Abhinav\_Kashyap - Midterm

48202300

#Read data  
high\_note <- read.csv("~/Desktop/UCI/Coursework/BANA 277 LEC A- CUST & SOCIAL ANLYTICS/Midterm/HighNote Data Midterm.csv")  
  
str(high\_note)

## 'data.frame': 43827 obs. of 16 variables:  
## $ ID : int 1 2 3 4 5 6 7 8 9 10 ...  
## $ age : int 22 35 27 21 24 21 20 23 24 34 ...  
## $ male : int 0 0 1 0 0 1 0 1 0 1 ...  
## $ friend\_cnt : int 8 2 2 28 65 12 15 57 4 13 ...  
## $ avg\_friend\_age : num 22.6 28 23 22.9 22.3 ...  
## $ avg\_friend\_male : num 0.429 1 1 0.5 0.914 ...  
## $ friend\_country\_cnt : int 1 2 1 7 9 1 1 14 1 3 ...  
## $ subscriber\_friend\_cnt: int 0 0 0 1 0 0 0 1 0 0 ...  
## $ songsListened : int 9687 0 508 1357 89984 124547 24852 99877 6125 15997 ...  
## $ lovedTracks : int 194 0 0 32 20 10 391 125 42 82 ...  
## $ posts : int 0 0 0 0 2 0 6 89 0 0 ...  
## $ playlists : int 1 0 1 0 0 1 1 1 0 1 ...  
## $ shouts : int 8 0 2 1 81 2 67 44 5 3 ...  
## $ adopter : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ tenure : int 59 35 42 25 67 53 56 71 34 49 ...  
## $ good\_country : int 1 0 0 0 0 1 1 0 1 1 ...

#Part 1 - Summary Statistics  
library(psych)  
describeBy(high\_note, high\_note$adopter)

##   
## Descriptive statistics by group   
## group: 0  
## vars n mean sd median trimmed  
## ID 1 40300 20150.50 11633.75 20150.50 20150.50  
## age 2 40300 23.95 6.37 23.00 23.09  
## male 3 40300 0.62 0.48 1.00 0.65  
## friend\_cnt 4 40300 18.49 57.48 7.00 10.28  
## avg\_friend\_age 5 40300 24.01 5.10 23.00 23.40  
## avg\_friend\_male 6 40300 0.62 0.32 0.67 0.65  
## friend\_country\_cnt 7 40300 3.96 5.76 2.00 2.66  
## subscriber\_friend\_cnt 8 40300 0.42 2.42 0.00 0.13  
## songsListened 9 40300 17589.44 28416.02 7440.00 11817.64  
## lovedTracks 10 40300 86.82 263.58 14.00 36.35  
## posts 11 40300 5.29 104.31 0.00 0.23  
## playlists 12 40300 0.55 1.07 0.00 0.45  
## shouts 13 40300 29.97 150.69 4.00 8.84  
## adopter 14 40300 0.00 0.00 0.00 0.00  
## tenure 15 40300 43.81 19.79 44.00 43.72  
## good\_country 16 40300 0.36 0.48 0.00 0.32  
## mad min max range skew kurtosis se  
## ID 14937.19 1 40300 40299 0.00 -1.20 57.95  
## age 4.45 8 79 71 1.97 6.80 0.03  
## male 0.00 0 1 1 -0.50 -1.75 0.00  
## friend\_cnt 7.41 1 4957 4956 32.67 2087.42 0.29  
## avg\_friend\_age 3.95 8 77 69 1.84 7.15 0.03  
## avg\_friend\_male 0.35 0 1 1 -0.52 -0.72 0.00  
## friend\_country\_cnt 1.48 0 129 129 4.74 38.29 0.03  
## subscriber\_friend\_cnt 0.00 0 309 309 72.19 8024.62 0.01  
## songsListened 10576.87 0 1000000 1000000 6.05 105.85 141.55  
## lovedTracks 20.76 0 12522 12522 13.12 335.93 1.31  
## posts 0.00 0 12309 12309 73.92 7005.34 0.52  
## playlists 0.00 0 98 98 28.21 1945.28 0.01  
## shouts 4.45 0 7736 7736 22.53 779.12 0.75  
## adopter 0.00 0 0 0 NaN NaN 0.00  
## tenure 22.24 1 111 110 0.05 -0.70 0.10  
## good\_country 0.00 0 1 1 0.59 -1.65 0.00  
## --------------------------------------------------------   
## group: 1  
## vars n mean sd median trimmed  
## ID 1 3527 42064.00 1018.30 42064.00 42064.00  
## age 2 3527 25.98 6.84 24.00 25.05  
## male 3 3527 0.73 0.44 1.00 0.79  
## friend\_cnt 4 3527 39.73 117.27 16.00 23.69  
## avg\_friend\_age 5 3527 25.44 5.21 24.36 24.83  
## avg\_friend\_male 6 3527 0.64 0.25 0.67 0.65  
## friend\_country\_cnt 7 3527 7.19 8.86 4.00 5.36  
## subscriber\_friend\_cnt 8 3527 1.64 5.85 0.00 0.84  
## songsListened 9 3527 33758.04 43592.73 20908.00 25811.69  
## lovedTracks 10 3527 264.34 491.43 108.00 161.68  
## posts 11 3527 21.20 221.99 0.00 1.44  
## playlists 12 3527 0.90 2.56 1.00 0.59  
## shouts 13 3527 99.44 1156.07 9.00 23.89  
## adopter 14 3527 1.00 0.00 1.00 1.00  
## tenure 15 3527 45.58 20.04 46.00 45.60  
## good\_country 16 3527 0.29 0.45 0.00 0.23  
## mad min max range skew kurtosis se  
## ID 1307.65 40301 43827 3526 0.00 -1.20 17.15  
## age 4.45 8 73 65 1.68 4.39 0.12  
## male 0.00 0 1 1 -1.03 -0.94 0.01  
## friend\_cnt 17.79 1 5089 5088 26.04 1013.79 1.97  
## avg\_friend\_age 3.91 12 62 50 1.68 5.05 0.09  
## avg\_friend\_male 0.25 0 1 1 -0.54 -0.05 0.00  
## friend\_country\_cnt 4.45 0 136 136 3.61 24.53 0.15  
## subscriber\_friend\_cnt 0.00 0 287 287 34.05 1609.52 0.10  
## songsListened 23276.82 0 817290 817290 4.71 46.64 734.03  
## lovedTracks 140.85 0 10220 10220 6.52 80.96 8.27  
## posts 0.00 0 8506 8506 26.52 852.38 3.74  
## playlists 1.48 0 118 118 28.84 1244.31 0.04  
## shouts 11.86 0 65872 65872 52.52 2969.09 19.47  
## adopter 0.00 1 1 0 NaN NaN 0.00  
## tenure 20.76 0 111 111 0.02 -0.62 0.34  
## good\_country 0.00 0 1 1 0.94 -1.12 0.01

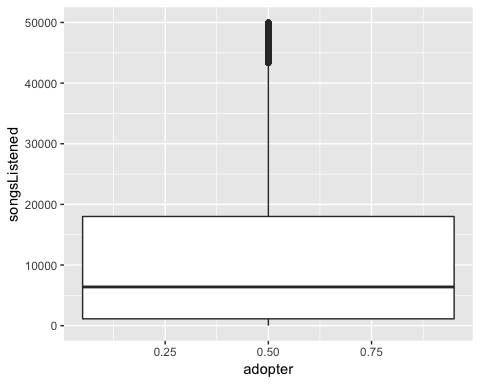
**Conclusions:**

So, of the total 43,827 customer data, only 3527 customers adopted for the premium subscription. Upon looking at the difference between customers who adopt and those who didn’t, we can say that all the people who adopt the premium subscription:

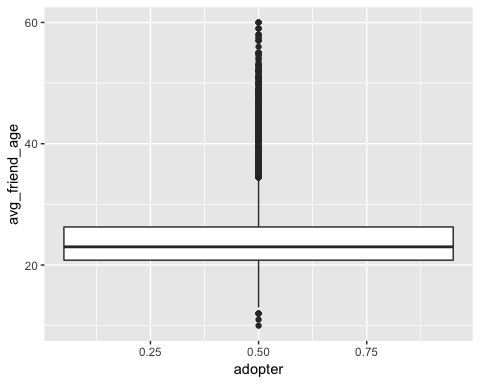
* Have higher number of friends on average
* Have friends from different countries (7 on average compared to 3 of non-subscribers)
* Listen to more songs on average and also have a high activity on the app such as more loved songs, more playlists, more shouts and more posts

#Part 2 - Data Visualizations  
library(ggplot2)

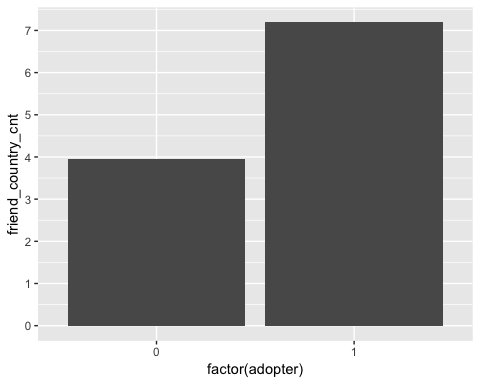
#Songs listened by adopters and non-adopters  
ggplot(high\_note, aes(x=adopter, y=songsListened)) + geom\_boxplot() + ylim(c(0,50000))



#Average friend age by adopters and non-adopters  
ggplot(high\_note, aes(x=adopter, y=avg\_friend\_age)) + geom\_boxplot() + ylim(c(10,60))

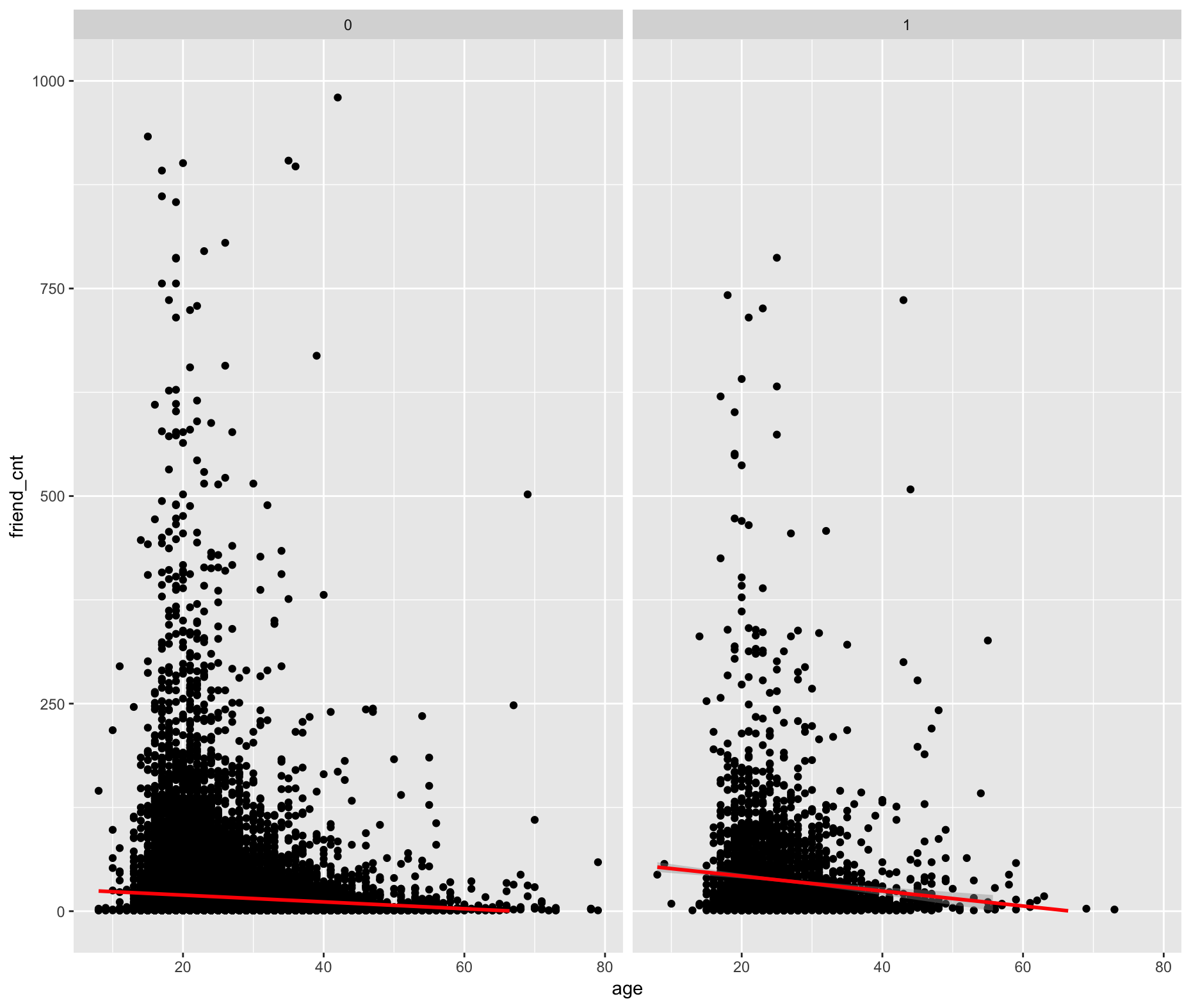


# Average number of countries where friends belong from by adopters and non-adopters  
ggplot(high\_note, aes(x=factor(adopter), y=friend\_country\_cnt)) + stat\_summary(fun.y = mean, geom = 'bar') + scale\_y\_continuous(breaks = seq(0,8, by = 1))



The above 3 visualizations re-confirms our conclusion from Part 1 – General statistics.

#Scatterplot of number of friends by adopters and non-adopters - Indicates that younger users on the platform have more friends  
ggplot(high\_note, aes(x = age, y = friend\_cnt)) + geom\_point() + facet\_wrap(~ adopter)+ ylim(c(0, 1000)) + geom\_smooth(method = 'lm', color='red')

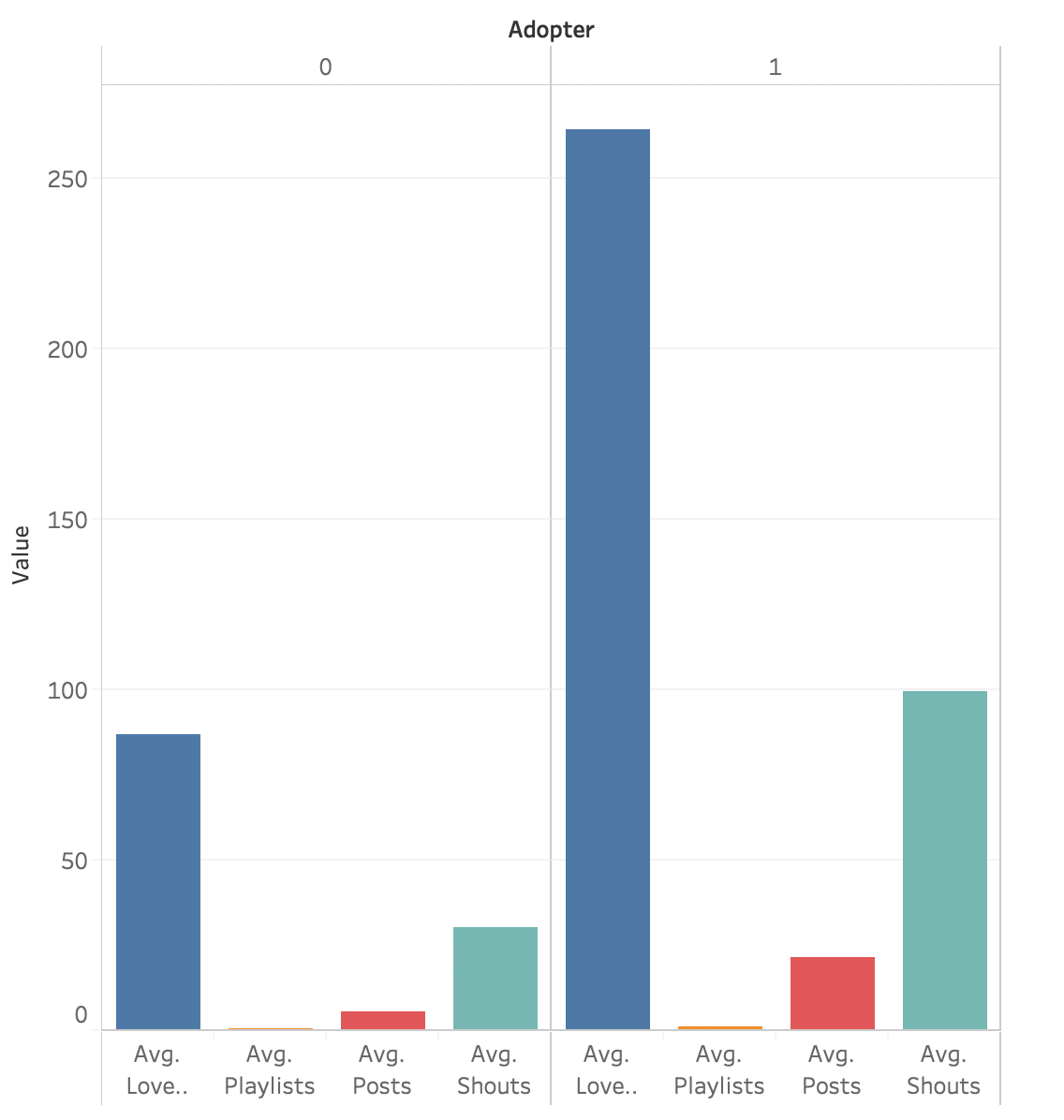


The above visualization suggests that most of the users are approximately aged around 18-25 years. The red line for adopter is slightly higher indicating more number of friends for adopters than non-adopters.

Now we look at some visualizations on Tableau:

The average usage of application resources by adopter and non-adopters:

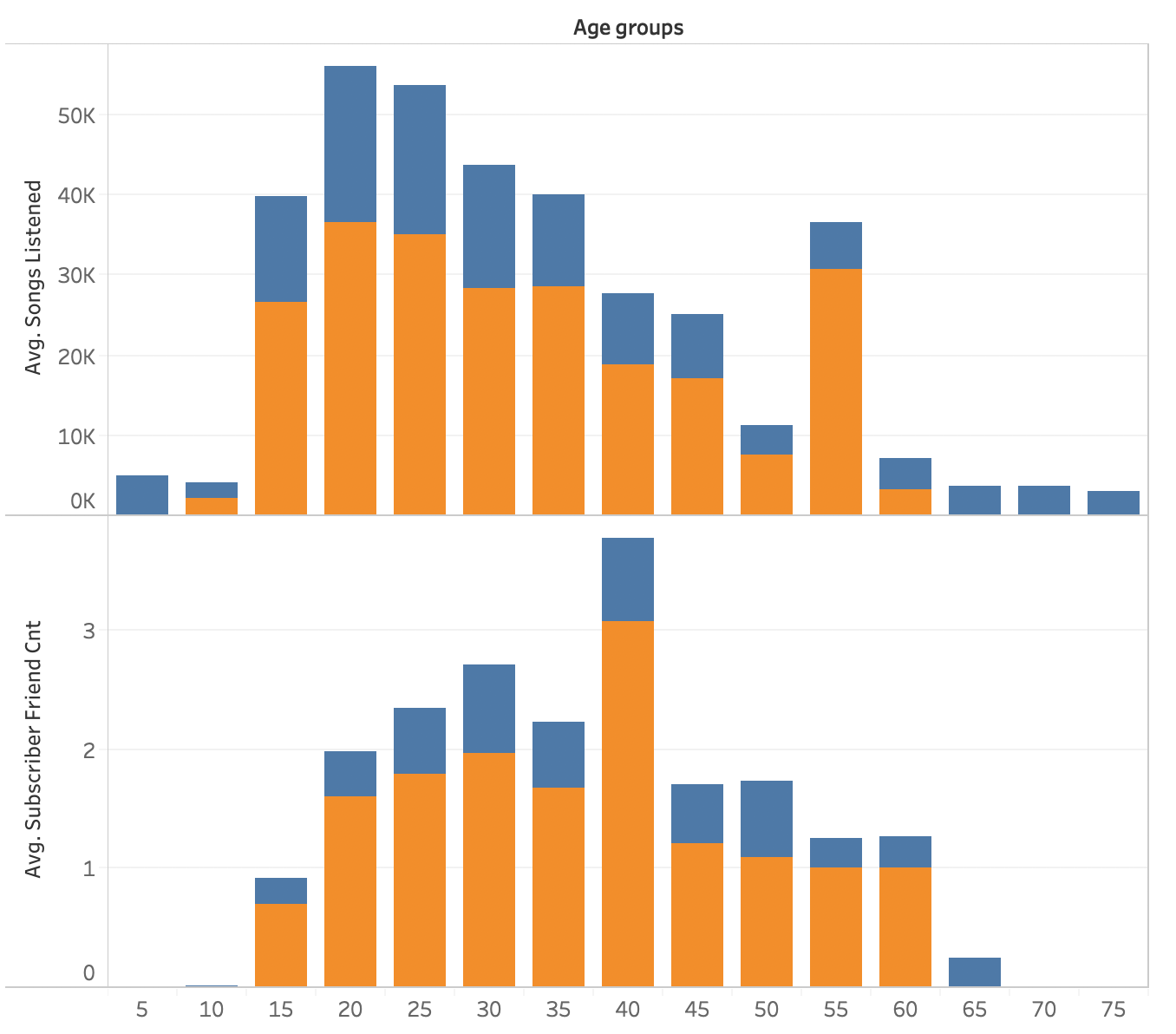
The graph below indicates that on average the adopters are much more active on the application. Thus, they tend to love more songs, make more posts, more playlists and also more shouts.

