MSIN0094 Third Assignment

Due 10 am, 24 Dec for SORA students

```r
data_full <- read.csv("https://www.dropbox.com/s/pc690z638w828v8/amazon.csv?dl=1")
```

1.

```r
#part (a)
data_full <- data_full %>%
  mutate(recency = rowSums(select(., c(last))),
         frequency = rowSums(select(., c(home, sports, clothes, health, books,
                                         digital, toys))),
         monetaryvalue = rowSums(select(., c(electronics, nonelectronics))) )

#part (b)
colMeans(data_full[sapply(data_full, is.numeric)])
```

```
##        user_id          first           last    electronics nonelectronics
##     15000.5000        25.3360        12.2612        46.4248       161.7368
##           home         sports        clothes         health          books
##         0.8352         0.3936         0.9150         0.4656         0.3079
##        digital           toys        recency      frequency  monetaryvalue
##         0.3821         0.5511        12.2612         3.8505       208.1616
```

```r
#OR
mean(data_full$recency)
```

```
## [1] 12.2612
```

```r
mean(data_full$frequency)
```

```
## [1] 3.8505
```

```r
mean(data_full$monetaryvalue)
```

```
## [1] 208.1616
```

2.

```r
## please finish all 4 steps (a to d) in this single code block
#parts (a) and (b)
data_full <- data_full%>%
  mutate(R_group = ntile(recency,4))%>%
  group_by(R_group)%>%
  mutate(F_group = ntile(-frequency,4))%>%
  ungroup()%>%
  group_by(R_group, F_group) %>%
  mutate(M_group = ntile(-monetaryvalue,4))%>%
  ungroup()%>%
  arrange(R_group, F_group, M_group) %>%
  mutate(new_group = ifelse(R_group != lag(R_group) |
                              F_group != lag(F_group) |
                              M_group != lag(M_group), 1L, 0L)) %>%
  mutate(new_group = ifelse(is.na(new_group),1L,new_group)) %>%
  mutate(RFM_group = cumsum(new_group))
```

3.

```r
#part (a)
data_full <- data_full%>%
  mutate(binary_subscribe = ifelse(subscribe == "yes", 1L, 0L))%>%
  group_by(RFM_group) %>%
  mutate(avg_response_rate = mean(binary_subscribe, na.rm = T))%>%
  ungroup()
```

part(b) Group 1 has the highest average response rate and RFM group 63 has the smallest.

```r
#highest response rate
data_full %>% group_by(RFM_group) %>%
  summarise(maximum = max(avg_response_rate)) %>%
  arrange(-maximum) %>%
  head()
```

```
## # A tibble: 6 x 2
##   RFM_group maximum
##       <int>   <dbl>
## 1         1   0.255
## 2         4   0.244
## 3        19   0.192
## 4        17   0.185
## 5         3   0.179
## 6         2   0.173
```

