Data Challenge 3

Tao Peng, UZH ID: 21-738-927

Network Analysis

1. Data Pre-processing

- If segments have the same pair of nodes, only the one with shorter distance is kept.

2. Create street graphs and dual graphs

- To create graphs, the package ‘networkx’ was utilized. I first defined a function, draw_nx, to create graphs once the graph information is provided.
- For the undirected weighted street graph, the weight is added using the function, [add_weighted_edges_from].
- To create dual graphs, I used the function [line_graph]. The line graph of a graph G has a node for each edge in G and an edge joining those nodes if the two edges in G share a common node. Therefore, line graph is the same as dual graph.
- The results are shown in Figure 1.

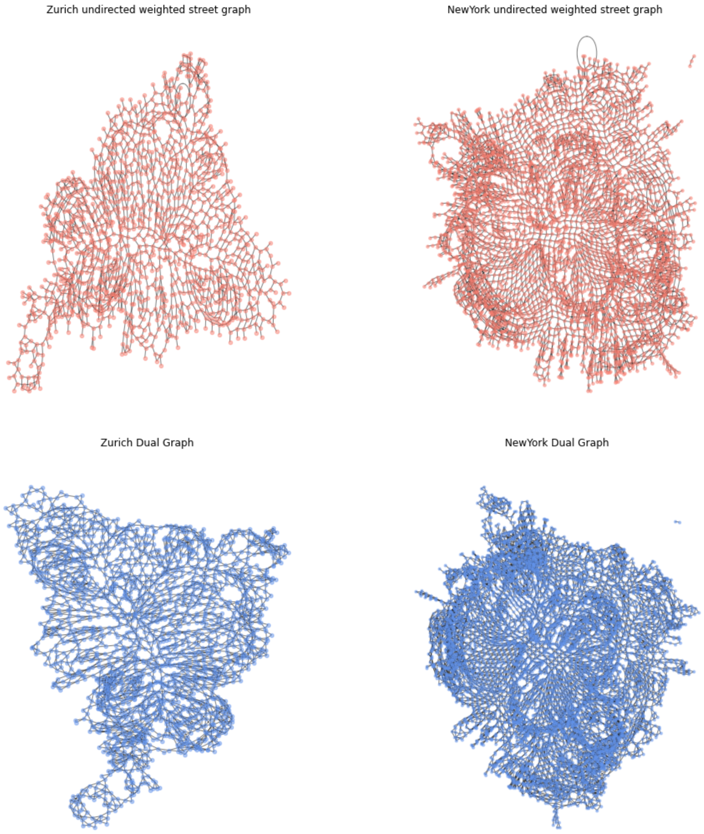

Zurich undirected weighted street graph
NewYork undirected weighted street graph
Zurich Dual Graph
NewYork Dual Graph

Figure 1.1 Undirected weighted street graphs (top) &Dual graph (bottom)

3. Centrality calculation of the graphs

- To calculate the degree, closeness, betweenness centralities, I used the functions [degree_centrality], [nx.closeness_centrality], [nx.betweenness_centrality] from ‘networks’. I defined a function, cal_centrality, which returns a list of lists with the three centralities.

- To visualize the centraliy, I defined a function, plot_centrality, which shows the value of centrality of each vertex by colors.

- The results of the centralities for NewYork and Zurich are shown in Figure 1.3.

4. Visual and numerical analytics

To use map to summarize the characteristics of centrality calculation, the following four kinds of maps are visualized for both cities. The results are shown in Figure 1.3 & 1.4 & 1.5.

- The maps of the road network
- The centralities of street graph with points position information
- The centralities of street graph
- The centralities of dual graph

To use statistics to summarize the characteristics of centrality calculation, the cumulative distribution (CDF) of the centralities three kinds of graphs is visualized for both cities:

- The CDF of centralities of street graph
- The CDF of centralities of dualgraph

To use charts to summarize the characteristics of centrality calculation, the histogram of the centralities of the three kinds of graphs is visualized for both cities:

- The histogram of centralities of street graph
- The histogram of centralities of dualgraph

5. Briefly discuss of the results from the previous step

Comparing street graph and dual graph:

- Geographic features are retained in street graph but lost in dual graph, which only keep the topology features of the road. Dual graph allows for discovering hidden structural properties of road networks, such as the hierarchy of roads and the true connectivity of the road network.

