**Uber-Data-Analysis**

* Data Analysis of Uber trips using Python, Pandas, and Jupyter Notebook. Seaborn and matplotlib visualization tools are used to analyse and develop different plots.
* Uber trips datasets available in kaggle.
* Dataset name is **uber-raw-data-apr14.csv** In this data set there are 4 columns like…
* Date/Time
* Latitude(Lat)
* Longitude(Lon)
* Base
* Based on this dataset analyze the data and drawing different plots.

**Importing required Packages**

import pandas

import seaborn

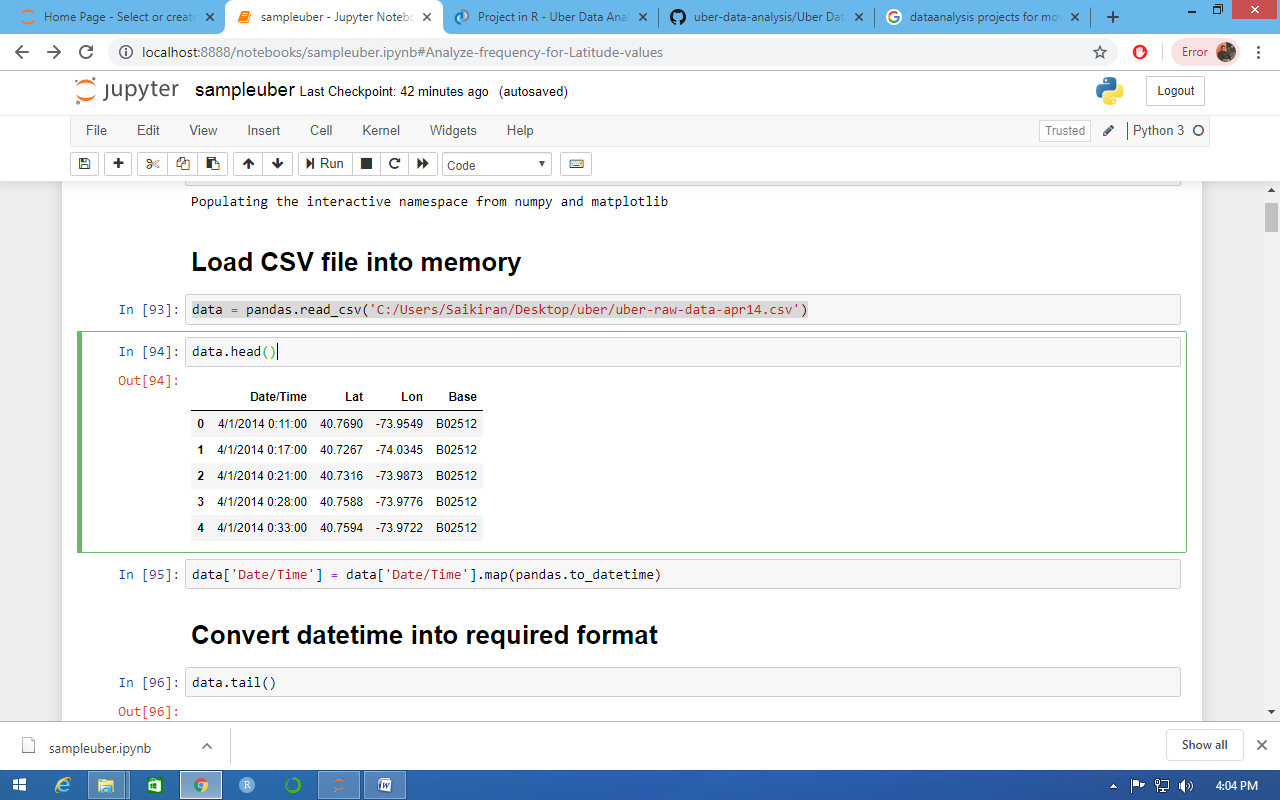
%pylab inline

When the**%pylab** magic function is entered at the IPython prompt, it triggers the import of various modules within Matplotlib. Inline is used for plot the graphs in IPython prompt only instead of ploting other windows.

**Load CSV file into memory**

data = pandas.read\_csv('C:/Users/Saikiran/Desktop/uber/uber-raw-data-apr14.csv')

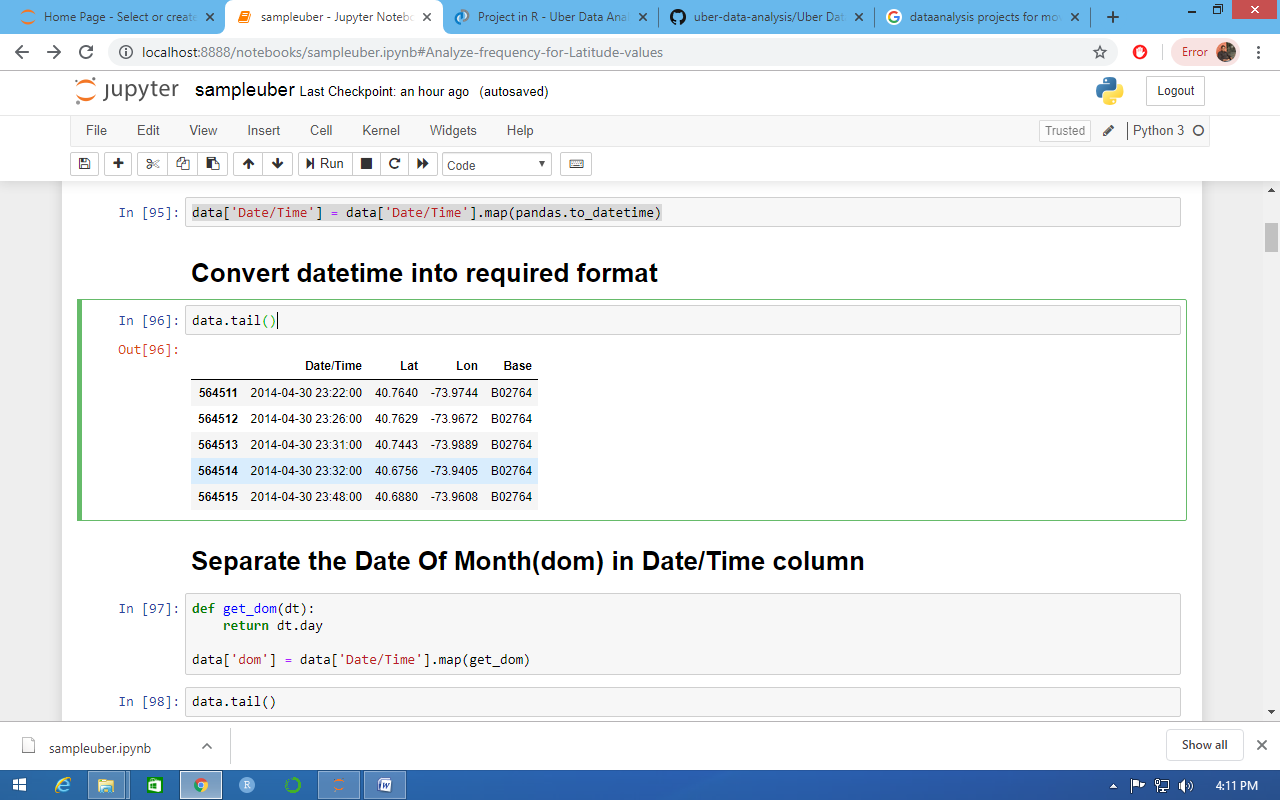
data.head()



