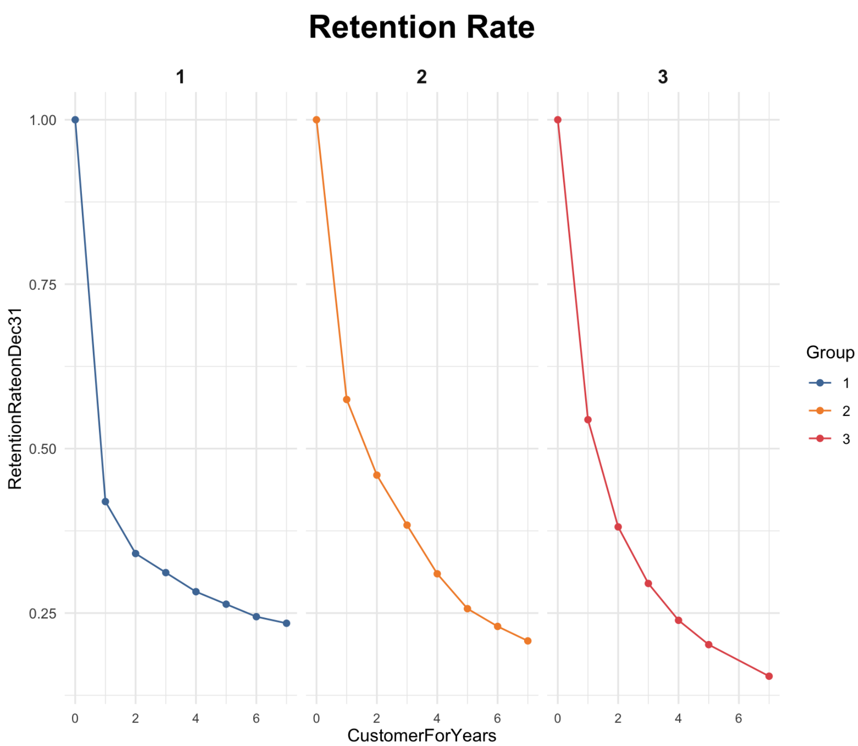
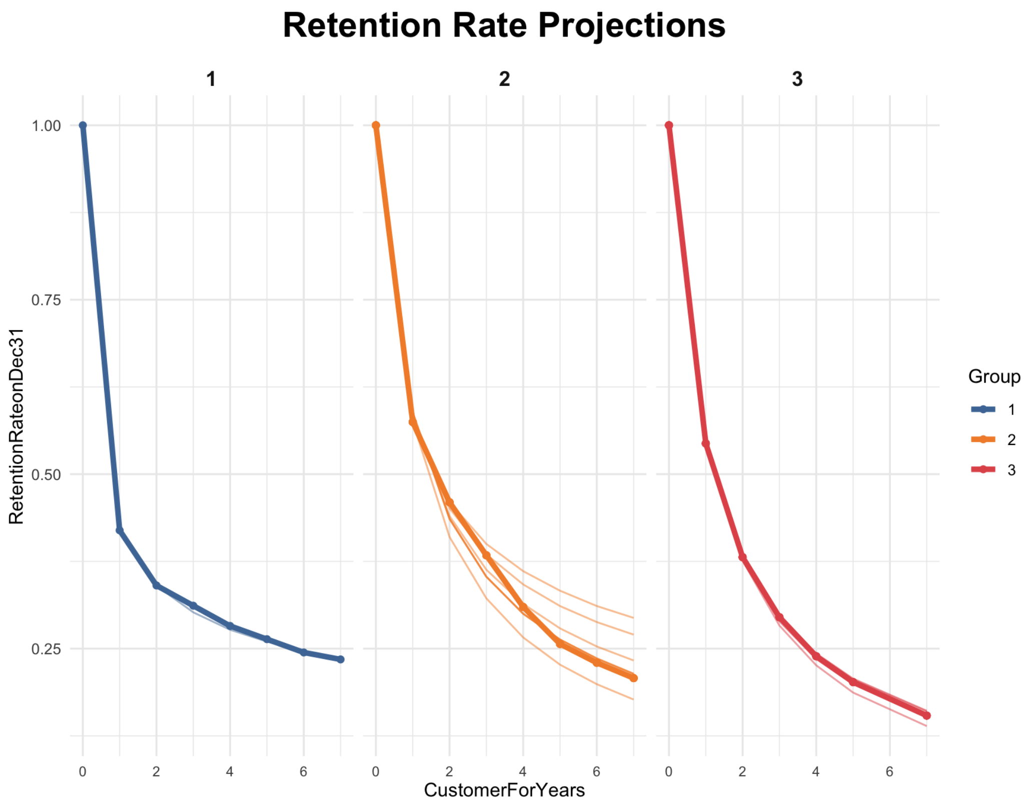
In this assignment, I was given information on retention rates for different groups of customers who purchased car insurance. I was also given the average yearly profit for each customer group (Group 1 = \$250/year, Group 2 = \$311/year, Group 3 = \$279/year). Given this information, I used the Fader and Hardie sBG distribution method to calculate the customer lifetime value, given that the expected tenure of the customer at the firm is 12 years.

