

# Linear Regression Python

December 23, 2015

Linear Regression Python Tutorial by Michael Galarnyk  
youtube video on how to install anaconda on mac osx:  
<https://www.youtube.com/watch?v=B6d5LrA8bNE>  
youtube video explaining linear regression using python (this notebook):  
<https://www.youtube.com/watch?v=dSYJVbj4Eew>

```
In [4]: import numpy as np
import pandas as pd
from sklearn.linear_model import LinearRegression
%pylab inline
import matplotlib.pyplot as plt
```

Populating the interactive namespace from numpy and matplotlib

```
In [5]: raw_data = pd.read_csv("linear.csv") #any dataset will work. You can get the data from my github
# https://github.com/mGalarnyk/Linear_Regression
raw_data.head(3)
```

```
Out[5]:
```

	x	y
0	82.583220	134.907414
1	73.922466	134.085180
2	34.887445	NaN

1) Preprocess the data to remove any points with a missing y value

```
In [6]: filtered_data = raw_data[~np.isnan(raw_data["y"])] #removes rows with NaN in them
filtered_data.head(3)
```

```
Out[6]:
```

	x	y
0	82.583220	134.907414
1	73.922466	134.085180
3	61.839983	114.530638

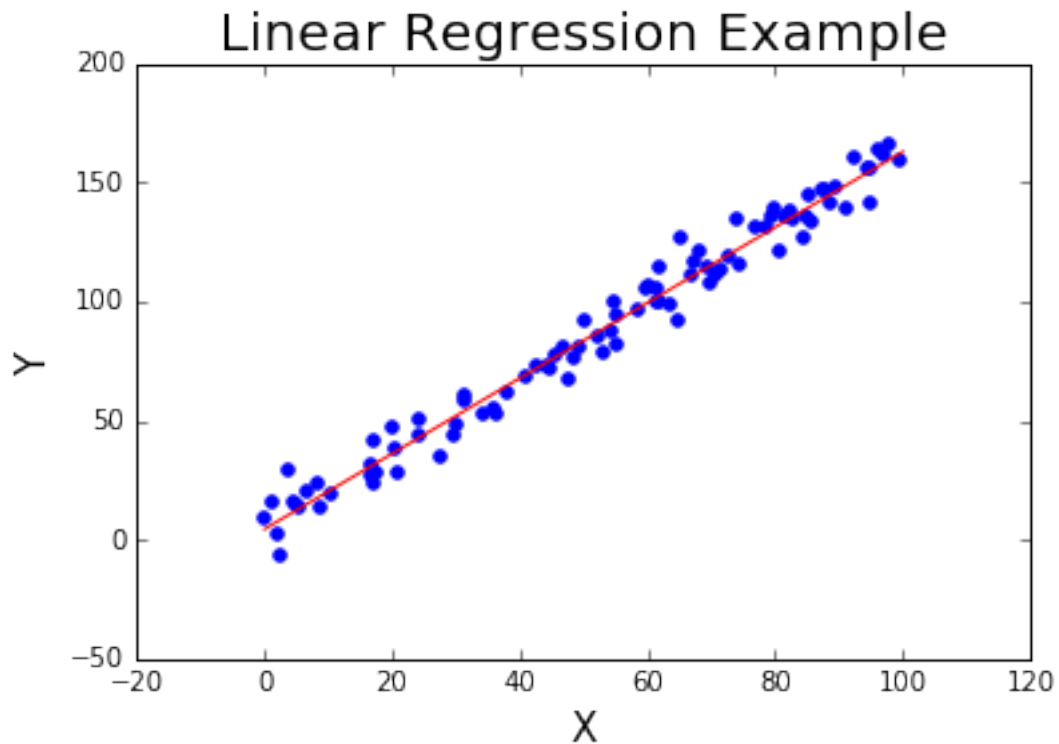
2) Fit a linear regression model using sklearn's LinearRegression package

```
In [7]: npMatrix = np.matrix(filtered_data)
X, Y = npMatrix[:,0], npMatrix[:,1]
mdl = LinearRegression().fit(X,Y) # either this or the next line
#mdl = LinearRegression().fit(filtered_data[['x']],filtered_data.y)
m = mdl.coef_[0]
b = mdl.intercept_
print "formula: y = {0}x + {1}".format(m, b) # following slope intercept form
```

```
formula: y = [ 1.5831968]x + [ 4.4701969]
```

```
In [8]: plt.scatter(X,Y, color='blue')
plt.plot([0,100],[b,m*100+b], 'r')
plt.title('Linear Regression Example', fontsize = 20)
plt.xlabel('X', fontsize = 15)
plt.ylabel('Y', fontsize = 15)
```

Out[8]: <matplotlib.text.Text at 0x10ba4e250>



## 1 official documentation

[http://scikit-learn.org/stable/auto\\_examples/linear\\_model/plot\\_ols.html](http://scikit-learn.org/stable/auto_examples/linear_model/plot_ols.html)