ScriptScriptScriptScriptGrip: The Sparks Foundation

Data Science and Business Analytics Intern (May 2021)

Author: Lakshita M Bhandari

Task1 Prediction using Supervised ML

In this task we have to predict the percentage score of students based on the number of hours studied. The data set has 2 variables 'hours' and 'scores'. This can be solved using Simple linear regression.

In [112]:

#Importing the python libraries

In [113]:

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

%matplotlib inline

Getting the data

In [114]:

data=pd.read\_csv("D:/Desktop/Sample.csv")

data

Out[114]:

|  | Hours | Scores |
| --- | --- | --- |
| 0 | 2.5 | 21 |
| 1 | 5.1 | 47 |
| 2 | 3.2 | 27 |
| 3 | 8.5 | 75 |
| 4 | 3.5 | 30 |
| 5 | 1.5 | 20 |
| 6 | 9.2 | 88 |
| 7 | 5.5 | 60 |
| 8 | 8.3 | 81 |
| 9 | 2.7 | 25 |
| 10 | 7.7 | 85 |
| 11 | 5.9 | 62 |
| 12 | 4.5 | 41 |
| 13 | 3.3 | 42 |
| 14 | 1.1 | 17 |
| 15 | 8.9 | 95 |
| 16 | 2.5 | 30 |
| 17 | 1.9 | 24 |
| 18 | 6.1 | 67 |
| 19 | 7.4 | 69 |
| 20 | 2.7 | 30 |
| 21 | 4.8 | 54 |
| 22 | 3.8 | 35 |
| 23 | 6.9 | 76 |
| 24 | 7.8 | 86 |

In [115]:

data.isnull().sum()

Out[115]:

Hours 0

Scores 0

dtype: int64

In [116]:

data.describe()

Out[116]:

|  | Hours | Scores |
| --- | --- | --- |
| count | 25.000000 | 25.000000 |
| mean | 5.012000 | 51.480000 |
| std | 2.525094 | 25.286887 |
| min | 1.100000 | 17.000000 |
| 25% | 2.700000 | 30.000000 |
| 50% | 4.800000 | 47.000000 |
| 75% | 7.400000 | 75.000000 |
| max | 9.200000 | 95.000000 |

In [117]:

data.info()

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 25 entries, 0 to 24

Data columns (total 2 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 Hours 25 non-null float64

1 Scores 25 non-null int64

dtypes: float64(1), int64(1)

memory usage: 528.0 bytes

In [118]:

corr=data.corr()

corr.shape

Out[118]:

(2, 2)

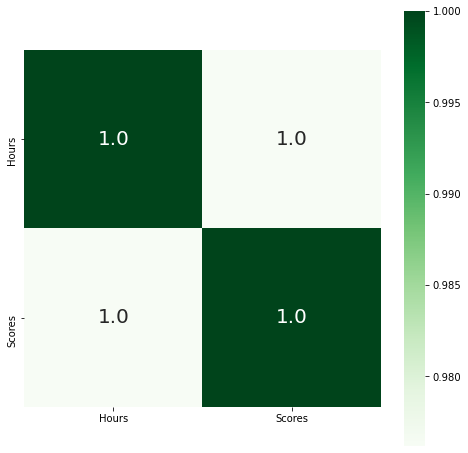
In [119]:

plt.figure(figsize=(8,8))

sns.heatmap(corr, cbar=True, square=True, fmt='.1f', annot=True, annot\_kws={'size':20}, cmap='Greens')

Out[119]:

<AxesSubplot:>



Defining the features and target value

In [120]:

x=data.Hours #FEATURE VALUE

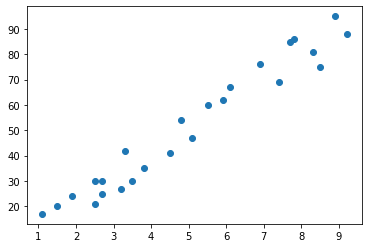
y=data.Scores #target value

In [121]:

plt.scatter(x,y)

Out[121]:

<matplotlib.collections.PathCollection at 0x10d8f9b17f0>



Machine Learning: Linear Regression(Line Fitting)

In [122]:

model = np.polyfit(x,y,1)

In [123]:

model

Out[123]:

array([9.77580339, 2.48367341])

In [129]:

predict = np.poly1d(model)

In [130]:

#model accuracy

from sklearn.metrics import r2\_score

r2\_score(y, predict(x))

Out[130]:

0.9529481969048356

# what will be the predicted score if a student studies for 9.25hrs/day?[¶](#what-will-be-the-predicted-score-if-a-student-studies-for-9.25hrs/day?)

In [131]:

Hours = 9.25

print("if a student studies for 9.25 hrs/day, then predicted score is:", predict(Hours))

if a student studies for 9.25 hrs/day, then predicted score is: 92.90985477015728

Plotting the linear regression model

In [132]:

x\_lin\_reg = range(0,12)

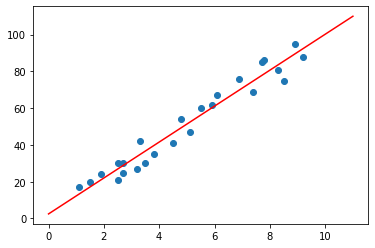
y\_lin\_reg = predict(x\_lin\_reg)

plt.scatter(x,y)

plt.plot(x\_lin\_reg,y\_lin\_reg, c='r')

Out[132]:

[<matplotlib.lines.Line2D at 0x10d90fb2910>]



In [ ]: