# Milliman Case Study

*Warren Hendricks*

*January 18, 2019*

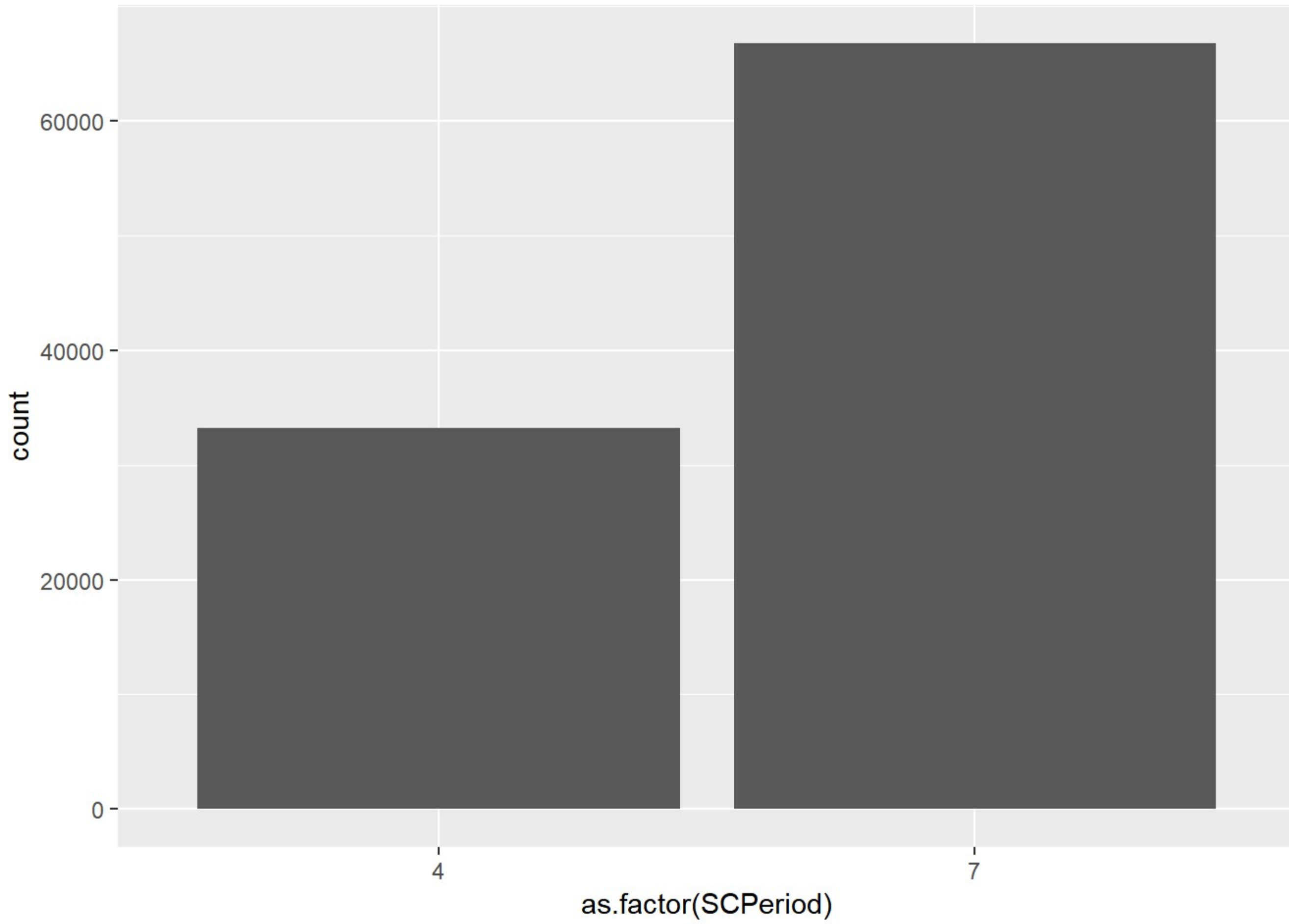
```
##      PolNum          SCPeriod           AV                   BB         
##  Min.   :100000   Min.   :4.000   Min.   :    1058   Min.   :   1375  
##  1st Qu.:125000   1st Qu.:4.000   1st Qu.:   51282   1st Qu.:  54715  
##  Median :150000   Median :7.000   Median :  100758   Median : 107756  
##  Mean   :150000   Mean   :6.002   Mean   :  166994   Mean   : 180710  
##  3rd Qu.:174999   3rd Qu.:7.000   3rd Qu.:  197696   3rd Qu.: 213114  
##  Max.   :199999   Max.   :7.000   Max.   :40673199   Max.   :8316085  
##                                                      NA's   :2        
##  RiderCode      Age               q               Surr        
##  A:25223   Min.   :-10.0   Min.   : 1.00   Min.   :0.00000  
##  B:12593   1st Qu.: 58.0   1st Qu.: 6.00   1st Qu.:0.00000  
##  C:62183   Median : 65.0   Median :14.00   Median :0.00000  
##  D:    1   Mean   : 64.5   Mean   :16.53   Mean   :0.05281  
##            3rd Qu.: 71.0   3rd Qu.:25.00   3rd Qu.:0.00000  
##            Max.   :110.0   Max.   :45.00   Max.   :1.00000  
## 
```

5.281% of these annuities were surrendered, so surrender is relatively rare.

The variables AV and BB each have at least one extreme outlier. This could possibly create high-leverage points.

Over 62% of these annuities have Rider C. About 1/4 have Rider A, and about 1/8 have Rider B

I also see three small concerns in this summary: BB has 2 NA values, one of the RiderCode observations is “D”, and the minimum age is -10. Since there is only one observation with Rider Code D, I will remove it from the dataset unless Rider Code does not end up being one of my predictors.




It appears about 2/3 of these annuities have a surrender period of 7 years, and 1/3 have a surrender period of 4 years.

I will look at the lowest values of Age, and see if there are anymore values that do not make sense.

```
## [1] -10  22  26  29  29  29
```

Luckily, -10 appears to be the only erroneous point in the Age column.

I would rather not arbitrarily delete columns, so I will see if I can use a basic regression model to impute the missing age.

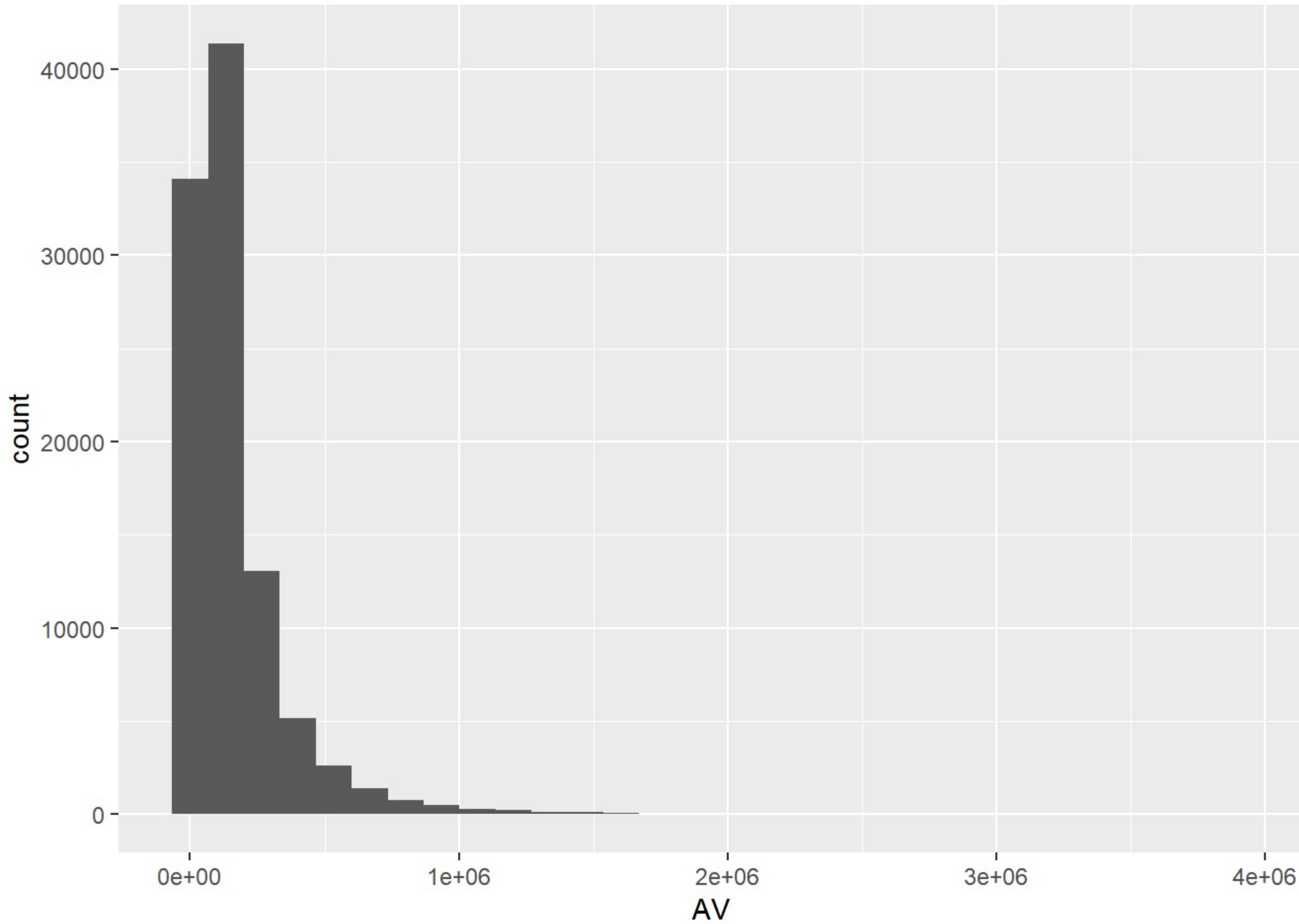
```
##
## Call:
## lm(formula = Age ~ . - PolNum, data = data, subset = Age > 0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -42.640  -6.413   0.360   6.384  45.519
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.462e+01  1.409e-01 458.676  < 2e-16 ***
## SCPeriod    -3.100e-02  2.017e-02  -1.537   0.1243
## AV           1.728e-07  2.018e-07   0.856   0.3920
## BB          -1.676e-07  2.141e-07  -0.783   0.4336
## RiderCodeB   6.300e-02  9.831e-02   0.641   0.5216
## RiderCodeC   7.963e-02  6.726e-02   1.184   0.2364
## RiderCodeD   1.947e+01  9.010e+00   2.161   0.0307 *
## q            2.175e-03  2.281e-03   0.954   0.3403
## Surr        -5.218e-01  1.282e-01  -4.069 4.72e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.009 on 99988 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.0002563,  Adjusted R-squared:  0.0001763
## F-statistic: 3.204 on 8 and 99988 DF,  p-value: 0.001215
```

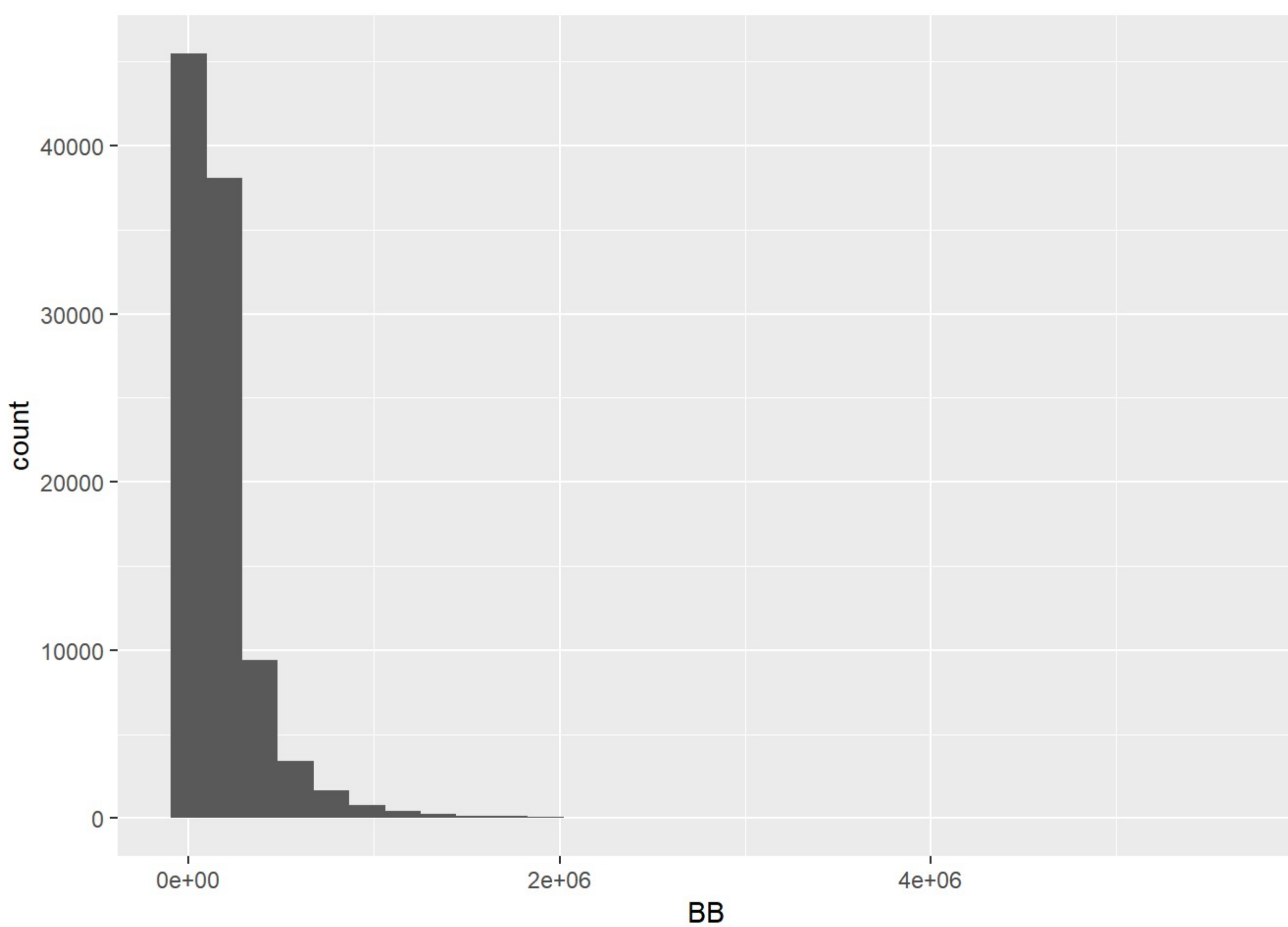
The R^2 is incredibly low here, so it looks like the other variables are not good predictors of age, so instead I will impute the median age to this value.

I will attempt to do the same thing with BB, since they are the only missing values in their respective rows.

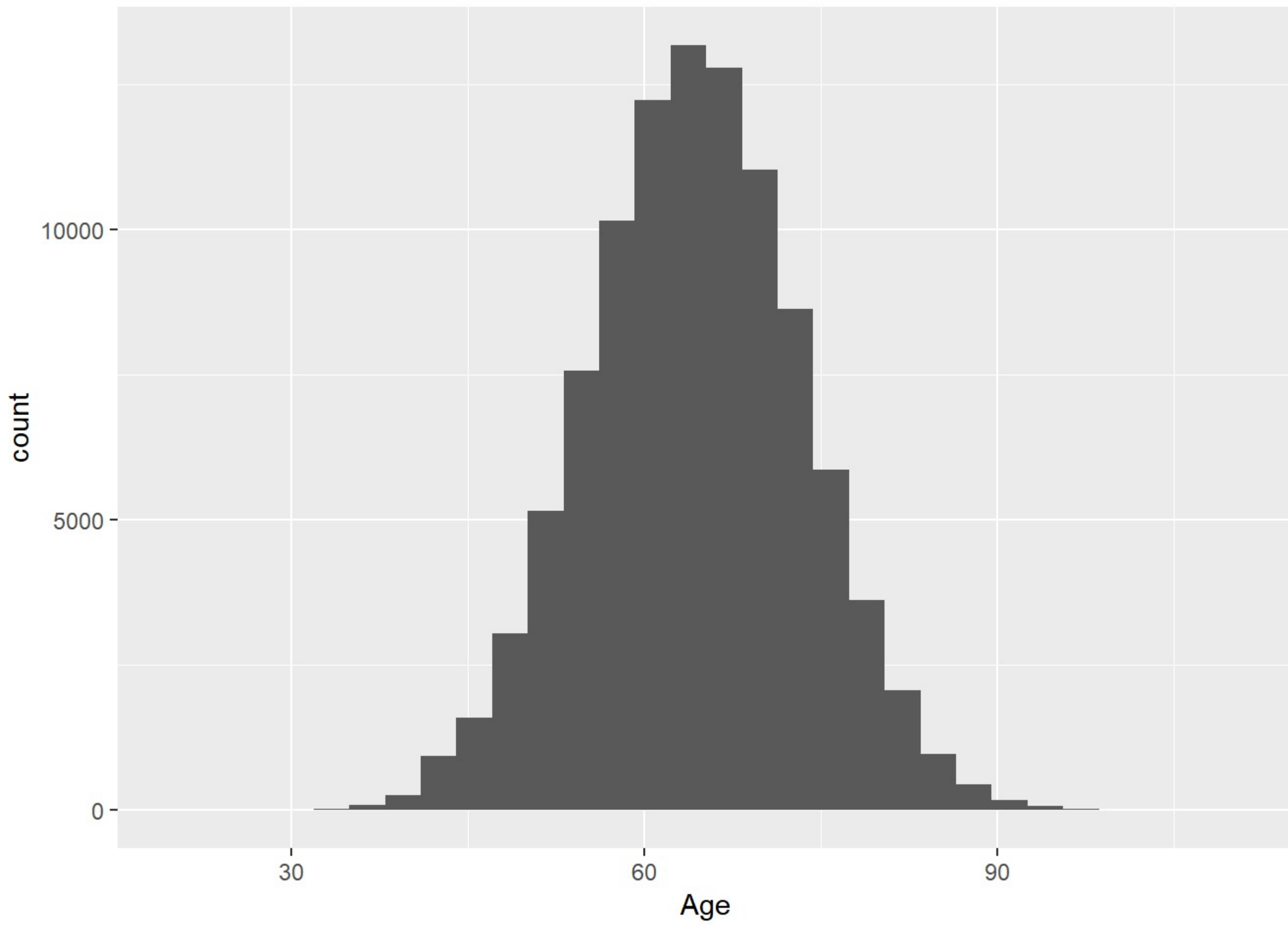
```
##
## Call:
## lm(formula = BB ~ . - PolNum, data = data)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -31973885    -37901    -22522      7978   3552841
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.807e+04  3.664e+03  13.120   <2e-16 ***
## SCPeriod     4.733e+02  2.980e+02   1.588    0.112
## AV           7.881e-01  1.637e-03 481.456   <2e-16 ***
## RiderCodeB   1.962e+02  1.452e+03   0.135    0.893
## RiderCodeC   1.349e+03  9.936e+02   1.357    0.175
## RiderCodeD   6.487e+04  1.331e+05   0.487    0.626
## Age         -3.659e+01  4.672e+01  -0.783    0.434
## q            4.912e+01  3.370e+01   1.458    0.145
## Surr        -2.141e+04  1.893e+03 -11.310   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 133100 on 99989 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.6996, Adjusted R-squared:  0.6996
## F-statistic: 2.911e+04 on 8 and 99989 DF,  p-value: < 2.2e-16
```

This model appears to be a decently good fit for BB, so I will use this to impute the two missing values.

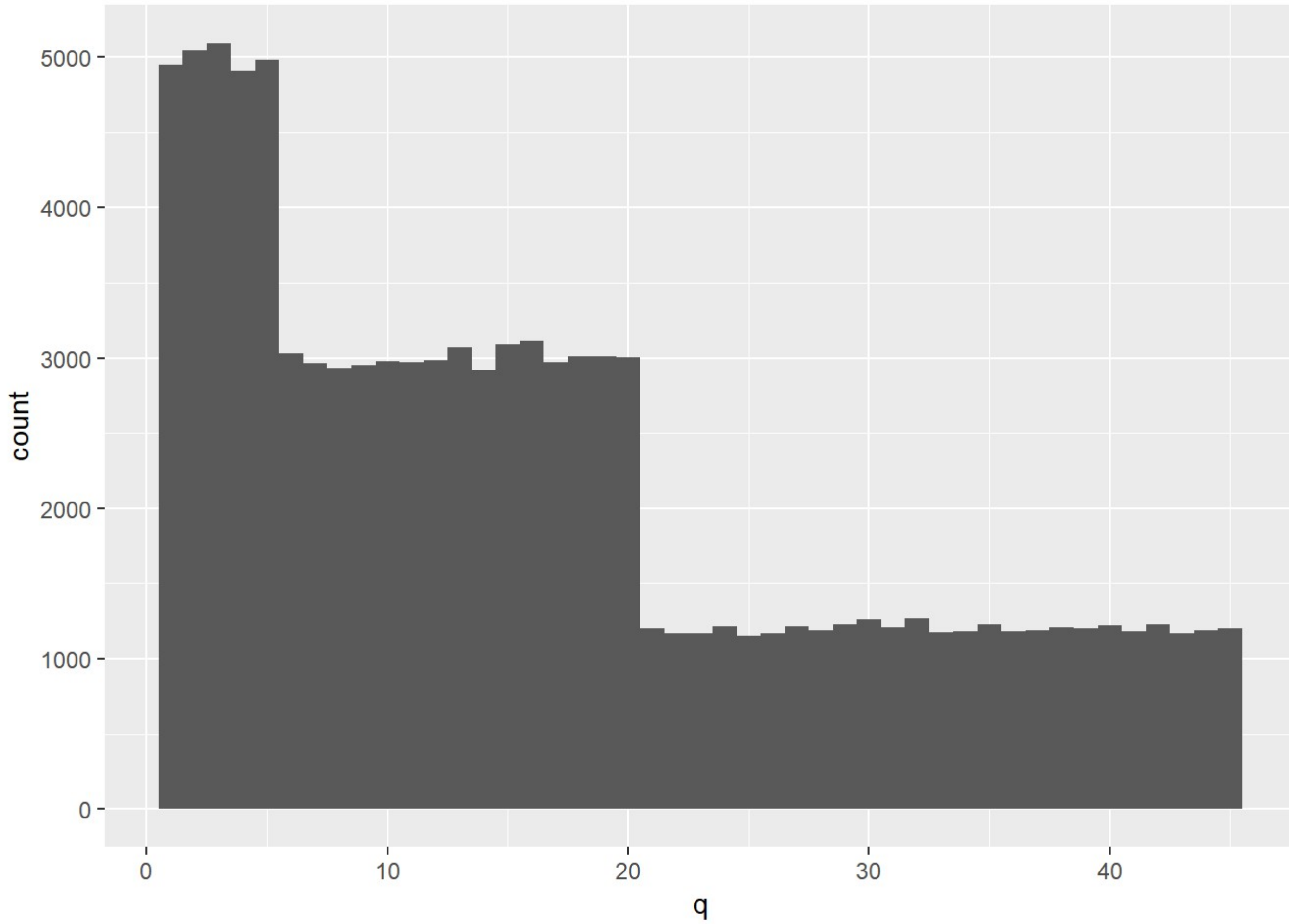

40000
30000
20000
10000
0
count
0e+00
1e+06
2e+06
3e+06
4e+06
AV


40000
30000
20000
10000
0
count
0e+00
2e+06
4e+06
BB

Removing some of the more extreme values, it’s still clear that AV and BB are both heavily right-skewed. Both appear to follow Gamma distributions, but BB looks closer to an exponential distribution.




