Problem Identification

The onset of Covid-19 pandemic in the US in March of 2020 had crippling effects on the travel industry. These industries have yet to fully recover from mass flight cancellations limited occupancy rules implemented in order to slow the spread of the virus. As such, airports, airlines and all relevant industries including retail and restaurants in airports would benefit from a predictive model that would estimate traffic at airports and on flights could be very useful for said business to be able to make smart business decisions regarding staffing and allocating funds and resources, that would minimize the economic loss that they have been experiencing in addition to working on solutions that will allow these businesses to recover to pre-pandemic numbers.

TSA throughput (number of people going through security) is a good measure of traffic and potentially business for retailers and restaurants in airports as well as airlines. The data is hourly counts of people going through security for each gate for 18 US airport. The data starts on December 30th, 2018 and ends on February 5th, 2022 which is a good amount of data for time series analysis and modeling.

Challenges: unpredictable events: weather, war, pandemics etc

Data Wrangling

Data Collection - The data is stored in https://github.com/mikelor/TsaThroughput repository. The repository includes the raw data collected from the FOIA Electronic Reading Room in addition to individual cleaned and transformed JSON and CSV files for each airport. At the time of initial forking of this repository, I was able to download the data for the following 18 airports.

- ANC - Anchorage
- ATL - Atlanta
- BOI - Boise
- BZN - Bozeman
- DEN - Denver
- DFW - Dallas Fort Worth
- FLL - Fort Lauderdale
- LAS - Las Vegas
- LAX - Los Angeles
- MCO - Orlando
- MIA - Miami
- MSO - Missoula Montana
- PDX - Portland
- PHX - Phoenix
- SEA - Seattle
- SFO - San Francisco
- SJC - San Jose
- TPA - Tampa

Since then, more clean individual CSV files with recent updates have been uploaded. I have not included this data in this initial analysis. However, the hope is to automate getting the data from this stream as they are updated until all US airports and most recent counts are included in the pipeline.

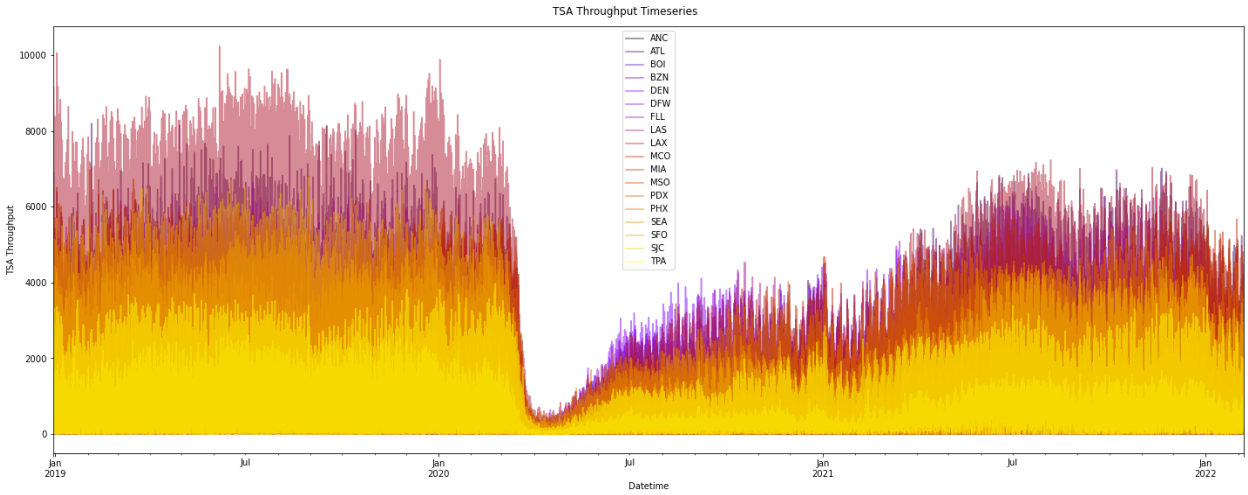
Within each CSV file corresponding to an individual airport, gate-level data is available in separate columns. I wish to do a analysis an airport-wide analysis and will aggregat over all gates to consider the airport as a whole. I joined individual time series of each airport into a single dataframe so that we now have a single data frame with a column for each airport and datetime index in order to perform my time series analysis.

Data Exploration - Our data frame as 18 columns, one for each airport and 27,216 rows. Data type for all columns is float64 and so we don't need to do any type conversion. There are no duplicates in our dataframe.

From the summary statistics, we can identify small vs. large airports. The highest mean and maximum belong to LAX at 3,097 and 10,250. The smallest mean and max values belong to MSO at 69 people and maximum 546. We can divide airports into small and large based on number of people going through on average. Airports with less than 1,000 individuals on average are ANC, BOI, BZN, PDX, SJC and MSO. These we classify as small. The remaining airports have mean throughput over 1,000 individuals with LAX, ATL, and MCO above 2,000.

Exploratory Data Analysis

Time Series Visualization - A visualization of the data over time shows the general trend as well as relative sizes of the airports. There are some very small airports barely visible at the bottom of the graph (orange series along x-axis as well as larger ones with nearly 8,000 throughput on average during the time before the pandemic.

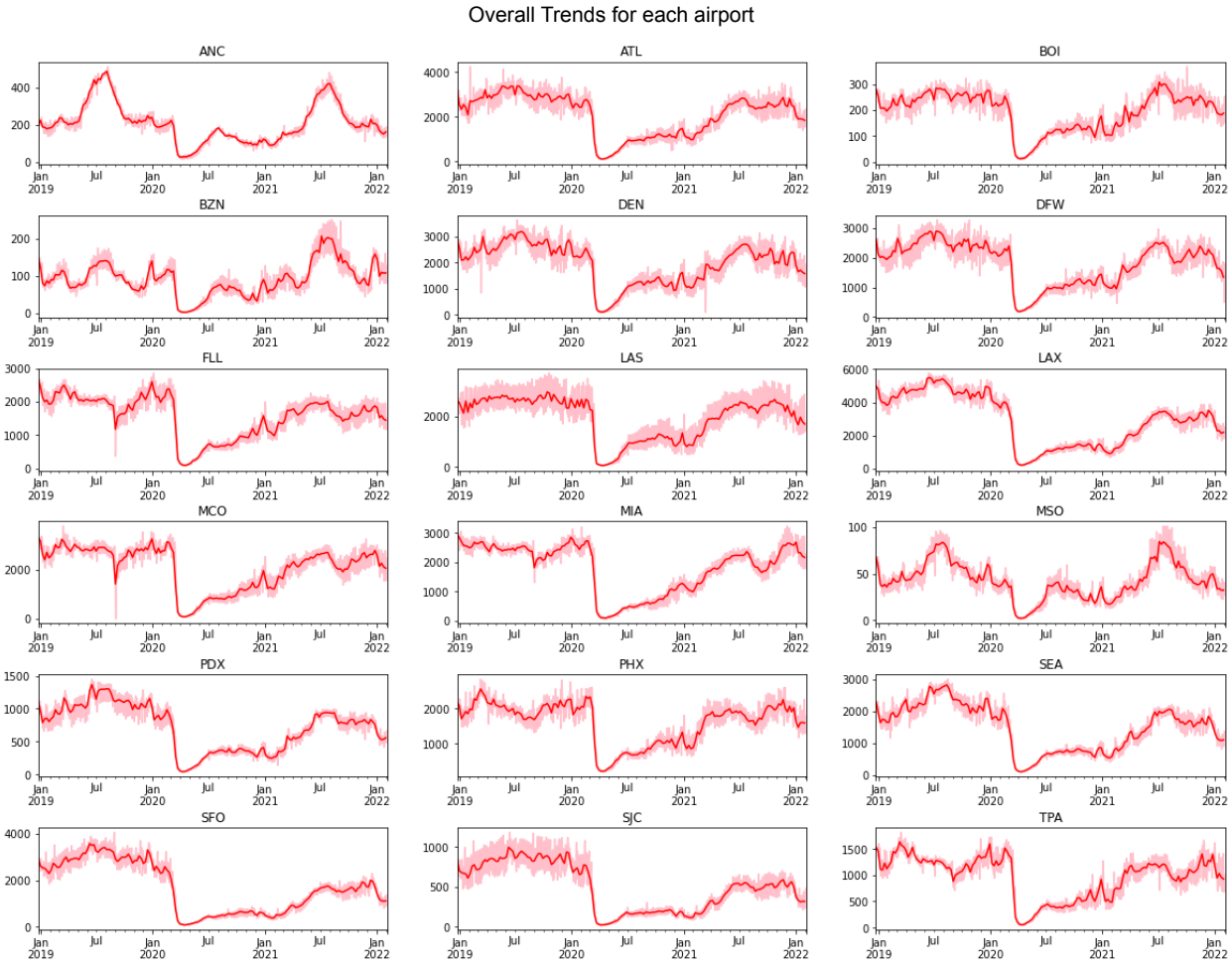


The large drop in numbers in March of 2020 represents beginning of the Covid 19 pandemic. Since then, we see a slow upward trend but 2020 numbers still seemingly low relative to those in 2021. However, on average, total values don't seem to have fully recovered to pre-pandemic numbers.

