Statistical Computation - Assignment 1

Philip Dahlqvist-Sjöberg

2nd of April 2020

1 Solving linear equations and matrix calculation

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{1}$$

$$\left(\boldsymbol{X^T X}\right)\hat{\boldsymbol{\beta}} = \boldsymbol{X^T Y} \tag{2}$$

a) The necessary matrices to compute and find least square $\hat{\boldsymbol{\beta}}$ in (2), are

$$\left(\boldsymbol{X^T X}\right) = \begin{bmatrix} 29 & 3729 & 50694 & 14 & 971 \\ 3729 & 538015 & 8076649 & 1707 & 123518 \\ 50694 & 8076649 & 181760034 & 13691 & 2080553 \\ 14 & 1707 & 13691 & 14 & 475 \\ 971 & 123518 & 2080553 & 475 & 48517 \end{bmatrix}, \ \boldsymbol{X^T Y} = \begin{bmatrix} 79740 \\ 11676140 \\ 181202430 \\ 37340 \\ 2620070 \end{bmatrix}.$$

To solve for $\hat{\boldsymbol{\beta}}$, the $\left(\boldsymbol{X^T X}\right)$ matrix need to be non-singular which means that the determinant is non-zero, i.e., inevitable. If we multiply a inverse matrix with itself, we will get the identity matrix as

$$\left(\boldsymbol{X^T X}\right)^{-1}\left(\boldsymbol{X^T X}\right) = \boldsymbol{I} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

b) Now we can calculate least square $\hat{\boldsymbol{\beta}}$ coefficients. We use the $\left(\boldsymbol{X^T T}\right)^{-1}$ as followed,

$$\left(\boldsymbol{X^T X}\right)^{-1}\left(\boldsymbol{X^T X}\right)\hat{\boldsymbol{\beta}} = \left(\boldsymbol{X^T X}\right)^{-1}\boldsymbol{X^T Y} = \boldsymbol{I}\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\beta}}$$

$$\hat{\boldsymbol{\beta}} = \begin{bmatrix} -104.3523 \\ 19.8272 \\ 0.1858 \\ 377.5787 \\ -6.0513 \end{bmatrix}.$$

c) Now, using the standard function, $lm()$, in R, we get the $\hat{\boldsymbol{\beta}}$ coefficients.

$$\hat{\boldsymbol{\beta}} = \begin{bmatrix} -104.3523 \\ 19.8272 \\ 0.1858 \\ 377.5787 \\ -6.0513 \end{bmatrix}.$$

d) After looping both method b) and c) 10000 times, we can see that b) took roughly 5.3% of the time it took for c), with the same result, i.e., b) is much faster in calculating the coefficients. However, with the lm() function, one gets more than just the coefficients, such as residuals etc.

Table 1: Local maximum from bisection and secant method.

	X value	Y value
Local maximum	0.9611	0.3468

2 Optimization of a general function of one variable

We have,

$$g(x) = \frac{log(x+1)}{x^{3/2}+1} \tag{3}$$

$$g'(x) = \frac{\frac{1}{x+1}x^{\frac{3}{2}} + 1 - \frac{3}{2}x^{\frac{3}{2}-1}log(x+1))}{(x^{\frac{3}{2}}+1)^2} \tag{4}$$

a) Looking at the left graph in Figure 1, we can assume that the (local) maximum of our function (3) would exist around one.

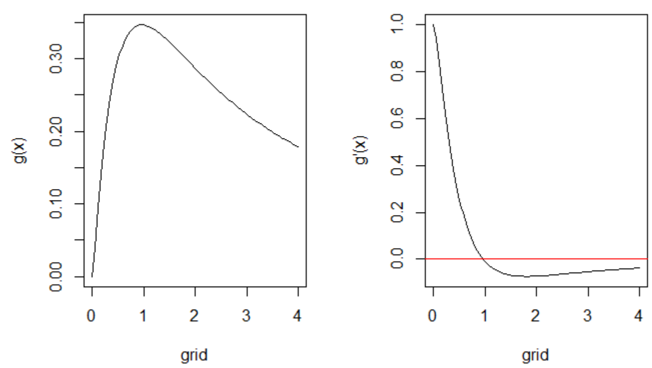

Figure 1: Plot of $g(x)$ and $g'(x)$ over the interval [0,4]

b) The derivative function (4), is shown in the right graph in Figure 1, with a red line at the zero derivative. Much like in a), we can assume that the (local) maximum is close to one.

$$cc_{bisection} = |b-a| \tag{5}$$

$$cc_{secant} = |x^{t+1} - x^t| \tag{6}$$

c) With the bisection method, we can find the local maximum in the given interval. See Table 1 for the given X value and Y value for the maximum. For stopping criteria, we use equation (5), and the algorithm is shown in the attached R-code.

d) From the secant method, we get the same results as with bisection. See Table 1 for the values. For stopping criteria, we use equation (6), and the algorithm is shown in the attached R-code.

