Analyzing NYC Subway Dataset

Intro to Data Science - Project 1
Emmanuelle JEAN
emmanuelle.jean@gmail.com

Section 0. References

references :
http://pandas.pydata.org/pandas-docs/stable/
www.udacity.com/wiki/cs101/unit1-python-reference
http://www.cyberciti.biz/faq/python-convert-string-to-int-functions/
http://docs.python.org/2/library/datetime.html#datetime.datetime.strptime
http://www.tutorialspoint.com/python/time_strptime.htm
https://bespokeblog.wordpress.com/2011/07/11/basic-data-plotting-with-matplotlib-part-3-histograms/
http://docs.scipy.org/doc/numpy/reference/index.html
http://www.creative-wisdom.com/teaching/WBI/parametric_test.shtml
http://www.itl.nist.gov/div898/handbook/pri/section2/pri24.htm
http://blog.yhathq.com/posts/facebook-ggplot-tutorial.html
http://discussions.udacity.com/c/nd002-2015-02-04/project-1

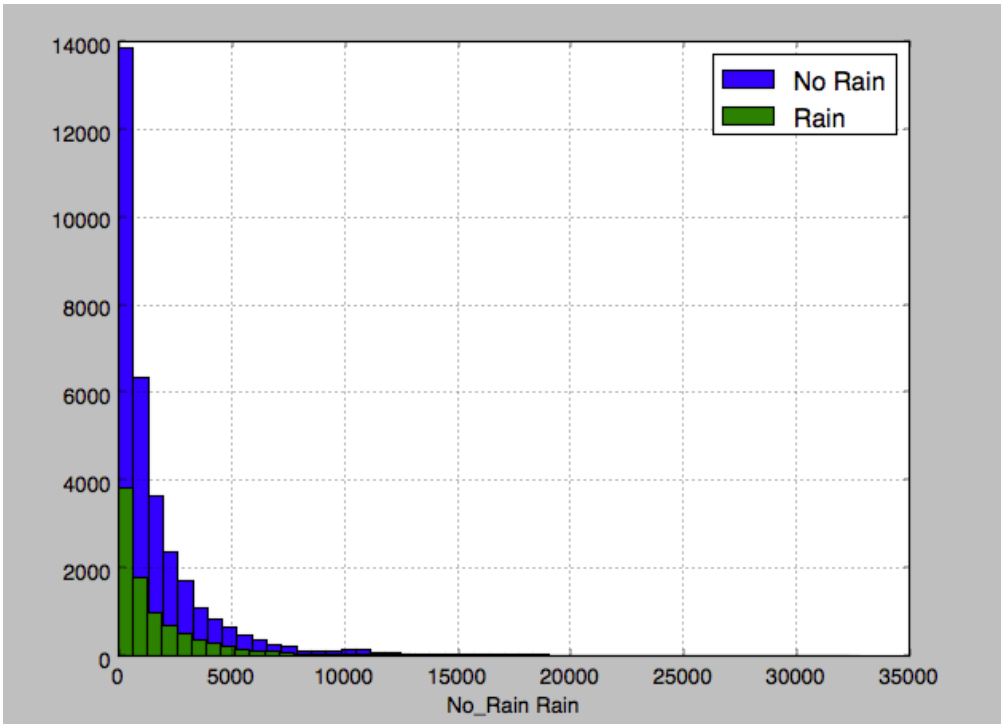
Shapiro Wilk Test :
http://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.shapiro.html
http://unsupervised-learning.com/shapiro-wilk-test-normality/

Section 1. Statistical test

From the first exploratory data analysis done in the problem set 3-1, the histogram shows that the data are not normalized and I concluded that we can not perform the Welch's t test.

