cas_2023

Lingjun Meng

2023/1/23

```r
data <- read.csv("SalesData.csv")
data <- data %>%
  mutate(price_bin = ifelse(price == " $999 ",0,1))


model <- glm(price_bin ~ year + age_group + region + gender, data = data)
summary(model)
```

```
##
## Call:
## glm(formula = price_bin ~ year + age_group + region + gender,
##     data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.7398  -0.4022  -0.2310   0.4254   0.8274
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          8.3717460 10.4863928   0.798    0.425
## year                -0.0040609  0.0051959  -0.782    0.434
## age_group18 to 25    0.0584036  0.0129086   4.524 6.11e-06 ***
## age_group26 to 65    0.2236056  0.0104879  21.320  < 2e-16 ***
## age_group65+         0.1716741  0.0163885  10.475  < 2e-16 ***
## regionNorth          0.2295031  0.0123015  18.657  < 2e-16 ***
## regionSouth          0.3353377  0.0118090  28.397  < 2e-16 ***
## regionWest           0.0584423  0.0135505   4.313 1.62e-05 ***
## genderMale          -0.0001725  0.0084016  -0.021    0.984
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.2155168)
##
##     Null deviance: 3017.5  on 12222  degrees of freedom
## Residual deviance: 2632.3  on 12214  degrees of freedom
## AIC: 15940
##
## Number of Fisher Scoring iterations: 2
```

```r
data <- data %>%
  mutate(customer_type = case_when(
    age_group == '1 to 17' &  region == "North" & gender == "Male" ~   "1",
    age_group == '1 to 17' &  region == "North" & gender == "Female" ~ "2",
```

```
    age_group == '1 to 17' &  region == "East" &  gender == "Male" ~   "3",
    age_group == '1 to 17' &  region == "East" &  gender == "Female" ~ "4",
    age_group == '1 to 17' &  region == "South" & gender == "Male" ~   "5",
    age_group == '1 to 17' &  region == "South" & gender == "Female" ~ "6",
    age_group == '1 to 17' &  region == "West" &  gender == "Male" ~   "7",
    age_group == '1 to 17' &  region == "West" &  gender == "Female" ~ "8",
    age_group == '18 to 25' & region == "North" & gender == "Male" ~   "9",
    age_group == '18 to 25' & region == "North" & gender == "Female" ~ "10",
    age_group == '18 to 25' & region == "East" &  gender == "Male" ~   "11",
    age_group == '18 to 25' & region == "East" &  gender == "Female" ~ "12",
    age_group == '18 to 25' & region == "South" & gender == "Male" ~   "13",
    age_group == '18 to 25' & region == "South" & gender == "Female" ~ "14",
    age_group == '18 to 25' & region == "West" &  gender == "Male" ~   "15",
    age_group == '18 to 25' & region == "West" &  gender == "Female" ~ "16",
    age_group == '26 to 65' & region == "North" & gender == "Male" ~   "17",
    age_group == '26 to 65' & region == "North" & gender == "Female" ~ "18",
    age_group == '26 to 65' & region == "East" &  gender == "Male" ~   "19",
    age_group == '26 to 65' & region == "East" &  gender == "Female" ~ "20",
    age_group == '26 to 65' & region == "South" & gender == "Male" ~   "21",
    age_group == '26 to 65' & region == "South" & gender == "Female" ~ "22",
    age_group == '26 to 65' & region == "West" &  gender == "Male" ~   "23",
    age_group == '26 to 65' & region == "West" &  gender == "Female" ~ "24",
    age_group == '65+' &      region == "North" & gender == "Male" ~   "25",
    age_group == '65+' &      region == "North" & gender == "Female" ~ "26",
    age_group == '65+' &      region == "East" &  gender == "Male" ~   "27",
    age_group == '65+' &      region == "East" &  gender == "Female" ~ "28",
    age_group == '65+' &      region == "South" & gender == "Male" ~   "29",
    age_group == '65+' &      region == "South" & gender == "Female" ~ "30",
    age_group == '65+' &      region == "West" &  gender == "Male" ~   "31",
    age_group == '65+' &      region == "West" &  gender == "Female" ~ "32",))


data <- data %>%
  group_by(customer_type) %>%
  mutate(Premium_Model_Protion = mean(price_bin))