**Convert datetime into required format**

data['Date/Time'] = data['Date/Time'].map(pandas.to\_datetime)

data.tail()



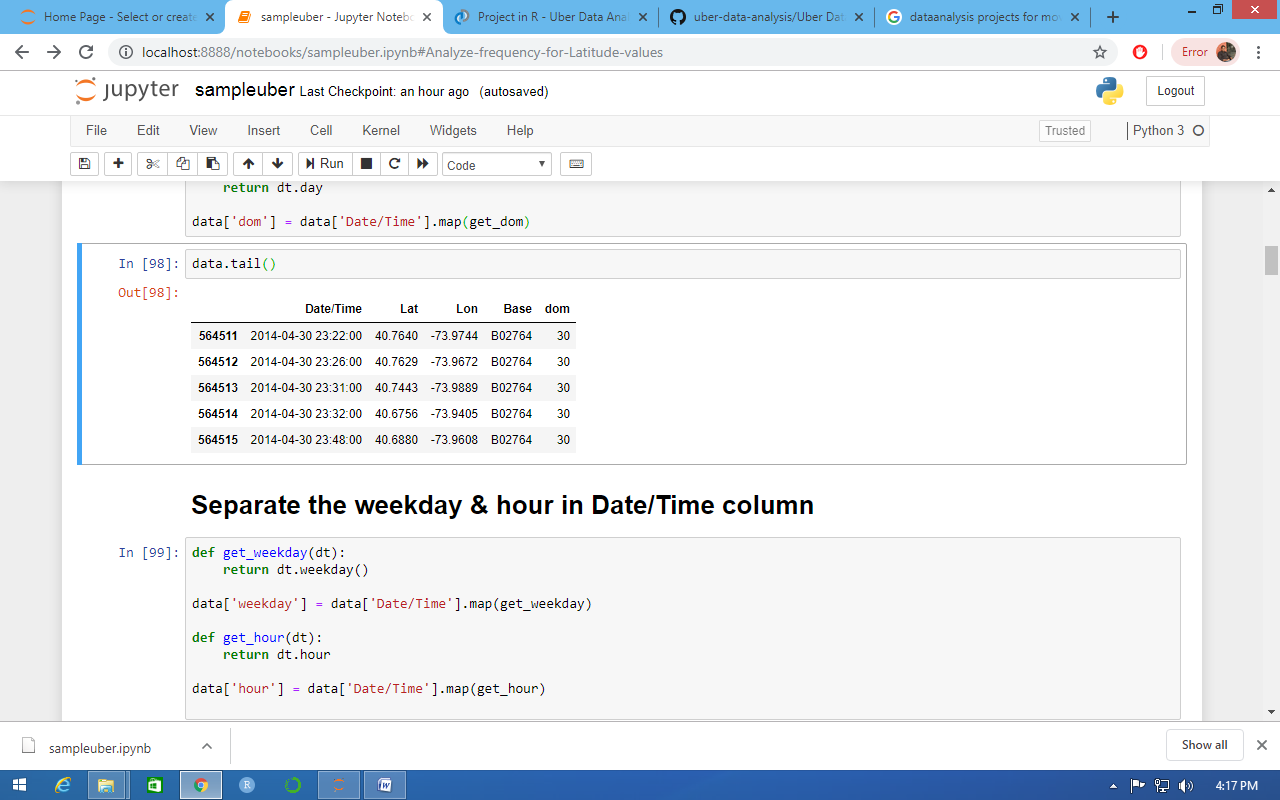
**Separate the Date Of Month(dom) in Date/Time column**

def get\_dom(dt):

return dt.day

data['dom'] = data['Date/Time'].map(get\_dom)

data.tail()



**Separate the weekday & hour in Date/Time column**

def get\_weekday(dt):

return dt.weekday()