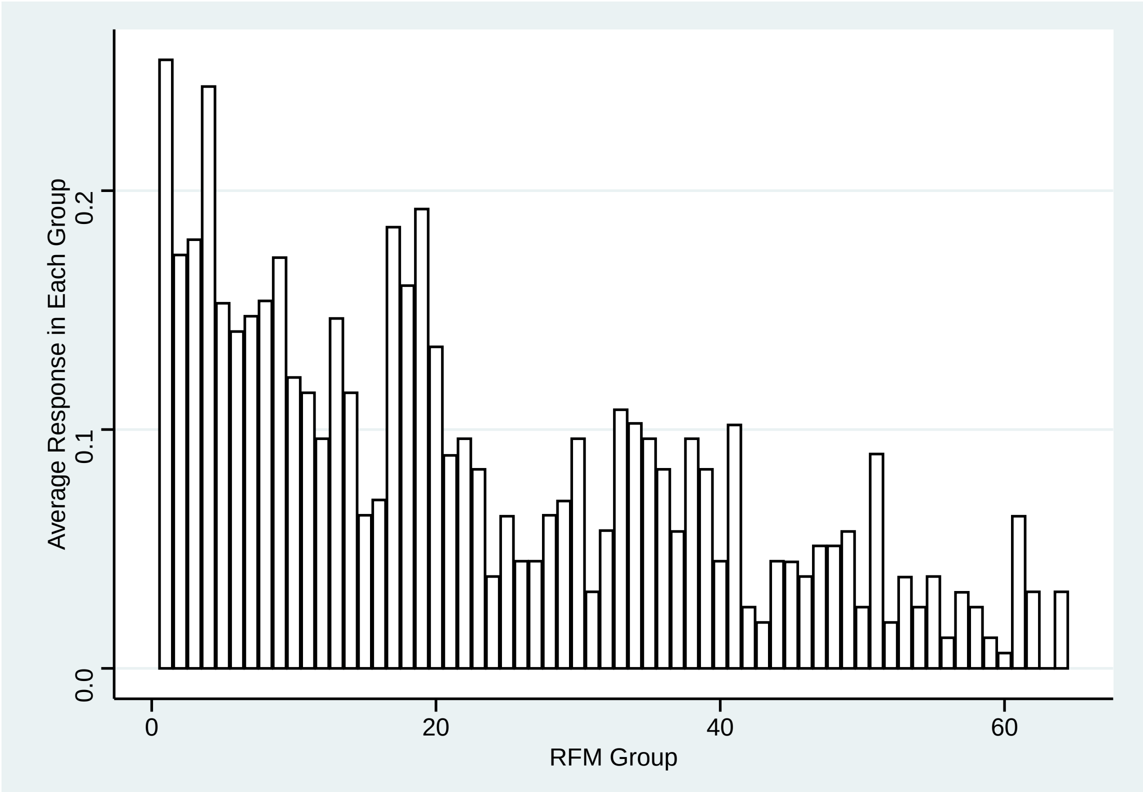
```r
#lowest response rate
data_full %>% group_by(RFM_group) %>%
  summarise(maximum = max(avg_response_rate)) %>%
  arrange(maximum) %>%
  head()
```

```
## # A tibble: 6 x 2
##   RFM_group maximum
##       <int>   <dbl>
## 1        63 0
## 2        60 0.00641
## 3        56 0.0128
## 4        59 0.0128
## 5        43 0.0192
## 6        52 0.0192
```

It can also be from the figure below where 1 bar is equal to 1 group. The highest bar indicating the highest response rate is in group 1 as seen on axis x, while there is an non-existing bar for the lowest response rate in group 63, implying there were no responses at all.

```r
#part (b) alternative
data_RFM <- data_full %>%
  group_by(RFM_group) %>%
  summarise(avg_response = mean(binary_subscribe),
            R_min = min(recency), R_max = max(recency),
            F_min = min(frequency), F_max = max(frequency),
            M_min = min(monetaryvalue), M_max = max(monetaryvalue))%>%
    ungroup()

library(ggthemes)
library(ggplot2)
ggplot(data = data_RFM) +
  geom_bar(aes(x = RFM_group, y = avg_response), stat="identity",
           color = "Black", fill = "white") +
  theme_stata() +
  xlab("RFM Group") +
  ylab("Average Response in Each Group")
```



part (c) A smaller RFM_group ID leads to a higher average response rate only in general terms, as an *overall trend*, because that's where the individual responses are the highest on the graph above. However, *group-wise individually*, it is not the case. For instance, group 4 has the second highest average response rate, instead of group 2 which comes 6th overall. If the statement held true, the value of RFM groups would significantly diminish, as we would not see "batches" within each segment. (I really hope it makes sense.)

4.

```
COGS <- 0.5
cost_per_offer <- 2
profit_per_customer <- (40-4) * (1 - COGS)
# where 4 is average shipping costs for Amazon, not included to COGS shipping
# costs; and 40 is the average revenue of goods purchased by new subscribers

breakeven_response_rate <- cost_per_offer/profit_per_customer

breakeven_response_rate
```

```
## [1] 0.1111111
```

5.

```
#part (a)
data_full <- data_full%>%
  mutate(is_target_RFM = ifelse(avg_response_rate > breakeven_response_rate,
                                1L, 0L))

#part (b)
sum(data_full$is_target_RFM == 1)
```

```
## [1] 2657
```

part (b) 2657 customers are targeted.

6.