Age seems to be very close to normally distributed.




This looks like a series of uniform distributions. One going from 1 to 5, then 6 to 20, and 21 to 45. This may imply that many people get 1 year or 5 year annuities.

Now I will add ITM (In-the-moneyness) and SCPhase (Surrender Charge Phase) to my dataset.

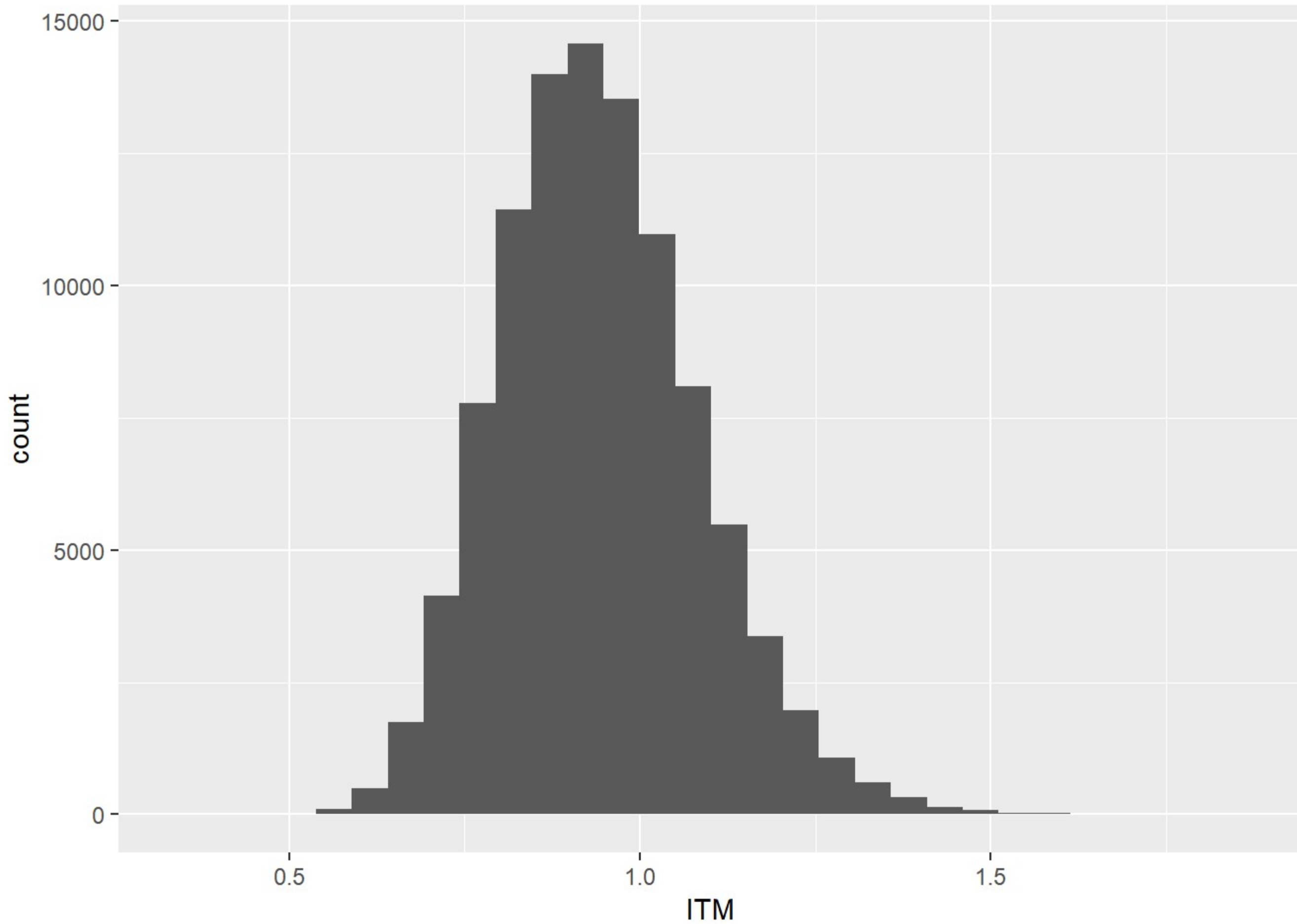
ITM

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
##   0.3371   0.8427   0.9326   0.9479   1.0314 315.4418
```

There is at least one outlier in the ITM column, so I will check the highest values for any others.

