Notebook

December 14, 2019

0.0.1 Question 1d

In the following cell, print a summary of the data selection and cleaning you performed. For example, you should print something like: "Of the original 1000 trips, 21 anomolous trips (2.1%) were removed through data cleaning, and then the 600 trips within Manhattan were selected for further analysis." (Note that the numbers in this example are not accurate.)

Your Python code should not include any number literals, but instead should refer to the shape of `all_taxi`, `clean_taxi`, and `manhattan_taxi`. Your response will be scored based on whether you generate an accurate description and do not include any number literals in your Python expression, but instead refer to the dataframes you have created.

One way to do this is with Python's f-strings. For instance,

```
name = "Joshua"
print(f"Hi {name}, how are you?")
```

prints `Hi Joshua, how are you?`.

Please ensure that your Python code does not contain any very long lines, or we can't grade it.

```
In [242]: og_rows = all_taxi.shape[0]
          clean_rows = clean_taxi.shape[0]
          row_diff = abs(clean_rows - og_rows)
          percent_diff = np.round(row_diff/og_rows, 3)
          man_rows = manhattan_taxi.shape[0]

          print(f"Of the original {og_rows} trips, {row_diff} anomolous trips {percent_diff} were \
          removed through data cleaning, \nand then the {man_rows} trips within Manhattan were \
          selected for further analysis.")
```

```
Of the original 97692 trips, 1247 anomolous trips 0.013 were removed through data cleaning,
and then the 82800 trips within Manhattan were selected for further analysis.
```

0.0.2 Question 2b

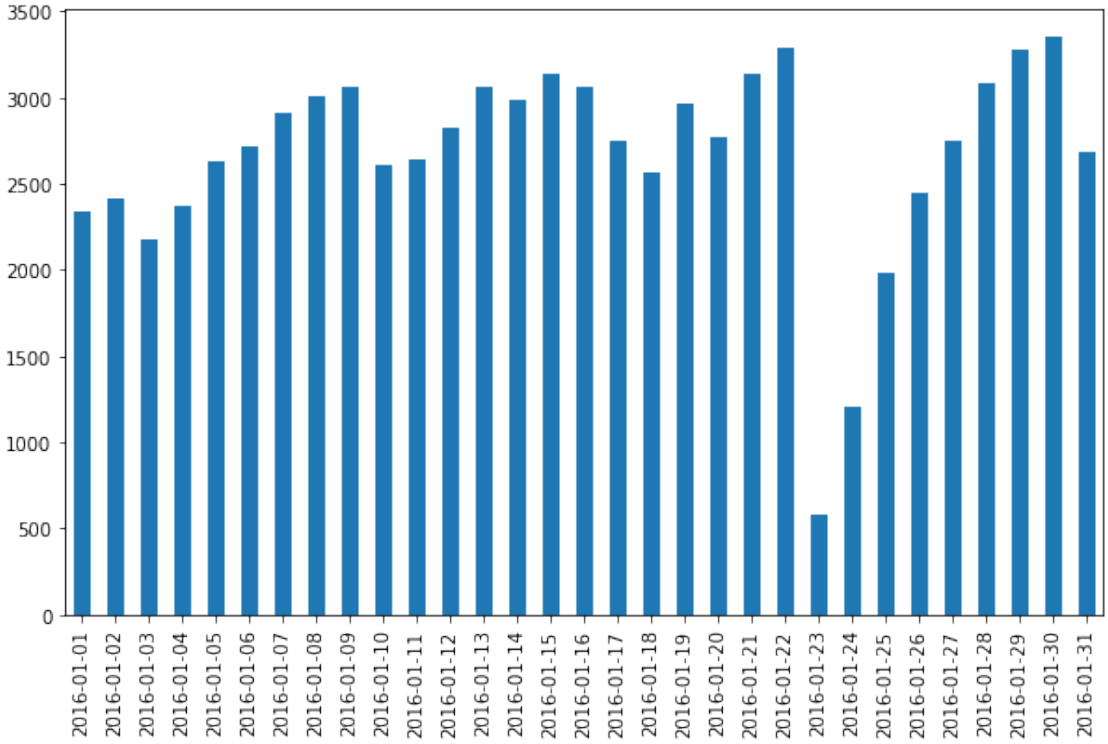
Create a data visualization that allows you to identify which dates were affected by the historic blizzard of January 2016. Make sure that the visualization type is appropriate for the visualized data.

Hint: How do you expect taxi usage to differ on blizzard days?