0.1 Compare Blanket Marketing and Target Marketing

part (a) If the company does blanket marketing:

```
total_costs_of_mailing_blanket <- cost_per_offer * 10000

total_profit_blanket <- sum(data_full$binary_subscribe) * profit_per_customer

#ROI=(profit from the campaign-cost of the campaign)/cost of the campaign
ROI_blanket <- (total_profit_blanket - total_costs_of_mailing_blanket)/total_costs_of_maili

ROI_blanket
```

```
## [1] -0.2458
```

part (b) If the company uses RFM analysis and conducts targeted marketing:

```
#we only selectively send the campaign to those whose 'is_target_RFM' == 1
total_costs_of_mailing_RFM <- cost_per_offer * sum(data_full$is_target_RFM)

#how many of them are actually subscribed to us?
total_profit_RFM <- sum((data_full%>%filter(is_target_RFM==1))$binary_subscribe)*profit_per

ROI_RFM <- (total_profit_RFM - total_costs_of_mailing_RFM)/total_costs_of_mailing_RFM

ROI_RFM
```

```
## [1] 0.4768536
```

part (c) Tom should go with RFM targeted marketing, as the simple predictive analytics model RFM analysis can help the company boost the ROI by a large extent.

7.

```
data_full_2 <- data_full%>%
  mutate(R_group_2 = ntile(recency,10))%>%
  group_by(R_group_2)%>%
  mutate(F_group_2 = ntile(-frequency,10))%>%
  ungroup()%>%
  group_by(R_group_2, F_group_2) %>%
  mutate(M_group_2 = ntile(-monetaryvalue,10))%>%
  ungroup()%>%
  arrange(R_group_2, F_group_2, M_group_2) %>%
  mutate(new_group_2 = ifelse(R_group_2 != lag(R_group_2) |
                               F_group_2 != lag(F_group_2) |
                               M_group_2 != lag(M_group_2), 1L, 0L)) %>%
  mutate(new_group_2 = ifelse(is.na(new_group_2),1L,new_group_2)) %>%
  mutate(RFM_group_2 = cumsum(new_group_2))

data_full_2 <- data_full_2 %>%
  group_by(RFM_group_2) %>%
  mutate(avg_response_rate_2 = mean(data_full_2$binary_subscribe, na.rm = T))%>%
  ungroup()

data_full_2 <- data_full_2 %>%
  mutate(is_target_RFM_2=ifelse(avg_response_rate_2 > breakeven_response_rate,
                                 1L, 0L))

total_costs_of_mailing_RFM_2<-cost_per_offer * sum(data_full_2$is_target_RFM_2)

total_profit_RFM_2<-sum((data_full_2%>%filter(is_target_RFM_2==1))$
                          binary_subscribe)*profit_per_customer

ROI_RFM_2<-(total_profit_RFM_2-total_costs_of_mailing_RFM_2)/
  total_costs_of_mailing_RFM_2

ROI_RFM_2
```

```
## [1] NaN
```

After rerunning the RFM analysis with 10 groups, we can see that the ROI outputs are the same as with 4 groups. Hence, we should not have many quantile groups in each R, F, M group as possible so as to increase the effectiveness of our targeting, as after a certain threshold, it does not improve the accuracy of the marketing decision. Adding more groups will be very time and money-consuming. The extreme case is dividing groups into 1 individual, there is no difference between RFM and the original data.

8. *RFM analysis* in terms of conducting targeted marketing:

- It can greatly boost marketing ROI.
- Works well only when we have a large customer database, so that we can categorize future customers into one of the existing RFM groups. Hence, RFM may be inconvenient for start-ups and SMEs.
- It may not be obvious which number of groups is best to be per each segment.
- "RFM analysis normally does not use this or other customer information such as gender", as stated in the case study, hence it hinders the sophistication of the modelling approach.

Linear probability model (LPM) in terms of conducting targeted marketing: + Works for both continuous, categorical predictors, interpretation terms, and discrete outcome variables. + Can be used to estimate the parameters and make predictions, albeit dependent variable being binary. + Can overcome the problems with RFM even on a small training set (would be beneficial for Tom but not necessary for Amazon). +/- Simpler models are easier to interpret but gives lower accuracy. +/- Complicated models may have higher prediction accuracy but results are not intuitive to interpret. ("Accuracy" means how close our prediction is to the ground truth.) - Predicted probabilities of occurring may fall out of the [0,1] range. - Cannot handle multi-categorical classification problems (doesn't fit the data well).