Individual time series plots for each airport illuminate trends within each airport. The y-axes in the following plots are on different scales such that trends are clear for each airport. For example, Anchorage max throughput seems to be close to 500 people whereas in Atlanta numbers get close to 4,000.

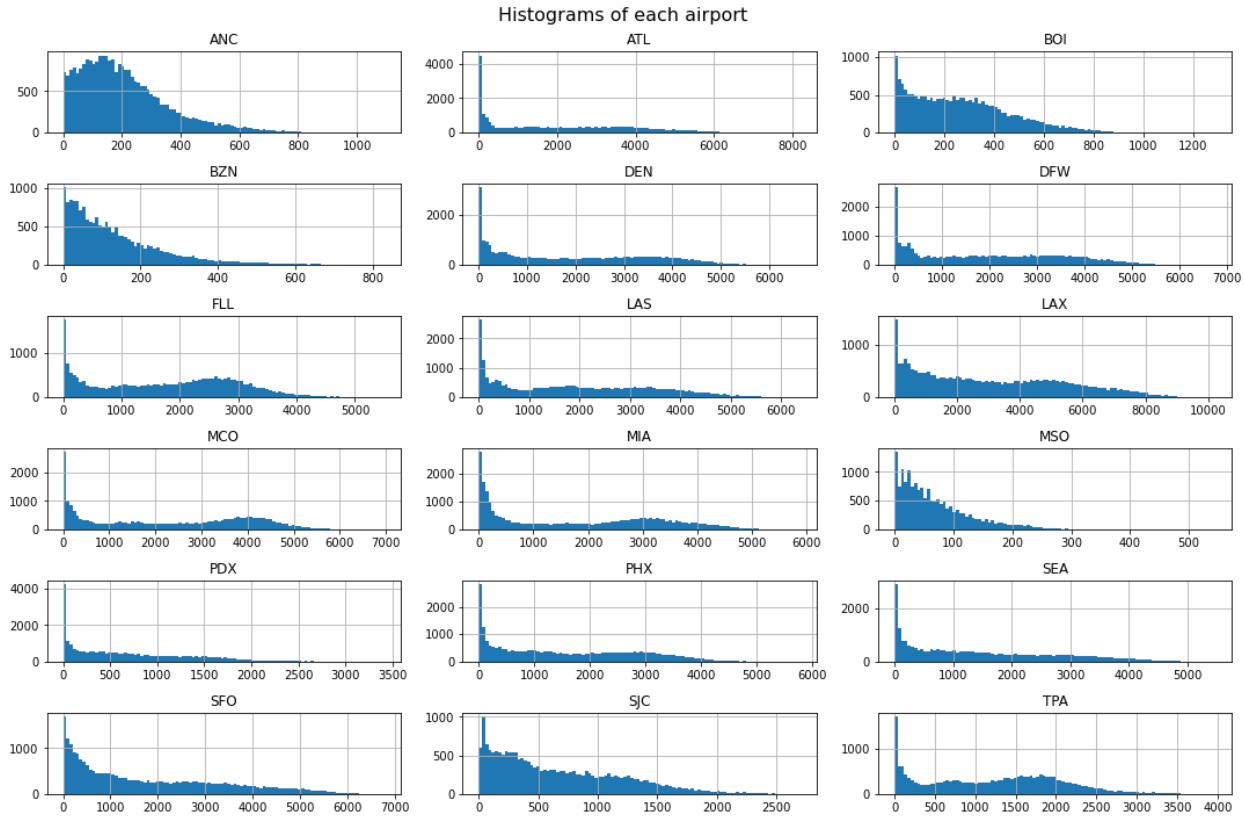
As expected, there is a steep drop in March 2020 for every single airport, however overall trends differ somewhat from airport to airport. For example Anchorage has a trend significantly different than that of Las Vegas or SFO and is more similar to Bozeman or Missoula with two pronounced peaks during summer months. Considering these are located in colder regions (Alaska and

Montana) and since all are small airports, peaking during summer months relative to other, larger airports is reasonable

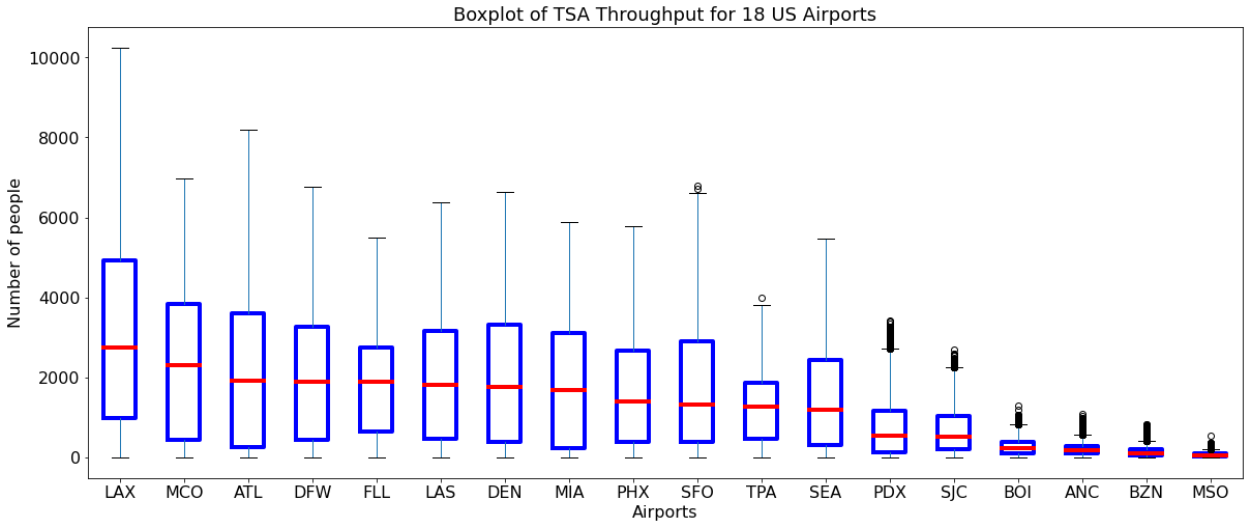


Distribution, Histogram - From the histograms of data for each airport, very low values dominate. As the number of people going through security during a given hour increases, the frequency at which these values occur dramatically decrease. So it is much more common for only a few people at a time (in relative terms, with respect to the size of airport) to go through security as it is for many people going through within the hour. Note once again that the y scale is different for each histogram so we are able to see the histograms more clearly.

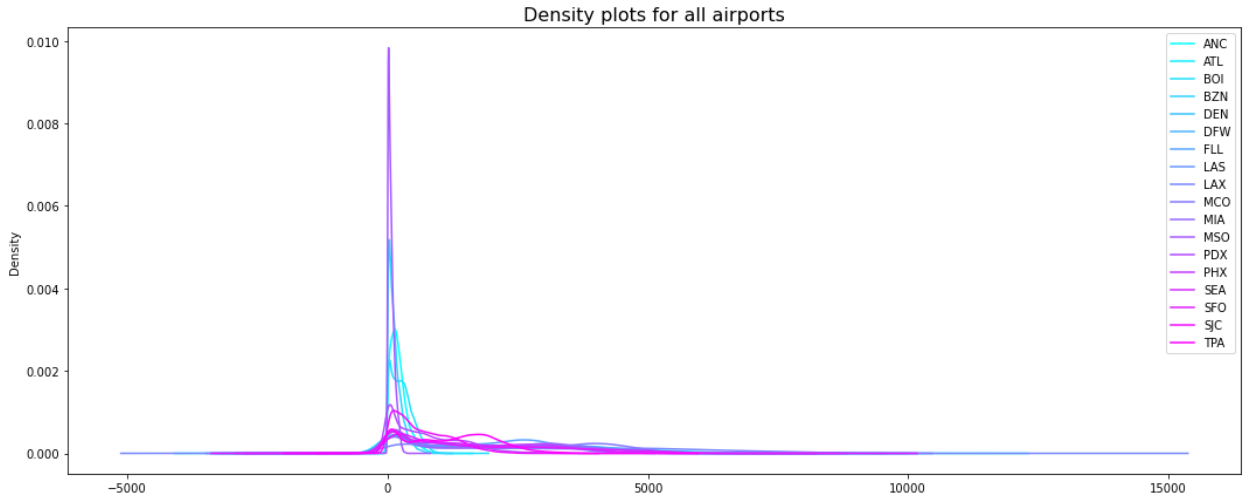
Relatively large airports have a second (but small) peak where there are more people are going through TSA than the next lower bin. One potential reason for this second bump could be due to the changes after Covid 19 pandemic. Perhaps because small airports don't already have many people going through, pandemic restrictions on number of people in airports and on flights (less crowded flights from smaller, lower population regions such as Anchorage even pre-pandemic are more common) didn't affect their throughput as dramatically as it did with large airports.



