---------------------------------- **DS challenge** ---------------------------------------------

Chicago Ridership

This DS challenge is designed to evaluate your skills and intuition in a real world data problem. Please using the following dataset from 2002 to 2016
Dataset: https://data.cityofchicago.org/Transportation/CTA-Ridership-L-Station-Entries-Daily-Totals/5neh-572f

CTA - Ridership - 'L' Station Entries - Daily Totals | City of Chicago | Data Portal CTA - Ridership - 'L' Station Entries - Daily Totals - This list shows daily totals of ridership, by station entry, for each 'L' station dating back to 2001. Dataset shows entries at all turnstiles, combined, for each station. Daytypes are as follows: W=Weekday, A=Saturday, U=Sunday/Holiday. See attached readme file for information on how these numbers are calculated.

Load data and perform exploratory data analysis.

What are the characteristics of the data?
What are your findings?
What did you learn from the data?
The transportation department wants to improve service in the next few years. Can you build a model for them to forecast daily rides? (Please use 2017 data as testing set for evaluation)
Which aspects should you consider for this model?

Please explain how you built the model and justify the choices you made.
How would you evaluate this model to ensure it would be robust for production usage?

Please submit the result in the form of runnable notebooks or scripts. A link to GitHub or other code repository would be great.
Please let us know if we need to do anything special to run your notebook (install packages, get extra data etc.)

This is a n-dimensional time-series problem. Essentially, we have to forecast ridership for every station. I have used Facebook's Prophet for the Time Series Modelling. In general, in these cases time series forecasting helps in capacity planning and improving efficiency.

**What are the characteristics of the data?**

```
#########  Data Types   ##########
station_id          int64
stationname        object
date               object
daytype            object
rides              int64
```

Number of Data-Points in Testing Data (2017)       = 52,713
Number of Data-Points in Training Data (2001-2016) = 831,358

```
######### Day Type Values ########
['A' 'W' 'U']

########  Total number of NaN  ######
station_id      0
stationname     0
date            0
daytype         0
rides           0
Month           0
Day             0
Year            0
Date_Format     0
Day_of_Week     0
dtype: int64
######### Number of Unique Station Id #######
146
########  Number of Unique Station Name  ######
147
```

**What are your findings?**

## Holiday List - Deciphered from the Data
## New Year's Day
## Memorial Day
## Independence Day
## Labor Day
## Thanksgiving Day
## Christmas Day

I have added this curated list of holidays to the model.

The missing values in the rides column are interpreted as 0.

####### Top 20 Stations with Maximum Missing Values    #######

| | Station | Num_Zeroes |
|---|---|---|
| 62 | Madison/Wabash | 1042 |
| 96 | Skokie | 746 |
| 3 | Washington/State | 730 |
| 9 | Randolph/Wabash | 529 |
| 64 | Wellington | 484 |
| 26 | Damen-Brown | 388 |
| 128 | Paulina | 368 |
| 53 | Irving Park-Brown | 368 |
| 12 | Southport | 361 |
| 114 | Addison-Brown | 361 |
| 2 | Montrose-Brown | 356 |
| 31 | Damen-Cermak | 308 |
| 136 | Kostner | 304 |
| 1 | Central Park | 282 |
| 49 | California-Cermak | 280 |
| 23 | Pulaski-Cermak | 279 |
| 56 | Kedzie-Cermak | 277 |
| 85 | Diversey | 277 |
| 95 | Western-Cermak | 268 |
| 127 | 18th | 250 |

Although I have no information about the geographical information (latitude ,longitude) I have observed that there is at-least 2 cohorts in the data. Stations with majority weekday ridership (Possibly Office/Business locations) and majority weekend ridership (Possibly Tourist locations). I have taken two such stations (reflective of the two cohorts) and done my analysis.

**Number of zeroes in the rides column in a sample(2) of the stations**

|  | 2001 | 2002 | 2003 | 2004 | 2005 | 2006 | 2007 | 2008 | 2009 |
|---|---|---|---|---|---|---|---|---|---|
| Library | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Central Park | 71 | 78 | 69 | 64 | 0 | 0 | 0 | 0 | 0 |

|  | 2010 | 2011 | 2012 | 2013 | 2014 | 2015 | 2016 |
|---|---|---|---|---|---|---|---|
| Library | 0 | 0 | 19 | 2 | 0 | 0 | 0 |
| Central Park | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

---

**Total number of rides across years in the rides column in a sample(2) of the stations**

|  | 2001 | 2002 | 2003 | 2004 | 2005 | 2006 |
|---|---|---|---|---|---|---|
| Library | 683280 | 711961 | 705501 | 642950 | 691128 | 853604 |
| Central Park | 116859 | 83915 | 44105 | 100812 | 172977 | 228622 |

|  | 2007 | 2008 | 2009 | 2010 | 2011 | 2012 |
|---|---|---|---|---|---|---|
| Library | 1106293 | 1344912 | 1251260 | 1256159 | 1309878 | 1338638 |
| Central Park | 271699 | 307676 | 323544 | 336337 | 355568 | 397764 |

|  | 2013 | 2014 | 2015 | 2016 |
|---|---|---|---|---|
| Library | 1259201 | 1240103 | 1219248 | 1198160 |
| Central Park | 408274 | 417750 | 425670 | 384087 |

