Data Science for Actuaries (ACT6100)

Arthur Charpentier

Rappels # 4.5 (Linear Programming)

automne 2020

https://github.com/freakonometrics/ACT6100/

Linear Programming

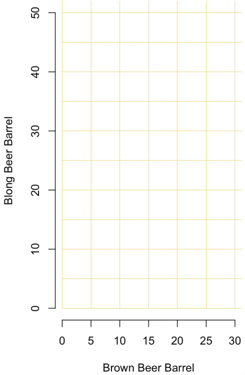
The beer problem: we want to produce beer, either blonde, or brown. We produce q_{blond} and q_{brown} barrels.

$\begin{cases} \text{hops}: & 5\text{kg} \\ \text{barley}: & 10\text{kg} \\ \text{price}: & 40\$ \end{cases}$

$\begin{cases} \text{hops}: & 2\text{kg} \\ \text{barley}: & 14\text{kg} \\ \text{price}: & 30\$ \end{cases}$

$\begin{cases} \text{hops}: & 100\text{kg} \\ \text{barley}: & 280\text{kg} \end{cases}$

We want to find q_{brown} (x) and q_{blond} (y) that is $\underbrace{\text{feaseable}}_{(1)}$ and $\underbrace{\text{maximize profit...}}_{(2)}$



Linear Programming

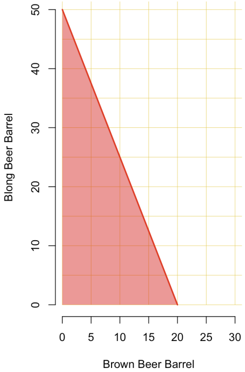
The beer problem: we want to produce beer, either blonde, or brown. We produce q_{blond} and q_{brown} barrels.

$$\begin{cases} \text{hops}: & 5\text{kg} \\ \text{barley}: & 10\text{kg} \\ \text{price}: & 40\$ \end{cases}$$

$$\begin{cases} \text{hops}: & 2\text{kg} \\ \text{barley}: & 14\text{kg} \\ \text{price}: & 30\$ \end{cases}$$

$$\begin{cases} \text{hops}: & 100\text{kg} \\ \text{barley}: & 280\text{kg} \end{cases}$$

(1) Our hops stock is 100kg,
it takes 5kg per barrel of blonde
it takes 2kg per barrel of brown
so $5x + 2y \leq 100$



Linear Programming

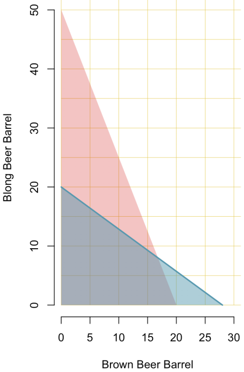
The beer problem: we want to produce beer, either blonde, or brown. We produce q_{blond} and q_{brown} barrels.

$$\begin{cases} \text{hops}: & 5\text{kg} \\ \text{barley}: & 10\text{kg} \\ \text{price}: & 40\$ \end{cases}$$

$$\begin{cases} \text{hops}: & 2\text{kg} \\ \text{barley}: & 14\text{kg} \\ \text{price}: & 30\$ \end{cases}$$

$$\begin{cases} \text{hops}: & 100\text{kg} \\ \text{barley}: & 280\text{kg} \end{cases}$$

(1) Our barley stock is 280kg,
it takes 10kg per barrel of blonde
it takes 14kg per barrel of brown
so $10x + 14y \leq 280$



Linear Programming

The beer problem: we want to produce beer, either blonde, or brown. We produce q_{blond} and q_{brown} barrels.

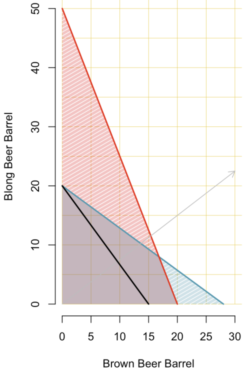
$$\left\{\begin{array}{ll}\text{hops :} & 5\text{kg}\\ \text{barley :} & 10\text{kg}\\ \text{price :} & 40\$\end{array}\right.$$

$$\left\{\begin{array}{ll}\text{hops :} & 2\text{kg}\\ \text{barley :} & 14\text{kg}\\ \text{price :} & 30\$\end{array}\right.$$

$$\left\{\begin{array}{ll}\text{hops :} & 100\text{kg}\\ \text{barley :} & 280\text{kg}\end{array}\right.$$

Under the two feasibility conditions,

(2) we want to maximize our profit
$\max\{40x + 30y\}$



Linear Programming

The beer problem: we want to produce beer, either blonde, or brown. We produce q_{blond} and q_{brown} barrels.

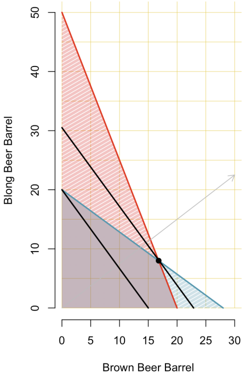
$$\begin{cases} \text{hops :} & 5\text{kg} \\ \text{barley :} & 10\text{kg} \\ \text{price :} & 40\$ \end{cases}$$

$$\begin{cases} \text{hops :} & 2\text{kg} \\ \text{barley :} & 14\text{kg} \\ \text{price :} & 30\$ \end{cases}$$

$$\begin{cases} \text{hops :} & 100\text{kg} \\ \text{barley :} & 280\text{kg} \end{cases}$$

Our problem is

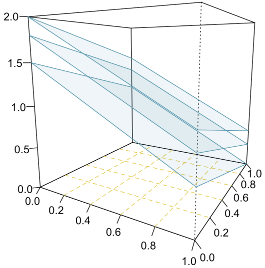
$$\begin{aligned} &\max \left\{40x + 30y\right\} \\ \text{s.t. } & 10x + 14y \leq 280 \\ & 5x + 2y \leq 100 \\ & x, y \geq 0 \end{aligned}$$



Linear Programming

Our problem is here

$$\begin{aligned} &\max\left\{\boldsymbol{c}^\top \boldsymbol{x}\right\} \\ \text{s.t. } &\boldsymbol{A}\boldsymbol{x} \le \boldsymbol{b} \\ &\boldsymbol{x} \ge 0 \end{aligned}$$

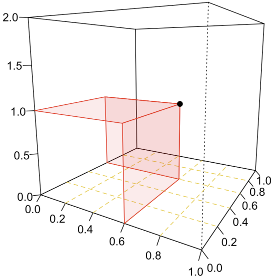


e.g.

$$\max\left\{ax + by + cz\right\}$$

subject to

$$\begin{cases} x \le \alpha \\ y \le \beta \\ z \le \gamma \end{cases}$$



The red volume is the set of feasible points (x, y, z)

Linear Programming

Our problem is here

$$\max\left\{\boldsymbol{c}^\top \boldsymbol{x}\right\}$$
$$\text{s.t. } \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}$$
$$\boldsymbol{x} \geq 0$$

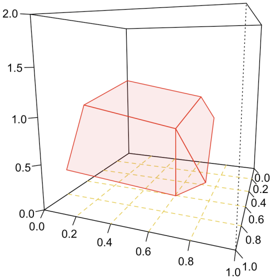
e.g., more generally

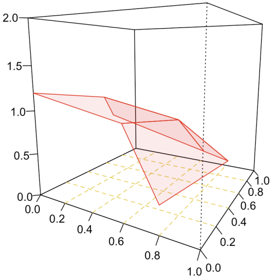
$$\max\left\{ax + by + cz\right\}$$

subject to

$$\left\{\begin{array}{c} \alpha_1 x + \beta_1 y + \gamma_1 z \leq \delta_1 \\ \alpha_2 x + \beta_2 y + \gamma_2 z \leq \delta_2 \\ \vdots \\ \alpha_k x + \beta_k y + \gamma_k z \leq \delta_k \end{array}\right.$$

which is a (convex) polyhedron.





Simplex

Our problem is here

$$\max\left\{\boldsymbol{c}^\top \boldsymbol{x}\right\}$$
$$\text{s.t. } \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}$$
$$\boldsymbol{x} \geq 0$$

First step: enlarge the parameter space, $10x_1 + 14x_2 \leq 280$ becomes $10x_1 + 14x_2 + u_1 = 280$ (so called slack variables)

$$\max\left\{40x_1 + 30x_2\right\}$$
$$\text{s.t. } 10x_1 + 14x_2 + u_1 = 280$$
$$2x_1 + 5x_2 + u_2 = 100$$
$$x_1, x_2, u_1, u_2 \geq 0$$

which is a problem of the general form

$$\max\left\{\boldsymbol{c}^\top \boldsymbol{x}\right\}$$
$$\text{s.t. } \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$$
$$\boldsymbol{x} \geq 0$$

Simplex Method

A linear program in standard form can be written

$$\begin{aligned} \max \left\{ \boldsymbol{c}^\top \boldsymbol{x} \right\} \\ \text{s.t. } \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \\ \boldsymbol{x} \geq 0 \end{aligned}$$

and it can be represented as a tableau (matrix) of the form

$$\begin{bmatrix} \mathbf{1} & -\boldsymbol{c}^\top & \mathbf{0} \\ \mathbf{0} & \boldsymbol{A} & \boldsymbol{b} \end{bmatrix}$$

Use some linear algebra, one can solve the optimization problem

Application to median computation

$\boldsymbol{y} = \{y_1, \cdots, y_n\}$, the median is a solution to $\displaystyle\min_{\mu}\left\{\sum_{i=1}^n |y_i - \mu|\right\}$.

Equivalently, we want to solve

$$\min_{\mu,\mathbf{a},\mathbf{b}}\left\{\sum_{i=1}^n a_i + b_i\right\}$$

with $a_i, b_i \geq 0$ and $y_i - \mu = a_i - b_i$, $\forall i = 1, \cdots, n$.

Heuristically, the idea is to write $y_i = \mu + \varepsilon_i$, and then define a_i's and b_i's so that $\varepsilon_i = a_i - b_i$ and $|\varepsilon_i| = a_i + b_i$, i.e.

$$a_i = (\varepsilon_i)_+ = \max\{0, \varepsilon_i\} = |\varepsilon| \cdot \mathbf{1}_{\varepsilon_i > 0}$$

and

$$b_i = (-\varepsilon_i)_+ = \max\{0, -\varepsilon_i\} = |\varepsilon| \cdot \mathbf{1}_{\varepsilon_i < 0}$$

Application to median computation

Thus, set $\boldsymbol{z} = \left(\mu^+;\mu^-;\boldsymbol{a},\boldsymbol{b}\right)^\top \in \mathbb{R}_+^{2n+2}$, and then write the constraint as $\boldsymbol{A}\boldsymbol{z} = \boldsymbol{b}$ with $\boldsymbol{b} = \boldsymbol{y}$ and $\boldsymbol{A} = \left[\mathbf{1}_n;-\mathbf{1}_n;\mathbb{I}_n;-\mathbb{I}_n\right]$
For the objective function $\boldsymbol{c} = \left(\mathbf{0},\mathbf{1}_n,-\mathbf{1}_n\right)^\top \in \mathbb{R}_+^{2n+2}$ and our program is $\min_{\boldsymbol{z}}\left\{\boldsymbol{c}^\top\boldsymbol{z}\right\}$ s.t. $\boldsymbol{A}\boldsymbol{z} = \boldsymbol{b}, \boldsymbol{z} \geq \mathbf{0}$.

```
> n = 101
> set.seed(1)
> y = rlnorm(n)
> median(y)
[1] 1.077415
>
> library(lpSolve)
```

```
> X = rep(1,n)
> A = cbind(X, -X, diag(n), -diag(n))
> b = y
> c = c(rep(0,2), rep(1,n), rep(1,n))
> r = lp("min", c, A, rep("=", n), b)
> head(r$solution,1)
[1] 1.077415
```

since the median is a solution to

$$\min_{\mu}\left\{\sum_{i=1}^{n}|y_i-\mu|\right\} = \min_{\mu}\left\{\sum_{i=1}^{n}\max\{(y_i-\mu),-(y_i-\mu)\}\right\}$$

Application to quantile computation

More generally, if the quantile of order τ is a solution of $\tau \in (0,1)$,

$$\min_{q} \left\{ \sum_{i=1}^{n} \max\{\tau(y_i - \mu), (1-\tau)(y_i - \mu)\} \right\}$$

The linear program is now

$$\min_{q^+, q^-, \mathbf{a}, \mathbf{b}} \left\{ \sum_{i=1}^{n} \tau a_i + (1-\tau) b_i \right\}$$

with $a_i, b_i, q^+, q^- \geq 0$ and $y_i = q^+ - q^- + a_i - b_i$, $\forall i = 1, \cdots, n$.

```
> tau = .3
> quantile(y,tau)
      30%
0.6741586
```

```
> c = c(rep(0,2), tau*rep(1,n), (1-
    tau)*rep(1,n))
> r = lp("min", c, A, rep("=",n), b)
> head(r$solution,1)
[1] 0.6741586
```

Application to quantile regression

In a regression, we use $\boldsymbol{x}_i^\top\boldsymbol{\beta}$ instead of $\boldsymbol{\mu}$. The linear program is

$$\min_{\boldsymbol{\beta}^+,\boldsymbol{\beta}^-,\mathbf{a},\mathbf{b}}\left\{\sum_{i=1}^n \tau a_i + (1-\tau)b_i\right\}$$

with $a_i, b_i \geq 0$ and $y_i = \boldsymbol{x}^\top[\boldsymbol{\beta}^+ - \boldsymbol{\beta}^-] + a_i - b_i$, $\forall i = 1, \cdots, n$ and $\beta_j^+, \beta_j^- \geq 0 \ \forall j = 0, \cdots, k$.

```
> n=nrow(Davis)
> X = cbind( 1, Davis$height)
> y =Davis$weight
> K = ncol(X)
> N = nrow(X)
> A = cbind(X,-X,diag(N),-diag(N))
> c =c(rep(0,2*ncol(X)),tau*rep(1,N),(1-tau)*rep(1,N))
> b = y
> r = lp("min",c,A,rep("=",N),b)
> beta = r$sol[1:K] -  r$sol[(1:K+K)]
> beta
[1] -110     1
```

Application to quantile regression

See

```
> library(quantreg)
> reg=rq(weight~height,data=Davis,tau=tau)
> summary(reg)

tau: [1] 0.3

Coefficients:
            coefficients    lower bd   upper bd
(Intercept) -110.00000   -135.94169  -70.39413
height         1.00000      0.73202    1.17531
```

to compare with

```
> reg=(lm(weight~height,data=Davis))
> cbind(reg$coefficient,confint(reg))
                               2.5 %      97.5 %
(Intercept) -130.910400 -153.643662 -108.177137
height         1.150092    1.016991    1.283192
```

Application to SVM (Support Vector Machine)

Points (x_i, y_i) with $y_i \in \{-1, +1\}$
We want, either

$$\vec{\boldsymbol{w}} \cdot \vec{\boldsymbol{x}}_i - b \geq +1, \text{ if } y_i = +1,$$

or

$$\vec{\boldsymbol{w}} \cdot \vec{\boldsymbol{x}}_i - b \leq -1, \text{ if } y_i = -1$$

i.e.

$$y_i(\vec{\boldsymbol{w}} \cdot \vec{\boldsymbol{x}}_i - b) \geq 1, \forall i = 1, \cdots, n.$$

Thus, solve

$$\min\left\{\boldsymbol{w}^\top \boldsymbol{w}\right\}$$
$$\text{s.t. } y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i - b) \geq 1, \forall i$$

... not linear but quadratic programming