KKT for local optima

$D = \{x \mid f_i(x) \le b_i\}$

$\min \quad c(x)$

x_0 is a local minima $\iff$ $I(x_0) = \{1, \dots, r\}$

$-\nabla c \in \text{cone}\{g_1, g_2, \dots, g_r\}$

Moreover, if x_0 is not a local minima then $\exists u \in Dir(x_0)$ s.t. $c(x_0 + \varepsilon u) < c(x_0)$ for a suitable $\varepsilon > 0$.

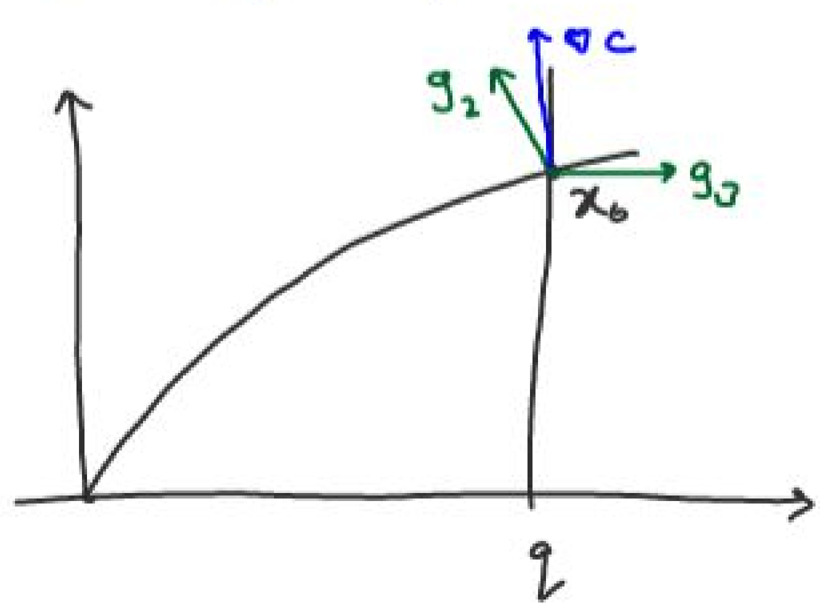
Recall the example of Beer Maker's Problem:

$y \le u(x) \iff y - u(x) \le 0$ — (2)

$-y \le 0$ — (1)

$x \le q$ — (3)

$c(x, y) = y - \lambda p x$



$I(x_0) = \{2, 3\}$

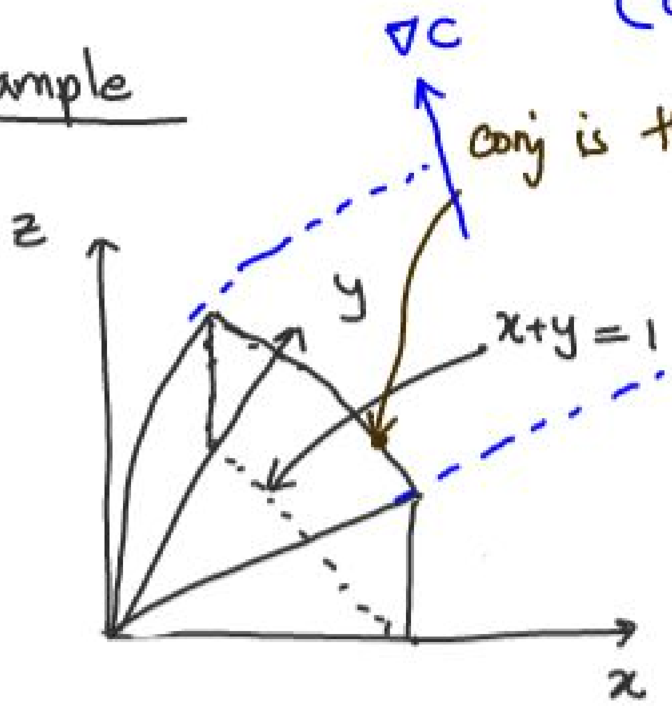
$\text{Cone}(x_0) =$

$\text{cone}\left\{ \left[-1, \frac{\partial u}{\partial x}\right], [0, 1] \right\}$

Another example

(i.e. without quota ∇c could have been much further.)

conj is that $x+y \le 1$ is tight



p : price of x

q : price of y

$$z - u(x,y) \le 0 \quad —①$$
$$-y \le 0 \quad —②$$
$$-x \le 0 \quad —③$$
$$x+y \le 1 \quad —④$$

quota.

$$\max \quad c(x,y,z) = z - \lambda p x - \lambda q y$$

$$\begin{array}{cccc} & 3 & 2 & 4 \\ \nabla c = & [\,1, & -\lambda p, & -\lambda q\,] \end{array}$$

$$I(x_0) = \{1,4\} \qquad \nabla f_4 = [0, 1, 1]$$

$$\nabla f_1 = \left[1, -\frac{\partial u}{\partial x}, -\frac{\partial u}{\partial y}\right]$$

$\exists \lambda_1, \lambda_2 \geq 0$ s.t.

$$[\lambda_1, \lambda_2] \begin{bmatrix} 0 & 1 & 1 \\ 1 & -\frac{\partial u}{\partial x} & -\frac{\partial u}{\partial y} \end{bmatrix} = [1, -\lambda p, -\lambda q]$$

we get $\lambda_2 = 1$. Say $x_0 = (a, b, c)$

Now the unknowns are λ_1, a, b, c and the 4 equations are

$$x+y \le 1, \ —Ⓐ \qquad z - u(x,y) \le 0, \ —Ⓑ$$

$$\lambda_1 - \frac{\partial u}{\partial x} = -\lambda p \quad —Ⓒ$$

$$\lambda_1 - \frac{\partial u}{\partial y} = -\lambda q \quad —Ⓓ$$

Ⓒ − Ⓓ

$$\frac{\partial u}{\partial y} - \frac{\partial u}{\partial x} = \lambda(q-p)$$

Suppose $u(x,y) = r\sqrt{x} + s\sqrt{y}$

Assuming q is given then one could compute a p for which revenues are maximized.

$$\frac{\partial u}{\partial x} = \frac{r}{2\sqrt{x}} , \quad \frac{\partial u}{\partial y} = \frac{s}{2\sqrt{y}}$$

$$\frac{s}{2\sqrt{y}} - \frac{r}{2\sqrt{x}} = \lambda(q-p)$$

Now we can solve for x, y, z etc.

Suppose the domain is unconstrained then we would simply look at $\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}$. But for a bounded domain, KKT allows us to convert solving a bounded domain problem into a problem of solving a bunch of equations.

In general, to compute opt $x_0 \in \mathbb{R}^n$

This is the Combinatorial step. → Step 1: Guess $I(x_0) = \{1, 2, \ldots, r\}$

Step 2: Solve equations ← if guess was wrong, we may get a "bad" set of λs.

Check λs thus obtained.

$I(x_0) = \{1, 2, \ldots, r\}$ this guess gives you 'r' equations.

Now look at $c \in \text{cone}\{g_1, \dots, g_r\}$

Note that this gives $(n-r)$ equations as follows:-

$$[\lambda_1, \dots, \lambda_r] \begin{array}{c} n \\ \left[\begin{array}{c:c} \boxed{r \times r} & \quad \end{array} \right] \end{array} \begin{array}{c} | \\ r \\ | \end{array}$$

$\lambda_1, \dots \lambda_r$ can be combuted using only an $r \times r$ submatrix

But when $[\lambda_1, \dots, \lambda_r]$ applied to the remaining columns, we get $n-r$ equations.

$$G = \begin{bmatrix} g_{11} & g_{12} & \dots & g_{1n} \\ \vdots & & & \\ g_{r1} & g_{r2} & & g_{rn} \end{bmatrix}$$

$$c = [c_1 \quad c_2 \qquad c_n]$$

$$[\lambda_1 \ \lambda_2 \dots \lambda_r]\, G = [c_1 \dots c_n]$$

Say $x_0 = (x_{01}, x_{02}, \dots, x_{0n})$

We now have $n+r$ vars : $\lambda_1, \lambda_2, \dots, \lambda_r,$
$x_{01}, x_{02}, \dots, x_{0n}.$

And $(n+r)$ equations : $I(x_0)$ gives r equations.