First, I produced retention curves with the given data. Those graphs are shown below.



In these graphs, we are able to see a reflection of the basic data (customer group and retention rate) that the insurance company provided us with. We can see that for all groups, the retention tapers as time goes on, which makes sense. Notably, Group 3's retention decreases the quickest, while Group 1's retention decreases the slowest. Group 2's retention curve is somewhere in between Group 1 and Group 3. These graphs only show us about 7 years worth of data.

Next, we applied the sBG distribution function to create predicted retention curves. The mean average percentage error, shown visually in the plots below, tells us how far off from the original curves the predicted curves are.



We can see that even with only supplying the model a few years worth of data, the sBG distribution function created predictions that were extremely accurate. This is especially true when looking at Group 1 and Group 3, as the thin 'prediction' lines are almost difficult to see because they are so close to the 'observed' curve. With Group 2, there is a larger difference between predicted and observed, but the curves are still very similar. Due to the small difference between predicted and actual for all three groups, these predictions would likely be satisfactory for the insurance company.

Next, we can easily calculate the lifetime value of the customers, again assuming a 12 year lifetime with the firm, and the given yearly profits per group. We did this for each group, results are below.

Group 1

```
> df_ltv_01
# A tibble: 13 x 6
   CustomerForYears RetentionRateonDec31 retention_pred RetentionRateonDec31_calc ltv_monthly ltv_cum
              <int>                <dbl>          <dbl>                     <dbl>       <dbl>   <dbl>
 1                0                1             NA                         1           250     250
 2                1                0.420         NA                         0.420       105.    355.
 3                2                0.340         NA                         0.340        85.1   440
 4                3               NA              0.302                     0.302        75.5   516.
 5                4               NA              0.278                     0.278        69.5   585
 6                5               NA              0.261                     0.261        65.2   650.
 7                6               NA              0.247                     0.247        61.8   712
 8                7               NA              0.237                     0.237        59.2   771.
 9                8               NA              0.228                     0.228        57     828.
10                9               NA              0.221                     0.221        55.2   884.
11               10               NA              0.214                     0.214        53.5   937
12               11               NA              0.208                     0.208        52     989
13               12               NA              0.203                     0.203        50.8  1040.
>
```

