05-Lab03

January 13, 2021

1 Lab 3

In this lab, you will complete the full process of describing and then implementing an algorithm. You will do so on a topic relevant to data science: *(simple) linear regression*. Please note that this lab is assessed, i.e., you will need to submit the results of your work on QM+. To submit your work, first download/export your Jupyter notebook as PDF. Then upload the PDF file in the submission area on QM+.

1.1 Example: Computing the mean value

Input: collection of integer values Output: average/mean of the input integers

Concrete input example values: 10, 20, 25, 1, 3 Compute the mean: $(10+20+25+1+3)/5 = 11.8$

Algorithm: 1. Compute the sum of the input integers 2. Determine the number of inputs 3. Divide the the sum computed in step 1 by the number determined in step 2

[1]:

```python
# Compute the mean of a collection of input values
input_integers = [10, 20, 25, 1, 3]
# 1. Compute the sum of the input integers
computed_sum = 0
for x in input_integers:
    computed_sum = computed_sum + x
# 2. Determine the number of inputs
number_of_inputs = 0
for x in input_integers:
    number_of_inputs = number_of_inputs + 1
# 3. Divide the the sum computed in step 1 by the number determined in step 2
computed_sum / number_of_inputs
```

[1]:

```
11.8
```

[2]:

```python
# Compute the mean of a collection of input values
input_integers = [10, 20, 25, 1, 3]
# 1. Compute the sum of the input integers
computed_sum = sum(input_integers)
# 2. Determine the number of inputs
number_of_inputs = len(input_integers)
# 3. Divide the the sum computed in step 1 by the number determined in step 2
```

```
computed_sum / number_of_inputs
```

```
[2]: 11.8
```

1.2 Preparation: Simple Linear Regression

If necessary (depending on your background), read up on (simple) linear regression. Wikipedia (https://en.wikipedia.org/wiki/Simple_linear_regression) provides a good starting point, but you might as well choose other sources. (You do not need to include any information about this in your submission.)

1.3 Task 1: Describe Linear Regression in English

Given a dataset of n values (x_i, y_i) for $1 \leq i \leq n$, describe the process to compute values α and β such that

$$\sum_{i=1}^{n}(y_i - \alpha x_i - \beta)^2$$

is minimal. This will yield a (linear) model $y = \alpha x + \beta$ adhering to the least-squares condition. > Marking information: Up to 30 points: 15 points for an algorithm that can be followed, and 15 points for the correctness of this algorithm against the specification "linear regression."

Description of Linear Regression

Linear regression is a technique used to model the relationships between observed variables. The idea behind simple linear regression is to "fit" the observations of two variables into a linear relationship. This linear relationship is referred to as the 'line of best fit' To get the best 'fit' we assume that the observed values have a perfect linear relationship, a correlation equal to one. In order to get the best fit the aim is to minimize the square of the distance between the observed variables and the line. This is achieved by taking the observed value and deducting this from the value of y at the line. The aim is to find the lowest value (residual) the lower the residual the higher the certainty of best fit of our line.

Linear Regression Algorithm

1.Consider the observed variables in the dataset

2.Calculate the mean of the X and y observed variables from the dataset

3.Calculate the deviation of the X observed variables from the mean of X

4.Calculate the deviation of the y observed variables from the mean of y

5.Find the product of the X and y deviations for each observed variable by multiplying the X and y deviations and then summing them to come to total

6.Square each of the X deviations and sum them to arrive at a total

7.Calculate the slope (b1): This is achieved by taking the sum product from step 5 and dividing it by the sum deviations in step 6.

8.Calculate the intercept (b0): This is achieved by multiplying the slope by the X mean value and subtracting the result from the y mean.

