Group Assignment 4 - Creative Gaming

Section 51
Gaurav Agrawal, Ajitesh Abhishek, Tarun Joshi

```
   checking for file 'C:\Users\a1aji\AppData\Local\Temp\RtmpOMWxfW\remotes25b46a72602b\fzettelmeyer-mktg

v  checking for file 'C:\Users\a1aji\AppData\Local\Temp\RtmpOMWxfW\remotes25b46a72602b\fzettelmeyer-mktg



-  preparing 'mktg482':
   checking DESCRIPTION meta-information ...

   checking DESCRIPTION meta-information ...

v  checking DESCRIPTION meta-information



-  checking for LF line-endings in source and make files and shell scripts


-  checking for empty or unneeded directories



-  building 'mktg482_0.0.3.0.tar.gz'
```

Read in the data:

```
# use load("filename.Rdata") for .Rdata files
data = load("PentathlonTargeting.Rdata")
```

##Question 1: Step 1

```
data.nptb.train <- pent %>%
filter(training==1)
```

```
data.nptb.test <- pent %>%
filter(training==0)
```

```
logit.endurance <- glm(buyer ~ age + female + income + children + freq_endurance + freq_strength + freq

data.nptb.test <- data.nptb.test %>%
mutate(pr.endurance = predict(logit.endurance, newdata = data.nptb.test, type = "response"))
```

```
logit.strength <- glm(buyer ~ age + female + income + children + freq_endurance + freq_strength + freq_

data.nptb.test <- data.nptb.test %>%
mutate(pr.strength = predict(logit.strength, newdata = data.nptb.test, type = "response"))
```

```
logit.water <- glm(buyer ~ age + female + income + children + freq_endurance + freq_strength + freq_wat

data.nptb.test <- data.nptb.test %>%
mutate(pr.water = predict(logit.water, newdata = data.nptb.test, type = "response"))
```

```
logit.team <- glm(buyer ~ age + female + income + children + freq_endurance + freq_strength + freq_wate

data.nptb.test <- data.nptb.test %>%
mutate(pr.team = predict(logit.team, newdata = data.nptb.test, type = "response"))
```

```
logit.backcountry <- glm(buyer ~ age + female + income + children + freq_endurance + freq_strength + fr

data.nptb.test <- data.nptb.test %>%
mutate(pr.backcountry = predict(logit.backcountry, newdata = data.nptb.test, type = "response"))
```

```
logit.racquet <- glm(buyer ~ age + female + income + children + freq_endurance + freq_strength + freq_wa

data.nptb.test <- data.nptb.test %>%
mutate(pr.racquet = predict(logit.racquet, newdata = data.nptb.test, type = "response"))
```

```
logit.control <- glm(buyer ~ age + female + income + children + freq_endurance + freq_strength + freq_wa

data.nptb.test <- data.nptb.test %>%
mutate(pr.control = predict(logit.control, newdata = data.nptb.test, type = "response"))
```

```
data.nptb.test <- data.nptb.test %>%
mutate(pr.max = pmax(pr.endurance, pr.team, pr.water, pr.strength, pr.racquet, pr.backcountry, pr.contr
mail.offer = case_when(
pr.endurance == pr.max ~ "endurance",
pr.team == pr.max ~ "team",
pr.water== pr.max ~ "water",
pr.strength== pr.max ~ "strength",
pr.racquet== pr.max ~ "racquet",
pr.backcountry== pr.max ~ "backcountry",
pr.control== pr.max ~ "control"))
```

Question 1: Step 2

```
data.nptb.test %>%
tabyl(mail.offer)
```

```
  mail.offer      n      percent
 backcountry   2313 1.285000e-02
     control      7 3.888889e-05
   endurance 116425 6.468056e-01
     racquet  10430 5.794444e-02
    strength  45938 2.552111e-01
        team   2095 1.163889e-02
       water   2792 1.551111e-02
```

Question 1: Step 3

```
avg_order_db <- pent %>% filter(buyer==1)%>%
  group_by(message)%>%
  summarise(average_profit = mean(total_os)*0.4)
avg_order_db
```

```
# A tibble: 7 x 2
  message     average_profit
  <fct>                <dbl>
1 backcountry           25.0
2 control               20.0
3 endurance             22.0
4 racquet               23.1
5 strength              22.4
6 team                  23.0
7 water                 25.0
```

Question 1: Step 4

```
data.nptb.test <- data.nptb.test %>%
mutate(epr.backcountry = pr.backcountry *24.99912, epr.control = pr.control*19.97782, epr.endurance = pr
head(data.nptb.test)
```

```
# A tibble: 6 x 33
  custid buyer buyer.numeric total_os age   female income education
   <dbl> <fct>         <dbl>    <dbl> <fct>  <int>  <dbl>     <int>
1      3 0                 1        0 45 t~      0  35000        22
2     25 0                 1        0 >= 60      1  65000        32
3     30 0                 1        0 < 30       1  65000        62
4     55 0                 1        0 30 t~      0  35000        17
5     97 0                 1        0 < 30       1  65000        40
6    119 0                 1        0 30 t~      0  45000        23
# ... with 25 more variables: children <dbl>, freq_endurance <dbl>,
#   freq_strength <dbl>, freq_water <dbl>, freq_team <dbl>,
#   freq_backcountry <dbl>, freq_racquet <dbl>, message <fct>,
```

```
#   training <dbl>, pr.endurance <dbl>, pr.strength <dbl>, pr.water <dbl>,
#   pr.team <dbl>, pr.backcountry <dbl>, pr.racquet <dbl>,
#   pr.control <dbl>, pr.max <dbl>, mail.offer <chr>,
#   epr.backcountry <dbl>, epr.control <dbl>, epr.endurance <dbl>,
#   epr.racquet <dbl>, epr.strength <dbl>, epr.team <dbl>, epr.water <dbl>
```

```
data.nptb.test <- data.nptb.test %>%
mutate(epr.max = pmax(epr.endurance, epr.team, epr.water, epr.strength, epr.racquet, epr.backcountry, ep
mail.offer.ep = case_when(
epr.endurance == epr.max ~ "endurance",
epr.team == epr.max ~ "team",
epr.water== epr.max ~ "water",
epr.strength== epr.max ~ "strength",
epr.racquet== epr.max ~ "racquet",
epr.backcountry== epr.max ~ "backcountry",
epr.control== epr.max ~ "control"))
```

The epr.max column showcases the action that we would take for the customer.

```
data.nptb.test %>%
tabyl(mail.offer.ep)
```

```
 mail.offer.ep     n     percent
   backcountry 51620 0.286777778
     endurance 75420 0.419000000
       racquet  7379 0.040994444
      strength 33869 0.188161111
          team  1428 0.007933333
         water 10284 0.057133333
```

Based on profit, we should message in above propotion.

Question 1: Step 5

Average profit before using model:

```
data.nptb.test %>%summarise(mean(total_os))*0.4
```

```
  mean(total_os)
1        0.55188
```

Average profit after using model:

```
model_pr <- data.nptb.test %>%summarise(mean(epr.max))
model_pr
```

```
# A tibble: 1 x 1
  `mean(epr.max)`
            <dbl>
1           0.672
```

Question 1: Step 6

```
profit_db <-  data.nptb.test %>%
  summarise_at(vars(epr.backcountry, epr.team, epr.water, epr.strength, epr.endurance, epr.racquet, epr
```

```
profit_db
```

```
# A tibble: 1 x 7
  epr.backcountry epr.team epr.water epr.strength epr.endurance epr.racquet
            <dbl>    <dbl>     <dbl>        <dbl>         <dbl>       <dbl>
1           0.580    0.551     0.608        0.590         0.611       0.535
# ... with 1 more variable: epr.control <dbl>
```

Question 1: Step 7

```
average_random_alloted_pr <- rowMeans(profit_db[1,1:6])
average_random_alloted_pr
```

```
[1] 0.5789654
```

Question 1: Step 8

```
incremental_pr <- model_pr - average_random_alloted_pr
total_pr <- incremental_pr*5000000
Incremental_percent <- ((incremental_pr/average_random_alloted_pr)*100)
total_pr
```

```
  mean(epr.max)
1      464918.5
```

```
Incremental_percent
```

```
  mean(epr.max)
1      16.06032
```

Question 2

Issue - In the NPTB model that we have discussed, we are overtly reliant on historic/experimental data to predict a starting point for the consumer that is the first two emails we send to the customer. and then we use customer's response to these emails to predict the future promotional email to send. We believe this biases the prediction.

Explanation - For example, let's assume first two emails to send to a customer are on water and raquet sports. The consuer did not respond well to a majority of the emails send by these departments. However, we did not build any data on this customer's response to other department's emails. Hence, the customer's

response rate for other departments (say team) is 0. And thus, the NPTB model will again predict that we should send emails on water and racquet

Suggested solution - 1. Either reseve few emails every month to develop test data on customer responses 2. Define a cut-off for probability of response. If the predicted probability falls below this cut-off, we shoul