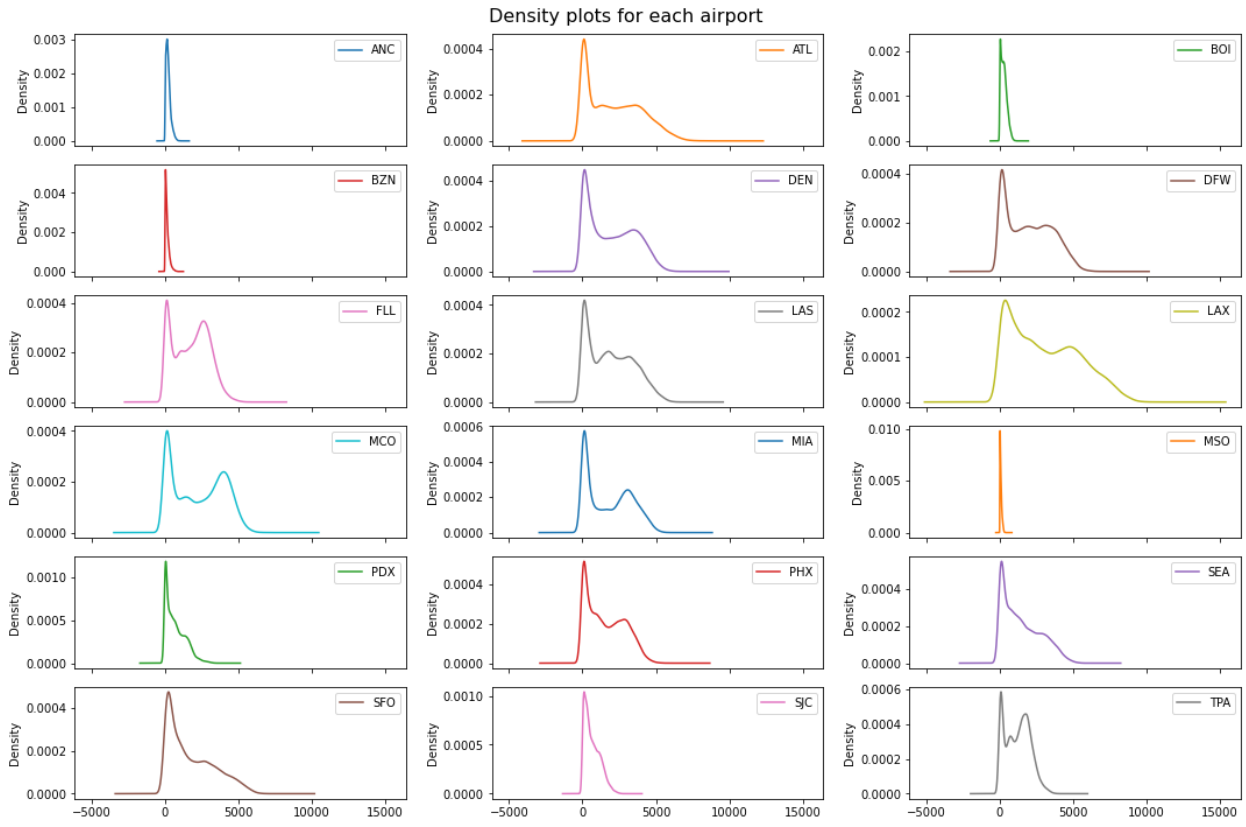
Distribution, Boxplots - Boxplots ordered in decreasing median values show large vs small airports clearly. Here we can see the smallest airports are MSO (Missoula MT), BZN (Bozeman MT), ANC (Anchorage AK), and BOI (Biose ID) and the largest is LAX.



Distribution, Density Plots - Density plots give us another visualization of the data distribution. If we plot them on the same y-scale as below, we have another way of visualizing the relative size of airports.

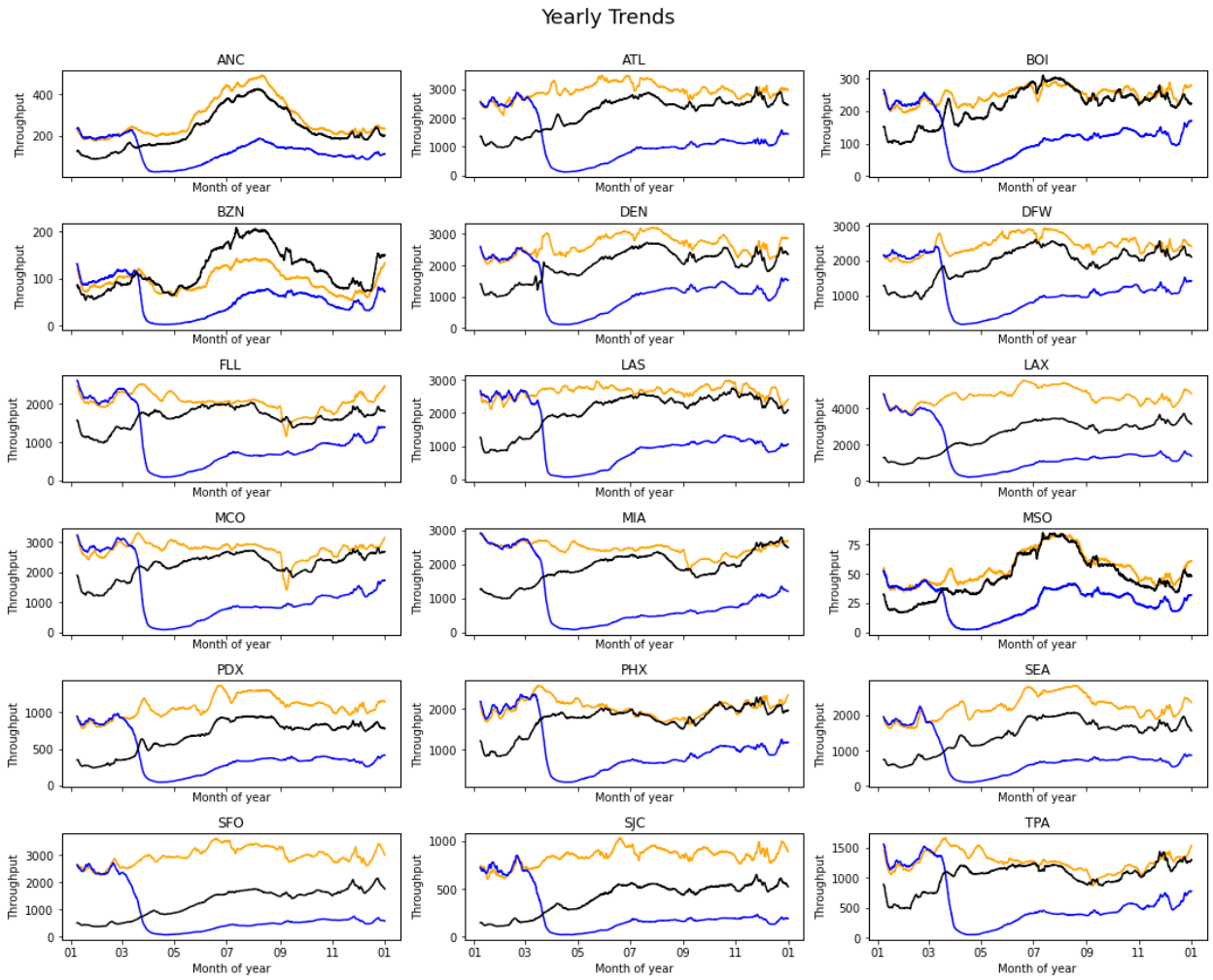


Density plots with different y scales show us more clearly that small airports have a single peak and drop off quickly after. As airports become larger, the drop off flattens out before dropping down to zero. For even larger airports, we see a second smaller peak before it drops down quickly again to zero.



Yearly Trends - In order to visualize major differences in yearly trends we superimpose smoothed out timeseries of each year in a single plot for a given airport. In all the plots below, yellow represents 2019, blue is 2020 and black is 2021. These plots were smoothed over one week in order to get the underlying trends in the data.