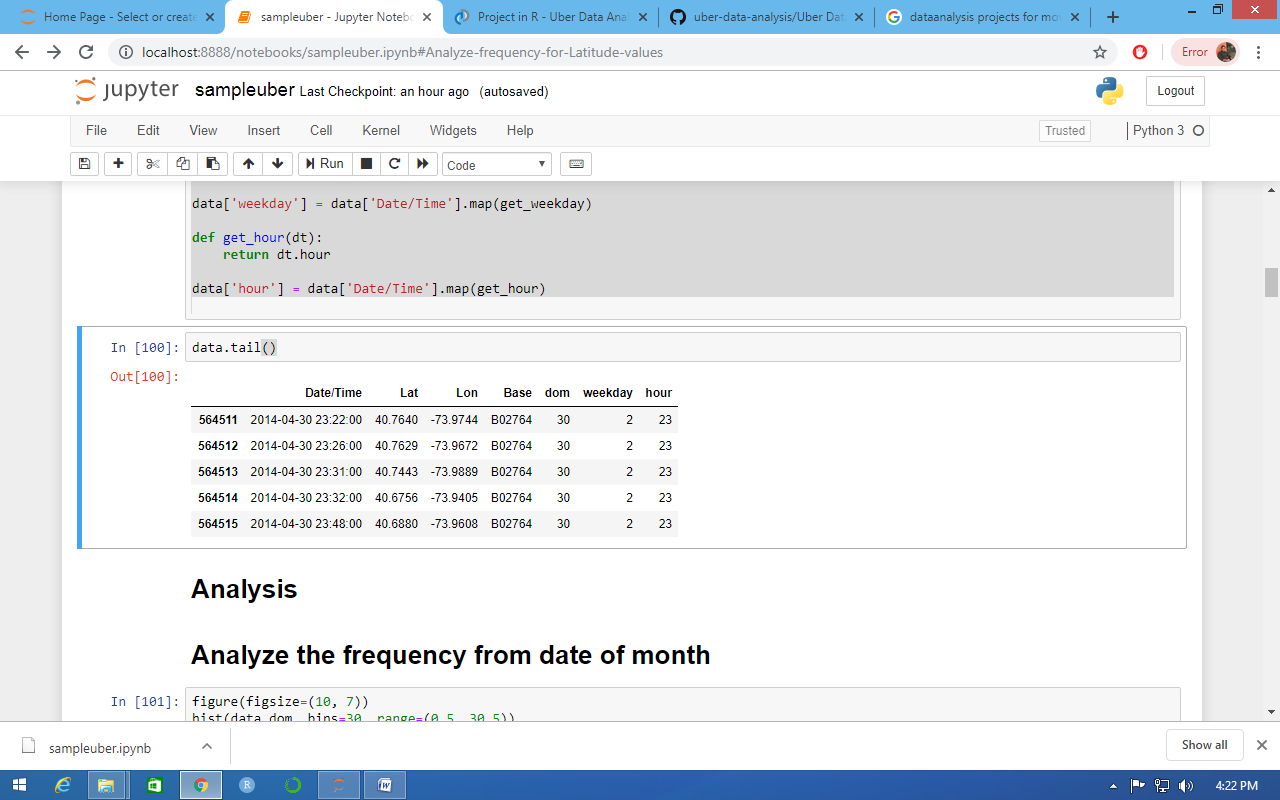
data['weekday'] = data['Date/Time'].map(get\_weekday)

def get\_hour(dt):

return dt.hour

data['hour'] = data['Date/Time'].map(get\_hour)

data.tail()



# Analysis

# Analyze the frequency from date of month

figure(figsize=(10, 7))

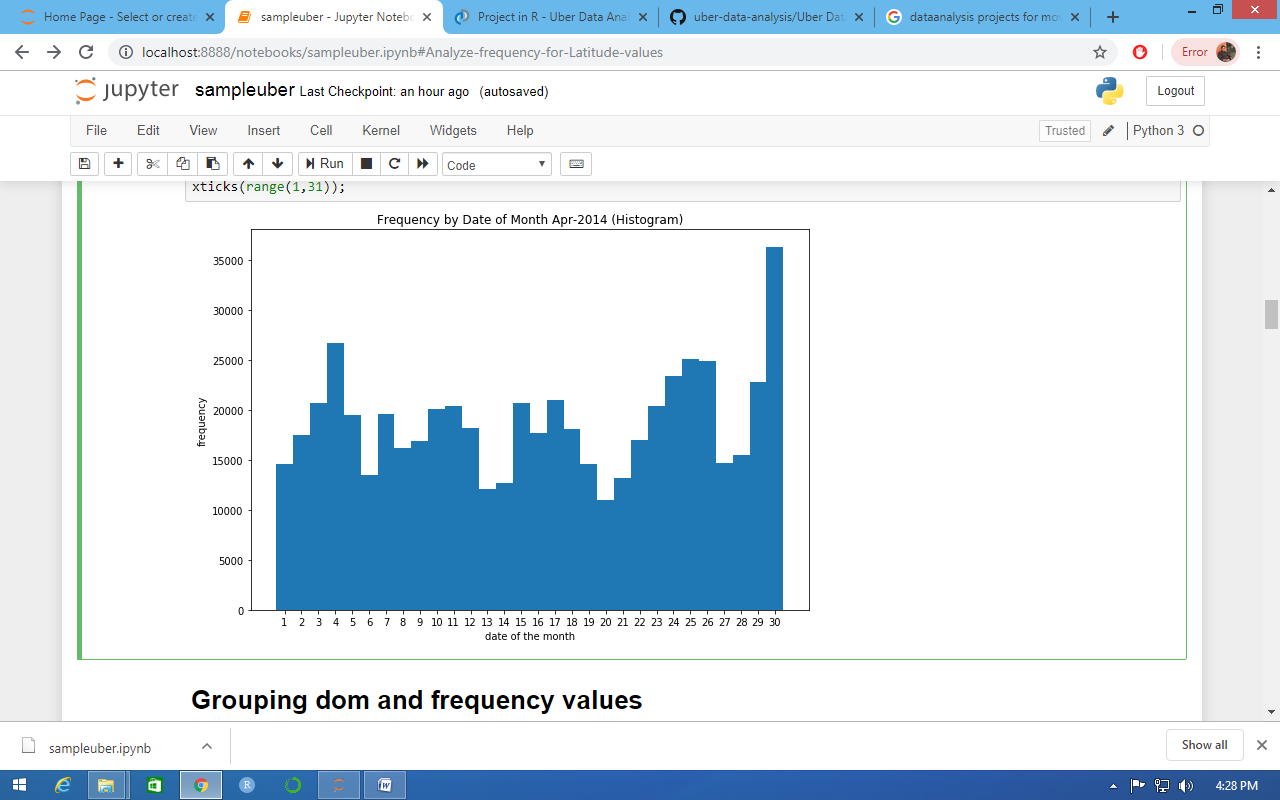
hist(data.dom, bins=30, range=(0.5, 30.5))

xlabel('date of the month')

ylabel('frequency')

title('Frequency by Date of Month Apr-2014 (Histogram)')

xticks(range(1,31));