- Street graph and dual graph reveal similar patterns for the road network for both cities. Degree and closeness centrality is larger in dual graph. Betweenness centrality is almost the same in dual graph and street graph, both of which show high betweenness in several nodes. Those could be the main roads that connect different part of the city.

Comparing Zurich and New York:

- The road network pattern of New York is grid. The road network pattern in Zurich is irregular. The possible reasons of the difference between the two cities could be of landscape, the number of populations, and mainly transportation methods. As we all know Zurich is mountainous with no more than 500 thousand people. While Manhattan is a flat island with a population of over 1.5 million.

- For the street graphs, the average degree and closeness centralities are higher in Zurich, which indicates that the intersections of the street can be influenced more easily, and each intersection are closer in Zurich. However, the average betweenness centrality in Zurich is smaller, which indicate that there are less roads in New York connecting different parts of the city.

- For the dual graph, the degree centralities are almost the same for both cities. The closeness centralities are higher, but the betweenness is smaller in in New York. The results are similar to street graphs

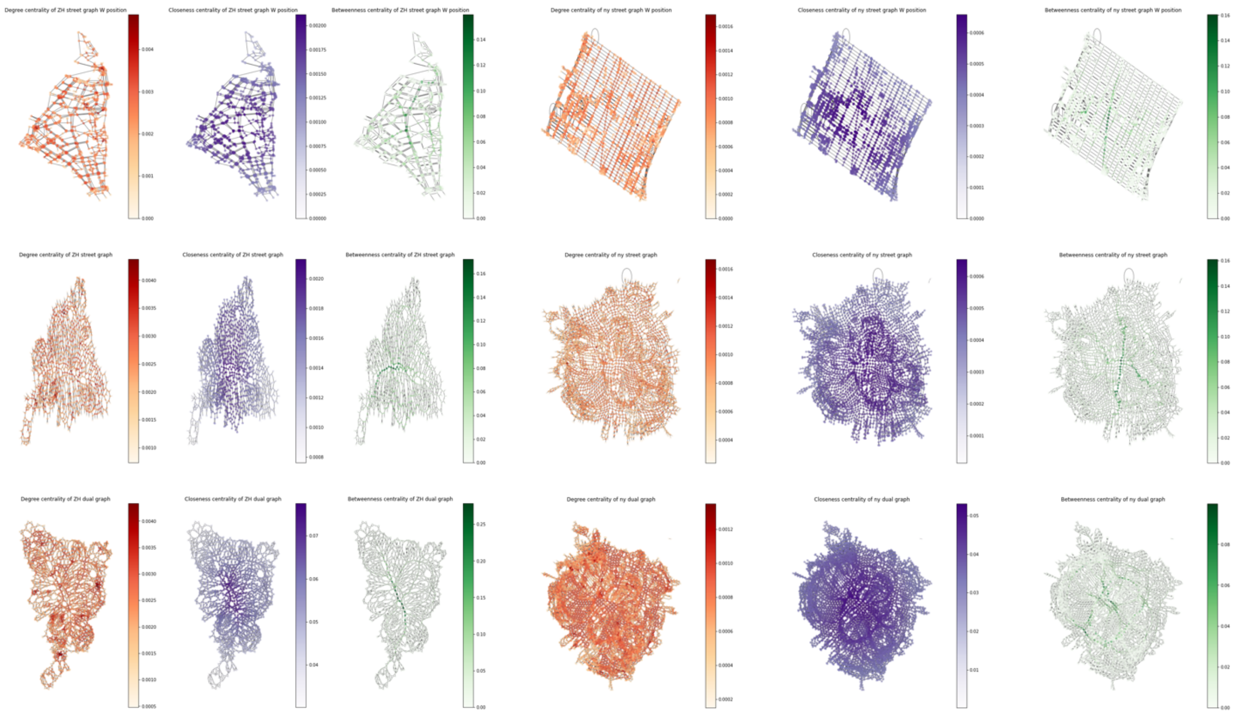
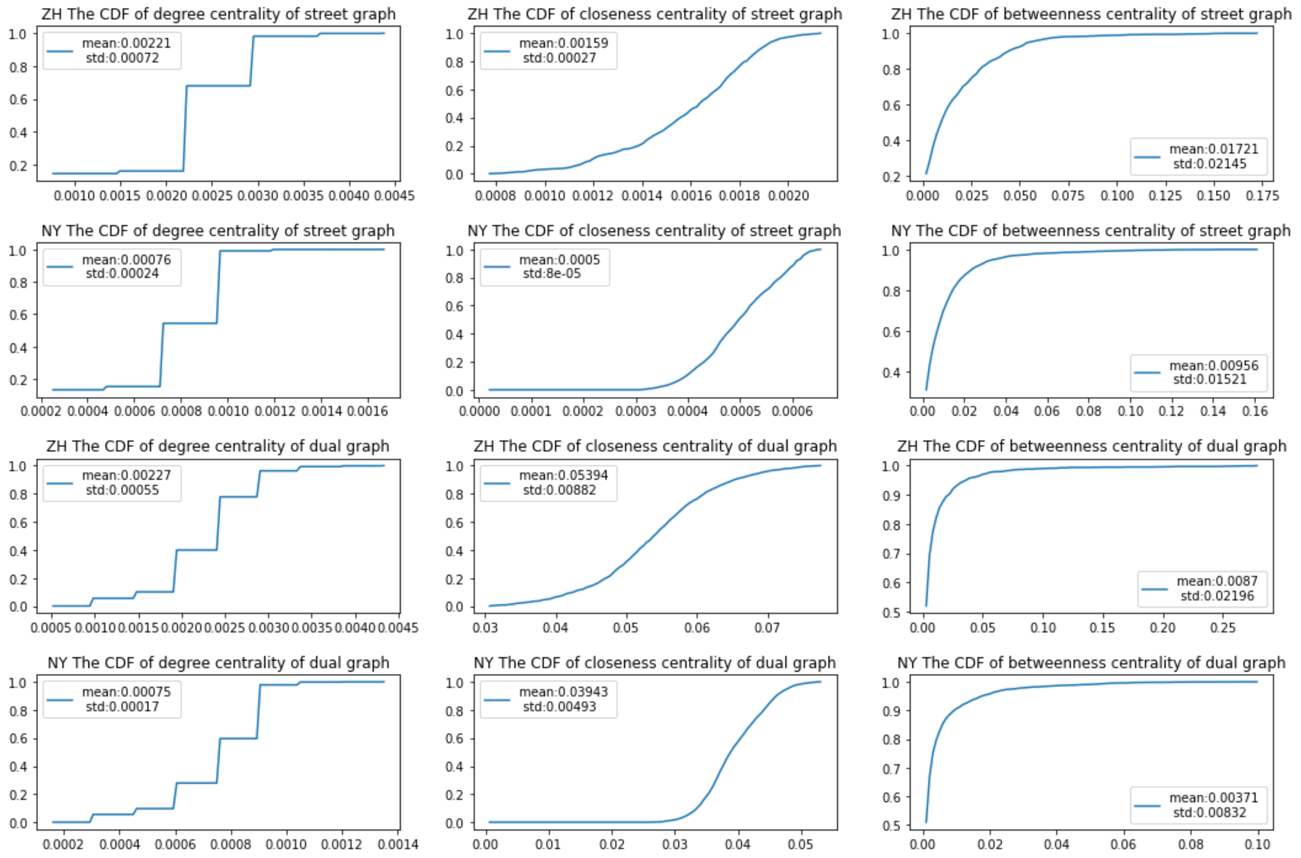

Degree centrality of ZH street graph W position
Closeness centrality of ZH street graph W position
Betweenness centrality of ZH street graph W position
Degree centrality of ny street graph W position
Closeness centrality of ny street graph W position
Betweenness centrality of ny street graph W position
Degree centrality of ZH street graph
Closeness centrality of ZH street graph
Betweenness centrality of ZH street graph
Degree centrality of ny street graph
Closeness centrality of ny street graph
Betweenness centrality of ny street graph
Degree centrality of ZH dual graph
Closeness centrality of ZH dual graph
Betweenness centrality of ZH dual graph
Degree centrality of ny dual graph
Closeness centrality of ny dual graph
Betweenness centrality of ny dual graph

