NYC Transportation Domain

KAVYA RAO B

- **Data Set** : NYC Traffic Collisions
- **Problem Statement:** Analyze NYC Collision(Accident) data set using Pandas and Numpy
  1. Clean data , remove Null values
  2. Drop the unwanted columns
  3. Divide the data set based on the year
  4. Analyze fatalities based on Borough
  5. Analyze fatalities based on Month
  6. Analyze fatalities based on Time (24 Hours)
  7. Plot graphs.
  8. Apply Polynomial Regression to the Accident data set
- Apply Polynomial regression to predict Person Injuries

Data SetUp

Read CSV File.
Remove unwanted columns and rename them.
Drop Null values

In [100]:
```
import pandas as pd

data = pd.read_csv("NYPD_Motor_Vehicle_Collisions.csv" ,parse_dates=[['DATE', 'TIME']])
#Drop non-continuous variables
data.drop(["VEHICLE TYPE CODE 5", "VEHICLE TYPE CODE 4", "VEHICLE TYPE CODE 3" ,"CONTRIBUTING FACTOR VEHICLE 5",
           "CONTRIBUTING FACTOR VEHICLE 4","CONTRIBUTING FACTOR VEHICLE 3", "ON STREET NAME","CROSS STREET NAME",
           "OFF STREET NAME", "CONTRIBUTING FACTOR VEHICLE 1", "CONTRIBUTING FACTOR VEHICLE 2", "VEHICLE TYPE CODE 1",
           "VEHICLE TYPE CODE 2"], axis = 1, inplace = True)
data= data.dropna()
cols = data.columns
cols = cols.map(lambda x: x.replace(' ', '_') if isinstance(x, (str, unicode)) else x)
data.columns = cols
data
```

Divide data by Year

```
In [3]: data_2012 = data[data.DATE_TIME.dt.year==2012]

        data_2013 = data[data.DATE_TIME.dt.year==2013]

        data_2014 = data[data.DATE_TIME.dt.year==2014]

        data_2015 = data[data.DATE_TIME.dt.year==2015]
```

For each year Find fatalities based on – BOROUGH, MONTH, TIME

```
In [ ]: data_2012_borough = data_2012.groupby(data.BOROUGH).sum().sort(['NUMBER_OF_PERSONS_KILLED','NUMBER_OF_PEDESTRIANS_INJURED'])

        data_2013_borough = data_2013.groupby(data.BOROUGH).sum().sort(['NUMBER_OF_PERSONS_KILLED','NUMBER_OF_PEDESTRIANS_INJURED'])

        data_2014_borough = data_2014.groupby(data.BOROUGH).sum().sort(['NUMBER_OF_PERSONS_KILLED','NUMBER_OF_PEDESTRIANS_INJURED'])

        data_2015_borough = data_2015.groupby(data.BOROUGH).sum().sort(['NUMBER_OF_PERSONS_KILLED','NUMBER_OF_PEDESTRIANS_INJURED'])
```

```
In [11]: data_2015_borough
```

Out[11]:

	ZIP_CODE	LATITUDE	LONGITUDE	NUMBER_OF_PERSONS_INJURED	NUMBER_OF_PERSONS_KILLED	NUMBER_OF_
BOROUGH						
STATEN ISLAND	58615046	230809.497599	-421558.138122	1412	11	209
MANHATTAN	387638357	1576351.058076	-2860539.275847	5895	20	2037
BRONX	201480011	786688.286172	-1422826.756528	5099	25	1236
QUEENS	438513159	1571316.884946	-2849057.384436	9424	45	1742
BROOKLYN	518460182	1878887.958159	-3417547.823545	12408	56	2671

Data by Month

```
In [23]: data_2012_month = data_2012.groupby(data_2012.DATE_TIME.dt.month).sum()[['NUMBER_OF_PERSONS_INJURED','NUMBER_OF_PERSONS_KILLED']]

         data_2013_month = data_2013.groupby(data_2013.DATE_TIME.dt.month).sum()[['NUMBER_OF_PERSONS_INJURED','NUMBER_OF_PERSONS_KILLED']]

         data_2014_month = data_2014.groupby(data_2014.DATE_TIME.dt.month).sum()[['NUMBER_OF_PERSONS_INJURED','NUMBER_OF_PERSONS_KILLED']]

         data_2015_month = data_2015.groupby(data_2015.DATE_TIME.dt.month).sum()[['NUMBER_OF_PERSONS_INJURED','NUMBER_OF_PERSONS_KILLED']]
```

```
In [28]: data_2014_month
```

Out[28]:

	NUMBER_OF_PERSONS_INJURED	NUMBER_OF_PERSONS_KILLED
DATE_TIME		
1	2987	17
2	2325	8
3	2901	13
4	3068	18
5	3399	18
6	3769	16
7	3458	23
8	3353	12
9	3299	19
10	3652	12
11	3248	16
12	3170	11

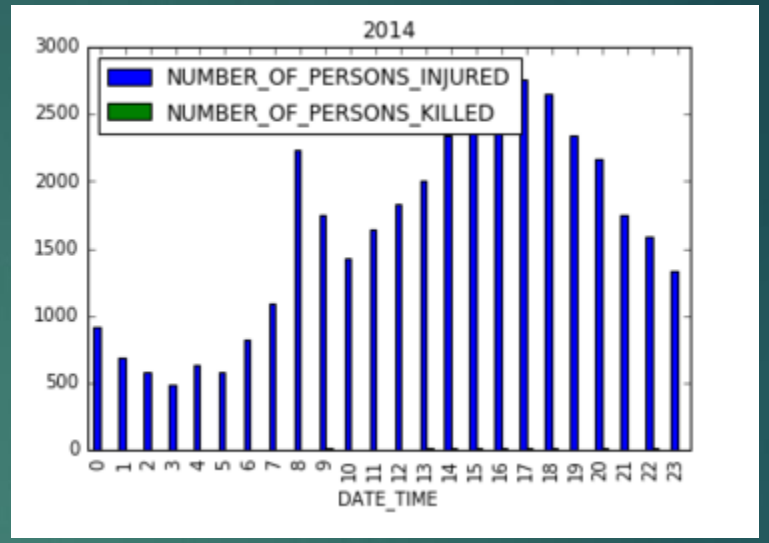
Analysis : During the colder months people probably take public transport and avoid walking/cycling.

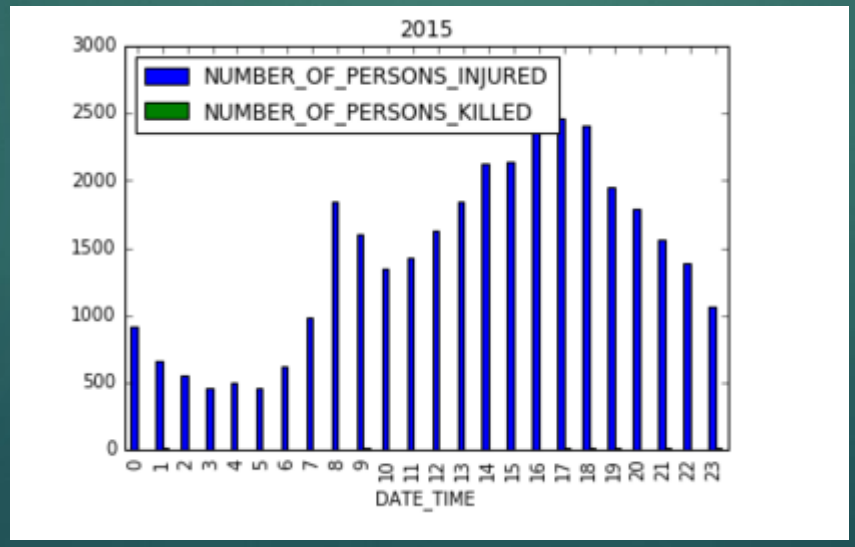
Data by Hour

```
In [26]: data_2012_hour = data_2012.groupby(data_2012.DATE_TIME.dt.hour).sum()[['NUMBER_OF_PERSONS_INJURED','NUMBER_OF_PERSONS_KILLED']]

         data_2013_hour = data_2013.groupby(data_2013.DATE_TIME.dt.hour).sum()[['NUMBER_OF_PERSONS_INJURED','NUMBER_OF_PERSONS_KILLED']]

         data_2014_hour = data_2014.groupby(data_2014.DATE_TIME.dt.hour).sum()[['NUMBER_OF_PERSONS_INJURED','NUMBER_OF_PERSONS_KILLED']]

         data_2015_hour = data_2015.groupby(data_2015.DATE_TIME.dt.hour).sum()[['NUMBER_OF_PERSONS_INJURED','NUMBER_OF_PERSONS_KILLED']]
```

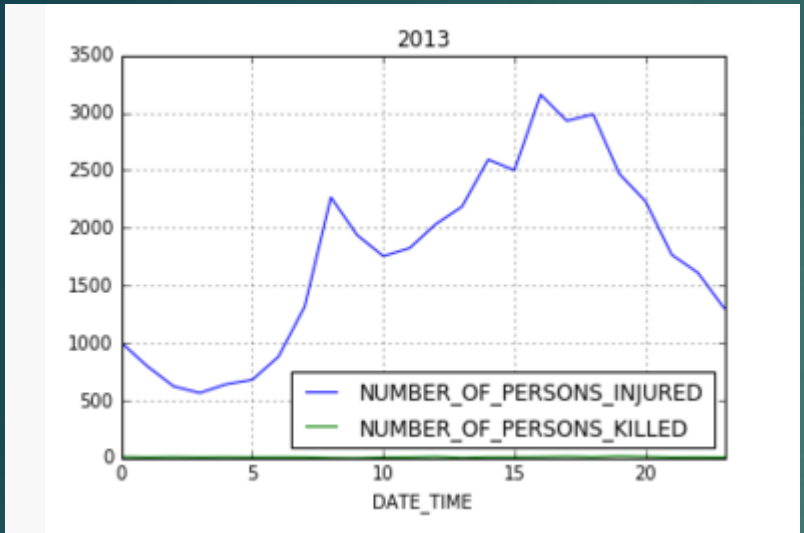
	NUMBER_OF_PERSONS_INJURED	NUMBER_OF_PERSONS_KILLED
DATE_TIME		
0	919	6
1	667	10
2	554	1
3	463	3
4	501	5
5	457	5
6	614	1
7	984	3
8	1848	4
9	1597	10
10	1341	4
11	1428	4
12	1622	5
13	1845	4
14	2126	5
15	2144	5
16	2502	7
17	2465	17
18	2410	11
19	1954	15
20	1788	6
21	1567	9
22	1383	8
23	1059	9

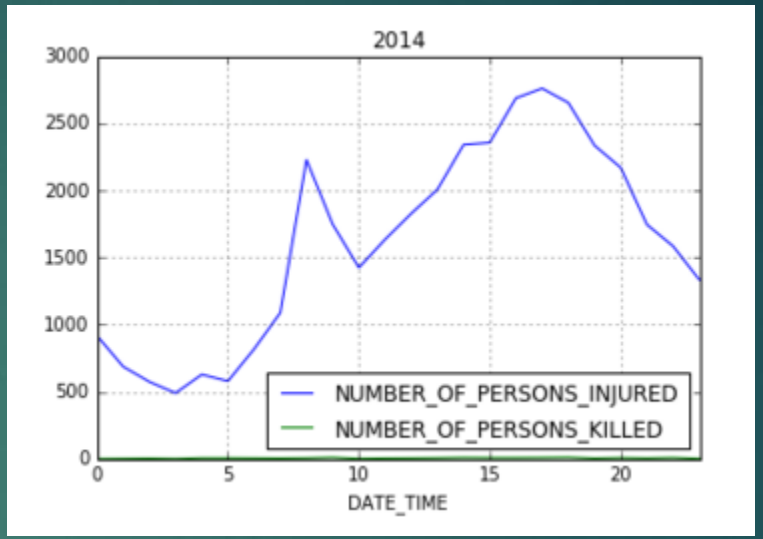
Graph of Number_of_persons_injured for 3 years

