenous variable in a SARIMAX model is a float variable for moon phase. It is the only variable that would be known for future periods, and therefore be used to make predictions. I opted not to include it because–as will be discussed below–I decided to aggregate the daily data to a weekly time period. Aggregating a daily moon phase by week would not be particularly informative and could confuse results.

I also retained data for only four of the fifteen cities. I chose the four using the criteria of most populous cities that represent four diverse regions of the country. The four I chose were: New Delhi, Kolkata (Calcutta), Mumbai, and Bengaluru (Bangalore).

EDA & DATA CLEANING

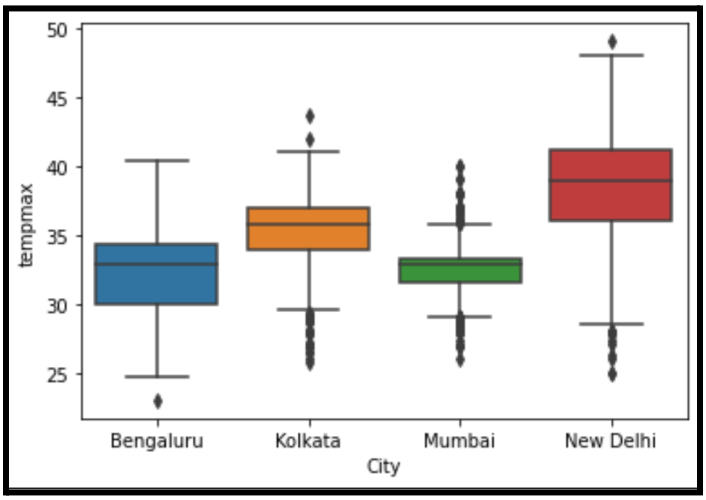
The variables 'City', 'Date' and 'tempmax' were read in as *object* data types. Therefore the first step was to convert 'Date' to *DateTime64* and 'tempmax' to *float64*.

I checked for nulls using '.info', and as shown below there were no nulls in the data.

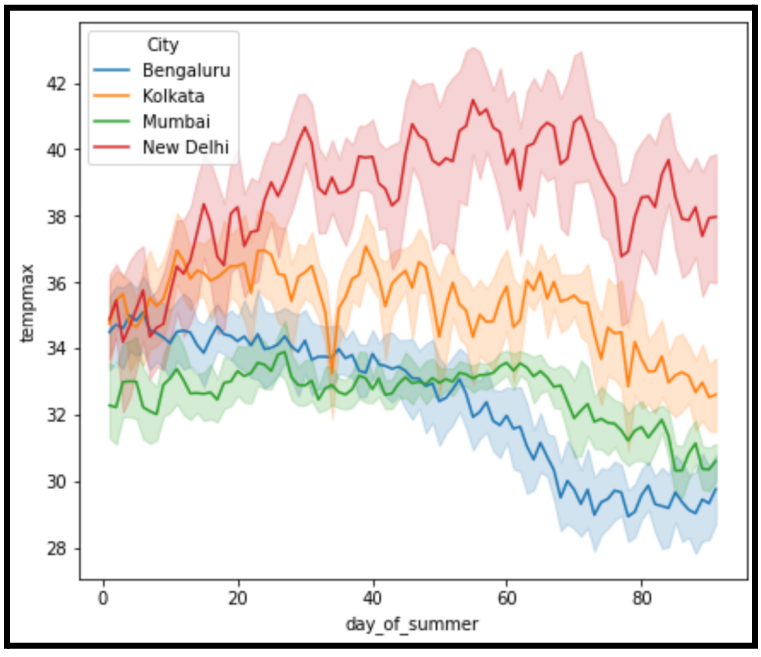
```
#   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   City     5459 non-null   object
 1   Date     5459 non-null   datetime64[ns]
 2   tempmax  5459 non-null   float64
```

I created a variable, called 'day_of_summer' that assigns an integer value between 1 and 91 to represent the dates in the data shown. Therefore April 1st for each year will be represented by 1 and June 30th will be represented by 91. I also created a variable that only shows the year. It will be easier to work with the data using these two variables instead of the 'Date' variable.