**Heat-Map of the total number of Rides across stations (sample of 20) across years**
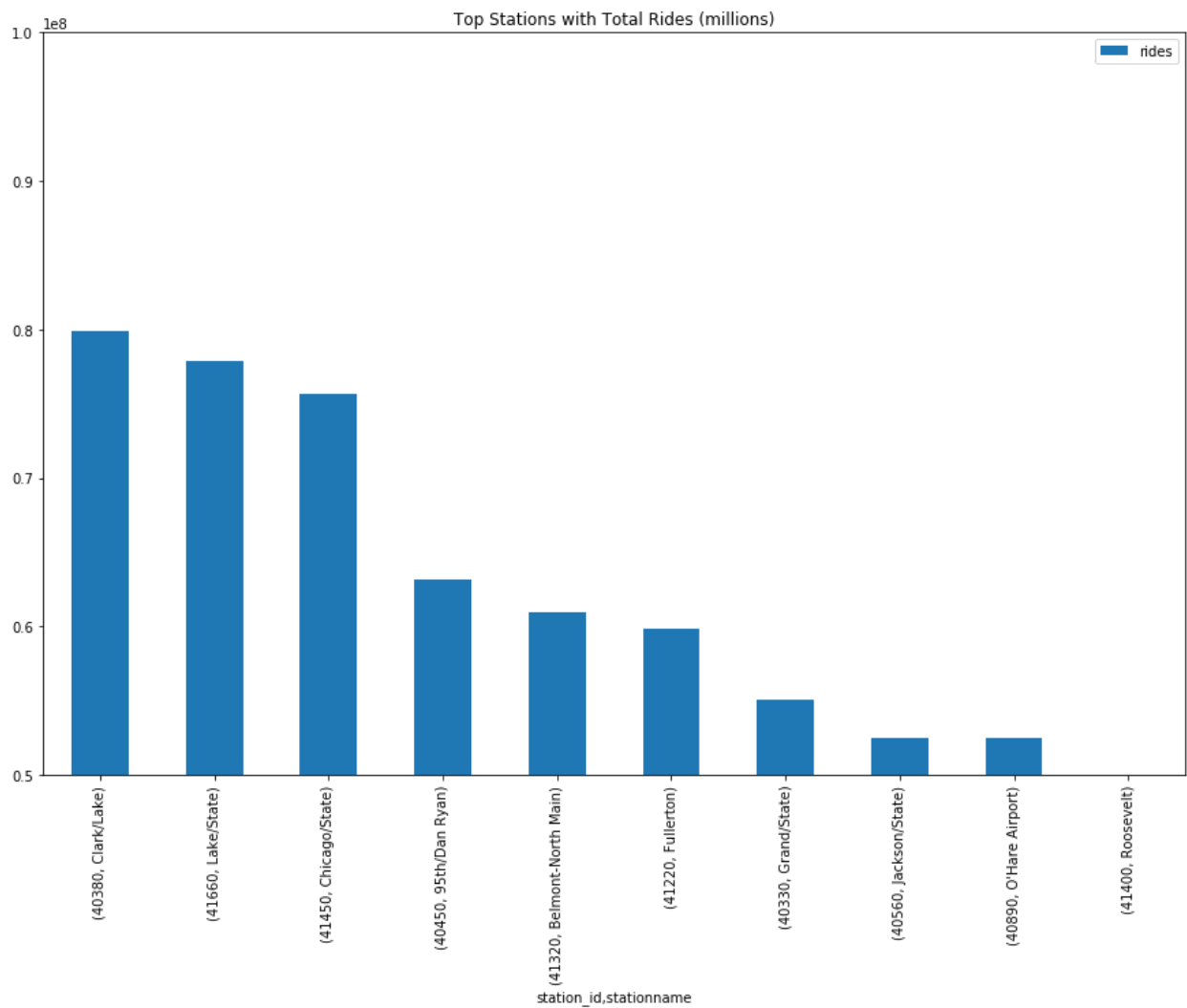
*The busiest stations can be identified according to the intensity in the heatmap.*
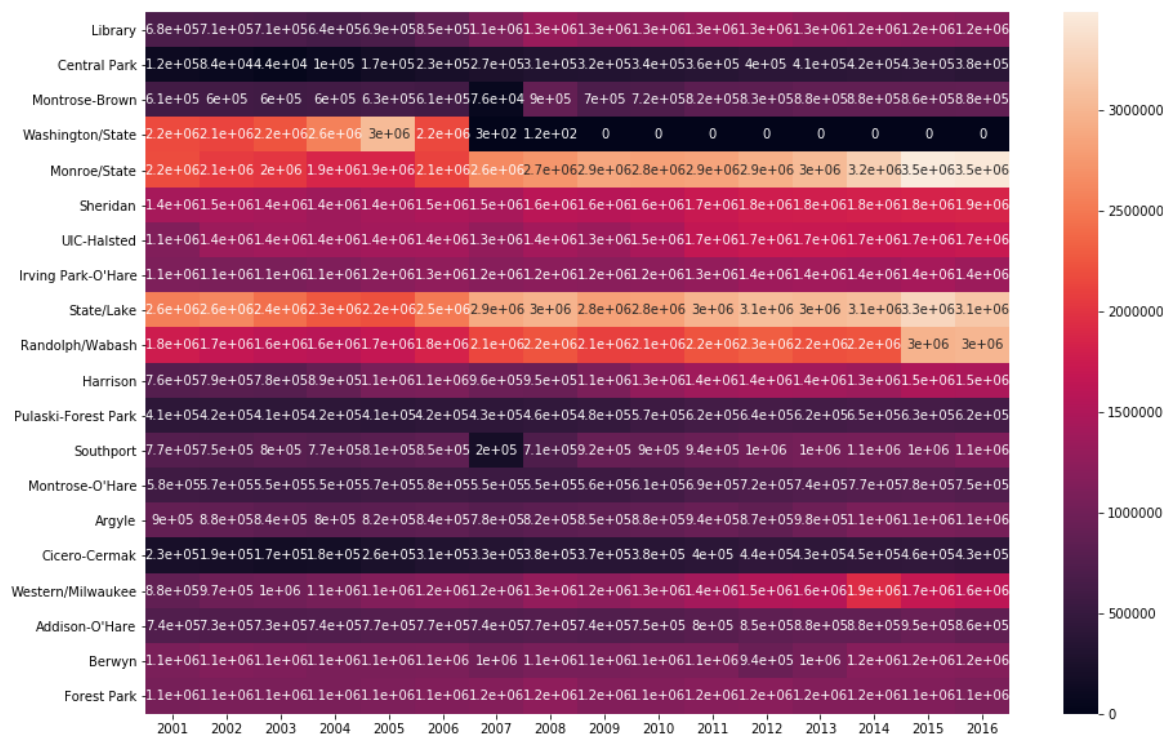


