Computing Porject: Gas Price Simulation

Benjamin Chu

2/10/2017

Problem motivation

In Taiwan, the government owns 70% of the gas distributed nationally. They publically announces any price changes in gas price at least 1 week in advance, but does not always adjust prices. As such, gas stations are often flooded with cars even when the following week's gas price is about to increase by ϵ amount. I wish to investigate whether waiting hours in line to save a couple of cents is necessary. In the long run, how much money can be saved if we deviate away from refueling whenever we need to?

Computing project description

My final project is going to be a combination of simulation, surveying, and data analysis on gas price. For the computing project I chose to implement (a major) part of the code needed for my final project simulation. My strategy for simulating the refueling process can be accomplished by modeling the total expense of two groups of people. A **control group** will refuel whenever their tank drop below a certain number. A **test group** will start considering refueling whenever their tank drop below a different number, and depending on whether they have information on the following week's gas price, choose a strategy that will (1) not run out of fuel at any point, and (2) minimize their spending on gas. Ultimately I can run these simulations 100 times each and compare their average spending as a function of time. Hopefully that will give me useful information to answer my master question.

Code and Data Description

Here I implemented the control group. Everyday they will lose `X-VALUE` of fuel, where currently `X-VALUE` is taken from a normal distribution with mean 10 and SD 5. They will blindly refuel whenever they drop below `Y-VALUE` = 20% of gas. All of these numbers (mean, SD, tank size, x-value…etc.) have not been obtained through survey and are all chosen based on my intuition. Then I run this simulation on two datasets CPC_data_short.xlsx and CPC_data.xlsx, representing Taiwan's gas price fluctuations of 2009 and from 1999~2017, respectively. Below are descriptions of major functions:

`control`: Simulates the refueling process of the control group. The input is a data set of interest and the fuel type (currently limited to 92, 95, or 98). The output is expense_list, which is a list element constructed the following way: everytime we refuel, we will increase the total_expense by some amount, and the new total expense will be added to the expense_list. This way, we could keep track of the total money we spent on gas as a function of time.

`lose-fuel`: Everyday people will lose x% of fuels, where $x \sim N(\mu, \sigma^2)$. If $x < 0$, we set it equal to zero (people can't gain fuel other than refueling).

```r
library(readxl) # parsing excel
library(lubridate) # dates/months/years addition made easy
```

```
##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##     date
```

```
mydata <- read_excel("CPC_data_short.xlsx")

X_VALUE = 10
Y_VALUE = 30
Z_VALUE = 40

assign_fuel_type <- function(fuel_type) { # identity fuel type (92/95/98)
  dummy_var = 0
  if (fuel_type == 92) {dummy_var = 2}
  else if (fuel_type == 95) {dummy_var = 3}
  else if (fuel_type == 98) {dummy_var = 4}
  else {
    print("please enter either 92, 95, or 98")
    return()
  }
  return(dummy_var)
}

lose_fuel <- function() { # everyday lose fuel ~ normal
  SD = 5
  lost = rnorm(1, X_VALUE, SD) #generate x ~ N(mean, SD)

  if (lost < 0) {
    lost = 0
  }
  return(lost)
}

control <- function(fuel_type, table) {
  # fuel_type = 92, 95, or 98

  which_column = assign_fuel_type(fuel_type)
  my_price_list = table[[which_column]]
  my_date_list = table[[1]]

  current_date = ymd(my_date_list[1])
  current_price = my_price_list[1]
  my_price_list <- my_price_list[-1] #delete first element
  my_date_list <- my_date_list[-1] #delete first element
  last_day = ymd(my_date_list[length(my_date_list)])

  total_expense = 0 # keep track of total money spent
  expense_list <- list(total_expense) # keeps track of marginal expense
  current_fuel = 100

  while (current_date != last_day) {
    if(current_fuel < Y_VALUE) { # refuel when necessary
      total_expense = total_expense + 57.0 * current_price #57 liters ~ 15 gallons ~ approx gas tank si
      expense_list <- append(expense_list, total_expense / 30.0)
      current_fuel = 100
    }

    current_fuel = current_fuel - lose_fuel()
```