```
## [1] -315.441822 -209.772844   -1.823033   -1.765487   -1.760806   -1.745504
```

There appears to be one more major outlier here.



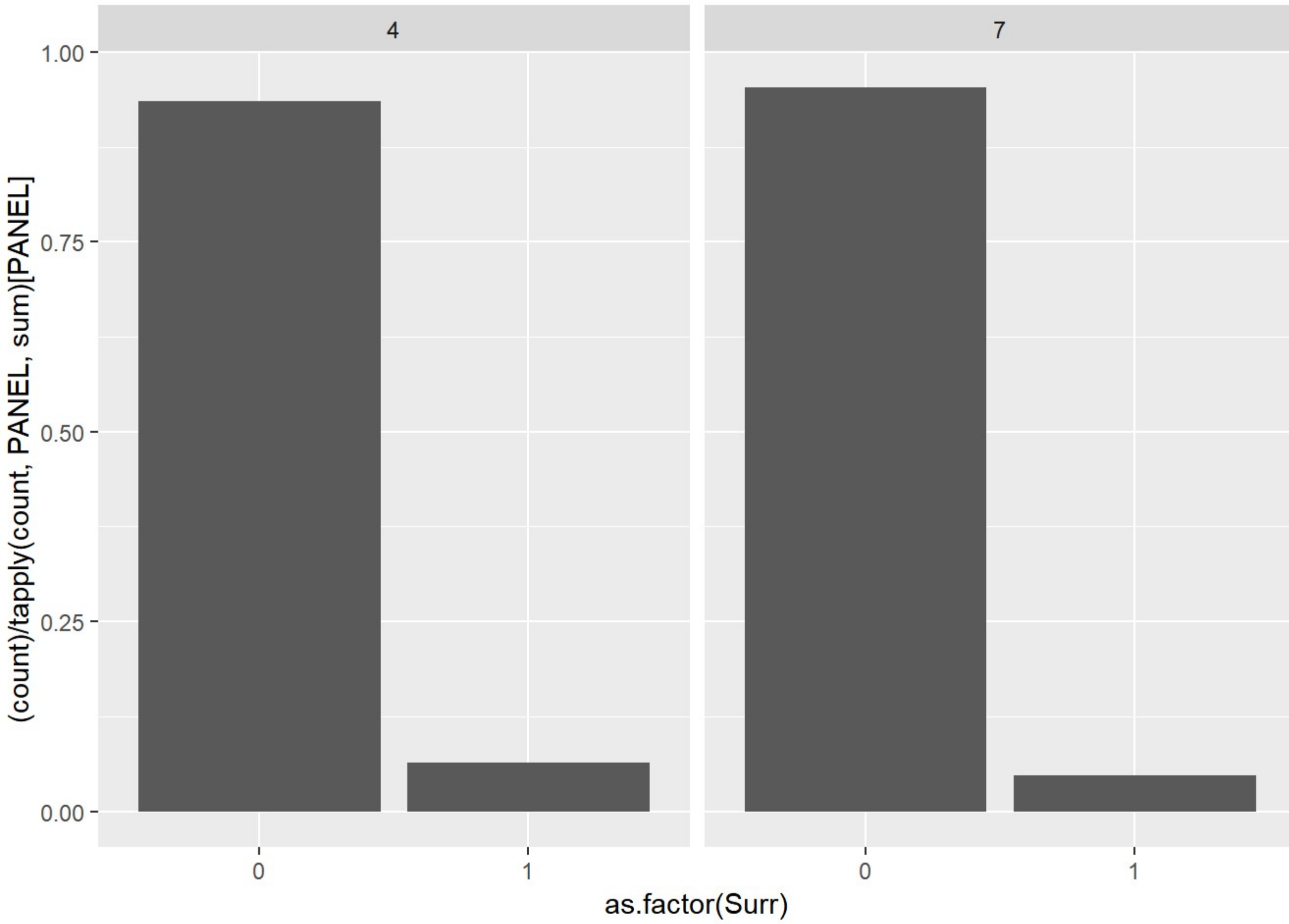

Removing the two major outliers reveals the pattern of ITM is relatively normal, but there is some visible right-skewness.

SCPhase

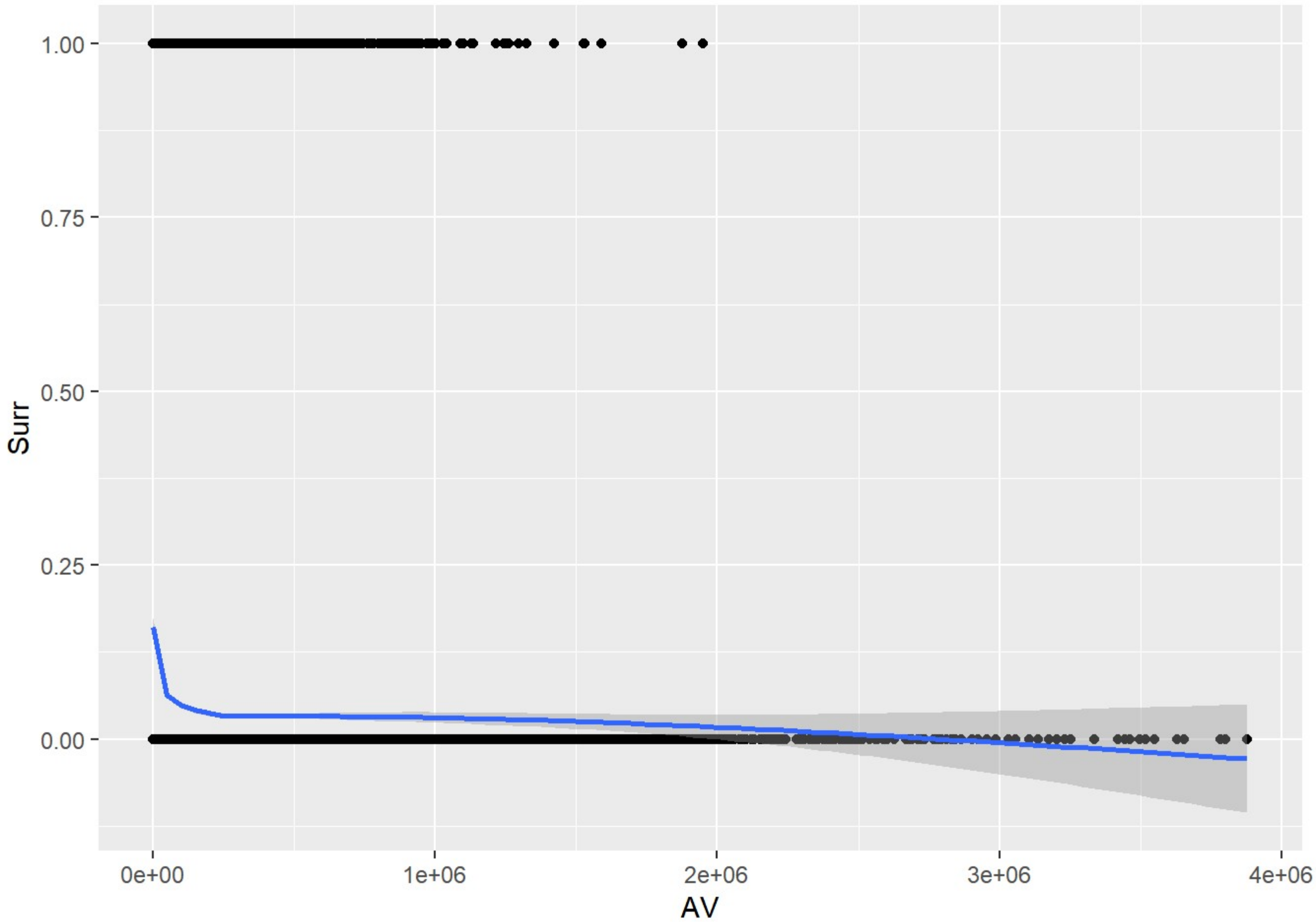
```
##   END    IN   OUT
##  1851 70403 27746
```

Most of the annuities are still in the surrender penalty phase.

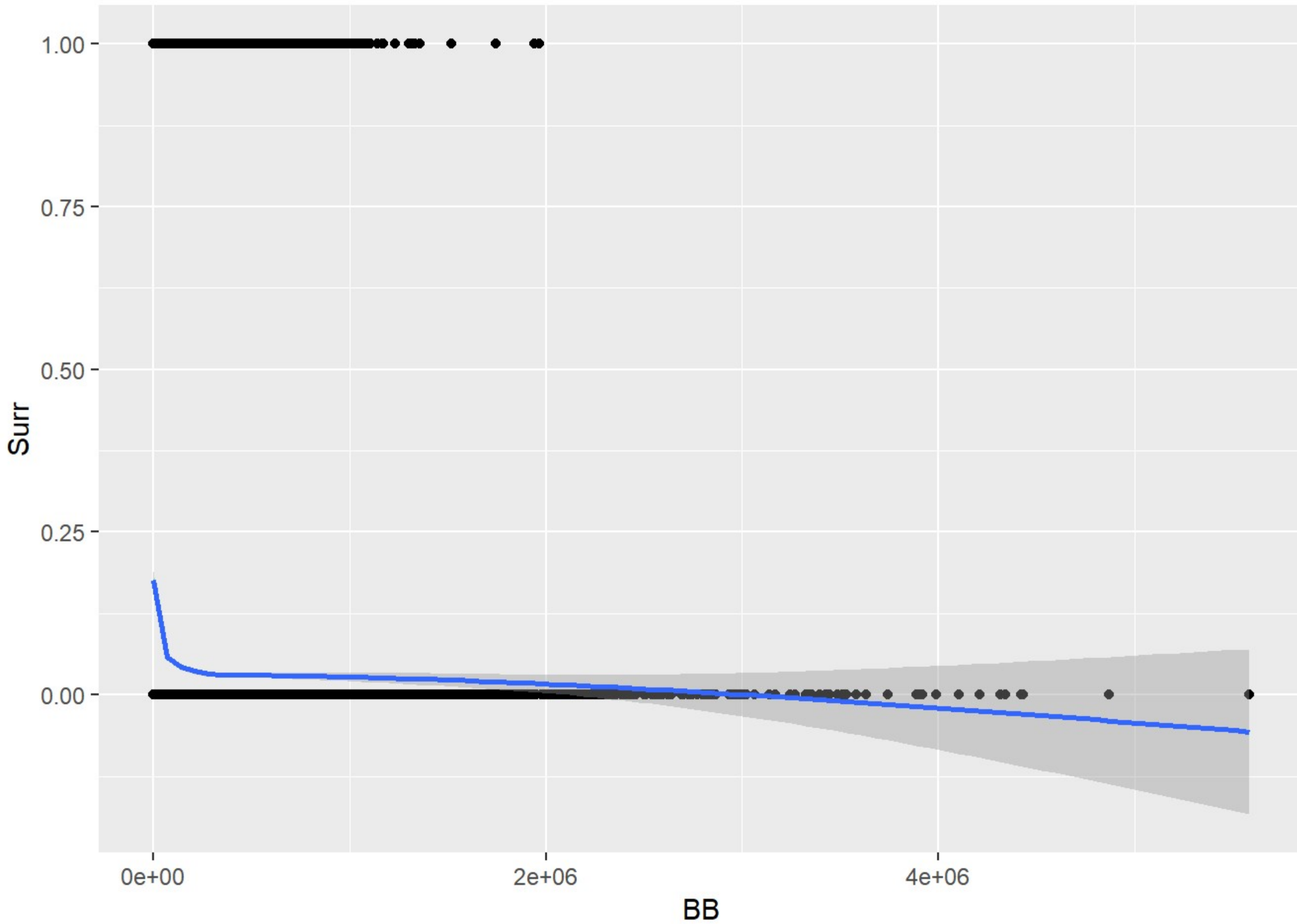
Now I will look at the relationship between the variables and the Surrender rate.



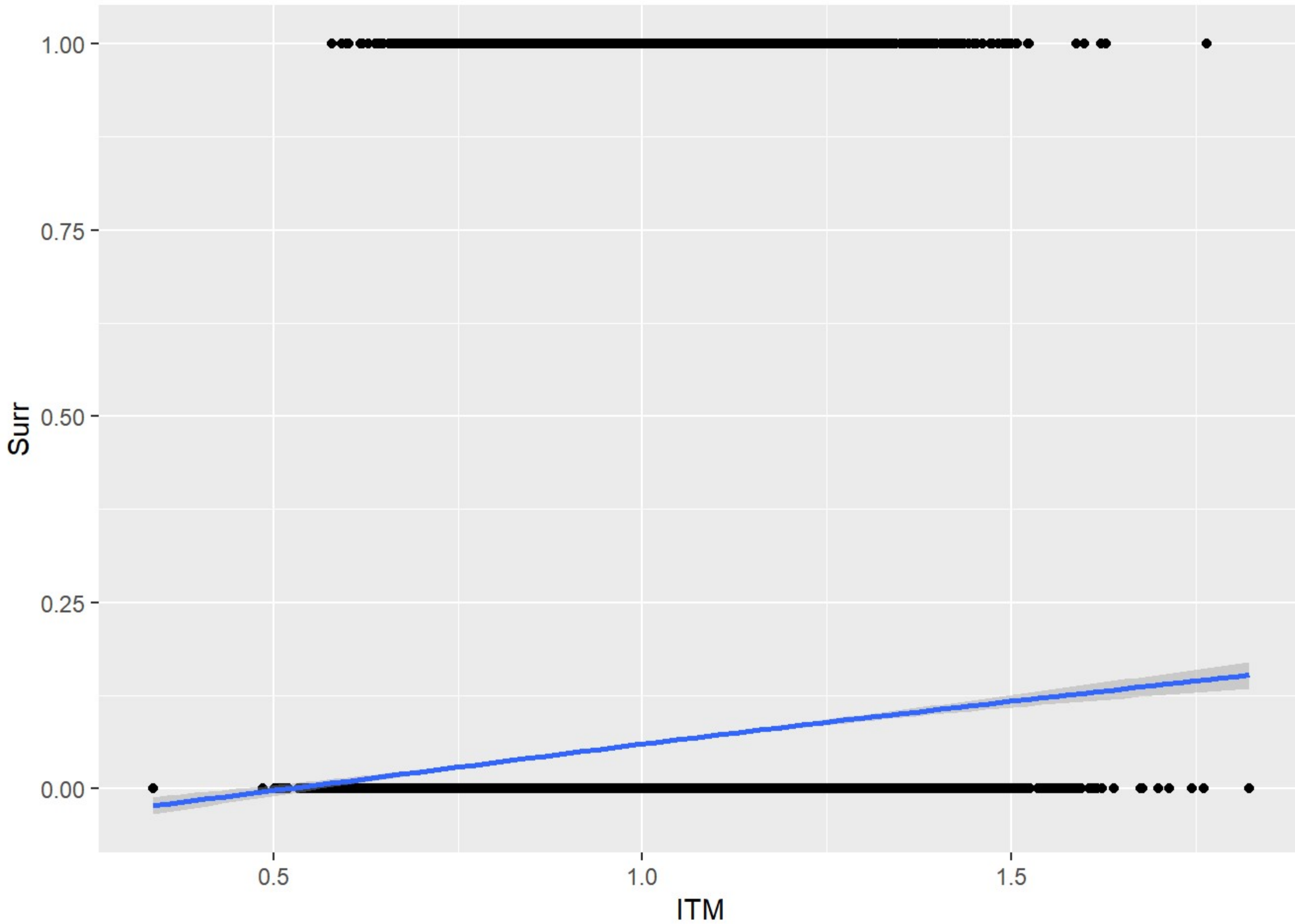

Annuities with a 7-year surrender penalty have a slightly lower surrender rate than annuities with a 4-year surrender penalty.



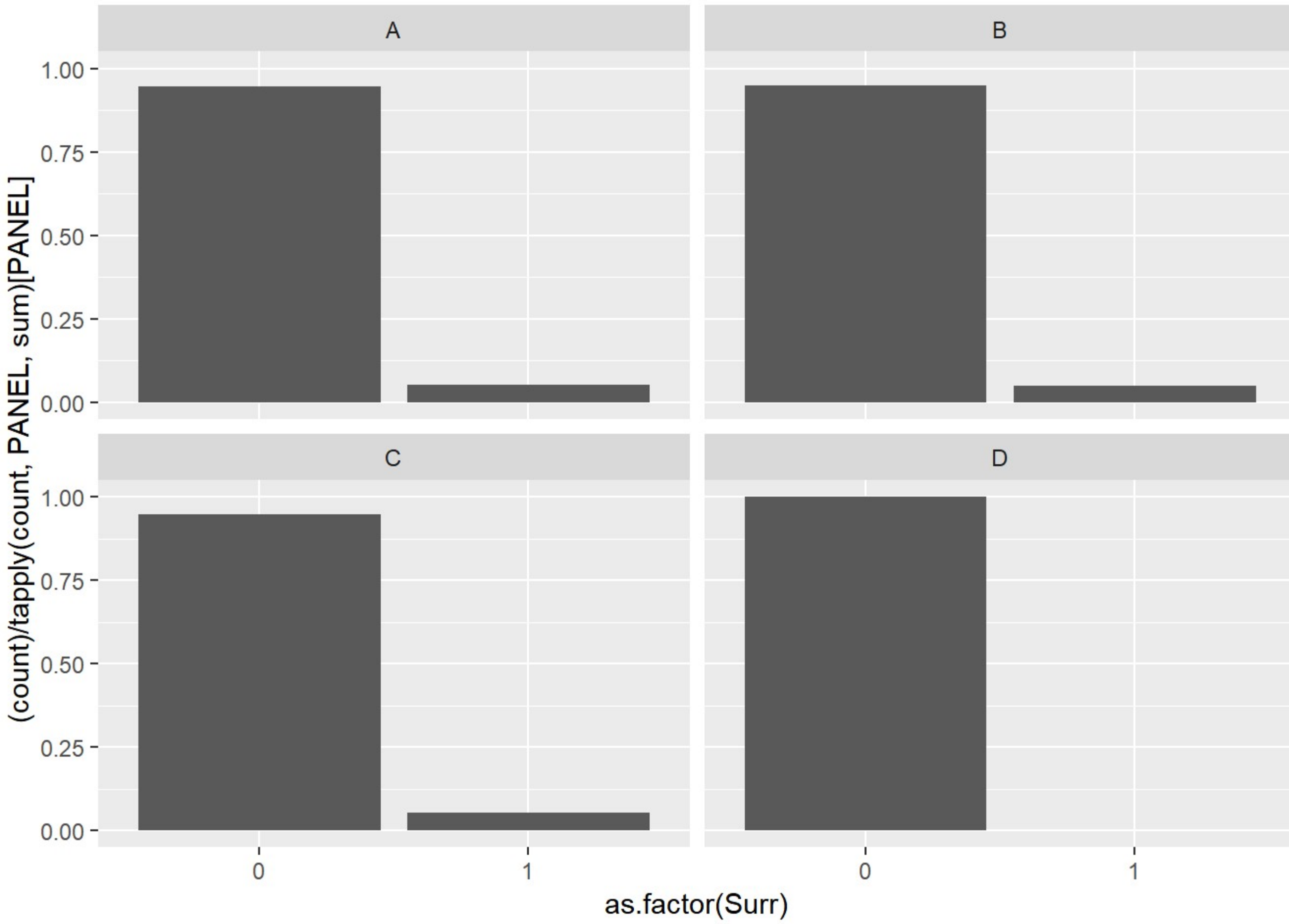

The surrender rate decreases as AV increases.




The surrender rate also decreases as BB increases in a very similar pattern to AV.

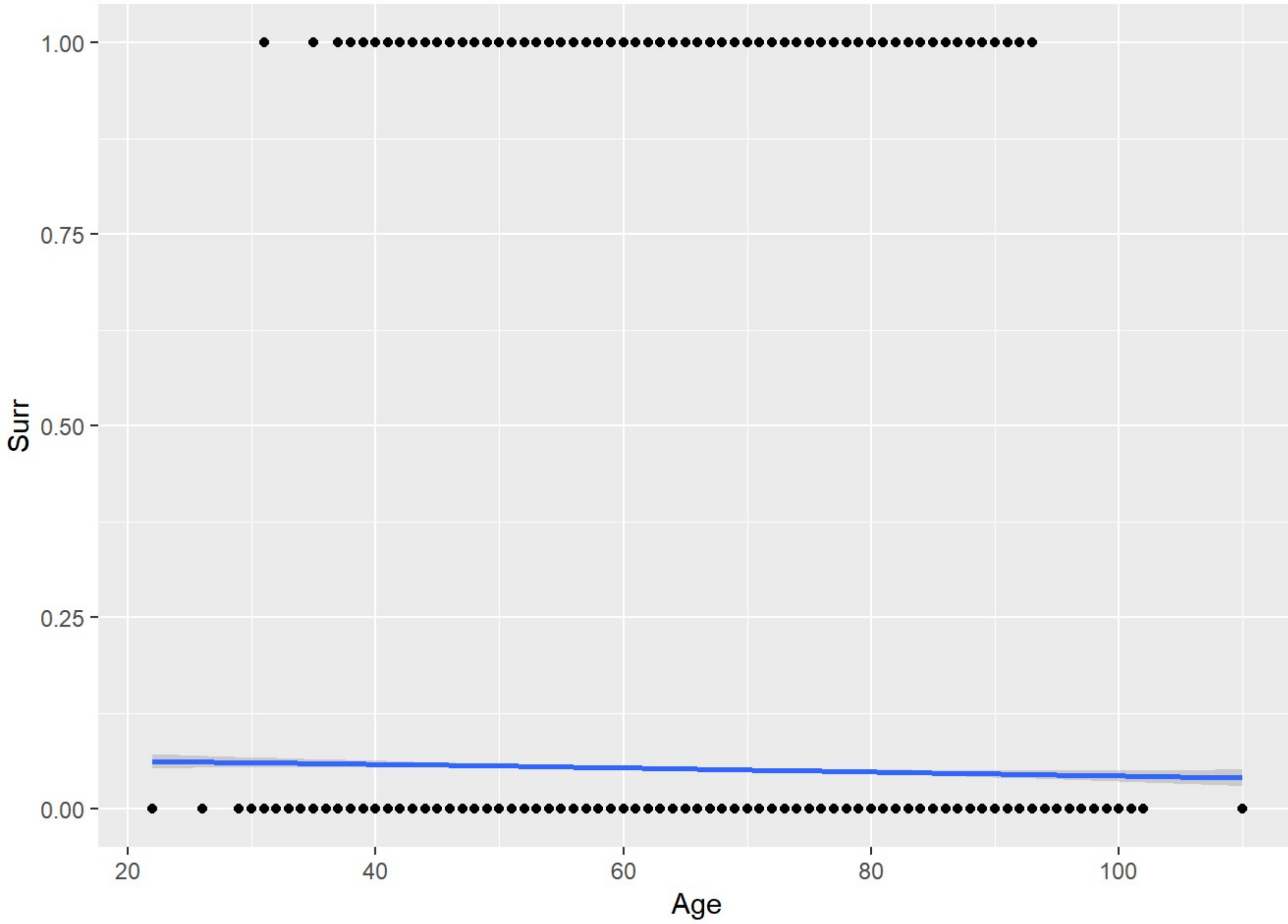



As ITM increases, the probability of surrender seems to increase in a very linear fashion.

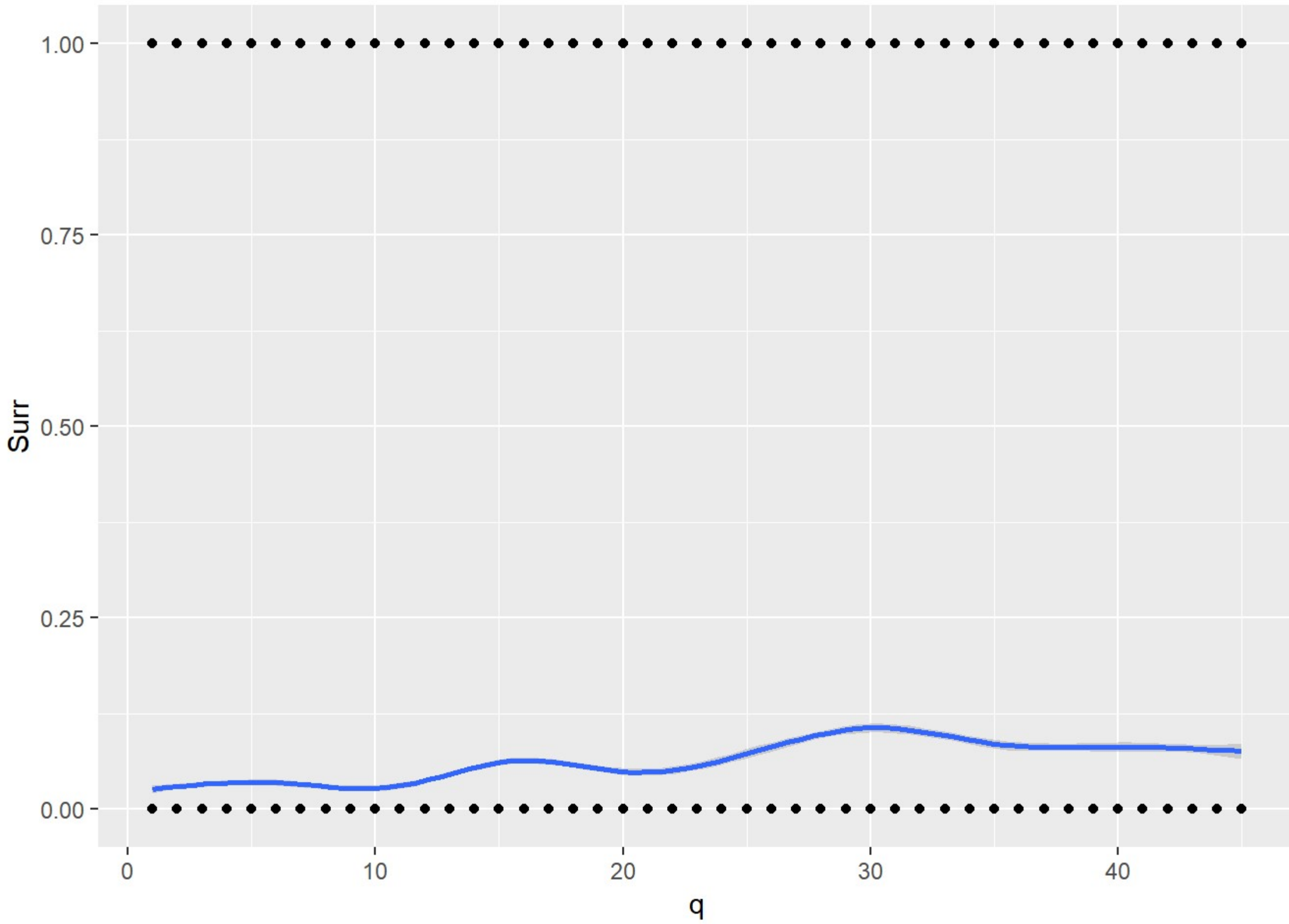



```
##
## Call:
## glm(formula = Surr ~ RiderCode, family = "binomial", data = data2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.3319  -0.3298  -0.3298  -0.3298   2.4424
##
## Coefficients:
##             Estimate Std. Error  z value Pr(>|z|)
## (Intercept) -2.87108    0.02796 -102.703   <2e-16 ***
## RiderCodeB  -0.05949    0.04932   -1.206    0.228
## RiderCodeC  -0.01343    0.03320   -0.405    0.686
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 41341  on 99998  degrees of freedom
## Residual deviance: 41340  on 99996  degrees of freedom
## AIC: 41346
##
## Number of Fisher Scoring iterations: 5
```

Visually, it doesn't look like RiderCode has an effect on surrender rate. A simple logistic regression seems to confirm this.

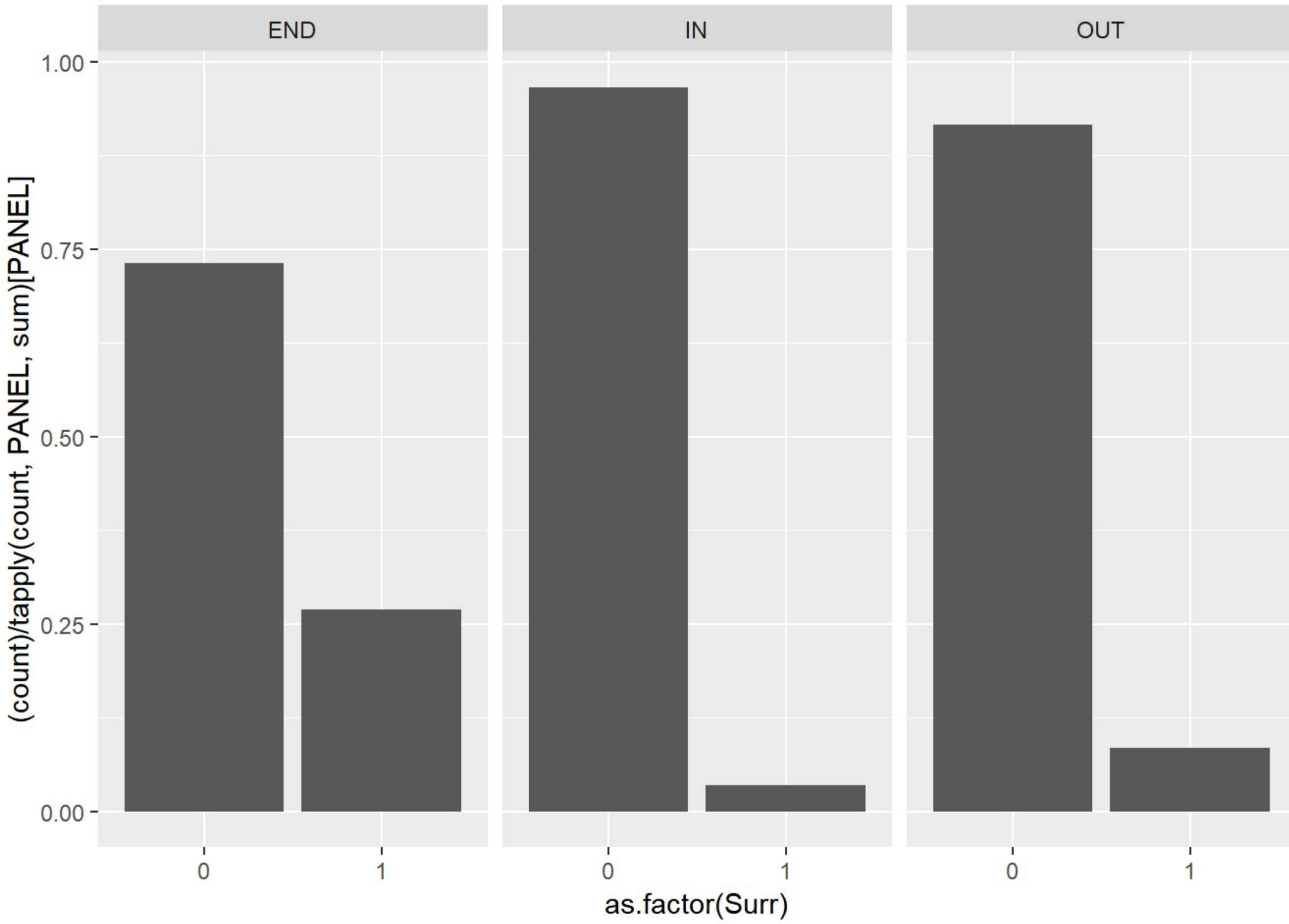



As age increases, the surrender rate seems to decrease in a very linear fashion.


1.00
0.75
0.50
0.25
0.00
Surr
0
10
20
30
40
q

