COMP116 – Work Sheet Four – Solutions

Associated Module Learning Outcomes

1. basic understanding of the range of techniques used to analyse and reason about computational settings.
2. Ability to apply basic rules to differentiate commonly arising functions.

Question 1: Optimizing single argument functions

In the lectures (and pages 144-152 of the module textbook) a number of examples of finding a maximizing value for some single variable functions were discussed. This question concerns generalizing the structures analyzed in one of these examples.

The general form of the League of Legends example (pages 145, 148) and similar cases such as the Software Development study involves **four constant** values which we will denote using $\{k,\ r,\ \alpha,\ \beta\}$. The function to be optimized being

$$E(x) \ = \ \alpha x^k \ - \ \beta x^{k+r}$$

For example in the League of Legends analysis we had $k = r = 1$, $\alpha = (1800 + T)/1800$ (T being the initial number of tokens) and $\beta = 2.5/1800$.

1. Assuming that $k \geq 1$, $r \geq 0$, and both $\alpha > 0$, $\beta \geq 0$, derive the expression describing $E'(x)$. That is find
$$\frac{d}{dx} \ \alpha x^k \ - \ \beta x^{k+r}$$
2. Identify the critical point(s) of $E'(x)$ (as functions of the **constants** $\{k, r, \alpha, \beta\}$).
3. Determine whether these are minima or maxima by applying the second derivative test, having constructed $E''(x)$.

Answers:

1. $$E'(x) \;=\; k\alpha x^{k-1} \;-\; (k+r)\beta x^{k+r-1}$$

2. Critical points are those x for which $k\alpha x^{k-1} - (k+r)\beta x^{k+r-1} = 0$, that is such that
$$x^{k-1}(k\alpha - (k+r)\beta x^r) \;=\; 0$$
This, immediately, gives $x = 0$ as one critical point.
Rearranging $(k\alpha - (k+r)\beta x^r) = 0$ as
$$x^r \;=\; \frac{k\alpha}{(k+r)\beta}$$
identifies,
$$x \;=\; \left(\frac{k\alpha}{(k+r)\beta}\right)^{\frac{1}{r}}$$
as another.

3. $$E''(x) \;=\; k(k-1)\alpha x^{k-2} \;-\; (k+r)(k+r-1)\beta x^{k+r-2}$$
For the critical point, $x = 0$ (assuming $k > 1$), $E''(0) = 0$ and we are unable to deduce anything about the behaviour of $E(x)$.
For the critical point,
$$x \;=\; \left(\frac{k\alpha}{(k+r)\beta}\right)^{\frac{1}{r}}$$
$E''(x)$ is,
$$k(k-1)\alpha\left(\frac{k\alpha}{(k+r)\beta}\right)^{\frac{k-2}{r}} \;-\; (k+r)(k+r-1)\beta\left(\frac{k\alpha}{(k+r)\beta}\right)^{\frac{k+r-2}{r}}$$
Since $k-2 \leq k+r-2$ we can rearrange this expression into
$$\left(\frac{k\alpha}{(k+r)\beta}\right)^{\frac{k-2}{r}}\left(k(k-1)\alpha \;-\; (k+r)(k+r-1)\beta\left(\frac{k\alpha}{(k+r)\beta}\right)\right)$$
The term $(k\alpha)/((k+r)\beta)$ (under the premises of $\alpha, \beta, k, r \geq 0$) will always be non-negative. This leaves the three further cases

a.

$$k(k-1)\alpha = (k+r)(k+r-1)\beta\left(\frac{k\alpha}{(k+r)\beta}\right)$$

Since $E''(x) = 0$ no further deductions can be made.

b.

$$k(k-1)\alpha \;\; > \;\; (k+r)(k+r-1)\beta\left(\frac{k\alpha}{(k+r)\beta}\right)$$

In which case $E''(x) > 0$ and the critical point a (local) minimum.

c.

$$k(k-1)\alpha \;\; < \;\; (k+r)(k+r-1)\beta\left(\frac{k\alpha}{(k+r)\beta}\right)$$

In which case $E''(x) < 0$ and the critical point a (local) maximum.

Notice that whichever of these three holds is purely dependent on the values $\{k, r, \alpha, \beta\}$ which are **constants**.

Question 2: Optimizing multi argument functions

In addition to "standard" best value optimization cases (for example the large example described in detail of pages 169–173 of the module textbook) another very important application in CS where multi-variable optimization arises in the arena of Data Fitting. This is reviewed later in the module (see pages 337–343 of the module textbook) and, in much greater depth, in the Y2 module COMP229, Introduction to Data Science. Typically in such cases we are trying to find a "best function", $f : \mathbb{R} \to \mathbb{R}$ fitting a collection of paired observations: $\{< x_1, y_1 >, \ldots, < x_n, y_n >\}$ in such a way that the difference between $f(x_i)$ and y_i is minimized.[1] Two conditions that are exploited is that we may have a large sample of points (i.e. n can be made arbitarily large by, e.g. taking more samples) and, secondly, it is, often the case that the function, $f(x)$ sought is what is called *monotonic*: informally, $x > x'$ means that $f(x) > f(x')$.

Suppose, however, we have not been given an arbitrary number of points and have no reason to suspect that the "best-fitting function" is monotone. In the example of Figure 1 we have 3 observations and, in principle, would like to find a "good" polynomial function matching it. Notice that this function has at least one turning point (we do not know "what happens" for values less that x_1 or greater than x_3) and so might correspond to a quadratic polynomial. The classical method of matching k data points to a degree $k-1$ polynomial that fits the points **exactly**,

[1]There are a number of ways in which the concept "difference between $f(x_i)$ and y_i is minimized" may be defined, see pages 334-335 of recommended text.

i.e. for all k points $p(x_i) = y_i$ is Lagrange Interpolation (module textbook pages 350-351). This, however, has some rather subtle technical drawbacks.