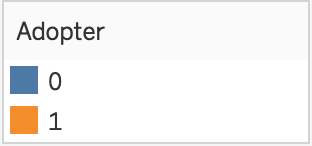


Average songs listened and average number of friends who are subscribers by age groups:

I divided the age groups into bins with a range of 5 years and plotted the average count for songs listened and number of subscriber friends. We can observe that most of the people who are active on the application are within the age groups 20-40.

Adopters usually listen to many more songs and also have more subscriber friends in their group, specially people who are aged around 40 years.





#Part 3 - Propensity score matching  
  
#Converting the subscriber friends column into treatement and control group  
#Treatment Group = Subscriber\_friend\_cnt > 1  
#Control Group = Subscriber\_friend\_cnt = 0  
  
#Creating new variable treatment where treatment group = 1 and control group = 0  
high\_note$treatment <- ifelse(high\_note$subscriber\_friend\_cnt >0,1,0)  
  
#Difference in means: Output variable 'adopter'  
library(magrittr)  
library(dplyr)

high\_note %>%  
 group\_by(treatment) %>%  
 summarise(n\_people = n(),  
 mean\_adopter = mean(adopter),  
 std\_error = sd(adopter)/sqrt(n\_people))

## # A tibble: 2 x 4  
## treatment n\_people mean\_adopter std\_error  
## <dbl> <int> <dbl> <dbl>  
## 1 0 34004 0.0524 0.00121  
## 2 1 9823 0.178 0.00386

#Indicates that the mean of treatment group is very large compared to the control group  
  
#t-test  
with(high\_note, t.test(adopter ~ treatment))

##   
## Welch Two Sample t-test  
##   
## data: adopter by treatment  
## t = -30.961, df = 11815, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -0.1330281 -0.1171869  
## sample estimates:  
## mean in group 0 mean in group 1   
## 0.05243501 0.17754250

#The difference in means is statistically significant at conventional levels of confidence

#Difference in means: Pre-treatment covariates  
#The following covariates shall be used:  
high\_note\_cov <- c('age', 'male', 'friend\_cnt', 'avg\_friend\_age', 'avg\_friend\_male', 'friend\_country\_cnt', 'subscriber\_friend\_cnt', 'songsListened', 'lovedTracks', 'posts', 'playlists', 'shouts', 'tenure', 'good\_country')  
  
high\_note %>%  
 group\_by(treatment) %>%  
 select(one\_of(high\_note\_cov)) %>%  
 summarise\_all(funs(mean(., na.rm = T)))

## Adding missing grouping variables: `treatment`

## # A tibble: 2 x 15  
## treatment age male friend\_cnt avg\_friend\_age avg\_friend\_male  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 0 23.7 0.629 10.4 23.8 0.613  
## 2 1 25.4 0.636 54.0 25.4 0.636  
## # ... with 9 more variables: friend\_country\_cnt <dbl>,  
## # subscriber\_friend\_cnt <dbl>, songsListened <dbl>, lovedTracks <dbl>,  
## # posts <dbl>, playlists <dbl>, shouts <dbl>, tenure <dbl>,  
## # good\_country <dbl>

#We can observe the people in treatment group have higher number of friends and from different countries. Also, they are more active on the platform as they listen to more song, make more posts, etc.  
  
#Carry out t-tests to evaluate whether these means are statistically distinguishable:  
lapply(high\_note\_cov, function(v) {  
 t.test(high\_note[, v] ~ high\_note[, 'treatment'])})

## [[1]]  
##   
## Welch Two Sample t-test  
##   
## data: high\_note[, v] by high\_note[, "treatment"]  
## t = -20.841, df = 14645, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -1.778544 -1.472749  
## sample estimates:  
## mean in group 0 mean in group 1   
## 23.74756 25.37321   
##   
##   
## [[2]]  
##   
## Welch Two Sample t-test  
##   
## data: high\_note[, v] by high\_note[, "treatment"]  
## t = -1.3459, df = 15986, p-value = 0.1784  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -0.018236129 0.003388028  
## sample estimates:  
## mean in group 0 mean in group 1   
## 0.6288378 0.6362618   
##   
##   
## [[3]]  
##   
## Welch Two Sample t-test  
##   
## data: high\_note[, v] by high\_note[, "treatment"]  
## t = -33.707, df = 9903.1, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -46.12459 -41.05469  
## sample estimates:  
## mean in group 0 mean in group 1   
## 10.43133 54.02097   
##   
##   
## [[4]]  
##   
## Welch Two Sample t-test  
##   
## data: high\_note[, v] by high\_note[, "treatment"]  
## t = -27.658, df = 15667, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -1.744514 -1.513611  
## sample estimates:  
## mean in group 0 mean in group 1   
## 23.76137 25.39043   
##   
##   
## [[5]]  
##   
## Welch Two Sample t-test  
##   
## data: high\_note[, v] by high\_note[, "treatment"]  
## t = -7.7114, df = 23020, p-value = 1.294e-14  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -0.02846397 -0.01692672  
## sample estimates:  
## mean in group 0 mean in group 1   
## 0.6131124 0.6358077   
##   
##   
## [[6]]  
##   
## Welch Two Sample t-test  
##   
## data: high\_note[, v] by high\_note[, "treatment"]  
## t = -65.05, df = 10372, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -6.861271 -6.459857  
## sample estimates:  
## mean in group 0 mean in group 1   
## 2.725062 9.385626   
##   
##   
## [[7]]  
##   
## Welch Two Sample t-test  
##   
## data: high\_note[, v] by high\_note[, "treatment"]  
## t = -39.894, df = 9822, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -2.413449 -2.187386  
## sample estimates:  
## mean in group 0 mean in group 1   
## 0.000000 2.300417   
##   
##   
## [[8]]  
##   
## Welch Two Sample t-test  
##   
## data: high\_note[, v] by high\_note[, "treatment"]  
## t = -41.505, df = 11447, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -20037.04 -18229.80  
## sample estimates:  
## mean in group 0 mean in group 1   
## 14602.22 33735.64   
##   
##   
## [[9]]  
##   
## Welch Two Sample t-test  
##   
## data: high\_note[, v] by high\_note[, "treatment"]  
## t = -31.265, df = 10585, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -170.1918 -150.1102  
## sample estimates:  
## mean in group 0 mean in group 1   
## 65.21365 225.36465   
##   
##   
## [[10]]  
##   
## Welch Two Sample t-test  
##   
## data: high\_note[, v] by high\_note[, "treatment"]  
## t = -7.3649, df = 9933.6, p-value = 1.914e-13  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -22.76492 -13.19424  
## sample estimates:  
## mean in group 0 mean in group 1   
## 2.543377 20.522956   
##   
##   
## [[11]]  
##   
## Welch Two Sample t-test  
##   
## data: high\_note[, v] by high\_note[, "treatment"]  
## t = -10.492, df = 11238, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -0.2546958 -0.1745100  
## sample estimates:  
## mean in group 0 mean in group 1   
## 0.5294671 0.7440700   
##   
##   
## [[12]]  
##   
## Welch Two Sample t-test  
##   
## data: high\_note[, v] by high\_note[, "treatment"]  
## t = -11.426, df = 9888.1, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -100.04703 -70.74591  
## sample estimates:  
## mean in group 0 mean in group 1   
## 16.42304 101.81951   
##   
##   
## [[13]]  
##   
## Welch Two Sample t-test  
##   
## data: high\_note[, v] by high\_note[, "treatment"]  
## t = -14.696, df = 15805, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -3.792309 -2.899752  
## sample estimates:  
## mean in group 0 mean in group 1   
## 43.20268 46.54871   
##   
##   
## [[14]]  
##   
## Welch Two Sample t-test  
##   
## data: high\_note[, v] by high\_note[, "treatment"]  
## t = 2.0956, df = 16030, p-value = 0.03613  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## 0.0007383591 0.0220968020  
## sample estimates:  
## mean in group 0 mean in group 1   
## 0.3546936 0.3432760