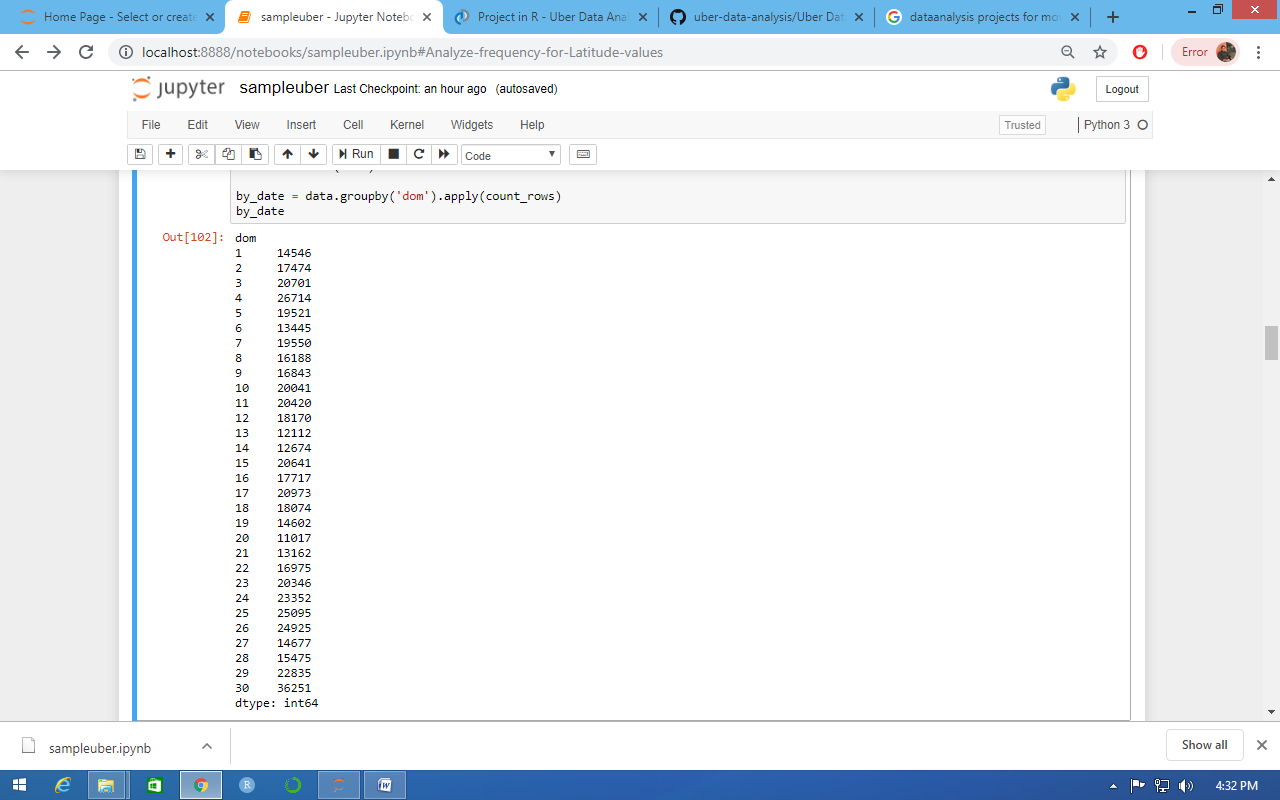
**Grouping dom and frequency values**

def count\_rows(rows):

return len(rows)

by\_date = data.groupby('dom').apply(count\_rows)

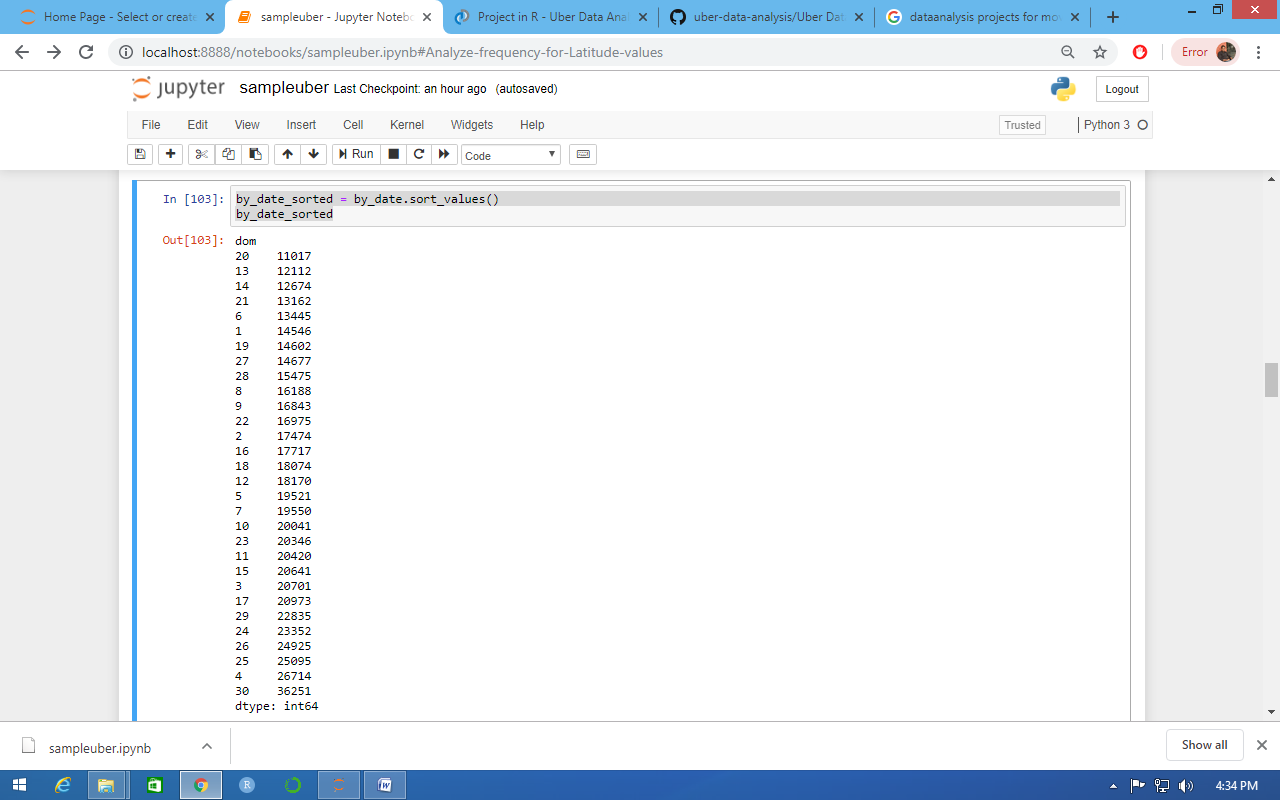
by\_date



**Sorting the grouped values**

by\_date\_sorted = by\_date.sort\_values()

by\_date\_sorted



**Analyze the frequency for sorted values**

figure(figsize=(10, 7))

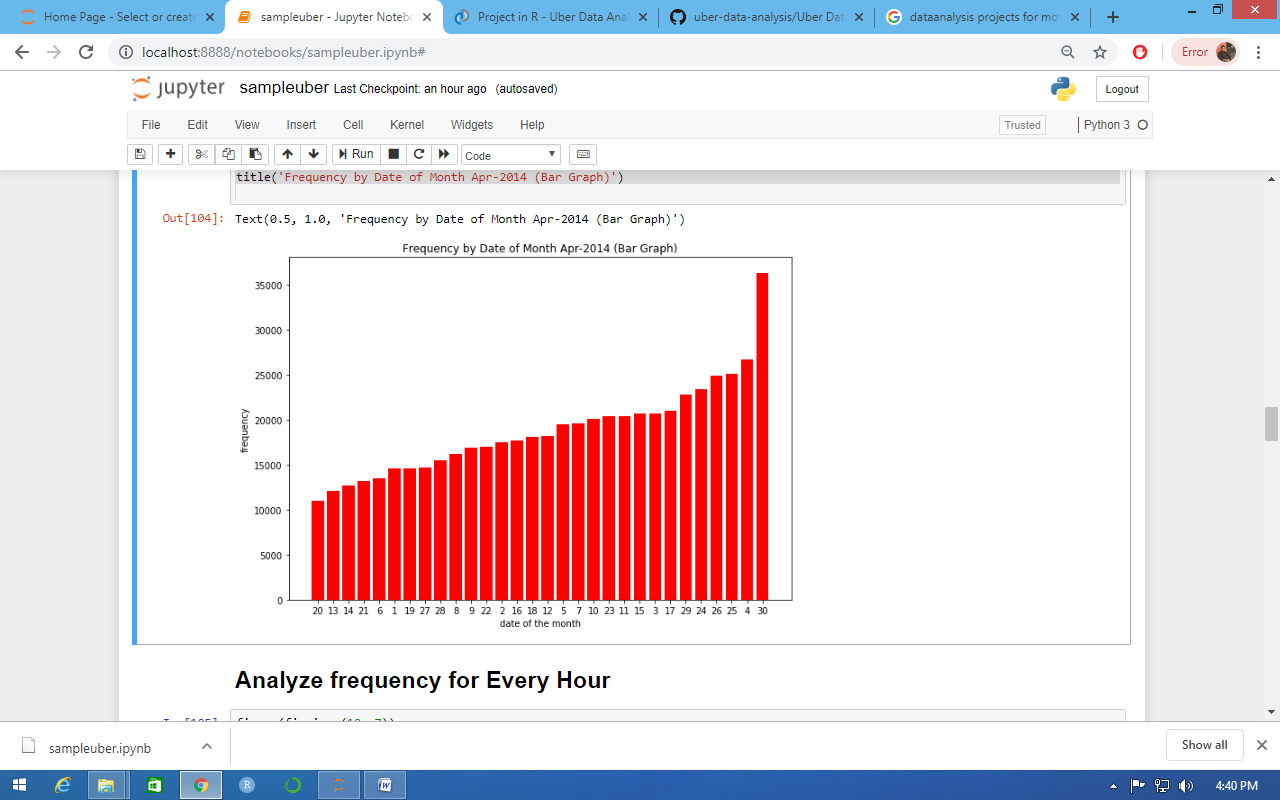
bar(range(1, 31), by\_date\_sorted, color='red')

xticks(range(1,31), by\_date\_sorted.index)

xlabel('date of the month')

ylabel('frequency')

title('Frequency by Date of Month Apr-2014 (Bar Graph)')



**Analyze frequency for Every Hour**

figure(figsize=(10, 7))

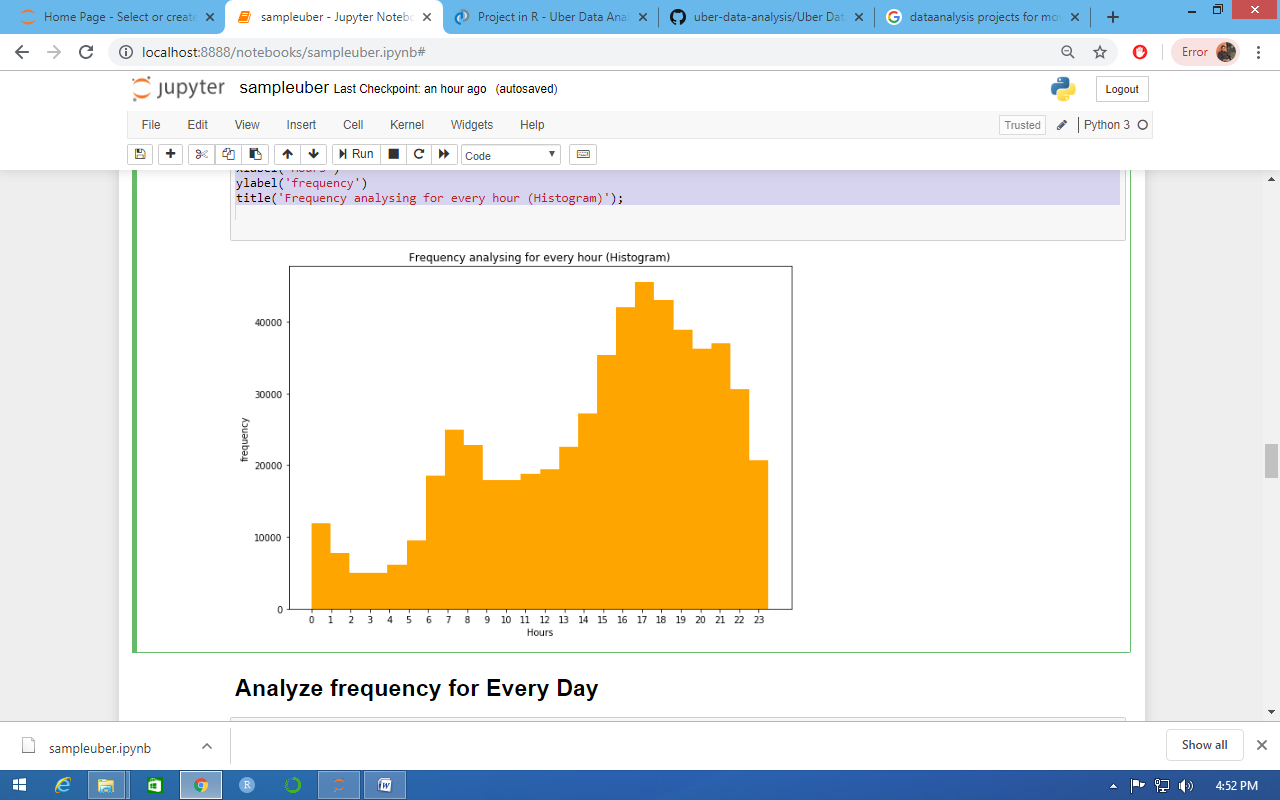
hist(data.hour, bins=24, range=(0,23.5), color='orange')

xticks(range(0,24))

xlabel('Hours')

ylabel('frequency')

title('Frequency analysing for every hour (Histogram)');



**Analyze frequency for Every Day**

figure(figsize=(10, 7))

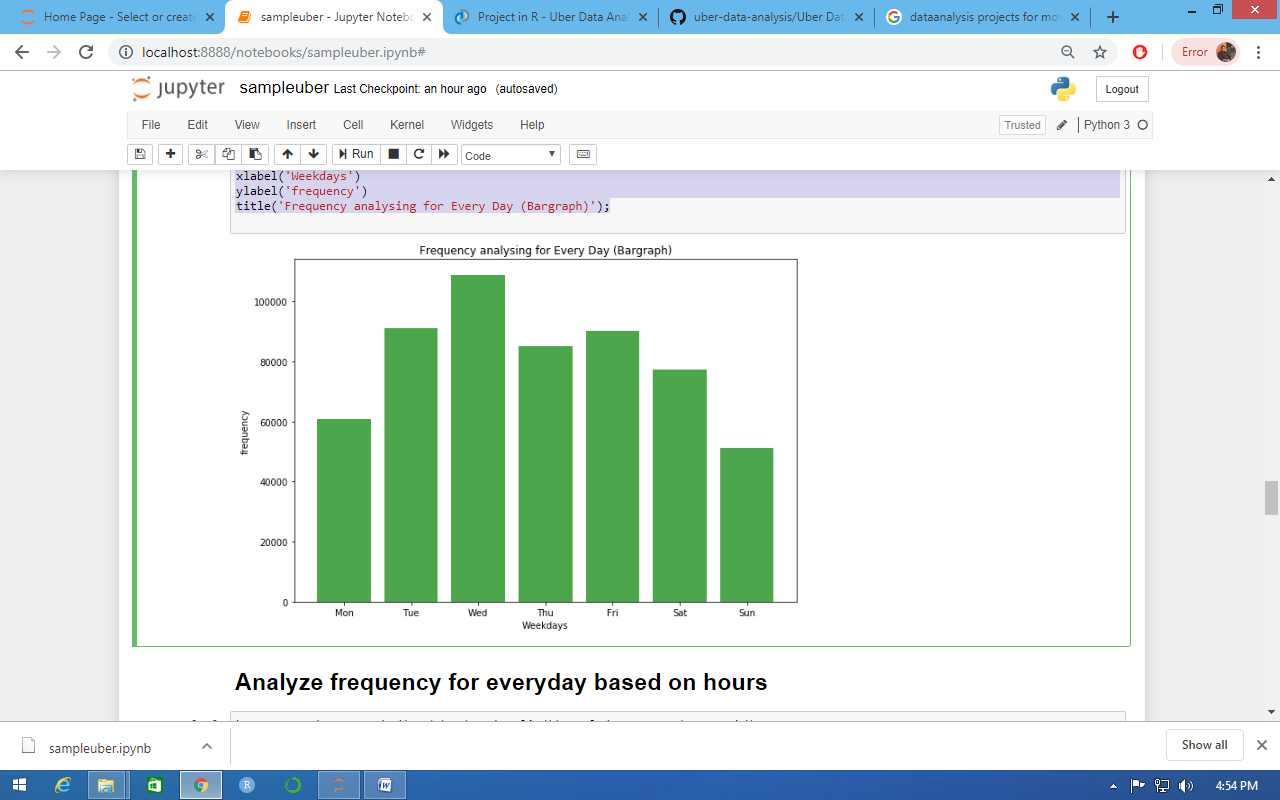
hist(data.weekday, bins=7, range =(-.5,6.5), rwidth=.8, color='green', alpha=.7)

xticks(range(7), 'Mon Tue Wed Thu Fri Sat Sun'.split())

xlabel('Weekdays')

ylabel('frequency')

title('Frequency analysing for Every Day (Bargraph)');



**Analyze frequency for everyday based on hours**

by\_cross = data.groupby('weekday hour'.split()).apply(count\_rows).unstack()

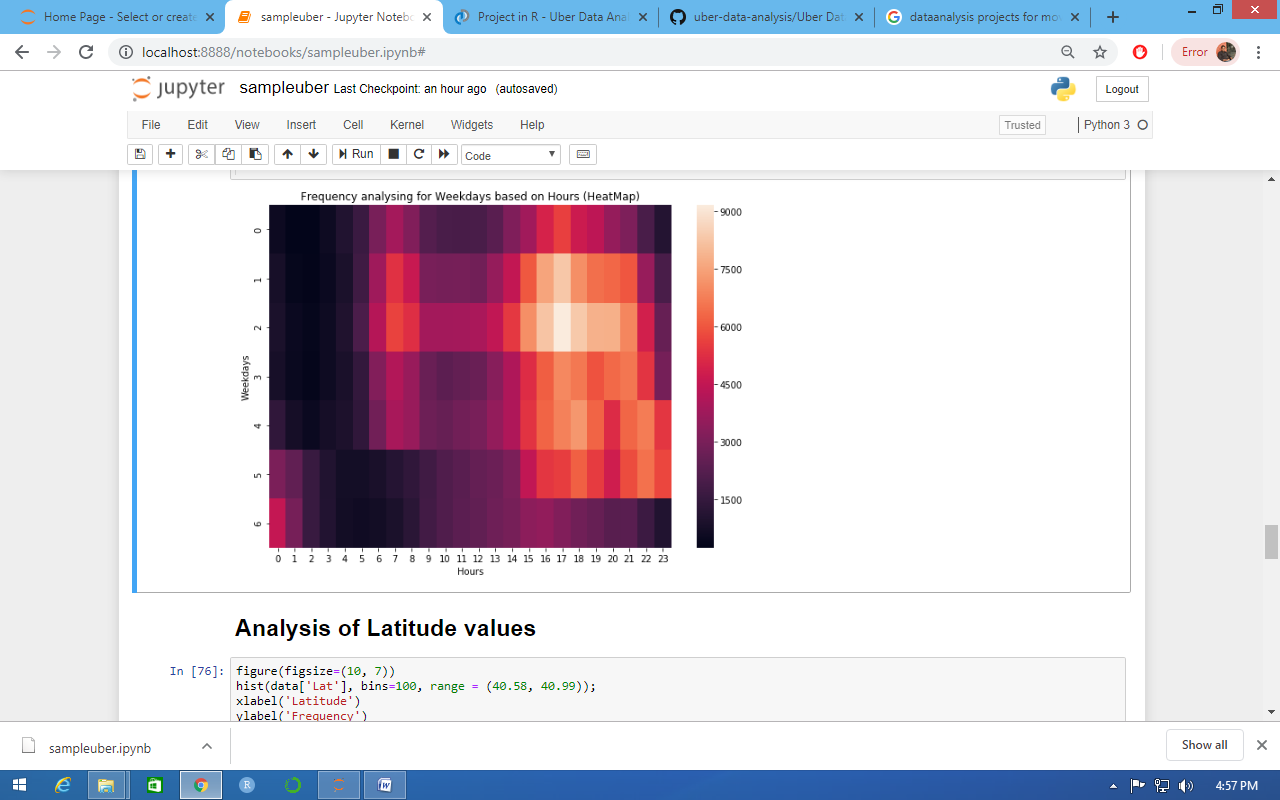
figure(figsize=(10, 7))