3 Optimization in experimental design

$$\boldsymbol{X} = \begin{bmatrix} 1 & -1 & 1 & -1 \\ 1 & -a & a^2 & -a^3 \\ 1 & a & a^2 & a^3 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

a) We want to find $f(a) = det(\boldsymbol{X^T X})$, so first we need to find $\boldsymbol{X^T X}$.

$$\boldsymbol{X^T X} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & -a & a & 1 \\ 1 & a^2 & a^2 & 1 \\ -1 & -a^3 & a^3 & 1 \end{bmatrix} \begin{bmatrix} 1 & -1 & 1 & -1 \\ 1 & -a & a^2 & -a^3 \\ 1 & a & a^2 & a^3 \\ 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 4 & 0 & 2+2a^2 & 0 \\ 0 & 2+2a^2 & 0 & 2+2a^4 \\ 2+2a^2 & 0 & 2+2a^4 & 0 \\ 0 & 2+2a^4 & 0 & 2+2a^6 \end{bmatrix}.$$

To simplify calculation of determinant, we can change the matrix to a block diagonal matrix ($\boldsymbol{B}$),

$$\begin{bmatrix} 4 & 0 & 2+2a^2 & 0 \\ 0 & 2+2a^2 & 0 & 2+2a^4 \\ 2+2a^2 & 0 & 2+2a^4 & 0 \\ 0 & 2+2a^4 & 0 & 2+2a^6 \end{bmatrix} \xrightarrow{\text{Swap C2 and C3}} \begin{bmatrix} 4 & 2+2a^2 & 0 & 0 \\ 0 & 0 & 2+2a^2 & 2+2a^4 \\ 2+2a^2 & 2+2a^4 & 0 & 0 \\ 0 & 0 & 2+2a^4 & 2+2a^6 \end{bmatrix}$$

$$\begin{bmatrix} 4 & 2+2a^2 & 0 & 0 \\ 0 & 0 & 2+2a^2 & 2+2a^4 \\ 2+2a^2 & 2+2a^4 & 0 & 0 \\ 0 & 0 & 2+2a^4 & 2+2a^6 \end{bmatrix} \xrightarrow{\text{Swap R2 and R3}} \begin{bmatrix} 4 & 2+2a^2 & 0 & 0 \\ 2+2a^2 & 2+2a^4 & 0 & 0 \\ 0 & 0 & 2+2a^2 & 2+2a^4 \\ 0 & 0 & 2+2a^4 & 2+2a^6 \end{bmatrix}.$$

No we can calculate the determinant with,

$$\begin{aligned} det(\boldsymbol{X^T X}) = (-1)(-1)\boldsymbol{B} &= det \begin{bmatrix} 4 & 2+2a^2 \\ 2+2a^2 & 2+2a^4 \end{bmatrix} det \begin{bmatrix} 2+2a^2 & 2+2a^4 \\ 2+2a^4 & 2+2a^6 \end{bmatrix} \\ &= (4(2+2a^4) - (2+2a^2)^2) \times ((2+2a^2)(2+2a^6) - (2+2a^4)^2) \\ &= 16a^2(a^2-1)^4 \end{aligned}$$

Here, is a polynomial function of the determinant for the given matrix $\boldsymbol{X}$.

b) See attached R-code to see functions.

c) Looking in Figure 2, both functions in b) are shown. By looking at the graphs, they seem to be identical with each other, however we can further analyze with a difference plot.

In Figure 3 we can see a slight difference in the function values between the two functions given in b). However, the difference is to small to acknowledge. The end points where a is zero and one, indicates that the determinants are zero, which indicates that the $\boldsymbol{X^T X}$ matrix is singular for those values of a.

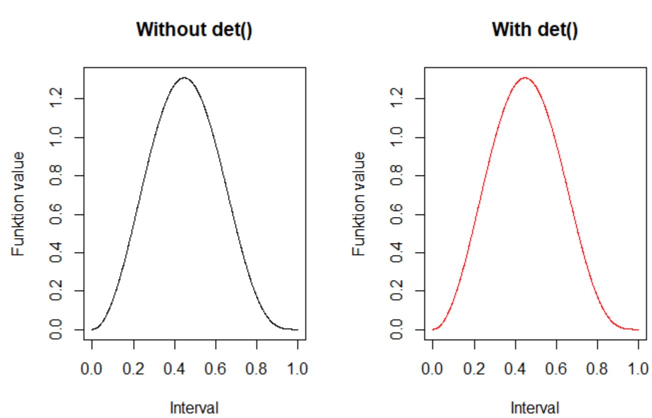

Figure 2: Function plots for both with and without det() function in R.

d) In Figure 4, the local maximum of the function, given in b), over the interval zero to one is shown, by using the optimize() function in R. The a-value for the maximum is 0.447, where the function value is 1.311. Since both functions in b) are almost equal, they also have the same maximum on the standard number of decimals in the optimize() function.