Figure 1.3 Map of centralities



ZH The CDF of degree centrality of street graph

ZH The CDF of closeness centrality of street graph

ZH The CDF of betweenness centrality of street graph

NY The CDF of degree centrality of street graph

NY The CDF of closeness centrality of street graph

NY The CDF of betweenness centrality of street graph

ZH The CDF of degree centrality of dual graph

ZH The CDF of closeness centrality of dual graph

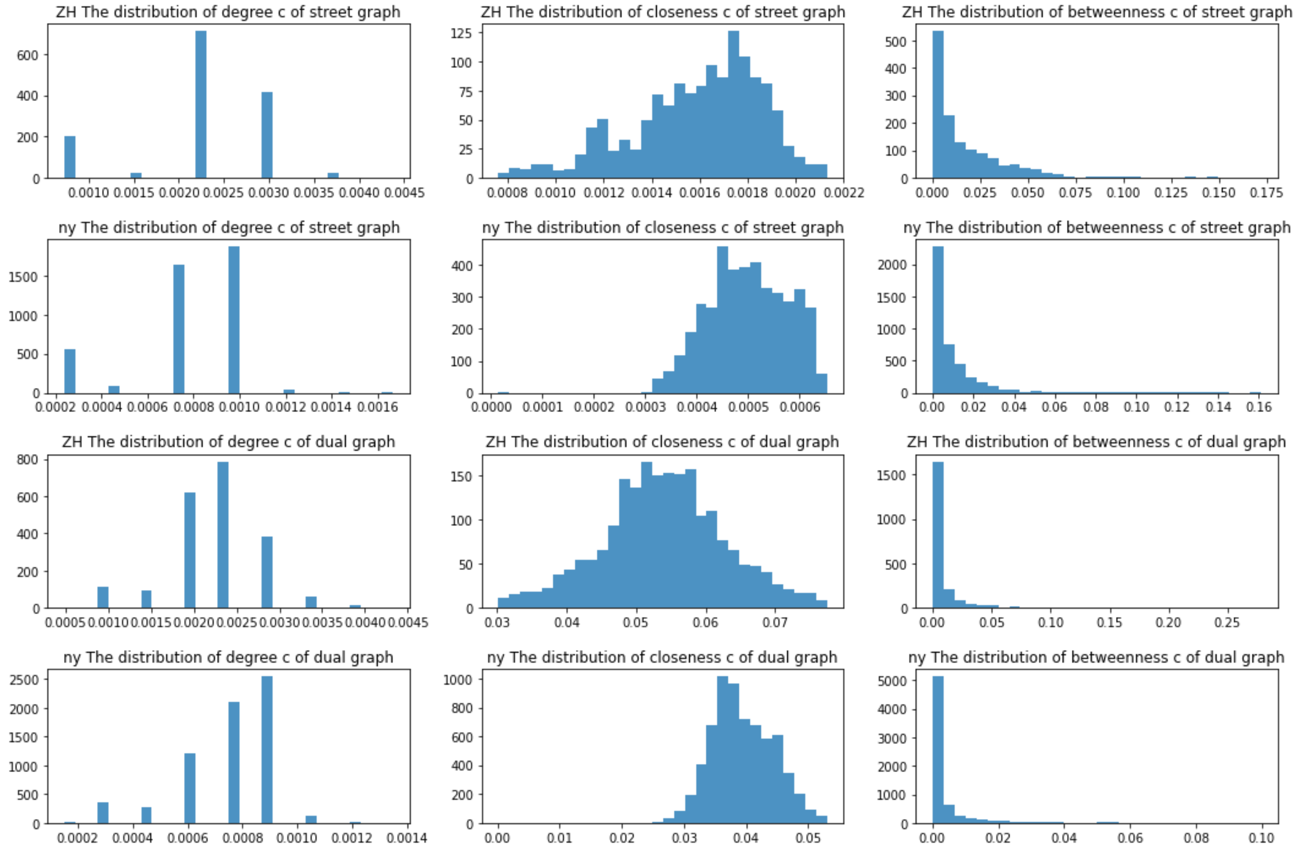
ZH The CDF of betweenness centrality of dual graph