Udacity provided an enhanced data source and before going further, I looked at the same histogram than in the problem set 3-1 to see if the histogram of frequencies of ENTRIESn_hours have the same shape than the one for the 1st data set.

```
In [*]: import numpy as np
        import pandas
        from ggplot import *
        import matplotlib.pyplot as plt


        turnstile_data = pandas.read_csv('turnstile_weather_v2.csv')


        def entries_histogram(turnstile_weather):

            fig = plt.figure()
            plt.xlabel("ENTRIESn_hourly")
            plt.ylabel("frequency")
             plt.title("frequency ofENTRIESn_hourly histogram")
            turnstile_weather['ENTRIESn_hourly'][turnstile_weather['rain']==0].hist(bins=50, label="No Rain"
            turnstile_weather['ENTRIESn_hourly'][turnstile_weather['rain']==1].hist(bins=50, label="Rain")
            plt.legend()

            return plt

        entries_histogram(turnstile_data).show()
```



The resulted histogram has the same shape than the histogram of the original dataset.

1.1 Let's compute the Mann-Whitney U-Test on the improved dataset:

- One-tail p value
- H_0 : the distribution of No Rainy day have the same mean of ENTRIESn_hourly that the distribution of rainy day.
- p critical value of 0.05

1.2 This Mann-Whitney U-Test is applicable because we do not need to make any assumption regarding the distribution of the population and we have more than 20 values in each samples.

1.3 The Mann-Whitney U-Test results are :

```
import numpy as np
import scipy
import scipy.stats
import pandas

def mann_whitney_plus_means(turnstile_weather):

    print np.percentile(turnstile_weather['ENTRIESn_hourly'][turnstile_weather['rain']==0],50)
    print np.percentile(turnstile_weather['ENTRIESn_hourly'][turnstile_weather['rain']==1],50)

    without_rain_mean = np.mean(turnstile_weather['ENTRIESn_hourly'][turnstile_weather['rain']==0])
    with_rain_mean = np.mean(turnstile_weather['ENTRIESn_hourly'][turnstile_weather['rain']==1])
    (U,p) = scipy.stats.mannwhitneyu(turnstile_weather['ENTRIESn_hourly'][turnstile_weather['rain']==0],
                                     turnstile_weather['ENTRIESn_hourly'][turnstile_weather['rain']==1])
    return with_rain_mean, without_rain_mean, U, p # leave this line for the grader

turnstile_data = pandas.read_csv('turnstile_weather_v2.csv')
mann_whitney_plus_means(turnstile_data)
```

```
893.0
939.0
(2028.1960354720918, 1845.5394386644084, 153635120.5, 2.7410695712437496e-06)
```

Mann-Whitney U-Test	**U = 153635120.5**	**P = 5.482e-06**
	Non Rainy Day	*Rainy day*
Mean ENTRIESn_hourly	1845.539	2028.196
Median ENTRIESn_hourly	893.0	939.0

1.4 The p value is smaller than the p critical value so we can reject the null hypothesis and say that the non rainy day distribution is most likely to have a different mean of ENTRIESn_hourly than the rain day distribution.
This conclusion is confirmed by the comparison of both the mean and the median of the ENTRIESn_hourly for our non rainy day and for rainy day samples.

Section 2. Linear Regression

Dataset : turnstile_weather_v2.csv

The code I used to compute the linear gradient, R^2 and the graph of residuals is in the file in the annexe “Intro To Data Science Problem 3-5.html”.

2.1 I used the Gradient descent approach to compute the coefficients theta and produce prediction for ENTRIESn_hourly in my regression model.

2.2 The features I use in my model are : hour, day_week, fog, precipi, pressurei, rain, tempi, wspdi, meanprecipi, meanpressurei, meantempi and meanwspdi .
I used the dummy variables related to UNIT.

2.3 My reasoning was to use all the numerical features hour, day_week, fog, precipi, pressurei, rain, tempi, wspdi, meanprecipi, meanpressurei, meantempi. The more features, the more precise my model will be and the R^2 will be close to one. I chose the ones that may have an impact on the decision of rider to take or not the NYC subway. I based largely these choices on examples I lived through while living in NYC and being a subway commuter.

