HW 6

Vanessa Liu 476144

Tian Wei 474157

Shuyuan Tang 476399

Zhuoyuan He 473913

Yuhan Xu 474154

```
data  = read.csv('pentathlon.csv')
View(data)
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

rep=data[data$representative==1,]
train=data[data$training==1,]



lg=glm(buyer~message*(age+female+income+education+children+freq_endurance+fre
q_strength+freq_water+freq_team+freq_backcountry+freq_winter+freq_racquet),da
ta = train,family = "binomial", weights = sweight)

## Warning in eval(family$initialize): non-integer #successes in a binomial
## glm!

rep <- rep  %>% mutate(message = "endurance")
rep$p_endurance <- predict(lg, newdata=rep, type = "response")

rep <- rep  %>% mutate(message = "strength")
rep$p_strength <- predict(lg, newdata=rep, type = "response")

rep <- rep  %>% mutate(message = "water")
rep$p_water <- predict(lg, newdata=rep, type = "response")

rep <- rep  %>% mutate(message = "team")
rep$p_team <- predict(lg, newdata=rep, type = "response")
```

```r
rep <- rep  %>% mutate(message = "backcountry")
rep$p_backcountry <- predict(lg, newdata=rep, type = "response")

rep <- rep  %>% mutate(message = "winter")
rep$p_winter <- predict(lg, newdata=rep, type = "response")

rep <- rep  %>% mutate(message = "racquet")
rep$p_racquet <- predict(lg, newdata=rep, type = "response")

rep <- rep %>% rowwise %>% mutate(p_max = max(p_endurance, p_strength, p_wate
r, p_team, p_backcountry, p_winter, p_racquet)) %>% ungroup

rep <- rep %>% mutate(message_target = case_when(
  p_max == p_endurance ~ "endurance",
  p_max == p_strength ~ "strength",
  p_max == p_water ~ "water",
  p_max == p_team ~ "team",
  p_max == p_backcountry ~ "backcountry",
  p_max == p_winter ~ "winter",
  p_max == p_racquet ~ "racquet"))
```

To predict probability of purchase, we use buyer as dependent variable, the interaction between message and other demographic variables as independent variable in logistic regression.

```r
library(tidyverse)

## Warning: package 'tidyverse' was built under R version 3.6.2

## -- Attaching packages --------------------------------------- tidyverse 1.
3.0 --

## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v stringr 1.4.0
## v tidyr   1.0.0     v forcats 0.4.0
## v readr   1.3.1

## Warning: package 'ggplot2' was built under R version 3.6.2

## Warning: package 'tidyr' was built under R version 3.6.2

## Warning: package 'purrr' was built under R version 3.6.2

## -- Conflicts ------------------------------------------ tidyverse_conflict
s() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(knitr)
rep %>%
  group_by(message_target) %>%
  summarise(n_per_message = n()) %>%
  mutate(percent_message = n_per_message / sum(n_per_message))
```

```
## # A tibble: 7 x 3
##   message_target n_per_message percent_message
##   <chr>                  <int>           <dbl>
## 1 backcountry            12781         0.0426
## 2 endurance             174407         0.581
## 3 racquet                 2245         0.00748
## 4 strength               30437         0.101
## 5 team                   26235         0.0874
## 6 water                  48189         0.161
## 7 winter                  5706         0.0190
```

```
lm=lm(total_os~message*(age+female+income+education+children+freq_endurance+f
req_strength+freq_water+freq_team+freq_backcountry+freq_winter+freq_racquet),
data = train[train$buyer==1,])

rep <- rep  %>% mutate(message = "endurance")
rep$pf_endurance <- predict(lm, newdata=rep, type = "response")*rep$p_enduran
ce*0.4

rep <- rep  %>% mutate(message = "strength")
rep$pf_strength <- predict(lm, newdata=rep, type = "response")*rep$p_strength
*0.4

rep <- rep  %>% mutate(message = "water")
rep$pf_water <- predict(lm, newdata=rep, type = "response")*rep$p_water*0.4

rep <- rep  %>% mutate(message = "team")
rep$pf_team <- predict(lm, newdata=rep, type = "response")*rep$p_team*0.4

rep <- rep  %>% mutate(message = "backcountry")
rep$pf_backcountry <- predict(lm, newdata=rep, type = "response")*rep$p_backc
ountry*0.4

rep <- rep  %>% mutate(message = "winter")
rep$pf_winter <- predict(lm, newdata=rep, type = "response")*rep$p_winter*0.4

rep <- rep  %>% mutate(message = "racquet")
rep$pf_racquet <- predict(lm, newdata=rep, type = "response")*rep$p_racquet*0
.4

rep <- rep %>% rowwise %>% mutate(pf_max = max(pf_endurance, pf_strength, pf_
water, pf_team, pf_backcountry, pf_winter, pf_racquet)) %>% ungroup
```

```r
rep <- rep %>% mutate(message_target_pf = case_when(
  pf_max == pf_endurance ~ "endurance",
  pf_max == pf_strength ~ "strength",
  pf_max == pf_water ~ "water",
  pf_max == pf_team ~ "team",
  pf_max == pf_backcountry ~ "backcountry",
  pf_max == pf_winter ~ "winter",
  pf_max == pf_racquet ~ "racquet"))

head(rep)
```

```
## # A tibble: 6 x 44
##   custid buyer total_os message age   female income education children
##    <int> <int>    <int> <chr>   <fct>  <int>  <int>     <int>    <dbl>
## 1     59     0        0 racquet >= 60      1  65000        36      1.2
## 2     64     0        0 racquet < 30       1  40000        30      0.5
## 3     67     0        0 racquet 45 t~      0  60000        43      0.6
## 4     72     0        0 racquet 30 t~      1  45000        31      0.6
## 5     75     0        0 racquet < 30       1  85000        25      1.3
## 6     85     0        0 racquet 30 t~      0  45000        30      0.8
## # ... with 35 more variables: freq_endurance <int>, freq_strength <int>,
## #   freq_water <int>, freq_team <int>, freq_backcountry <int>,
## #   freq_winter <int>, freq_racquet <int>, endurance_os <int>,
## #   strength_os <int>, water_os <int>, team_os <int>,
## #   backcountry_os <int>, winter_os <int>, racquet_os <int>,
## #   training <int>, representative <int>, sweight <dbl>,
## #   p_endurance <dbl>, p_strength <dbl>, p_water <dbl>, p_team <dbl>,
## #   p_backcountry <dbl>, p_winter <dbl>, p_racquet <dbl>, p_max <dbl>,
## #   message_target <chr>, pf_endurance <dbl>, pf_strength <dbl>,
## #   pf_water <dbl>, pf_team <dbl>, pf_backcountry <dbl>, pf_winter <dbl>,
## #   pf_racquet <dbl>, pf_max <dbl>, message_target_pf <chr>
```