*There is a progressive increase in the number of rides across the years.*
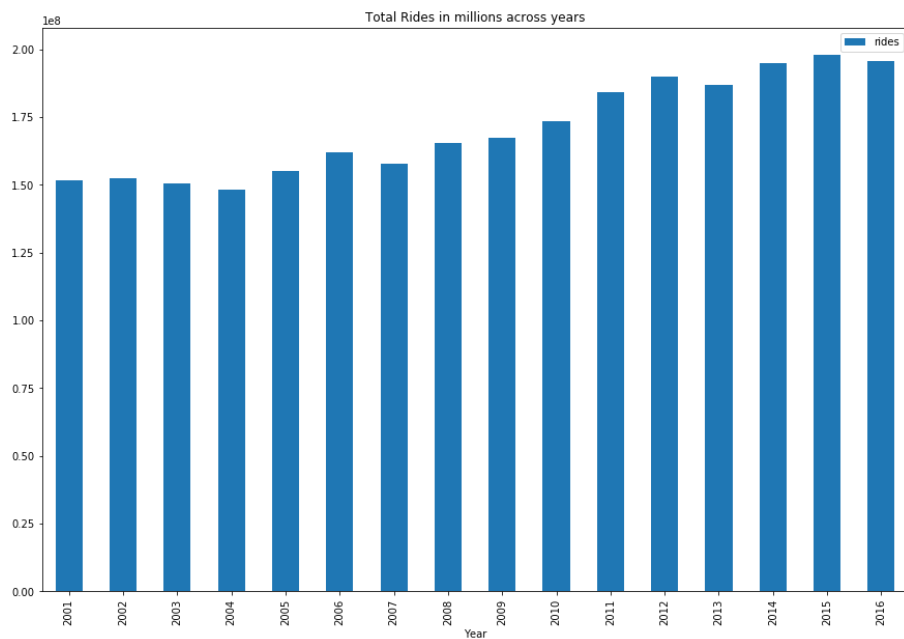
## Identification of the busiest stations



Top Stations with Total Rides (millions)

**Sample of Stations for further deep dive analysis**

*Station-Id : 40850*
*Station-Name : Library*

Autocorrelation Plot suggests that recent years (around 6 years) historical data should be sufficient (strong correlation) for the time series modelling. Prophet is intelligent enough to decipher this on its own and incorporate this in the modelling process.
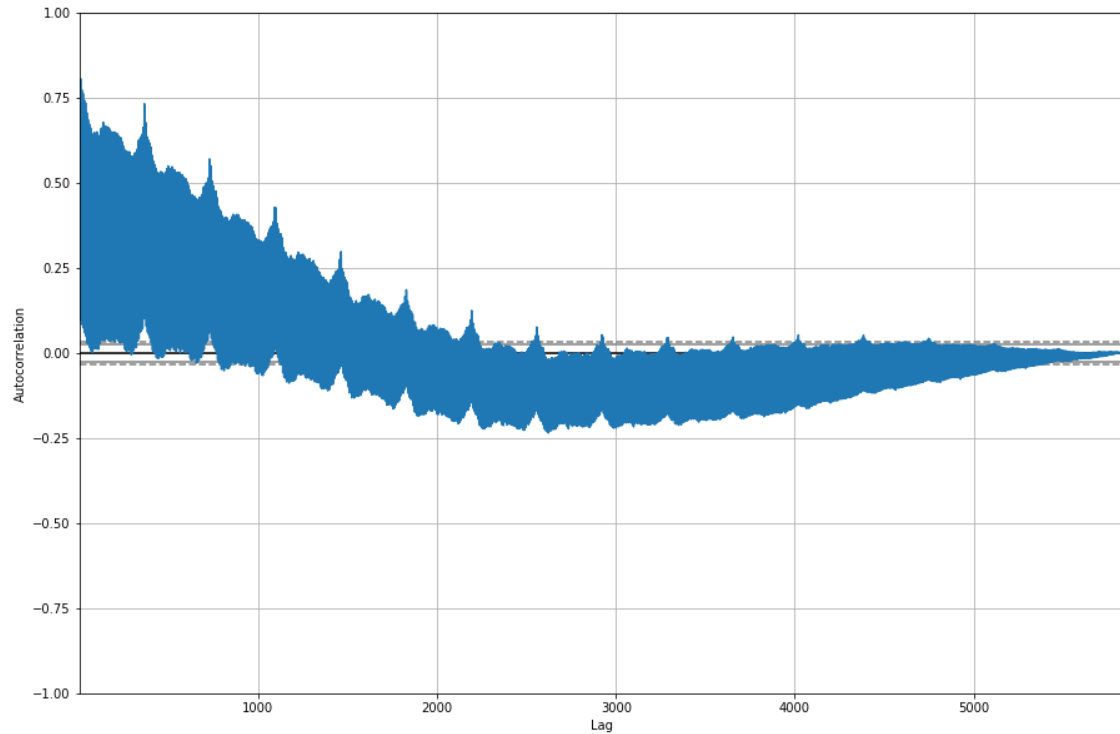
**Station-Id : 40780**
**Station-Name : Central Park**
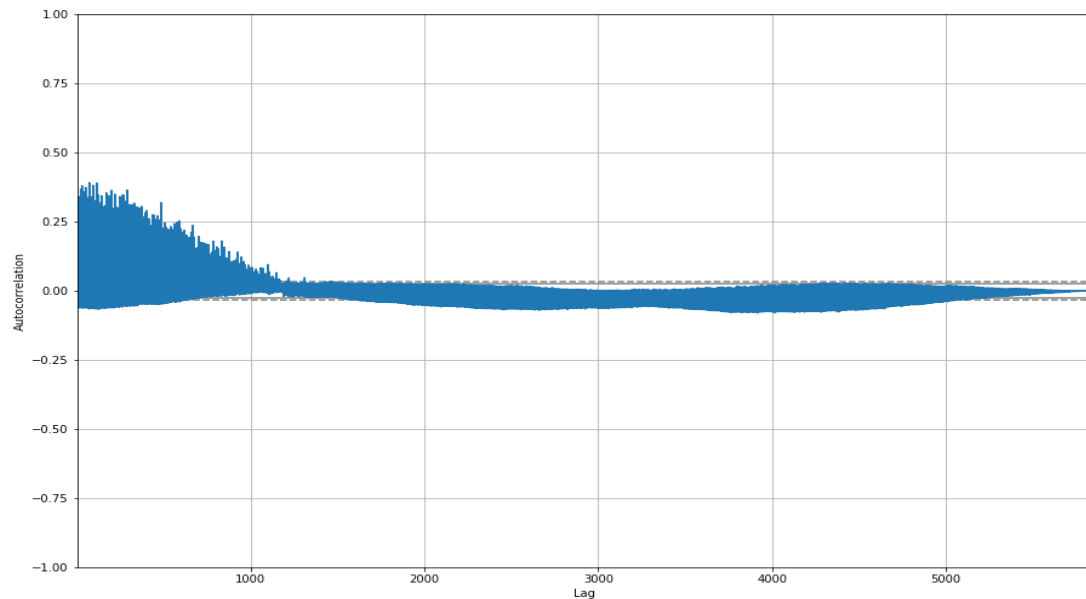