2019 2020 2021



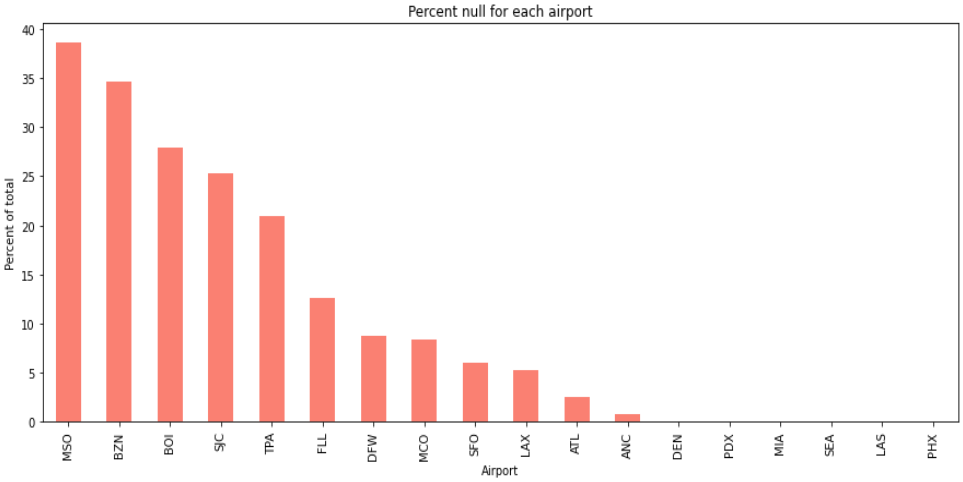
Overall trend for 2019 for nearly all airports is above values for 2020 and 2021 for corresponding months. The only exception is Bozeman airport in which 2019 values are just below 2021. Bozeman was already one of the fastest growing cities in the US before the pandemic. It is possible the pandemic boosted this growth as more people worked remotely and started migrating from large cities to more rural and smaller ones in Montana and other less dense states.

The blue line representing 2020 shows a similar trend for all airports witht he massive drop in March and a very gradual increase afterwards.

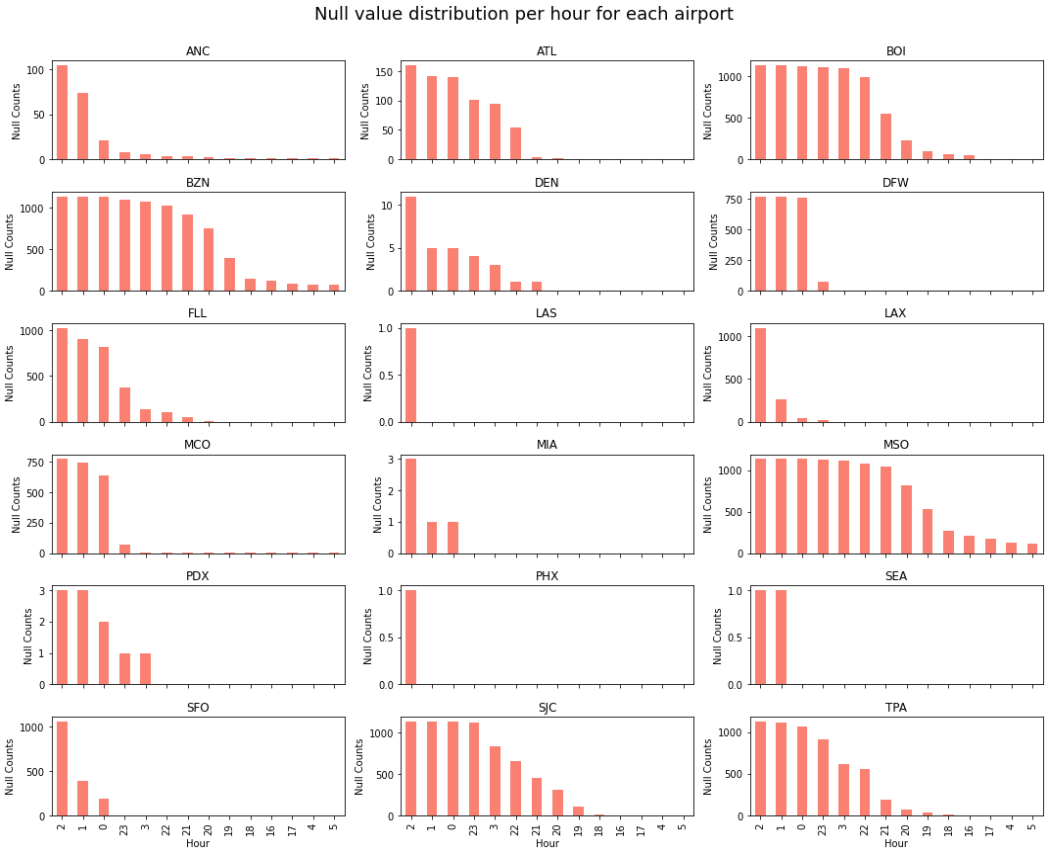
2021 trends show a further recovery. In some cases there seems to be full recovery during the later months of the year. However, some airports, specifically those on the west coast such as SFO, SJC

and LAX are struggling to regain 2019 numbers. Seattle and Portland airports on the west coast also show slow recovery, but not at the extend of the latter three which seems to really be struggling to get back to full recovery.

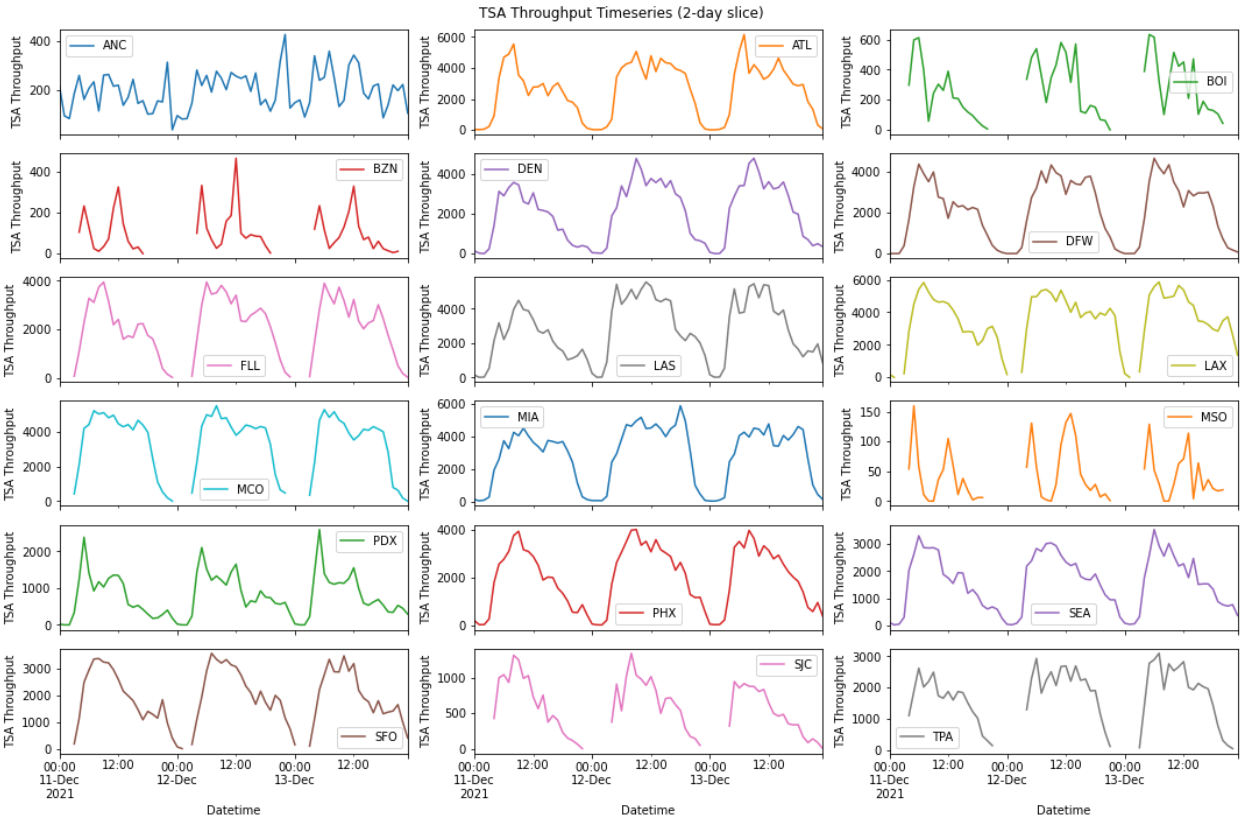
Null values - The small airports such as MSO, BZN, BOI, SJC and TPA have a relatively high number of null values (more than 20 percent).



In order to see where most of the null values are, I have plotted null value counts for those hours.



The small airports (ANC, BOI, BZN, MSO, SJC, and TPA) have a larger time range of null values. Phoenix and Las Vegas only have one missing value. Generally, it appears that all the missing values are in the middle of the night where airports are quiet and there are not many outgoing flights. The range of times for null values is much wider for smaller airports.