9.The result is then the line of best fit which is equal to the slope multiplied by X plus the intercept

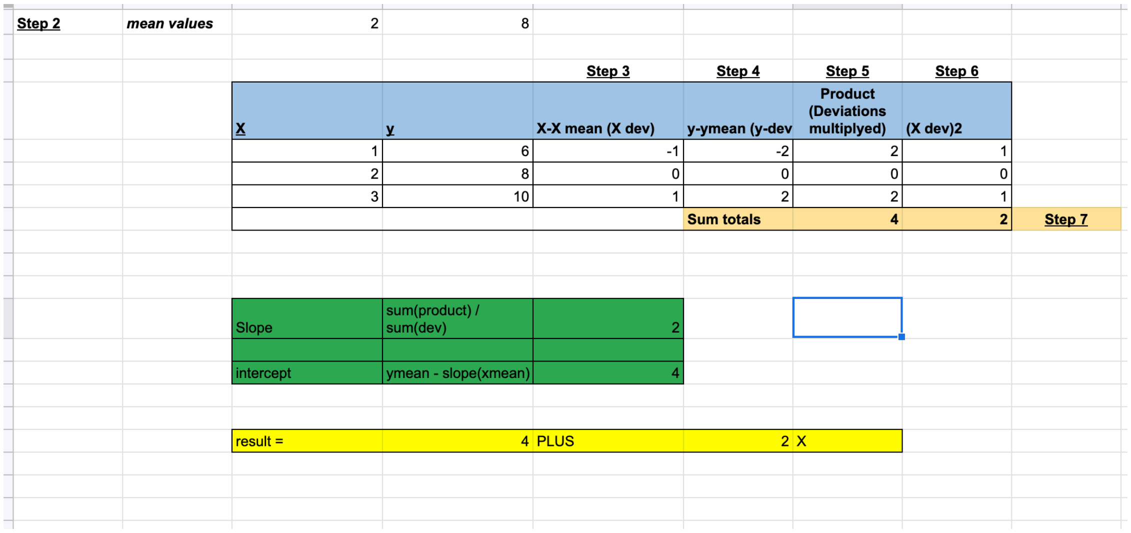
1.4 Task 2: Create and Complete a Simple Example

Come up with a dataset of just 3 values ($\{(x_1, y_1), (x_2, y_2), (x_3, y_3)\}$). Then exercise the algorithm of Task 1 on this small dataset. You might prefer to do this using pen and paper. For your submission, provide at least the dataset that you have chosen and the resulting values of α and β. > Marking information: Up to 20 points: 5 points for sample values, and up to 15 points for suitable values of α and β.

```
[3]: data = [[1,6], [2,8], [3,10]]
```

```
[4]: from IPython.display import Image
     Image(filename='Lab05.png',width=800, height=400)
```

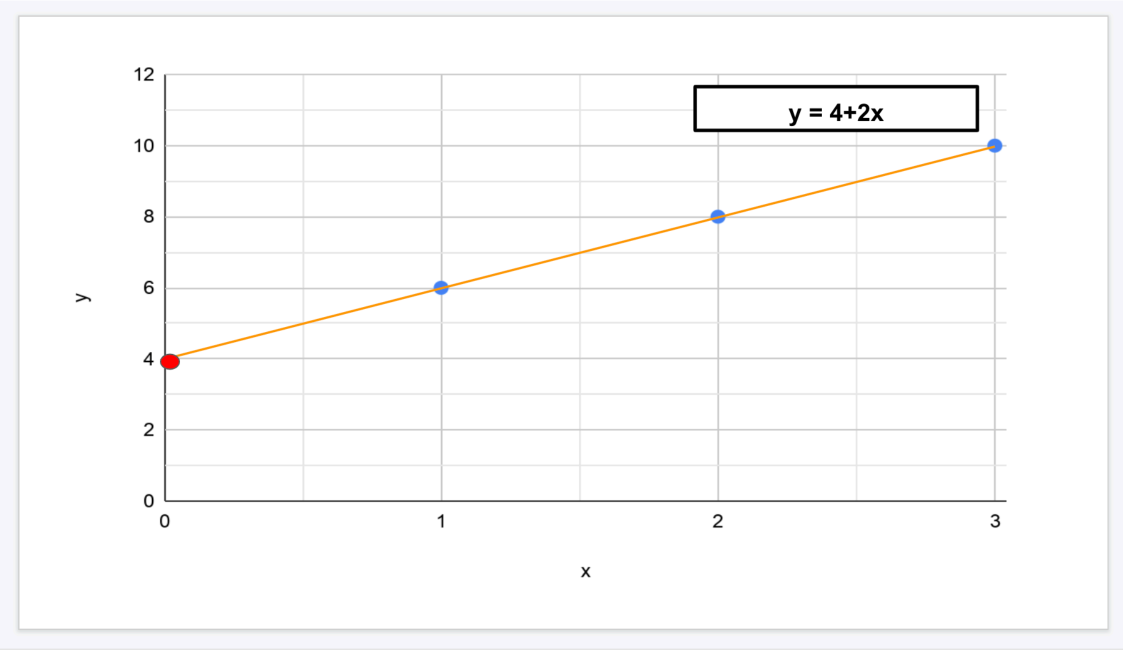
[4]:



