Week 6 Assignment

2022-11-23

```
library(dplyr)
library(reshape2)
library(ggplot2)
# reading in the file
df_ret <- read.csv("F:/RIT/MKTG 768/week 6/作业/CarInsurance.csv")
head(df_ret)
```

```
##   X CustomerForYears Group RetentionRateonDec31
## 1 1                0     1               1.0000
## 2 2                1     1               0.4195
## 3 3                2     1               0.3405
## 4 4                3     1               0.3115
## 5 5                4     1               0.2825
## 6 6                5     1               0.2635
```

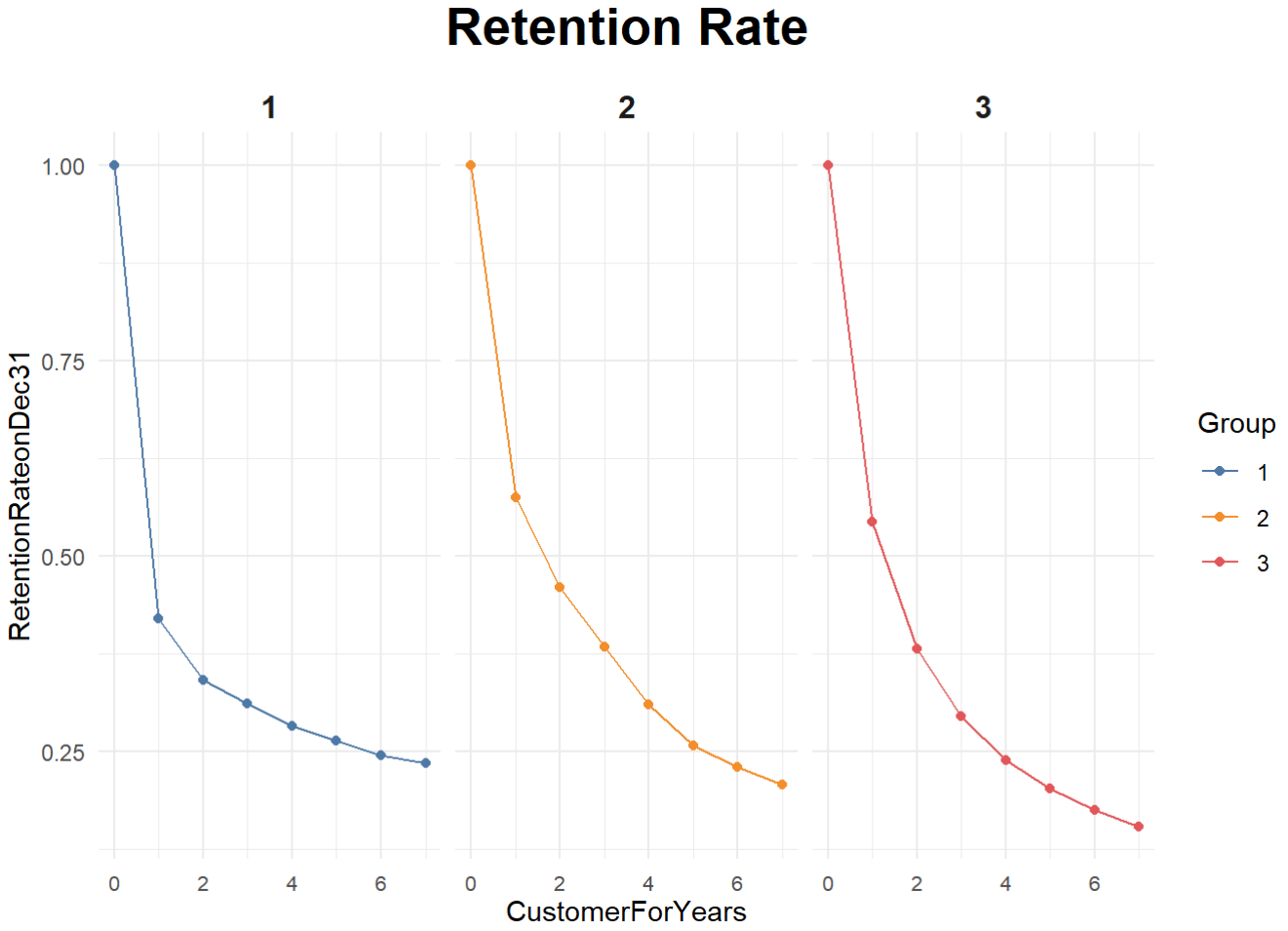
```
str(df_ret)
```

```
## 'data.frame':    24 obs. of  4 variables:
##  $ X                   : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ CustomerForYears    : int  0 1 2 3 4 5 6 7 0 1 ...