Autocorrelation Plot suggests that recent years (around 3 years) historical data should be sufficient (correlation) for the time series modelling.
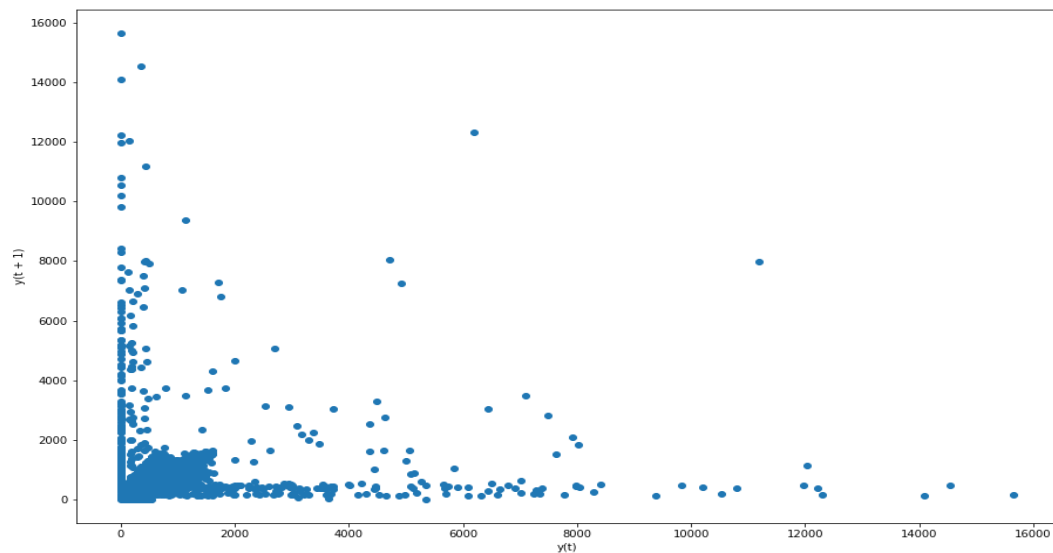


The Lag-1 Plot Corroborates the Autocorrelation plot that there is a strong correlation with the past day till around first 3 years of recent data.

**Forecasted_Results_Column_Set** =
['ds','trend','yhat_lower','yhat_upper','trend_lower','trend_upper'    'additive_terms', 'additive_terms_lower', 'additive_terms_upper','holidays_in_data_lower','holidays_in_data_upper', 'weekly', 'weekly_lower', 'weekly_upper', 'yearly',  'yearly_lower',  'yearly_upper', 'multiplicative_terms', 'multiplicative_terms_lower', 'multiplicative_terms_upper','yhat']


