

# Homework 3

Jack Kai Lim

April 26, 2022

## Problem 1

a)

Skips assumption is incorrect as proof I performed and calculated the regression coefficient and intercept by converting all the points at first which yield,

```
SD_F = [66, 66, 67, 69, 69, 72, 76, 77, 77, 74, 70, 66]
Osaka_C = [9, 10, 14, 20, 25, 28, 32, 33, 39, 23, 18, 12]
SD_C = []
for temp in SD_F:
    SD_C.append((5/9)*(temp - 32))
```

✓ 0.3s Python

Function to get regression gradient and intercept

```
def regression(x, y):
    x_mean = mean(x)
    y_mean = mean(y)
    grad_num = 0
    grad_denom = 0
    for i in range(len(x)):
        grad_num += (x[i] - x_mean)*(y[i] - y_mean)
        grad_denom += (x[i] - x_mean)**2

    grad = grad_num/grad_denom
    intercept = y_mean - grad*(x_mean)
    return grad, intercept
```

✓ 0.5s Python

```
gradient, intercept = regression(SD_C, Osaka_C)
print("The regression equation when we convert all individual points is given as y = " + str(gradient) + "x + " + str(intercept) )
```

✓ 0.3s Python

The regression equation when we convert all individual points is given as y = 3.80290909090906x + -59.95151515151515

And when I tried to do the conversions on the gradient and intercept on the gradient and intercept when San Diego's temperature is not converted I get a different gradient and intercept,

```

gradient, intercept = regression(SD_F, Osaka_C)
gradient, intercept
✓ 0.3s

(2.1127272727272723, -127.55878787878784)

gradient = (5/9)*(gradient - 32)
intercept = (5/9)*(intercept - 32)
gradient, intercept
✓ 0.3s

(-16.604040404040404, -88.64377104377101)

```

Therefore Skip's assumption is incorrect.

A shortcut that can be used instead is to multiply the gradient coefficient by  $\frac{9}{5}$ . Why this works will be shown in part b.

**b)**

First we look at the formula to get  $w_1$ ,

$$w_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

We can apply the function of  $f(x)$  to the all the values of  $x$  and as for the mean of  $x$ , we know that it is linear. Therefore the  $mean(f(x_i)) = a \times mean(x_i) + b$  therefore we get,

$$\begin{aligned}
 w_1 &= \frac{\sum_{i=1}^n (f(x_i) - f(\bar{x}))(y_i - \bar{y})}{\sum_{i=1}^n (f(x_i) - f(\bar{x}))^2} = \frac{a \sum_{i=1}^n (x_i + b - \bar{x} - b)(y_i - \bar{y})}{a^2 \sum_{i=1}^n (x_i + b - \bar{x} - b)^2} \\
 &\Rightarrow \frac{a \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{a^2 \sum_{i=1}^n (x_i - \bar{x})^2} = \frac{1}{a} \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}
 \end{aligned}$$

Therefore we get that more generally, when a linear function is applied to the  $x$  variable, the linear regression coefficient changes by a factor of  $\frac{1}{a}$

## Problem 2

First I wrote a function to calculate the Mean squared error.

```
def MSE(y_actual, y_pred):
    sum = 0
    for i in range(len(y_actual)):
        sum += (y_actual[i] - y_pred[i])**2
    return sum/len(y_actual)
```

✓ 0.3s

a)

I used the regression function which I defined in the Problem 1 which uses least square regression to get the gradient and the intercept of the regression line, which I then used to get the MSE.

```
x = [1, 3, 0, 4]
y = [8, 12, 4, 17]
gradient, intercept = regression(x, y)
pred = []
for i in range(4):
    pred.append(intercept + gradient*x[i])

MSE1 = MSE(y, pred)
print("The value of b0 is " + str(intercept) + " and the value of b1 is " + str(gradient))
print("The mean squared error is " + str(MSE1))
```

✓ 0.3s

... The value of b0 is 4.25 and the value of b1 is 3.0  
The mean squared error is 0.6875

b)

For this I did the exact same steps but with the number of inches of rain instead

```
z = [3, 7, 1, 9]
y = [8, 12, 4, 17]
gradient, intercept = regression(z, y)
pred = []
for i in range(4):
    pred.append(intercept + gradient*z[i])
print("The value of d0 is " + str(intercept) + " and the value of d1 is " + str(gradient))
MSE2 = MSE(y, pred)
print("The mean squared error is " + str(MSE2))
```

✓ 0.5s

... The value of d0 is 2.75 and the value of d1 is 1.5  
The mean squared error is 0.6875

c)