and $\sum_{i=1}^{r} \lambda_i g_{ij} = c_j \quad 1 \le j \le n$

Simpler versions of KKT

LP $\max\{\overset{1\times n}{c}x\}$

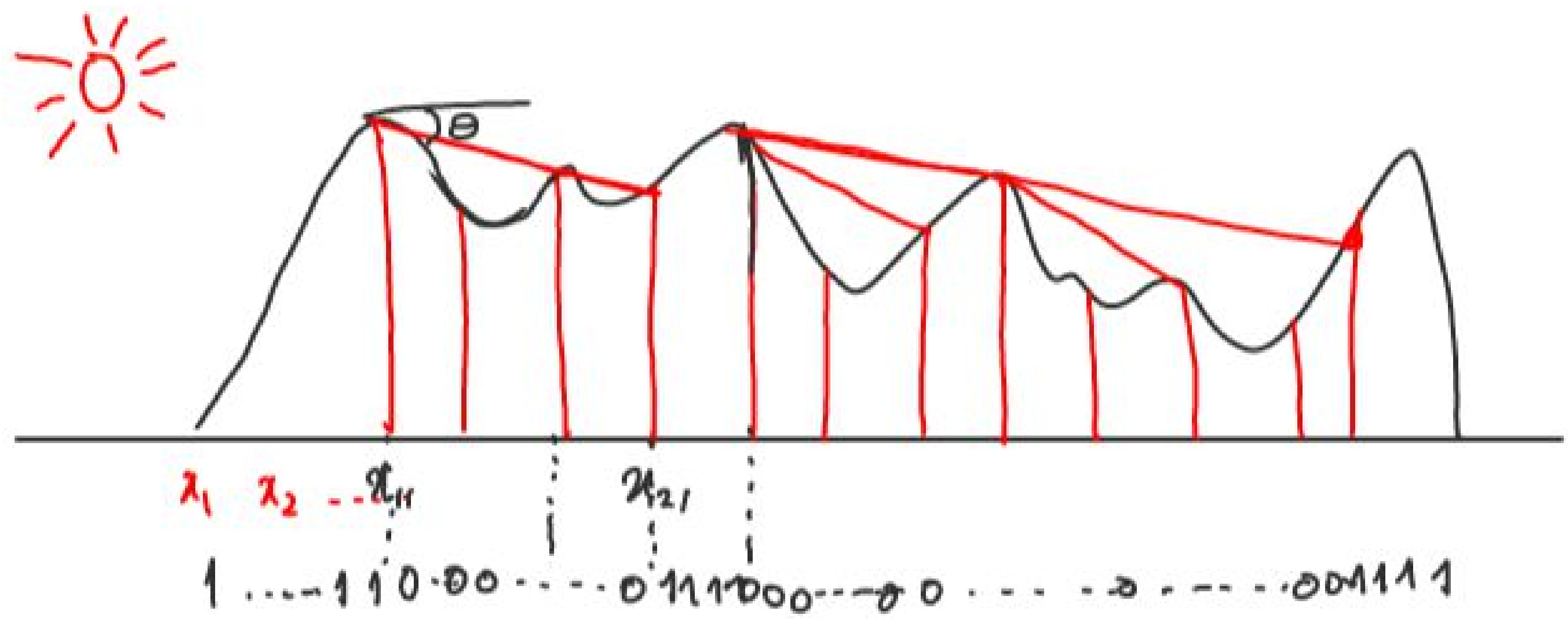
$$\underset{m\times n}{A}\,\underset{n\times 1}{x} \le \underset{m\times 1}{b}$$

QP: $\min \; Q(x)$

$$Ax \le b$$

$$Q(x) = x\underset{n\times n}{Q^T}x + cx$$

Example:

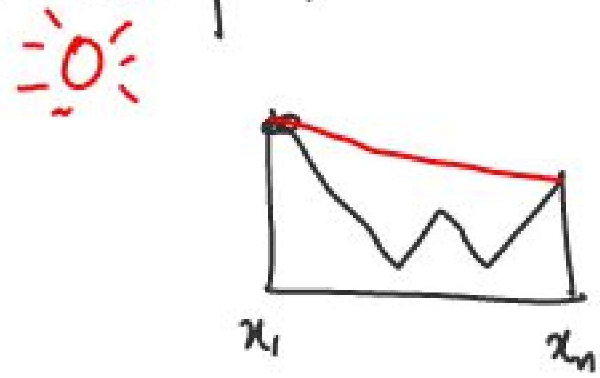


$$x_{21} \le x_{11} - \underset{\text{horizontal distance}}{40} \cdot \tan\theta$$

Problem defn: We wish to plot the 3D elevations given snapshots of the terrain from the bird's eye view. One method could be to use the sunlight & shadows. The above figure shows a method to genera inequalities using this method.

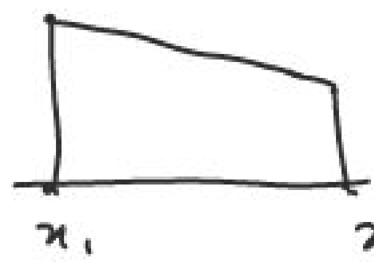
Such inequalities will give you many solutions.

For example, the following was the terrain

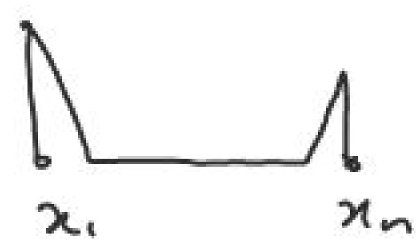


we only get elevation (x_1) > elevat'n (x_n) as our equat'n.

This can have many sol'ns. as follows:-



,



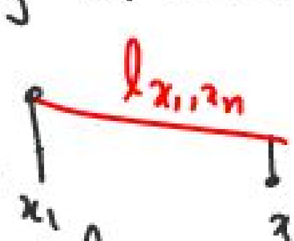
etc.

Now, the cost function can be designed so that we try to make the better among the above two guesses.

<u>Possible cost functions</u>:

① simply $\min\{\sum x_i\}$. This gives an LP

② Let l_{x_1,x_n} be the line joining x_1 & x_n

Min f'n could be min $\Big\{$ dist from l_{x_1,x_n}  $\Big\}$

This will give a QP.

③ $\sum_i (x_i - x_{i+1})^2$. This will also give a QP.