

Simple Liner Regression

```
In [1]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

import sklearn

from pylab import rcParams
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import scale
```

```
In [3]: %matplotlib inline
rcParams['figure.figsize'] = 10,8
```

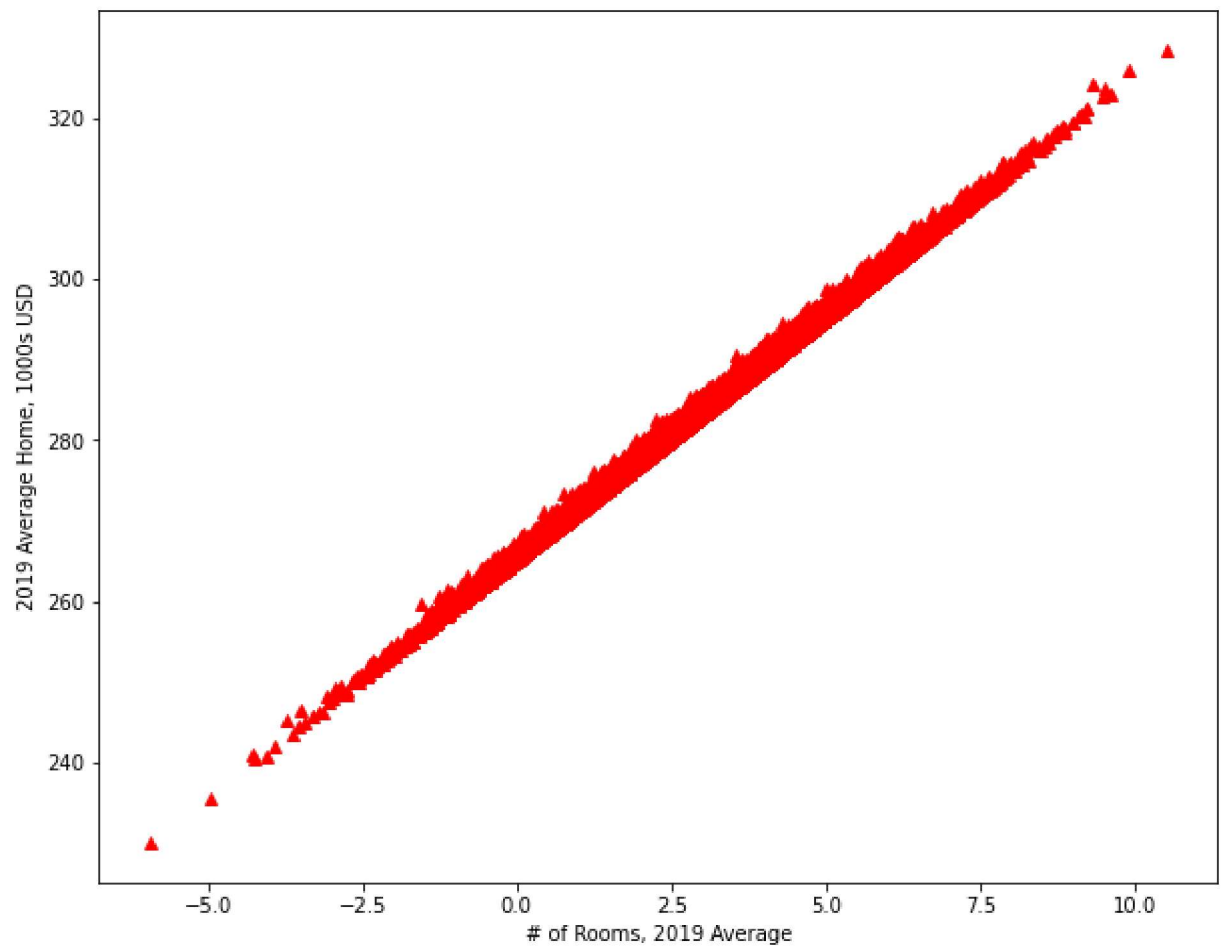
```
In [25]: # we are using random number generator to populate the variable
rooms = 2*np.random.randn(10000,1)+3
rooms[1:10]
```

```
Out[25]: array([[ 2.02586698],
 [ 1.61179875],
 [ 0.45132361],
 [-0.1085739 ],
 [ 3.01857583],
 [ 5.88662856],
 [ 0.50871375],
 [ 2.71598556],
 [ 0.47297598]])
```

```
In [26]: price = 265 + 6*rooms+abs(np.random.randn(10000,1)) # Creating Y Variable
price[1:10]
```

```
Out[26]: array([[277.22442074],
 [275.44214703],
 [268.16429965],
 [264.85838981],
 [283.65950116],
 [301.31982612],
 [269.11255181],
 [282.13878255],
 [268.50464914]])
```

```
In [27]: #Creating a scatterplot
plt.plot(rooms,price,'r^')
plt.xlabel("# of Rooms, 2019 Average")
plt.ylabel("2019 Average Home, 1000s USD")
plt.show()
```



```
In [28]: x = rooms #predictor
y = price
LinReg = LinearRegression()
LinReg.fit(x,y)
print(LinReg.intercept_, LinReg.coef_)
```

```
[265.79308353] [[6.00092655]]
```

In [29]: `print(LinReg.score(x,y))`

0.9974160009771208

In []: