

## **Workshop 9**

### **Steepest descent algorithm**

#### **Exercise 1 One-dimensional optimisation**

The function minimised in the lecture demonstration was  $f(x) = 3x^4 + 5x^3 - 20x^2 + 8x + 10$ .

The code is given in the file `Workshop9.R`

- (a) Work through this example yourself.
- (b) Define a new function called `sinf` which corresponds to  $f(x) = \sin(x)$ .  
Copy the 1-dim minimisation code and adapt it for this problem. You will need to define  $f'(x)$
- (c) Try different starting values to see how this affects the found minimum.
- (d) Try adapting the step length  $s$  so that the convergence is quicker. For example try increasing the step size slightly with iteration number.
- (e) Try finding a local minimum for  $f(x) = e^{x^2+2x-4}$ . You will need to use the chain rule to calculate the derivative.

#### **Exercise 2 Two-dimensional optimisation**

The code for the second demonstration is also given in `Workshop9.R`

- (a) Work through this code, trying different starting points.
- (b) The outline code has been given for this part. The function to minimise is  $f(x) = x_1^2 + x_2^2 + 2x_1 - 4x_2 - 1$ . Calculate the partial derivatives and complete the code to find the approximate minimum.
- (c) Use calculus to obtain the exact location of the function minimum.

### Exercise 3 Linear model parameter optimisation

The code for exercise 3 defines a quadratic regression model of the form

$$y_i = \beta_1 + \beta_2 x_i + \beta_3 x_i^2 + e_i$$

Where  $e_i$  is the error term with a  $N(0, 0.2^2)$  distribution. There are  $n = 25$  observations.

The loss function for a linear model is

$$L(\beta) = \sum_{i=1}^{25} (y_i - \beta_1 - \beta_2 x_i - \beta_3 x_i^2)^2$$

Derive the partial derivatives. As a hint the third partial derivative is given for you

$$\frac{\partial L}{\partial \beta_3} = -2 \sum_{i=1}^{25} x_i^2 (y_i - \beta_1 - \beta_2 x_i - \beta_3 x_i^2)$$

Complete the code and confirm that the parameters converge to those used in the simulation.

## Homework exercises

### Exercise 4 Newton-Raphson method

**Finding a root of  $f$ .** A function can be approximated around a given point  $x_0$  using the first order Taylor series

$$f(x) \approx f(x_0) + f'(x_0)(x - x_0)$$

This is a linear polynomial, which uses the function, and the derivative value evaluated at the point  $x_0$

Rearrange the expression so that it is in the form

$$x \approx \tag{1}$$

Let  $x_1$  be a root of  $f(x)$ , then  $f(x_1) = 0$ . Using this and Equation (1), it follows that  $x_1 \approx x_0 - \frac{f(x_0)}{f'(x_0)}$ .  $x$  will not be an exact root, because of the approximation, but multiple iterations of the following formula should approach a root of  $f(x)$ .

This leads directly to the Newton-Raphson method for finding a root of  $f$ :

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

Let  $f(x) = 3x^2 - 4x + 5$  and  $x_0 = -2$ . Carry out 2 iterations of the N-R method to obtain  $x_2$ .

**Finding a minimum of  $f$ .** A very similar method can be used to minimise a function using the second order Taylor series.

$$f(x) \approx f(x_0) + f'(x_0)(x - x_0) + \frac{1}{2}f''(x_0)(x - x_0)^2$$

This is a quadratic polynomial, which uses the function, first and second derivative value evaluated at the point  $x_0$

Differentiate the right hand side to obtain

$$f'(x) \approx \quad (2)$$

Use an analogous argument to the above, to obtain an iterative procedure that finds a point where  $f'(x_i) = 0$

Show that when  $f(x) = 3x^2 - 4x + 5$  and  $x_0 = -2$ , the true minimum is found after just one iteration. The reason for this is that  $f$  is quadratic and the second order Taylor series is exact for quadratic functions.

## Exercise 5 Chain rule in back propagation

(a) Using the NN from Lecture 9, obtain the following partial derivatives

$$\frac{\partial R}{\partial w_{12}^{(2)}} \quad \text{and} \quad \frac{\partial R}{\partial b_1^{(2)}}$$

(b) Show that the derivative of the sigmoid function  $\sigma(v) = (1 + e^{-v})^{-1}$  is  $\sigma'(v) = e^{-v}(1 + e^{-v})^{-2}$  and that  $\sigma'(v) = \sigma(v)(1 - \sigma(v))$