Final Project - Mental Health in Tech Survey Data

Indrayani Deshmukh, Snehal Bende, Sindhu Ramaswamy

5/13/2020

1. Project Description

Mental health costs in the US are continuously rising since 2010 and are expected to double by the year 2030. Mental health thus, is of extreem importance.This project is a kaggle dataset titled " Mental health in Tech Survey". This dataset is a survey conducted in 2014 by the Open Sourcing Mental illness (OSMI) to monitor mental health disorders in the Tech industry. OSMI is a non-profit organization and their aim is to help people in the tech industry with mental health disorders so they have a good work life balance.

2.Project Goal

Our idea behind choosing this dataset is identifying the people who need to seek mental health care in the tech industry and, what are the factorsthat are contributing to the increase in mental health problems in the industry? In today's fast paced world there are many reasons for mental health issues and they often result in poor work-life balance. Thus, actions are needed to be taken by companies by providing assistance with mental health care and having a good environment and work life for better performance of their employees. We are also curious to see if the factors like gender, age or employees with family history are more susceptible to having mental health disorder? Our goal for the project is to find the answers to these interesting questions.

3.Steps followed for the project

The project code is done in R and the final report is compiled with R markdown. The detailed steps, data analysis and the reuslts are in the following sections of the report.

a) Data Exploration and summary Statistics

b) Data Munging and Preparation

c) Feature Engineering

d) Modeling

e) Optimization

f) Results and Conclusion

Loading libraries

The following libraries were loaded and packages were installed which were required to perform tasks for the project.

```
#Installing the packages and loading them.
install_load <- function (packages)  {

  for(package in packages){

    # If package is installed
    if(package %in% rownames(installed.packages()))
      do.call('library', list(package))

    # If package is not installed
    else {
      install.packages(package, dependencies = TRUE)
      do.call("library", list(package))
    }
  }
}

# loading the required librarires
libs <- c("ggplot2", "maps","magrittr","plotly", "plyr", "dplyr", "rworldmap","stringr","lubridate", "pl
install_load(libs)

# Loading specific methods from libraries
libs.methods <- c("C50", "lattice", "caret", "nnet", "e1071","Matrix", "foreach","glmnet","C50","randoml
install_load(libs.methods)
```

The data file survey.csv was read to perform further tasks.

```
survey_data <- read.csv("survey.csv")
```