NY The CDF of degree centrality of dual graph

NY The CDF of closeness centrality of dual graph

NY The CDF of betweenness centrality of dual graph

Figure 1.4 The cumulative distribution of centralities



ZH The distribution of degree c of street graph

ZH The distribution of closeness c of street graph

ZH The distribution of betweenness c of street graph

ny The distribution of degree c of street graph

ny The distribution of closeness c of street graph

ny The distribution of betweenness c of street graph

ZH The distribution of degree c of dual graph

ZH The distribution of closeness c of dual graph

ZH The distribution of betweenness c of dual graph

ny The distribution of degree c of dual graph

ny The distribution of closeness c of dual graph

ny The distribution of betweenness c of dual graph

Figure 1.5 The distribution of centralities

II Time Series Analysis

1. Analytics of the AirPassengers Data

1.1 Decomposition Analysis

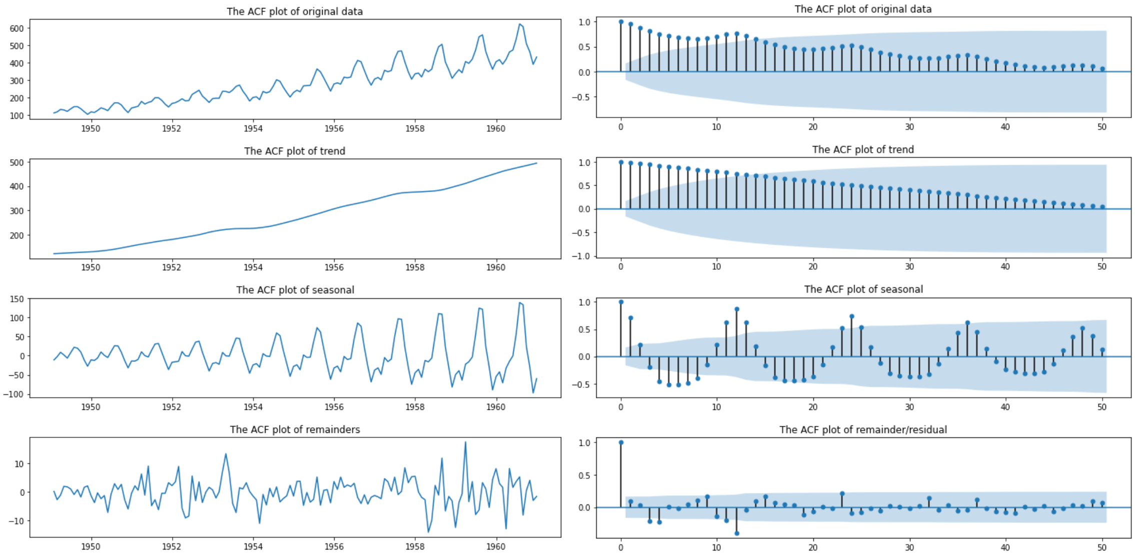


Figure 2.1 STL decomposition (left) and ACF (right)

1.1.1 Code Design

After loading the data, I converted the data to a pandas series (indexed by date). In the first step, STL decomposition, the function STL from statsmodels is applied to decompose the time series data into three components: trend, seasonal and residual (Figure 2.1). The second step is to conduct autocorrelation. The acf function from statsmodels is used for original data and the three components from STL decomposition (Figure 2.1). The third step is to explore the statistics of the residual, I plotted the histogram of the residual and normal quantile-quantile plot (QQ plot) (Figure 2.2).

1.1.2 Results & Discussion

STL decomposition:

- The data shows an increasing linear trend. The passenger numbers increase over this period.
- The data is seasonal.
- The residuals' variance seems to increase a little bit through time, showing that the series exhibits a little more random behavior at the end.
- The data is a multiplicative process, as the magnitude of the seasonal component changes with time & when the number of passengers increases, the seasonality increases as well.