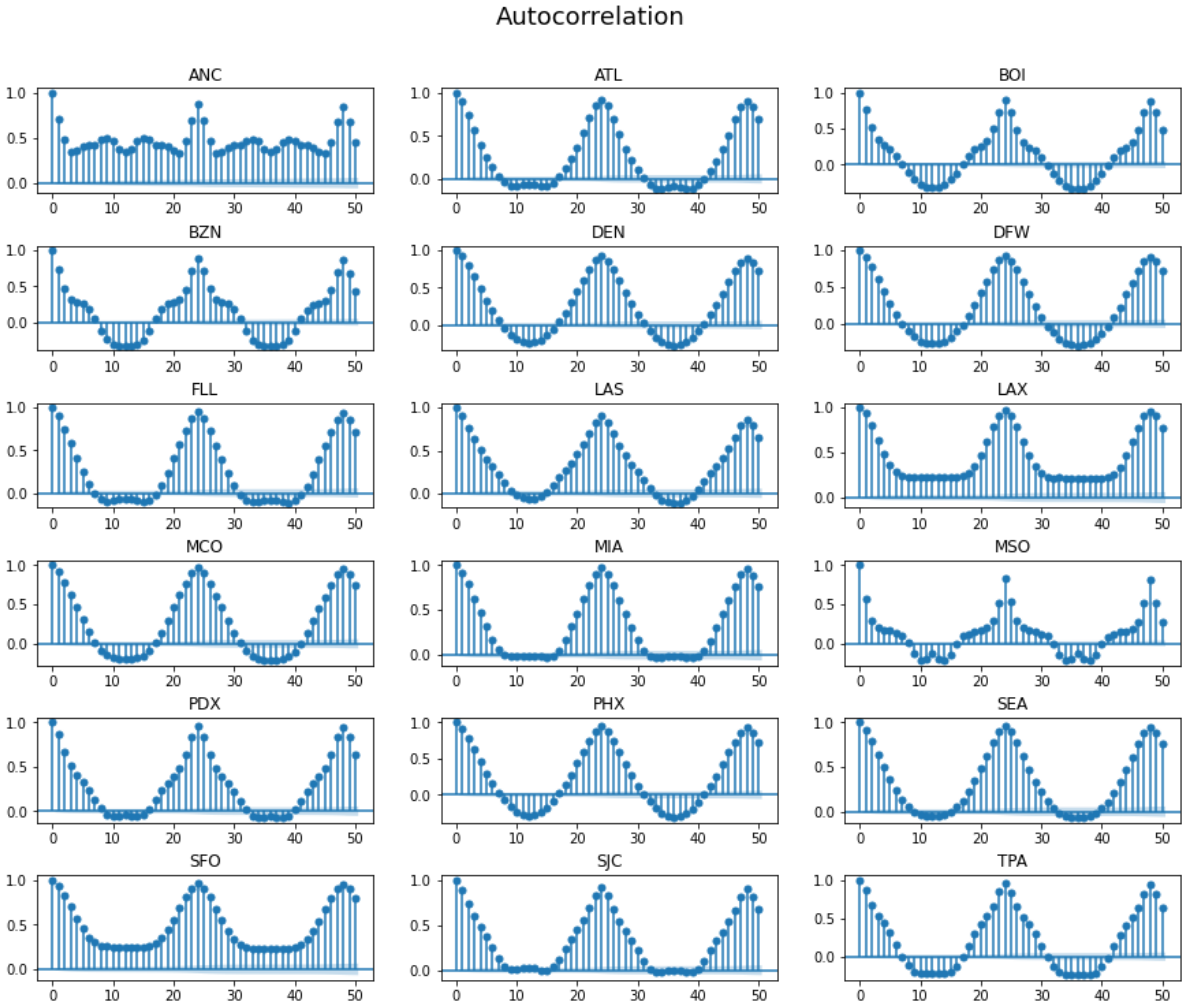
The zoomed-in time series above allow us to visualize the time range of missing values better. This further verifies that missingness occurs in the evening and early morning hours and from experience in small airports such as those in Montana (BZN and MCO) as well as researching flight departures, I can verify that fewer to no outbound flights occur around midnight until 4 am and thus it is most likely that no one passed through TSA checkpoints during those times where values are missing. Based on this analysis, null values later during feature engineering will be replaced with zero.

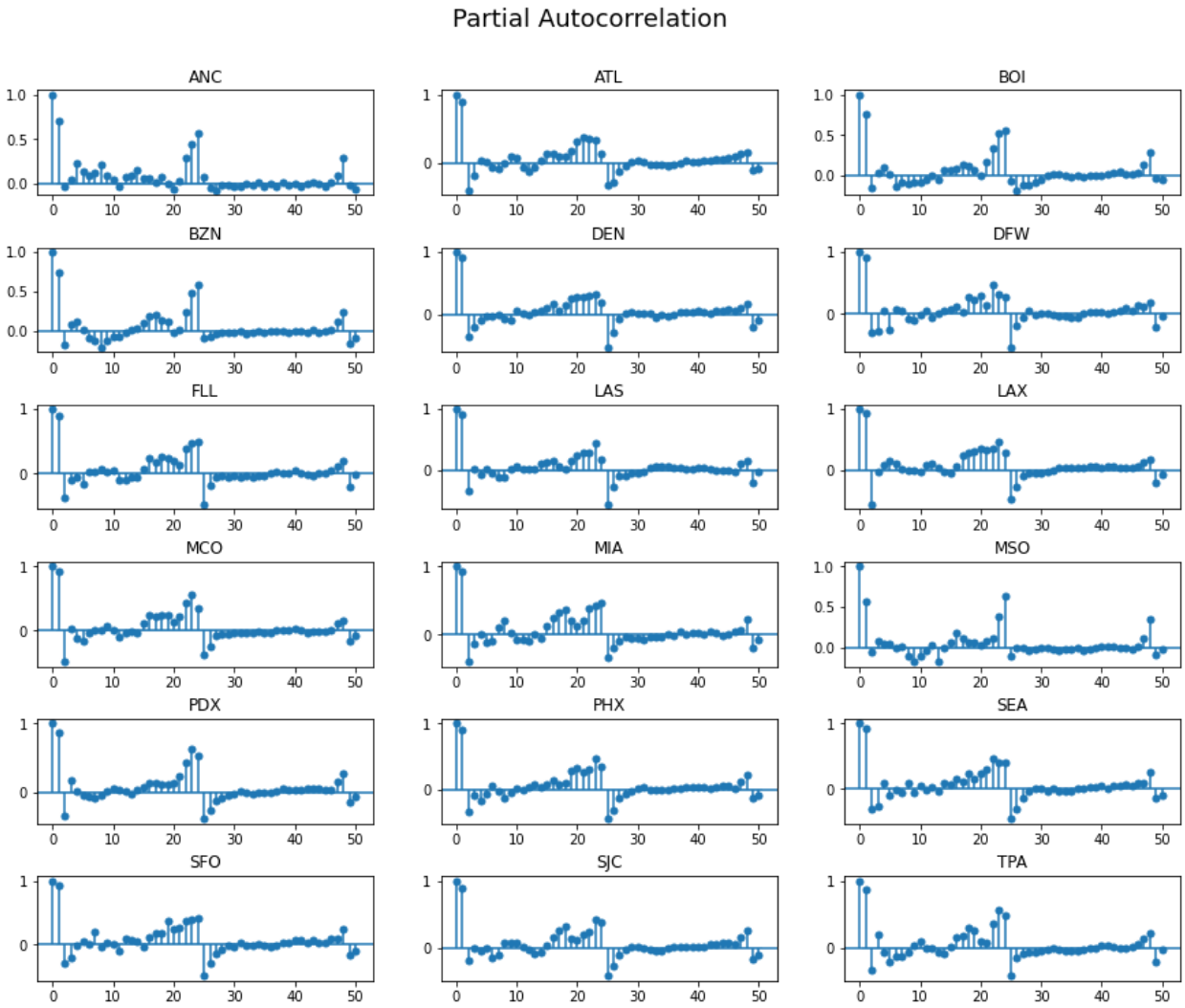
Preprocessing and training

Train/test split - Used TimeSeriesSplit from scikit-learn and set n=5 to define train and test sets with max train size of 10,000 and max test size of 1,000.

Seasonality - The data has very strong seasonality with a period of 24 hours. This period is clear from both autocorrelation and partial autocorrrelation plots.

There are definitely enough seasons in our data for an ARIMA model.