```
##
## Call:
## glm(formula = Surr ~ poly(q, 2) + cos(q), family = "binomial",
##     data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.4616  -0.3747  -0.3098  -0.2508   2.7647
##
## Coefficients:
##              Estimate Std. Error  z value Pr(>|z|)
## (Intercept)  -2.97812    0.01540 -193.348  < 2e-16 ***
## poly(q, 2)1 136.26788    4.59884   29.631  < 2e-16 ***
## poly(q, 2)2 -43.07705    4.48273   -9.610  < 2e-16 ***
## cos(q)       -0.16313    0.02038   -8.003 1.22e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 41341  on 99999  degrees of freedom
## Residual deviance: 40380  on 99996  degrees of freedom
## AIC: 40388
##
## Number of Fisher Scoring iterations: 6
```

Q seems to have a strange relationship to the surrender rate. It looks somewhat cyclical, and there may be a slightly quadratic relationship at play here. A simple logistic regression using q^2 and cos(q) seems to confirm this. I will likely remove cos(q), because it has the highest p-value, and I want to avoid overfitting.




Surrender rates differ significantly by surrender charge phase. End has the highest rate, followed by out.

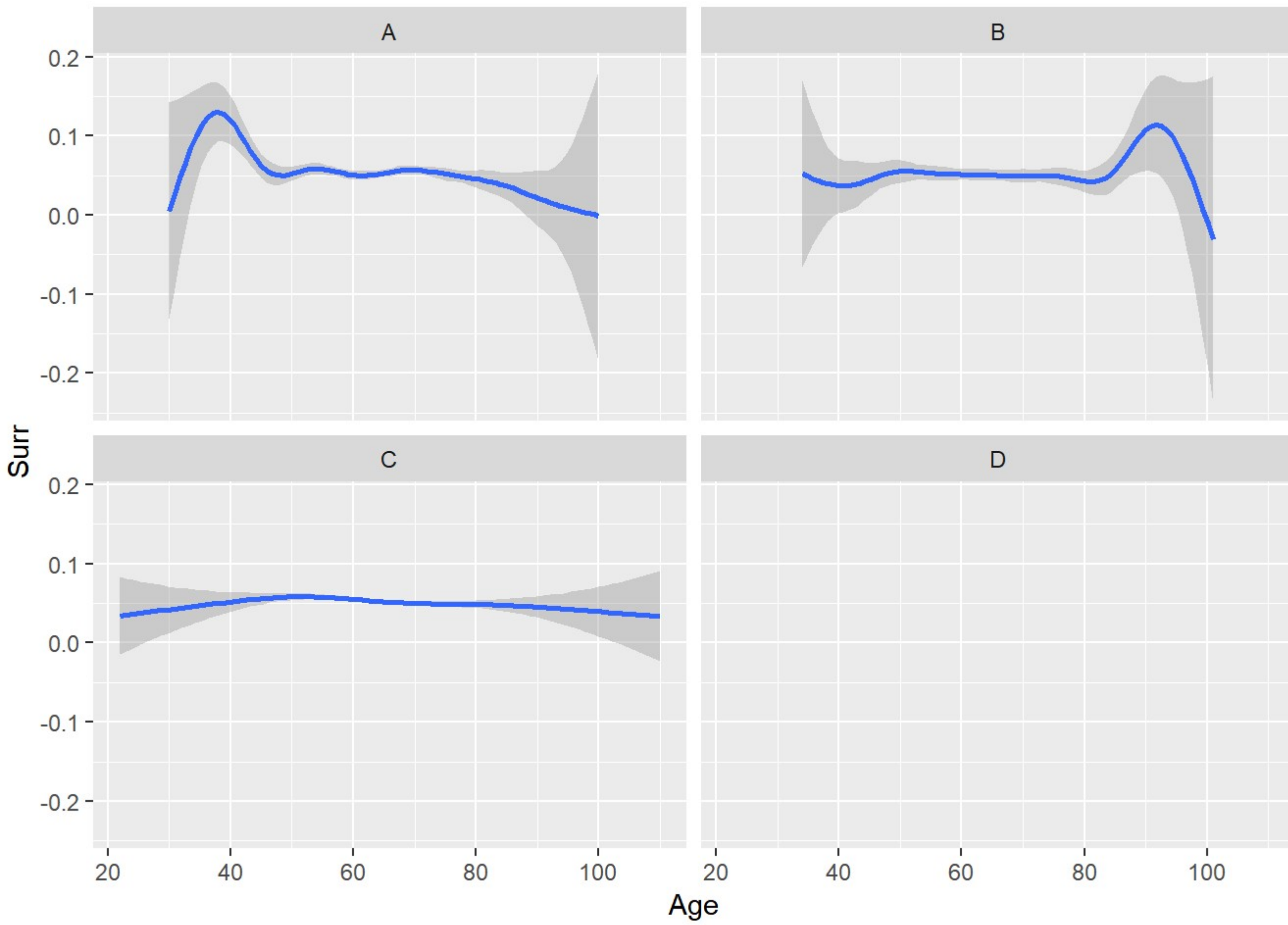
Just to be safe, I will check for multicollinearity between the numeric variables.

```
##              SCPeriod           AV           BB          Age
## SCPeriod  1.000000000 -0.002299307  0.0015388389 -0.0044517657
## AV       -0.002299307  1.000000000  0.8361767010  0.0017410051
## BB        0.001538839  0.836176701  1.0000000000  0.0003492089
## Age      -0.004451766  0.001741005  0.0003492089  1.0000000000
## q        -0.005448204 -0.003173014 -0.0019749521  0.0018272871
## Surr     -0.035913066 -0.042862186 -0.0553721499 -0.0124706410
## ITM      -0.006656554  0.501789052 -0.0147055066  0.0028255249
##                     q         Surr          ITM
## SCPeriod -0.005448204 -0.035913066 -0.006656554
## AV       -0.003173014 -0.042862186  0.501789052
## BB       -0.001974952 -0.055372150 -0.014705507
## Age       0.001827287 -0.012470641  0.002825525
## q         1.000000000  0.092658130 -0.002407825
## Surr      0.092658130  1.000000000  0.008217657
## ITM      -0.002407825  0.008217657  1.000000000
```

AV and BB are strongly correlated. AV also has some significant correlation with ITM, while BB does not. I think it may be more effective to use BB only instead of BB and AV in the model.

There are a few interaction effects that I think may be important.

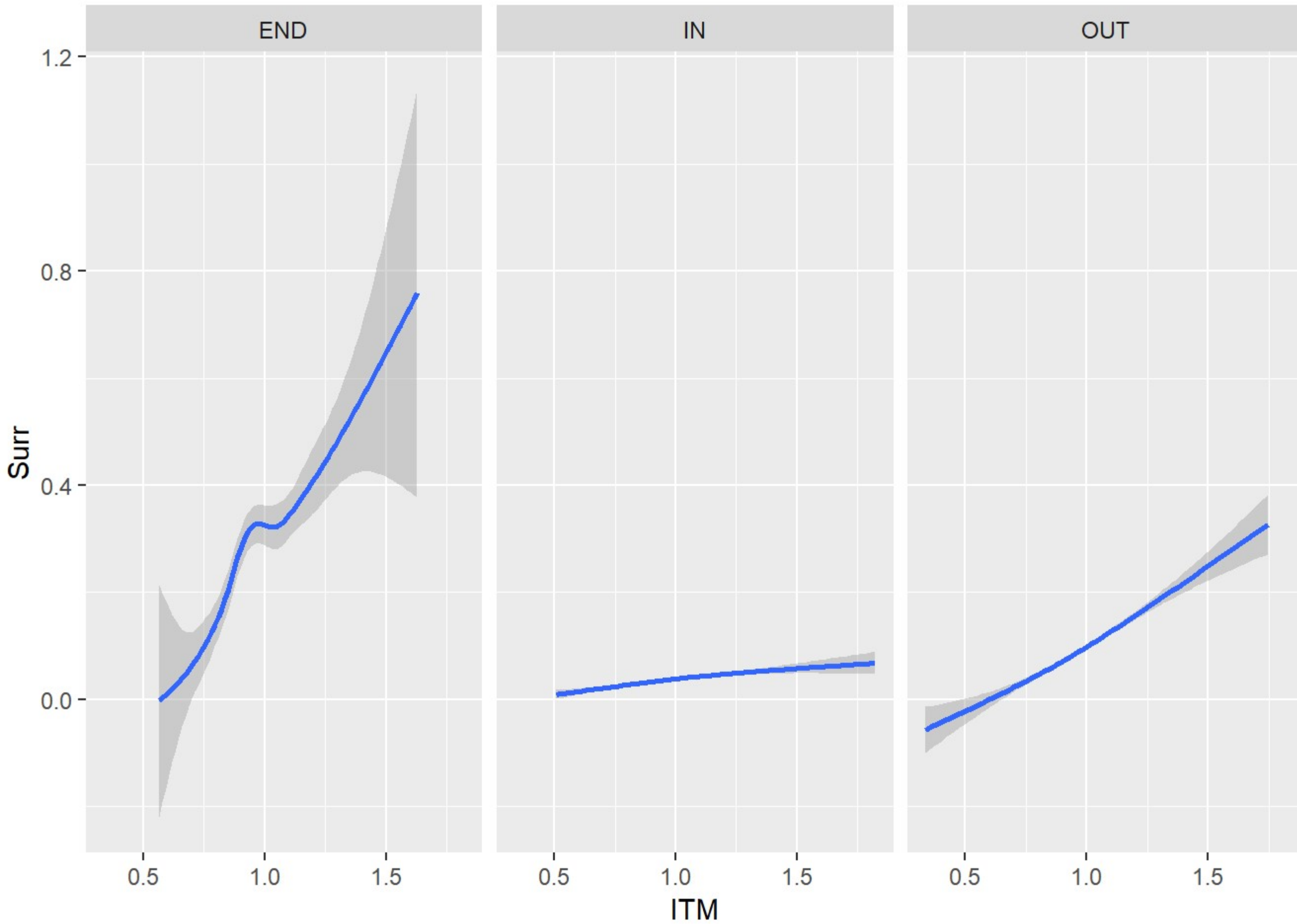
The different riders have different withdrawal rates at each age, so the effect of age may change depending on the rider.