- Hour : the time of the day is important for the rider as for example, during night time, the frequency of the subway is very low and therefore the rider would prefer to take a cab instead of waiting half an hour a subway.
- Day of the week : affects the decision of the rider in a similar way than the hour on the decision to take or not the subway. For example a rider might decide to walk few blocks and not take the subway at the end of the working day because he does not want to ride in a crowded car whereas he might take the subway during the week-end as less people ride it.
- Fog : the presence or not of fog may affect the rider in his decision to not take his car or bike and take the subway to travel, as fog can brings delays on the roads.
- Precipitation is one of the weather elements that the rider will asses in his decision to take or not the subway. Let’s take the example of lunch break when our rider has a lunch schedule few blocks away, if there is no precipitation, the rider can decide to walk those few blocks and not take the subway.
- Barometric pressure can have an impact on the decision as well. While the rider probably does not know the exact value of the pressure, he has a feeling of that pressure and might expect a weather change in the few hours and will decide his mean of transportation accordingly.
- Rain : the presence of rain that day at the location is also an information that the rider takes in consideration in his decision to ride or not the subway.
- Temperature at the time of the location as an impact on the decision of taking or not the subway. One might decide to bike if temperatures are clement enough.

- The wind can be strong in New York and a rider will likely to decide to take the subway instead of walking or riding a bike.
- The average of precipitation during the day can also have an impact on the rider's decision. If he knows that it will rain all day, even if it is not rainy at the time of the departure, he might decide to take his car because he does not want to walk the last few blocks under the rain later when he comes back.
- The average of pressure and the average of temperature are influential on the decision to take the NYC subway is a decision made not only on one way but also, in most of the case how you come back from your journey. The rider does not think only about how he goes to work, but also how he will come back from work. If the temperature have a risk of going to high or too low later in the, he might change his mind and take the subway instead of taking his bike.

I had a R^2 of 0.476.
I reduced the number of features and experimented with different alpha and number of iteration and computed the R^2 value.

features	alpha	iteration	R^2
'hour','day_week','fog','precipi','pressurei','rain','tempi','wspdi','meanprecipi','meanpressurei','meantempi','meanwspdi'	0.1 0.5 0.5 0.1 0.8	75 75 150 1000 75	0.476 0.476 0.476 -1.405
'rain', 'precipi', 'Hour', 'hour','day_week','fog','precipi','pressurei','rain','tempi','wspdi'	0.1 0.8	1000 75	0.471 0.470
'hour','day_week','fog','precipi','rain'	0.1 0.8	1000 75	0.470 0.470
'day_week','rain'	0.8 0.1	75 1000	0.385 0.385

From my different tests, it appears that R^2 decreases a little with less features. I find the the R^2 far from 1 anyway. A higher value of alpha, introduce a risk of missing the lowest point.

2.4 The coefficients for the 12 features are :

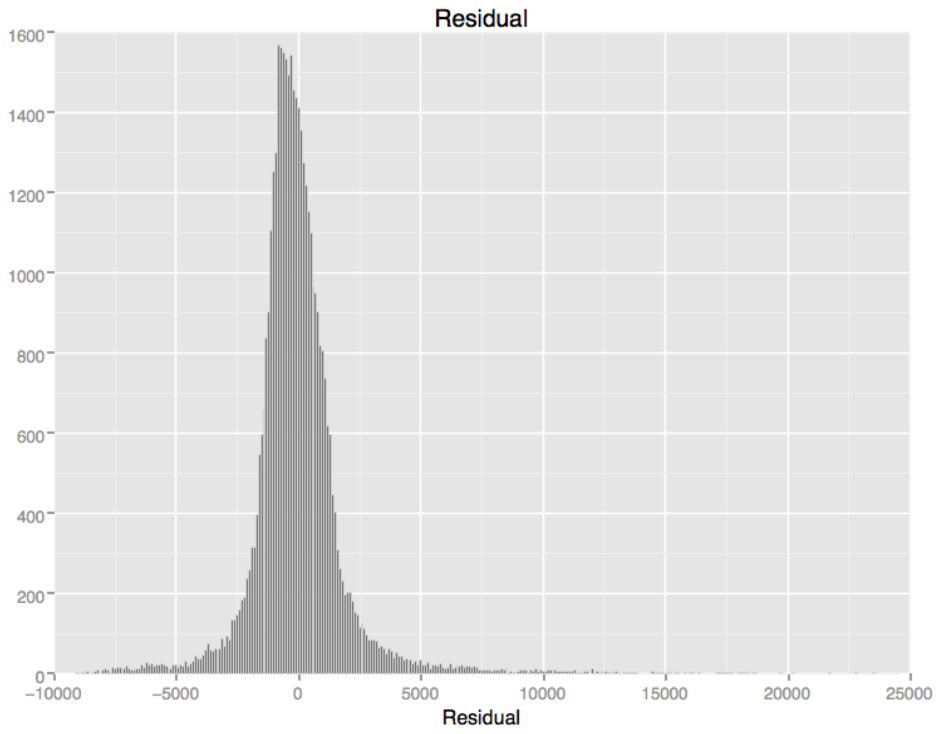
features	coefficients		features	coefficients
hour	7.284e+02		tempi	3.4247e+02
day_week	-3.235e+02		wspdi	6.598e+01