We have that  $y_i = d_0 + d_1 z_i$ , first we need to find  $d_1$  in terms of b and c,

$$b_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

Using that fact and the fact we we proved in problem 1 where the least regression formula, when a linear transformation is applied to  $x$ , we get a change of the factor of  $\frac{1}{c_1}$  where  $c_1$  is the scalar in the linear transformation. Using these facts we get,

$$d_1 = \frac{1}{c_1} \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} = \frac{b_1}{c_1}$$

Next using the equation of the line and the known values of  $z$  and  $y$  we get,

$$d_0 = \bar{y} - \frac{b_1}{c_1} \bar{z}$$

Hence we can finally get the predicted line to be,

$$\hat{y}_i = \bar{y} - \frac{b_1}{c_1} \bar{z} + \frac{b_1}{c_1} z_i$$

Now we can get the mean squared error for  $z$  and  $y$  in terms of  $c$ ,  $b$  and  $x$ .

$$\begin{aligned} R_{sq}((d_0; d_1), D_z) &= \frac{1}{n} \sum_{i=1}^n (y_i - \bar{y} + \frac{b_1}{c_1} \bar{z} - \frac{b_1}{c_1} z_i)^2 \\ &= \frac{1}{n} \sum_{i=1}^n (y_i - \bar{y} + \frac{b_1}{c_1} [c_0 + c_1 \bar{x} - c_0 - c_1 x_i])^2 \\ &= \frac{1}{n} \sum_{i=1}^n (y_i - \bar{y} + b_1(\bar{x} - x_i))^2 \end{aligned}$$

Now we find the mean square error for  $x$  and  $y$ , but first we need to obtain  $b_0$  in terms of only  $b_1$  and  $x$ ,

$$\begin{aligned} b_0 &= \bar{y} - b_1 \bar{x} \\ \hat{y}_i &= \bar{y} - b_1 \bar{x} + b_1 x_i \end{aligned}$$

Using that we get the mean square error to be,

$$\begin{aligned} R_{sq}((b_0; b_1), D_x) &= \frac{1}{n} \sum_{i=1}^n (y_i - \bar{y} + b_1 \bar{x} - b_1 x_i)^2 \\ &= \frac{1}{n} \sum_{i=1}^n (y_i - \bar{y} + b_1(\bar{x} - x_i))^2 \end{aligned}$$

As we can see, the two equations are equal, therefore the relationship is true.

### Problem 3

First we calculate the least square regression for the six data points and let  $w_1$  be the gradient for the line,

$$\begin{aligned}w_1 &= \frac{(y_1 - \bar{y})(5 - 10) + (y_2 - \bar{y})(5 - 10) + (y_5 - \bar{y})(15 - 10) + (y_6 - \bar{y})(15 - 10)}{25 + 25 + 25 + 25} \\ &= \frac{5(y_5 + y_6 - y_1 - y_2)}{100}\end{aligned}$$

Then to get the equation of the line we do,

$$w_0 = y - \frac{5(y_5 + y_6 - y_1 - y_2)}{100}x = \bar{y} - \frac{1(y_5 + y_6 - y_1 - y_2)}{2}$$

Therefore we get the equation of the line to be,

$$\hat{y}_i = \bar{y} - \frac{1(y_5 + y_6 - y_1 - y_2)}{2} + \frac{5(y_5 + y_6 - y_1 - y_2)}{100}x_i$$

Next we look at the equation with 3 points instead, first we redefine the values of  $y$  to have similar variables. So we let,  $\bar{y}_1 = \frac{y_1 + y_2}{2}$  and  $\bar{y}_2 = \frac{y_3 + y_4}{2}$  and  $\bar{y}_3 = \frac{y_5 + y_6}{2}$ . Then we can find the gradient for this regression line, let  $d_1$  be the gradient of the line.

$$d_1 = \frac{(\bar{y}_1 - \bar{y})(5 - 10) + (\bar{y}_3 - \bar{y})(15 - 10)}{25 + 25} = \frac{5(\bar{y}_3 - \bar{y}_1)}{50}$$

Now we can input our definition for the points of  $y$  and we will get,

$$d_1 = \frac{5\left(\frac{y_5 + y_6}{2} - \frac{y_1 + y_2}{2}\right)}{50} = \frac{5(y_5 + y_6 - y_1 - y_2)}{100}$$

We can notice that  $w_1 = d_1$  and we know that both  $\bar{x}$  and  $\bar{y}$  are equal in both problems, therefore we can conclude that the regression lines for both the 6 points and 3 points are identical.