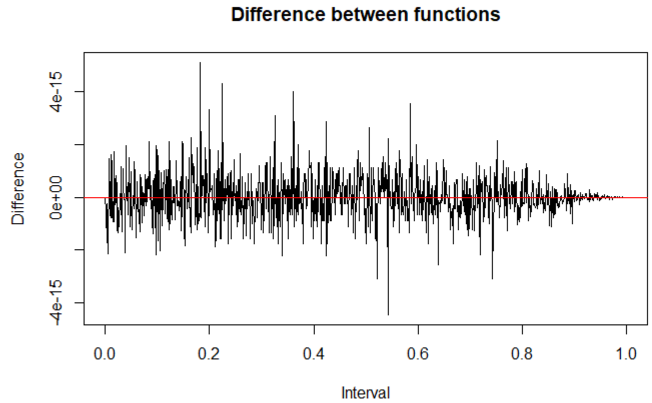

Figure 3: Difference between function values for both functions in b).

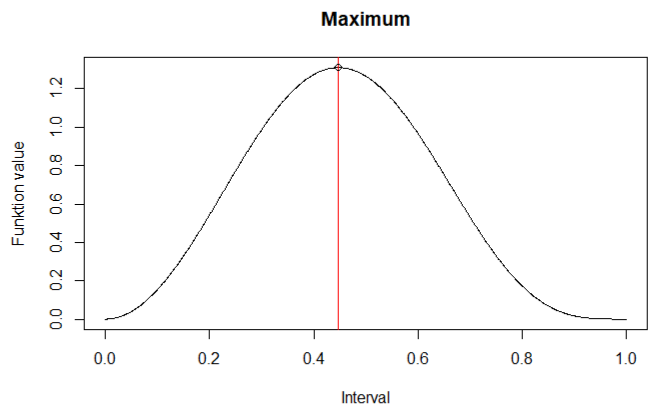

Figure 4: Maximum of the function given in b).

4 Maximization with Nelder-Mead algorithm

For the first plot, see Figure 5, we want to construct the ***next two*** iterations using the Nelder-Mead algorithm. We start with the black triangle, and follow the steps in the algorithm.

Iteration 1.
1. Use the given α values: $(\alpha_r, \alpha_e, \alpha_c, \alpha_s) = (1, 2, 0.5, 0.5)$
2. Find $x_{best} = (0.72, 0.31)$, $x_{bad} = (0.71, 0.26)$, $x_{worst} = (0.63, 0.19)$, based on their respective function values $g(x_{best}) = 150$, $g(x_{bad}) = 140$ and $g(x_{worst}) = 130$.
3. Find $c = (x_{best} + x_{bad})/2 = (0.715, 0.285)$
4. Find reflection point $x_r = c + \alpha_r(c - x_{worst}) = (0.80, 0.38)$ with $g(x_r) = 130$. Since $g(x_{worst}) \geq g(x_r)$ is true, perform inner contraction.
6,b. Find $x_c = c + \alpha_c(x_{worst} - c) = (0.6725, 0.2375)$ with $g(x_c) = 140$. Since $g(x_c) > g(x_{worst})$ is true, discard x_{worst} and accept x_c as new vertices and go to stopping.
8. Check convergence criteria, however, we will do one more iteration. And follow the same steps again.

Iteration 2.
Now we start with the red triangle, which is the result of the first iteration. Here, in step 2. there are two points with equal function value. Hence, the x_{bad} and x_{worst} have been chosen at random where $x_{bad} = x_c$ and $x_{worst} = x_{bad}$ from last iteration. Following the different steps we first find $x_r = (0.6825, 0.2875)$ with $g(x_r) = 150$. Since $g(x_{best}) \geq g(x_r) > g(x_{bad})$ is true, discard x_{worst} and accept x_r, i.e., stopping step.

Table 2: Two iteration Nelder-Mead algorithm from Figure 5.

	X-axis	Y-axis	g(X,Y)
x_{best}	0.7200	0.3100	150
x_{bad}	0.6825	0.2875	150
x_{worst}	0.6725	0.2375	140

Our final vertices are shown in Table 2, where our local maximum is the x_{best} value of 150.