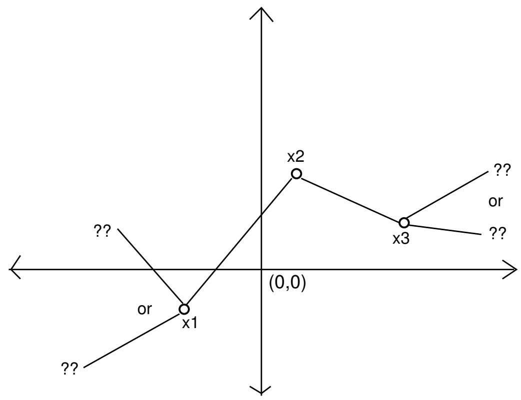

Figure 1: Best Curve Fit Environment

Suppose we want to find a "good" quadratic function matching our data. This might, e.g. involve finding **three** constants $< a, b, c >$ for which,

$$\sum_{i=1}^{3} ((ax_i^2 + bx_i + c) - y_i)^2$$

is *minimized*.

In total the minimization involves finding choices of a, b and c for which

$$F(a,b,c) \quad = \quad \sum_{i=1}^{3} (ax_i^2 + bx_i + c - y_i)^2$$

is minimized. Notice that $\{x_1, x_2, x_3, y_1, y_2, y_3\}$ are all **constant**: these are the given sample values.

a. What are the three first-order partial derivatives of $F(a, b, c)$? That is, what are

$$\left\{ \frac{\partial F}{\partial a}, \frac{\partial F}{\partial b}, \frac{\partial F}{\partial c} \right\}$$

b. Recalling that the 6 values $\{x_1, x_2, x_3, y_1, y_2, y_3\}$ are constants, find a way of rewriting your expressions for $\{\partial F/\partial a,\ \partial F/\partial b,\ \partial F/\partial c\}$.

[**Hint:** Write

$$\begin{aligned} W_x &= x_1 + x_2 + x_3 \\ W_y &= y_1 + y_2 + y_3 \\ W_{xy} &= x_1 y_1 + x_2 y_2 + x_3 y_3 \\ W_{xxy} &= x_1^2 y_1 + x_2^2 y_2 + x_3^2 y_3 \\ W_{xx} &= x_1^2 + x_2^2 + x_3^2 \\ W_{xxx} &= x_1^3 + x_2^3 + x_3^3 \\ W_{4x} &= x_1^4 + x_2^4 + x_3^4 \end{aligned}$$

and substitute these in your derived expressions.]

c. Using the shorthand notations W_x, W_y, etc. from part (b) **state** the 3 simultaneous eqautations involving a, b, and c whose solution prescribes the best fit quadratic for $\{x_1, y_1, x_2, y_2, x_3\}$.

d. (**Difficult, optional**) Outline how the system described in your answer to (c) can be rephrased as a calculation involving inverting a 3×3-matrix.

Answers

a.

$$\begin{aligned} \frac{\partial F}{\partial a} &= 2\sum_{i=1}^{3} x_i^2 (a x_i^2 + b x_i + c - y_i) \\ \frac{\partial F}{\partial b} &= 2\sum_{i=1}^{3} x_i (a x_i^2 + b x_i + c - y_i) \\ \frac{\partial F}{\partial c} &= 2\sum_{i=1}^{3} (a x_i^2 + b x_i + c - y_i) \end{aligned}$$

b.

$$\frac{\partial F}{\partial a} = 2a\sum_{i=1}^{3} x_i^4 + 2b\sum_{i=1}^{3} x_i^3 + 2c\sum_{i=1}^{3} x_i^2 - 2\sum_{i=1}^{3} x_i^2 y_i$$

Hence

$$\frac{\partial F}{\partial a} = 2aW_{4x} + 2bW_{xxx} + 2cW_{xx} - 2W_{xxy}$$

Similarly

$$\frac{\partial F}{\partial b} = 2a\sum_{i=1}^{3} x_i^3 + 2b\sum_{i=1}^{3} x_i^2 + 2c\sum_{i=1}^{3} x_i - 2\sum_{i=1}^{3} x_i y_i$$

Leading to

$$\frac{\partial F}{\partial b} \;=\; 2aW_{xxx} + 2bW_{xx} + 2cW_x - 2W_{xy}$$

Finally,

$$\frac{\partial F}{\partial c} \;=\; 2a\sum_{i=1}^{3} x_i^2 + 2b\sum_{i=1}^{3} x_i + 6c - 2\sum_{i=1}^{3} y_i$$

which gives

$$\frac{\partial F}{\partial c} \;=\; 2aW_{xx} + 2bW_x + 6c - 2W_y$$

c. We can cancel out the common constant factor 2 and so have the 3 equations

$$\begin{aligned} aW_{4x} + bW_{xxx} + cW_{xx} &= W_{xxy} \\ aW_{xxx} + bW_{xx} + cW_x &= W_{xy} \\ aW_{xx} + bW_x + 3c &= W_y \end{aligned}$$

d. The 3 equations in (c) may be described using the matrix-vector product

$$\begin{pmatrix} W_{4x} & W_{xxx} & W_{xx} \\ W_{xxx} & W_{xx} & W_x \\ W_{xx} & W_x & 3 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} \;=\; \begin{pmatrix} W_{xxy} \\ W_{xy} \\ W_y \end{pmatrix}$$

Letting $\mathbf{M}$ denote this 3×3 matrix and using $\mathbf{M}^{-1}$ for its inverse, we find the optimizing choice to be

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} \;=\; \mathbf{M}^{-1} \begin{pmatrix} W_{xxy} \\ W_{xy} \\ W_y \end{pmatrix}$$