population_mean <- mean(data$price_bin)


data <- data %>%
  mutate(group_scaler = Premium_Model_Protion/population_mean)

Premium_Sales <- data %>%
  filter(phone_type == "Premium")

MidTier_Sales <- data %>%
  filter(phone_type == "Mid-Tier")
```

```
Model_group <- glm(price_bin ~ customer_type, data = data)
summary(Model_group)
```

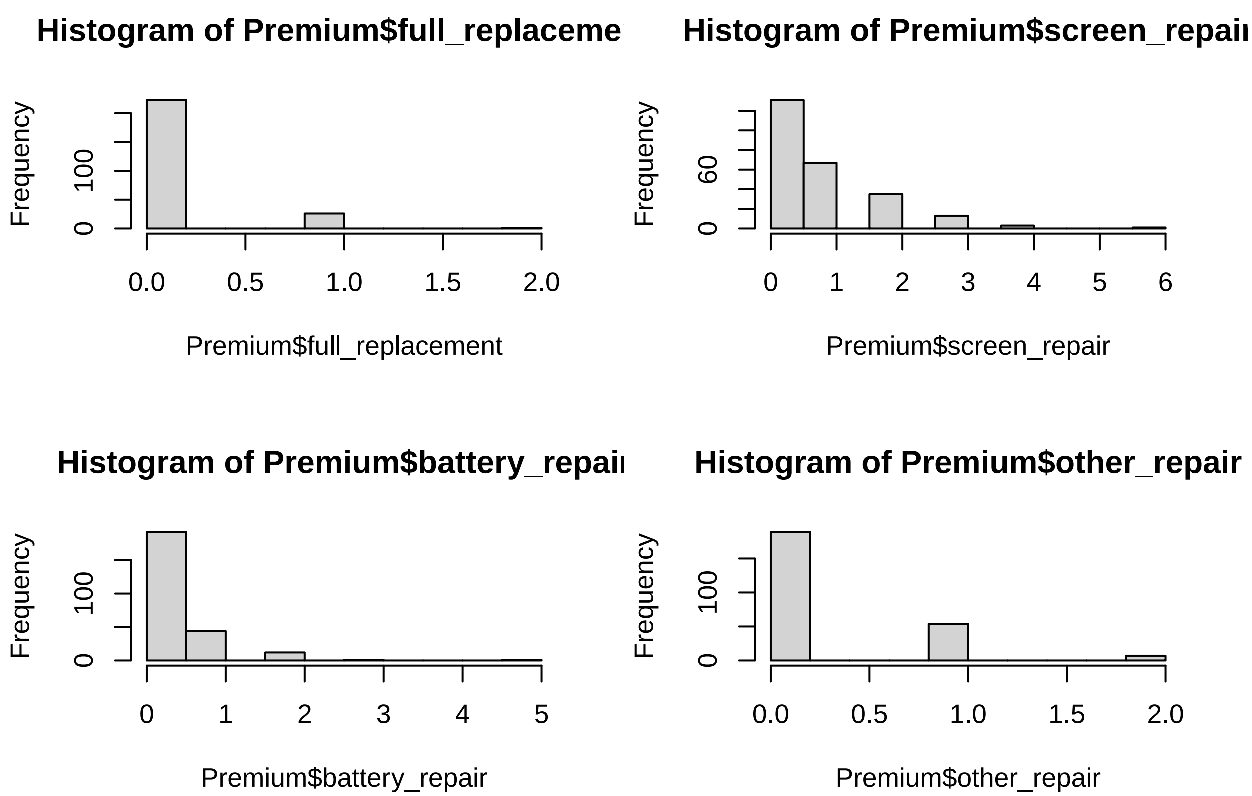
```
##
## Call:
## glm(formula = price_bin ~ customer_type, data = data)
```

```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.7742  -0.3611  -0.2187   0.3889   0.7813
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.38602    0.01803  21.416  < 2e-16 ***
## customer_type10  0.18375    0.05302   3.466 0.000530 ***
## customer_type11 -0.13232    0.02315  -5.715 1.12e-08 ***
## customer_type12 -0.16730    0.02329  -7.184 7.19e-13 ***
## customer_type13  0.22268    0.07052   3.158 0.001593 **
## customer_type14  0.22509    0.05739   3.922 8.84e-05 ***
## customer_type15 -0.14573    0.02905  -5.017 5.33e-07 ***
## customer_type16 -0.09927    0.02898  -3.425 0.000617 ***
## customer_type17  0.24164    0.02549   9.479  < 2e-16 ***
## customer_type18  0.26647    0.02527  10.547  < 2e-16 ***
## customer_type19 -0.06053    0.02728  -2.219 0.026510 *
## customer_type2  -0.02253    0.02577  -0.874 0.382124
## customer_type20 -0.04119    0.02670  -1.543 0.122928
## customer_type21  0.38822    0.02519  15.415  < 2e-16 ***
## customer_type22  0.38507    0.02472  15.577  < 2e-16 ***
## customer_type23  0.09293    0.03808   2.440 0.014687 *
## customer_type24  0.07012    0.03553   1.973 0.048466 *
## customer_type25  0.19580    0.06490   3.017 0.002558 **
## customer_type26  0.33328    0.06384   5.221 1.81e-07 ***
## customer_type27 -0.10494    0.03848  -2.727 0.006396 **
## customer_type28 -0.02491    0.03889  -0.640 0.521916
## customer_type29  0.29819    0.03450   8.643  < 2e-16 ***
## customer_type3  -0.07947    0.03100  -2.563 0.010379 *
## customer_type30  0.28203    0.03481   8.101 5.96e-16 ***
## customer_type31  0.10334    0.06981   1.480 0.138810
## customer_type32  0.06496    0.06721   0.967 0.333766
## customer_type4  -0.12044    0.03007  -4.005 6.24e-05 ***
## customer_type5   0.08239    0.02487   3.313 0.000927 ***
## customer_type6   0.09674    0.02467   3.922 8.83e-05 ***
## customer_type7  -0.07567    0.04242  -1.784 0.074451 .
## customer_type8  -0.12150    0.04128  -2.943 0.003254 **
## customer_type9   0.16844    0.04941   3.409 0.000655 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.2137872)
##
##     Null deviance: 3017.5  on 12222  degrees of freedom
## Residual deviance: 2606.3  on 12191  degrees of freedom
## AIC: 15864
##
## Number of Fisher Scoring iterations: 2
```

Reduced base premium and extra fee for extra claim

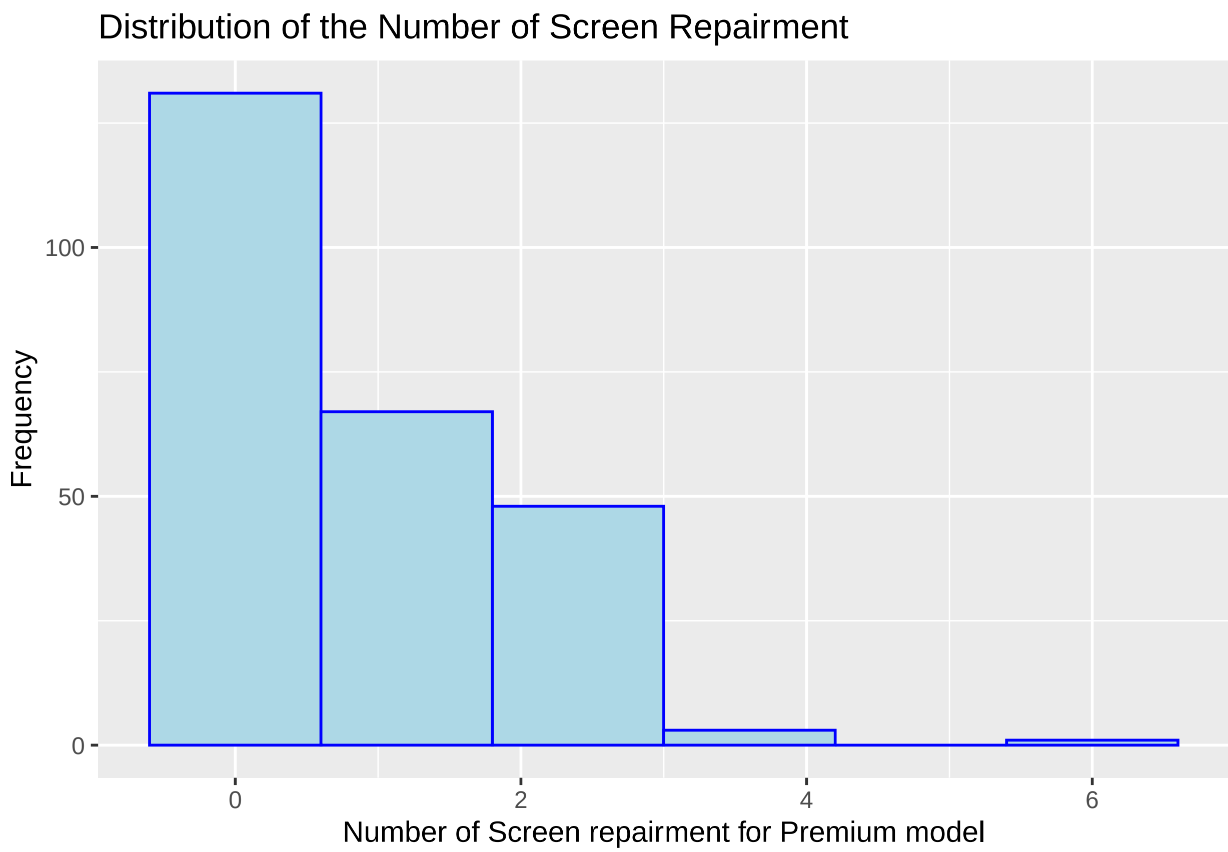
```
Claimdata <- read.csv("ClaimsByParticipantID.csv")
Premium <- Claimdata %>%
  filter(phone_type == "Premium") %>%
  mutate(total_cost = full_replacement * 650 +
         screen_repair * 200 +
         battery_repair * 150 +
         other_repair * 100)

MidTier <- Claimdata %>%
  filter(phone_type == "Mid-Tier") %>%
  mutate(total_cost = full_replacement * 450 +
         screen_repair * 150 +
         battery_repair * 125 +
         other_repair * 100)
```

```
par(mfrow=c(2,2))
hist(Premium$full_replacement)
hist(Premium$screen_repair)
hist(Premium$battery_repair)
hist(Premium$other_repair)
```

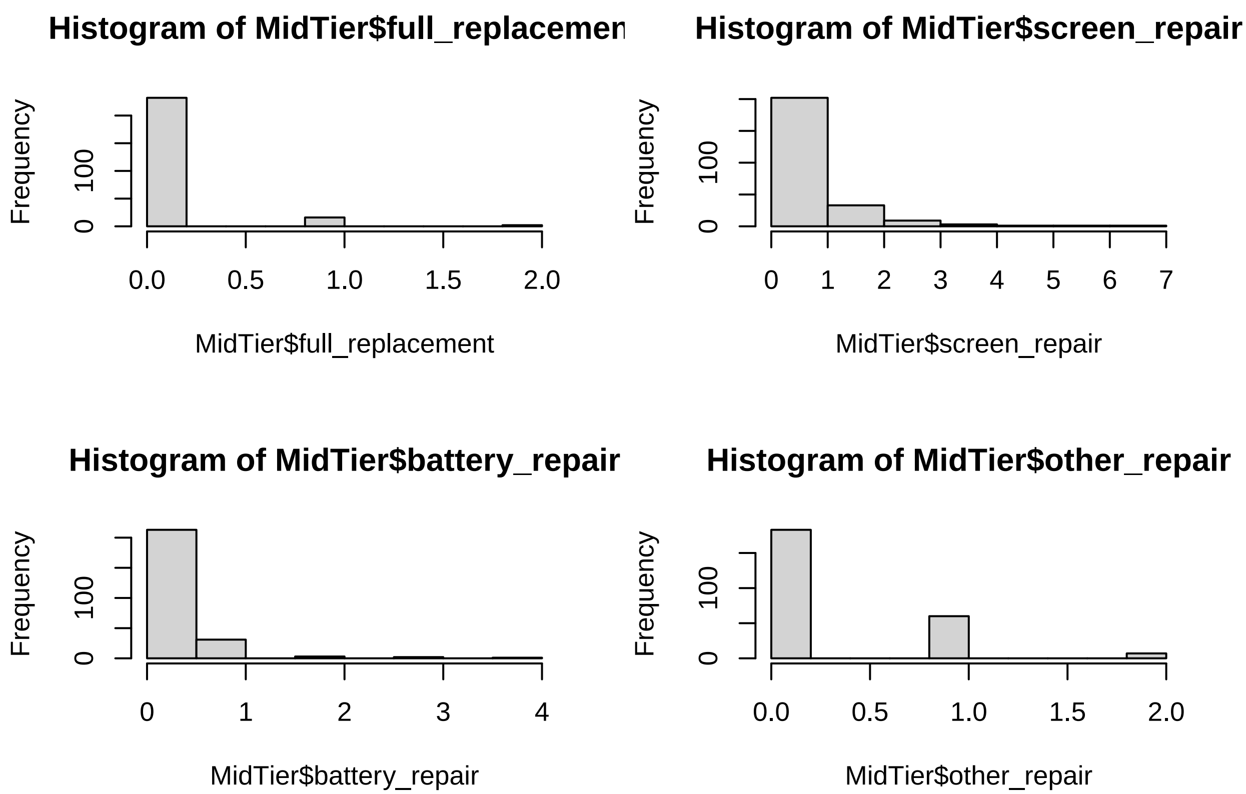



```
ggplot(data = Premium, aes(x = screen_repair)) +
  geom_histogram(bins = 6, fill = "light blue", color = "blue") +
  labs(x = "Number of Screen repairment for Premium model",
```

```
       y = "Frequency",
       title = "Distribution of the Number of Screen Repairment")
```




```
par(mfrow=c(2,2))
hist(MidTier$full_replacement)
hist(MidTier$screen_repair)
hist(MidTier$battery_repair)
hist(MidTier$other_repair)
```




```
base_fatcor <- 0.7
claim_factor <- 0.5

Mid_Base <- 200 * base_fatcor
Pre_Base <- 300 * base_fatcor



Pre_replace_cost <- 650
Pre_screen_cost <- 200
Pre_battery_cost <- 150
Pre_other_cost <- 100

Pre_Addtional_replace <- claim_factor * Pre_replace_cost
Pre_Addtional_screen <- claim_factor * Pre_screen_cost
Pre_Addtional_battery <- claim_factor * Pre_battery_cost
Pre_Addtional_other <- claim_factor * Pre_other_cost


Mid_replace_cost <- 450
Mid_screen_cost <- 150
Mid_battery_cost <- 125
Mid_other_cost <- 100


Mid_Addtional_replace <- claim_factor * Mid_replace_cost
```

```r
Mid_Addtional_screen <- claim_factor * Mid_screen_cost
Mid_Addtional_battery <- claim_factor * Mid_battery_cost
Mid_Addtional_other <- claim_factor * Mid_other_cost
```

```r
#Assume that all the claim frequencies are poisson distributed
#We checked the mean and the variance of the sample and the assumption seems to hold

mean(MidTier$full_replacement)
```

```
## [1] 0.08
```

```r
var(MidTier$full_replacement)
```

```
## [1] 0.08995984
```

```r
mean(MidTier$screen_repair)
```

```
## [1] 0.744
```

```r
var(MidTier$screen_repair)
```

```
## [1] 1.187213
```

```r
mean(MidTier$battery_repair)
```

```
## [1] 0.188
```

```r
var(MidTier$battery_repair)
```

```
## [1] 0.273751
```

```r
mean(MidTier$other_repair)
```

```
## [1] 0.296
```

```r
var(MidTier$other_repair)
```

```
## [1] 0.2654458
```

```r
mean(Premium$full_replacement)
```

```
## [1] 0.112
```

```r
var(Premium$full_replacement)
```

```
## [1] 0.1078876
```

```r
mean(Premium$screen_repair)
```

```
## [1] 0.776
```

```r
var(Premium$screen_repair)
```

```
## [1] 1.03396
```

```r
mean(Premium$battery_repair)
```

```
## [1] 0.304
```

```r
var(Premium$battery_repair)
```

```
## [1] 0.4132369
```

```r
mean(Premium$other_repair)
```

```
## [1] 0.272
```

```r
var(Premium$other_repair)
```

```
## [1] 0.2550361
```

```r
loss_ratio <- NULL

for (i in 1:1000){
  Sim_Mid_replace <- rpois(5428, mean(MidTier$full_replacement))
  Sim_Mid_screen <- rpois(5428, mean(MidTier$screen_repair))
  Sim_Mid_battery <- rpois(5428, mean(MidTier$battery_repair))
  Sim_Mid_other <- rpois(5428, mean(MidTier$other_repair))

  Sim_Pre_replace <- rpois(6795, mean(Premium$full_replacement))
  Sim_Pre_screen <- rpois(6795, mean(Premium$screen_repair))
  Sim_Pre_battery <- rpois(6795, mean(Premium$battery_repair))
  Sim_Pre_other <- rpois(6795, mean(Premium$other_repair))

  Mid_Premium_Earned <- Mid_Base * MidTier_Sales$group_scaler +
    Sim_Mid_replace * Mid_Addtional_replace +
    Sim_Mid_screen * Mid_Addtional_screen +
    Sim_Mid_battery * Mid_Addtional_battery +
    Sim_Mid_other * Mid_Addtional_other

  Mid_total_cost <- Sim_Mid_replace * Mid_replace_cost +
    Sim_Mid_screen * Mid_screen_cost +
    Sim_Mid_battery * Mid_battery_cost +
    Sim_Mid_other * Mid_other_cost


  Pre_Premium_Earned <- Pre_Base * Premium_Sales$group_scaler +
    Sim_Pre_replace * Pre_Addtional_replace +
    Sim_Pre_screen * Pre_Addtional_screen +
    Sim_Pre_battery * Pre_Addtional_battery +
    Sim_Pre_other * Pre_Addtional_other

  Pre_total_cost <- Sim_Pre_replace * Pre_replace_cost +
    Sim_Pre_screen * Pre_screen_cost +
    Sim_Pre_battery * Pre_battery_cost +
    Sim_Pre_other * Pre_other_cost

  loss_ratio[i] <- (sum(Mid_total_cost) + sum(Pre_total_cost))/(sum(Mid_Premium_Earned) + sum(Pre_Premi
}
```

```r
mean(MidTier$total_cost)
```

```
## [1] 200.7
```

```r
mean(Premium$total_cost)
```

```
## [1] 300.8
```

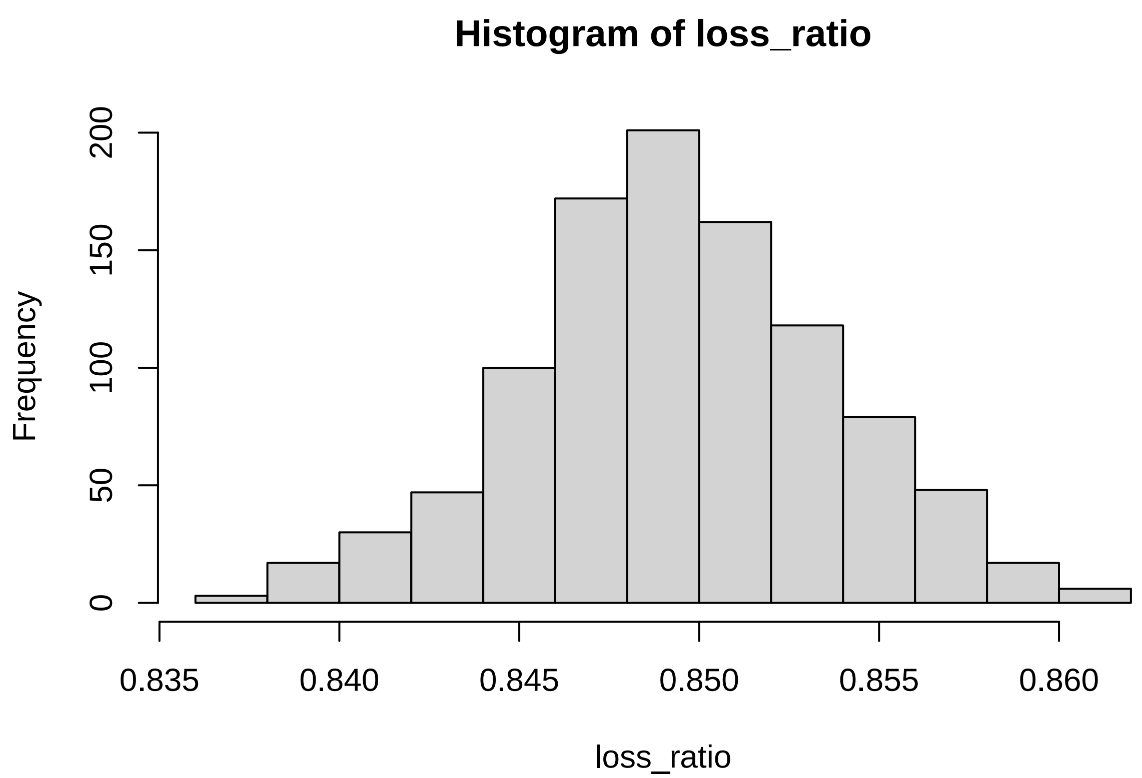
```r
mean(Pre_total_cost)
```

```
## [1] 301.3687
```

```r
mean(Mid_total_cost)
```

```
## [1] 202.2062
```

```r
hist(loss_ratio)
```



Frequncy Limit

```r
set.seed(1223)

Sim_Mid_replace <- rpois(5428, mean(MidTier$full_replacement))
Sim_Mid_screen <- rpois(5428, mean(MidTier$screen_repair))
Sim_Mid_battery <- rpois(5428, mean(MidTier$battery_repair))
Sim_Mid_other <- rpois(5428, mean(MidTier$other_repair))