Step 2	mean values	2	8					
				Step 3	Step 4	Step 5	Step 6	
		X	y	X-X mean (X dev)	y-ymean (y-dev	Product (Deviations multiplyed)	(X dev)2	
		1	6	-1	-2	2	1	
		2	8	0	0	0	0	
		3	10	1	2	2	1	
					Sum totals	4	2	Step 7
		Slope	sum(product) / sum(dev)	2				
		intercept	ymean - slope(xmean)	4				
		result =	4	PLUS	2	X		

```
[5]: from IPython.display import Image
     Image(filename='Lab05_1.png',width=800, height=400)
```

[5]:



In the image above you can see the line of best fit visualised on a scatter graph. When looking for guidance in making predictions. We can interpret this as telling us the value of y at 0 is 4. This is our intercept(b0). The relationship that we can see here is that when X increase by 1, y will increase by 2. You can see that the line goes through 2 squares on the graph each time it goes up by one to support our findings.

1.5 Task 3: Implement the Algorithm of Task 1 in Python

Starting from your description of the steps to undertake, transfer English and Maths to Python. Note that you are expected to fully implement the mathematical operations instead of using a library function such as `scikit` or `statsmodels`. > Marking information: Up to 40 points: 30 points for a correctly working Python implementation, and 10 points for comments and overall readability.

[6]:
```
x_values = [d[0] for d in data]
y_values = [d[1] for d in data]
```

[7]:
```
x_values
```

[7]:
```
[1, 2, 3]
```

[8]:
```
y_values
```

[8]:
```
[6, 8, 10]
```

[9]:

```
def sls_fit(X, Y):
    '''Takes two lists of values
    refering to predictor and target variable,
    returns the regression coefficients beta_0 and beta_1.'''

    if len(X) != len(Y):
        print('Passed arrays of unequal length.')
        return None

    y_bar = sum(Y) / len(Y)
    x_bar = sum(X) / len(X)
    std_y = (sum([(y_i-y_bar)**2 for y_i in Y]) / (len(Y)-1))**0.5
    std_x = (sum([(x_i-x_bar)**2 for x_i in X]) / (len(X)-1))**0.5
    r_xy = float(sum([(x_i-x_bar)*(y_i-y_bar)
                      for x_i, y_i in zip(X, Y)])) / (len(Y)-1) / (std_x*std_y)

    beta_1 = r_xy * (std_y / std_x)
    beta_0 = y_bar - beta_1 * x_bar

    return beta_0, beta_1
```

1.6 Task 4: Test the Implementation of Task 4

Use at least the dataset of Task 2, or possibly also other datasets, to exercise your implementation of Task 3. > Marking information: Up to 10 points for Python instructions to run the test and the comparison to expected values previously found using pen and paper.

[10]:

```
b_0, b_1 = sls_fit(x_values, y_values)
b_0, b_1
```

[10]:

```
(4.0, 2.0)
```