fog	-1.617e+01		meanprecipi	1.030e+02
precipi	-1.314e+02		meanpressuri,	5.188e+01
pressurei	-1.098e+02		meantempi	-3.979e+02
rain	-4.696e+00		meanwspdi	-1.119e+02

The coefficients are the twelve first coefficients of the *theta_gradient_descent* data-frame computed in the *predictions* procedure. They correspond to the first twelve columns of the features dataframe, the rest being the UNIT features using dummies data.

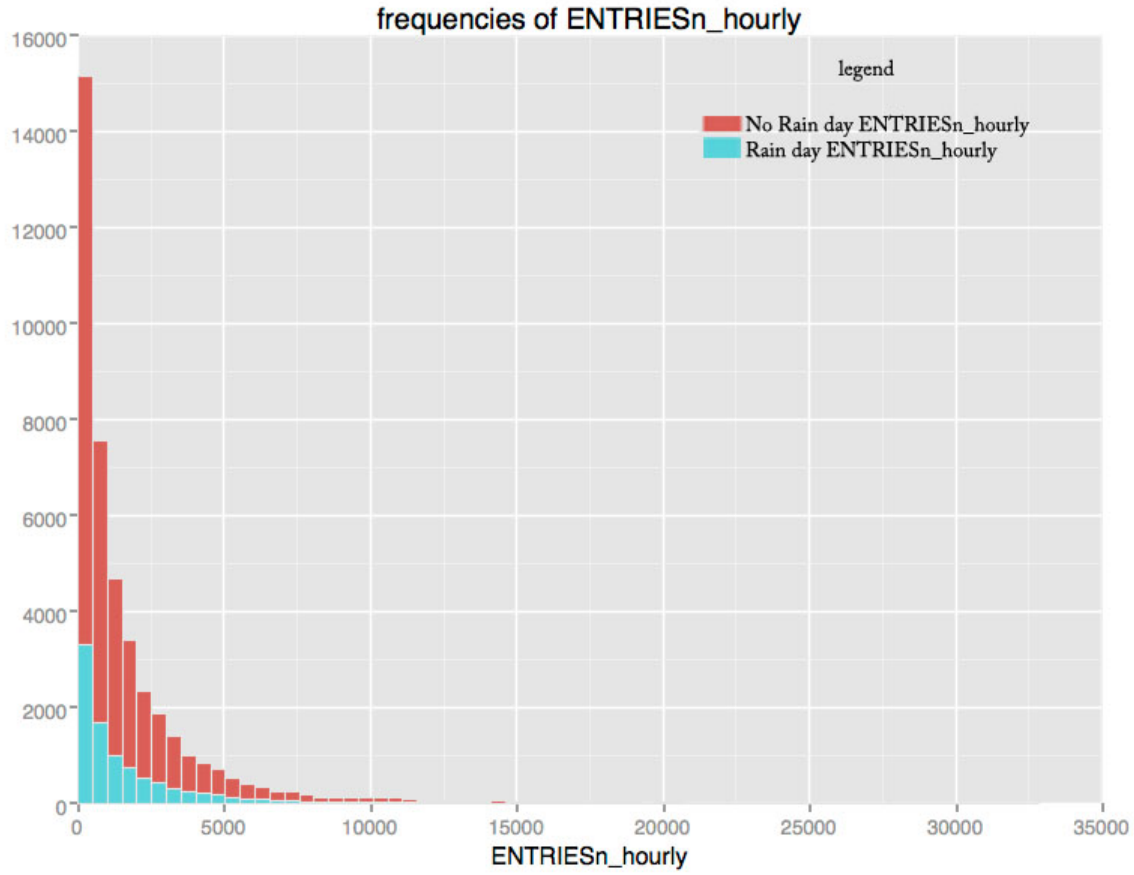
2.5 My model R^2 is 0.476.