Sim_Pre_replace <- rpois(6795, mean(Premium$full_replacement))
Sim_Pre_screen <- rpois(6795, mean(Premium$screen_repair))
Sim_Pre_battery <- rpois(6795, mean(Premium$battery_repair))
Sim_Pre_other <- rpois(6795, mean(Premium$other_repair))

Mid_Premium_Earned <- Mid_Base * MidTier_Sales$group_scaler +
  Sim_Mid_replace * Mid_Addtional_replace +
  Sim_Mid_screen * Mid_Addtional_screen +
  Sim_Mid_battery * Mid_Addtional_battery +
```

```r
  Sim_Mid_other * Mid_Addtional_other

Mid_total_cost <- Sim_Mid_replace * Mid_replace_cost +
  Sim_Mid_screen * Mid_screen_cost +
  Sim_Mid_battery * Mid_battery_cost +
  Sim_Mid_other * Mid_other_cost


Pre_Premium_Earned <- Pre_Base * Premium_Sales$group_scaler +
  Sim_Pre_replace * Pre_Addtional_replace +
  Sim_Pre_screen * Pre_Addtional_screen +
  Sim_Pre_battery * Pre_Addtional_battery +
  Sim_Pre_other * Pre_Addtional_other

Pre_total_cost <- Sim_Pre_replace * Pre_replace_cost +
  Sim_Pre_screen * Pre_screen_cost +
  Sim_Pre_battery * Pre_battery_cost +
  Sim_Pre_other * Pre_other_cost

# If we set a frequency limit of 3 for each of the 4 types of repairment:
limit <- 3


Sim_Mid_replace_limited <- NULL
for(i in 1:5248){ifelse(Sim_Mid_replace[i]>limit,
                        Sim_Mid_replace_limited[i] <- limit,
                        Sim_Mid_replace_limited[i] <- Sim_Mid_replace[i])}

Sim_Mid_screen_limited <- NULL
for(i in 1:5248){ifelse(Sim_Mid_screen[i]>limit,
                        Sim_Mid_screen_limited[i] <- limit,
                        Sim_Mid_screen_limited[i] <- Sim_Mid_screen[i])}

Sim_Mid_battery_limited <- NULL
for(i in 1:5248){ifelse(Sim_Mid_battery[i]>limit,
                        Sim_Mid_battery_limited[i] <- limit,
                        Sim_Mid_battery_limited[i] <- Sim_Mid_battery[i])}

Sim_Mid_other_limited <- NULL
for(i in 1:5248){ifelse(Sim_Mid_other[i]>limit,
                        Sim_Mid_other_limited[i] <- limit,
                        Sim_Mid_other_limited[i] <- Sim_Mid_other[i])}


Mid_total_cost_limited <- Sim_Mid_replace_limited * Mid_replace_cost +
  Sim_Mid_screen_limited * Mid_screen_cost +
  Sim_Mid_battery_limited * Mid_battery_cost +
  Sim_Mid_other_limited * Mid_other_cost



Sim_Pre_replace_limited <- NULL
for(i in 1:6795){ifelse(Sim_Pre_replace[i]>limit,
```

```
                        Sim_Pre_replace_limited[i] <- limit,
                        Sim_Pre_replace_limited[i] <- Sim_Pre_replace[i])}

Sim_Pre_screen_limited <- NULL
for(i in 1:6795){ifelse(Sim_Pre_screen[i]>limit,
                        Sim_Pre_screen_limited[i] <- limit,
                        Sim_Pre_screen_limited[i] <- Sim_Pre_screen[i])}

Sim_Pre_battery_limited <- NULL
for(i in 1:6795){ifelse(Sim_Pre_battery[i]>limit,
                        Sim_Pre_battery_limited[i] <- limit,
                        Sim_Pre_battery_limited[i] <- Sim_Pre_battery[i])}

Sim_Pre_other_limited <- NULL
for(i in 1:6795){ifelse(Sim_Pre_other[i]>limit,
                        Sim_Pre_other_limited[i] <- limit,
                        Sim_Pre_other_limited[i] <- Sim_Pre_other[i])}

Pre_total_cost_limited <- Sim_Pre_replace_limited * Pre_replace_cost +
  Sim_Pre_screen_limited * Pre_screen_cost +
  Sim_Pre_battery_limited * Pre_battery_cost +
  Sim_Pre_other_limited * Pre_other_cost


loss_eliminated = 1 - (sum(Mid_total_cost_limited) + sum(Pre_total_cost_limited))/(sum(Mid_total_cost) +

loss_eliminated
```

```
## [1] 0.01615456
```

```
set.seed(1223)

Sim_Mid_replace <- rpois(5428, mean(MidTier$full_replacement))
Sim_Mid_screen <- rpois(5428, mean(MidTier$screen_repair))
Sim_Mid_battery <- rpois(5428, mean(MidTier$battery_repair))
Sim_Mid_other <- rpois(5428, mean(MidTier$other_repair))

Sim_Pre_replace <- rpois(6795, mean(Premium$full_replacement))
Sim_Pre_screen <- rpois(6795, mean(Premium$screen_repair))
Sim_Pre_battery <- rpois(6795, mean(Premium$battery_repair))
Sim_Pre_other <- rpois(6795, mean(Premium$other_repair))

