

Cross Optimization

Freddy R. Drennan

November 25, 2017

Loading the necessary packages

```
library(tidyverse)
library(lpSolve)
```

Starting Data:

```
# Revenue variables
```

```
initial_data <-
  tibble(
    origin = c(
      'search',
      'display',
      'youtube',
      'gmail'
    ),
    revenue = c(
      225776,
      99292,
      4120,
      2472
    ),
    cost = c(
      8827,
      5172,
      833,
      423
    )
  ) %>%
  mutate(
    roi = (revenue - cost) / cost
  )
```

Constraints:

- Total budget for all digital ads will not exceed 15000
- Minimum: 3000 should be spent on Display ads
- Minimum: 400 should be spent on Social YouTube
- Display Ads and YouTube ads must exceed 60% of total budget
- YouTube Ads should not cost more than Gmail Ads.
- Minimum 1500 franchise application form.
- x_1 = Search Ads
- x_2 = Display Ads
- x_3 = YouTube Ads

- x_4 = Gmail Ads

We want to increase revenue, so the objective function should be maximized regarding the constraints.

```
A <-
  matrix(
    c(
      1,    1,    1,    1,
      0,    1,    0,    0,
      0,    0,    1,    0,
      -0.6, 0.4, 0.4, -0.6,
      0,    0,   -1,    1,
      .13, .09, .02, .03
    ),
    ncol = 4,
    byrow = TRUE
  )

b <-
  c(
    15000,
    3000,
    400,
    0,
    0,
    1500
  )

dir <-
  c(
    "<=",
    ">=",
    ">=",
    ">=",
    ">=",
    ">="
  )

obj <-
  c(
    initial_data$roi
  )
```

$$\max (x_1 + x_2 + x_3 + x_4)$$

```
cbind(A, dir, b) %>%
  noquote()
```

```
##           dir b
## [1,] 1      1  1  1  <= 15000
## [2,] 0      1  0  0  >= 3000
## [3,] 0      0  1  0  >= 400
## [4,] -0.6 0.4 0.4 -0.6 >= 0
## [5,] 0      0  -1  1  >= 0
## [6,] 0.13 0.09 0.02 0.03 >= 1500
```

```

s = lp("max", obj, A, dir, b)

solution <-
  s$solution

names(solution) <-
  initial_data %>%
  pull(origin) %>%
  paste0("Budget-", .)

solution

## Budget-search Budget-display Budget-youtube Budget-gmail
##          5600          8600          400          400

A %>% solution %>%
  round(2)

##      [,1]
## [1,] 15000
## [2,]  8600
## [3,]   400
## [4,]    0
## [5,]    0
## [6,] 1522

sol_tib <-
  tibble(
    x_1 = rep(NA, 6),
    x_2 = rep(NA, 6),
    x_3 = rep(NA, 6),
    x_4 = rep(NA, 6)
  )

for(i in 1:nrow(A)) {
  solz <-
    paste0(A[i,], " * ", solution, " + ") %>%
    noquote

  sol_tib[i, 1:4] <-
    solz
}

sol_tib <-
  sol_tib %>%
  mutate(
    equals      = rep("=", 6),
    sol         = A%>%solution %>% round(2),
    x_4         = sub("\\\\+", "", x_4) %>% str_trim,
    'constraints==>' = rep("...", 6),
    dir         = dir,
    b           = b
  ) %>%
  as.data.frame

```

```
sol_tib
```

```
##          x_1          x_2          x_3          x_4 equals    sol
## 1      1 * 5600 +      1 * 8600 +      1 * 400 +      1 * 400      = 15000
## 2      0 * 5600 +      1 * 8600 +      0 * 400 +      0 * 400      = 8600
## 3      0 * 5600 +      0 * 8600 +      1 * 400 +      0 * 400      = 400
## 4 -0.6 * 5600 +      0.4 * 8600 +      0.4 * 400 +     -0.6 * 400      = 0
## 5      0 * 5600 +      0 * 8600 +     -1 * 400 +      1 * 400      = 0
## 6 0.13 * 5600 +      0.09 * 8600 +      0.02 * 400 +      0.03 * 400      = 1522
## constraints==> dir      b
## 1          ...  <= 15000
## 2          ...  >= 3000
## 3          ...  >= 400
## 4          ...  >= 0
## 5          ...  >= 0
## 6          ...  >= 1500
```