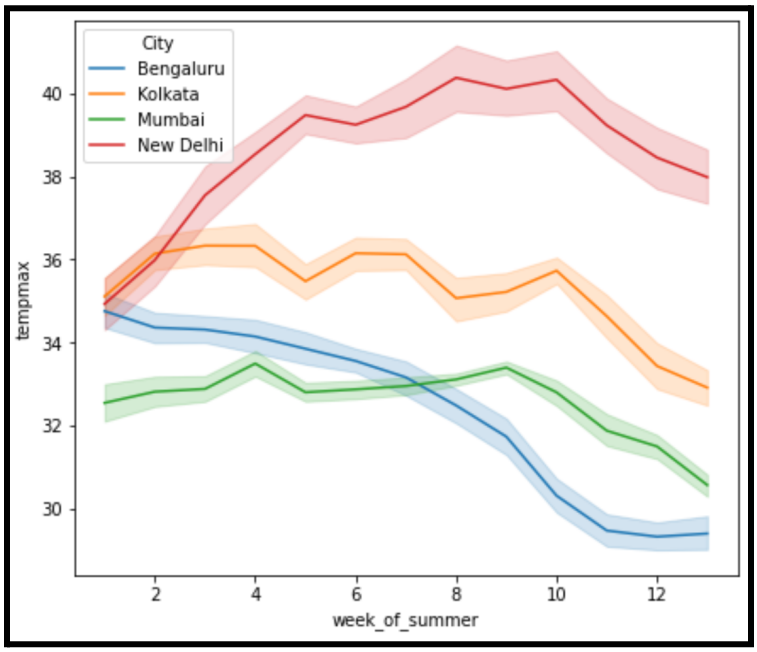
I created a box plot to show the ranges of 'tempmax' for each city, and to inspect whether any of the outlier data points fall outside the range that one would expect for maximum temperatures in India. The plot shows that none of the outlier data points appear to be errors in the data. The plot also shows the max temperatures for Kolkata and Mumbai mostly fall within a narrow range, while the temperatures for Bengaluru and especially New Delhi cover a wide range.



The variability seen in the box plot can be viewed in more detail by using a line plot that aggregates the data for each 'day_of_summer' and shows the confidence intervals. We see that at one extreme, Mumbai–in comparison to the other cities–has a relatively smooth plot with few extreme jumps from day to day and the confidence intervals are relatively narrow.



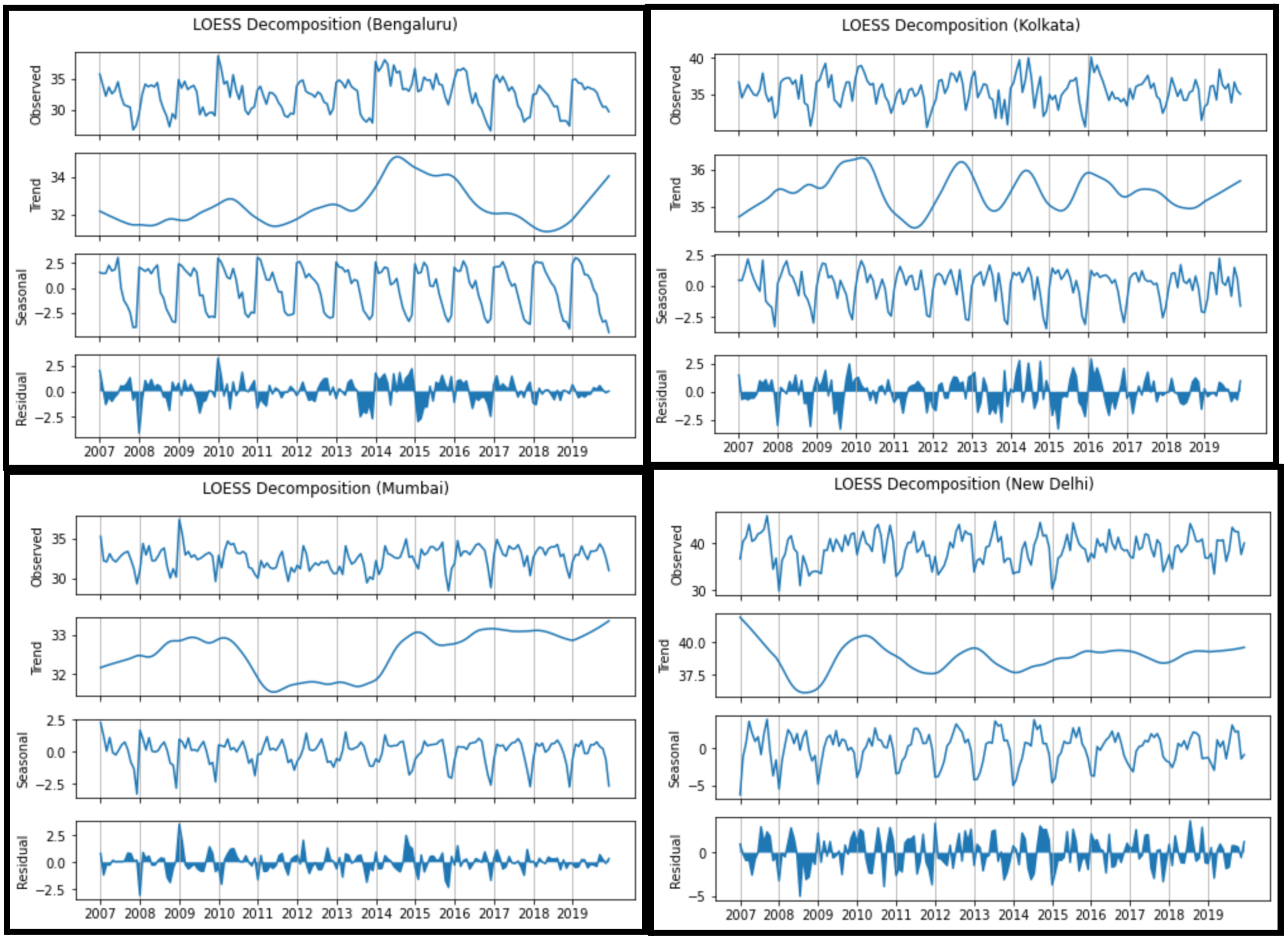
This indicates the max temperatures are relatively consistent from day to day and year to year. New Delhi represents the opposite extreme where the graph is relatively jagged with large jumps day to day and wide confidence intervals indicating inconsistency from year to year.



It is impractical to work with daily data for purposes of decomposing the data into trend, seasonality and residuals, and ultimately making predictions. Even in the best case scenarios daily data is too noisy to accurately make predictions. It would be better to reduce the resolution to weekly data. I, therefore, created a variable for 'week_of_summer' which ranges between week 1 through week 13. The 'tempmax' field was then aggregated by taking the mean of the daily data. When plotting the line graph again using weeks instead of days we can get a sense of which cities may be easier to predict than others. Bengaluru has a smooth curve with narrow confidence intervals, indicating that its pattern of max temperatures is relatively consistent and, therefore, easier to predict. Kolkata, on the other hand, shows a more jagged plot, with multiple peaks and valleys which may indicate a lack of consistency that could make predictions more difficult. New Delhi, whose plot is relatively smooth, nevertheless shows wide confidence intervals which could also make it difficult to predict.

MODEL #1 - LOESS + ARMA

For the first model I wanted to decompose the time-series data of max temperature into its trend, seasonal and residual components using LOESS, then use the LOESS decomposition to make predictions. To do this, I first separated the last two years of the data (2020-2021) from the rest of the data in order to create train and test datasets.