In [73]:
```
#Blizzard took place Jan 22 night-Jan 24. Taxi rides fell drastically on Jan 23-24.

plt.figure(figsize = (10,6))
manhattan_taxi['date'].value_counts().sort_index().plot(kind='bar')
```

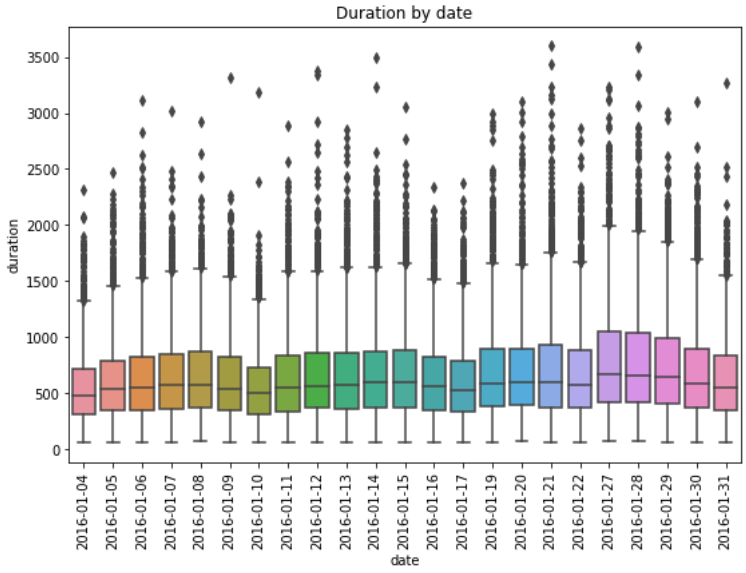
Out[73]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fc9573255c0>
```



0.0.3 Question 3a

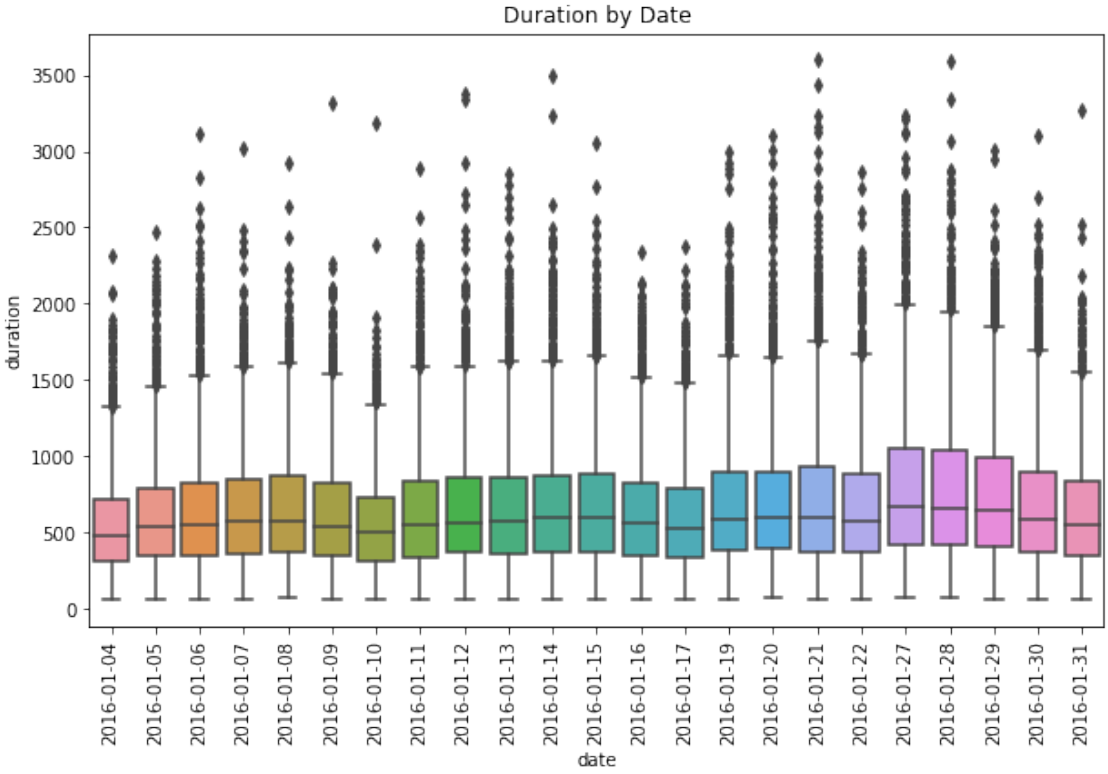
Create a box plot that compares the distributions of taxi trip durations for each day **using `train` only**. Individual dates shoud appear on the horizontal axis, and duration values should appear on the vertical axis. Your plot should look like the following.

Hint: Use ***`sns.boxplot`****.*




