MKTG 562: Group Assignment 4

Aishwary Jadhav, Jaivardhan Chauhan, Maitreyi Ekbote, Shreyansh Bhatia

Question 1)

The data in Exhibit 1 shows that:

- Both subscribed and unsubscribed customers generate more revenue as they receive more emails per week.
- The unsubscription rate increases with more emails per week.

<u>Possible explanations</u>:

1. **Increased Exposure Leads to More Purchases**: Customers who receive more emails are reminded more frequently about promotions and new products, increasing the likelihood of them making a purchase.
2. **Higher Engagement Among Subscribed Customers**: Customers who remain subscribed despite frequent emails may be more engaged and willing to spend more.
3. **Unsubscribed Customers Might Be Prior Shoppers**: Those who unsubscribe may still make occasional purchases because they were previously interested in the brand.
4. **Diminishing Returns and Overload**: As emails increase, unsubscription rates rise sharply (especially at 3+ emails per week), indicating a tipping point where customers feel overwhelmed.

Emails per Week	Unsubscription Rate	Revenue (Subscribed)	Revenue (Unsubscribed)	Profit (Subscribed)	Profit (Unsubscribed)	LTV (Subscribed)	LTV (Unsubscribed)	Final Weighted LTV
1	0.024	9.97	6.32	3.988	2.528	23.11236018	14.65096453	22.90928668
2	0.034	14.21	6.36	5.684	2.544	32.94148828	14.74369215	32.32276321
3	0.097	14.56	6.7	5.824	2.68	33.75285499	15.53187695	31.98542012
4	0.171	14.89	7.11	5.956	2.844	34.51785788	16.48233509	31.43378349
						Final LTV	32.32276321	

Final LTV: $32.32

Months	Discount Rate
1	0.990099
2	0.980296
3	0.97059
4	0.96098
5	0.951466
6	0.942045

Interpreting the LTV Results

- The highest LTV occurs at 2 emails per week ($32.32).
- While sending 3 or 4 emails increases immediate revenue, the higher unsubscription rate lowers overall LTV.
- LTV drops from $32.32 (at 2 emails) to $31.98 (at 3 emails) and further to $31.43 (at 4 emails).
- This means sending more than 2 emails erodes long-term value.

Optimal Recommendation: Send 2 Emails per Week

2 emails per week is the best balance between engagement and customer retention. This frequency keeps customers engaged while minimizing churn, ensuring the highest long-term value.

Problems of more/less than 2 emails per week

- Unsubscription nearly triples from 3.4% (at 2 emails) to 9.7% (at 3 emails).
- This offsets the small revenue increase and lowers total LTV.
- While 1 email has the lowest unsubscription rate (2.4%), it also generates the least revenue.
- The LTV at 1 email/week is $22.91, significantly lower than 2 emails/week ($32.32).

Question 2)

The logistic regression models showed distinct message assignment patterns under probability-based versus profitability-based targeting:

Probability-based targeting prioritized messages with the highest predicted engagement rates. "Beauty" (43.42%) and "home" (35.65%) dominated, reflecting the strong predicted likelihood of driving clicks and purchases. Detailed results are shown as prob_percentages in the code output.