##  $ Group               : int  1 1 1 1 1 1 1 1 2 2 ...
##  $ RetentionRateonDec31: num  1 0.419 0.341 0.311 0.282 ...
```

```
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)
```

```
## 'data.frame':    24 obs. of  4 variables:
##  $ X                   : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ CustomerForYears    : int  0 1 2 3 4 5 6 7 0 1 ...
##  $ Group               : chr  "1" "1" "1" "1" ...
##  $ RetentionRateonDec31: num  1 0.419 0.341 0.311 0.282 ...
```

```
# 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')
```


Retention Rate
1
2
3
1.00
0.75
0.50
0.25
RetentionRateonDec31
0
2
4
6
CustomerForYears
Group
1
2
3

```
# 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()

## group 1
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))
```

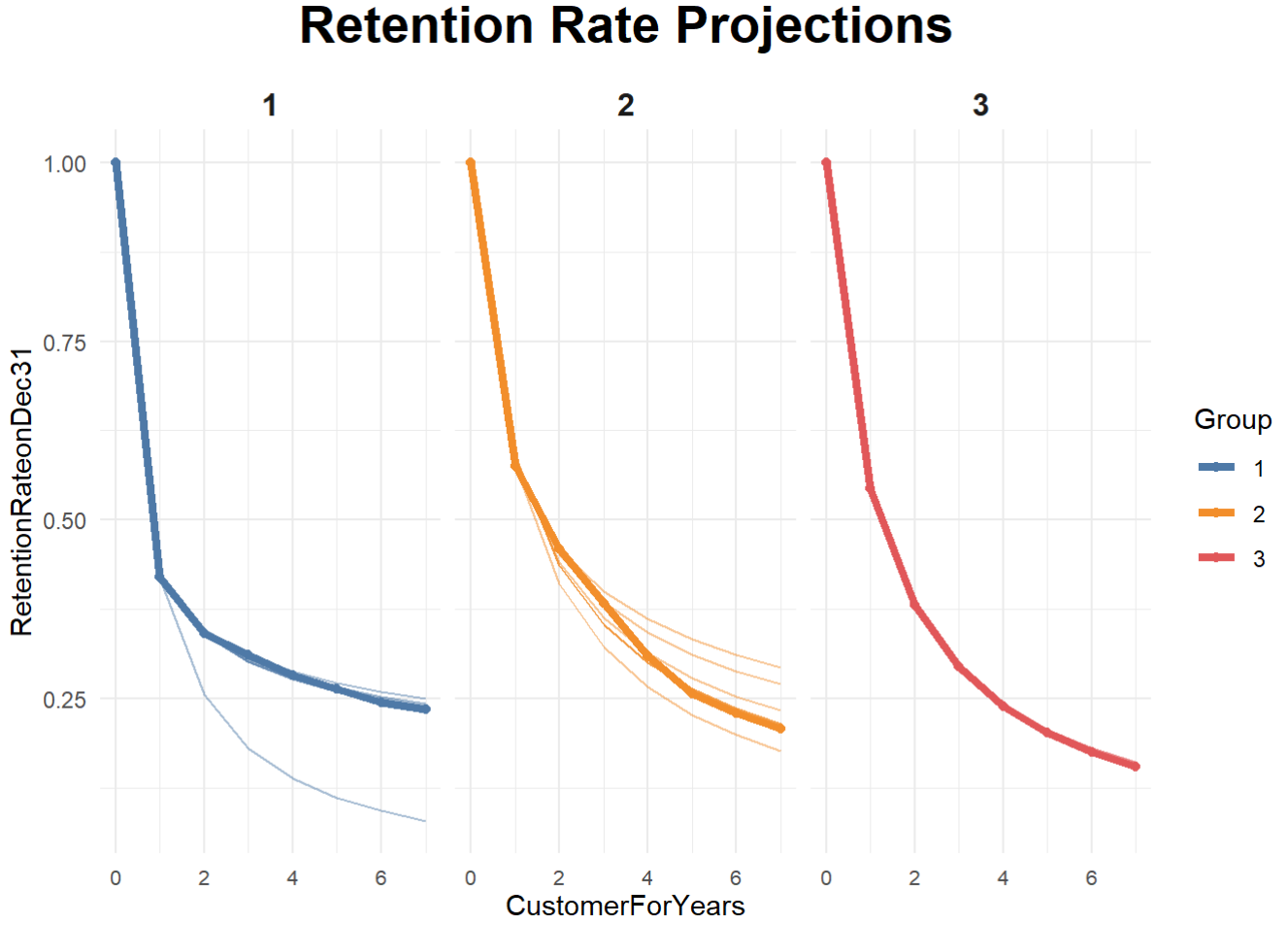
```
# group 2
ret_preds02 <-vector('list', 7)
for (i in c(1:7)) {
  df_ret_filt <-df_ret %>%
    filter(between(CustomerForYears, 1, i) == TRUE & Group == '2')
  activeCust <-c(df_ret_filt$activeCust)
  lostCust <-c(df_ret_filt$lostCust)
  opt <-optim(c(1, 1), MLL)
  retention_pred <-round(c(1, survivalBG(alpha = 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))


# group 3
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))