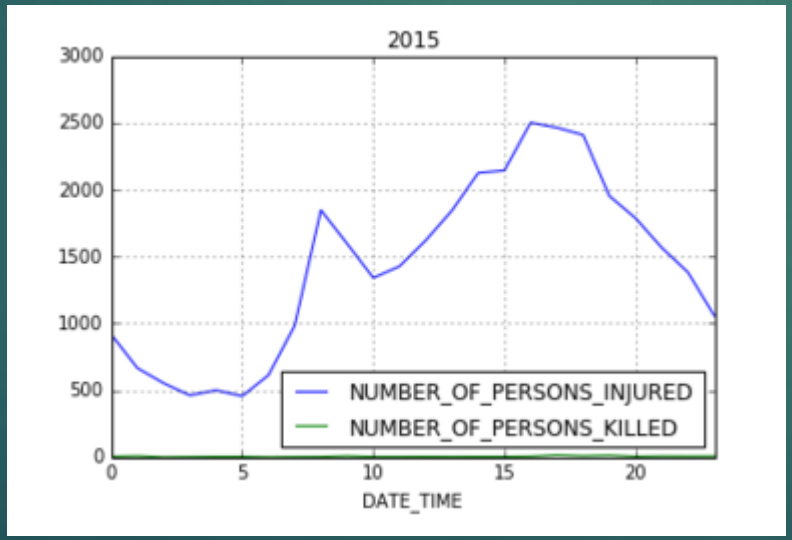




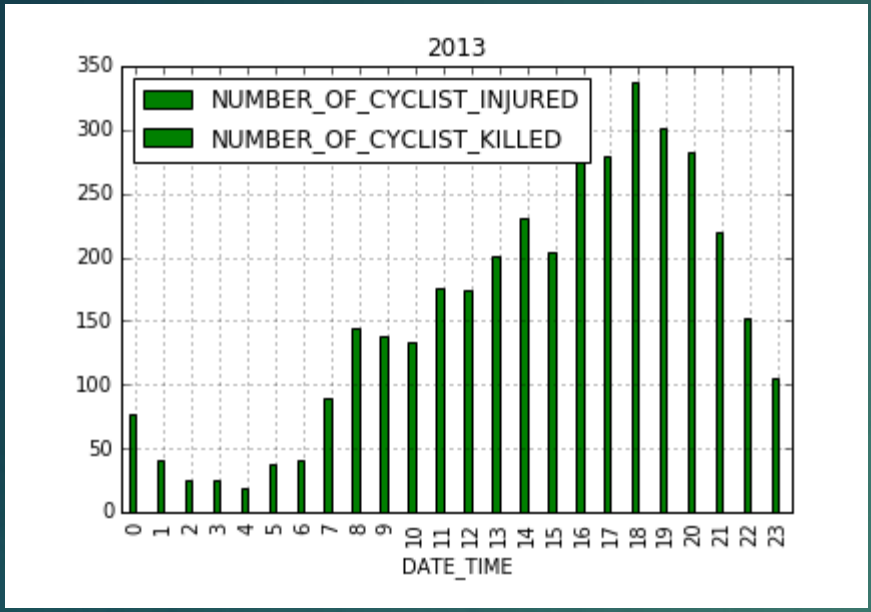
Graph of Number_of_persons_injured for 3 years

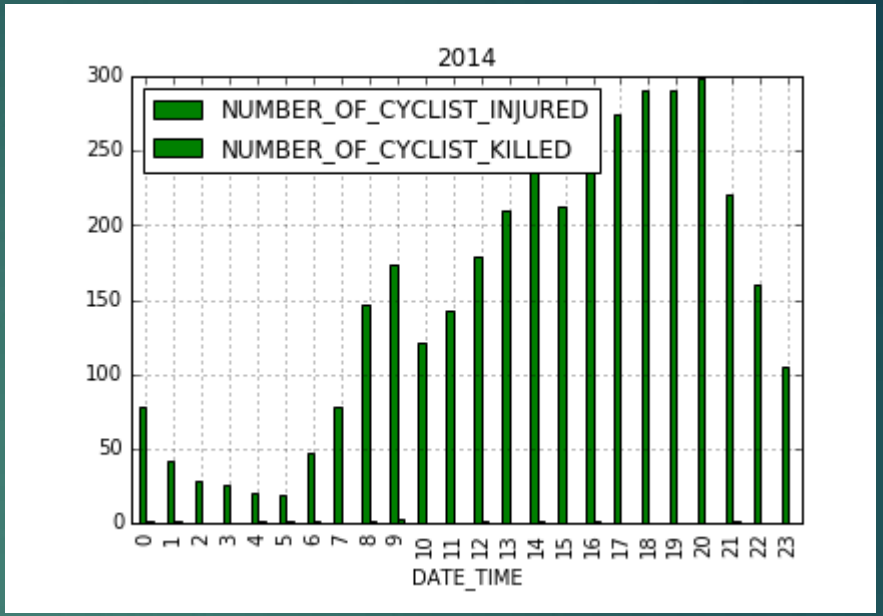


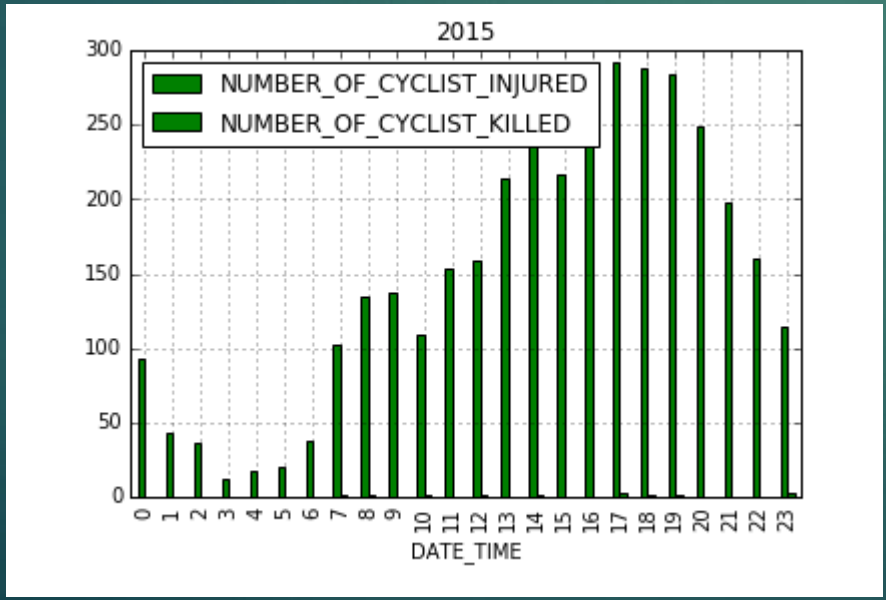




Number of Cyclists Injured each hour







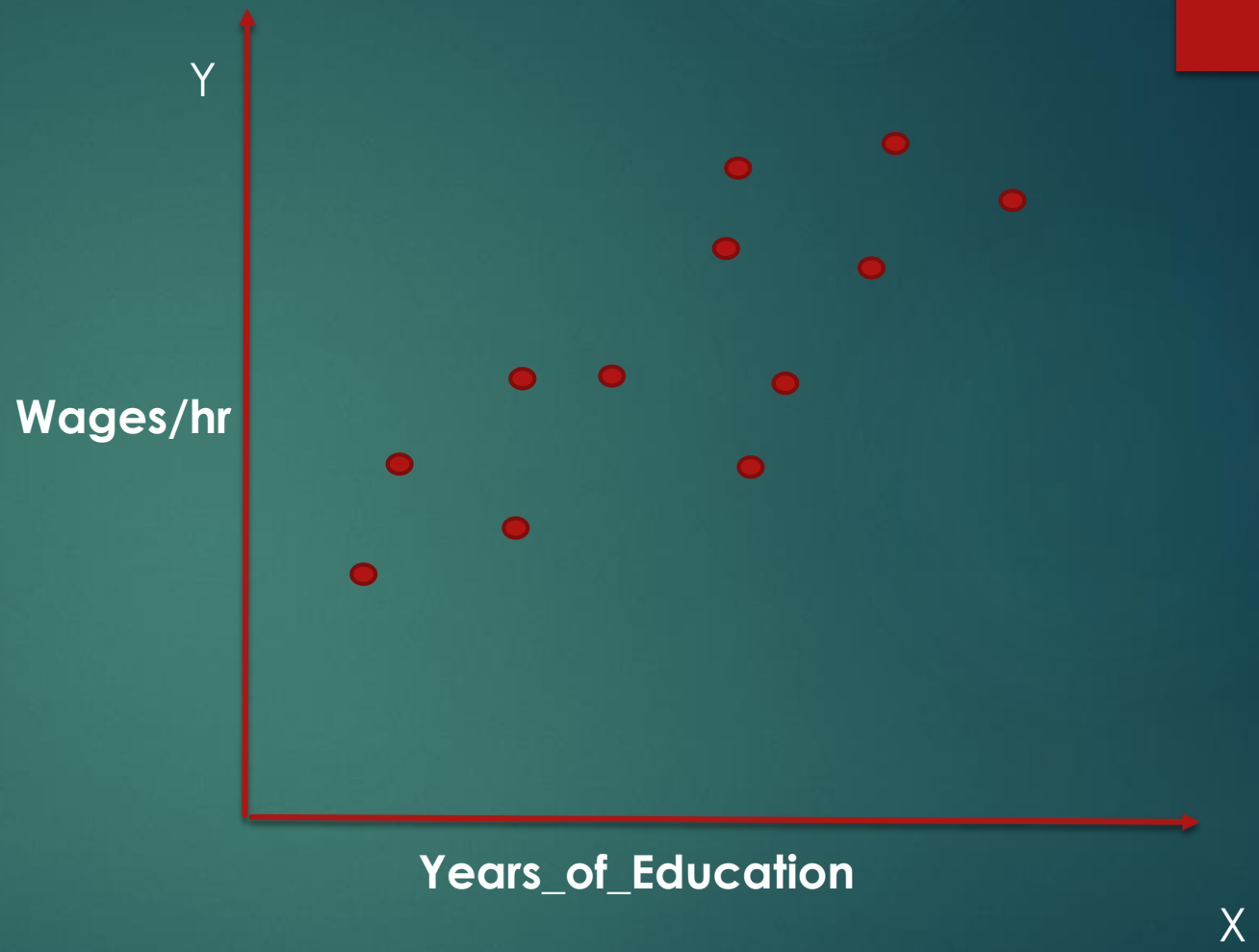
Analysis:
4PM to 8PM is not the best time for Cyclists

Regression Analysis

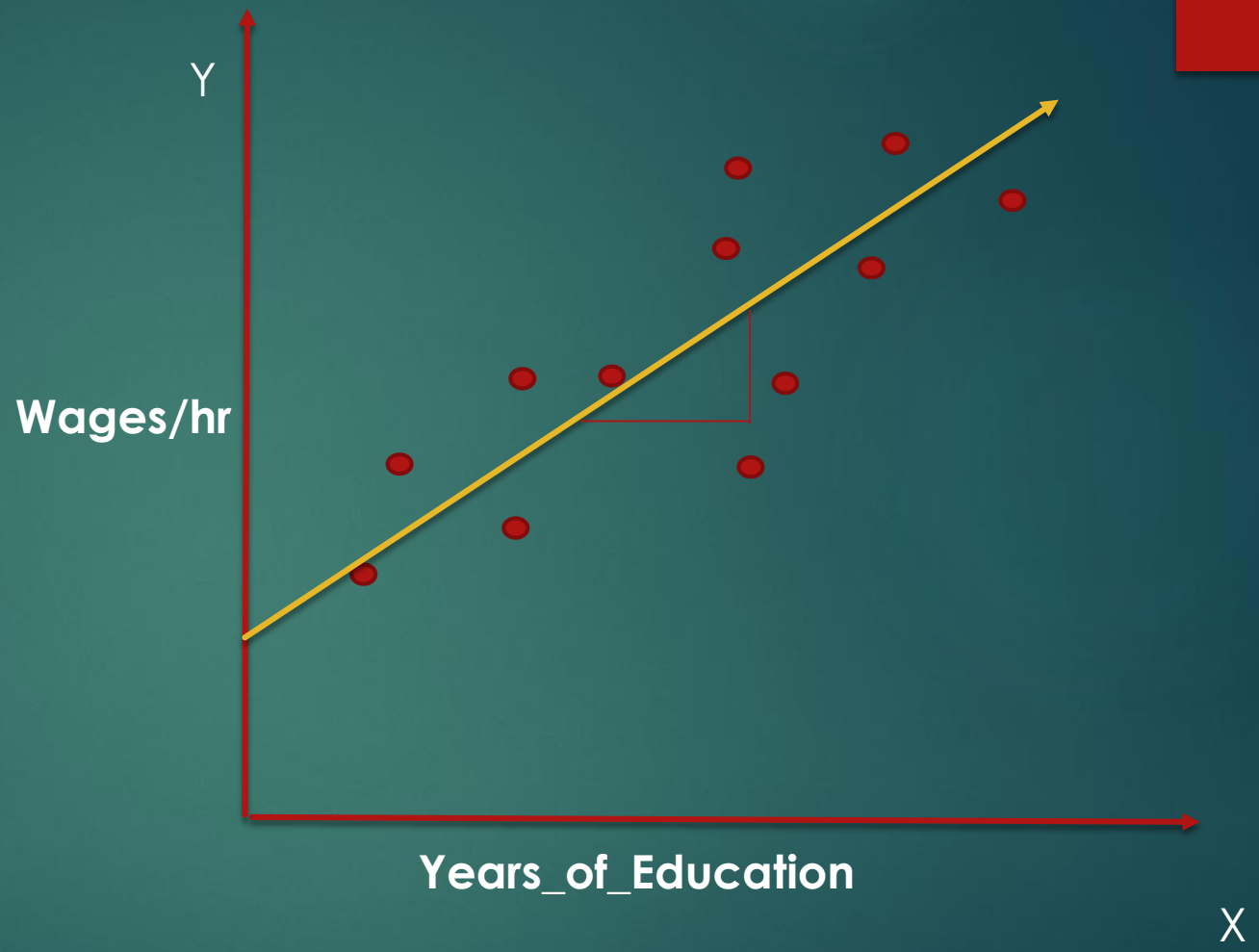
- A statistical process of establishing a relationship between the variables.
- Relationship between the independent variable ‘x’ and dependent variable ‘y’ is calculated as a nth degree polynomial.