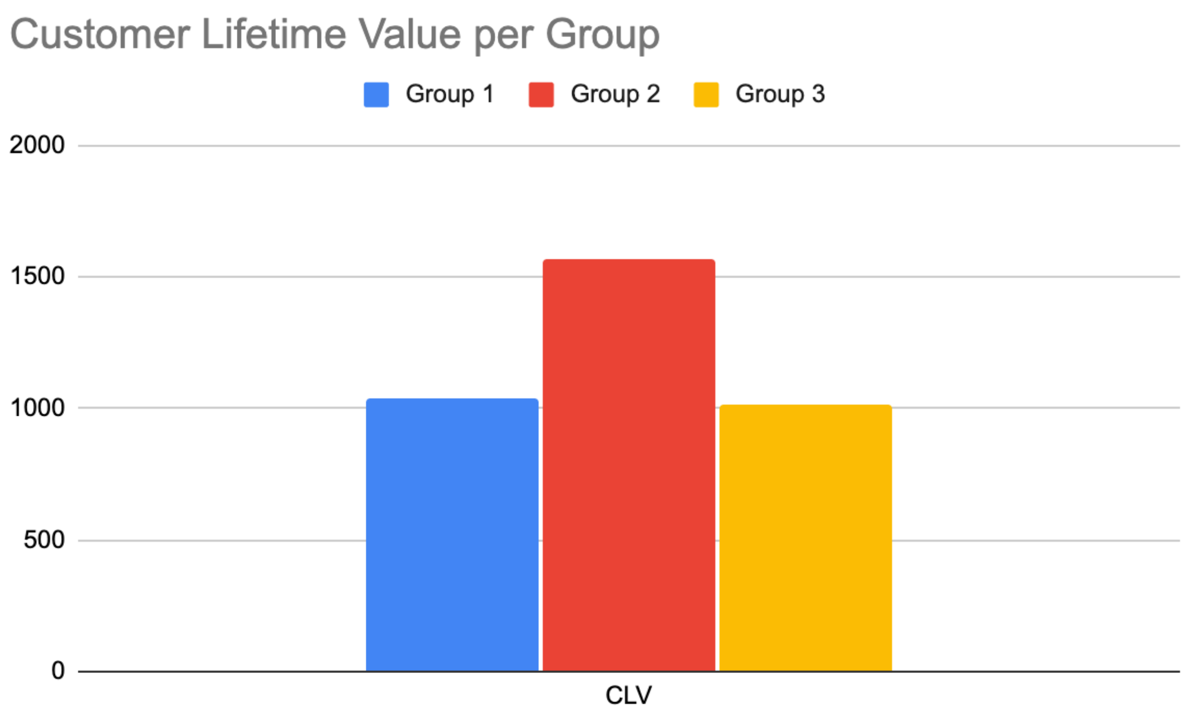
Group 2

```
> df_ltv_02
# A tibble: 13 x 6
   CustomerForYears RetentionRateonDec31 retention_pred RetentionRateonDec31_calc ltv_monthly ltv_cum
              <int>                <dbl>          <dbl>                     <dbl>       <dbl>   <dbl>
 1                0                1             NA                         1           311     311
 2                1                0.575         NA                         0.575       179.    490.
 3                2                0.460         NA                         0.460       143.    633.
 4                3               NA              0.4                       0.4         124.    757.
 5                4               NA              0.361                     0.361       112.    869.
 6                5               NA              0.333                     0.333       104.    973.
 7                6               NA              0.311                     0.311        96.7  1070.
 8                7               NA              0.294                     0.294        91.4  1161.
 9                8               NA              0.28                      0.28         87.1  1248.
10                9               NA              0.268                     0.268        83.3  1332.
11               10               NA              0.258                     0.258        80.2  1412.
12               11               NA              0.249                     0.249        77.4  1489.
13               12               NA              0.241                     0.241        75.0  1564.
>
```

Group 3

```
> df_ltv_03
# A tibble: 13 x 6
   CustomerForYears RetentionRateonDec31 retention_pred RetentionRateonDec31_calc ltv_monthly ltv_cum
              <int>                <dbl>          <dbl>                     <dbl>       <dbl>   <dbl>
 1                0                1             NA                         1           279     279
 2                1                0.544         NA                         0.544       152.    431.
 3                2                0.381         NA                         0.381       106.    537.
 4                3               NA              0.296                     0.296        82.6   620.
 5                4               NA              0.243                     0.243        67.8   687.
 6                5               NA              0.207                     0.207        57.8   745.
 7                6               NA              0.181                     0.181        50.5   796.
 8                7               NA              0.161                     0.161        44.9   841.
 9                8               NA              0.145                     0.145        40.5   881.
10                9               NA              0.133                     0.133        37.1   918.
11               10               NA              0.122                     0.122        34.0   952.
12               11               NA              0.113                     0.113        31.5   984.
13               12               NA              0.105                     0.105        29.3  1013.
>
```

To summarize the above tables, we can see that over 12 years, the customer lifetime value for Group 1 is \$1040, the CLV for Group 2 is \$1564, and the CLV for Group 3 is \$1013. This information is reflected in the graph below created in Excel for easy absorption.