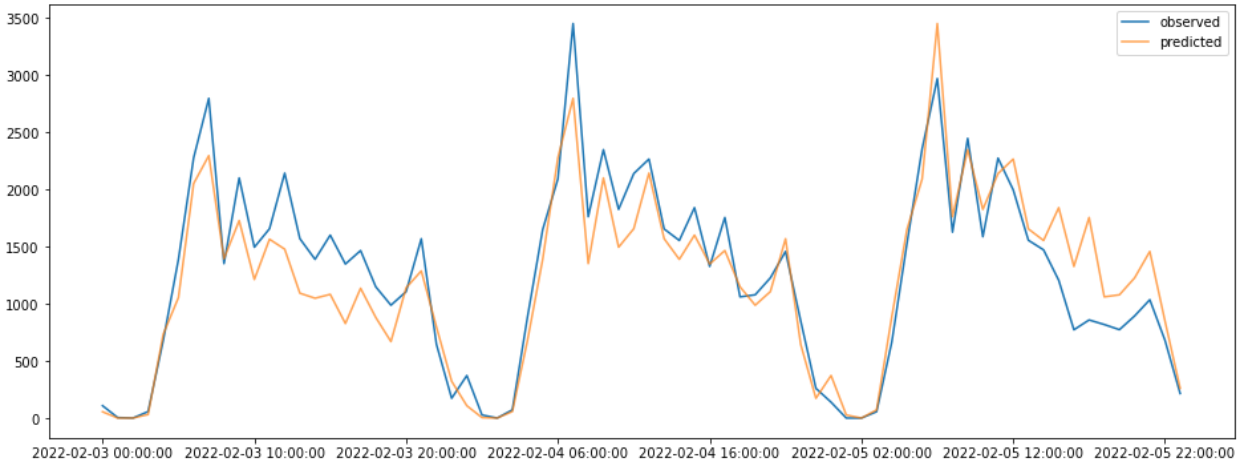
Stationarity - Using only SFO data for processing and modeling.

- Trend stationarity - Used Dickey Fuller test to determine if there any trend in the SFO series. The result of the test was inconclusive. The p-value reported from the test is 0.02 which is significant if we use 5% significance limit.
- There is not much amplitude variation (or clear trend) until march of 2020 when there was a large decrease in values in a very short period of time. There is however a second layer of seasonality apparent. After March there is a general trend upward back to or closer to original values.
- Autocorrelation plot - The autocorrelation plot shows high correlation with previous lags so the series is definitely not stationary.

Modeling

Baseline Model - Use yesterday's values

Mean absolute error: 211 people



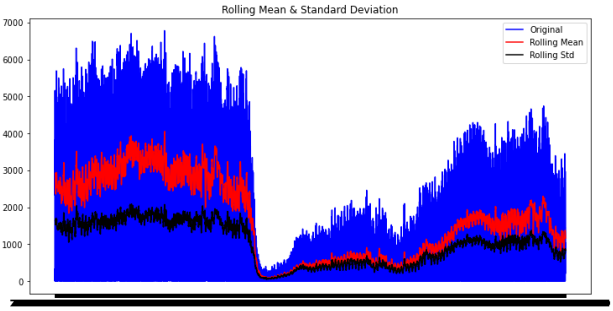
Seasonal ARIMA Model - We determined that there is strong seasonality in our data (with a period of 24 hours) so we use a seasonal ARIMA Model (p, d, q) X (P, D, Q, S=24).

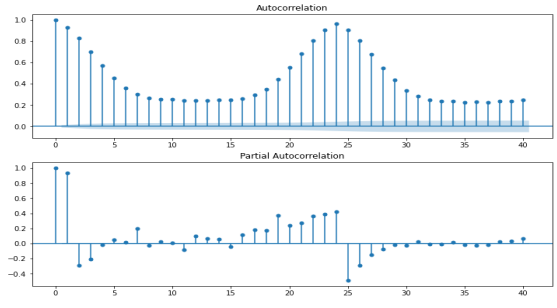
- p - Auto Regressive parameter
- q - Moving Average Parameter
- d - Order of differencing
- P - Seasonal AR parameter
- Q - Seasonal MA parameter
- D - Seasonal difference order
- S - Period

In order to build an initial model, I looked at the autocorrelation and partial autocorrelation plots.

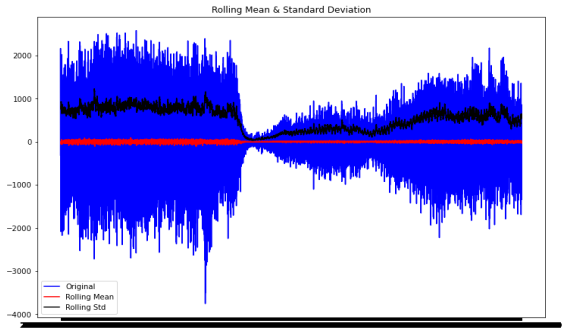
SFO Original time series- Based on the autocorrelation plot, there is significant correlation between lags so our data is non stationary and seasonal with a period of 24 hours.

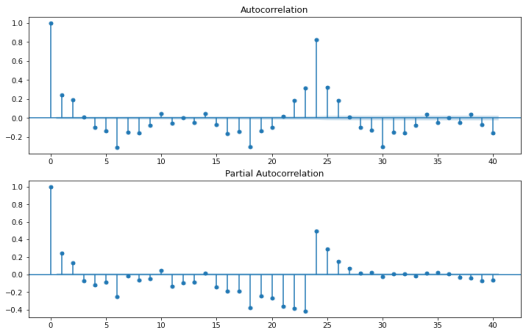
In the plots below, blue represents all the values. The red is the rolling mean of the data and black is the rolling standard deviation.



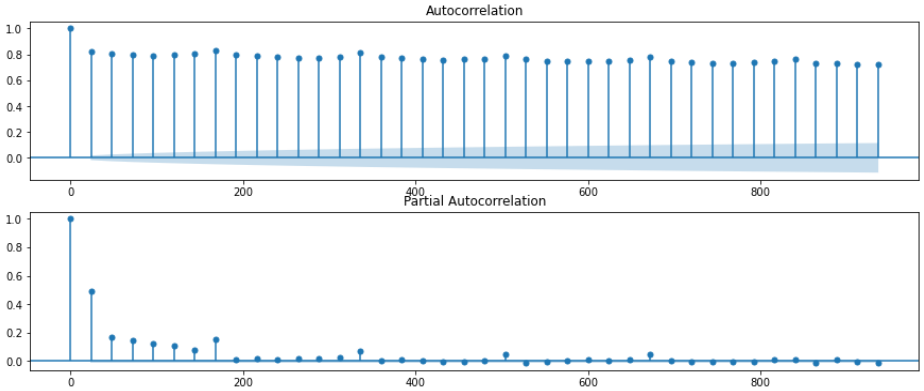


If we take the first difference, that is d=1 the results look much better, however there is strong seasonality.

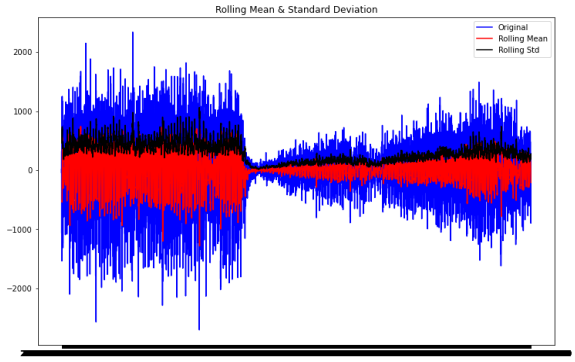


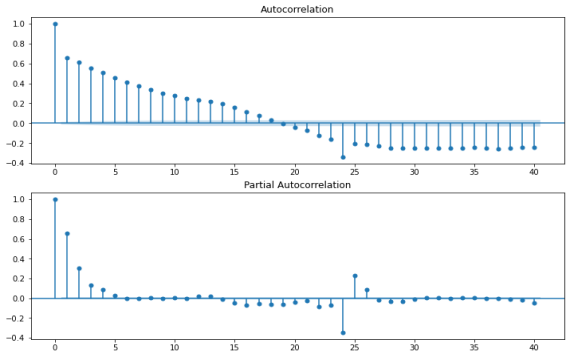


If we look at the seasonal autocorrelation plot, we see there is considerable seasonality.

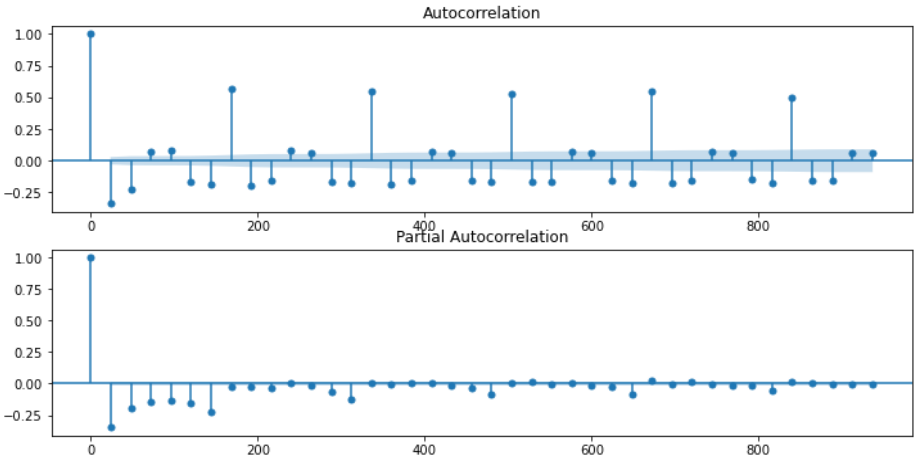


Next we take the first seasonal difference, D=1 without taking the first difference.