Logistic regression in terms of conducting targeted marketing: + Can accommodate continuous, categorical predictors, interpretation terms, only that the dependent variable is binary. + Also, works good with odds, binary decisions and random utility/choice problems. +/- Predicts a probability, between 0 and 1, of purchase or response, which can be used for targeting and prediction decisions, but what if again, our predicted probability is not int he [0,1] range? - Logit models only work with discrete outcome variables. - More complicated and time-consuming to estimate than linear models; especially, if the model has a large number of fixed effects, it will be extremely time costly to estimate logistic regression models.

Noteworthy, no model can always perform the best on all datasets.

9. Complete the following code block to split the `data_full` into a training set that accounts for 70% of total data, and a test set that accounts for the remaining 30% of data. (Please do not modify the seed, or you will get different results) (**4pts** in total)

```
set.seed(888)  #to be able to replicate the results every time we run the code

#we want the size of the new dataset to be 70% of those 10,000 rows
```

```r
training_set_index <- sample(x = 1:nrow(data_full),
                             size = 0.7 * nrow(data_full),
                             replace = F)

data_training <- data_full[training_set_index,]
data_test <- data_full[-training_set_index,]
#minus sign says "remove those individuals"

data_training %>% head
```

```
## # A tibble: 6 x 26
##   user_id gender first  last electronics nonelectronics  home sports clothes
##     <int> <chr>  <int> <int>       <int>          <int> <int>  <int>   <int>
## 1   17649 M          5     1          25            106     1      0       1
## 2   16325 F         17    11          25            298     0      0       2
## 3   15977 F         19    11          25            160     0      0       2
## 4   11033 F          5     3          25             72     0      0       2
## 5   16757 F         19    13          25             31     1      0       1
## 6   19999 F         31    25          27            124     0      0       1
## # ... with 17 more variables: health <int>, books <int>, digital <int>,
## #   toys <int>, subscribe <chr>, city <chr>, recency <dbl>, frequency <dbl>,
## #   monetaryvalue <dbl>, R_group <int>, F_group <int>, M_group <int>,
## #   new_group <int>, RFM_group <int>, binary_subscribe <int>,
## #   avg_response_rate <dbl>, is_target_RFM <int>
```

```r
# please check if the first observation in the data_training after this step
# has user_id 17649
```

10.

0.2 Linear Probability Model

part(a)

```r
#Step 1: Run a linear probability model to decide which customers to target on
# the training set.
LPM <- lm(data = data_training,
          formula = binary_subscribe ~ factor(gender, c("M", "F")) + last +
            electronics + nonelectronics + home + sports + clothes + health +
            books + digital + toys) #response is now a binary variable

#Step 2: Generate a variable that predicts the probabilities of subscribing to
```

```r
# Amazon Prime for customers in the test set (a predicted probability).
data_test <- data_test %>%
  mutate(predicted_prob_LPM = predict(LPM, data_test))
```

part(b)

```r
data_test%>%slice(which.max(predicted_prob_LPM))
```

```
## # A tibble: 1 x 27
##   user_id gender first  last electronics nonelectronics  home sports clothes
##     <int> <chr>  <int> <int>       <int>          <int> <int>  <int>   <int>
## 1   10927 F         45     9         109             50     2      0       1
## # ... with 18 more variables: health <int>, books <int>, digital <int>,
## #   toys <int>, subscribe <chr>, city <chr>, recency <dbl>, frequency <dbl>,
## #   monetaryvalue <dbl>, R_group <int>, F_group <int>, M_group <int>,
## #   new_group <int>, RFM_group <int>, binary_subscribe <int>,
## #   avg_response_rate <dbl>, is_target_RFM <int>, predicted_prob_LPM <dbl>
```

Customer with user_id 10927 has the highest predicted probability of subscribing from LPM.