ACF:

- The ACF of trend is decreasing. The data shows an increasing trend.
- The data shows a seasonality of 12 months, as the ACF coefficients have the max values at lag 1 or 12 months, indicates a positive relationship with the 12 months season. In each season, there are more passengers travelling in months 6 to 9.

Residual:

- The residual is white noise because the histogram shows a normal distribution
- The Normal QQ plot is close to 45-degree diagonal, which also indicate the residual is normally distributed.

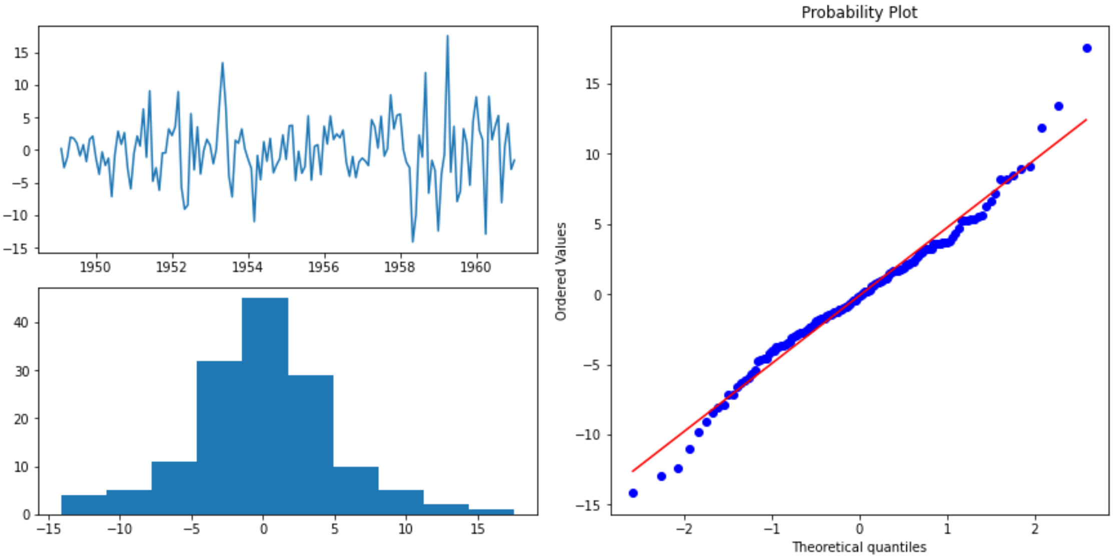


Figure 2.2 Residual analysis

1.2 Autoregressive Analysis

1.2.1 Code Design

Both Exponential Smoothing models and ARIMA models are used to forecast time series day. However, Exponential Smoothing models are appropriate for non-stationary data. ARIMA models should be used on stationary data only. My first step is to create a function (Stationarity) to check if data is stationary, employing two methods, rolling means & standard deviation, as well as ADF (Augmented Dickey-Fuller Test). As the original data is tested non-stationary, the second step is to apply the function ExponentialSmoothing on the original data.

To use the ARIMA model, the first step is to stationalize the data by differencing the logged data. The second step is to choose order (p, d, q). After applying ACF and PACF on the logged-first-difference data, lag-1 ACF and PACF are both positive. So, I use 2 for both MA order and AR order. As the logged-first-difference data is stationary, I use 1 for the differencing order. The logged-first-difference data is then used in the ARIMA model.

1.2.2 Results & Discussion

Exponential Smoothing and ARIMA models are used to predict the number of passengers in the next 5 years after 1960 (Figure 2.3).

A stationary time series is one whose statistical properties do not depend on the time at which the series is observed. Exponential Smoothing models are appropriate for non-stationary data. ARIMA models should be used on stationary data only. We can convert non-stationary data to stationary using differencing and logging.