We use total order size as dependent variable, the interaction between message and other demographic variables as independent variable in linear regression. To calculate expected profit, we multiple the predicted order size by 0.4.

```r
rep %>%
  group_by(message_target_pf) %>%
  summarise(n_per_message_pf = n()) %>%
  mutate(percent_message_pf = n_per_message_pf / sum(n_per_message_pf))
```

```
## # A tibble: 7 x 3
##   message_target_pf n_per_message_pf percent_message_pf
##   <chr>                        <int>              <dbl>
## 1 backcountry                  33204             0.111
## 2 endurance                   101169             0.337
## 3 racquet                      20315             0.0677
## 4 strength                     12080             0.0403
## 5 team                         22158             0.0739
```

```
## 6 water                            102547            0.342
## 7 winter                             8527            0.0284
```

```r
rep %>%
  summarise(average_expected_profit = round(mean(pf_max),3))
```

```
## # A tibble: 1 x 1
##   average_expected_profit
##                     <dbl>
## 1                   0.214
```

```r
rep %>%
  summarise(mean_profit_water = round(mean(pf_water),3))
```

```
## # A tibble: 1 x 1
##   mean_profit_water
##               <dbl>
## 1             0.172
```

```r
rep %>%
  summarise(mean_profit_random = round(mean(cbind(pf_endurance, pf_strength,
pf_water, pf_team, pf_backcountry, pf_winter, pf_racquet)),3))
```

```
## # A tibble: 1 x 1
##   mean_profit_random
##                <dbl>
## 1              0.168
```

```r
improvement = (0.214 - 0.168)/0.168
improvement
```

```
## [1] 0.2738095
```

Case Question 2: 1. Data in the last week of each month is not in use. An improvement is do the analytics by the last day of each month.

2. Using the current month's data to make prediction for next month can be problematic. There can be seasonality issues. For example, in July all the customers are buying more water products. We suggest Anna do the following: For the first year, use the data from emails sent during the first three weeks in that month and repeats the analysis described in step B. For the sequent years, use the data from emails sent in the same calendar month in previous year and repeats the analysis described in step B.