## Problem 4

a)

I personally think that a linear model may work under the assumption that all the avocado trees are identical and that each tree will produce a set number more avocados. Therefore I think that there is a function of another form which would be better as it is safer and much more true to assume that each avocado tree has a different production rate, or are a different size or will yield a different number of avocados to be much more realistic.

b)

We can calculate the difference by subtracting  $w'_1$  by  $w_1$ ,

$$\begin{aligned}w'_1 - w_1 &= \frac{(15 - 80)(100 - \bar{y}) + \sum_{i=1}^{99} (x_i - 80)(y_i - \bar{y})}{\sum_{i=1}^{100} (x_i - 80)^2} \\ &\quad - \frac{(15 - 80)(650 - \bar{y}) + \sum_{i=1}^{99} (x_i - 80)(y_i - \bar{y})}{\sum_{i=1}^{100} (x_i - 80)^2} \\ &= \frac{-65(100 - \bar{y})}{\sum_{i=1}^{100} (x_i - 80)^2} - \frac{-65(650 - \bar{y})}{\sum_{i=1}^{100} (x_i - 80)^2} \\ &= \frac{35750}{\sum_{i=1}^{100} (x_i - 80)^2}\end{aligned}$$

This can be further simplified, as we know that variance of a dataset is equal to  $\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2$ , therefore using the standard deviation we can get the value of the denominator to be,

$$\sum_{i=1}^{100} (x_i - 80)^2 = 30^2 \times 100 = 90000$$

Therefore the difference in the slope is,

$$\frac{35750}{90000} = 0.397222 \dots$$

c)

The farmer with 20 avocado trees will be affected more, as the change in data happened to someone with a small number of trees. Which skews the data more for farmers with a more similar number of trees.

d)

We can tell based for the farmer. If the number of trees the farmer has is below the mean, increasing his yield would make the slope steeper as it would increase

the gradient of the slope making it steeper. While if the farmer has more than the mean in number of avocado trees, increasing the farmers yield will make the slope less steep as it would lower the value of the gradient. Furthermore it is also possible to increase a farmers yield and not affect the slope at all, and that is by changing the value of the farmer with exactly the mean number of avocado trees. As that would not affect the slope of the regression line whatsoever.

e)

First we need to minimize the mean squared error in order to do so we will take the partial derivatives of the mean squared error,

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

And since we are forcing the line to cross the origin, we get the mean square error to be,

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - w_1 x_i)^2$$

To minimize the mean squared error, we will take the derivatives of the equation with respect to  $w_1$  as we are looking for the best estimate of the value of  $w_1$

$$\frac{\delta}{\delta w_1} MSE = \frac{2}{n} \sum_{i=1}^n x_i (y_i - w_1 x_i) = 2 \frac{\sum_{i=1}^n x_i y_i}{n} - 2w_1 \frac{\sum_{i=1}^n x_i^2}{n}$$

Which if we look carefully we can change the equation into,

$$\frac{\delta}{\delta w_1} MSE = 2\bar{xy} - 2w_1\bar{x}^2$$

Now to minimize, we equate the equation to 0,

$$\begin{aligned} 2\bar{xy} - 2w_1\bar{x}^2 &= 0 \\ \bar{xy} &= w_1\bar{x}^2 \\ w_1 &= \frac{\bar{xy}}{\bar{x}^2} \end{aligned}$$

Therefore the best choice for  $w_1$  is given as  $\frac{\bar{xy}}{\bar{x}^2}$ .

## Problem 5

By using the transformation of the equation to,

$$\frac{x^2}{y} = ay + b$$

We can then applied linear regression to find the best fitting curve. In order to do so, first let  $z = \frac{x^2}{y}$  be the dependant variable and y be the independant variable. Since we are already given the values of  $\frac{x^2}{y}$  we are able to use the least square regressions formulas to calculate a equation for the regression line,

```
def regression(x, y):
    x_mean = mean(x)
    y_mean = mean(y)
    grad_num = 0
    grad_denom = 0
    for i in range(len(x)):
        grad_num += (x[i] - x_mean)*(y[i] - y_mean)
        grad_denom += (x[i] - x_mean)**2

    grad = grad_num/grad_denom
    intercept = y_mean - grad*(x_mean)
    return grad, intercept
```

5] ✓ 0.7s Python

I coded a function that uses the least regressions formula to get the gradient and intercept of the line, in this case, x is the values of y and y is the values of z which grants the equation (for clarification),

$$a = \frac{\sum_{i=1}^{12} (y_i - \bar{y})(z_i - \bar{z})}{\sum_{i=1}^{12} (y_i - \bar{y})^2}$$

for the gradient and  $b = \bar{z} - a\bar{y}$  to get the intercept. Which after entering the values of y and z we get,

```
y = [9, 10, 14, 20, 25, 28, 32, 33, 29, 23, 18, 12]
z = [484, 435, 320, 238, 190, 185, 180, 179, 204, 238, 272, 363]
grad, inter = regression(y, z)

print("The gradient of the regression line is " + str(grad) + " and the intercept of the line is " + str(inter))
```

✓ 0.5s Python

The gradient of the regression line is -11.37976128697457 and the intercept of the line is 513.92330804670472