#It is observed that there is statistically significant difference in means of all the above covariates except the covariate 'male'  
  
#Propensity score estimation  
#Running a logit model  
m\_ps <- glm(treatment ~ age + male + friend\_cnt + avg\_friend\_age + avg\_friend\_male + friend\_country\_cnt + songsListened + lovedTracks + posts + playlists + shouts + adopter + tenure + good\_country, family = binomial(), data = high\_note)

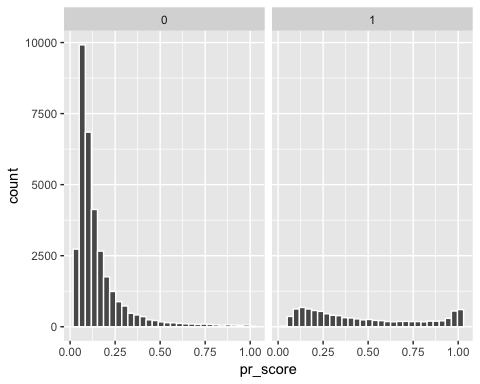
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

summary(m\_ps)

##   
## Call:  
## glm(formula = treatment ~ age + male + friend\_cnt + avg\_friend\_age +   
## avg\_friend\_male + friend\_country\_cnt + songsListened + lovedTracks +   
## posts + playlists + shouts + adopter + tenure + good\_country,   
## family = binomial(), data = high\_note)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.3540 -0.5624 -0.4144 -0.2958 2.5640   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -5.119e+00 7.753e-02 -66.020 < 2e-16 \*\*\*  
## age 1.812e-02 2.833e-03 6.394 1.62e-10 \*\*\*  
## male 1.569e-02 3.017e-02 0.520 0.60296   
## friend\_cnt 3.085e-02 1.038e-03 29.714 < 2e-16 \*\*\*  
## avg\_friend\_age 7.804e-02 3.499e-03 22.302 < 2e-16 \*\*\*  
## avg\_friend\_male 2.479e-01 5.060e-02 4.900 9.59e-07 \*\*\*  
## friend\_country\_cnt 1.106e-01 4.781e-03 23.127 < 2e-16 \*\*\*  
## songsListened 6.379e-06 5.184e-07 12.306 < 2e-16 \*\*\*  
## lovedTracks 5.553e-04 5.613e-05 9.893 < 2e-16 \*\*\*  
## posts 5.229e-04 2.612e-04 2.002 0.04525 \*   
## playlists -5.828e-03 1.271e-02 -0.459 0.64646   
## shouts -5.683e-05 3.817e-05 -1.489 0.13646   
## adopter 8.026e-01 4.429e-02 18.119 < 2e-16 \*\*\*  
## tenure -2.108e-03 7.804e-04 -2.701 0.00691 \*\*   
## good\_country 5.970e-02 2.940e-02 2.031 0.04229 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 46640 on 43826 degrees of freedom  
## Residual deviance: 33856 on 43812 degrees of freedom  
## AIC: 33886  
##   
## Number of Fisher Scoring iterations: 6

#Using the above model, we can calculate the propensity score for each person. It is simply the person's predicted probability of being treated, given the estimates from the logit model.  
#We use predict() and create a dataframe that has the propensity score and person's actual treatment status  
  
prs\_df <- data.frame(pr\_score = predict(m\_ps, type = 'response'),  
 treatment = m\_ps$model$treatment)  
  
#Histogram of estimated propensity scores by treatment status:  
prs\_df %>%  
 ggplot(aes(x = pr\_score)) +  
 geom\_histogram(color = "white") +  
 facet\_wrap(~treatment)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



The propensity score for treatment group is fairly constant while it is highly right-skewed for the control group.

#Creating a tableone pre-matching table  
library(tableone)  
table1 <- CreateTableOne(vars = high\_note\_cov, strata = "treatment", data = high\_note, test = FALSE)  
print(table1, smd = TRUE)

## Stratified by treatment  
## 0   
## n 34004   
## age (mean (sd)) 23.75 (6.22)   
## male (mean (sd)) 0.63 (0.48)   
## friend\_cnt (mean (sd)) 10.43 (15.28)   
## avg\_friend\_age (mean (sd)) 23.76 (5.06)   
## avg\_friend\_male (mean (sd)) 0.61 (0.33)   
## friend\_country\_cnt (mean (sd)) 2.73 (3.10)   
## subscriber\_friend\_cnt (mean (sd)) 0.00 (0.00)   
## songsListened (mean (sd)) 14602.22 (23214.29)  
## lovedTracks (mean (sd)) 65.21 (181.48)   
## posts (mean (sd)) 2.54 (33.79)   
## playlists (mean (sd)) 0.53 (0.97)   
## shouts (mean (sd)) 16.42 (79.74)   
## tenure (mean (sd)) 43.20 (19.72)   
## good\_country (mean (sd)) 0.35 (0.48)   
## Stratified by treatment  
## 1 SMD   
## n 9823   
## age (mean (sd)) 25.37 (6.97) 0.246  
## male (mean (sd)) 0.64 (0.48) 0.015  
## friend\_cnt (mean (sd)) 54.02 (127.91) 0.479  
## avg\_friend\_age (mean (sd)) 25.39 (5.17) 0.319  
## avg\_friend\_male (mean (sd)) 0.64 (0.23) 0.079  
## friend\_country\_cnt (mean (sd)) 9.39 (10.01) 0.899  
## subscriber\_friend\_cnt (mean (sd)) 2.30 (5.72) 0.569  
## songsListened (mean (sd)) 33735.64 (43952.34) 0.544  
## lovedTracks (mean (sd)) 225.36 (498.23) 0.427  
## posts (mean (sd)) 20.52 (241.27) 0.104  
## playlists (mean (sd)) 0.74 (1.96) 0.139  
## shouts (mean (sd)) 101.82 (739.51) 0.162  
## tenure (mean (sd)) 46.55 (19.92) 0.169  
## good\_country (mean (sd)) 0.34 (0.47) 0.024

#The SMD or Standardized Mean differences indicates whether there is imbalance among the variables in the dataset. Variables with SMD > 0.1 show imbalance in the dataset and that is where we actually need to do the propensity score matching. So, we should consider all variables except 'male', 'avg\_friend\_male' and 'good\_country' for our matching algorithm.  
  