```
##
## Call:
## glm(formula = Surr ~ Age + RiderCode + Age * RiderCode, family = "binomial",
##     data = data2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.3781  -0.3355  -0.3275  -0.3216   2.4985
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -2.400911   0.200846 -11.954   <2e-16 ***
## Age            -0.007326   0.003112  -2.354   0.0186 *
## RiderCodeB     -0.447849   0.355489  -1.260   0.2077
## RiderCodeC     -0.054802   0.238394  -0.230   0.8182
## Age:RiderCodeB  0.006057   0.005479   1.105   0.2690
## Age:RiderCodeC  0.000655   0.003691   0.177   0.8592
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 41341  on 99998  degrees of freedom
## Residual deviance: 41323  on 99993  degrees of freedom
## AIC: 41335
##
## Number of Fisher Scoring iterations: 5
```

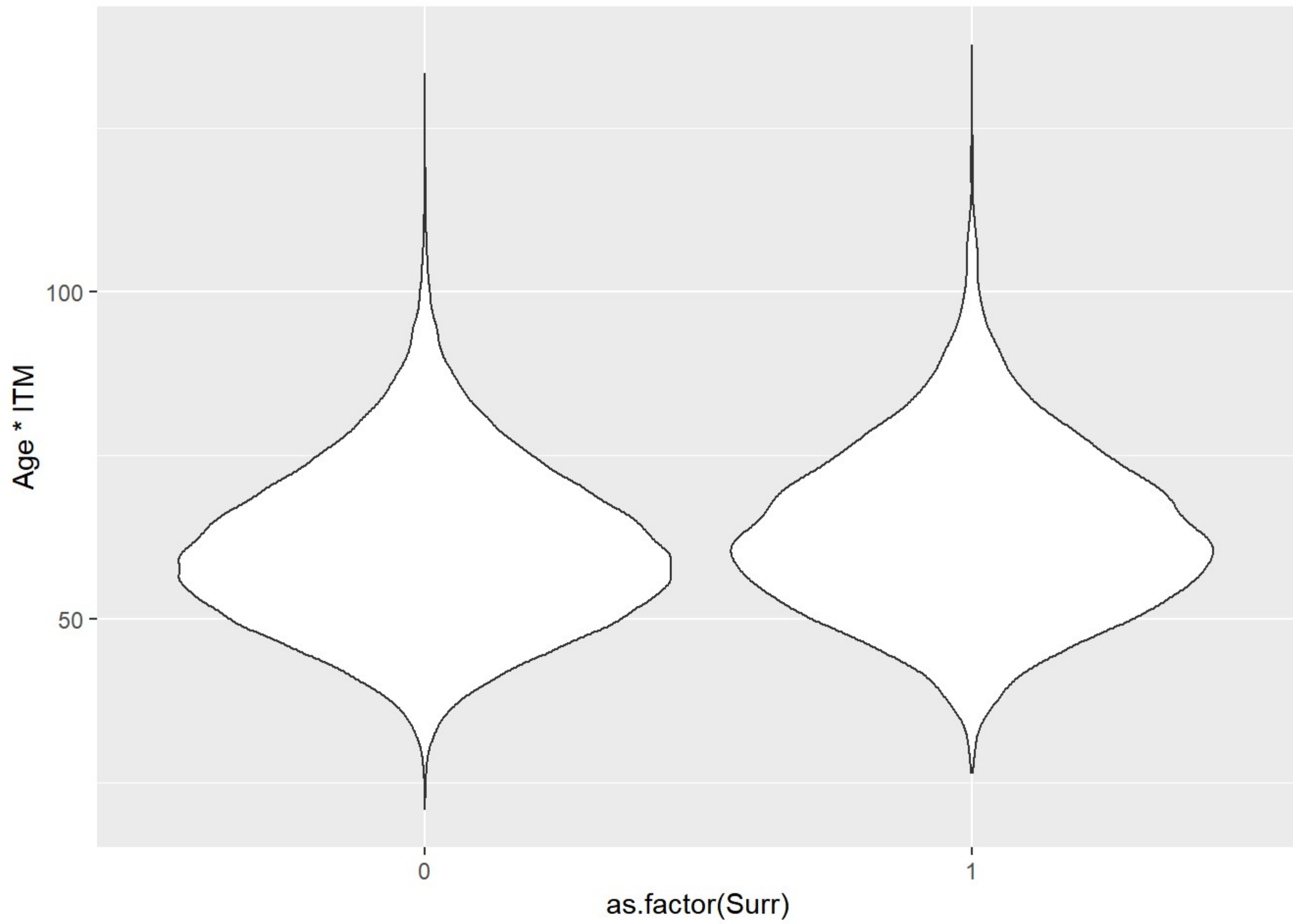
In the middle chunk of ages where most of the data is, all 3 riders seem to hover around a 5% surrender rate. There may be a slight difference on the tail ends, but it doesn't seem to be significant. Ridercode no longer needs to be included in the model.

Surrender rates appear to increase as ITM increases. However, I don't think this effect will be the same across all surrender charge phases. If someone is still in the surrender phase, it makes sense that their annuity would need to be more in the money to make the surrender charge worth taking.



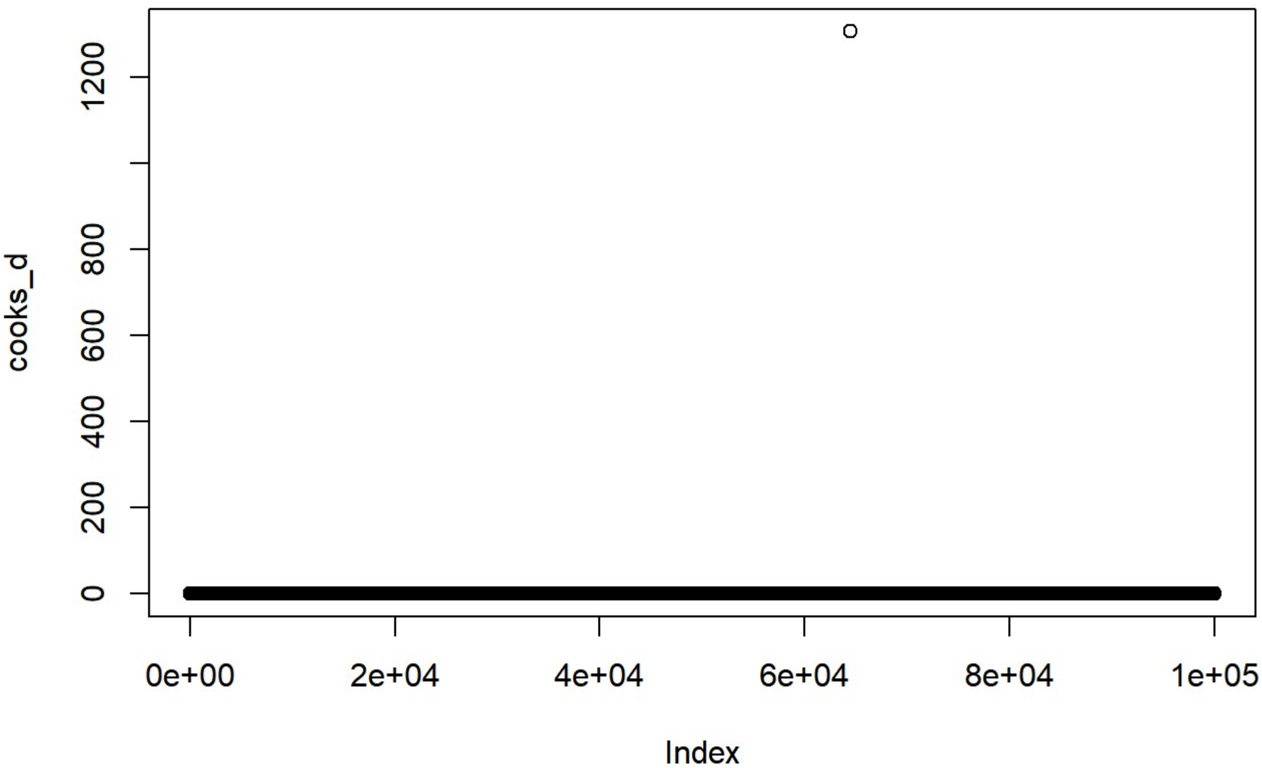

As I thought, there is a very clear difference in slope between all three groups. Out and End are relatively similar to each other, but In has a much weaker slope.

I think the interaction between Age and ITM is also worth exploring. If an older person’s annuity is more in the money, it would make sense if they were more likely to take all of the money at once.




There is a clear upward shift in Age*ITM for the surrender group. It may be minor, but I think it looks significant.

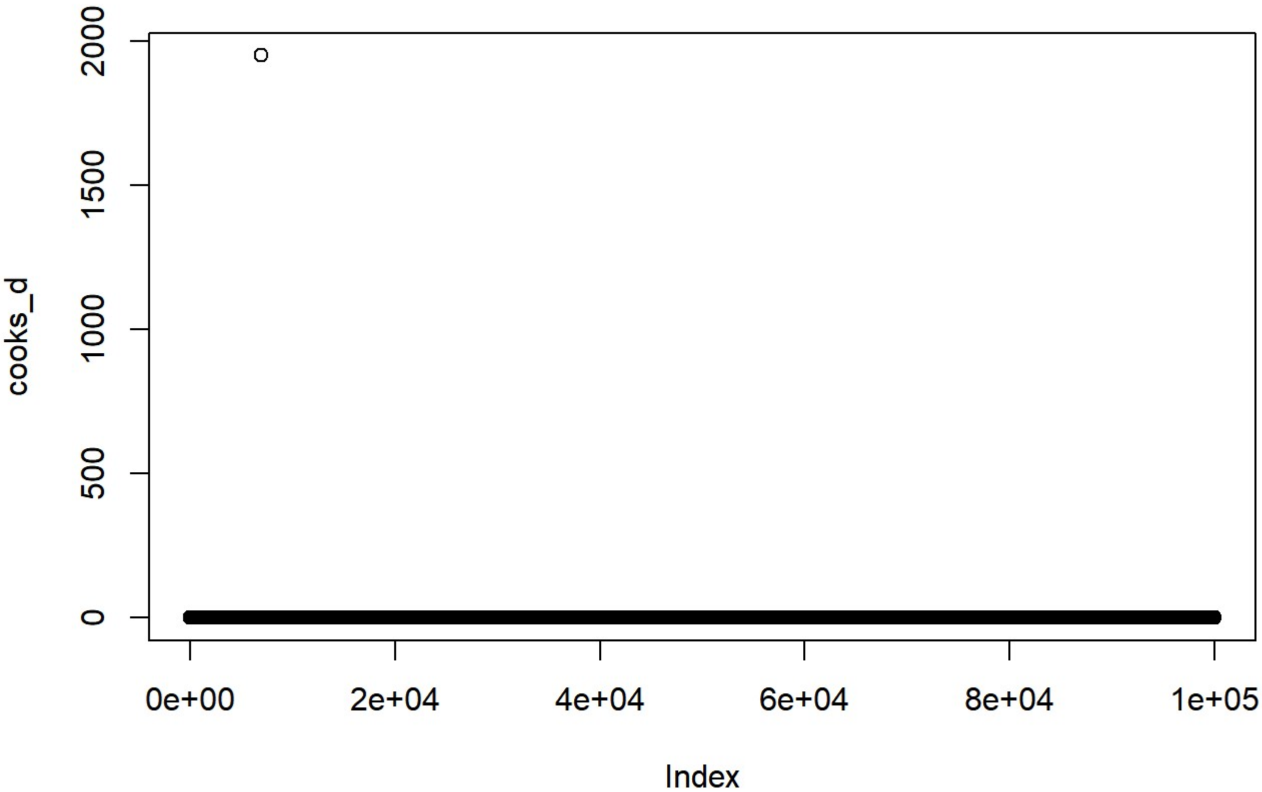
Before I start choosing my final model, I am going to check for any influential observations using Cook's Distance.




One of these observations has a gigantic Cook's Distance. There may be some other values that are too large as well, so I will look at the 20 largest.

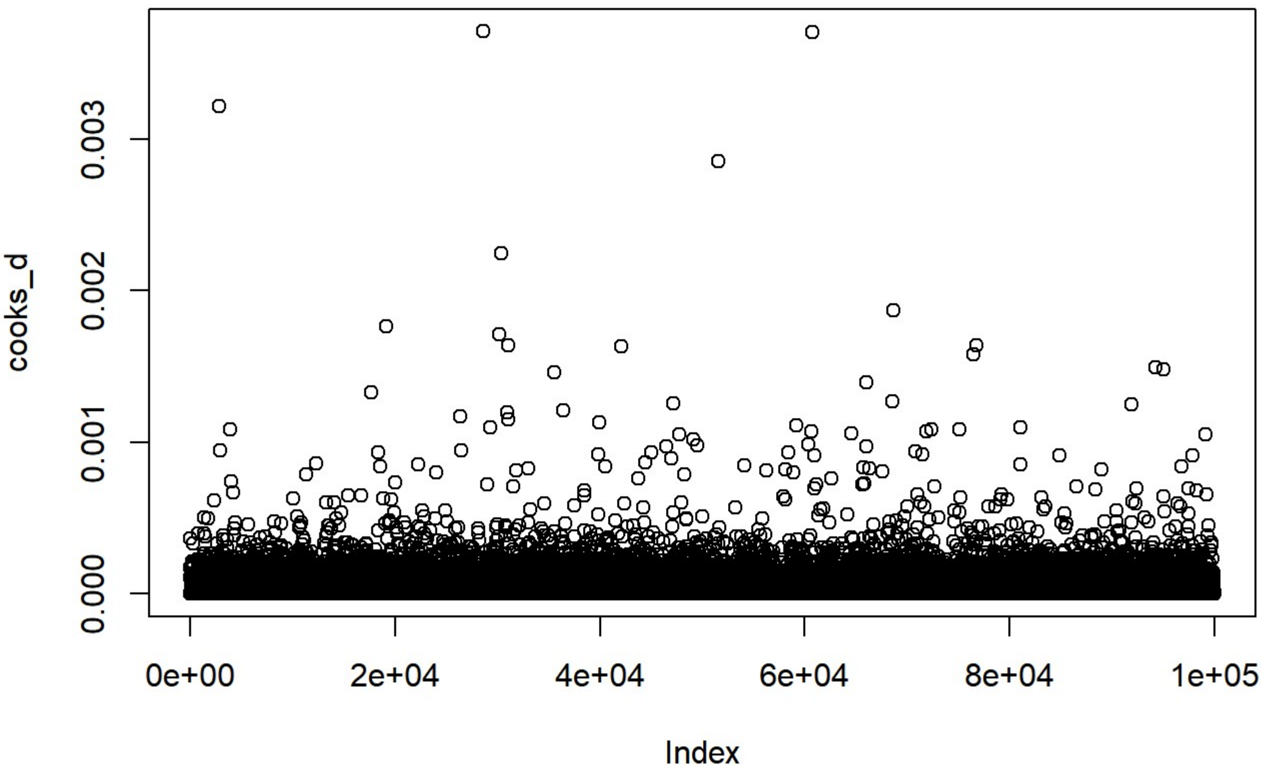
```
##         64438         28659         60775          2878         51529
## -1.305878e+03 -3.738945e-03 -3.723556e-03 -3.460609e-03 -2.881783e-03
##         30428         19165         76794         42119         30154
## -2.272696e-03 -1.720925e-03 -1.655508e-03 -1.650765e-03 -1.644831e-03
##         31031         76433         94261         35562         95034
## -1.627575e-03 -1.572426e-03 -1.462783e-03 -1.403338e-03 -1.357870e-03
##         17695         66011         47203         68593         36425
## -1.349679e-03 -1.339519e-03 -1.276788e-03 -1.227229e-03 -1.220907e-03
```

It looks like observation 64,438 is the only one with an abnormally large Cook's distance, so I will remove it.