Mid_Premium_Earned <- Mid_Base * MidTier_Sales$group_scaler +
  Sim_Mid_replace * Mid_Addtional_replace +
  Sim_Mid_screen * Mid_Addtional_screen +
  Sim_Mid_battery * Mid_Addtional_battery +
  Sim_Mid_other * Mid_Addtional_other

Mid_total_cost <- Sim_Mid_replace * Mid_replace_cost +
  Sim_Mid_screen * Mid_screen_cost +
  Sim_Mid_battery * Mid_battery_cost +
  Sim_Mid_other * Mid_other_cost
```

```r
Pre_Premium_Earned <- Pre_Base * Premium_Sales$group_scaler +
  Sim_Pre_replace * Pre_Addtional_replace +
  Sim_Pre_screen * Pre_Addtional_screen +
  Sim_Pre_battery * Pre_Addtional_battery +
  Sim_Pre_other * Pre_Addtional_other

Pre_total_cost <- Sim_Pre_replace * Pre_replace_cost +
  Sim_Pre_screen * Pre_screen_cost +
  Sim_Pre_battery * Pre_battery_cost +
  Sim_Pre_other * Pre_other_cost

# If we set a frequency limit of 2 for each of the 4 types of repairment:
limit <- 2


Sim_Mid_replace_limited <- NULL
for(i in 1:5248){ifelse(Sim_Mid_replace[i]>limit,
                        Sim_Mid_replace_limited[i] <- limit,
                        Sim_Mid_replace_limited[i] <- Sim_Mid_replace[i])}

Sim_Mid_screen_limited <- NULL
for(i in 1:5248){ifelse(Sim_Mid_screen[i]>limit,
                        Sim_Mid_screen_limited[i] <- limit,
                        Sim_Mid_screen_limited[i] <- Sim_Mid_screen[i])}

Sim_Mid_battery_limited <- NULL
for(i in 1:5248){ifelse(Sim_Mid_battery[i]>limit,
                        Sim_Mid_battery_limited[i] <- limit,
                        Sim_Mid_battery_limited[i] <- Sim_Mid_battery[i])}

Sim_Mid_other_limited <- NULL
for(i in 1:5248){ifelse(Sim_Mid_other[i]>limit,
                        Sim_Mid_other_limited[i] <- limit,
                        Sim_Mid_other_limited[i] <- Sim_Mid_other[i])}


Mid_total_cost_limited <- Sim_Mid_replace_limited * Mid_replace_cost +
  Sim_Mid_screen_limited * Mid_screen_cost +
  Sim_Mid_battery_limited * Mid_battery_cost +
  Sim_Mid_other_limited * Mid_other_cost



Sim_Pre_replace_limited <- NULL
for(i in 1:6795){ifelse(Sim_Pre_replace[i]>limit,
                        Sim_Pre_replace_limited[i] <- limit,
                        Sim_Pre_replace_limited[i] <- Sim_Pre_replace[i])}

Sim_Pre_screen_limited <- NULL
for(i in 1:6795){ifelse(Sim_Pre_screen[i]>limit,
                        Sim_Pre_screen_limited[i] <- limit,
                        Sim_Pre_screen_limited[i] <- Sim_Pre_screen[i])}
```

```
Sim_Pre_battery_limited <- NULL
for(i in 1:6795){ifelse(Sim_Pre_battery[i]>limit,
                        Sim_Pre_battery_limited[i] <- limit,
                        Sim_Pre_battery_limited[i] <- Sim_Pre_battery[i])}

Sim_Pre_other_limited <- NULL
for(i in 1:6795){ifelse(Sim_Pre_other[i]>limit,
                        Sim_Pre_other_limited[i] <- limit,
                        Sim_Pre_other_limited[i] <- Sim_Pre_other[i])}

Pre_total_cost_limited <- Sim_Pre_replace_limited * Pre_replace_cost +
  Sim_Pre_screen_limited * Pre_screen_cost +
  Sim_Pre_battery_limited * Pre_battery_cost +
  Sim_Pre_other_limited * Pre_other_cost


loss_eliminated = 1 - (sum(Mid_total_cost_limited) + sum(Pre_total_cost_limited))/(sum(Mid_total_cost) +

loss_eliminated
```

```
## [1] 0.04860756
```

Partially Refund

```
#Here we try to refund a portion of the premium to individuals who had 0 total claim
set.seed(1223)

Sim_Mid_replace <- rpois(5428, mean(MidTier$full_replacement))
Sim_Mid_screen <- rpois(5428, mean(MidTier$screen_repair))
Sim_Mid_battery <- rpois(5428, mean(MidTier$battery_repair))
Sim_Mid_other <- rpois(5428, mean(MidTier$other_repair))

Sim_Pre_replace <- rpois(6795, mean(Premium$full_replacement))
Sim_Pre_screen <- rpois(6795, mean(Premium$screen_repair))
Sim_Pre_battery <- rpois(6795, mean(Premium$battery_repair))
Sim_Pre_other <- rpois(6795, mean(Premium$other_repair))

Mid_Premium_Earned <- Mid_Base * MidTier_Sales$group_scaler +
  Sim_Mid_replace * Mid_Addtional_replace +
