Notebook

April 30, 2019

0.0.1 Question 1d

Print a summary of the data selection and cleaning you performed. Your Python code should not include any number literals, but instead should refer to the shape of `all_taxi`, `clean_taxi`, and `manhattan_taxi`.

E.g., 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 the example above are not accurate.)

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

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.

```
In [12]: print( \
         """Of the %d original trips, %d anomolous trips (%f%%) were removed through data cleaning,
         and then the %d trips within Manhattan were selected for further analysis"""
         % (len(all_taxi),(len(all_taxi) - len(clean_taxi)), \
         (100 * (len(all_taxi) - len(clean_taxi)) / len(all_taxi)), len(manhattan_taxi)) \
         )
```

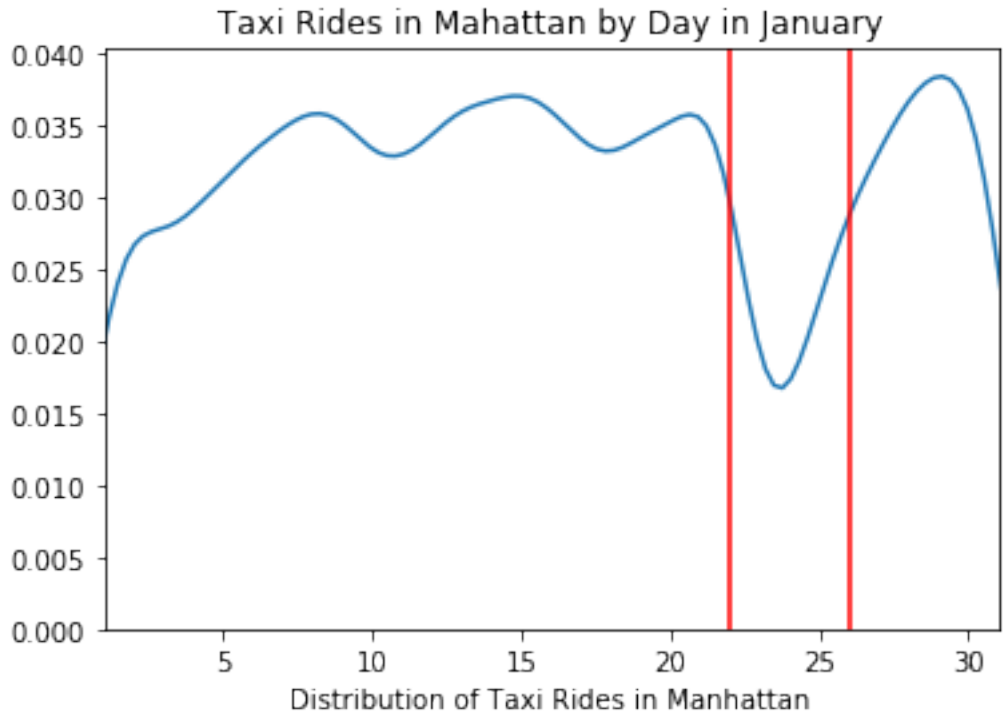
```
Of the 97692 original trips, 1247 anomolous trips (1.276461%) were removed through data cleaning,
and then the 82800 trips within Manhattan were selected for further analysis
```

