We will be predicting the future price of Google’s stock using simple linear regression. The data that we will be using is real data obtained from [Google Finance](https://www.google.com/finance/historical?cid=304466804484872&startdate=Feb+1%2C+2016&enddate=Feb+28%2C+2016&num=30) saved to a CSV file, google.csv

|  |  |
| --- | --- |
| Date | Open |
| 26 | 708.58 |
| 25 | 700.01 |
| 24 | 688.92 |
| 23 | 701.45 |
| 22 | 707.45 |
| 19 | 695.03 |
| 18 | 710 |
| 17 | 699 |
| 16 | 692.98 |
| 12 | 690.26 |
| 11 | 675 |
| 10 | 686.86 |
| 9 | 672.32 |
| 8 | 667.85 |
| 5 | 703.87 |
| 4 | 722.81 |
| 3 | 770.22 |
| 2 | 784.5 |
| 1 | 750.46 |

In the above dataset, we have the prices at which the Google stock opened from February 1 – February 26, 2016. Using this data, we will try to predict the price at which the stock will open on February 29, 2016. We will be using scikit-learn, csv, numpy and matplotlib packages to implement and visualize simple linear regression.

First, let’s import the above modules:

**import csv**

**import numpy as np**

**from sklearn import linear\_model**

**import matplotlib.pyplot as plt**

csv module is used to read data from the file “google.csv”. numpy is used for array processing and conversion. Sklearn (scikitlearn) is used to implement linear regression. And, matplotlib is used to plot the data-points on graph.

First, let’s define a method to read data from google.csv .

**dates = []**

**prices = []**

**def get\_data(filename):**

**with open(filename,'r') as csvfile:**

**csvFileReader = csv.reader(csvfile)**

**next(csvFileReader) #skipping column names**

**for row in csvFileReader:**

**dates.append(int(row[0]))**

**prices.append(float(row[1]))**

**return**

Now, let’s define a function to predict the price of Google’s stock on a given date.

**def predict\_price(dates,prices,x):**

**linear\_mod = linear\_model.LinearRegression() #defining the linear regression model**

**dates = np.reshape(dates,(len(dates),1)) # converting to matrix of n X 1**

**prices = np.reshape(prices,(len(prices),1))**

**linear\_mod.fit(dates,prices) #fitting the data points in the model**

**predicted\_price =linear\_mod.predict(x)**

**return predicted\_price[0][0],linear\_mod.coef\_[0][0] ,linear\_mod.intercept\_[0]**

**The method predict\_price takes 3 arguments,**

– dates: the list of dates in integer type

– prices: the opening price of stock for the corresponding date

– x: the date for which we want to predict the price (i.e. 29)

The fit method fits the dates and prices (x’s and y’s) to generate coefficient and constant for regression. Finally, the predict method finds the price(y) for the given date (x) and returns the predicted price, the coefficient and the constant of the relationship equation.

To understand the concept of regression better, we can use matplotlib python module to plot the data-points and the relationship formed between them.

Note: The show\_plot method draws the graph using matplotlib. Do not worry if you do not understand the below code completely. It is more important to understand the graph which follows the below code. However, the show\_plot method below is commented to help you in understanding the code.

**def show\_plot(dates,prices):**

**linear\_mod = linear\_model.LinearRegression()**

**dates = np.reshape(dates,(len(dates),1)) # converting to matrix of n X 1**

**prices = np.reshape(prices,(len(prices),1))**

**linear\_mod.fit(dates,prices) #fitting the data points in the model**

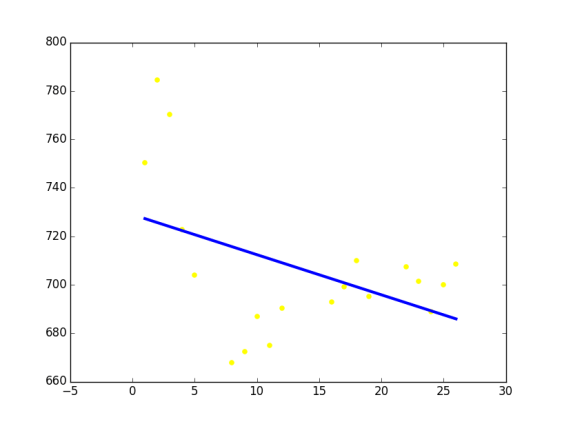
**plt.scatter(dates,prices,color='yellow') #plotting the initial datapoints**

**plt.plot(dates,linear\_mod.predict(dates),color='blue',linewidth=3) #plotting the line made by linear regression**

**plt.show()**

**return**

|  |
| --- |
|  |



**Plot/Graph of data points(yellow) and relationship equation(blue)**

The yellow dots in the above plot show the data-points plotted for each date and price (i.e. the initial dataset)

The blue line is the equation formed by the fit method of the linear model (see predict\_price method above)

Now, when we input the date February 29 to the regression model, it just uses the equation of the blue straight line in the above plot, and finds the corresponding value on y axis.

**See the full program code below.**

import csv

import numpy as np

from sklearn import linear\_model

import matplotlib.pyplot as plt

dates = []

prices = []

def get\_data(filename):

with open(filename,'r') as csvfile:

csvFileReader = csv.reader(csvfile)

next(csvFileReader) #skipping column names

for row in csvFileReader:

dates.append(int(row[0]))

prices.append(float(row[1]))

return

def show\_plot(dates,prices):

linear\_mod = linear\_model.LinearRegression()

dates = np.reshape(dates,(len(dates),1)) # converting to matrix of n X 1

prices = np.reshape(prices,(len(prices),1))

linear\_mod.fit(dates,prices) #fitting the data points in the model

plt.scatter(dates,prices,color='yellow') #plotting the initial datapoints

plt.plot(dates,linear\_mod.predict(dates),color='blue',linewidth=3) #plotting the line made by linear regression

plt.show()

return

def predict\_price(dates,prices,x):

linear\_mod = linear\_model.LinearRegression() #defining the linear regression model

dates = np.reshape(dates,(len(dates),1)) # converting to matrix of n X 1

prices = np.reshape(prices,(len(prices),1))

linear\_mod.fit(dates,prices) #fitting the data points in the model

predicted\_price =linear\_mod.predict(x)

return predicted\_price[0][0],linear\_mod.coef\_[0][0] ,linear\_mod.intercept\_[0]

get\_data('google.csv') # calling get\_data method by passing the csv file to it

print dates

print prices

print "\n"

show\_plot(dates,prices)

#image of the plot will be generated. Save it if you want and then Close it to continue the execution of the below code.

predicted\_price, coefficient, constant = predict\_price(dates,prices,29)

print "The stock open price for 29th Feb is: $",str(predicted\_price)

print "The regression coefficient is ",str(coefficient),", and the constant is ", str(constant)

print " the relationship equation between dates and prices is: price = ",str(coefficient),"\* date + ",str(constant)

**The above program gives the below output:**

**[26, 25, 24, 23, 22, 19, 18, 17, 16, 12, 11, 10, 9, 8, 5, 4, 3, 2, 1]**

**[708.58, 700.01, 688.92, 701.45, 707.45, 695.03, 710.0, 699.0, 692.98, 690.26, 675.0, 686.86, 672.32, 667.85, 703.87, 722.81, 770.22, 784.5, 750.46]**

**The stock open price for 29th Feb is: $ 680.925520298**

**The regression coefficient is -1.65535514798 , and the constant is 728.93081959**

**the relationship equation between dates and prices is: price = -1.65535514798 \* date + 728.93081959**

**[Finished in 6.2s]**

**See the last line of the output. They show the equation of the blue line formed in the plot.**