part(c)

```r
summary(LPM)
```

```
##
## Call:
## lm(formula = binary_subscribe ~ factor(gender, c("M", "F")) +
##     last + electronics + nonelectronics + home + sports + clothes +
##     health + books + digital + toys, data = data_training)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.57068 -0.11970 -0.05249  0.00349  1.05362
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   1.494e-01  1.214e-02  12.308  < 2e-16 ***
## factor(gender, c("M", "F"))F -5.699e-02  6.971e-03  -8.175 3.48e-16 ***
## last                         -5.176e-03  3.782e-04 -13.686  < 2e-16 ***
## electronics                  -1.507e-03  1.695e-03  -0.889  0.37414
## nonelectronics                8.869e-05  3.524e-05   2.517  0.01186 *
## home                          4.737e-03  1.677e-02   0.282  0.77762
```

```
## sports                          7.402e-03  1.680e-02   0.441  0.65944
## clothes                        -7.522e-03  1.816e-02  -0.414  0.67865
## health                         -1.823e-02  1.920e-02  -0.949  0.34250
## books                           4.098e-02  2.035e-02   2.014  0.04405 *
## digital                         1.270e-01  2.158e-02   5.888 4.10e-09 ***
## toys                            7.211e-02  2.249e-02   3.206  0.00135 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2579 on 6988 degrees of freedom
## Multiple R-squared:  0.1343, Adjusted R-squared:  0.1329
## F-statistic: 98.54 on 11 and 6988 DF,  p-value: < 2.2e-16
```

Keeping other variables unchanged, the probability of a customer subscribing to Amazon Prime is 8.175 (see t value in the regression summary above) less for females than to the probability of male.

11. *part (a)* The number of targeted customers is 1024.

```
data_test <- data_test %>%
  mutate(is_target_LPM = ifelse(predicted_prob_LPM > breakeven_response_rate,
                                1L, 0L))

sum(data_test$is_target_LPM==1)
```

```
## [1] 1024
```

part (b)

```
total_costs_of_mailing_LPM <- cost_per_offer * sum(data_test$is_target_LPM)

total_profit_LPM <-sum((data_test%>%filter(is_target_LPM==1))$binary_subscribe)*  profit_per

ROI_LPM<-(total_profit_LPM-total_costs_of_mailing_LPM)/total_costs_of_mailing_LPM

ROI_LPM
```

```
## [1] 0.7138672
```

12. *part (a)*

```r
logistic <- glm(data = data_training,
                formula = binary_subscribe ~ factor(gender, c("F", "M")) + last
                + electronics + home + sports + clothes + health + books +
                  digital + toys,
                family = "binomial")

data_test <- data_test %>%
mutate(predicted_prob_logistic=predict(logistic, data_test, type = "response"))
```

part (b)

```r
data_test %>% slice(which.max(data_test$predicted_prob_logistic))
```

```
## # A tibble: 1 x 29
##   user_id gender first  last electronics nonelectronics  home sports clothes
##     <int> <chr>  <int> <int>       <int>          <int> <int>  <int>   <int>
## 1   10723 F         35     1         105            110     0      1       0
## # ... with 20 more variables: health <int>, books <int>, digital <int>,
## #   toys <int>, subscribe <chr>, city <chr>, recency <dbl>, frequency <dbl>,
## #   monetaryvalue <dbl>, R_group <int>, F_group <int>, M_group <int>,
## #   new_group <int>, RFM_group <int>, binary_subscribe <int>,
## #   avg_response_rate <dbl>, is_target_RFM <int>, predicted_prob_LPM <dbl>,
## #   is_target_LPM <int>, predicted_prob_logistic <dbl>
```

Customer with user_id=10723 has the highest predicted probability of subscribing, as seen from the tibble above.

part (c)

```r
summary(logistic)
```

```
##
## Call:
## glm(formula = binary_subscribe ~ factor(gender, c("F", "M")) +
##     last + electronics + home + sports + clothes + health + books +
##     digital + toys, family = "binomial", data = data_training)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4100  -0.3960  -0.2615  -0.1682   3.3050
##
## Coefficients:
##                                Estimate Std. Error z value Pr(>|z|)
```

```
## (Intercept)                  -2.110690   0.145159 -14.541  < 2e-16 ***
## factor(gender, c("F", "M"))M  0.866822   0.099433   8.718  < 2e-16 ***
## last                         -0.105426   0.007957 -13.249  < 2e-16 ***
## electronics                  -0.029341   0.024852  -1.181  0.23775
## home                          0.150837   0.249199   0.605  0.54499
## sports                        0.207081   0.246500   0.840  0.40086
## clothes                      -0.007262   0.268287  -0.027  0.97840
## health                       -0.216469   0.284248  -0.762  0.44633
## books                         0.670828   0.299325   2.241  0.02502 *
## digital                       1.503611   0.315192   4.770 1.84e-06 ***
## toys                          0.975959   0.327147   2.983  0.00285 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 4028.5  on 6999  degrees of freedom
## Residual deviance: 3169.0  on 6989  degrees of freedom
## AIC: 3191
##
## Number of Fisher Scoring iterations: 6
```

Keeping everything else unchanged, male customers have 0.866822 times the odds of female customers.

part (d) Keeping everything else unchanged, if the day of a customer's last purchase is one day before, the odds of this individual subscribing to Amazon prime are decreased by 0.105426.