# combine all
ret_preds <- bind_rows(ret_preds01, ret_preds02, ret_preds03) #, ret_preds04)
head(df_ret)
```

```
## # A tibble: 6 × 6
##       X CustomerForYears Group RetentionRateonDec31 activeCust lostCust
##   <int>            <int> <chr>                <dbl>      <dbl>    <dbl>
## 1     1                0 1                    1          1000        0
## 2     2                1 1                    0.420       420.     580.
## 3     3                2 1                    0.340       340.      79
## 4     4                3 1                    0.312       312.      29
## 5     5                4 1                    0.282       282.      29
## 6     6                5 1                    0.264       264.      19
```

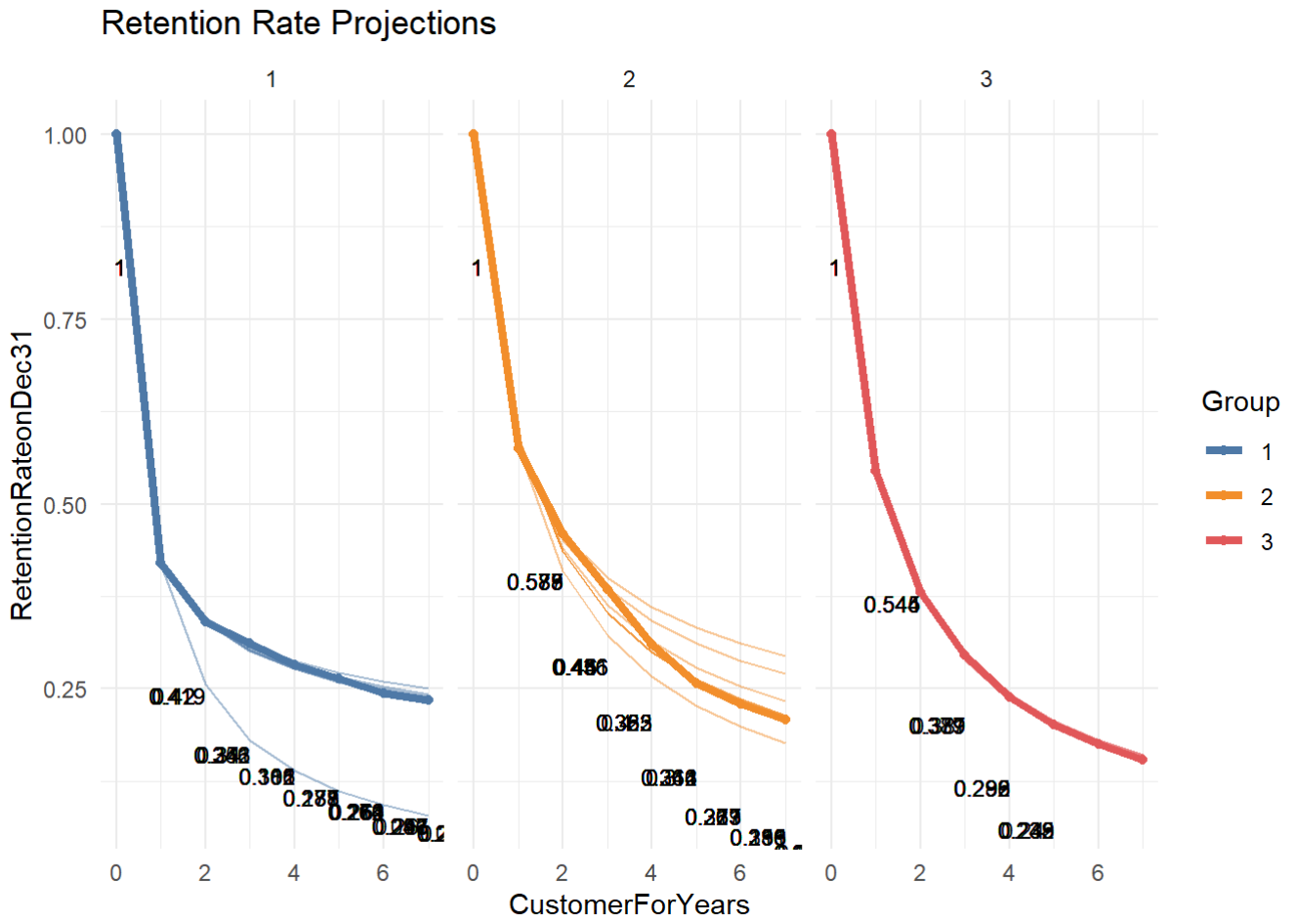
```
df_ret_all <- df_ret %>%
  dplyr::select(CustomerForYears, Group, RetentionRateonDec31) %>%
  left_join(., ret_preds, by = c('CustomerForYears', 'Group'))
head(df_ret_all)
```

```
## # A tibble: 6 × 5
##   CustomerForYears Group RetentionRateonDec31 fact_months retention_pred
##              <int> <chr>                <dbl>       <int>          <dbl>
## 1                0 1                        1           1              1
## 2                0 1                        1           2              1
## 3                0 1                        1           3              1
## 4                0 1                        1           4              1
## 5                0 1                        1           5              1
## 6                0 1                        1           6              1
```

```
#View(df_ret_all)

# 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")
```




```
# 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")+
  geom_text(aes(label=retention_pred), vjust=8, hjust=0.2,color="black", size=3)+
  theme_minimal()
```




```
# 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

# case 3
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
head(df_ltv_03)
```

```
## # A tibble: 6 × 6
##   CustomerForYears RetentionRateonDec31 retention_pred Retenti…¹ ltv_m…² 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.
## # … with abbreviated variable names ¹RetentionRateonDec31_calc, ²ltv_monthly
```

```
#View(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
head(df_ltv_02)
```

```
## # A tibble: 6 × 6
##   CustomerForYears RetentionRateonDec31 retention_pred Retenti…¹ ltv_m…² 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.