  Sim_Mid_screen * Mid_Addtional_screen +
  Sim_Mid_battery * Mid_Addtional_battery +
  Sim_Mid_other * Mid_Addtional_other

Mid_total_cost <- Sim_Mid_replace * Mid_replace_cost +
  Sim_Mid_screen * Mid_screen_cost +
  Sim_Mid_battery * Mid_battery_cost +
  Sim_Mid_other * Mid_other_cost


Pre_Premium_Earned <- Pre_Base * Premium_Sales$group_scaler +
  Sim_Pre_replace * Pre_Addtional_replace +
  Sim_Pre_screen * Pre_Addtional_screen +
  Sim_Pre_battery * Pre_Addtional_battery +
```

```
  Sim_Pre_other * Pre_Addtional_other

Pre_total_cost <- Sim_Pre_replace * Pre_replace_cost +
  Sim_Pre_screen * Pre_screen_cost +
  Sim_Pre_battery * Pre_battery_cost +
  Sim_Pre_other * Pre_other_cost

refund_factor <- 0.5

Mid_Premium_Earned_refunded <- NULL
Pre_Premium_Earned_refunded <- NULL

for (i in 1:5248){
  ifelse(Mid_total_cost[i] == 0,
         Mid_Premium_Earned_refunded[i] <- Mid_Premium_Earned[i]*(1 - refund_factor),
         Mid_Premium_Earned_refunded[i] <- Mid_Premium_Earned[i])
}

for (i in 1:6795){
  ifelse(Pre_total_cost[i] == 0,
         Pre_Premium_Earned_refunded[i] <- Pre_Premium_Earned[i]*(1 - refund_factor),
         Pre_Premium_Earned_refunded[i] <- Pre_Premium_Earned[i])
}

loss_ratio_refund <- (sum(Mid_total_cost) + sum(Pre_total_cost))/(sum(Mid_Premium_Earned_refunded) + sum
loss_ratio_refund
```

```
## [1] 0.9235912
```

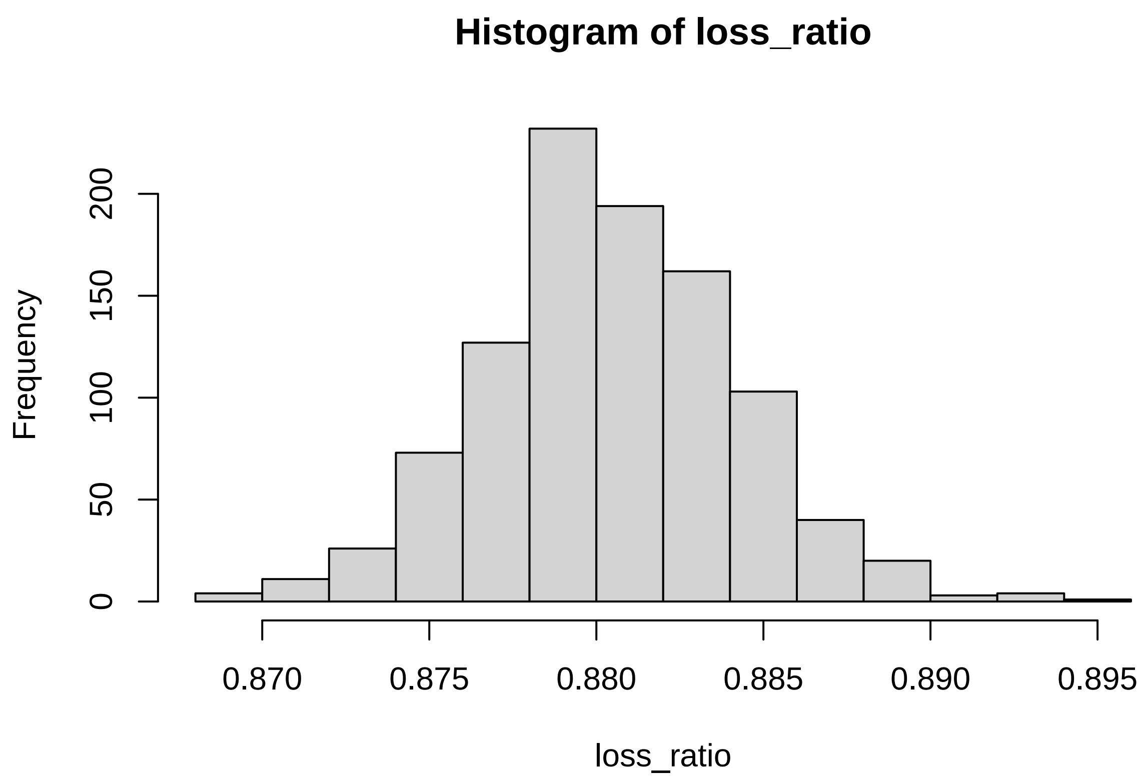
All three method in the same model

```
#Here we put all three methods together

#===Base Premium and fees Set up===
base_fatcor <- 0.7
claim_factor <- 0.5

Mid_Base <- 200 * base_fatcor
Pre_Base <- 300 * base_fatcor



Pre_replace_cost <- 650
Pre_screen_cost <- 200
Pre_battery_cost <- 150
Pre_other_cost <- 100

Pre_Addtional_replace <- claim_factor * Pre_replace_cost
Pre_Addtional_screen <- claim_factor * Pre_screen_cost
Pre_Addtional_battery <- claim_factor * Pre_battery_cost
Pre_Addtional_other <- claim_factor * Pre_other_cost
```

```r
Mid_replace_cost <- 450
Mid_screen_cost <- 150
Mid_battery_cost <- 125
Mid_other_cost <- 100


Mid_Addtional_replace <- claim_factor * Mid_replace_cost
Mid_Addtional_screen <- claim_factor * Mid_screen_cost
Mid_Addtional_battery <- claim_factor * Mid_battery_cost
Mid_Addtional_other <- claim_factor * Mid_other_cost

#===Frequency Limit and refund factor Setup===
limit <- 2
refund_factor <- 0.5

#===Simulation for the 12223 cases===
loss_ratio <- NULL

for (x in 1:1000){
  Sim_Mid_replace <- rpois(5428, mean(MidTier$full_replacement))
  Sim_Mid_screen <- rpois(5428, mean(MidTier$screen_repair))
  Sim_Mid_battery <- rpois(5428, mean(MidTier$battery_repair))
  Sim_Mid_other <- rpois(5428, mean(MidTier$other_repair))

  Sim_Pre_replace <- rpois(6795, mean(Premium$full_replacement))
  Sim_Pre_screen <- rpois(6795, mean(Premium$screen_repair))
  Sim_Pre_battery <- rpois(6795, mean(Premium$battery_repair))
  Sim_Pre_other <- rpois(6795, mean(Premium$other_repair))

  Mid_Premium_Earned <- Mid_Base * MidTier_Sales$group_scaler +
    Sim_Mid_replace * Mid_Addtional_replace +
    Sim_Mid_screen * Mid_Addtional_screen +
    Sim_Mid_battery * Mid_Addtional_battery +
    Sim_Mid_other * Mid_Addtional_other

  Mid_total_cost <- Sim_Mid_replace * Mid_replace_cost +
    Sim_Mid_screen * Mid_screen_cost +
    Sim_Mid_battery * Mid_battery_cost +
    Sim_Mid_other * Mid_other_cost


  Pre_Premium_Earned <- Pre_Base * Premium_Sales$group_scaler +
    Sim_Pre_replace * Pre_Addtional_replace +
    Sim_Pre_screen * Pre_Addtional_screen +
    Sim_Pre_battery * Pre_Addtional_battery +
    Sim_Pre_other * Pre_Addtional_other

  Pre_total_cost <- Sim_Pre_replace * Pre_replace_cost +
    Sim_Pre_screen * Pre_screen_cost +
    Sim_Pre_battery * Pre_battery_cost +
    Sim_Pre_other * Pre_other_cost

  Sim_Mid_replace_limited <- NULL
```

```r
for(i in 1:5248){ifelse(Sim_Mid_replace[i]>limit,
                        Sim_Mid_replace_limited[i] <- limit,