##################################################################

# Plots for station (40850, Library)
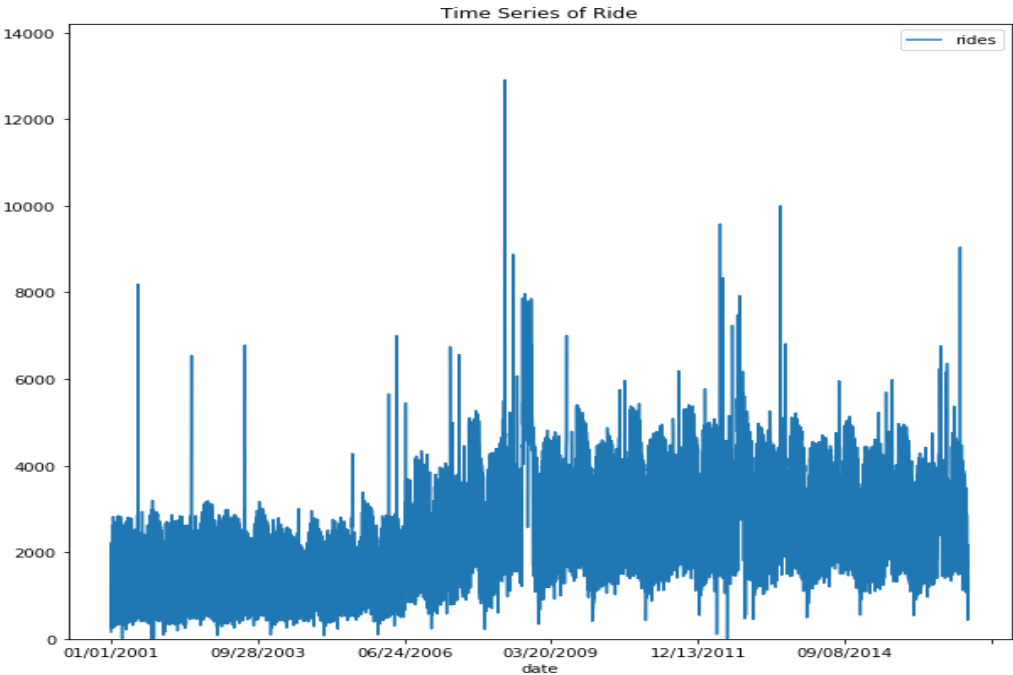


Time Series Line Plot.
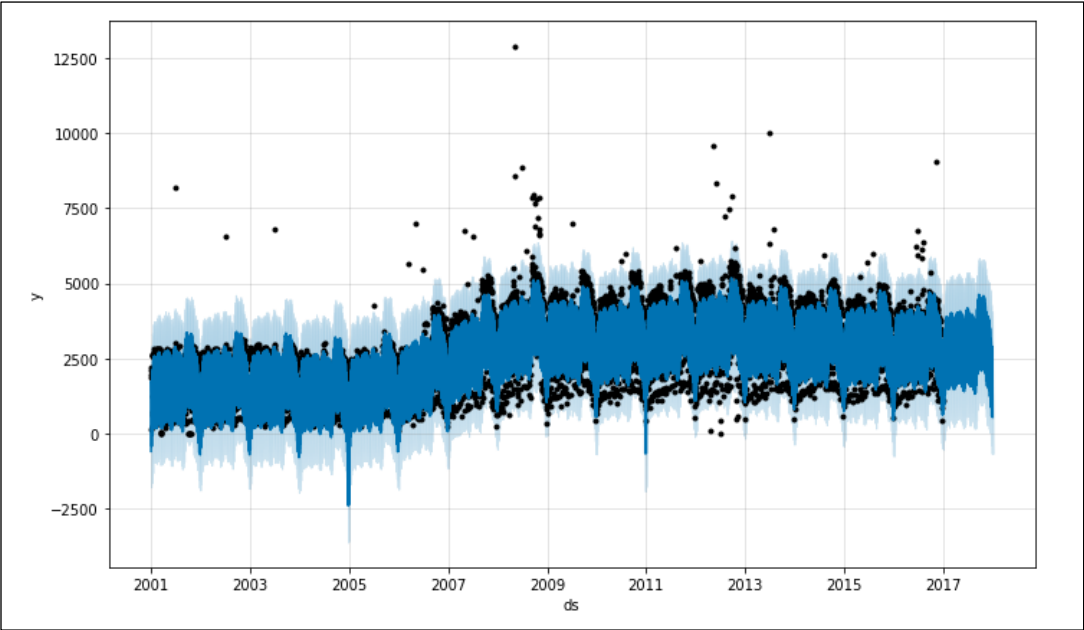