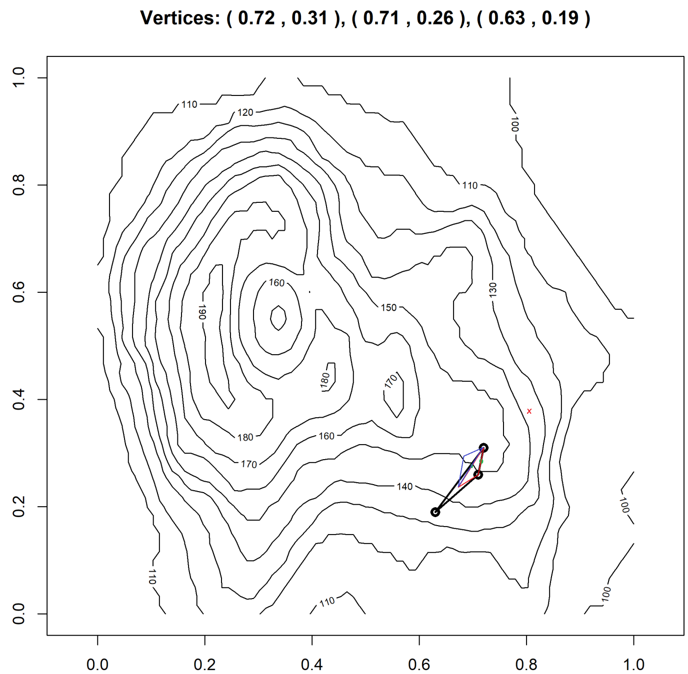

Figure 5: Contour plot 1, with two iterations of Nelder-Mead vertices. Red is the first iteration, and blue is the second iteration.

For the second plot, see Figure 6, we want to calculate ***next one*** iteration using the Nelder-Mead algorithm. As before, we start with the given black triangle, and find $x_{best} = (0.38, 0.79)$, $x_{bad} = (0.45, 0.66)$, $x_{worst} = (0.52, 0.72)$, based on their respective function values $g(x_{best}) = 160$, $g(x_{bad}) = 150$ and $g(x_{worst}) = 120$.

We find that $c = (0.415, 0.725)$ providing a $x_r = (0.31, 0.73)$ with $g(x_r) = 180$. Since $g(x_r) > g(x_{best})$ is true, perform expansion.
5. Find $x_e = c + \alpha_e(x_r - c) = (0.205, 0.735)$ with $g(x_e) = 160$. Since $g(x_e) > g(x_r)$ is not true, discard x_{worst} and accept x_r.

Our final vertices are found in Table 3, where x_{best} with value 180, is our local maximum after one iteration.

Table 3: One iteration Nelder-Mead algorithm from Figure 6

	X-axis	Y-axis	g(X,Y)
x_{best}	0.31	0.73	180
x_{bad}	0.38	0.79	160
x_{worst}	0.45	0.66	150

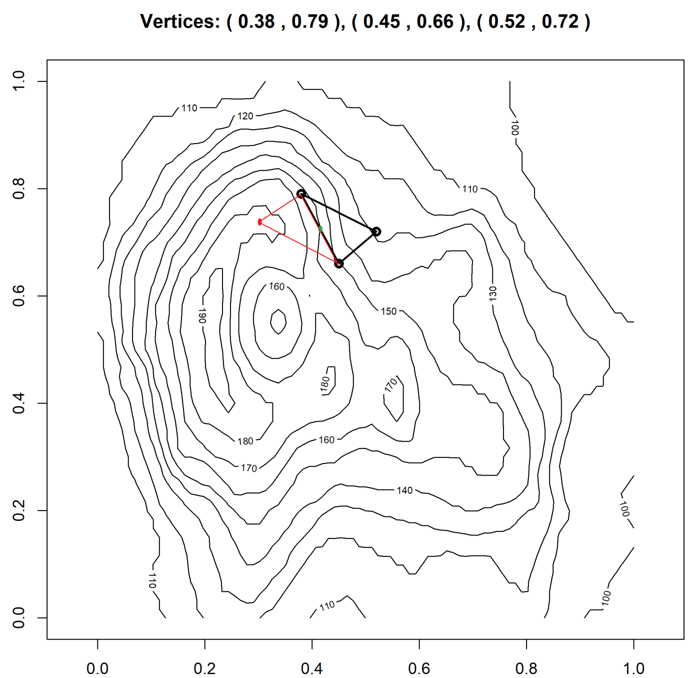

Figure 6: Contour plot 2, with one iterations of Nelder-Mead vertices. Red is the first iteration.