2.6 My calculated coefficient of determination is pretty far from one.
But the plot of the residual data is normalized and show a higher frequencies around 0. It indicates that our model is pretty accurate.



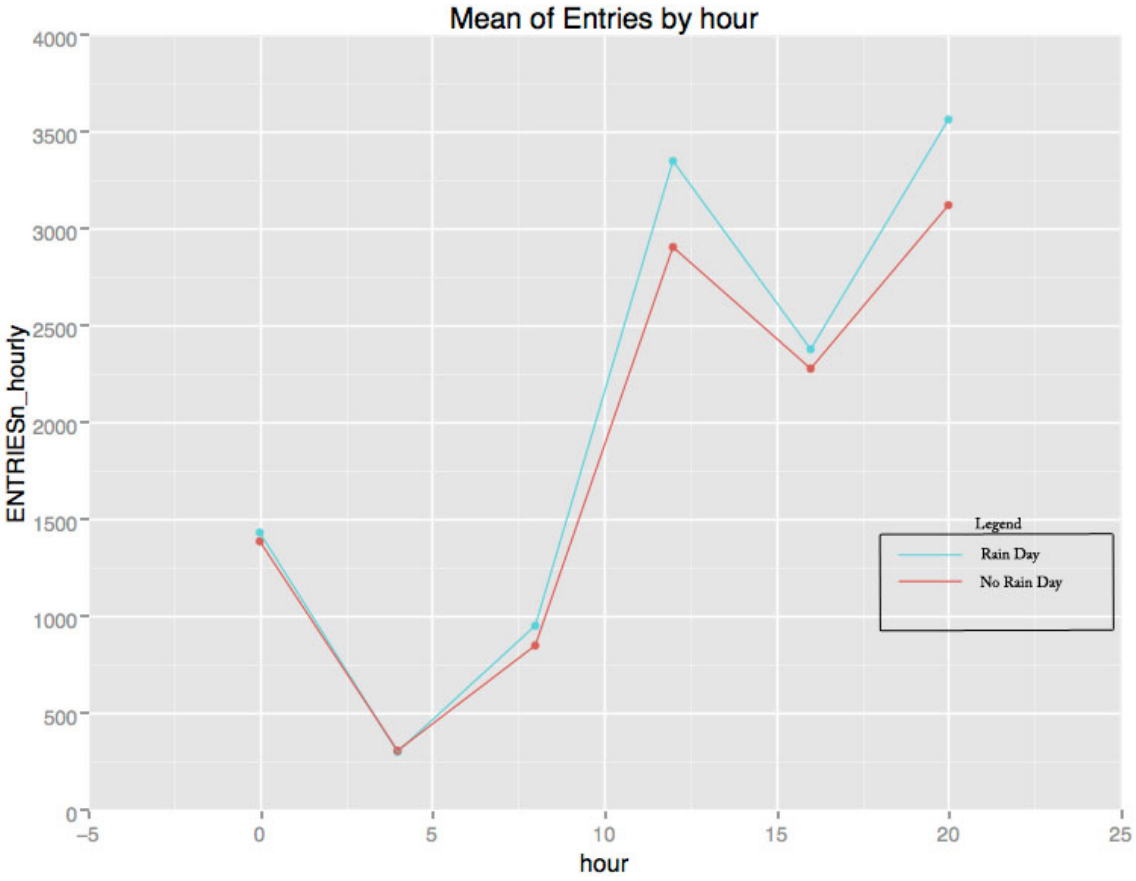
Section 3. Visualization

3.1 Histogram representing the frequencies of ENTRIESn_hourly for rainy days (blue) and non rainy days (red).




```
: from pandas import *
from ggplot import *
from datetime import *


"""Histogram of the frequencies of ENTRIESn_hourly """
def plot_weather_data(turnstile_weather):

    pandas.options.mode.chained_assignment = None

    plot= ggplot( aes(x="ENTRIESn_hourly", fill ="rain" ), data = turnstile_weather ) +\
    geom_histogram(binwidth =500)+\
    ggtitle("frequencies of ENTRIESn_hourly")


    return plot

turnstile_weather = pandas.read_csv('turnstile_weather_v2.csv')
plot_weather_data(turnstile_weather)
```

