Choice Model

Ziyi(Sylvia) Ye

3/10/2018

Suppose you are in charge of MacGruber, an on-demand car service where drivers dress up like the popular character MacGruber. You offer two services to customers, solo and pool. Solo service means that a car comes to pick you up and drops you off without any interruptions. Pool service means that a car may have other passengers, and may need to stop to drop off and receive passengers.

Let T_n denote the expected travel time in minutes for user n if they were to drive themselves (or if they used solo service). Clearly, we expect users to pay more the larger T_n is (i.e., you should be willing to pay more if you know the trip will take longer). Let Wsn and Wpn denote the waiting time for user n when using solo (s) and pool (p) services respectively. The waiting time for solo service is $W_{s_n} = 2$ minutes on average. The waiting for pool service is $W_{p_n} = 3 + 0.2T_n$ minutes, which includes the extra time needed due to pooling (the travel time is 20% longer on average).

MacGruber also knows user n's relative wealth by estimating the average apartment monthly rent in the building that the user has designated as 'Home', which they call An. When users open up the MacGruber app, they must choose between solo, pool, and nothing (o). The price charged for the services are P_{s_n} and P_{p_n}, respectively. Note that you must always have $P_{p_n} \leq 0.8P_{s_n}$ so users do not hate you.

The expected utility from nothing is $V_{o_n} = 0$. The expected utility from solo is$ V_{s_n} = 4+0.9T_n-0.5W_{s_n}+0.003A_n-P_{s_n}$ and frompool is $V_{p_n} = 3 + 0.9T_n - 0.5W_{p_n} + 0.001A_n - P_{p_n}$. Finally, suppose the noise around the expected utility is zero-mean and distributed according to a Gumbel with scale parameter 2.

a) Suppose a user i arrives with Ti = 10 minutes and Ai = 2000. If you charge Psi = 10 and Ppi = 8, what is the probability of user i choosing each option? How much revenue do you make in expectation?

```
Vs = function(user_A, user_T, price_s){
  3 + 0.9*user_T + 0.003*user_A - price_s
}

Vp = function(user_A, user_T, price_p){
  Wp = 3 + 0.2*user_T
  3 + 0.9*user_T - 0.5*Wp + 0.001*user_A - price_p
}

ChoiceModel = function(user_A, user_T, price_s, price_p, theta){
  V_s = Vs(user_A, user_T, price_s)
  V_p = Vp(user_A, user_T, price_p)
  P_s = exp(V_s/theta) / sum(c(exp(V_s/theta), exp(V_p/theta), 1))
  P_p = exp(V_p/theta) / sum(c(exp(V_s/theta), exp(V_p/theta), 1))
  P_o = 1 - P_s - P_p
  expected_revenue = P_s * price_s + P_p * price_p
  return(list(prob_of_solo = P_s, prob_of_pool = P_p,
              prob_of_nothing = P_o, expected_revenue = expected_revenue))
}

user_i_A = 2000
```

```r
user_i_T = 10
price_s = 10
price_p = 8

ChoiceModel(user_i_A, user_i_T, price_s, price_p, 2)
```

```
## $prob_of_solo
## [1] 0.8899055
##
## $prob_of_pool
## [1] 0.09379535
##
## $prob_of_nothing
## [1] 0.01629919
##
## $expected_revenue
## [1] 9.649417
```

(b) What would happen to part (a) if you could speed up your pool service by 1 minute? Why does your answer make sense in comparison to (a)?

```r
Vp_new = function(user_A, user_T, price_p){
  Wp = 2 + 0.2*user_T
  3 + 0.9*user_T - 0.5*Wp + 0.001*user_A - price_p
}

ChoiceModel_new = function(user_A, user_T, price_s, price_p, theta){
  V_s = Vs(user_A, user_T, price_s)
  V_p = Vp_new(user_A, user_T, price_p)
  P_s = exp(V_s/theta) / sum(c(exp(V_s/theta), exp(V_p/theta), 1))
  P_p = exp(V_p/theta) / sum(c(exp(V_s/theta), exp(V_p/theta), 1))
  P_o = 1 - P_s - P_p
  expected_revenue = P_s * price_s + P_p * price_p
  return(list(prob_of_solo = P_s, prob_of_pool = P_p,
              prob_of_nothing = P_o, expected_revenue = expected_revenue))
}

ChoiceModel_new(user_i_A, user_i_T, price_s, price_p, 2)
```

```
## $prob_of_solo
## [1] 0.8668133
##
## $prob_of_pool
## [1] 0.1173104
##
## $prob_of_nothing
## [1] 0.01587624
##
## $expected_revenue
## [1] 9.606617
```

Compared to part (a), the probability of user i choosing pool service increases due to the decreasing time of pool service. And in the meantime the probability of choosing solo and doing nothing decreases. The expected revenue also decreases.

(c) Suppose that you wanted to keep the pool price the same for the user from (a), but you want to find the optimal solo price. What choice of Psi maximizes your expected revenue from i, and what is the revenue?

```r
f <- function(price_s){
  -ChoiceModel(2000, 10, price_s, 8, 2)$expected_revenue
}
ans <- nlm(f, 8)

# Choice of Psi
ans$estimate
```

```
## [1] 12.80072
```

```r
# Maximum_expected_revenue
-ans$minimum
```

```
## [1] 10.80072
```

We should set Psi as 12.8 to maximize my expected revenue. And it is 10.8.

(d) Suppose you want to optimize Psi and Ppi for the user in i. What are the optimal prices, and the optimal expected revenue?

```r
f2 <- function(price, A_i, T_i){
  # To satisfy Ppi <= 0.8*Psi
  # Ppi = price[1]; Psi = price[1]/0.8 + price[2].
  # price[1] > 0, price[2] > 0
  -ChoiceModel(A_i, T_i, price[1]/0.8 + price[2], price[1], 2)$expected_revenue
}
ans2 <- optim(c(0,0), f2, A_i = 2000, T_i = 10, lower = c(0,0), method="L-BFGS-B")

# Choice of Psi
ans2$par[1]/0.8 + ans2$par[2]
```

```
## [1] 14.60702
```

```r
# Choice of Ppi
ans2$par[1]
```

```
## [1] 11.68561
```

```r
# Maximum_expected_revenue
-ans2$value
```

```
## [1] 12.26258
```

The optimal price of solo service should be 14.6 and price of pool service should be 11.7. The optimal expected revenue is about 12.26.

(e) Suppose that the distribution of all travel times T i is Uniform[7, 20] and the apartment prices Ai are distributed Normal(1500,500). What is the expected revenue of a policy that sets Psi = Ti and Ppi = 0.80Psi?

```
Sim_One <- function(){
  T_i <- runif(1, 7, 20)
  A_i <- rnorm(1, 1500, sqrt(500))
  expected_revenue <- ChoiceModel(A_i, T_i, T_i, 0.8*T_i, 2)$expected_revenue
  return(expected_revenue)
}

mean(replicate(100000, Sim_One()))
```

```
## [1] 12.49799
```

Using the condition given in (e), the expected revenue is about 12.5.

(f) The discount from (e) was 20%. What is the optimal discount?

```
f3 <- function(discount, A_i, T_i, Ps){
  # To satisfy Ppi <= 0.8*Psi
  # Ppi = price[1]; Psi = price[1]/0.8 + price[2].
  # price[1] > 0, price[2] > 0
  -ChoiceModel(A_i, T_i, Ps, Ps*(1 - discount), 2)$expected_revenue
}

Sim_One_Opt <- function(){
  T_i <- runif(1, 7, 20)
  A_i <- rnorm(1, 1500, sqrt(500))
  Ps <- T_i
  discount <- optim(0.2, f3, A_i = A_i, T_i = T_i, Ps = Ps,
                    lower = 0.2, upper = 1, method="L-BFGS-B")$par
  Pp <- Ps*(1 - discount)
  expected_revenue <- ChoiceModel(A_i, T_i, Ps, Pp, 2)$expected_revenue
  return(c(Expected_revenue = expected_revenue, Discount = discount))
}

ans <- replicate(100000, Sim_One_Opt())

# Choice of Discount
mean(ans[2,])
```

```
## [1] 0.2
```

```
# Maximum_expected_revenue
mean(ans[1,])
```

```
## [1] 12.49097
```

The average optimal discount is 0.2. So the expected revenue has no significant change.

(g) Assume the prices for each customer can be optimized, and do not need to follow the formula in (e). In other words, for each user we optimize the prices like in (d). What is the optimal expected revenue from a random customer?

```
Sim_One_Opt2 <- function(){
  T_i <- runif(1, 7, 20)
  A_i <- rnorm(1, 1500, sqrt(500))
  p <- optim(c(0,0), f2, A_i = A_i, T_i = T_i, lower = c(0,0), method="L-BFGS-B")
  Ps <- ans2$par[1]/0.8 + ans2$par[2]
  Pp <- ans2$par[1]
  expected_revenue <- ChoiceModel(A_i, T_i, Ps, Pp, 2)$expected_revenue
  return(c(expected_revenue,T_i, Ps))
}

mean(replicate(100000, Sim_One_Opt2()))
```

```
## [1] 13.45091
```

The optimal expected revenue from a random customer is about 13.4

(h) Now suppose we do not want to discriminate based on apartment price, so we justs set An = $1500 for every user. Every customer can still be offered optimized prices based on travel time. What is the optimal expected revenue from a random customer? Why does your answer make sense in comparison to (g)?

```
Sim_One_Opt3 <- function(){
  T_i <- runif(1, 7, 20)
  A_i <- 1500
  p <- optim(c(0,0), f2, A_i = A_i, T_i = T_i, lower = c(0,0), method="L-BFGS-B")
  Ps <- ans2$par[1]/0.8 + ans2$par[2]
  Pp <- ans2$par[1]
  expected_revenue <- ChoiceModel(A_i, T_i, Ps, Pp, 2)$expected_revenue
  return(c(expected_revenue,T_i, Ps))
}

mean(replicate(100000, Sim_One_Opt3()))
```

```
## [1] 13.43136
```

The optimal expected revenue from a random customer is still about 13.4. Compared to (g), we can conclude that if we correctly choose the solo and pool price based on the customers' partment prices and customers' traveling time, our final optimal expected revenue won't be significantly affected by the customers' partment prices. In another word, no matter what the customers' partment prices is, we can expect roughly the same revenue once we set the optimal service prices.

(i) Suppose you want to model your competition, Dryft, which offers similar services as you. Specifically, how would you change the choice model to be able to answer the previous questions? What other things would you need to consider?

If there is a competition, which means people has other choices, these similar services should be introduced into our choice model. So the probability of pool and solo service of my company to be chosen will change (decrease).

We need to learn more about other company's service and their utility for people in order to estimate the optimal price we should take. Then we can calculate the estimate revenue as below

$$Prob(solo) = \frac{exp(\frac{V_s}{\theta})}{\sum_i^{|choices|} \frac{V_i}{\theta}}, i \in choices$$

$$Prob(pool) = \frac{exp(\frac{V_p}{\theta})}{\sum_i^{|choices|} \frac{V_i}{\theta}}, i \in choices$$

$$estimated\_revenue = P_{solo}^n * Prob(solo) + P_{pool}^n * Prob(pool)$$

Then we can use the same way to estimate the optimal prices of solo and pool services.