Profitability-based targeting prioritized messages with the highest expected profit (probability × order size × 40% margin). Here, "home" captured 98.86% of assignments due to its significantly higher average order value, despite lower engagement probabilities. This highlights that profitability optimization favors high-value departments even if their engagement rates are moderate. Check profit_percentages for results.

```
data = read.csv("email_recommendations.csv")

logit_shoes <- glm(
  click_and_purch ~ age + female + income + education + children + region +
    numitems_womens_clothing + numitems_shoes + numitems_handbags +
    numitems_jewelry + numitems_beauty + numitems_men + numitems_home, subset
= (message == "shoes"),
  data = data,
  family = binomial()
)

data$predicted_prob_shoes <- predict(logit_shoes, newdata = data, type =
"response")


logit_womens_clothing <- glm(
  click_and_purch ~ age + female + income + education + children + region +
    numitems_womens_clothing + numitems_shoes + numitems_handbags +
    numitems_jewelry + numitems_beauty + numitems_men + numitems_home, subset
= (message == "womens_clothing"),
  data = data,
  family = binomial()
)

data$predicted_prob_womens_clothing <- predict(logit_womens_clothing, newdata
= data, type = "response")


logit_beauty <- glm(
  click_and_purch ~ age + female + income + education + children + region +
    numitems_womens_clothing + numitems_shoes + numitems_handbags +
    numitems_jewelry + numitems_beauty + numitems_men + numitems_home, subset
```

```
= (message == "beauty"),
  data = data,
  family = binomial()
)

data$predicted_prob_beauty <- predict(logit_beauty, newdata = data, type =
"response")

logit_handbags <- glm(
  click_and_purch ~ age + female + income + education + children + region +
    numitems_womens_clothing + numitems_shoes + numitems_handbags +
    numitems_jewelry + numitems_beauty + numitems_men + numitems_home, subset
= (message == "handbags"),
  data = data,
  family = binomial()
)

data$predicted_prob_handbags <- predict(logit_handbags, newdata = data, type
= "response")

logit_jewelry <- glm(
  click_and_purch ~ age + female + income + education + children + region +
    numitems_womens_clothing + numitems_shoes + numitems_handbags +
    numitems_jewelry + numitems_beauty + numitems_men + numitems_home, subset
= (message == "jewelry"),
  data = data,
  family = binomial()
)

data$predicted_prob_jewelry <- predict(logit_jewelry, newdata = data, type =
"response")

logit_home <- glm(
  click_and_purch ~ age + female + income + education + children + region +
    numitems_womens_clothing + numitems_shoes + numitems_handbags +
    numitems_jewelry + numitems_beauty + numitems_men + numitems_home, subset
= (message == "home"),
  data = data,
  family = binomial()
)

data$predicted_prob_home <- predict(logit_home, newdata = data, type =
"response")

logit_men <- glm(
  click_and_purch ~ age + female + income + education + children + region +
    numitems_womens_clothing + numitems_shoes + numitems_handbags +
    numitems_jewelry + numitems_beauty + numitems_men + numitems_home, subset
= (message == "men"),
```

```
  data = data,
  family = binomial()

)

data$predicted_prob_men <- predict(logit_men, newdata = data, type =
"response")

data$max <- pmax(data$predicted_prob_shoes,
data$predicted_prob_womens_clothing, data$predicted_prob_beauty,
data$predicted_prob_handbags, data$predicted_prob_jewelry,
data$predicted_prob_home, data$predicted_prob_men )

# Assign the message with the highest predicted probability to each customer
data$optimal_message_prob <- names(data[, c("predicted_prob_shoes",
"predicted_prob_womens_clothing",
                                           "predicted_prob_beauty",
"predicted_prob_handbags",
                                           "predicted_prob_jewelry",
"predicted_prob_home",

"predicted_prob_men")])[max.col(data[, c("predicted_prob_shoes",

"predicted_prob_womens_clothing",

"predicted_prob_beauty",

"predicted_prob_handbags",

"predicted_prob_jewelry",

"predicted_prob_home",

"predicted_prob_men")],

ties.method = "first")]

# Convert to message names
data$optimal_message_prob <- gsub("predicted_prob_", "",
data$optimal_message_prob)

# percentages for each message
prob_percentages <- prop.table(table(data$optimal_message_prob)) * 100
prob_percentages <- round(prob_percentages, 2)

prob_percentages

##
##            beauty          handbags                home          jewelry
men
```

```
##           43.42            9.25            35.65            5.53
0.35
##           shoes  womens_clothing
##            2.19             3.60

avg_os <- list()
messages <- c("shoes", "womens_clothing", "beauty", "handbags", "jewelry",
"home", "men")

for (msg in messages) {
  os_col <- paste0("os_", gsub(" ", "_", msg))  # e.g., "os_shoes"
  avg_os[[msg]] <- mean(data[data$message == msg & data$click_and_purch == 1,
os_col], na.rm = TRUE)
}

avg_os <- lapply(avg_os, function(x) ifelse(is.nan(x), 0, x))

# expected profit for each message
data$profit_shoes <- data$predicted_prob_shoes * avg_os$shoes * 0.4  # COGS =
60%
data$profit_womens_clothing <- data$predicted_prob_womens_clothing *
avg_os$womens_clothing * 0.4
data$profit_beauty <- data$predicted_prob_beauty * avg_os$beauty * 0.4
data$profit_handbags <- data$predicted_prob_handbags * avg_os$handbags * 0.4
data$profit_jewelry <- data$predicted_prob_jewelry * avg_os$jewelry * 0.4
data$profit_home <- data$predicted_prob_home * avg_os$home * 0.4
data$profit_men <- data$predicted_prob_men * avg_os$men * 0.4

# Assign the message with the highest expected profit to each customer
data$optimal_message_profit <- names(data[, c("profit_shoes",
"profit_womens_clothing",
                                              "profit_beauty",
"profit_handbags",
                                              "profit_jewelry", "profit_home",
                                              "profit_men")])[max.col(data[,
c("profit_shoes",

"profit_womens_clothing",

"profit_beauty",

"profit_handbags",

"profit_jewelry",

"profit_home",

"profit_men")],

ties.method = "first")]
```

```r
# Convert to message names
data$optimal_message_profit <- gsub("profit_", "",
data$optimal_message_profit)

# percentages for each message
profit_percentages <- prop.table(table(data$optimal_message_profit)) * 100
profit_percentages <- round(profit_percentages, 2)

profit_percentages

##
## handbags      home     shoes
##     1.10    98.86      0.04

# Targeted Profit (using max profitability)
data$profit_max <- pmax(
  data$profit_shoes,
  data$profit_womens_clothing,
  data$profit_beauty,
  data$profit_handbags,
  data$profit_jewelry,
  data$profit_home,
  data$profit_men
)
targeted_profit <- mean(data$profit_max, na.rm = TRUE)
```

Question 3)

Comparing targeted and random messaging strategies:

Targeted profit: Generated **$2.77 per customer** by selecting the most profitable message for each individual.

Random profit: Averaged **$1.51 per customer** when messages were assigned randomly.

Profit gain: Targeted messaging yielded an additional $1.26 per customer, translating to $1,260,540 in total incremental profit for 1 million customers. This demonstrates that profitability-driven personalization substantially outperforms random allocation, justifying Neiman Marcus' shift to data-driven targeting. Code and output below.

```r
# Random Profit (average of all message profits)
data$profit_random <- rowMeans(
  data[, c("profit_shoes", "profit_womens_clothing", "profit_beauty",
           "profit_handbags", "profit_jewelry", "profit_home",
"profit_men")],
  na.rm = TRUE
)
random_profit <- mean(data$profit_random, na.rm = TRUE)

# Profit Gain
profit_gain_per_customer <- targeted_profit - random_profit
total_gain <- profit_gain_per_customer * 1e6  # For 1,000,000 customers

cat("Average Targeted Profit per Customer: $", round(targeted_profit, 2),
"\n")

## Average Targeted Profit per Customer: $ 2.77

cat("Average Random Profit per Customer: $", round(random_profit, 2), "\n")

## Average Random Profit per Customer: $ 1.51

cat("Additional Profit (Targeted vs Random): $", round(total_gain, 2))

## Additional Profit (Targeted vs Random): $ 1260540
```