#Executing a matching algorithm  
library(MatchIt)  
  
high\_note\_nomiss <- high\_note %>%  
 select(treatment, adopter, one\_of(high\_note\_cov)) %>% na.omit()

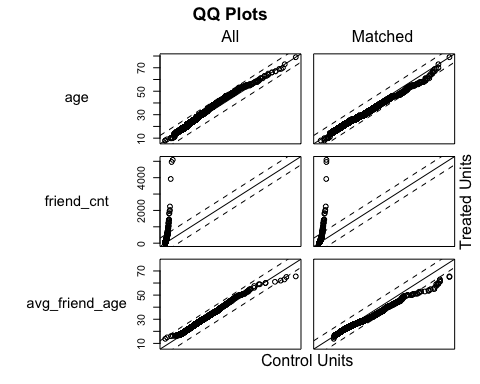
high\_note\_match <- subset(high\_note\_nomiss, select = -c(4,7,16))  
  
mod\_match <- matchit(treatment ~ age + friend\_cnt + avg\_friend\_age + friend\_country\_cnt + songsListened + lovedTracks + posts + playlists + shouts + tenure, method = 'nearest', data = high\_note\_match)

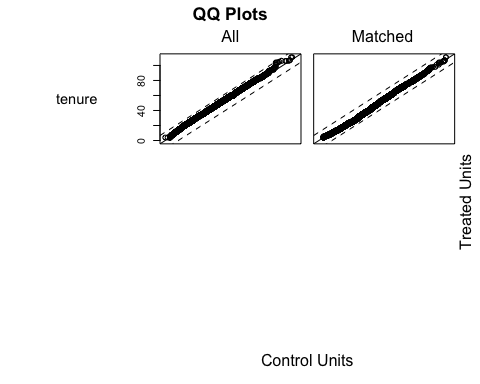
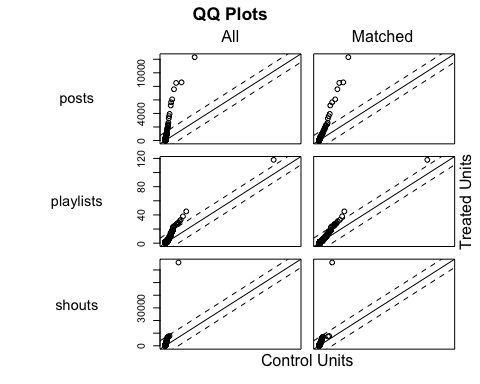
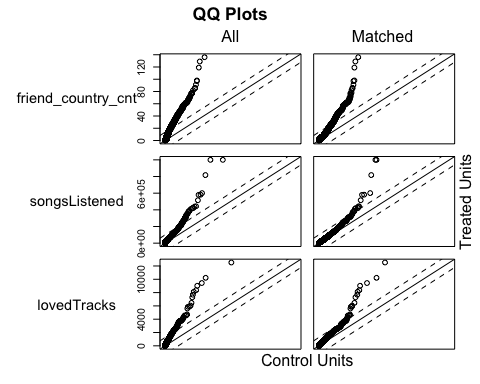
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

summary(mod\_match)

##   
## Call:  
## matchit(formula = treatment ~ age + friend\_cnt + avg\_friend\_age +   
## friend\_country\_cnt + songsListened + lovedTracks + posts +   
## playlists + shouts + tenure, data = high\_note\_match, method = "nearest")  
##   
## Summary of balance for all data:  
## Means Treated Means Control SD Control Mean Diff  
## distance 0.4631 0.1551 0.1437 0.3080  
## age 25.3732 23.7476 6.2245 1.6256  
## friend\_cnt 54.0210 10.4313 15.2769 43.5896  
## avg\_friend\_age 25.3904 23.7614 5.0577 1.6291  
## friend\_country\_cnt 9.3856 2.7251 3.1024 6.6606  
## songsListened 33735.6404 14602.2205 23214.2898 19133.4199  
## lovedTracks 225.3647 65.2137 181.4812 160.1510  
## posts 20.5230 2.5434 33.7947 17.9796  
## playlists 0.7441 0.5295 0.9673 0.2146  
## shouts 101.8195 16.4230 79.7381 85.3965  
## tenure 46.5487 43.2027 19.7212 3.3460  
## eQQ Med eQQ Mean eQQ Max  
## distance 0.2512 0.3080 0.6831  
## age 1.0000 1.6296 5.0000  
## friend\_cnt 22.0000 43.5838 4794.0000  
## avg\_friend\_age 1.5909 1.6369 11.5000  
## friend\_country\_cnt 5.0000 6.6598 95.0000  
## songsListened 15471.0000 19126.1623 653702.0000  
## lovedTracks 65.0000 159.9562 6343.0000  
## posts 0.0000 17.8829 9535.0000  
## playlists 0.0000 0.2092 26.0000  
## shouts 15.0000 85.1764 59168.0000  
## tenure 3.0000 3.3473 10.0000  
##   
##   
## Summary of balance for matched data:  
## Means Treated Means Control SD Control Mean Diff  
## distance 0.4631 0.3042 0.1917 0.1590  
## age 25.3732 26.3818 7.9222 -1.0086  
## friend\_cnt 54.0210 21.4487 23.5364 32.5722  
## avg\_friend\_age 25.3904 26.5966 6.7085 -1.2062  
## friend\_country\_cnt 9.3856 5.1003 4.6449 4.2854  
## songsListened 33735.6404 27204.2095 33850.1715 6531.4309  
## lovedTracks 225.3647 135.0392 299.0576 90.3255  
## posts 20.5230 6.1302 59.9354 14.3928  
## playlists 0.7441 0.6794 1.4089 0.0646  
## shouts 101.8195 37.7836 139.5482 64.0359  
## tenure 46.5487 47.9127 19.0388 -1.3639  
## eQQ Med eQQ Mean eQQ Max  
## distance 0.1086 0.1590 0.4492  
## age 1.0000 1.0086 7.0000  
## friend\_cnt 12.0000 32.5722 4794.0000  
## avg\_friend\_age 0.4916 1.2804 14.0000  
## friend\_country\_cnt 2.0000 4.2854 95.0000  
## songsListened 4854.0000 6531.4309 566867.0000  
## lovedTracks 38.0000 90.3255 6180.0000  
## posts 0.0000 14.3928 9535.0000  
## playlists 0.0000 0.1056 22.0000  
## shouts 10.0000 64.0359 59168.0000  
## tenure 1.0000 1.4152 4.0000  
##   
## Percent Balance Improvement:  
## Mean Diff. eQQ Med eQQ Mean eQQ Max  
## distance 48.3926 56.7805 48.3904 34.2371  
## age 37.9600 0.0000 38.1122 -40.0000  
## friend\_cnt 25.2753 45.4545 25.2653 0.0000  
## avg\_friend\_age 25.9587 69.1005 21.7815 -21.7391  
## friend\_country\_cnt 35.6608 60.0000 35.6533 0.0000  
## songsListened 65.8638 68.6252 65.8508 13.2836  
## lovedTracks 43.5998 41.5385 43.5311 2.5698  
## posts 19.9495 0.0000 19.5168 0.0000  
## playlists 69.8773 0.0000 49.5377 15.3846  
## shouts 25.0134 33.3333 24.8196 0.0000  
## tenure 59.2370 66.6667 57.7203 60.0000  
##   
## Sample sizes:  
## Control Treated  
## All 34004 9823  
## Matched 9823 9823  
## Unmatched 24181 0  
## Discarded 0 0

plot(mod\_match)





The QQ plots for treated and control groups indicate how well did our propensity score model do. An ideal case would be all points of the matched data lie on the diagonal line. In this case, we can observe that for each variable, some points have drifted towards the diagonal line suggesting that our algorithm has worked fine.