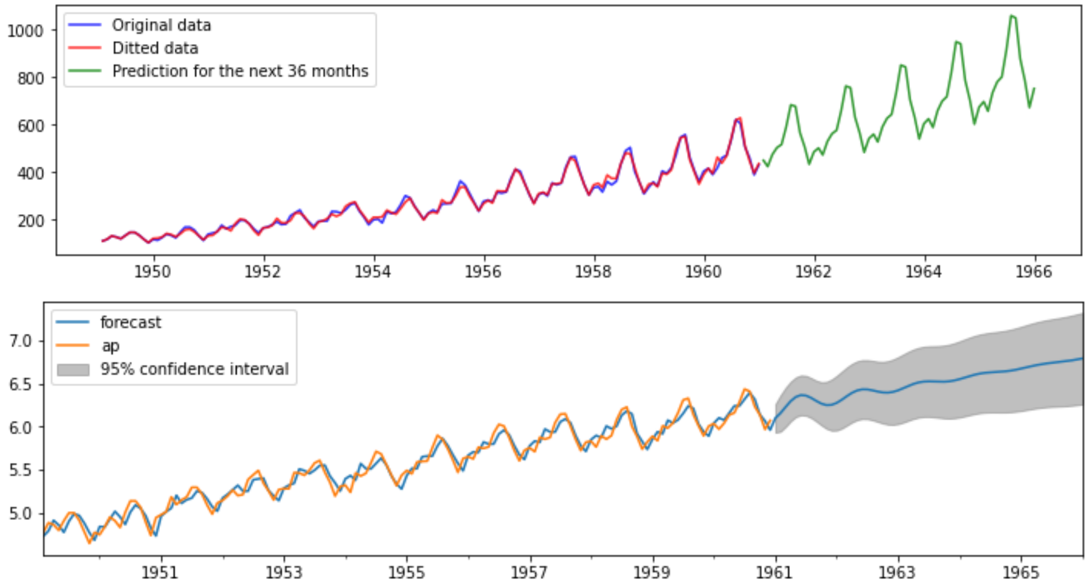


Figure 2.3 Exponential Smoothing model (top) and ARIMA model (bottom)

2. Analytics of the 2020 Pedestrian Data

2.1 Break Detection

2.1.1 Hypothesis

There could be 3 change points. The first change point could be near March 2020, as the first several measures were issued between the end of February to middle March 2020. The second change point could be near May 2020, as measures were gradually removed from late April to June 2020. The third change point could be in October, because of the measures in October 2020.

2.1.2 Code Design

Strucchange (from R) is to compare the changes in time series regression model. Pelt (from python package ruptures) is based on similar method to "strucchange". Changepoint (from

R) is to compare the changes in mean and variance of time series. Windows (from python package ruptures) is a Window-based change point detection, which is like "changepoint". Therefore, I utilized Pelt and Windows to detect break point for the 2020 data.

2.1.3 Results & Discussion

From the Pelt result (Figure 2.4), there are three breaks detected, which are on '2020-03-17', '2020-05-16', '2020-10-03'. The result is almost in line with the government measures and the same as my guess.

As the Windows need a predefined number of breaks. I tried break 1, 2, 3. The result from the 'Windows' method is slightly different from the pelt method. The first two breaks are the same as Pelt. The last is not, 'pelt' is on 2020-10-03, the 'Windows' is on 2020--6-15.

The 'pelt' method (based on linear regression) fit better to my guess than the 'Windows' method (based on comparing the change of means).

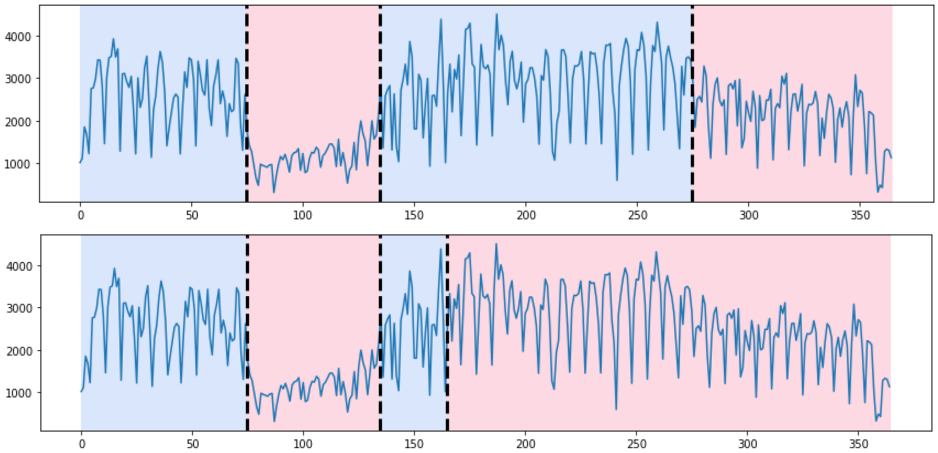


Figure 2.4 Pelt model (top) and Windows model (bottom)

2.2 ACF Features

I selected two segments. Segment 1 is from 01-01-2020 to 16-03-2020. Segment 2 is from 17-03-2020 to 16-05-2020.

