Bivariate Statistics Exploration

Jacob Bruner

September 23, 2022

1 Mean of X, Y for Datasets

1.1 Method

The mean is a commonly used measure of center for a given statistic. To calculate the mean, we can use the following formula: for a given indexed set of values, the mean is written:

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

Where N is the length of the data set. In other words, the mean is the sum of each value divided by the number of values for a given parameter.

Calculating these values for each dataset we obtain:

Data set	**I**		**II**		**III**		**IV**	
Parameter	x	y	x	y	x	y	x	y
Mean	9.0	7.5	9.0	7.5	9.0	7.5	9.0	7.5

1.2 Interpretation

After computing the mean for each, it's clear that the means for each x, y of the respective datasets are equal. Without any other prior knowledge, one might infer that these datasets have a similar center, save variation or spread. This is because the mean doesn't provide information about the variation or spread of the data. Despite this, however, one might assume it to be a reasonable assumption that these datasets and parameters might be highly similar, especially because they match in two variables.

2 Exploring Other Statistical Measures

In addition to computing the mean, there are a number of other tools that provide insight into the properties of a dataset. For instance, the variance and standard deviation convey a measure of the 'spread' or variation in a dataset. Similarly, 'Peterson's Correlation Coefficient' often denoted with its square, r^2, is meaasure of the *linear* coorelation of two parameters. Values of r range from -1 to 1, with numbers farther from zero denoting stronger coorelation. A closely related notion is the covarience, which measures the linear coorelation, but with a magnitude not necessarily between $(-1, 1)$. These measures are related like so: $r = \frac{cov(x,y)}{\sigma_x \sigma_y}$ where σ denotes the standard deviation.

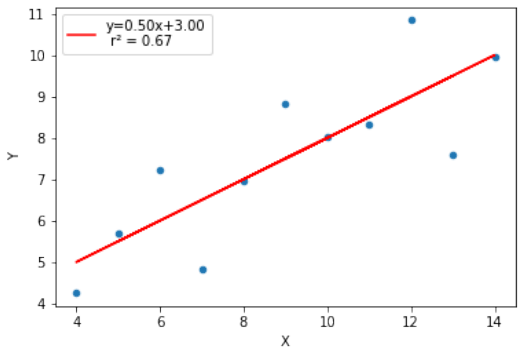
	$\bar{x}$	$\bar{y}$	σ_x^2	σ_y^2	σ_x	σ_y	r^2	$cov(x,y)$
set 1	9.0	7.5	11.0	4.13	3.32	2.03	0.67	5.5
set 2	9.0	7.5	11.0	4.13	3.32	2.03	0.67	5.5
set 3	9.0	7.5	11.0	4.12	3.32	2.03	0.67	5.5
set 4	9.0	7.5	11.0	4.12	3.32	2.03	0.67	5.5

3 Plots

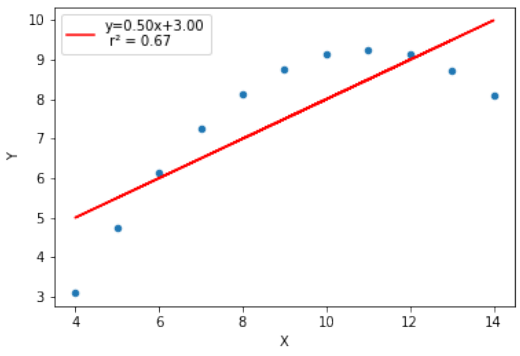
Using the python packages `matplotlib` and `numpy`, we can plot each dataset like so:

```
for i in range(4):
  dfs[i].plot.scatter(x='x', y='y')
  reg = np.polyfit(dfs[i]['x'], dfs[i]['y'], 1)
  rval = stats.pearsonr(dfs[i]['x'], dfs[i]['y'])[0]
  plt.plot(dfs[i]['x'], np.polyval(reg, dfs[i]['x']), 'r', label='y={:.2f}x+{:.2f}\n r^2 =
      {:.2f}'.format(reg[0], reg[1], rval**2))
  plt.xlabel('X')
  plt.ylabel('Y')
  plt.legend()
  plt.savefig('dataset{}.png'.format(i+1))
```

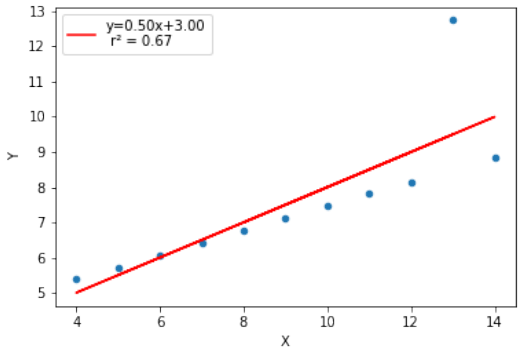
After doing so, we get the following interesting results:



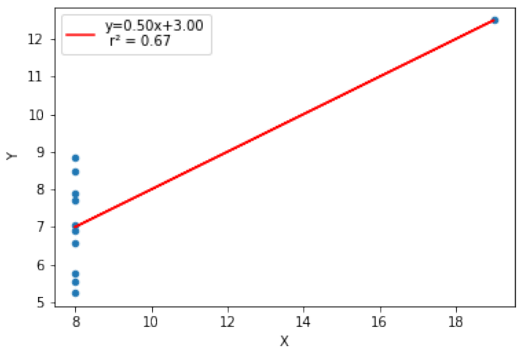
(a) Dataset I



(b) Dataset II



(c) Dataset III



(d) Dataset IV

Figure 1: Plot of Datasets I-IV — X against Y with Lin. Reg.