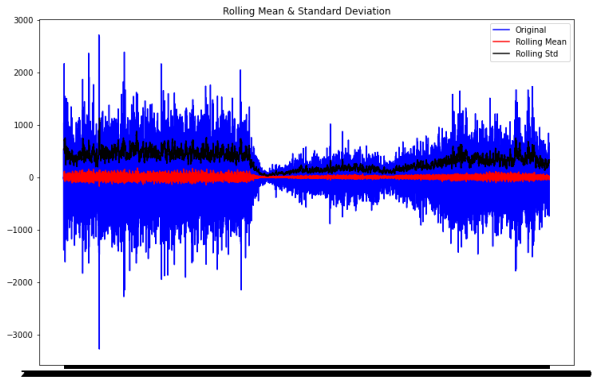


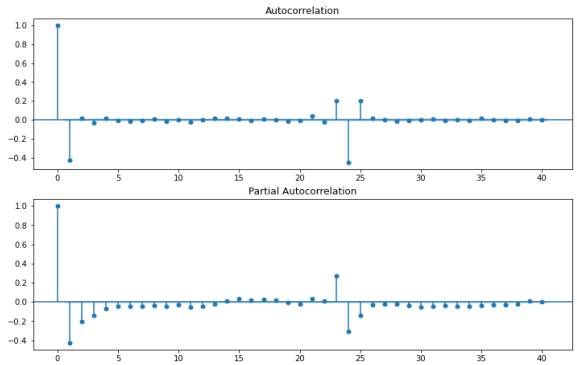
The seasonal autocorrelation shows some seasonality still remains after the first seasonal difference has been taken.



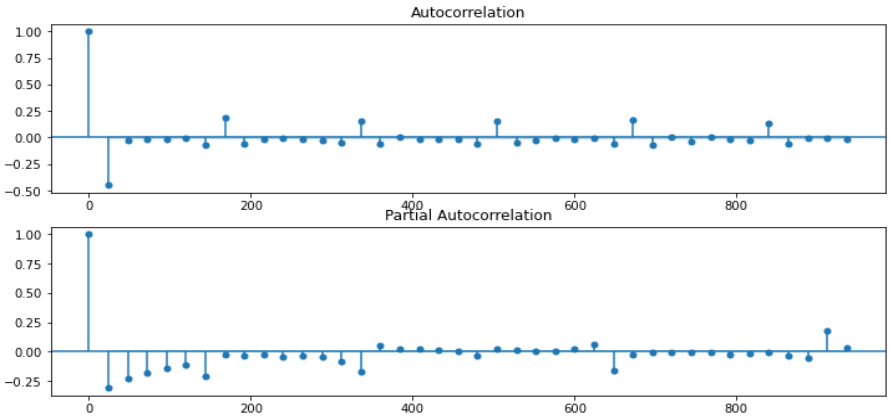
Since there is such strong seasonality, we need to have D = 1. Trying d=1, D=1, d+D=2, that is seasonal and first difference we can finally see a much more stationary timeseries.

Based on the autocorrelation plot below, lag 1 is significantly negative so we choose p=0 and q=1.



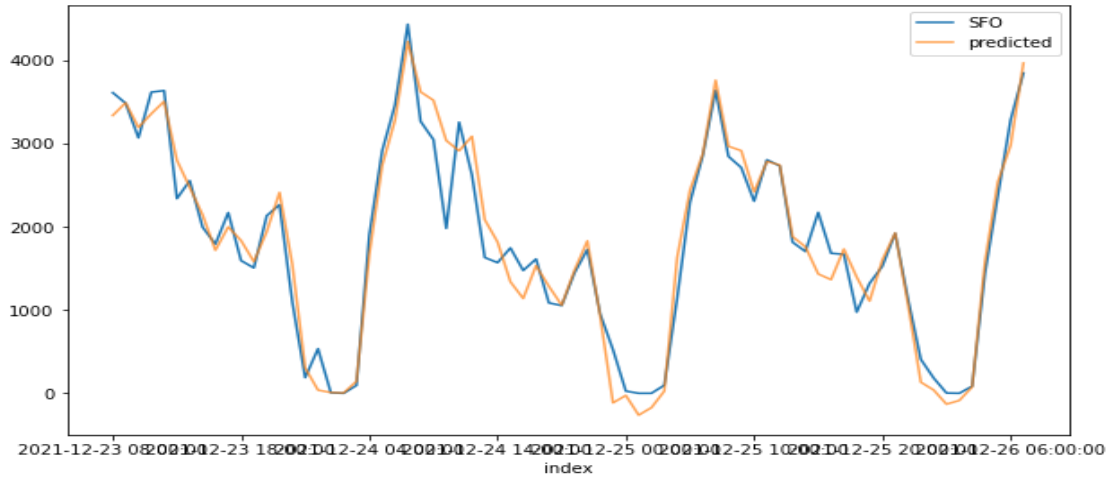


The seasonal autocorrelation plot of the seasonal and first differenced timeseires looks like it has much less seasonality remaining than before. There is significant negative lag at lag 1 for seasonal aurtocorrelation plot, so just as before P=0 and Q=1.



Initial models - ARIMA(0, 1, 1)X(0, 1, 1, 24)

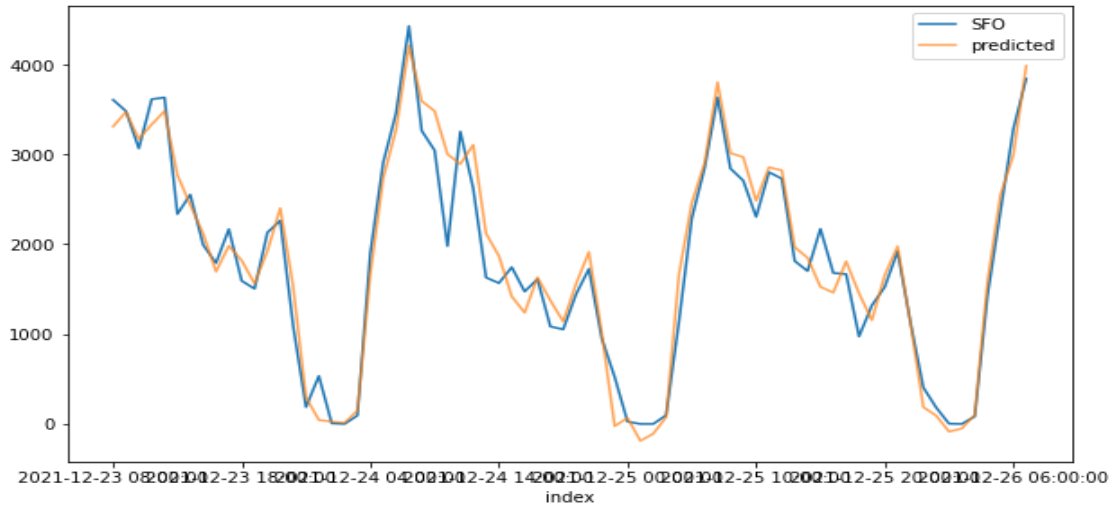
- Train - MSE = 205 people
- Train - AIC = 133,309



Also try d = 1 and will change p =1.

ARIMA(1, 0, 1)X(0, 1, 1, 24)

- Train - MSE = 210 people
- Train - AIC = 132,987



The two metrics for measuring the performance here are mean absolute error or MSE (in unit of number of people) and Akaike information criterion (AIC). MSE is easily interpretable and AIC is ideal for predictive models while penalizeing for more complicated models in order to avoid overfitting.

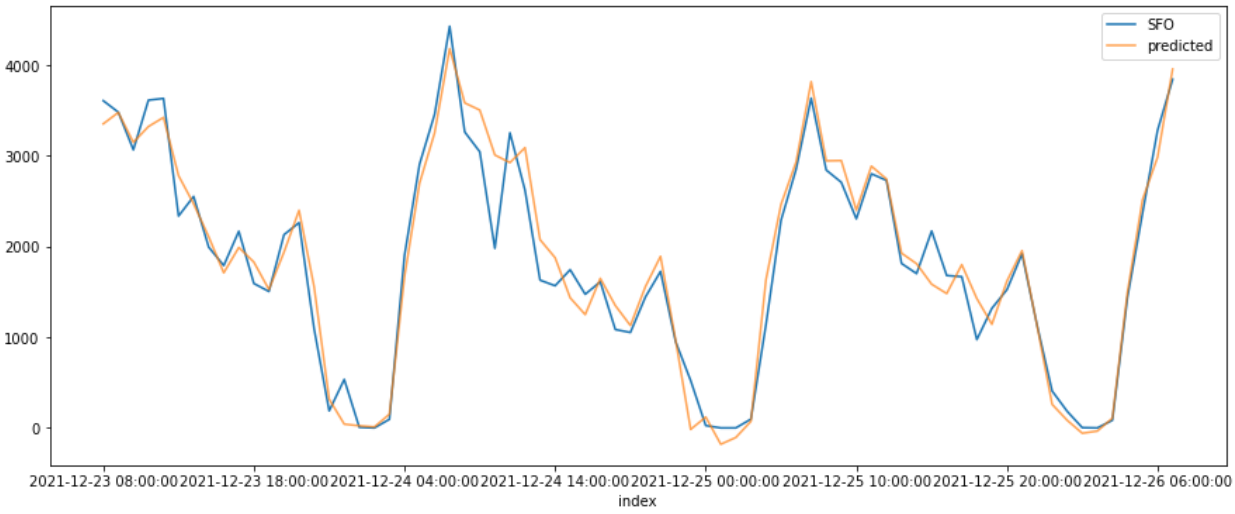
We get very similar results for both models.

Grid Search for optimal parameters - There are 7 parameters: 4 seasonal including period=24 hours, and 3 ARIMA parameters. For the Grid search in order to find optimal combination of parameters. D and d can take on values 0 to 2 (perhaps it is wise to chose D=1 and vary d between 0 and 1 only in interest of time; with the current grid search the program on a laptop takes nearly 3-4 weeks!). The remaining parameters, P, p, Q, q can range between 0 and 3 in this grid search (though in interested of time, it is best to use 2 as the maximum value for any of these parameters).

Optimal Model(s) - After running the grid search and evaluating models using MSE and AIC the best models are:

ARIMA(1, 0, 1)X(1, 1, 1, 24)

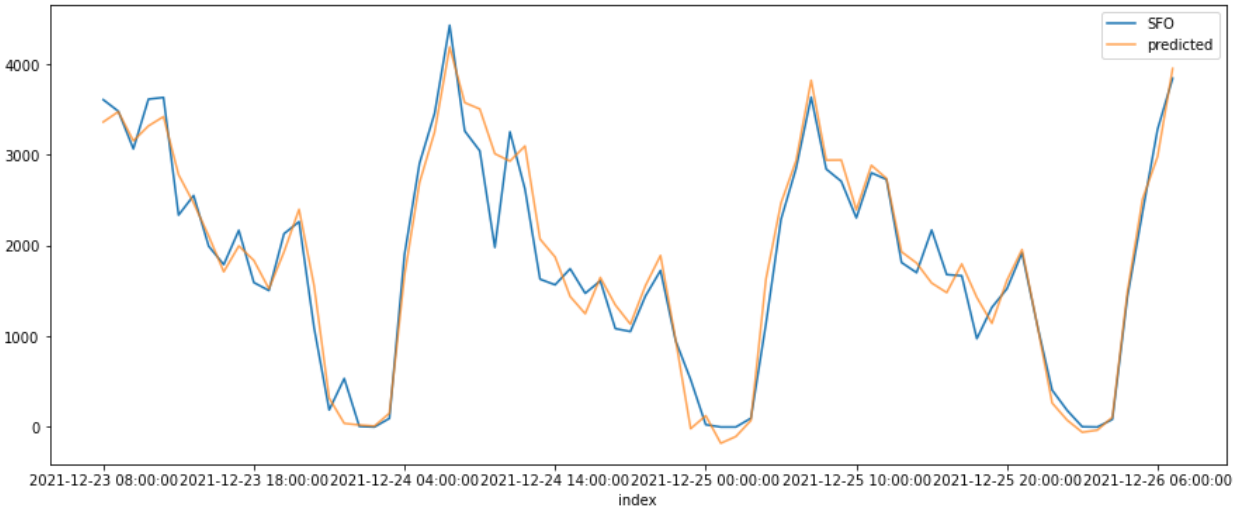
- Train - MSE = 198
- Train - AIC = 132,865



ARIMA(1, 0, 1)X(0, 1, 2, 24)

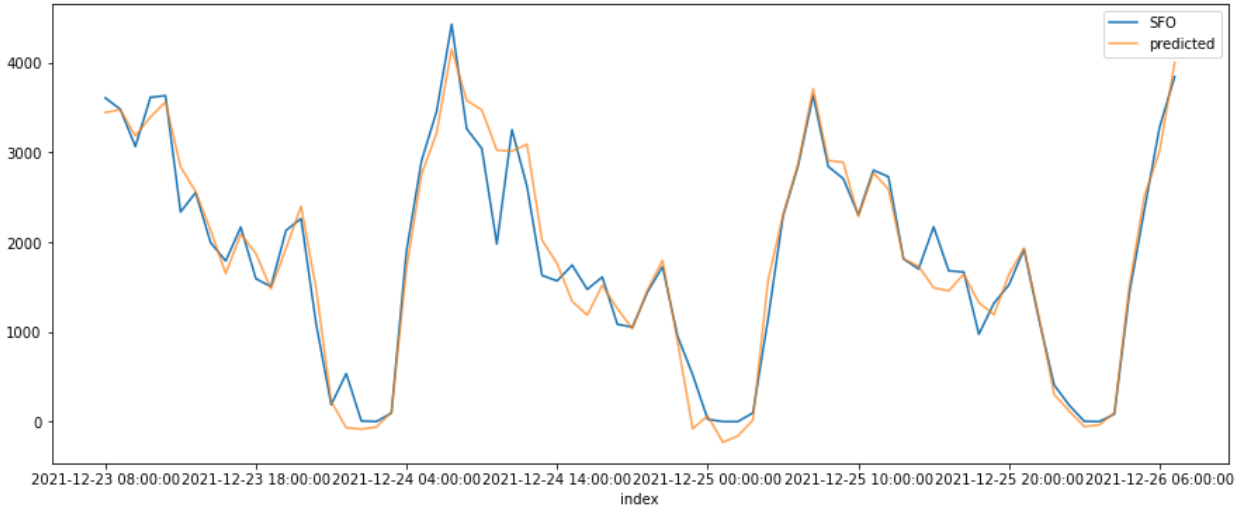
- Train - MSE = 198

- Train - AIC = 132,854



ARIMA(1, 0, 1)X(3, 1, 3, 24)

- Train - MSE = 181
- Train - AIC = 133,531



The last model has a much lower MSE but higher AIC and there is good chance of overfitting.

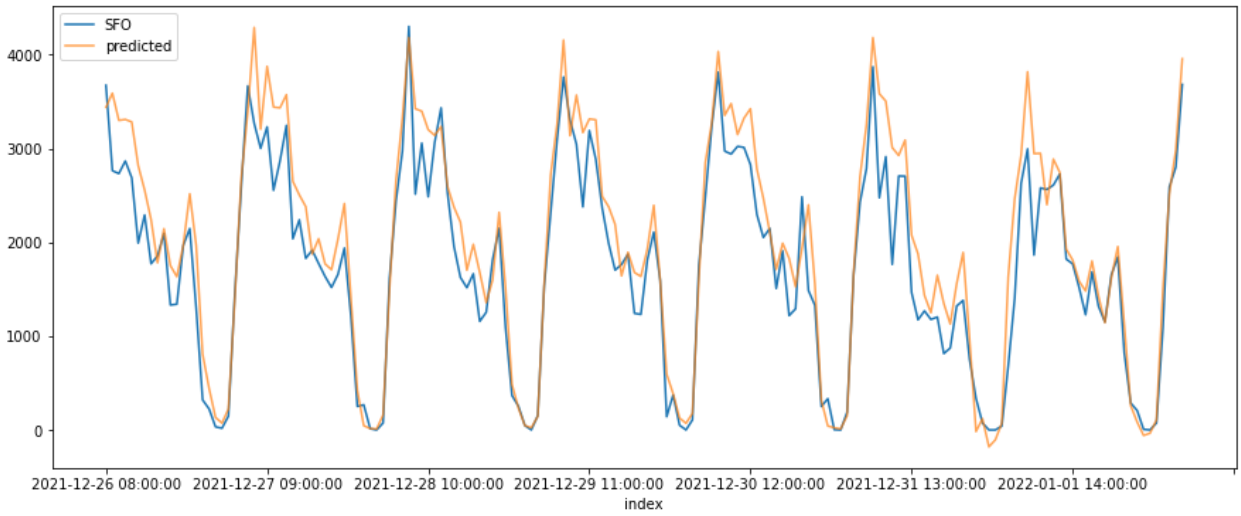
I'll test all these models against our test set and determine the best model.

Results and conclusions - Using static prediction, results with test data is as follows for each model:

Test MSE for (1, 0, 1)X(1, 1, 1, 24):

Test MSE for (1, 0, 1)X(0, 1, 2, 24):

Test MSE for (0, 1, 1)X(3, 1, 3, 24):



Future work:

Clustering