## # … with abbreviated variable names ¹RetentionRateonDec31_calc, ²ltv_monthly
```

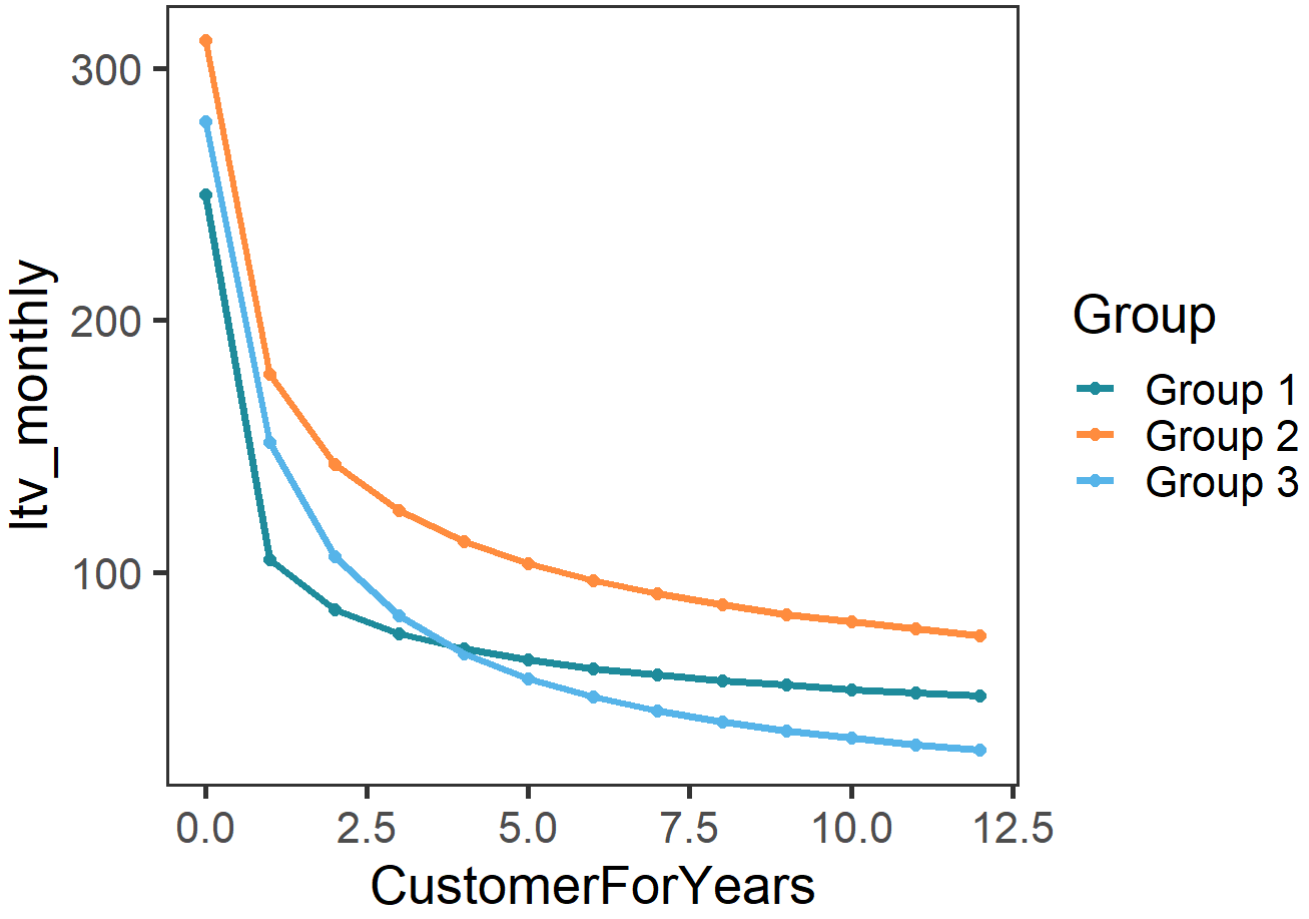
```
# CASE 1
# predicting LTV using the predicted retentions and add to the dataset
# to get this LTV prediction, we need to multiply the retention rate by the subscription
# price and calculate the cumulative amount for the required period
# we will start by  calculating the average LTV for Group 1 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
head(df_ltv_01)
```

```
## # A tibble: 6 × 6
##   CustomerForYears RetentionRateonDec31 retention_pred Retenti…¹ ltv_m…² 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.