2.2.1 Code Design

To understand the ACF features better, I write a function to calculate the features. I validated the result using the feat_acf function from R. To test if the two segments are seasonal or not. I plotted their autocorrelation coefficients (Figure 2.5).

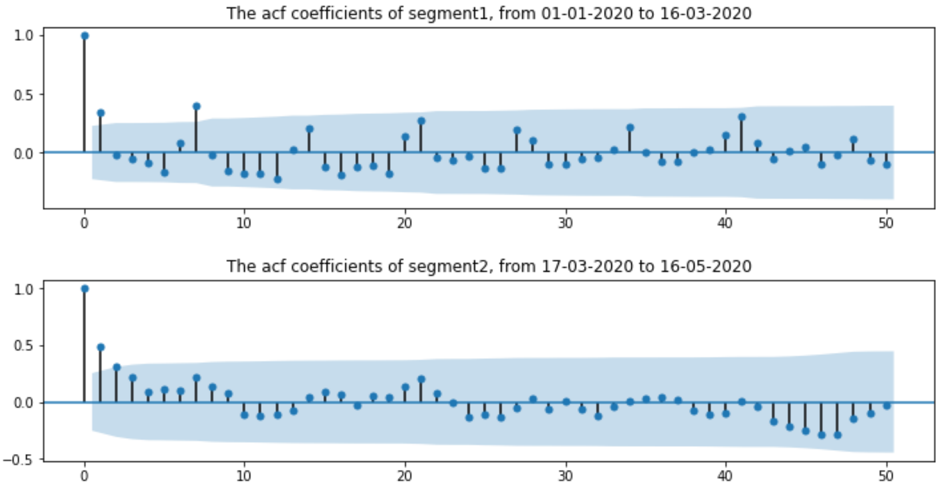


Figure 2.5 ACF for segment 1&2

2.2.2 Results & Discussion

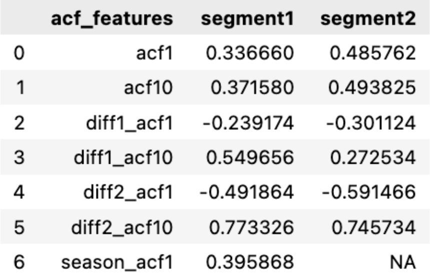
Figure 2.2 shows segment1 has a season of 7 days, while segment2 does not have season. Because ACF coefficients have the max values at lag 1 or 7 days for segment1.

ACF feature:

- The first acf is the correlation between values that are one time apart (1 day for this case).
- The sum of the first ten squared autocorrelation coefficients is a useful summary of how much autocorrelation there is in a series, regardless of lag.
- The diff1_acf computes autocorrelations of the changes in the series between periods. Second differencing helps make stationary if first differencing does not work.

Conclusion:

- For the original data, correlation between values that are one day apart is smaller for segment1. There is less autocorrelation in segment 1 than in segment 2
- For the differenced data (daily changes), there is larger autocorrelation in segment 1 than in segment 2 for the daily changes.
- For the second differenced data, segment 1 is more stationary than segment 2 Segment 1 is seasonal data with a season of 7 days. Segment 2 is not.



	acf_features	segment1	segment2
0	acf1	0.336660	0.485762
1	acf10	0.371580	0.493825
2	diff1_acf1	-0.239174	-0.301124
3	diff1_acf10	0.549656	0.272534
4	diff2_acf1	-0.491864	-0.591466
5	diff2_acf10	0.773326	0.745734
6	season_acf1	0.395868	NA

Table 1 ACF features for segment 1&2