seaborn.heatmap(by\_cross)

xlabel('Hours')

ylabel('Weekdays')

title('Frequency analysing for Weekdays based on Hours (HeatMap)');



**Analysis of Latitude values**

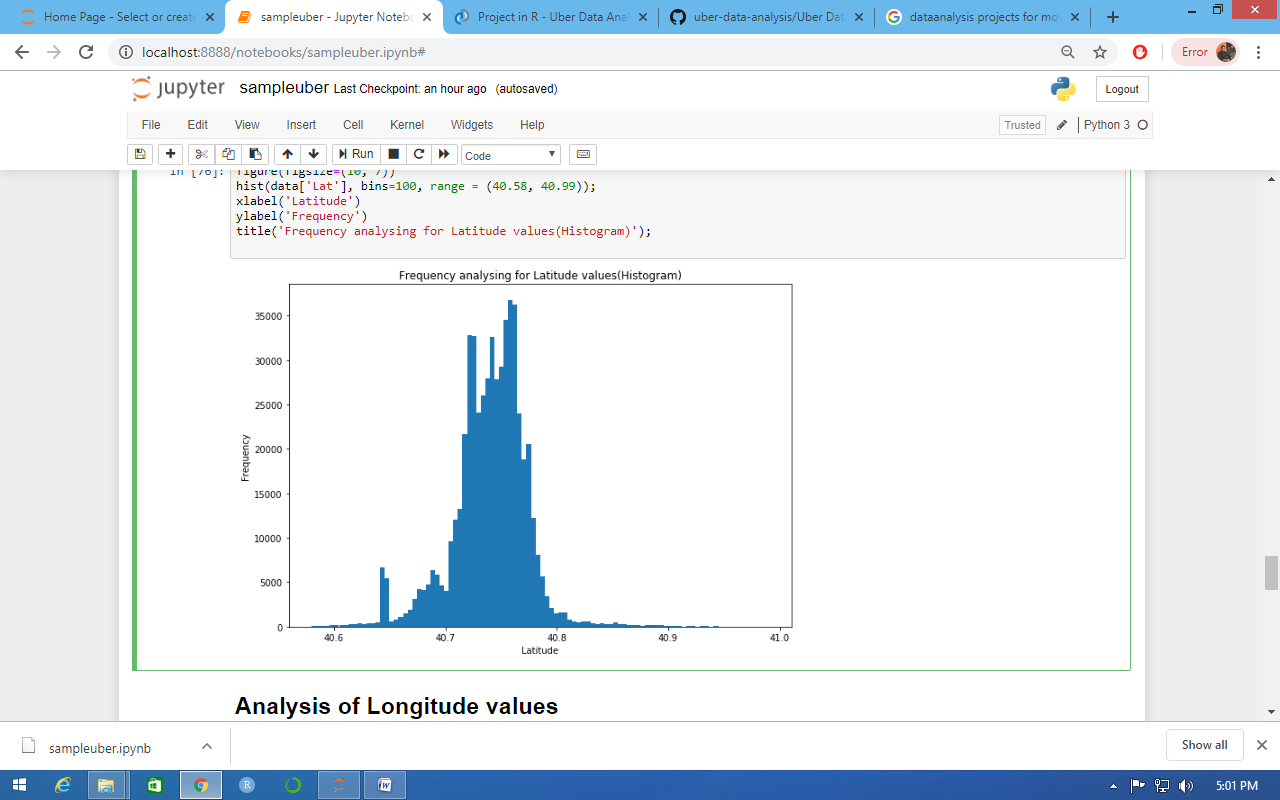
figure(figsize=(10, 7))

hist(data['Lat'], bins=100, range = (40.58, 40.99));

xlabel('Latitude')

ylabel('Frequency')

title('Frequency analysing for Latitude values(Histogram)');



**Analysis of Longitude values**

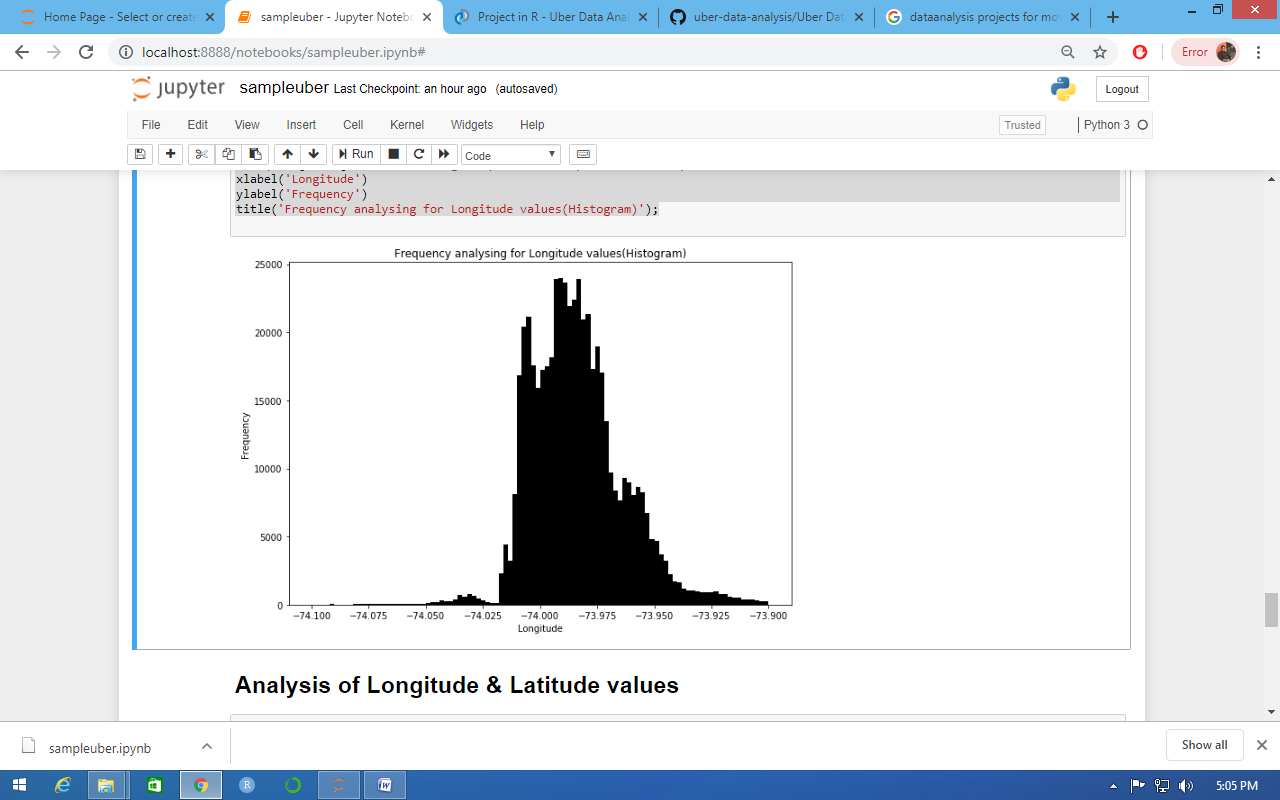
figure(figsize=(10, 7))

hist(data['Lon'], bins=100, range = (-74.1, -73.9),color='black')

xlabel('Longitude')

ylabel('Frequency')

title('Frequency analysing for Longitude values(Histogram)');



**Analysis of Longitude & Latitude values**

figure(figsize=(10, 7))

hist(data['Lon'], bins=100, range = (-74.1, -73.9), color='g', alpha=.5, label = 'longitude')

grid()

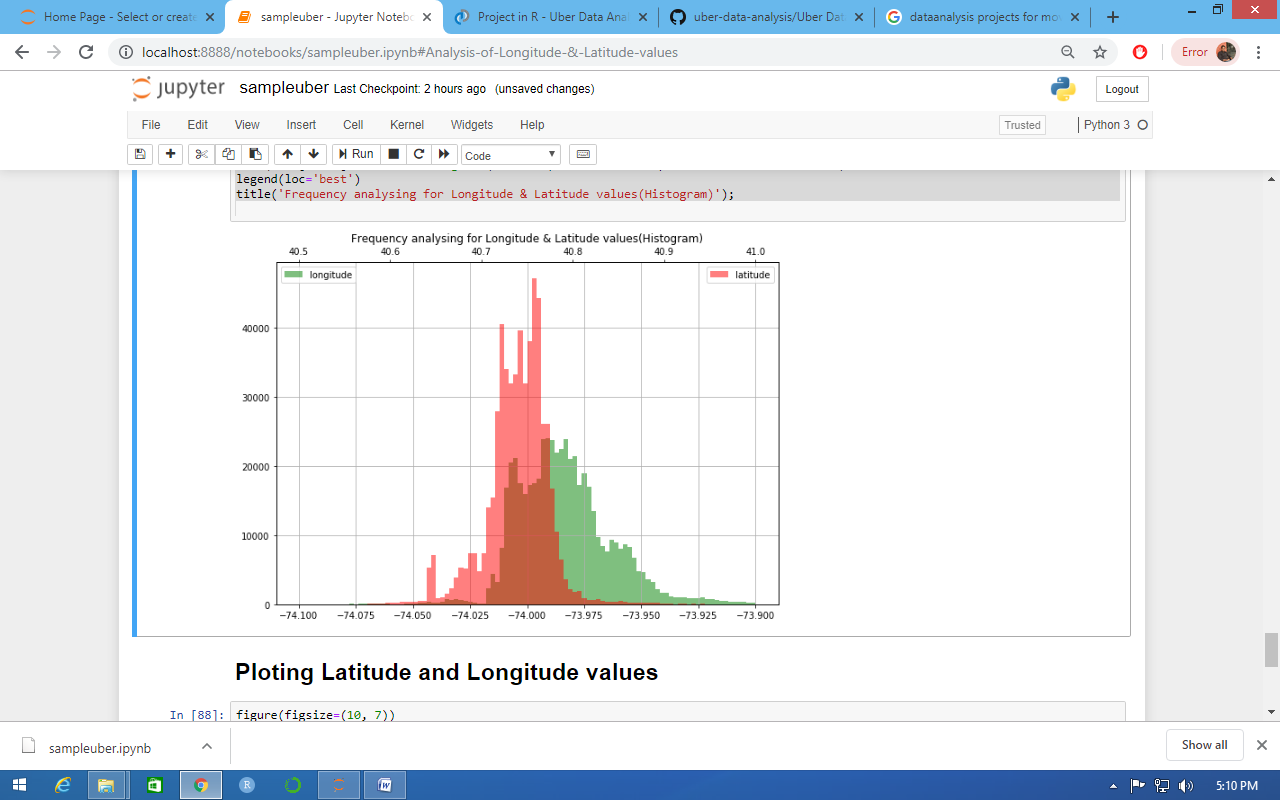
legend(loc='upper left')

twiny()

hist(data['Lat'], bins=100, range = (40.5, 41), color='r', alpha=.5, label = 'latitude')

legend(loc='best')

title('Frequency analysing for Longitude & Latitude values(Histogram)');



**Ploting Latitude and Longitude values**

figure(figsize=(10, 7))

plot(data['Lon'], data['Lat'], '.', ms=1, alpha=.8, color='orange')

xlim(-74.2, -73.7)

ylim(40.7, 41)

title('Ploting Latitude and Longitude values');