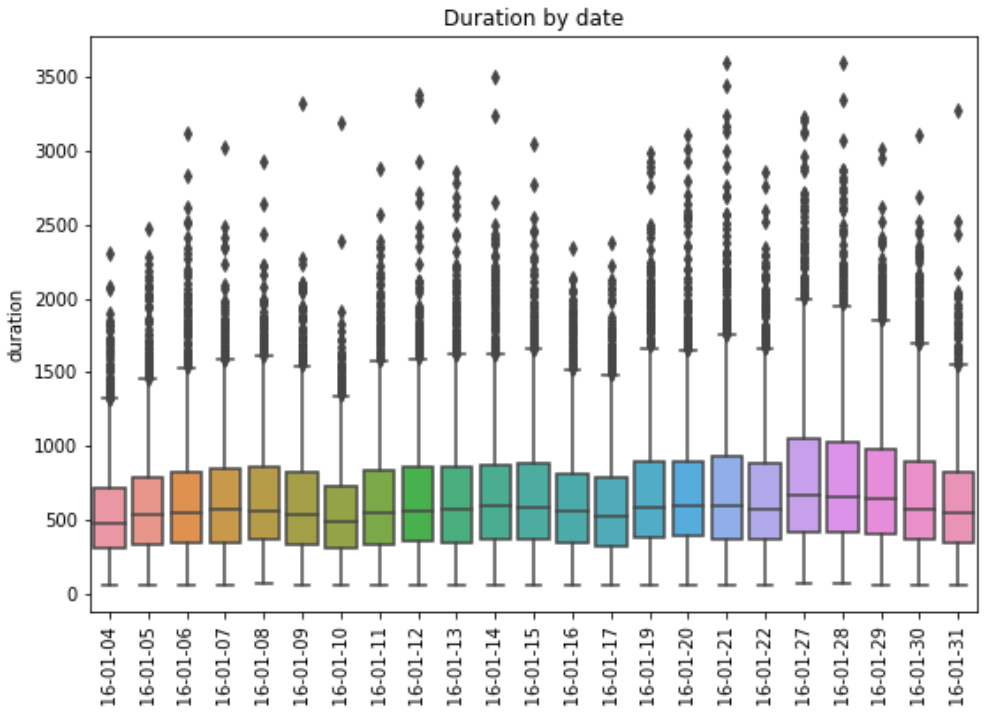
0.0.2 Question 2b

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.

```
In [15]: sns.distplot(pd.DatetimeIndex(manhattan_taxi['date']).day, hist = False, kde = True)
         plt.xlim((1, 31))
         plt.axvline(x=22, color='r')
         plt.axvline(x=26, color='r')
         plt.xlabel('Day in January')
         plt.xlabel('Distribution of Taxi Rides in Manhattan')
         plt.title('Taxi Rides in Mahattan by Day in January');
```

```
/srv/conda/envs/data100/lib/python3.6/site-packages/scipy/stats/stats.py:1713: FutureWarning: Using a n
  return np.add.reduce(sorted[indexer] * weights, axis=axis) / sumval
```