Consider a relation between the Number of years of Education and the wages per hour.

Wages/Hour	Years_Of_Education
15	10
10	5
12	6
16	14
30	25
45	30
33	29



Wages/Hour	Years_Of_Education
15	10
10	5
12	6
16	14
30	25
45	30
33	29



Least square Method Algorithm

Step 1: Calculate the mean of the x-values and the mean of the y-values.

$$\overline{X} = \frac{\sum_{i=1}^{n} x_i}{n}$$

$$\overline{Y} = \frac{\sum_{i=1}^{n} y_i}{n}$$

Step 2: The following formula gives the slope of the line of best fit:

$$m = \frac{\sum_{i=1}^{n} \left(x_i - \overline{X}\right)\left(y_i - \overline{Y}\right)}{\sum_{i=1}^{n} \left(x_i - \overline{X}\right)^2}$$

Step 3: Compute the y-intercept of the line by using the formula:

$$b = \overline{Y} - m\overline{X}$$

Step 4: Use the slope m and the y-intercept b to form the equation of the line.

Each point has coordinates (X, Y).
We read through all the points in the set and calculate the following:
Count = the number of points
SumX = sum of all the X values
SumY = sum of all the Y values
SumX2 = sum of the squares of the X values
*SumXY = sum of the products X*Y for all the points*

Now we can find the slope M and Y-intercept YInt of the line we want:
XMean = SumX / Count
YMean = SumY / Count
*Slope = (SumXY - SumX * YMean) / (SumX2 - SumX * XMean) YInt = YMean - Slope * Xmean*

The equation for the line is:
*Y = Slope * X + YInt*

Conclusions

- Regression Analysis on historic data set(20 years)
- Clustering gone wrong
- Divide data into train test and validation test
- Better graphical representation
- Analyze based on criteria like reason for collision, contributing factors.
- Do more analysis on GPS coordinates.

Reference:

- http://www.enlistq.com/analyzing-nyc-traffic-data-using-pandas/
- http://blog.minitab.com/blog/adventures-in-statistics/regression-analysis-how-do-i-interpret-r-squared-and-assess-the-goodness-of-fit
- https://www.youtube.com/watch?v=aq8VU5KLmkY
- https://www.youtube.com/watch?v=k_OB1tWX9PM
- https://data.cityofnewyork.us/Public-Safety/NYPD-Motor-Vehicle-Collisions/h9gi-nx95
- https://realpython.com/blog/python/analyzing-obesity-in-england-with-python/
- https://www.researchgate.net/post/Whats_the_difference_between_training_set_and_test_set
- http://faculty.cs.niu.edu/~hutchins/csci230/best-fit.htm
- http://hotmath.com/hotmath_help/topics/line-of-best-fit.html
- http://www.efunda.com/math/leastsquares/lstsqrmdcurve.cfm

Thank You