                        Sim_Mid_replace_limited[i] <- Sim_Mid_replace[i])}

Sim_Mid_screen_limited <- NULL
for(i in 1:5248){ifelse(Sim_Mid_screen[i]>limit,
                        Sim_Mid_screen_limited[i] <- limit,
                        Sim_Mid_screen_limited[i] <- Sim_Mid_screen[i])}

Sim_Mid_battery_limited <- NULL
for(i in 1:5248){ifelse(Sim_Mid_battery[i]>limit,
                        Sim_Mid_battery_limited[i] <- limit,
                        Sim_Mid_battery_limited[i] <- Sim_Mid_battery[i])}

Sim_Mid_other_limited <- NULL
for(i in 1:5248){ifelse(Sim_Mid_other[i]>limit,
                        Sim_Mid_other_limited[i] <- limit,
                        Sim_Mid_other_limited[i] <- Sim_Mid_other[i])}


Mid_total_cost_limited <- Sim_Mid_replace_limited * Mid_replace_cost +
  Sim_Mid_screen_limited * Mid_screen_cost +
  Sim_Mid_battery_limited * Mid_battery_cost +
  Sim_Mid_other_limited * Mid_other_cost


Sim_Pre_replace_limited <- NULL
for(i in 1:6795){ifelse(Sim_Pre_replace[i]>limit,
                        Sim_Pre_replace_limited[i] <- limit,
                        Sim_Pre_replace_limited[i] <- Sim_Pre_replace[i])}

Sim_Pre_screen_limited <- NULL
for(i in 1:6795){ifelse(Sim_Pre_screen[i]>limit,
                        Sim_Pre_screen_limited[i] <- limit,
                        Sim_Pre_screen_limited[i] <- Sim_Pre_screen[i])}

Sim_Pre_battery_limited <- NULL
for(i in 1:6795){ifelse(Sim_Pre_battery[i]>limit,
                        Sim_Pre_battery_limited[i] <- limit,
                        Sim_Pre_battery_limited[i] <- Sim_Pre_battery[i])}

Sim_Pre_other_limited <- NULL
for(i in 1:6795){ifelse(Sim_Pre_other[i]>limit,
                        Sim_Pre_other_limited[i] <- limit,
                        Sim_Pre_other_limited[i] <- Sim_Pre_other[i])}

Pre_total_cost_limited <- Sim_Pre_replace_limited * Pre_replace_cost +
  Sim_Pre_screen_limited * Pre_screen_cost +
  Sim_Pre_battery_limited * Pre_battery_cost +
  Sim_Pre_other_limited * Pre_other_cost

for (i in 1:5248){
  ifelse(Mid_total_cost[i] == 0,
```

```
          Mid_Premium_Earned_refunded[i] <- Mid_Premium_Earned[i]*(1 - refund_factor),
          Mid_Premium_Earned_refunded[i] <- Mid_Premium_Earned[i])
  }

  for (i in 1:6795){
    ifelse(Pre_total_cost[i] == 0,
          Pre_Premium_Earned_refunded[i] <- Pre_Premium_Earned[i]*(1 - refund_factor),
          Pre_Premium_Earned_refunded[i] <- Pre_Premium_Earned[i])
  }


  loss_ratio[x] <- (sum(Mid_total_cost_limited) + sum(Pre_total_cost_limited))/(sum(Mid_Premium_Earned_r
                                                                                  sum(Pre_Premium_Earned
}

hist(loss_ratio)
```




```
quantile(loss_ratio, probs=c(0.95))
```

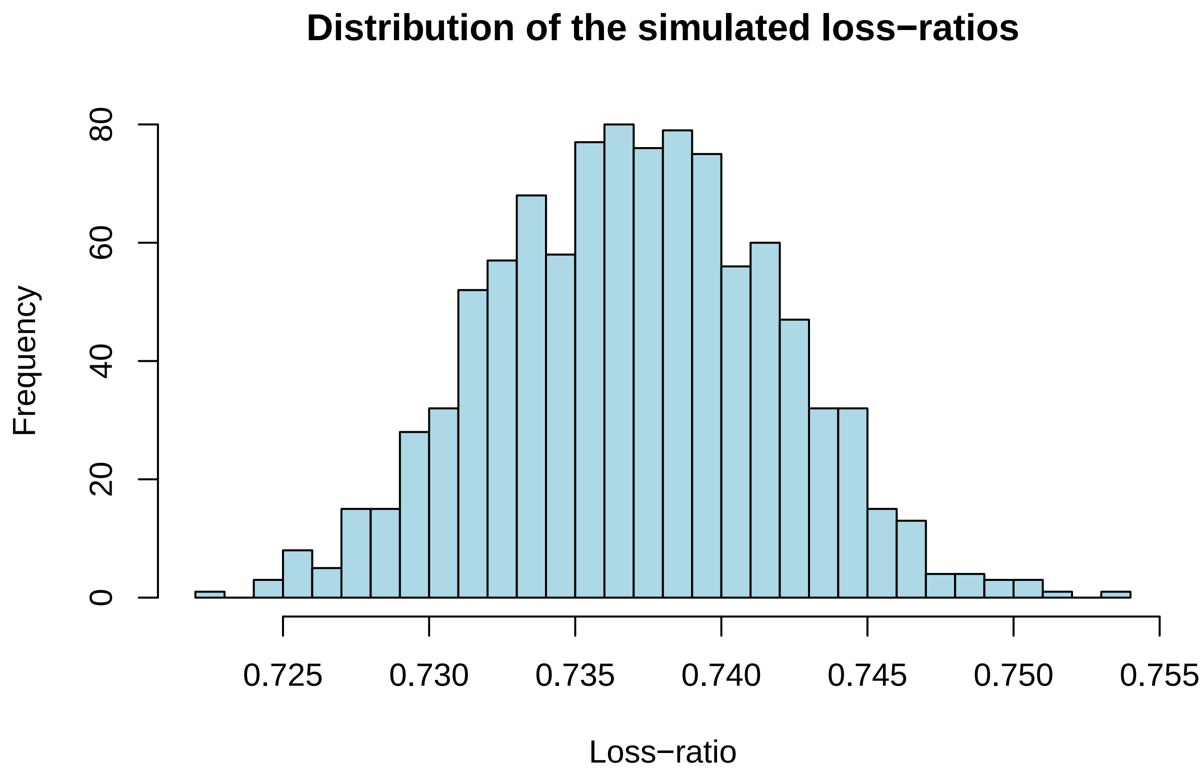
```
##       95%
## 0.8866222
```

Final Model

```r
#Here we put all three methods together

#===Base Premium and fees Set up===
base_fatcor <- 1.225
claim_factor <- 0.5

Mid_Base <- 209.66   * base_fatcor
Pre_Base <- 385.02   * base_fatcor



Pre_replace_cost <- 670
Pre_screen_cost <- 220
Pre_battery_cost <- 170
Pre_other_cost <- 120

Pre_Addtional_replace <- claim_factor * Pre_replace_cost
Pre_Addtional_screen <- claim_factor * Pre_screen_cost
Pre_Addtional_battery <- claim_factor * Pre_battery_cost
Pre_Addtional_other <- claim_factor * Pre_other_cost


Mid_replace_cost <- 470
Mid_screen_cost <- 170
Mid_battery_cost <- 145
Mid_other_cost <- 120


Mid_Addtional_replace <- claim_factor * Mid_replace_cost
Mid_Addtional_screen <- claim_factor * Mid_screen_cost
Mid_Addtional_battery <- claim_factor * Mid_battery_cost
Mid_Addtional_other <- claim_factor * Mid_other_cost

#===Frequency Limit and refund factor Setup===
limit <- 2
refund_factor <- 0.3

#===Simulation for the 12223 cases===
loss_ratio <- NULL

for (x in 1:1000){
  Sim_Mid_replace <- rpois(5428, mean(MidTier$full_replacement))
  Sim_Mid_screen <- rpois(5428, mean(MidTier$screen_repair))