## # … with abbreviated variable names ¹RetentionRateonDec31_calc, ²ltv_monthly
```

```
###  combine & plot

Group1<-data.frame(df_ltv_01)
#View(Group1)
Group2<-data.frame(df_ltv_02)
#View(Group2)
Group3<-data.frame(df_ltv_03)
#View(Group3)

Group1$Group<-"Group 1"
Group2$Group<-"Group 2"
Group3$Group<-"Group 3"

all00<- rbind(Group1,Group2,Group3)
head(all00)
```

```
##   CustomerForYears RetentionRateonDec31 retention_pred
## 1                0               1.0000             NA
## 2                1               0.4195             NA
## 3                2               0.3405             NA
## 4                3                   NA          0.302
## 5                4                   NA          0.278
## 6                5                   NA          0.261
##   RetentionRateonDec31_calc ltv_monthly ltv_cum   Group
## 1                    1.0000     250.000  250.00 Group 1
## 2                    0.4195     104.875  354.88 Group 1
## 3                    0.3405      85.125  440.00 Group 1
## 4                    0.3020      75.500  515.50 Group 1
## 5                    0.2780      69.500  585.00 Group 1
## 6                    0.2610      65.250  650.25 Group 1
```

```
#View(all00)

## plot 12 years trend with prediction
ggplot(all00,aes(CustomerForYears,ltv_monthly,
                 color=Group,group=Group))+
  geom_point(size=2)+
  geom_line(cex=1.3)+
  scale_color_manual(values = c('#1e8b9b','#ff8c3e',"#56B4E9"))+
  theme_test(base_size = 20)
```




```
## Plot comparison plot for analysis -- Two axis plot
Group11<- Group1[13,]
Group11
```

```
##    CustomerForYears RetentionRateonDec31 retention_pred
## 13               12                   NA          0.203
##    RetentionRateonDec31_calc ltv_monthly ltv_cum   Group
## 13                     0.203       50.75 1039.75 Group 1
```

```
Group22<- Group2[13,]
Group22
```

```
##    CustomerForYears RetentionRateonDec31 retention_pred
## 13               12                   NA          0.241
##    RetentionRateonDec31_calc ltv_monthly ltv_cum   Group
## 13                     0.241      74.951 1564.14 Group 2
```

```
Group33<- Group3[13,]
Group33
```

```
##    CustomerForYears RetentionRateonDec31 retention_pred
## 13               12                   NA          0.105
##    RetentionRateonDec31_calc ltv_monthly ltv_cum   Group
## 13                     0.105      29.295 1013.05 Group 3
```

```
Group11$Profit<-  250
Group22$Profit<-  311
Group33$Profit<-  279


all<- rbind(Group11,Group22,Group33)
all
```

```
##     CustomerForYears RetentionRateonDec31 retention_pred
## 13                12                   NA          0.203
## 131               12                   NA          0.241
## 132               12                   NA          0.105
##     RetentionRateonDec31_calc ltv_monthly ltv_cum   Group Profit
## 13                      0.203      50.750 1039.75 Group 1    250
## 131                     0.241      74.951 1564.14 Group 2    311
## 132                     0.105      29.295 1013.05 Group 3    279
```

```
#all$id<-paste(all$Group,all$CustomerForYears,sep=" ")

all$id3<- seq(1,3,1)

ggplot(all,aes(factor(Group),Profit))+
  geom_col(aes(fill=Profit),position = 'dodge',width = 0.5)+
  labs(x='Group',y='Profit')+
  geom_text(aes(label=Profit), vjust=-0.5, hjust=0.1,color="black", size=3)+
  theme_minimal()+
  scale_y_continuous(sec.axis = sec_axis(~./1000,  # 先除以1000 双轴图
                                         name = 'Cumulative CLV'))+
  geom_point(data = all,
             aes(id3,RetentionRateonDec31_calc*1000),  # 再乘以1000 双轴图
             size=3)+
  geom_line(data = all,
            aes(id3,RetentionRateonDec31_calc*1000),
            cex=1.3)+
geom_text(aes(label=RetentionRateonDec31_calc), vjust=8, hjust=0.2,color="white", size=3)+
  theme_minimal()
```