These results make sense if we think about the data we were given. Group 2 has the highest profit of the groups by far, and the original retention curve showed a 'medium' taper relative to the other two groups, therefore it makes sense that it has the highest CLV. Group 1 has the lowest profit, but the best retention curve of the three groups, and Group 3 has a higher profit, but a worse retention curve. So it also makes sense that these two groups would have similar CLV.

My recommendation for the company would be to spend extra time acquiring customers that would fall into Group 2, as they are the most profitable. If the company could lower costs for Group 1, they would also be a good group to target further, as their retention rate does not decrease very quickly relative to the other groups. However, since all groups appear to be profitable, I would continue to market to all groups in general as well.

```
library(dplyr)
library(reshape2)
library(ggplot2)

# reading in the file
df_ret <- read.csv("/Users/dana/Downloads/CarInsurance.csv")

head(df_ret)
str(df_ret)

df_ret <- df_ret[df_ret$Group != 4, ]

# there is only one customer in group 4. Lets remove it from the df
df_ret$Group <- as.character(df_ret$Group)
str(df_ret)


# plotting the retention curves for the four cases we have in the dataset
# color values are optional
ggplot(df_ret, aes(x = CustomerForYears, y = RetentionRateonDec31, group = Group, color =
Group)) +
  theme_minimal() +
  facet_wrap(~ Group) +
  scale_color_manual(values = c('#4e79a7', '#f28e2b', '#e15759', '#76b7b2')) +
  geom_line() +geom_point() +
  theme(plot.title = element_text(size = 20, face = 'bold', vjust = 2, hjust = 0.5),
        axis.text.x = element_text(size = 8, hjust = 0.5, vjust = .5, face = 'plain'),
        strip.text = element_text(face = 'bold', size = 12)) +
  ggtitle('Retention Rate')

# the following section are the functions from Fader - Hardie used to create sBG dist
# functions for sBG distribution
churnBG <-Vectorize(function(alpha, beta, period) {
  t1 = alpha / (alpha + beta)
  result = t1
  if (period > 1) {
    result = churnBG(alpha, beta, period -1) * (beta + period -2) / (alpha + beta + period -1)}
  return(result)
}, vectorize.args = c("period"))

survivalBG <-Vectorize(function(alpha, beta, period) {
  t1 = 1 -churnBG(alpha, beta, 1)
  result = t1
```

```
  if(period > 1){
    result = survivalBG(alpha, beta, period -1) -churnBG(alpha, beta, period)}
  return(result)
}, vectorize.args = c("period"))

MLL <-function(alphabeta) {
  if(length(activeCust) != length(lostCust)) {
    stop("Variables activeCust and lostCust have different lengths: ",
         length(activeCust), " and ", length(lostCust), ".")
  }
  t = length(activeCust) # number of periods
  alpha = alphabeta[1]
  beta = alphabeta[2]
  return(-as.numeric(
    sum(lostCust * log(churnBG(alpha, beta, 1:t))) +
      activeCust[t]*log(survivalBG(alpha, beta, t))))}
```

```
# taking the retention data and predicting the outcomes using the Fader-Hardie functions
df_ret <-df_ret %>%group_by(Group) %>%
  mutate(activeCust = 1000 * RetentionRateonDec31,
         lostCust = lag(activeCust) -activeCust,
         lostCust = ifelse(is.na(lostCust), 0, lostCust)) %>%
  ungroup()

ret_preds01 <-vector('list', 7)
for (i in c(1:7)) {
  df_ret_filt <-df_ret %>%
    filter(between(CustomerForYears, 1, i) == TRUE & Group == '1')
  activeCust <-c(df_ret_filt$activeCust)
  lostCust <-c(df_ret_filt$lostCust)
  opt <-optim(c(1, 1), MLL)
  retention_pred <-round(c(1, survivalBG(alpha = opt$par[1], beta = opt$par[2], c(1:7))), 3)