The decomposition of Bengaluru shows a relatively consistent seasonal component. The trend is also relatively flat, with the exception of the section covering years 2014 and 2015 where the trend is elevated by one to two degrees.

The decomposition of Kolkata shows the seasonal component for the last three years differs from those preceding it in that the lower temperatures at the end of the season do not dip as low as they did in the preceding years. The trend component fluctuates often between 35 and 36 degrees.

The decomposition of Mumbai shows the seasonal component for years 2010-2015 differs from the other periods in that the lower temperatures at the end of the season do not dip as low as they did in the years surrounding this period. The trend also dips by roughly one degree during the same time period.

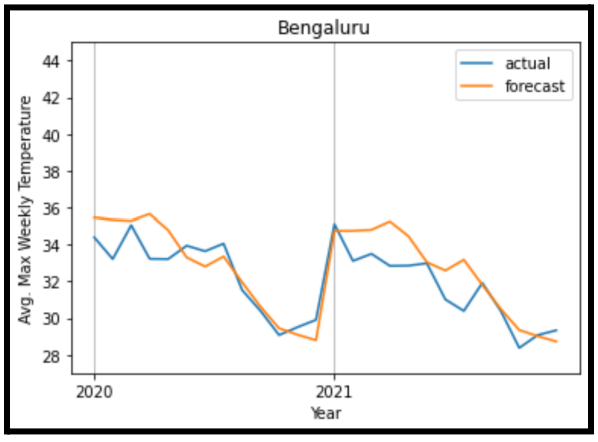
The decomposition of New Delhi shows the seasonal component maintaining the same periods of higher temperatures, however the periods of lower temperature at the beginning and end of the season become progressively higher. The trend component is more volatile during the first seven years then flattens out around 38-39 degrees in subsequent years.

Using the LOESS decomposed time series data, I needed to fit an ARMA model on the sum of the trend and residual components (i.e. seasonally adjusted data). This is necessary in order to make the data stationary before running the ARMA model. At this point I needed to establish the autoregressive and moving average parameters for the ARMA (p,q) order. I used a function to model different combinations of *p* and *q* using AIC as the information criteria to determine which combination is the best fit for each city's data. The results are shown below.

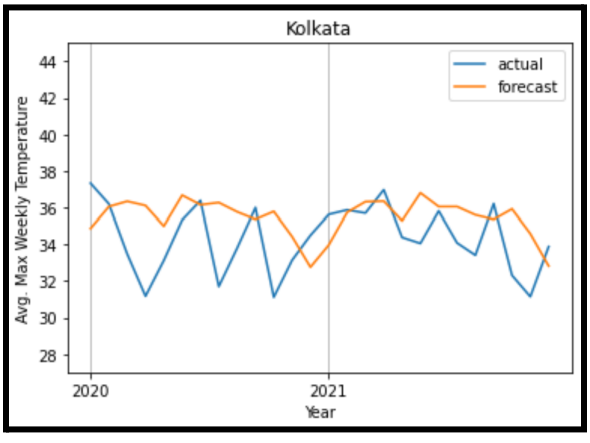
```
Bengaluru: ARMA_order=(2, 1)
Kolkata: ARMA_order=(2, 1)
Mumbai: ARMA_order=(2, 1)
New Delhi: ARMA_order=(2, 1)
```

At this point we could create ARMA models to make forecasts for 2020-2021 for each city. I use mean average percentage error (MAPE) to gauge how well the forecasts approximate the actual test data. Below are the MAPE scores along with the graph of the forecast against the actual data.

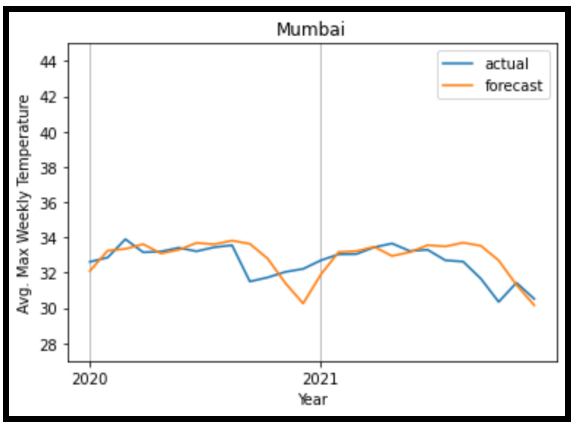
```
Bengaluru: MAPE=0.030939816220681055
```



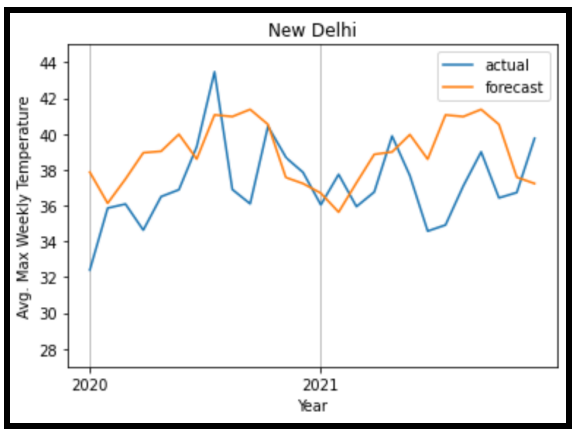

```
Kolkata: MAPE=0.05462311475320937
```



Mumbai: MAPE=0.020721438895428657



New Delhi: MAPE=0.06682889510189649



The models for Bengaluru and Mumbai were able to detect the seasonall contour. As discussed previously, there is less variance week to week and year to year for these cities relative to the other two. Because of this consistency, we should have a better idea of what data for future periods will look like. The graphs and MAPE scores confirm this is the case. On the other end of the spectrum, the max temperatures for Kolkata and especially New Delhi are highly variable week to week and year to year. We see from the graphs and the MAPE scores that the models were not as accurate for these two cities as the actual data exhibits a higher degree of volatility than the other cities.

MODEL #2 - SARIMA

For my second model I decided to use the SARIMA method where I would model the seasonality using a seasonal *P, D, Q* order. I used a function that determines the optimal *p,d,q* order and seasonal *P,D,Q* order by testing which combination has the lowest AIC. Below shows the order for each city (the last variable in the seasonal order is *m*, which indicates the number of periods in each seasonal cycles which in this case is thirteen weeks).

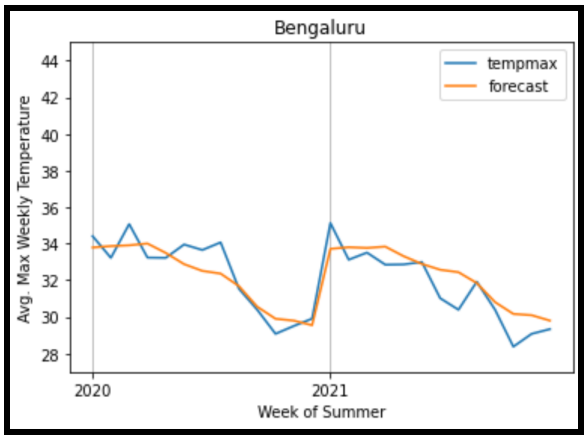
```
Bengaluru: Order=(2, 0, 0), Seasonal Order=(1, 0, 1, 13)
Kolkata: Order=(1, 0, 0), Seasonal Order=(1, 0, 0, 13)
Mumbai: Order=(1, 0, 0), Seasonal Order=(3, 0, 1, 13)
New Delhi: Order=(1, 0, 0), Seasonal Order=(0, 0, 0, 13)
```

Of particular note is that the *D* and *Q* for Kolkata are both zero and the *P* is only one. In addition, New Delhi's *P, D* and *Q* are all zero. This suggests that the most accurate models for these two cities are ones that have very little to no seasonal order. This is

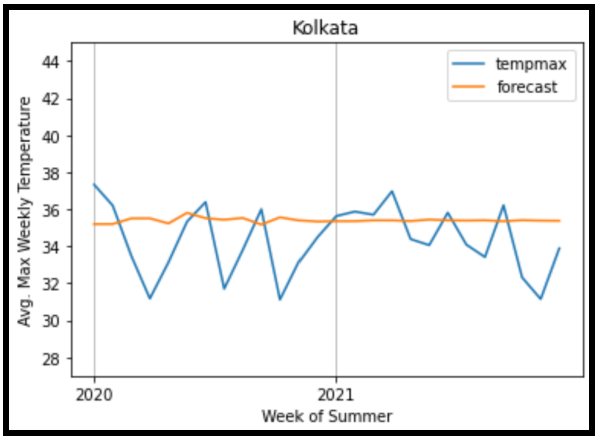
not entirely surprising given that we've seen that the max temperature for these cities are highly variable and inconsistent from year to year.

Once the orders are established, the SARIMA model makes forecasts for the two years in the test period.

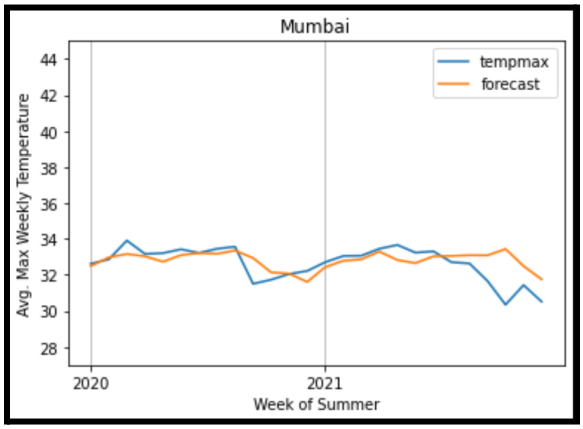
`Bengaluru: MAPE=0.024944717889173906`



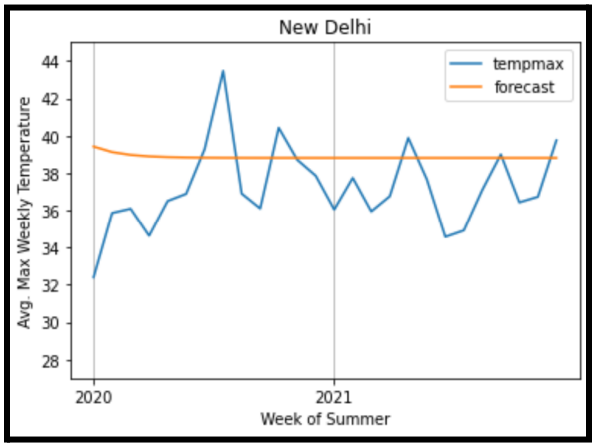
`Kolkata: MAPE=0.05087210250959146`



`Mumbai: MAPE=0.017755390433985968`



`New Delhi: MAPE=0.06265311898193084`



We can see from the plots of the forecasts for Bengaluru and Mumbai do a relatively good job of predicting the seasonal contour of the max temperatures. Conversely, Kolkata and New Delhi show predictions that are almost completely flat from week to week. This is not surprising given that we saw above little to no seasonal SARIMA order for these cities.

CONCLUSION

We see from the MAPE scores from the two models that SARIMA performed slightly better for all four of the cities. The LOESS + ARIMA model tended to overfit to the historical data while the SARIMA model better segregated the seasonal components from the noise. More broadly speaking, both models indicate the difficulty in forecasting the max temperatures for cities like Kolkata and New Delhi with highly variable and inconsistent seasonal patterns.

This model could potentially be improved by use of exogenous variables. There may be trends relating to concentration of certain chemical compounds in the atmosphere, changes in sea level, etc. that could be forecasted for the future then used in a SARIMAX model as exogenous variables used to predict max temperature.