Pentathlon (Part III)

true
true
true
true

1 Preliminaries

1.0.1 Determine notebook defaults:

1.0.2 Load packages:

1.0.3 Read in the data:

```
load("PentathlonTargeting.Rdata")

pent <- pent %>% mutate(age=factor(age), female=factor(female), buyer=factor(buyer))

#Now we create the training and testing split

pent.train<- pent%>% filter(training==1)
pent.test<- pent%>% filter(training==0)
```

2 Part 1 Analysing the different offers

Type your answer text here, intervowen with blocks of R code

```
# First  we create a logistic regression model for every type of offer ( endurance, backcountry, etc)

lr.backcountry <- glm(buyer ~ age + female + income + education + children+ freq_endurance+freq_strengt
family=binomial, data=pent.train %>% filter(message=="backcountry"))

pent.test <- pent.test %>%
mutate(pr.backcountry = predict(lr.backcountry, newdata=pent.test, type="response"))
```

And we repeat for every other message

```
# Endurance

lr.endurance <- glm(buyer ~ age + female + income + education + children+ freq_endurance+freq_strength+
family=binomial, data=pent.train %>% filter(message=="endurance"))

pent.test <- pent.test %>%
mutate(pr.endurance = predict(lr.endurance, newdata=pent.test, type="response"))
```

```
# Strength

lr.strength <- glm(buyer ~ age + female + income + education + children+ freq_endurance+freq_strength+fi
family=binomial, data=pent.train %>% filter(message=="strength"))

pent.test <- pent.test %>%
mutate(pr.strength = predict(lr.strength, newdata=pent.test, type="response"))
```

```
#Water

lr.water <- glm(buyer ~ age + female + income + education + children+ freq_endurance+freq_strength+freq
family=binomial, data=pent.train %>% filter(message=="water"))

pent.test <- pent.test %>%
mutate(pr.water = predict(lr.water, newdata=pent.test, type="response"))
```

```
#Team

lr.team <- glm(buyer ~ age + female + income + education + children+ freq_endurance+freq_strength+freq_
family=binomial, data=pent.train %>% filter(message=="team"))

pent.test <- pent.test %>%
mutate(pr.team = predict(lr.team, newdata=pent.test, type="response"))
```

```
#Winter

lr.winter <- glm(buyer ~ age + female + income + education + children+ freq_endurance+freq_strength+freq
family=binomial, data=pent.train %>% filter(message=="winter"))

pent.test <- pent.test %>%
mutate(pr.winter = predict(lr.winter, newdata=pent.test, type="response"))
```

```
#Racquet

lr.racquet <- glm(buyer ~ age + female + income + education + children+ freq_endurance+freq_strength+fr
family=binomial, data=pent.train %>% filter(message=="racquet"))

pent.test <- pent.test %>%
mutate(pr.racquet = predict(lr.racquet, newdata=pent.test, type="response"))
```

2.1 Question 2

```
pent.test<- pent.test %>%
   mutate(pr.max = pmax(pr.backcountry, pr.endurance, pr.water, pr.team, pr.winter, pr.racquet,pr.streng
          mail.offer=case_when(
             pr.backcountry==pr.max ~ "Backcountry",
             pr.endurance==pr.max~ "Endurance",
             pr.water==pr.max~ "Water",
             pr.team==pr.max ~ "Team",
             pr.winter==pr.max~ "Winter",
```

```
            pr.racquet==pr.max ~ "Racquet",
            pr.strength==pr.max~"Strength"))

pent.test%>% tabyl(mail.offer )
```

```
  mail.offer     n      percent
 Backcountry  2407 0.0133722222
   Endurance 50303 0.2794611111
     Racquet  5903 0.0327944444
    Strength 71650 0.3980555556
        Team 19850 0.1102777778
       Water 29758 0.1653222222
      Winter   129 0.0007166667
```

2.2 Question 3

```
pent.test %>% filter(buyer==1) %>%
  group_by(message) %>%
   summarise(avg_os_backcountry= mean(backcountry_os),
             avg_os_endurance=mean(endurance_os),
             avg_os_racquet=mean(racquet_os),
             avg_os_team=mean(team_os),
             avg_os_water=mean(water_os),
             avg_os_winter=mean(winter_os),
             avg_os_strength=mean(strength_os),
             avg_os_sum=avg_os_strength+avg_os_winter+avg_os_water+avg_os_team+avg_os_racquet+avg_os_ba
             avg_total=mean(total_os))
```

```
# A tibble: 7 x 10
  message avg_os_backcoun~ avg_os_endurance avg_os_racquet avg_os_team
  <fct>              <dbl>            <dbl>          <dbl>       <dbl>
1 endura~             8.71            11.1            6.54        5.04
2 streng~            11.2              6.49           8.03        4.91
3 water               9.47             6.56           7.95        6.02
4 team                9.01             5.55           9.70        8.32
5 backco~            12.6              5.33           9.16        6.30
6 winter              9.63             5.66           9.16        4.88
7 racquet             8.11             5.40          11.6         6.85
# ... with 5 more variables: avg_os_water <dbl>, avg_os_winter <dbl>,
#   avg_os_strength <dbl>, avg_os_sum <dbl>, avg_total <dbl>
```

```
#calculate bucket size for each of message. Bucket size for endurance 56.13240 / strength 56.46285 /tea


pent.test<- pent.test %>%
  mutate(bucket_endurance=56.13240,
         bucket_strength=56.46285,
         bucket_water=65.00817,
         bucket_team=60.43344,
         bucket_backcountry=61.62745,
```

```
         bucket_winter=57.28686,
         bucket_racquet=57.58275)

pent.test <- pent.test %>%
  mutate(ep.backcountry=pr.backcountry*bucket_backcountry*0.4,
         ep.endurance=pr.endurance*bucket_endurance*0.4,
         ep.racquet=pr.racquet*bucket_racquet*0.4,
         ep.team=pr.team*bucket_team*0.4,
         ep.water=pr.water*bucket_water*0.4,
         ep.winter=pr.winter*bucket_winter*0.4,
         ep.strength=pr.strength*bucket_strength*0.4)

pent.test<- pent.test %>%
   mutate(ep.max = pmax(ep.backcountry, ep.endurance, ep.water, ep.team, ep.winter, ep.racquet,ep.streng
          mail.offer.ep=case_when(
             ep.backcountry==ep.max ~ "Backcountry",
             ep.endurance==ep.max~ "Endurance",
             ep.water==ep.max~ "Water",
             ep.team==ep.max ~ "Team",
             ep.winter==ep.max~ "Winter",
             ep.racquet==ep.max ~ "Racquet",
             ep.strength==ep.max~"Strength"))

pent.test%>% tabyl(mail.offer.ep )
```

```
 mail.offer.ep     n      percent
   Backcountry  5386 0.0299222222
     Endurance 28321 0.1573388889
       Racquet   872 0.0048444444
      Strength 57916 0.3217555556
          Team 18858 0.1047666667
         Water 68616 0.3812000000
        Winter    31 0.0001722222
```