3.2 Mean of ENTRIESn by hour, for rain and no rain days



Code

```
from pandas import *
from ggplot import *
from datetime import *

def plot_weather_data(turnstile_weather):
    turnstile_weather['DateDatetime'] = pandas.to_datetime(turnstile_weather['DATEn'])
    #turnstile_weather['weekDay'] = turnstile_weather['DateDatetime'].map(lambda x: x.weekday())
    #******Visualisation of the mean of Entries by hour for rain and no rain day for the business day
    rain_day_group = turnstile_weather.groupby(["rain","hour"], as_index=False)
    mean_rain_day= rain_day_group["rain","hour", "ENTRIESn_hourly"].aggregate(np.mean)
    plot= ggplot(mean_rain_day, aes(y='ENTRIESn_hourly', x="hour", color='rain')) + geom_point() +geom_line()+\
    ggtitle("Mean of Entries by hour")
    return plot

turnstile_weather = pandas.read_csv('turnstile_weather_v2.csv')
plot_weather_data(turnstile_weather)
```

On this visualization, we see how the average of entries varies during the day by hour block, for non rainy day (red) and rainy day (blue).

We can see that there is more people, in average, riding the subway when it rains, between 8:00 am and 8:00 pm. The numbers of riders entering the subway between 12:00 am and 4:00 am does not seems as much influenced by rain than riders using the subway after 8:00 am.

Section 4. Conclusion

4.1 From my analysis of the NYC subway improved dataset and interpretation of the computed information, more people ride the NYC subway when it rains.

4.2
First, I compared the average of riders during rainy days to the average of riders during days without rain. There are more people who take the subway on rainy days. I compared, as well, the median of the entries during rainy days to the median of the entries of non rainy days. The comparison led to the same conclusion.

But I needed to make sure that this conclusion is representative for every months of the year. I had to verify that the data is representative to the whole population of entries in the NYC Subway.
I looked at the histogram of the two sets to see if they are normally distributed. The histogram did not show a Gaussian curve. The only test I could use was the Mann-Whitney U-Test. The Mann-Whitney U-Test does not need the assumption that the data follows a normal distribution. From the calculated Mann-Whitney U and p values, I interpreted that indeed, the sets of data differs and we have a difference of average of number of NYC Subway riders when it rains.

We have more than 5000 data from which we computed our statistic, we can trust our result. I chose the linear gradient as the prediction model, and use 12 features to compute the prediction values. Even if the R^2 value of 0.476 is far from one. The residual data plot is normal and indicates that my prediction model is accurate.

Section 5. Reflection

5.1 shortcoming of the method of my analysis :
Dataset :

- The number of datapoint for rainy days is low compared to to the number of data points for non rainy days. And knowing that the more data we have the more we can trust in our statistics. I am wondering if the difference of number of data entries can have an impact in our statistics.

- I could not help but notice the differences of the Mann-Whitney U-Test calculated on the first dataset and the second one. I understood that you kept only the data taken in

the same time bracket. In the classroom problem set, I got stuck because the p value calculated was very close to the critical value, almost equal. And I was not sure of my conclusion. Whereas with the second dataset the p value was very far from the critical value. It was easier to conclude.

- I have been a NYC subway commuter and my first reflexion was that ridership is not influenced by rain. One takes the NYC subway because it is the most convenient compared to the bus, the car, bike or feet. The analysis proved me that I was wrong in my way of thinking.

Analysis :

- In my linear regression, I did not included any non numerical features such as UNIT or station or even location. I took as much features as possible to keep my R^2 as big as I can. I have been able to provide an good reasoning of why to choose the features I chose but I am wondering if it would be better to chose less, even if the R^2 is smaller. What is the best practice in the statistical world?

- I only used the methods presented in the Intro to Data Science but I am aware that more methods are available.