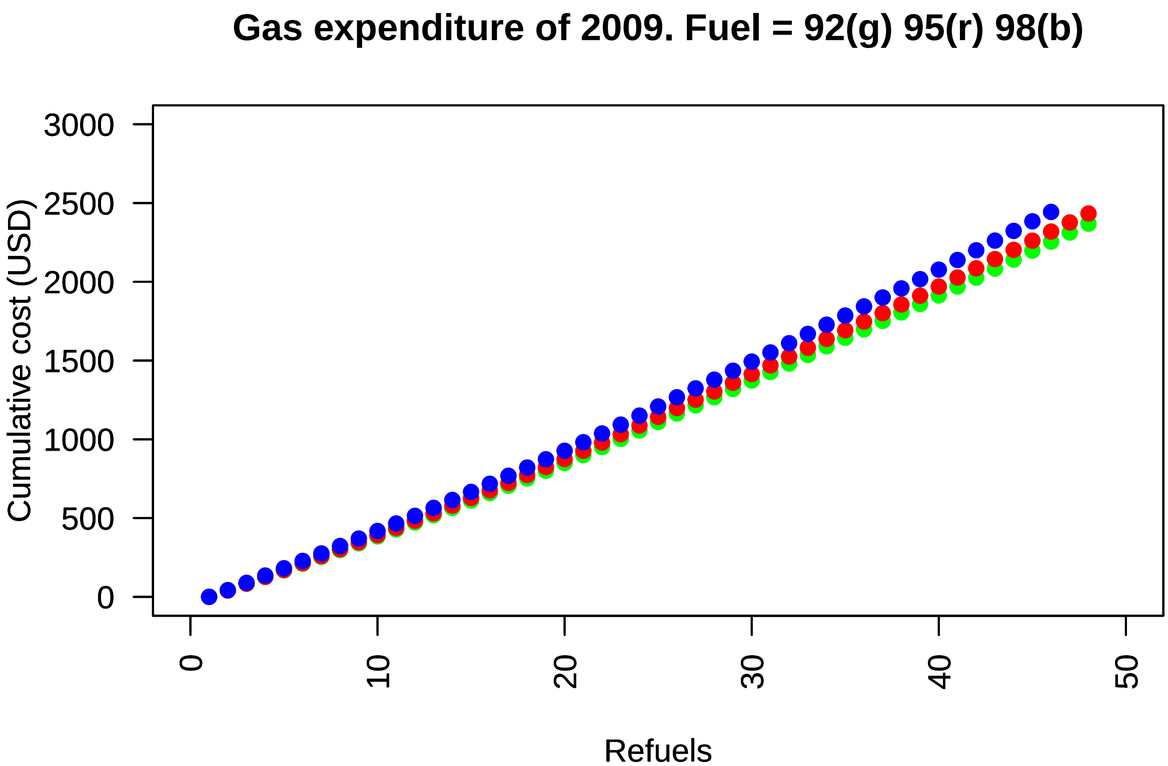
```
    current_date = current_date + ddays(1)

    #check if today needs price adjustments. Remember to watch out for NA.
    if (current_date == ymd(my_date_list[1])) {
      my_date_list <- my_date_list[-1]
      if (!is.na(my_price_list[1])) {
        current_price = my_price_list[1]
      }
      my_price_list <- my_price_list[-1]
    }
  }
  return(expense_list)
}


test_1 = control(92, mydata)
number_1 = length(test_1)
count_1 = 1:number_1
plot(count_1, test_1, col="green", las = 2, xlim = c(0, 50), ylim = c(0, 3000), main = "Gas expenditure

test_2 = control(95, mydata)
number_2 = length(test_2)
count_2 = 1:number_2
par(new=TRUE) #plots the second graph on top of first graph
plot(count_2, test_2, col="red", las = 2, xlim = c(0, 50), ylim = c(0, 3000), xlab="Refuels", ylab="Cumu

test_3 = control(98, mydata)
number_3 = length(test_3)
count_3 = 1:number_3
par(new=TRUE) #plots the second graph on top of first graph
plot(count_3, test_3, col="blue", las = 2, xlim = c(0, 50), ylim = c(0, 3000), xlab="Refuels", ylab="Cur
```



Conclusion

I ran my code on the gas price of 2009 because the gas price increased by about 50% by the end of the year. Since the control group have no strategy in the refueling process, we expect their gas expenditure to drift along with the market price. Here our plot was mildly able to capture that upward concavity. Also, more expensive gases incrued a higher cost, around 200 USD more if one chose 98 as opposed to 92. Every run could have a different number of points, because a different number of refueling could have taken place. In particular this implies that it is not always the case that choosing the cheapest gas will end up paying the least, because by chance alone you might have to refuel more often and end up paying more.

I think the idea that fuels are lost according to a normal distribution is neither realistic (because one cannot lose a negative amount of fuel) nor does it exhibit any “dramatic effect” potentially sensitive to the market price. For the final project, perhaps I should look into other distributions.