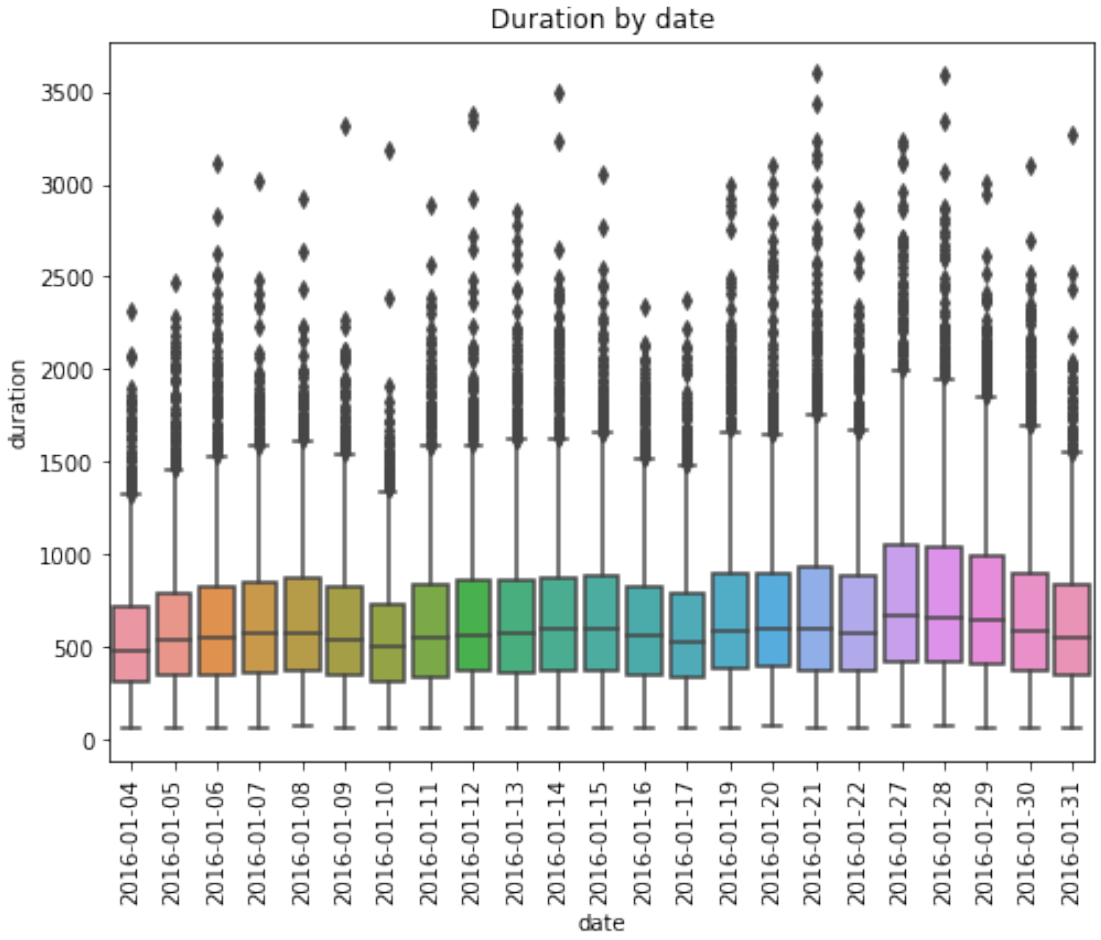




0.0.3 Question 3a

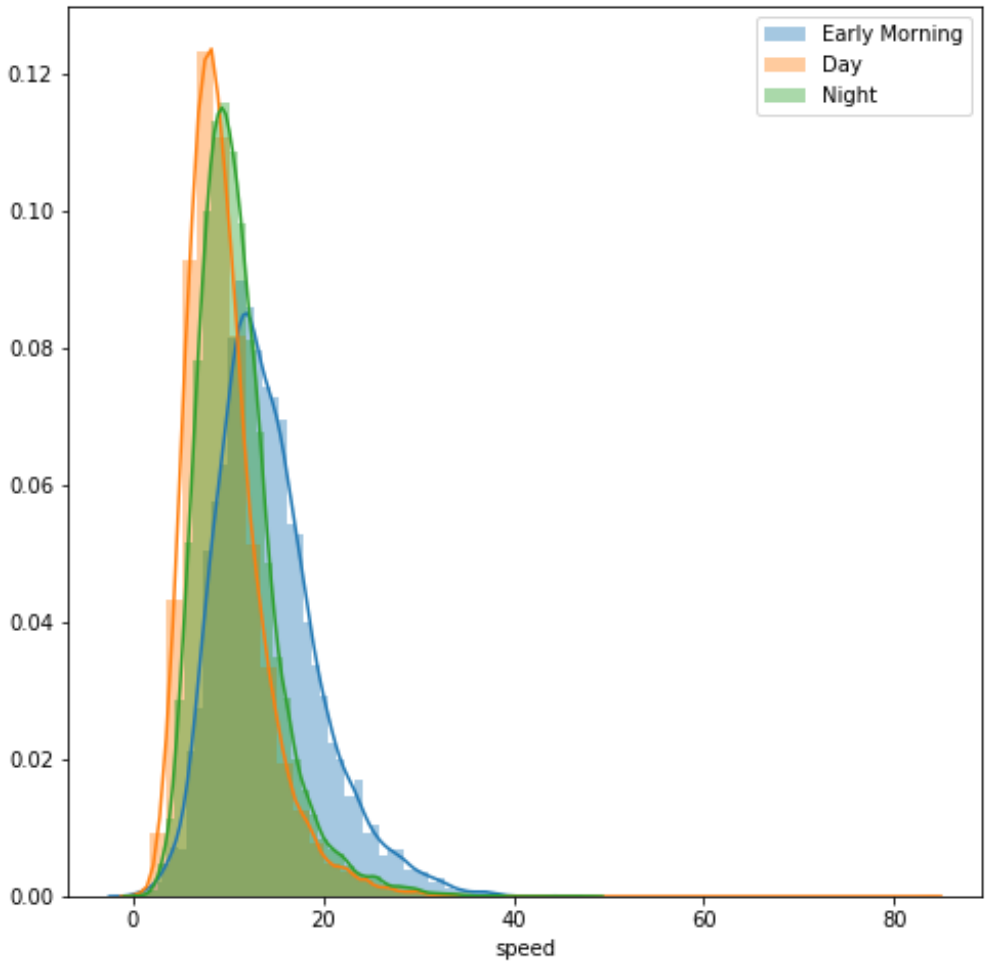
Create a box plot that compares the distributions of taxi trip durations for each day **using `train` only**. Individual dates shoud appear on the horizontal axis, and duration values should appear on the vertical axis. Your plot should look like this:

```
In [19]: plt.figure(figsize=(8, 6))
         ax = sns.boxplot(train['date'].sort_values(inplace=False), train['duration']);
         ax.set_xticklabels(ax.get_xticklabels(), rotation=90)
         plt.title('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.
Note: The end of Part 2 showed a calendar for these dates and their corresponding days of the week.
Write your answer here, replacing this text.



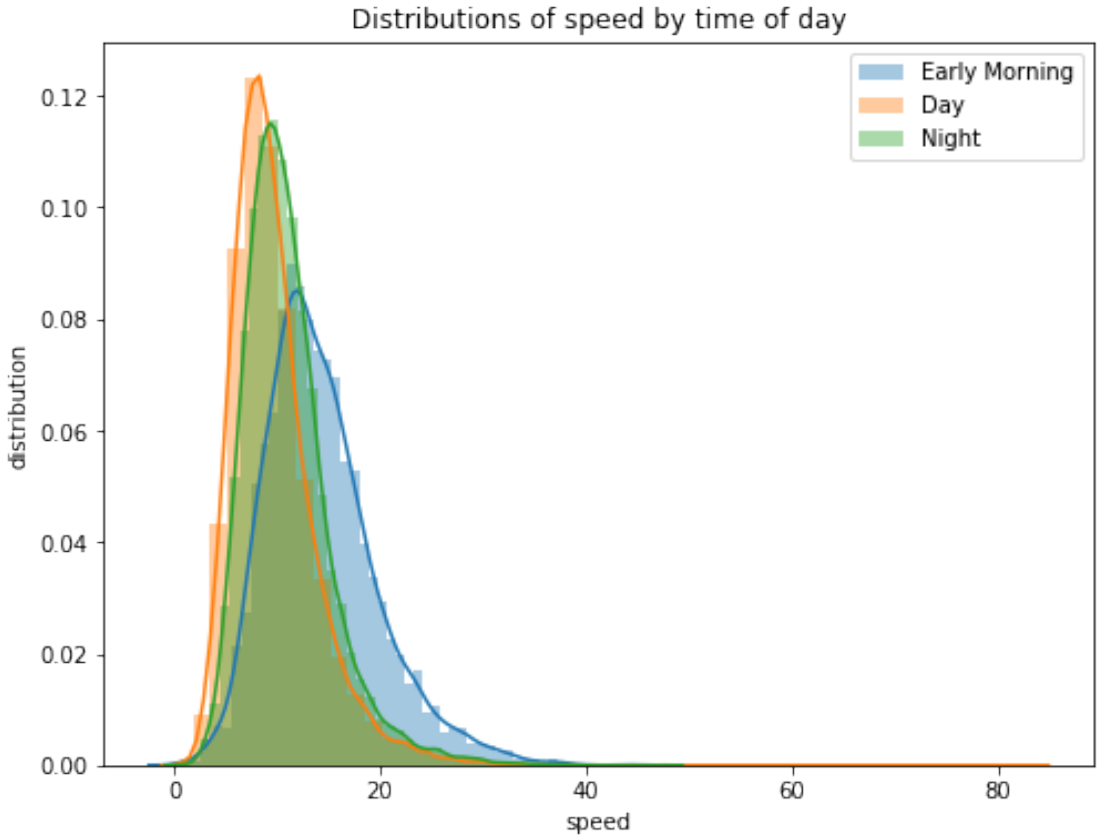
0.0.5 Question 3c

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 [21]: plt.figure(figsize=(8, 6))
         sns.distplot(train[train['period'] == 1]['speed'], hist=True, kde=True, label='Early Morning')
         sns.distplot(train[train['period'] == 2]['speed'], hist=True, kde=True, label='Day')
         sns.distplot(train[train['period'] == 3]['speed'], hist=True, kde=True, label='Night')

         plt.title('Distributions of speed by time of day')
         plt.xlabel('speed')
         plt.ylabel('distribution')
         plt.legend();
```

```
/srv/conda/envs/data100/lib/python3.6/site-packages/scipy/stats/stats.py:1713: FutureWarning: Using a n
  return np.add.reduce(sorted[indexer] * weights, axis=axis) / sumval
```


Distributions of speed by time of day
Early Morning
Day
Night
distribution
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 when the design matrix for linear regression already includes one feature for each possible hour, which can be combined linearly to determine the `period` value.

Splitting the model fitting into time periods allows each submodel to find features that fit the subset of data for that time period better, so the prediction for that time period will be more accurate (we've already determined that period is a good feature for determining speed and therefore indirectly duration in our EDA). Fitting to the data for each period then will fit the model to a more specific data set and reduce the variance and propensity to overfit for each submodel. For example, if location determines duration in period 1 than period 2, the period model will factor this in by weighting location more heavily in submodel 1.