Time Series Line Plot in Prophet

Decomposition of time-series into Trend, Holidays, Weekly and Yearly Seasonality

###############################################################

**Plots for station (40780, Central Park)**
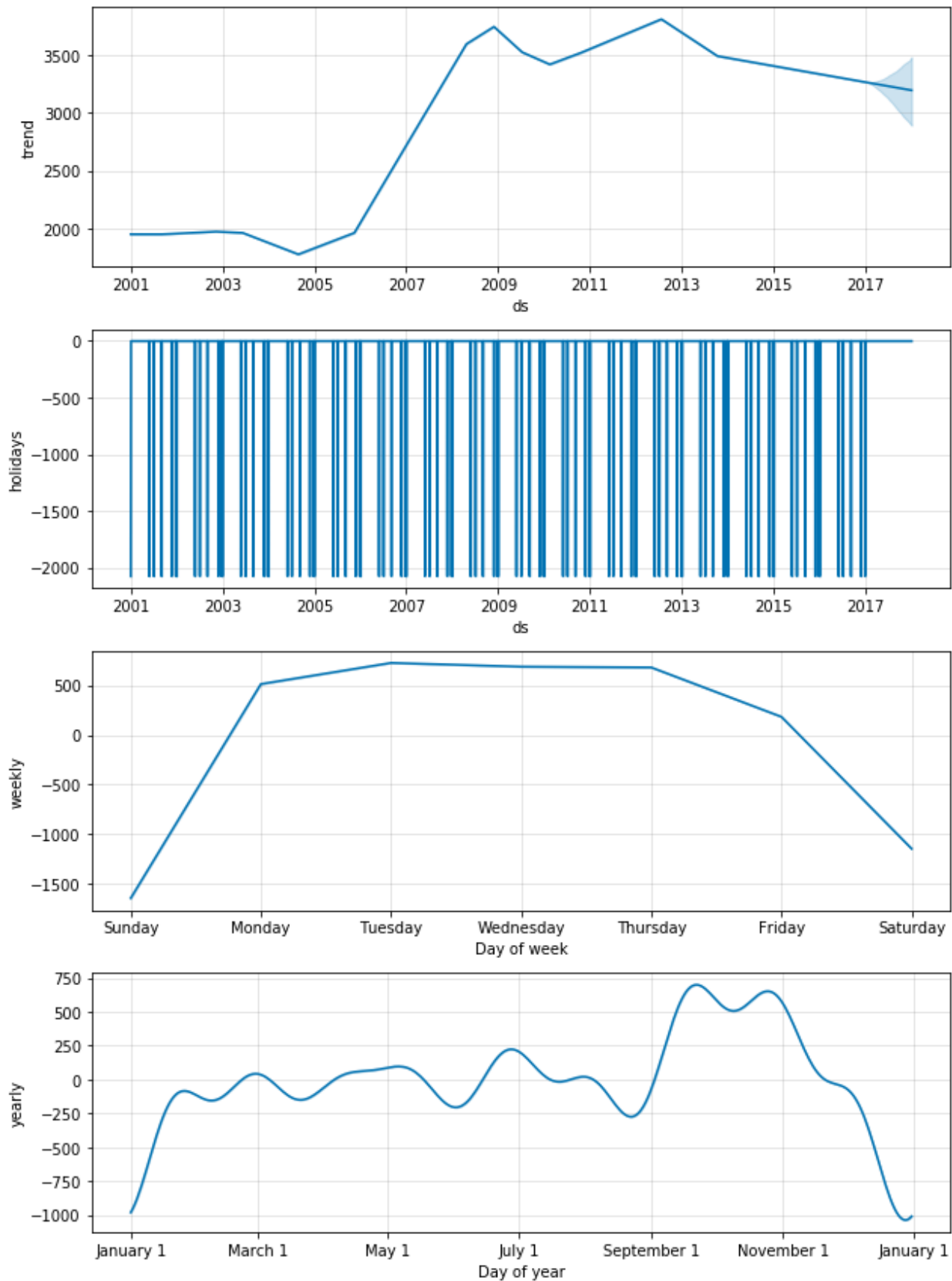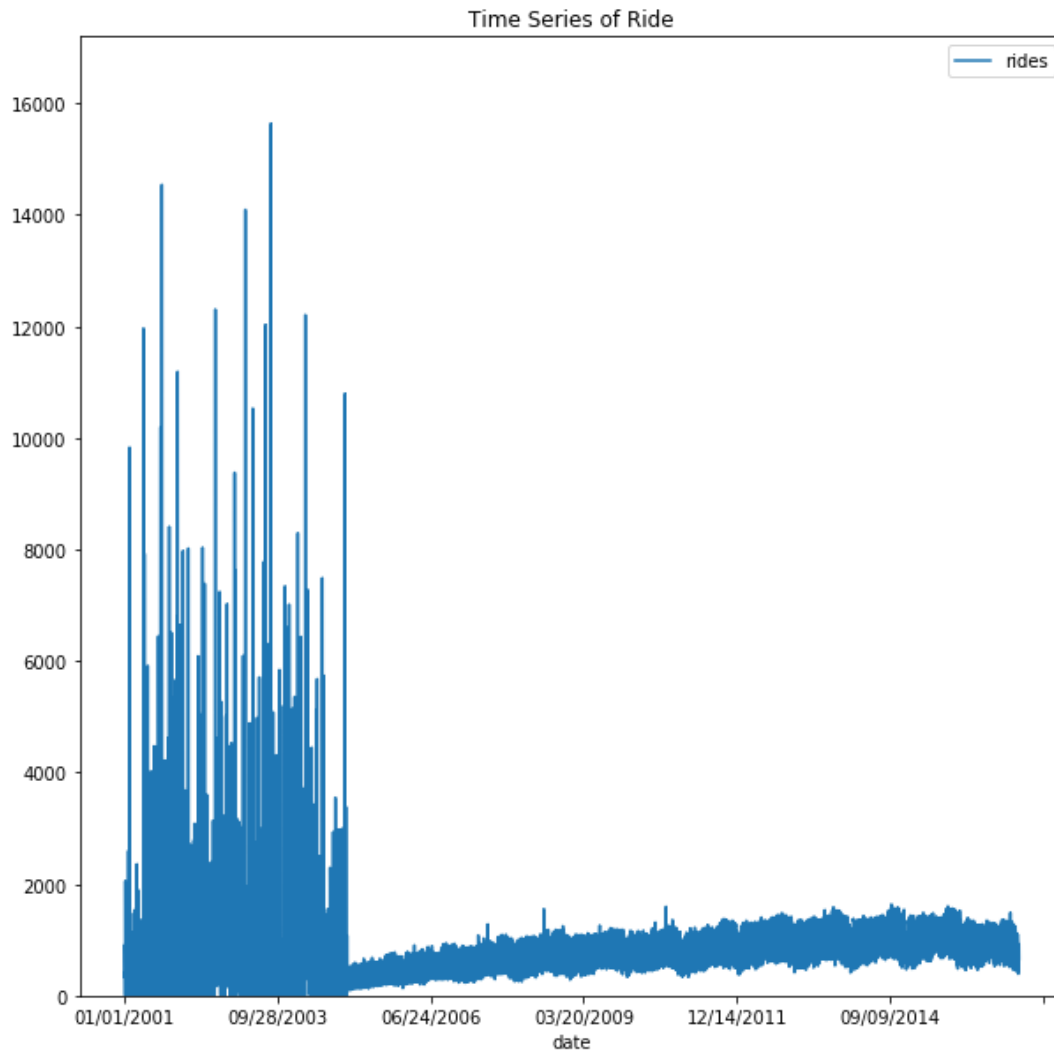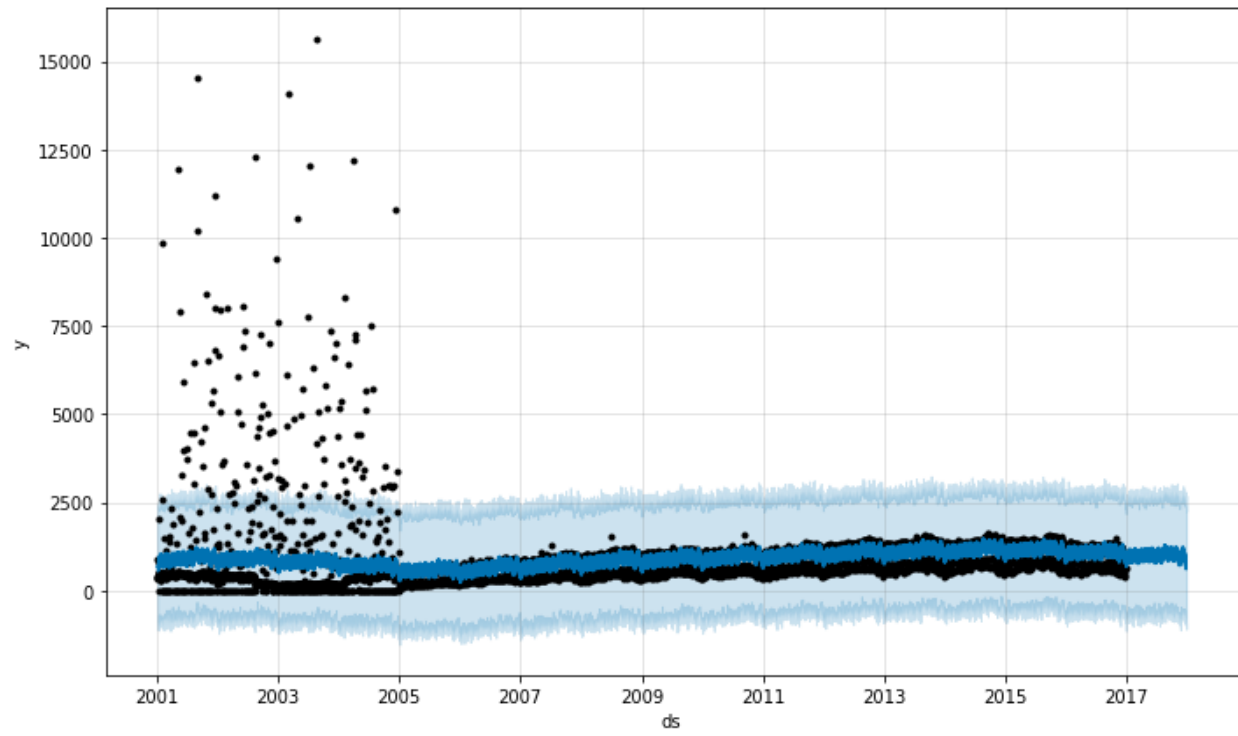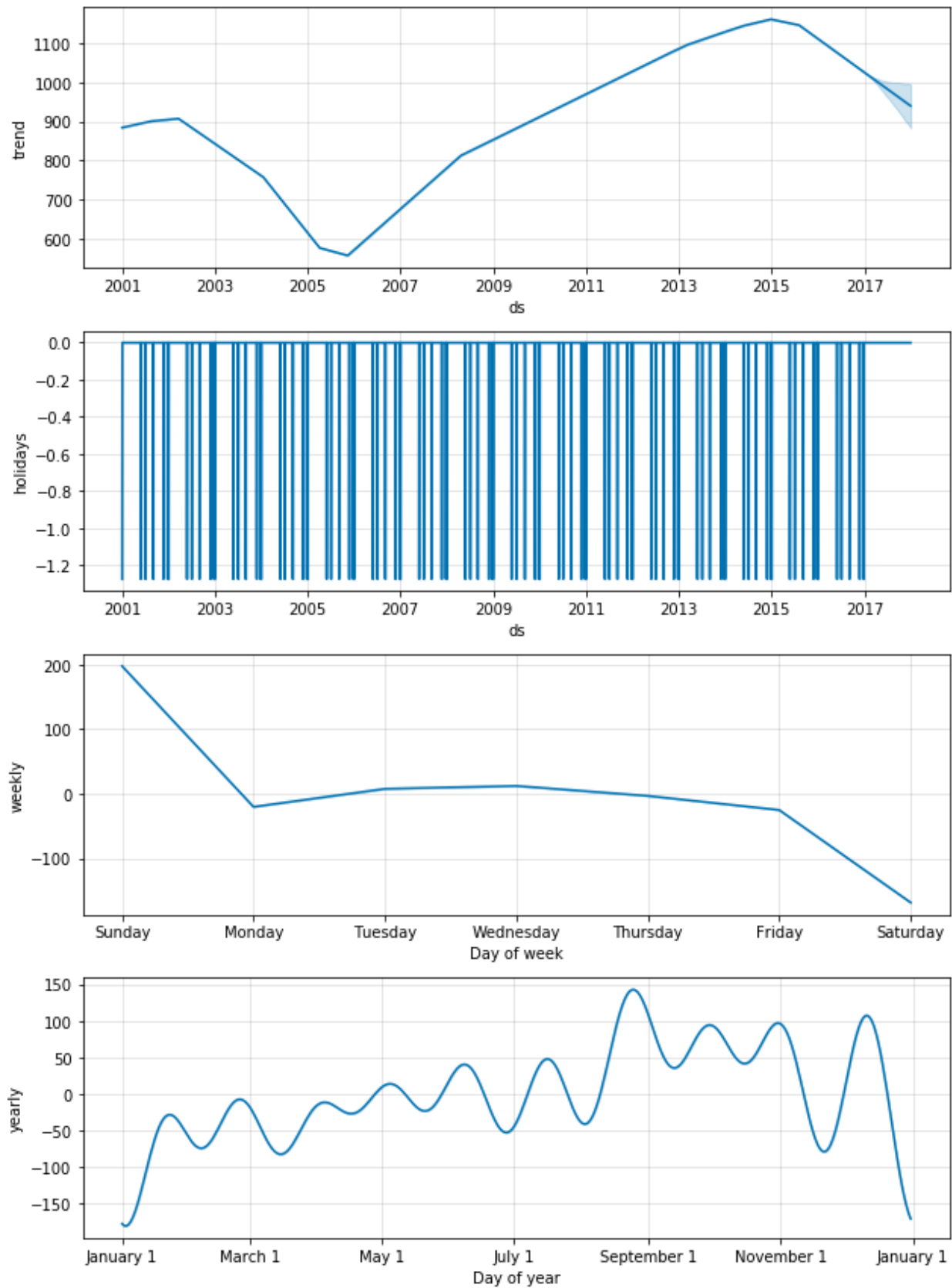