  Sim_Mid_battery <- rpois(5428, mean(MidTier$battery_repair))
  Sim_Mid_other <- rpois(5428, mean(MidTier$other_repair))

  Sim_Pre_replace <- rpois(6795, mean(Premium$full_replacement))
  Sim_Pre_screen <- rpois(6795, mean(Premium$screen_repair))
  Sim_Pre_battery <- rpois(6795, mean(Premium$battery_repair))
  Sim_Pre_other <- rpois(6795, mean(Premium$other_repair))

  Mid_Premium_Earned <- Mid_Base * (MidTier_Sales$group_scaler + 1)/2 +
    Sim_Mid_replace * Mid_Addtional_replace
```

```r
Mid_total_cost <- Sim_Mid_replace * Mid_replace_cost +
  Sim_Mid_screen * Mid_screen_cost +
  Sim_Mid_battery * Mid_battery_cost +
  Sim_Mid_other * Mid_other_cost


Pre_Premium_Earned <- Pre_Base * (Premium_Sales$group_scaler + 1)/2 +
  Sim_Pre_replace * Pre_Addtional_replace


Pre_total_cost <- Sim_Pre_replace * Pre_replace_cost +
  Sim_Pre_screen * Pre_screen_cost +
  Sim_Pre_battery * Pre_battery_cost +
  Sim_Pre_other * Pre_other_cost

Sim_Mid_replace_limited <- NULL
for(i in 1:5248){ifelse(Sim_Mid_replace[i]>limit,
                        Sim_Mid_replace_limited[i] <- limit,
                        Sim_Mid_replace_limited[i] <- Sim_Mid_replace[i])}

Sim_Mid_screen_limited <- NULL
for(i in 1:5248){ifelse(Sim_Mid_screen[i]>limit,
                        Sim_Mid_screen_limited[i] <- limit,
                        Sim_Mid_screen_limited[i] <- Sim_Mid_screen[i])}

Sim_Mid_battery_limited <- NULL
for(i in 1:5248){ifelse(Sim_Mid_battery[i]>limit,
                        Sim_Mid_battery_limited[i] <- limit,
                        Sim_Mid_battery_limited[i] <- Sim_Mid_battery[i])}

Sim_Mid_other_limited <- NULL
for(i in 1:5248){ifelse(Sim_Mid_other[i]>limit,
                        Sim_Mid_other_limited[i] <- limit,
                        Sim_Mid_other_limited[i] <- Sim_Mid_other[i])}


Mid_total_cost_limited <- Sim_Mid_replace_limited * Mid_replace_cost +
  Sim_Mid_screen_limited * Mid_screen_cost +
  Sim_Mid_battery_limited * Mid_battery_cost +
  Sim_Mid_other_limited * Mid_other_cost


Sim_Pre_replace_limited <- NULL
for(i in 1:6795){ifelse(Sim_Pre_replace[i]>limit,
                        Sim_Pre_replace_limited[i] <- limit,
                        Sim_Pre_replace_limited[i] <- Sim_Pre_replace[i])}

Sim_Pre_screen_limited <- NULL
for(i in 1:6795){ifelse(Sim_Pre_screen[i]>limit,
                        Sim_Pre_screen_limited[i] <- limit,
                        Sim_Pre_screen_limited[i] <- Sim_Pre_screen[i])}
```

```
  Sim_Pre_battery_limited <- NULL
  for(i in 1:6795){ifelse(Sim_Pre_battery[i]>limit,
                         Sim_Pre_battery_limited[i] <- limit,
                         Sim_Pre_battery_limited[i] <- Sim_Pre_battery[i])}

  Sim_Pre_other_limited <- NULL
  for(i in 1:6795){ifelse(Sim_Pre_other[i]>limit,
                         Sim_Pre_other_limited[i] <- limit,
                         Sim_Pre_other_limited[i] <- Sim_Pre_other[i])}

  Pre_total_cost_limited <- Sim_Pre_replace_limited * Pre_replace_cost +
    Sim_Pre_screen_limited * Pre_screen_cost +
    Sim_Pre_battery_limited * Pre_battery_cost +
    Sim_Pre_other_limited * Pre_other_cost

  for (i in 1:5248){
    ifelse(Mid_total_cost[i] == 0,
           Mid_Premium_Earned_refunded[i] <- Mid_Premium_Earned[i]*(1 - refund_factor),
           Mid_Premium_Earned_refunded[i] <- Mid_Premium_Earned[i])
  }

  for (i in 1:6795){
    ifelse(Pre_total_cost[i] == 0,
           Pre_Premium_Earned_refunded[i] <- Pre_Premium_Earned[i]*(1 - refund_factor),
           Pre_Premium_Earned_refunded[i] <- Pre_Premium_Earned[i])
  }


  loss_ratio[x] <- (sum(Mid_total_cost_limited) + sum(Pre_total_cost_limited))/(sum(Mid_Premium_Earned_r
                                                                                  sum(Pre_Premium_Earned
}

hist(loss_ratio, breaks = 30, main = "Distribution of the simulated loss-ratios", xlab = "Loss-ratio", c
```




```
quantile(loss_ratio, probs = c(0.995))
```

```
##     99.5%
## 0.7499955
```