EBA3500 Fall 2022
Exercises for lecture 3: Simple linear regression

Jonas Moss

September 13, 2022

1 The shape of the errors

In the two examples we looked at, $y = 2+3x+u$ and $y = e^{3x^2}+u$, the errors where uniform on $[-1, 1]$. Often the errors aren't this nice. First, they can be drawn from a different distribution, such as the t distribution. Second, they may depend on x. If the errors have mean 0 and are affected by x, we are dealing with something called "heteroskedasticity".

In this exercise we will use plotting to explore what happens to our two prototypical examples when the error distribution changes.

(a) Linear dependence on x

The following code reproduces the linear example from the lecture slides.

```
import numpy as np
import matplotlib.pyplot as plt
rng = np.random.default_rng(seed = 313)
x = np.linspace(0, 1, num = 100)
u = rng.uniform(-1, 1, 100)
y = 2 + 3 * x + u
plt.scatter(x, y)
plt.plot(x, 2 + 3 * x, color = "black")
plt.show()
```

Now, instead of the u above, use

```
u = rng.uniform(-1 - 3*x, 1 - 3*x, 100)
```

Now (i) plot the data using scatterplot, (ii) run your favourite kind of linear regression and add the line to the plot.

(b) Is any line possible?

Modify the code for u in (a) to make the regression line fitted by least squares and least absolute deviations close to $100 - \pi^2 x$. Don't touch $y = 2 + 3x + u$! (*Hint:* Look at the PowerPoint slides about making any line fit.)

(c) Large errors in the linear model

Now consider

```
u = rng.uniform(-33, 33, 100)
```

Recreate y, make a scatterplot, and impose the regression lines fitted by least squares and least absolute deviations. What's the estimated parameters? Comment and interpret.

(d) Heteroskedasticity in the linear model

Refering to the code in (a), modify u to be

```
u = rng.uniform(-exp(-3x), exp(-3x), 100)
```

Plot the data and fit least squares and least absolute deviations regression lines. Does the estimates of a and b appear to be affected much?

(e) Heteroskedasticity in the non-linear model

Continue to use the u in exercise (d), but simulate from the non-linear model

```
y = np.exp(3 * x ** 2) + u
```

Plot the data and fit least squares and least absolute deviations regression lines. Compare the lines and estimates to the case when

```
u = rng.uniform(-1, 1, 100)
```

Are the estimates affected much? Why or why not?

2 An explorative plotting function

We have made quite a lot of plots uptil now! Make a Python function that does all of this for you.

```
def plotreg(y, x, lad = True):
""" A scatterplot of y vs x, with the the least squares regression lines imposed.
If lad is True, then the least absolute deviation line is also added. """
```

Now you may use this function to exlore variants of residuals u and functional relationships. For instance, try out a plot with *periodic errors*,

```
y = 2 + 3*x + np.sin(x*5*np.pi) * rng.uniform(0, 1, 100)
plotreg(y, x, lad = True)
```

This function makes it easy to try out stuff – and I urge you to do it. Playing around is how you get good at data science.

3 Minimizing loss functions

We will work with the following loss functions:
Absolute value loss

$$d(y,x) = |x-y|.$$

Quadratic loss

$$d(y,x) = (x-y)^2.$$

Linex loss

$$d(y,x) = e^{y-x} - (y-x) - 1.$$

Welsch loss

$$d(y,x) = 1 - e^{-\frac{1}{2}(x-y)^2}.$$

Huber loss

$$d(y,x) = \begin{cases} |x-y| - \frac{1}{2} & \text{if } |x-y| \geq 1, \\ \frac{1}{2}(x-y)^2 & \text{if } |x-y| \leq 1. \end{cases}$$

All of these functions can be written as $f(y-x)$ for some function $f(z)$ of a single variable. For instance, for $d(x,y) = (x-y)^2$, one may write $f(z) = z^2$. This function will be equal to $f(z) = d(z,0)$.

(a) Implementation

Implement all the functions in Numpy. Make sure that they are vectorized, that is, `d(x,y)`should work when x and y are vectors.

(b) Plotting

Use Python to plot the functions $d(y,0)$. (*Hint:* See the Huber loss in the PowerPoint file for an example for what the figure should look like.)

(c) Verification

Verify that all of these ds are distance functions. That is, verify that $d(x,y) \geq 0$ and $d(x,y) = 0$ if and only if $x = y$. (*Hint:* Look at the plots in the previous exercise.)

(d) Interpreting

Plot the distances in the same window, and provide an interpretation for each of them relative to $|x-y|$. (*Hint:* For instance, $(x-y)^2$ is smaller than $|x-y|$ when $|x-y| < 1$, but quickly gets much larger than $|x-y|$ when $|x-y| > 1$. We could say that $(x-y)^2$ cares mostly about large absolute distances.)

4 An example

The rdatasets package contains about 1300 datasets, see the index here. (Sadly, not all of these are available, but most are!)

To load them, first import the data object.

```
from rdatasets import data
```

Now you may load a datset using the prototype

```
data(package, dataset)
```

For instance, to load the dataset boston from MASS, use

```
data("MASS", "boston")
```

If the dataset is in base R, such as the mtcars dataset, you do not need to specify the package.

(a)

Load the dataset "ducks" from "boot", documented here.

(b)

Run the two regression models with response "plumage" and covariate "behaviour". Is the relationship approximately linear?

5 The least squares solutions

The least squares estimator minimizes the function

$$f(a,b) = \sum_{i=1}^{n} (y_i - (a + bx_i))^2,$$

that is, the estimates are

$$(\hat{a}, \hat{b}) = \min_{a,b} \sum_{i=1}^{n} (y_i - (a + bx_i))^2.$$

From high school math we know that the optimas of $f(z)$ are found when $f'(z) = 0$ (or z is at the boundary of f's domain of definiton).

(a)

Assume that b is known. Use differentiation of $g(a) = f(a,b)$ to find the expression for $\hat{a}$. (*Hint:* Recall the chain rule, which implies that $\frac{d}{da}(f(a))^2 = [\frac{d}{da}f(a)]f(a)$.)

(b)

Plug the expression for $\hat{a}$ into $f(a,b)$ and define $h(b) = f(\hat{a}, b)$. Use differentiation on $h(b)$ to find the expression for $\hat{b}$.