Time Series Line Plot.

Time Series Line Plot in Prophet.

Decomposition of time-series into Trend, Holidays, Weekly and Yearly Seasonality

**What did you learn from the data?**

For Station -> 40850, Library – Ridership is high on the weekdays and falls on the weekend. I have observed that there is weekly and yearly seasonality. Seasonality in the data is large at the start of the time series and small at the end. In this time series, the seasonality grows with the trend, hence I have used multiplicative seasonality.

For Station -> 40780, Central Park – Ridership is low on the weekdays and high on Sunday. I have observed that there is weekly and yearly seasonality. In this time series, the seasonality seems to be a constant additive factor (default Prophet)

This study should be extended to cohort-based analysis (semi-automated, manual intervention required).

**Which aspects should you consider for this model?**

**Growth Trend Pattern :** Chicago's population will increase in the future. Different geographical locations will have varied growths but there should be an upward bound to this expansion (looking at the asymptotic nature of the trend) that is suggestive of logistic growth.

**Time-Series for every Station :** The forecasting should be done station wise as the characteristics of each station is unique and modeling it individually captures the station specific patterns.

**Recency of the Historical Data :** From the autocorrelation plots we can notice there is a strong correlation with recent past dates (around 6 years for Library and around 3 years for Central Park)

**Seasonality Effects :** I have observed there is a pronounced seasonality weekly and yearly in the series. My model (Prophet) should be able to detect and infer from multi modal yearly seasonality

**Please explain how you built the model and justify the choices you made.**

I have used Facebook's Prophet Library for the Univariate Time Series Analysis (Each Station is modelled as an Individual Univariate Time-Series). First, I augmented the Dataset with columns Year','Month','Day' to sort the dataset for chronological ordering to prepare the time series. Then segmented the data into Training Set (2001-2016) and Testing Set (2017). I build a custom Holiday list for the Training Data to feed into the Prophet Model.

Ideally while scaling, we have to identify cohorts for hyper-parameter tuning. Each cohort can then run an automated set of parameters but this is again subjective and manual parameter changing should be done after certain time intervals if the underlying data changes.

In this prototype case, I am manually changing parameters for each of the two stations under investigation. 40850 (Library) should be modeled with a logistic growth trend model. I use a configurable cap with an assumed max limit of max_ridership*1.1  (1.1 is a assumed constant that can also be an increasing sequence)

I build a model iterator that iterates over per data station, aggregates the Data as a Time-Series. It then builds a Prophet Model with the chosen Hyper-Parameters.

**Hyper-Parameters Tuned**

*CHANGEPOINTS* - Denotes timeslot where Sudden and Abrupt changes occurred in the trend. From the literature survey, I have concluded that automatic detection by Prophet allows the trend to adapt appropriately.

*CHANGEPOINT_RANGE*  - Proportion of the Time-Series where potential change points are inferred

*INTERVAL_WIDTH* - Uncertainty Interval to produce a confidence interval (95%) around the forecast.

*YEARLY_SEASONALITY*

*WEEKLY_SEASONALITY*

*CHANGEPOINT_PRIOR_SCALE*  - Denotes trend flexibility. Model attains greater flexibility when value is increased. Decrease in value increase Rigidity of the model.

################################################################################

**How would you evaluate this model to ensure it would be robust for production usage?**

For Initial evaluation, I have taken the following metrics :

**MAE - Mean Absolute Error**
Average Difference between Original and Predicted Values. It measures the distance of how far the predictions are from actual.

**MSE - Mean Squared Error**
Similar to MAE – MSE takes the average of the square of the difference between the original values and the predicted values. Larger errors are magnified (become more pronounced) than smaller errors.

**RMSE – Root Mean Squared Error**
RMSE is a square rooted average of the normalized distance between the vector of predicted values and the vector of observed values.

**MAPE - Mean Absolute Percentage Error**

Error metric depicting Weighted Version of MAE in a percentage format. Absolute error is divided by the target value, giving relative error.

### Results from Initial Evaluation

| | Station_Id = 40850 | | | | | | |
|---|---|---|---|---|---|---|---|
| | Station_Name = Library | | | | | | |
| EXPERIMENT_NUM | CHANGEPOINT_RANGE | INTERVAL_WIDTH | CHANGEPOINT_PRIOR_SCALE | MAPE | MAE | MSE | RMSE |
| 1 | 0.95 | 0.95 | 0.005 | 19.60 | 460.80 | 479552.90 | 692.49 |
| 2 | 0.8 | 0.95 | 0.005 | 20.04 | 473.95 | 492210.47 | 701.57 |
| 3 | 0.9 | 0.95 | 0.01 | 16.98 | 382.20 | 418715.56 | 647.08 |
| 4 | 0.9 | 0.95 | 0.1 | 15.15 | 335.45 | 386382.22 | 621.59 |
| 5 | 0.9 | 0.95 | 0.5 | 15.23 | 336.85 | 387473.40 | 622.47 |
| 6 | 0.8 | 0.95 | 0.5 | 15.32 | 338.50 | 388384.78 | 623.20 |
| 7 | 0.9 | 0.95 | 0.3 | 15.14 | 335.33 | 386295.19 | 621.52 |
| 8 | **0.9** | **0.99** | **0.25** | **15.13** | 335.09 | 386072.72 | 621.35 |

| | Station_Id = 40780 | | | | | | |
|---|---|---|---|---|---|---|---|
| | Station_Name = Central Park | | | | | | |
| EXPERIMENT_NUM | CHANGEPOINT_RANGE | INTERVAL_WIDTH | CHANGEPOINT_PRIOR_SCALE | MAPE | MAE | MSE | RMSE |
| 1 | 0.8 | 0.95 | 0.001 | 35.20 | 218.39 | 119256.96 | 345.33 |
| 2 | 0.8 | 0.95 | 0.01 | 36.77 | 224.51 | 131274.97 | 362.31 |
| 3 | 0.9 | 0.95 | 0.01 | 37.11 | 226.46 | 133489.83 | 365.36 |
| 4 | **0.9** | **0.95** | **0.1** | **33.29** | 247.66 | 95821.66 | 309.55 |
| 5 | 0.9 | 0.95 | 0.5 | 33.58 | 265.83 | 98813.72 | 314.34 |
| 6 | 0.8 | 0.99 | 0.25 | 33.47 | 226.59 | 100395.84 | 316.85 |
| 7 | 0.8 | 0.95 | 0.1 | 33.56 | 224.53 | 101798.86 | 319.05 |
| 8 | 0.95 | 0.95 | 0.1 | 33.31 | 249.29 | 95896.60 | 309.67 |

**For making the models robust for production usage**

The appropriate utilization of forecasted results is done only if there's enough time and opportunity for the team to optimize. I would recommend to forecast 7 days ahead to leave that window for decision making. But this window can be curated based on the need of the business. The design should be flexible to extend it to bi-weekly or monthly predictions. I will keep the model serving part as modular. There will be separate modules for data handling, preprocessing, feature generation, retraining, model evaluation and forecasting. I will build a logging framework to track output from each module so that it is easy to monitor and debug. Whenever there are large errors, I will setup notifications using default tools in job schedulers.

## References

Forecasting at Scale, Sean J. Taylor, Benjamin Lethamy, Facebook, Menlo Park, California, United States

Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M. A., Guo, J., Li, P. & Riddell, A. (2017), 'Stan: A probabilistic programming language', Journal of Statistical Software 76(1).