Pentathlon Part II

Section 81 Meaghan Fudge Krishna Surakanti Shiv Viswanathan

Read in the data:

```
load("PentathlonTargeting.RData")
```

Set Seed:

```
set.seed(1004)
```

Make Factor Variables:

```
pent <- pent %>%
mutate(custid = factor(custid),
buyer = factor(buyer),
age = factor(age),
female = factor(female),
message = factor(message),
training = factor(training))
```

Split into training and test:

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

Assignment answers

Part I: Analysis

Question 1: For each customer, determine the action (a message for endurance, strength, water, team, backcountry, racquet, or the no-message control condition) that is predicted to lead to the highest probability of purchase. Describe what approach you took to predict probability of purchase.

In order to predict probability of purchase, we trained a logistic regression for every message type using only people that received that message in train. Then, we applied the logistic regression to every respondent in

the test module across all messages. Then, by selecting the highest probability, that will determine what message they should receive in order to get the highest probability of purchase.

```
fm <- formula(buyer ~ age + female + income + education + children +
                freq_endurance + freq_water + freq_team + freq_backcountry +
                freq_racquet)

lr.endurance <- glm(fm, family=binomial, data=pent.train %>%
                      filter(message=="endurance"))

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

lr.strength <- glm(fm, family=binomial, data=pent.train %>%
                     filter(message=="strength"))

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

lr.water <- glm(fm, family=binomial, data=pent.train %>%
                  filter(message=="water"))

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

lr.team <- glm(fm, family=binomial, data=pent.train %>%
                 filter(message=="team"))

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

lr.backcountry <- glm(fm, family=binomial, data=pent.train %>%
                        filter(message=="backcountry"))

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

lr.racquet <- glm(fm, family=binomial,
                  data=pent.train %>% filter(message=="racquet"))

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

lr.control <- glm(fm, family=binomial,
                  data=pent.train %>% filter(message=="control"))

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

Next, we will select the message with the highest purchase probability for each person

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

Question 2: Report for each action the percent of customers in the test sample for whom that action maximizes their probability of purchase.

Endurance messages maximizied probability of purchase for most customers in this set.

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

```
  mail.offer      n     percent
 backcountry   1225 0.006805556
   endurance 121348 0.674155556
     racquet  13020 0.072333333
    strength  40575 0.225416667
        team   1911 0.010616667
       water   1921 0.010672222
```

Question 3: For each customer, determine the action (a message for endurance, strength, water, team, backcountry, racquet, or the no-message control condition) that is predicted tolead to the highest predicted profit (the COGS is 60%). Heads-up: There are different ways to predict order size; pick one that you think predicts order size the best. Explain how you calculated expected profit.

To calculate expected profit, we utilized a linear regression to predict order size in euros by training models for specific messages that only included buyers.After training the models, we used these models to predict the actual size for each message, for each person. Then we multiplied the predicted order size by the likelihood of purchase predicted in the previous question and then multiplied by .4 to get to the expected profit for each message. Then, we only selected the highest predicted profit of all the messages to determine which message they would receive if we were sending messages based on profit.

```
fm_os <- formula(total_os ~ age + female + income + education +
                   children + freq_endurance + freq_water + freq_team +
                   freq_backcountry + freq_racquet)

lm.endurance <- lm(fm_os, data=pent.train %>%
                     filter(message=="endurance", buyer==1))

pent.test <- pent.test %>%
mutate(pr.os.endurance = predict(lm.endurance,
```

```
                                newdata=pent.test, type="response"))

lm.strength <- lm(fm_os, data=pent.train %>%
                    filter(message=="strength", buyer==1))

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

lm.water <- lm(fm_os, data=pent.train %>%
                 filter(message=="water", buyer==1))

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

lm.team <- lm(fm_os, data=pent.train %>%
                filter(message=="team", buyer==1))

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

lm.backcountry <- lm(fm_os, data=pent.train %>%
                       filter(message=="backcountry", buyer==1))

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

lm.racquet <- lm(fm_os, data=pent.train %>%
                   filter(message=="racquet", buyer==1))

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

lm.control <- lm(fm_os, data=pent.train %>%
                   filter(message=="control", buyer==1))

pent.test <- pent.test %>%
mutate(pr.os.control = predict(lm.control, newdata=pent.test, type="response"))
```

```
pent.test <- pent.test %>%
mutate(ep.backcountry = pr.backcountry*pr.os.backcountry*.4, ep.control =
   pr.control*pr.os.control*.4, ep.endurance = pr.endurance*pr.os.endurance*.4,
   ep.racquet=pr.racquet*pr.os.racquet*.4, ep.strength=pr.strength*
     pr.os.strength*.4, ep.team=pr.team*pr.os.team*.4, ep.water=pr.water*
     pr.os.water*.4)

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

```
ep.backcountry == ep.max ~ "backcountry",
ep.control == ep.max ~ "no_msg",
ep.endurance == ep.max ~ "endurance",
ep.racquet == ep.max ~ "racquet",
ep.strength == ep.max ~ "strength",
ep.team == ep.max ~ "team",
ep.water == ep.max ~ "water"))
```

Question 4: Report for each action the percent of customers in the test sample for whom that action maximizes their predicted profit.