```
##          6930         28659         60775          2878         51529
## -1.951305e+03 -3.733363e-03 -3.724242e-03 -3.603381e-03 -2.875803e-03
##         30428         19165         30154         76794         42119
## -2.263309e-03 -1.708748e-03 -1.681859e-03 -1.652707e-03 -1.647623e-03
##         31031         76433         35562         94261         66011
## -1.621897e-03 -1.518947e-03 -1.461646e-03 -1.461470e-03 -1.342151e-03
##         17695         47203         68593         95034         36425
## -1.340513e-03 -1.273348e-03 -1.252805e-03 -1.248960e-03 -1.218404e-03
```

Now there is a new point with a high Cook's Distance, and it's even higher than the original one! Again, there is only one problematic point, so I will remove it.




All of the points now have very low Cook's Distance, so I will proceed.

I am going to test my model, which contains BB, Age, q, q^2, ITM, SCPeriod, SCPhase, SCPhase/ITM interaction and Age/ITM interaction against a model chosen from Lasso.

```
## 14 x 1 sparse Matrix of class "dgCMatrix"
##                             1
## (Intercept)    -3.345240e+00
## SCPeriod       -2.113540e-02
## AV             -3.927294e-07
## BB             -1.437765e-06
## Age            -6.048874e-03
## ITM             3.106303e+00
## SCPhaseIN      -4.186251e-01
## SCPhaseOUT     -1.406270e+00
## poly(q, 2)1     3.484241e+01
## poly(q, 2)2    -2.149871e+01
## cos(q)           .
## ITM:SCPhaseIN  -1.768770e+00
## ITM:SCPhaseOUT   .
## Age:ITM          .
```

It looks like Lasso picks similar predictors to my model, but it includes AV and removes the Age/ITM interaction effect.

First, I will use confusion matrices to compare the predictive power of each model.

My model

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 9424  570
##          1    3    2
##
##                Accuracy : 0.9427
##                  95% CI : (0.938, 0.9472)
##     No Information Rate : 0.9428
##     P-Value [Acc > NIR] : 0.5283
##
##                   Kappa : 0.0059
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.999682
##             Specificity : 0.003497
##          Pos Pred Value : 0.942966
##          Neg Pred Value : 0.400000
##              Prevalence : 0.942794
##          Detection Rate : 0.942494
##    Detection Prevalence : 0.999500
##       Balanced Accuracy : 0.501589
##
##        'Positive' Class : 0
##
```

Lasso Model

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 9454  537
##          1    2    6
##
##                Accuracy : 0.9461
##                  95% CI : (0.9415, 0.9504)
##     No Information Rate : 0.9457
##     P-Value [Acc > NIR] : 0.4412
##
##                   Kappa : 0.0202
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.99979
##             Specificity : 0.01105
##          Pos Pred Value : 0.94625
##          Neg Pred Value : 0.75000
##              Prevalence : 0.94569
##          Detection Rate : 0.94549
##    Detection Prevalence : 0.99920
##       Balanced Accuracy : 0.50542
##
##        'Positive' Class : 0
##
```

Both of these models seem to have very high sensitivity and very low specificity, which is unusual. This is likely because the default cutoff of 0.5 is ineffective here. The Lasso Model does slightly better on both counts.

I will also look at Pseudo R^2 and the ROC curves

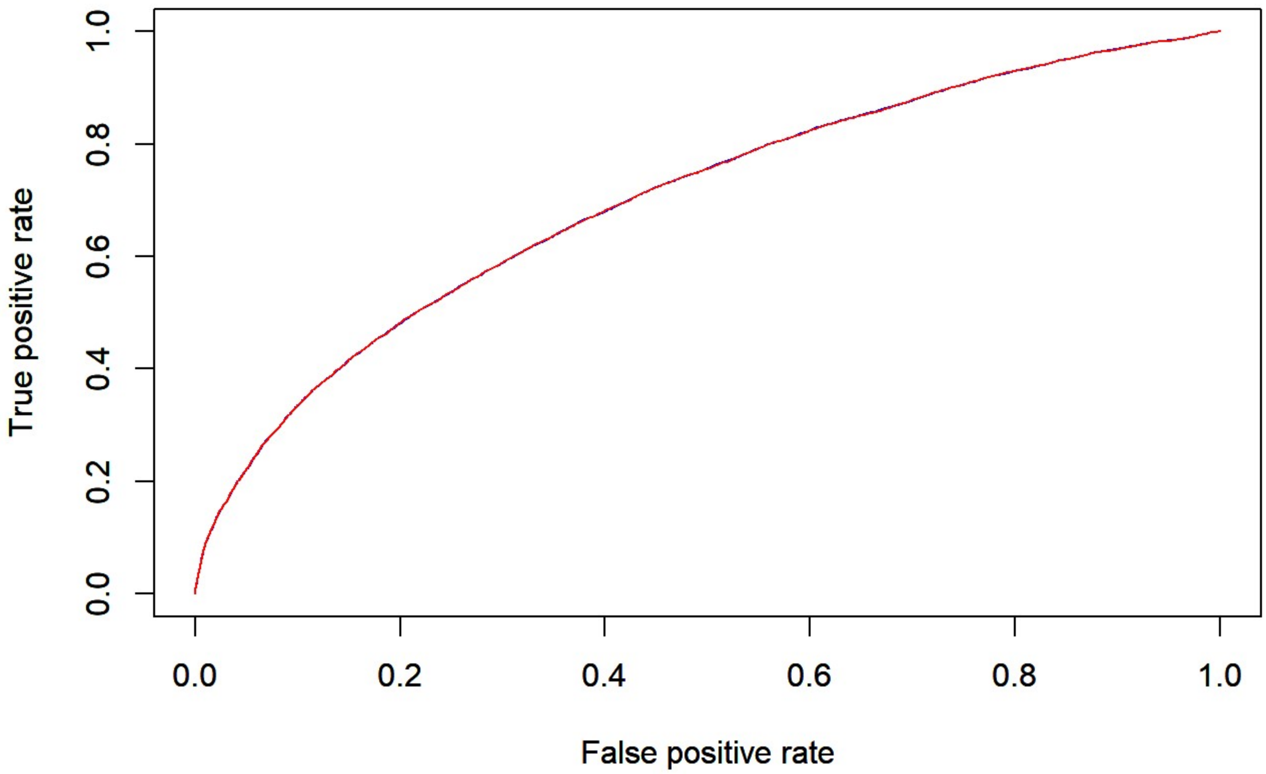
Pseudo R^2 for My Model

```
##           llh       llhNull            G2      McFadden          r2ML
## -1.914946e+04 -2.067058e+04  3.042234e+03  7.358851e-02  2.996513e-02
##          r2CU
##  8.849236e-02
```

Pseudo R^2 for Lasso Model

```
##           llh       llhNull            G2      McFadden          r2ML
## -1.914981e+04 -2.067058e+04  3.041551e+03  7.357197e-02  2.995850e-02
##          r2CU
##  8.847278e-02
```

The McFadden R^2 statistic for both models is very low (under 0.08), but it is slightly higher for my model. However, the difference is negligible.




The ROC curves for each model are so similar, that it's impossible to visually determine which one is better. I will calculate the AUC for each model.

AUC for My Model

```
## [1] 0.700278
```

AUC for Lasso Model

```
## [1] 0.7002345
```

My model has slightly higher AUC, but the difference is so small that it's negligible. The AUC is around 0.7, which isn't very impressive.

Overall, my model appears to be slightly better. However, the improvement is so negligible, it wouldn't matter much which model is used.