```
In [77]: plt.figure(figsize = (10,6))
         sorted_train = train.sort_values('date')
         sns.boxplot(x = 'date', y = 'duration', data = sorted_train)
         plt.xticks(rotation=90)
         plt.title("Duration by Date")
```

```
Out[77]: Text(0.5, 1.0, 'Duration by Date')
```


Duration by Date
duration
3500
3000
2500
2000
1500
1000
500
0
2016-01-04
2016-01-05
2016-01-06
2016-01-07
2016-01-08
2016-01-09
2016-01-10
2016-01-11
2016-01-12
2016-01-13
2016-01-14
2016-01-15
2016-01-16
2016-01-17
2016-01-19
2016-01-20
2016-01-21
2016-01-22
2016-01-27
2016-01-28
2016-01-29
2016-01-30
2016-01-31
date

0.0.4 Question 3b

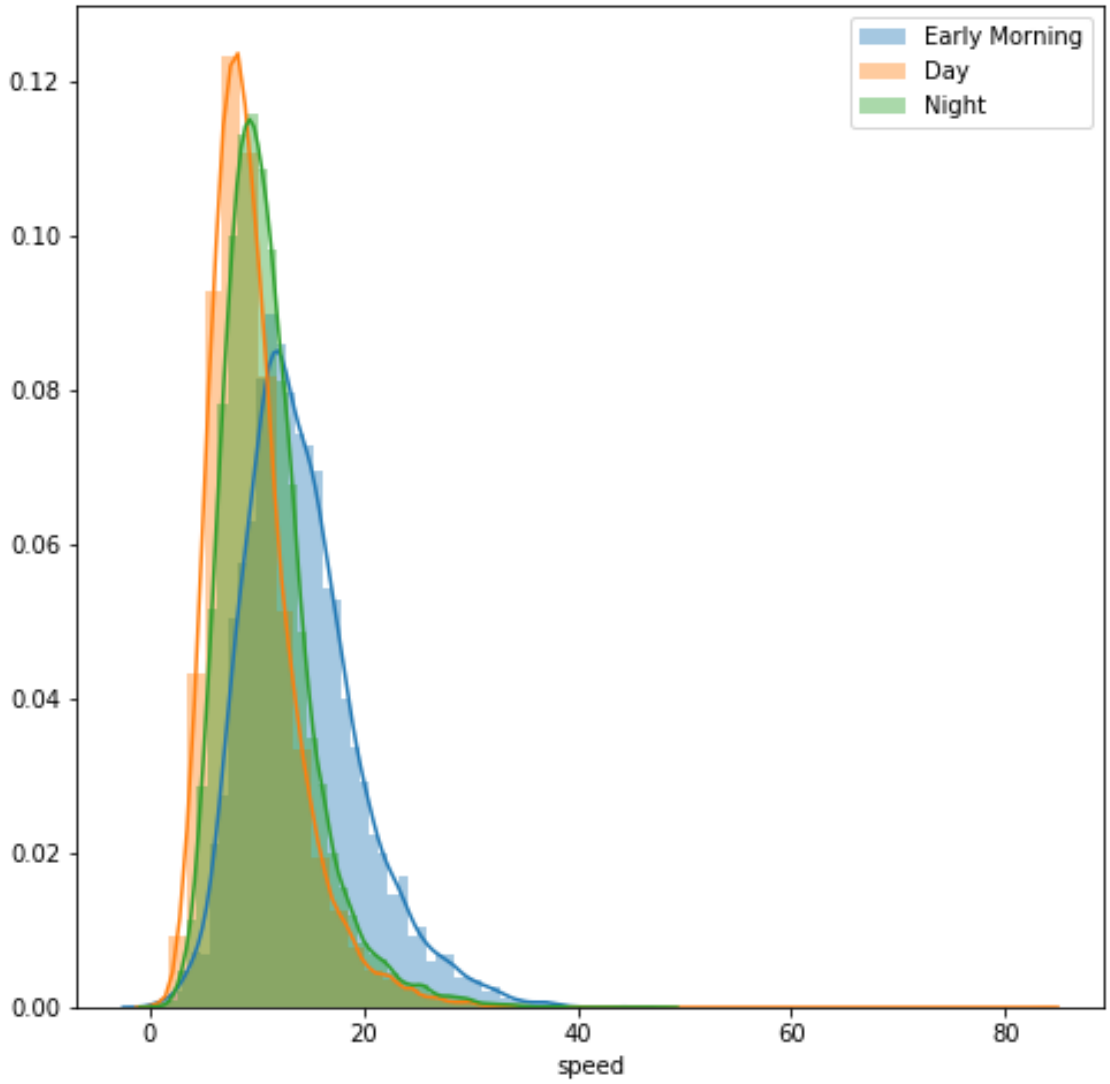
In one or two sentences, describe the assocation between the day of the week and the duration of a taxi trip. This question will be graded on whether your answer is justified by your boxplot and if it is at least somewhat meaningful.

Note: The end of Part 2 showed a calendar for these dates and their corresponding days of the week.

This plot shows slight difference in distribution on length of trips across the week. Particularly, we see that on weekdays in Jan 2016, the trips are slightly longer (peaking around Wednesday/Thursday) than on weekends.

0.0.5 Question 3c

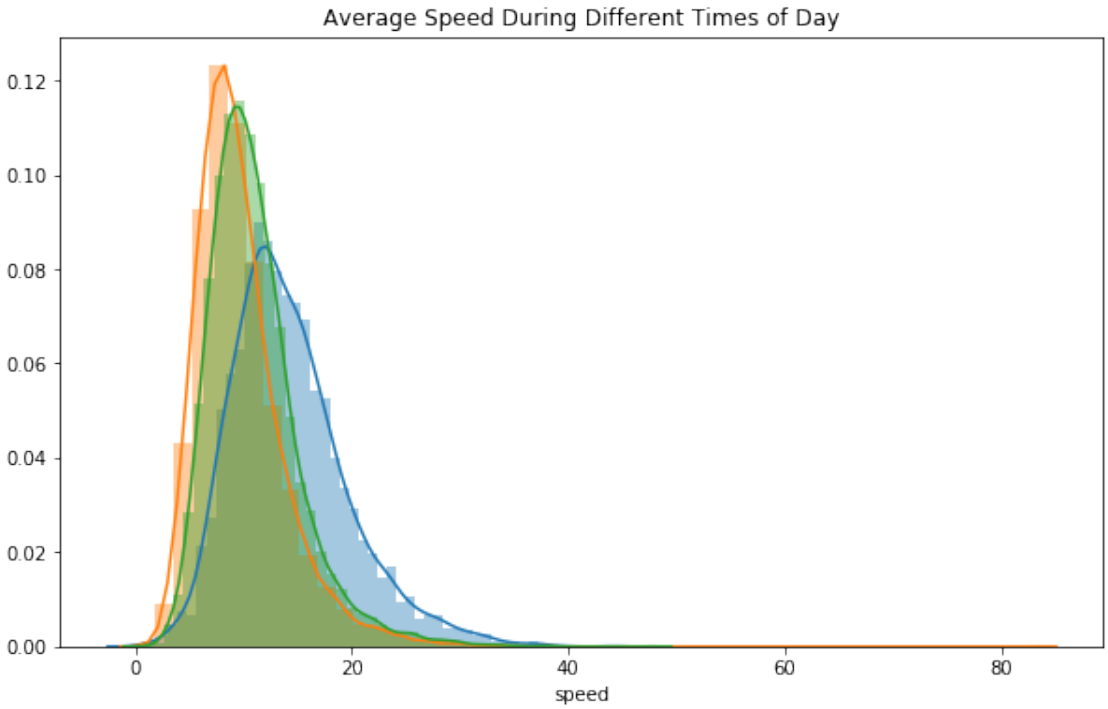
Use `sns.distplot` to create an overlaid histogram comparing the distribution of average speeds for taxi rides that start in the early morning (12am-6am), day (6am-6pm; 12 hours), and night (6pm-12am; 6 hours). Your plot should look like this:




```
In [79]: morning = train[train['period'] == 1]
         day = train[train['period'] == 2]
         night = train[train['period'] == 3]

         plt.figure(figsize = (10,6))
         sns.distplot(morning['speed'])
         sns.distplot(day['speed'])
         sns.distplot(night['speed'])
         plt.title("Average Speed During Different Times of Day")
```

```
Out[79]: Text(0.5, 1.0, 'Average Speed During Different Times of Day')
```


Average Speed During Different Times of Day
0.12
0.10
0.08
0.06
0.04
0.02
0.00
0
20
40
60
80
speed

0.0.6 Question 4e

In one or two sentences, explain how the `period` regression model could possibly outperform linear regression model, even when the design matrix of the latter includes one feature for each possible hour.

The period regression model uses the insight that the models for each of the periods can have different slopes. Specifically, the slope of the relationship between duration and period can be slightly different for the 3 different periods. Although the hours are present in the design matrix, distribution differences come out better when the hours are aggregated into their respective periods.