Despite what we saw earlier, there are some differences that we can see in terms of what breakdown of messages produce the most profit. Rather than endurance taking the large majority, there is now a decent proportion in backcountry and team messages

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

```
 mail.offer.ep     n    percent
   backcountry 26182 0.14545556
     endurance 68739 0.38188333
        no_msg  4497 0.02498333
       racquet 20430 0.11350000
      strength 10337 0.05742778
          team 37062 0.20590000
         water 12753 0.07085000
```

Question 5: Using the predicted profit for all customers in the test sample, what profit can we obtain on average per customer when we customize the message to each customer (including potentially sending no message)?

profit is on average 0.699 euros per person when you send a customized message

```
pent.test %>%
  summarize(avg.profit = sum(ep.max)/180000)
```

```
# A tibble: 1 x 1
  avg.profit
       <dbl>
1      0.699
```

Question 6: Using the predicted profit for all customers in the test sample, what profit can Pentathlon obtain on average per customer if every customer receives the same message (or the no-message control condition)? Answer the question for each of the seven possible actions (a message for endurance, strength, water, team, backcountry, racquet, or the no-message control condition).

Endurance yields the highest profit per person if everyone were to receive that message.

```
s_table <- pent.test %>%
  summarize(ap.backcountry = sum(ep.backcountry)/n(), ap.control =
              sum(ep.control)/n(), ap.endurance = sum(ep.endurance)/n(),
            ap.racquet = sum(ep.racquet)/n(), ap.strength = sum(ep.strength)/n(),
            ap.team = sum(ep.team)/n(), ap.water = sum(ep.water)/n())

s_table
```

```
# A tibble: 1 x 7
  ap.backcountry ap.control ap.endurance ap.racquet ap.strength ap.team ap.water
           <dbl>      <dbl>        <dbl>      <dbl>       <dbl>   <dbl>    <dbl>
1          0.594      0.431        0.628      0.522       0.602   0.540    0.609
```

Question 7: Using the predicted profit for all customers in the test sample, what profit can Pentathlon obtain on average per customer if every customer is assigned at random to receive one of the six messages?

If the messages were randomized, there is a 1/6 chance you'd receive a given message. On average if you randomized the messages you would net .58 euro profit on average per person.

```
s_table %>%
  summarize(ap.random.assignment = (ap.backcountry/6)+(ap.endurance/6)+
              (ap.racquet/6)+(ap.strength/6)+(ap.team/6)+(ap.water/6))
```

```
# A tibble: 1 x 1
  ap.random.assignment
                 <dbl>
1                0.583
```

Question 8: Based on the numbers calculated in questions 5 and 6, for the typical promotional e-mail blast to 5,000,000 customers, what improvement (in percent and in total Euro) does Pentathlon expect to get from customizing the message to each customer rather than assigning customers the message that is most profitable on average?

To compare the benefit of customization, you would take the highest single message profitability on average (endurance) as the baseline. You would compare that to the average profit from message customization that we found earlier.

```
baseline <- 0.6276
optimized <- 0.6993284
```

Euro improvement is 358,642

```
e.improvement = (optimized - baseline)*5000000

e.improvement
```

```
[1] 358642
```

Percentage improvement 11.43%

```
t_baseline <- baseline*5000000
t_optimized <- optimized*5000000

p.improvement = ((t_optimized-t_baseline)/t_baseline)*100
p.improvement
```

```
[1] 11.429
```

Part II: Analysis

Question 1: Comment on the draft for a new e-mail policy proposal. Are there any weaknesses? Can you suggest at least one improvement?

While the new email policy is more sound than the original strategy of randomly sending messages, there are some weaknesses that Anna should take into consideration going forward.

What's working well: they are using recent data each time to update the model/assumptions

What they will need to improve: First, because they are just using the most recent three weeks, they are not taking out external factors that may affect the data in specific period (e.g. perhaps it's seasonal to purchase from certain departments during a certain season, or perhaps there's a regional/global event that causes a stark increase in purchases from a certain department). Without taking this into consideration and looking at factors outside of the data sets, you may hit a transition of seasons and be using data that does not reflect trends into the future. Next, if there are two emails being changed every month, it will be unclear which email is impacting purchase - making this confounding. Finally, if you are using previous data, those individuals will not be randomly assigned to emails/messages. Therefore you are not running true tests each period and if you are sending far more emails in one category, the estimated profit will likely continue to stay high in that category. They will need to run true randomized tests each period (or ensure they separate out a subset of individuals to) if you want to update your models appropriately which may be costly when thinking about opportunity costs. They may need to evaluate how often they re-allocate the emails to each department.