13.

```r
#part (a)
data_test <- data_test %>%
  mutate(is_target_logistic = ifelse(predicted_prob_logistic >
                                       breakeven_response_rate, 1L, 0L))

sum(data_test$is_target_logistic)
```

```
## [1] 631
```

```r
#part (b)
total_costs_of_mailing_logistic <- cost_per_offer * sum(data_test$is_target_logistic)

total_profit_logistic <- sum((data_test%>%filter(is_target_logistic==1))$
```

```
                                        binary_subscribe) * profit_per_customer

ROI_logistic <- (total_profit_logistic - total_costs_of_mailing_logistic)/total_costs_of_ma

ROI_logistic
```

```
## [1] 1.353407
```

14. ROI of logistic regression = 1.353407, whereas ROI of LPM = 0.7138672. Hence, ROI of logistic regression is greater than that of LPM, implying that the former is better in targeting customers with binary dependent variable.

15. Harry assumes that the boost in sales for Amazon will be the same as John Lewis's after adopting the same 3 ways. Firstly, it is a strong and very precise statement which is not backed by any data or predictions, which may be inaccurate. Secondly, Tom asks specifically about customer development anf customer churn management, which are not directly correlated with sales. Rather, free-shipping, price discounts, and interest-free installment plan may incentivise existing customers to buy more for the Christmas period, or bring in new customers to Amazon who would want to benefit from saving money. Hence, Harry's suggestion is a better response to increasing customer base (most likely, for the short term), rather than customer loyalty over the long run.

Ron wants to train predictive models based on *all* customers in the dataset which is imprecise, as we would want to train the models based on the most responsive customers, whose response rate would be higher than breakeven (i.e. "is_target_RFM" == 1). Also, if we want to have the best one, we don't have to "pick" it, as Ron said - we could just run an automated model under unsupervised learning for as long as possible to give us the best model it could find.

Either way, targeted churn management is better to be proactive, i.e. contact customers before they churn using machine learning models, instead of calculating the aftermath of caveats in customer development in reactive targeted churn management, not to mention that the latter is more costly.

16. The first fundamental tradeoff in predictive analytics is accuracy versus interpretability. It means, that when building a CRM model, we should first decide the level of interpretation's complexity we want to get, as it affects which model we are going to build. As stated before, "simpler models are easier to interpret but gives lower accuracy; complicated models can give better prediction accuracy but results may not be intuitive to interpret". (Marketing lecture slides, Wei, 2021) We should also take into account that coefficients in OLS regression have economic meanings that can measure the marginal effect of X, while in deep learning, they [estimated weights] don't.

The second fundamental tradeoff in predictive analytics is bias versus variance (underfitting versus overfitting). "Overfitting means the predictive model heavily favors historical data points and hence is not flexible enough for future data points. Underfitting occurs when a predictive model cannot adequately capture the underlying structure of the data", hence it is over-flexible. (Marketing lecture slides, Wei, 2021) In other words, overfitting model fits the datapoints too perfectly due to selection bias, while underfitting model is so "relaxed" that it is no goof gor predicting future datapoints at all either. Such models tend to result in poor predictive performance. In order to avoid this, we should build CRM model by starting with division of the full dataset into different sets: (1) training set, (2) validation set, (3) test set, depending on the purpose of the model.

17. "Next product to buy" model is about recommending the right products to right customers; we need analytics to advance the prediction accuracy. These are models for making up-selling and cross-selling products.

"Up-selling is the practice of encouraging customers to purchase a comparable higher-end product than the one customer has purchased." (Marketing lecture slides, Wei, 2021) It is also about which customers NOT to reach out to - we are targeting loyal customers who have enough money to buy our new upgraded product.

Cross-selling identifies products that satisfy additional, complementary needs that are unfulfilled by the original item.

Step 1: Compile data needed. Step 2: Selecting an appropriate statistical/predictive model. Step 3: Estimating and evaluating the model. Step 4: Scoring and targeting customers. Step 5: Decide a decision rule.