#Making a dataframe of the matched data  
dta\_m <- match.data(mod\_match)  
dim(dta\_m)

## [1] 19646 15

#difference in means  
dta\_m %>%  
 group\_by(treatment) %>%  
 select(one\_of(high\_note\_cov)) %>%  
 summarise\_all(funs(mean))

## Warning: Unknown columns: `male`, `avg\_friend\_male`, `good\_country`

## Adding missing grouping variables: `treatment`

## # A tibble: 2 x 12  
## treatment age friend\_cnt avg\_friend\_age friend\_country\_…  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 0 26.4 21.4 26.6 5.10  
## 2 1 25.4 54.0 25.4 9.39  
## # ... with 7 more variables: subscriber\_friend\_cnt <dbl>,  
## # songsListened <dbl>, lovedTracks <dbl>, posts <dbl>, playlists <dbl>,  
## # shouts <dbl>, tenure <dbl>

#t-test with all the covariates  
high\_note\_cov\_new <- c('age', 'friend\_cnt', 'avg\_friend\_age', 'friend\_country\_cnt', 'subscriber\_friend\_cnt', 'songsListened', 'lovedTracks', 'posts', 'playlists', 'shouts', 'tenure')  
  
lapply(high\_note\_cov\_new, function(v) {  
 t.test(dta\_m[, v] ~ dta\_m$treatment)})

## [[1]]  
##   
## Welch Two Sample t-test  
##   
## data: dta\_m[, v] by dta\_m$treatment  
## t = 9.4733, df = 19330, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## 0.7998745 1.2172282  
## sample estimates:  
## mean in group 0 mean in group 1   
## 26.38176 25.37321   
##   
##   
## [[2]]  
##   
## Welch Two Sample t-test  
##   
## data: dta\_m[, v] by dta\_m$treatment  
## t = -24.822, df = 10486, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -35.14443 -30.00003  
## sample estimates:  
## mean in group 0 mean in group 1   
## 21.44874 54.02097   
##   
##   
## [[3]]  
##   
## Welch Two Sample t-test  
##   
## data: dta\_m[, v] by dta\_m$treatment  
## t = 14.119, df = 18441, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## 1.038727 1.373631  
## sample estimates:  
## mean in group 0 mean in group 1   
## 26.59661 25.39043   
##   
##   
## [[4]]  
##   
## Welch Two Sample t-test  
##   
## data: dta\_m[, v] by dta\_m$treatment  
## t = -38.488, df = 13864, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -4.503599 -4.067103  
## sample estimates:  
## mean in group 0 mean in group 1   
## 5.100275 9.385626   
##   
##   
## [[5]]  
##   
## Welch Two Sample t-test  
##   
## data: dta\_m[, v] by dta\_m$treatment  
## t = -39.894, df = 9822, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -2.413449 -2.187386  
## sample estimates:  
## mean in group 0 mean in group 1   
## 0.000000 2.300417   
##   
##   
## [[6]]  
##   
## Welch Two Sample t-test  
##   
## data: dta\_m[, v] by dta\_m$treatment  
## t = -11.669, df = 18441, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -7628.575 -5434.287  
## sample estimates:  
## mean in group 0 mean in group 1   
## 27204.21 33735.64   
##   
##   
## [[7]]  
##   
## Welch Two Sample t-test  
##   
## data: dta\_m[, v] by dta\_m$treatment  
## t = -15.406, df = 16086, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -101.8176 -78.8333  
## sample estimates:  
## mean in group 0 mean in group 1   
## 135.0392 225.3647   
##   
##   
## [[8]]  
##   
## Welch Two Sample t-test  
##   
## data: dta\_m[, v] by dta\_m$treatment  
## t = -5.7379, df = 11030, p-value = 9.836e-09  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -19.309560 -9.475943  
## sample estimates:  
## mean in group 0 mean in group 1   
## 6.130205 20.522956   
##   
##   
## [[9]]  
##   
## Welch Two Sample t-test  
##   
## data: dta\_m[, v] by dta\_m$treatment  
## t = -2.6548, df = 17836, p-value = 0.007943  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -0.11237258 -0.01691582  
## sample estimates:  
## mean in group 0 mean in group 1   
## 0.6794258 0.7440700   
##   
##   
## [[10]]  
##   
## Welch Two Sample t-test  
##   
## data: dta\_m[, v] by dta\_m$treatment  
## t = -8.4334, df = 10521, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -78.91997 -49.15190  
## sample estimates:  
## mean in group 0 mean in group 1   
## 37.78357 101.81951   
##   
##   
## [[11]]  
##   
## Welch Two Sample t-test  
##   
## data: dta\_m[, v] by dta\_m$treatment  
## t = 4.9057, df = 19604, p-value = 9.383e-07  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## 0.8189771 1.9089064  
## sample estimates:  
## mean in group 0 mean in group 1   
## 47.91265 46.54871

#Creating a tableone for matching data  
table\_match1 <- CreateTableOne(vars = high\_note\_cov\_new, strata = 'treatment', data = dta\_m, test = FALSE)  
print(table\_match1, smd = TRUE)

## Stratified by treatment  
## 0   
## n 9823   
## age (mean (sd)) 26.38 (7.92)   
## friend\_cnt (mean (sd)) 21.45 (23.54)   
## avg\_friend\_age (mean (sd)) 26.60 (6.71)   
## friend\_country\_cnt (mean (sd)) 5.10 (4.64)   
## subscriber\_friend\_cnt (mean (sd)) 0.00 (0.00)   
## songsListened (mean (sd)) 27204.21 (33850.17)  
## lovedTracks (mean (sd)) 135.04 (299.06)   
## posts (mean (sd)) 6.13 (59.94)   
## playlists (mean (sd)) 0.68 (1.41)   
## shouts (mean (sd)) 37.78 (139.55)   
## tenure (mean (sd)) 47.91 (19.04)   
## Stratified by treatment  
## 1 SMD   
## n 9823   
## age (mean (sd)) 25.37 (6.97) 0.135  
## friend\_cnt (mean (sd)) 54.02 (127.91) 0.354  
## avg\_friend\_age (mean (sd)) 25.39 (5.17) 0.201  
## friend\_country\_cnt (mean (sd)) 9.39 (10.01) 0.549  
## subscriber\_friend\_cnt (mean (sd)) 2.30 (5.72) 0.569  
## songsListened (mean (sd)) 33735.64 (43952.34) 0.167  
## lovedTracks (mean (sd)) 225.36 (498.23) 0.220  
## posts (mean (sd)) 20.52 (241.27) 0.082  
## playlists (mean (sd)) 0.74 (1.96) 0.038  
## shouts (mean (sd)) 101.82 (739.51) 0.120  
## tenure (mean (sd)) 46.55 (19.92) 0.070

#Estimating treatment effects  
with(dta\_m, t.test(adopter ~ treatment))

##   
## Welch Two Sample t-test  
##   
## data: adopter by treatment  
## t = -18.469, df = 18149, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -0.09819244 -0.07935006  
## sample estimates:  
## mean in group 0 mean in group 1   
## 0.08877125 0.17754250

#We can observe that there is significant difference in means between treatment and control groups after running the propensity score matching. This indicates that having more subscriber friends increases the chances of any person adopting the service i.e., becoming a premium member.