  df_pred <-data.frame(CustomerForYears = c(0:7),
                       Group = '1',
                       fact_months = i,
                       retention_pred = retention_pred)}

ret_preds01[[i]] <-df_pred
```

```
ret_preds01 <-as.data.frame(do.call('rbind', ret_preds01))

ret_preds02 <-vector('list', 7)

for (i in c(1:7)) {
  df_ret_filt <-df_ret %>%
    filter(between(CustomerForYears, 1, i) == TRUE & Group == '2')
  activeCust <-c(df_ret_filt$activeCust)
  lostCust <-c(df_ret_filt$lostCust)
  opt <-optim(c(1, 1), MLL)
  retention_pred <-round(c(1, survivalBG(alpha = opt$par[1], beta = opt$par[2], c(1:7))), 3)
  df_pred <-data.frame(CustomerForYears = c(0:7),
                       Group = '2',
                       fact_months = i,
                       retention_pred = retention_pred)
  ret_preds02[[i]] <-df_pred
}

ret_preds02 <-as.data.frame(do.call('rbind', ret_preds02))


ret_preds03 <-vector('list', 7)

for (i in c(1:7)) {
  df_ret_filt <-df_ret %>%
    filter(between(CustomerForYears, 1, i) == TRUE & Group == '3')
  activeCust <-c(df_ret_filt$activeCust)
  lostCust <-c(df_ret_filt$lostCust)
  opt <-optim(c(1, 1), MLL)
  retention_pred <-round(c(1, survivalBG(alpha = opt$par[1], beta = opt$par[2], c(1:7))), 3)
  df_pred <-data.frame(CustomerForYears = c(0:7),
                       Group = '3',
                       fact_months = i,
                       retention_pred = retention_pred)
  ret_preds03[[i]] <-df_pred
}

ret_preds03 <-as.data.frame(do.call('rbind', ret_preds03))

ret_preds04 <-vector('list', 7)

#for (i in c(1:7)) {
#  df_ret_filt <-df_ret %>%
```

```
#    filter(between(CustomerForYears, 1, i) == TRUE & Group == '4')
#  activeCust <-c(df_ret_filt$activeCust)
#  lostCust <-c(df_ret_filt$lostCust)
#  opt <-optim(c(1, 1), MLL)
#  retention_pred <-round(c(1, survivalBG(alpha = opt$par[1], beta = opt$par[2], c(1:7))), 3)
#  df_pred <-data.frame(CustomerForYears = c(0:7),
#                        Group = '4',
#                        fact_months = i,
#                        retention_pred = retention_pred)
 # ret_preds04[[i]] <-df_pred
#}

#ret_preds04 <-as.data.frame(do.call('rbind', ret_preds04))

ret_preds <- bind_rows(ret_preds01, ret_preds02, ret_preds03) #, ret_preds04)

head(df_ret)

df_ret_all <- df_ret %>%
  dplyr::select(CustomerForYears, Group, RetentionRateonDec31) %>%
  left_join(., ret_preds, by = c('CustomerForYears', 'Group'))



# plotting the retention curves again to see how the predicted curves differ from the observed
data curves
# the visualization of the predicted retention curves and mean average percentage error
(MAPE)
# that you get as output here shows how robust the sBG approach is in completing the retention
curves
# even with the limited data
ggplot(df_ret_all, aes(x = CustomerForYears, y = RetentionRateonDec31, group = Group, color
= Group)) +
  theme_minimal() +
  facet_wrap(~ Group) +
  scale_color_manual(values = c('#4e79a7', "#f28e2b", "#e15759", "#76b7b2")) +
  geom_line(size = 1.5) +
  geom_point(size = 1.5) +
  geom_line(aes(y = retention_pred, group = fact_months), alpha = .5) +
  theme(plot.title = element_text(size = 20, face = "bold", vjust = 2, hjust = .5),
        axis.text.x = element_text(size = 8, hjust = .5, vjust = .5, face = 'plain'),
        strip.text = element_text(face = "bold", size = 12)) +
  ggtitle("Retention Rate Projections")
```

```
# CASE 3


# predicting LTV using the predicted retentions and add to the dataset
# to get this LTV prediction, we need to multiply the retention rate by the subscription
# price and calculate the cumulative amount for the required period
# we will start by  calculating the average LTV for Group 3 based on two historical months with
# a forecast horizon of 12 years and a subscription price of $279
df_ltv_03 <- df_ret %>%