2.3 Question 5

Using the predicted profit for all consumers in the test sample, what profit can we obtain on average per e-mailed customer when we customize the message to each customer?

```
pent.test %>%
  group_by(mail.offer.ep) %>%
  summarise(sum_ep=sum(ep.max), num=n(),avg_ep_customer=sum_ep/num)
```

```
# A tibble: 7 x 4
  mail.offer.ep sum_ep   num avg_ep_customer
  <chr>          <dbl> <int>           <dbl>
1 Backcountry    7208.  5386           1.34
2 Endurance     17602. 28321           0.622
3 Racquet         369.   872           0.424
4 Strength      29378. 57916           0.507
5 Team           9242. 18858           0.490
6 Water         65789. 68616           0.959
```

```
7 Winter          145.    31           4.69
```

```
pent.test %>%
   summarise_at(vars(ep.max),list(mean))
```

```
# A tibble: 1 x 1
  ep.max
   <dbl>
1  0.721
```

For total 180,000 customers in test sample, average estimated profit will be 0.72.

2.4 Question 6

Using the predicted profit for all consumers in the test sample, what profit can Pentathlon obtain on average per e-mailed customer if every customer receives the same message? Answer the question for each of the seven possible message(endurance, strength, water, team, backcountry, winter, or racquet).

```
pent.test %>%
   summarise_at(vars(ep.backcountry, ep.endurance,ep.racquet,ep.team,ep.water,ep.winter, ep.strength, ep
```

```
# A tibble: 1 x 8
  ep.backcountry ep.endurance ep.racquet ep.team ep.water ep.winter
           <dbl>        <dbl>      <dbl>   <dbl>    <dbl>     <dbl>
1          0.570        0.587      0.531   0.589    0.666     0.507
# ... with 2 more variables: ep.strength <dbl>, ep.max <dbl>
```

If all the customers get the same message, most profitable one will be Water category. But estimated profit will be 0.66 euro/customer which is lower than our prediction in Q5(0.72/customer)

2.5 Question 7

Using the predicted profit for all consumers in the test sample, what profit can Pentathlon obtain on average per e-mailed customer if every customer is assigned at random to one of the seven messages?

```
pent.test<- pent.test %>%
   mutate(pr.random=case_when(
              message=="backcountry" ~ pr.backcountry,
              message=="endurance"~ pr.endurance,
              message=="water"~pr.water,
              message=="team"~pr.team,
              message=="winter"~pr.winter,
              message=="racquet"~pr.racquet,
              message=="strength"~pr.strength))

pent.test <- pent.test %>%
  mutate(ep.random=case_when(
              message=="backcountry" ~ pr.random*bucket_backcountry*0.4,
              message=="endurance"~  pr.random*bucket_endurance*0.4,
              message=="water"~ pr.random*bucket_water*0.4,
```

```
             message=="team"~ pr.random*bucket_team*0.4,
             message=="winter"~ pr.random*bucket_winter*0.4,
             message=="racquet"~ pr.random*bucket_racquet*0.4,
             message=="strength"~ pr.random*bucket_strength*0.4))

pent.test %>%
   summarise_at(vars(ep.random, ep.max),list(mean))
```

```
# A tibble: 1 x 2
  ep.random ep.max
      <dbl>  <dbl>
1     0.576  0.721
```

2.6 Question 8

what improvement (in percent and in total Euro) does Pentathlon expect to get from customizing the message to each customer rather than assigning customers a message at random?

If we send email at random, we get 0.57euro/ customer total of 2.88M Euro from 5M customers. But by targeting the customer we get 0.72euro/customer total of 3.60M Euro from 5M customers. Targeting increases the profit by25% (0.72/0.57-1)

3 Part 2. Comment on draft for new e-mail policy proposal

1- Keeping the same 2 categories in place for an entire month could result in diminishing returns, lowering overall profitability.

2- By only exposing customers to the selected two categories and then using the data gathered will lead to biases and loss of information.

3- For the seasonal categories, it would be a mistake to use the past purchases because of their short lifespan. Another method would be use the same model for last year for the first month of the season and then improve upon that.

4- As a recomendation, the last week or two of every month could be used as a test for other categories and keep the customers preferences updated, and generating good data for future promotions