#The above result can also be verified using OLS on the matched dataset for adopter vs treatment

lm\_treat1 <- lm(adopter ~ treatment, data = dta\_m)  
summary(lm\_treat1)

##   
## Call:  
## lm(formula = adopter ~ treatment, data = dta\_m)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.17754 -0.17754 -0.08877 -0.08877 0.91123   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.088771 0.003399 26.12 <2e-16 \*\*\*  
## treatment 0.088771 0.004806 18.47 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3368 on 19644 degrees of freedom  
## Multiple R-squared: 0.01707, Adjusted R-squared: 0.01702   
## F-statistic: 341.1 on 1 and 19644 DF, p-value: < 2.2e-16

#The result gives the coefficient of 0.08 for treatment which is same as the t-test performed before.

#Part 4 – Logistic Regression

#Regression analysis - With matched dataset  
lm\_treat2 <- lm(adopter ~ treatment + age + friend\_cnt + avg\_friend\_age + friend\_country\_cnt + songsListened + lovedTracks + posts + playlists + shouts   
+ tenure, family = binomial(), data = dta\_m)

summary(lm\_treat2)

##   
## Call:  
## lm(formula = adopter ~ treatment + age + friend\_cnt + avg\_friend\_age +   
## friend\_country\_cnt + songsListened + lovedTracks + posts +   
## playlists + shouts + tenure, data = dta\_m, family = binomial())  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.28665 -0.15334 -0.10763 -0.06367 0.96372   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.443e-02 1.259e-02 -1.146 0.25168   
## treatment 7.441e-02 4.932e-03 15.088 < 2e-16 \*\*\*  
## age 2.047e-03 4.502e-04 4.547 5.46e-06 \*\*\*  
## friend\_cnt -9.115e-06 3.609e-05 -0.253 0.80062   
## avg\_friend\_age 9.839e-04 5.679e-04 1.732 0.08321 .   
## friend\_country\_cnt 9.139e-04 4.446e-04 2.055 0.03985 \*   
## songsListened 6.474e-07 6.602e-08 9.805 < 2e-16 \*\*\*  
## lovedTracks 8.430e-05 6.006e-06 14.035 < 2e-16 \*\*\*  
## posts 2.883e-05 1.364e-05 2.114 0.03450 \*   
## playlists 7.280e-03 1.412e-03 5.154 2.57e-07 \*\*\*  
## shouts 1.262e-05 4.588e-06 2.751 0.00594 \*\*   
## tenure -3.345e-04 1.323e-04 -2.528 0.01147 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3324 on 19634 degrees of freedom  
## Multiple R-squared: 0.04309, Adjusted R-squared: 0.04255   
## F-statistic: 80.38 on 11 and 19634 DF, p-value: < 2.2e-16

#Regression analysis - With original dataset  
#Removing the variable 'id' and dummy variable 'treatment' which was created for PSM  
high\_note\_reg <- subset(high\_note, select = -c(1,17))  
  
lm\_treat3 <- lm(adopter ~ ., family = binomial(),data = high\_note\_reg)

summary(lm\_treat3)

##   
## Call:  
## lm(formula = adopter ~ ., data = high\_note\_reg, family = binomial())  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.02568 -0.09144 -0.06028 -0.03262 1.00882   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -5.132e-02 6.644e-03 -7.725 1.14e-14 \*\*\*  
## age 1.878e-03 2.783e-04 6.747 1.53e-11 \*\*\*  
## male 2.465e-02 2.693e-03 9.151 < 2e-16 \*\*\*  
## friend\_cnt -4.671e-04 3.964e-05 -11.786 < 2e-16 \*\*\*  
## avg\_friend\_age 2.063e-03 3.519e-04 5.863 4.58e-09 \*\*\*  
## avg\_friend\_male 5.822e-03 4.097e-03 1.421 0.155236   
## friend\_country\_cnt 4.408e-03 3.183e-04 13.847 < 2e-16 \*\*\*  
## subscriber\_friend\_cnt 1.039e-02 7.237e-04 14.350 < 2e-16 \*\*\*  
## songsListened 8.438e-07 4.674e-08 18.055 < 2e-16 \*\*\*  
## lovedTracks 9.655e-05 4.624e-06 20.879 < 2e-16 \*\*\*  
## posts 2.484e-05 1.078e-05 2.305 0.021155 \*   
## playlists 7.426e-03 1.018e-03 7.295 3.04e-13 \*\*\*  
## shouts 1.212e-05 3.629e-06 3.341 0.000836 \*\*\*  
## tenure -4.099e-04 7.047e-05 -5.816 6.07e-09 \*\*\*  
## good\_country -2.691e-02 2.671e-03 -10.076 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2635 on 43812 degrees of freedom  
## Multiple R-squared: 0.06231, Adjusted R-squared: 0.06201   
## F-statistic: 207.9 on 14 and 43812 DF, p-value: < 2.2e-16

#Calculating the exponential of coefficients because log(odds) are difficult to interpret

exp(coef(lm\_treat3))

## (Intercept) age male

## 0.9499703 1.0018797 1.0249519

## friend\_cnt avg\_friend\_age avg\_friend\_male

## 0.9995330 1.0020653 1.0058394

## friend\_country\_cnt subscriber\_friend\_cnt songsListened

## 1.0044176 1.0104395 1.0000008

## lovedTracks posts playlists

## 1.0000966 1.0000248 1.0074541

## shouts tenure good\_country

## 1.0000121 0.9995902 0.9734446

**#Part 5 - Estimating the model and coefficients:**

In the above model, we can state that except for ‘avg\_friend\_male’, all other variables are found to be significant in predicting adopter. The exponential coefficients suggest how much will the probability of adopter increase or decrease with 1-unit change in the variable.

Factors which are leading to an increase in probability of adopter are:

* Male
* Age
* Avg Friend age
* Avg Friend male
* Friend country count
* Subscriber Friend count
* Songs Listened
* Loved Tracks
* Posts
* Playlists
* Shouts

Factors which are leading to decrease in the probability of adopter are:

* Friend count
* Tenure
* Good country

Of all the above variables, we can state that ‘subscriber friend count’ is the most influential factor (exponential coefficient = 1.01) to predict adopter i.e., having more subscriber friends leads to high probability of becoming an adopter and buying the premium subscription.

**Action points for the company to build a ‘free to fee’ strategy**

* Identify people who have more subscriber friends in their community and focus their marketing budget on them. These people are more likely to purchase the premium subscription and stay loyal to the service.
* Identify people who are more active on the platform. These people are happy using High note platform for listening to songs, however are reluctant to pay for the premium subscription. Being active on the platform can be accounted through the following factors:
  + Loved Tracks
  + Posts
  + Playlists
  + Shouts

A specific form of marketing towards these members might lead them to switch to premium subscription over the long run.