  filter(between(CustomerForYears, 1,2) == TRUE & Group == '3')

activeCust <- c(df_ltv_03$activeCust)

lostCust <- c(df_ltv_03$lostCust)

opt <- optim(c(1,1), MLL)

retention_pred <- round(c(survivalBG(alpha = opt$par[1], beta = opt$par[2], c(3:12))), 3)

df_pred <- data.frame(CustomerForYears = c(3:12), retention_pred = retention_pred)

df_ltv_03 <- df_ret %>%
  filter(between(CustomerForYears, 0, 2) == TRUE & Group == '3') %>%
  dplyr::select(CustomerForYears, RetentionRateonDec31) %>%
  bind_rows(., df_pred) %>%
  mutate(RetentionRateonDec31_calc = ifelse(is.na(RetentionRateonDec31), retention_pred,
RetentionRateonDec31),
        ltv_monthly = RetentionRateonDec31_calc * 279,
        ltv_cum = round(cumsum(ltv_monthly), 2))

# examine the dataset for cumulative LTV for each case
# keep interpretation of the final output
df_ltv_03

# CASE 2

# predicting LTV using the predicted retentions and add to the dataset
# to get this LTV prediction, we need to multiply the retention rate by the subscription
# price and calculate the cumulative amount for the required period
# we will start by  calculating the average LTV for Group 2 based on two historical months with
# a forecast horizon of 12 years and a subscription price of $311
df_ltv_02 <- df_ret %>%
  filter(between(CustomerForYears, 1,2) == TRUE & Group == '2')
```

```
activeCust <- c(df_ltv_02$activeCust)

lostCust <- c(df_ltv_02$lostCust)

opt <- optim(c(1,1), MLL)

retention_pred <- round(c(survivalBG(alpha = opt$par[1], beta = opt$par[2], c(3:12))), 3)

df_pred <- data.frame(CustomerForYears = c(3:12), retention_pred = retention_pred)

df_ltv_02 <- df_ret %>%
  filter(between(CustomerForYears, 0, 2) == TRUE & Group == '2') %>%
  dplyr::select(CustomerForYears, RetentionRateonDec31) %>%
  bind_rows(., df_pred) %>%
  mutate(RetentionRateonDec31_calc = ifelse(is.na(RetentionRateonDec31), retention_pred,
RetentionRateonDec31),
         ltv_monthly = RetentionRateonDec31_calc * 311,
         ltv_cum = round(cumsum(ltv_monthly), 2))

# examine the dataset for cumulative LTV for each case
# keep interpretation of the final output
df_ltv_02

# CASE 1

# predicting LTV using the predicted retentions and add to the dataset
# to get this LTV prediction, we need to multiply the retention rate by the subscription
# price and calculate the cumulative amount for the required period
# we will start by  calculating the average LTV for Group 1 based on two historical months with
# a forecast horizon of 12 years and a subscription price of $250
df_ltv_01 <- df_ret %>%
  filter(between(CustomerForYears, 1,2) == TRUE & Group == '1')

activeCust <- c(df_ltv_01$activeCust)

lostCust <- c(df_ltv_01$lostCust)

opt <- optim(c(1,1), MLL)

retention_pred <- round(c(survivalBG(alpha = opt$par[1], beta = opt$par[2], c(3:12))), 3)

df_pred <- data.frame(CustomerForYears = c(3:12), retention_pred = retention_pred)
```

```
df_ltv_01 <- df_ret %>%
  filter(between(CustomerForYears, 0, 2) == TRUE & Group == '1') %>%
  dplyr::select(CustomerForYears, RetentionRateonDec31) %>%
  bind_rows(., df_pred) %>%
  mutate(RetentionRateonDec31_calc = ifelse(is.na(RetentionRateonDec31), retention_pred,
RetentionRateonDec31),
         ltv_monthly = RetentionRateonDec31_calc * 250,
         ltv_cum = round(cumsum(ltv_monthly), 2))

# examine the dataset for cumulative LTV for each case
# keep interpretation of the final output
df_ltv_01
```