Structure and summary of the survey data

```
str(survey_data)
```

```
## 'data.frame':	1259 obs. of  27 variables:
##  $ Timestamp                : Factor w/ 1246 levels "2014-08-27 11:29:31",..: 1 2 3 4 5 6 7 8 9 10 .
##  $ Age                      : num  37 44 32 31 31 33 35 39 42 23 ...
##  $ Gender                   : Factor w/ 49 levels "A little about you",..: 16 24 30 30 30 30 16 24 16
##  $ Country                  : Factor w/ 48 levels "Australia","Austria",..: 46 46 8 45 46 46 46 8 46
##  $ state                    : Factor w/ 45 levels "AL","AZ","CA",..: 11 12 NA NA 38 37 19 NA 11 NA .
##  $ self_employed            : Factor w/ 2 levels "No","Yes": NA NA NA NA NA NA NA NA NA NA ...
##  $ family_history           : Factor w/ 2 levels "No","Yes": 1 1 1 2 1 2 2 1 2 1 ...
##  $ treatment                : Factor w/ 2 levels "No","Yes": 2 1 1 2 1 1 2 1 2 1 ...
##  $ work_interfere           : Factor w/ 4 levels "Never","Often",..: 2 3 3 2 1 4 4 1 4 1 ...
##  $ no_employees             : Factor w/ 6 levels "1-5","100-500",..: 5 6 5 3 2 5 1 1 2 3 ...
##  $ remote_work              : Factor w/ 2 levels "No","Yes": 1 1 1 1 2 1 2 2 1 1 ...
##  $ tech_company             : Factor w/ 2 levels "No","Yes": 2 1 2 2 2 2 2 2 2 2 ...
##  $ benefits                 : Factor w/ 3 levels "Don't know","No",..: 3 1 2 2 3 3 2 2 3 1 ...
##  $ care_options             : Factor w/ 3 levels "No","Not sure",..: 2 1 1 3 1 2 1 3 3 1 ...
##  $ wellness_program         : Factor w/ 3 levels "Don't know","No",..: 2 1 2 2 1 2 2 2 2 1 ...
##  $ seek_help                : Factor w/ 3 levels "Don't know","No",..: 3 1 2 2 1 1 2 2 2 1 ...
##  $ anonymity                : Factor w/ 3 levels "Don't know","No",..: 3 1 1 2 1 1 2 3 2 1 ...
##  $ leave                    : Factor w/ 5 levels "Don't know","Somewhat difficult",..: 3 1 2 2 1 1 2
##  $ mental_health_consequence: Factor w/ 3 levels "Maybe","No","Yes": 2 1 2 3 2 2 1 2 1 2 ...
##  $ phys_health_consequence  : Factor w/ 3 levels "Maybe","No","Yes": 2 2 2 3 2 2 1 2 2 2 ...
##  $ coworkers                : Factor w/ 3 levels "No","Some of them",..: 2 1 3 2 2 3 2 1 3 3 ...
##  $ supervisor               : Factor w/ 3 levels "No","Some of them",..: 3 1 3 1 3 3 1 1 3 3 ...
##  $ mental_health_interview  : Factor w/ 3 levels "Maybe","No","Yes": 2 2 3 1 3 2 2 2 2 1 ...
##  $ phys_health_interview    : Factor w/ 3 levels "Maybe","No","Yes": 1 2 3 1 3 1 2 2 1 1 ...
##  $ mental_vs_physical       : Factor w/ 3 levels "Don't know","No",..: 3 1 2 2 1 1 1 2 2 3 ...
##  $ obs_consequence          : Factor w/ 2 levels "No","Yes": 1 1 1 2 1 1 1 1 1 1 ...
##  $ comments                 : Factor w/ 160 levels " ","-","(yes but the situation was unusual and in
```

```r
summary(survey_data)# summary of the survey data
```

```
##                Timestamp          Age                 Gender   
##  2014-08-27 12:31:41:   2   Min.   :-1.726e+03   Male   :615  
##  2014-08-27 12:37:50:   2   1st Qu.: 2.700e+01   male   :206  
##  2014-08-27 12:43:28:   2   Median : 3.100e+01   Female :121  
##  2014-08-27 12:44:51:   2   Mean   : 7.943e+07   M      :116  
##  2014-08-27 12:54:11:   2   3rd Qu.: 3.600e+01   female : 62  
##  2014-08-27 14:22:43:   2   Max.   : 1.000e+11   F      : 38  
##  (Other)            :1247                        (Other):101  
##            Country         state      self_employed family_history treatment
##  United States :751   CA     :138   No  :1095     No :767        No :622  
##  United Kingdom:185   WA     : 70   Yes : 146     Yes:492        Yes:637  
##  Canada        : 72   NY     : 57   NA's:  18                            
##  Germany       : 45   TN     : 45                                        
##  Ireland       : 27   TX     : 44                                        
##  Netherlands   : 27   (Other):390                                        
##  (Other)       :152   NA's   :515                                        
##    work_interfere         no_employees remote_work tech_company
##  Never    :213    1-5           :162   No :883     No : 228    
##  Often    :144    100-500       :176   Yes:376     Yes:1031    
##  Rarely   :173    26-100        :289                           
##  Sometimes:465    500-1000      : 60                           
##  NA's     :264    6-25          :290                           
##                   More than 1000:282                           
```

```
##
##        benefits      care_options   wellness_program      seek_help
##  Don't know:408   No      :501   Don't know:188     Don't know:363
##  No        :374   Not sure:314   No        :842     No        :646
##  Yes       :477   Yes     :444   Yes       :229     Yes       :250
##
##
##
##
##       anonymity                  leave     mental_health_consequence
##  Don't know:819   Don't know        :563   Maybe:477
##  No        : 65   Somewhat difficult:126   No   :490
##  Yes       :375   Somewhat easy     :266   Yes  :292
##                   Very difficult    : 98
##                   Very easy         :206
##
##
##  phys_health_consequence        coworkers          supervisor
##  Maybe:273               No          :260   No          :393
##  No   :925               Some of them:774   Some of them:350
##  Yes  : 61               Yes         :225   Yes         :516
##
##
##
##
##  mental_health_interview phys_health_interview  mental_vs_physical
##  Maybe: 207              Maybe:557             Don't know:576
##  No   :1008              No   :500             No        :340
##  Yes  :  44              Yes  :202             Yes       :343
##
##
##
##
##  obs_consequence
##  No :1075
##  Yes: 184
##
##
##
##
##
##
##  * Small family business - YMMV.
##
##  -
##  (yes but the situation was unusual and involved a change in leadership at a very high level in the o
##  A close family member of mine struggles with mental health so I try not to stigmatize it. My employe
##  (Other)
##  NA's
```

```
dim(survey_data) #dimension of the survey data
```

```
## [1] 1259   27
```

As we see there are total of 1259 observations and 27 columns related to mental health questions and the demographic information in the dataset. As we are interested to see the results in the tech industry we might not need to use all the variables.From the results we see that there are 1095 missing values and the variables age, state and self_employed have NA values. Gender variable has duplicate values.

library(dplyr)

Selection of Target and predictor Variables

Based on the project goal, we selected 'treatment' as the target variable. Treatment variable tells us if the interviewed employee have sought mental health treatment or not. It is categorical in nature. 'Tech company'(binary variable) another variable which is considered to be a predictor variable tells if the company is a tech or a non tech company. Gender (categorical variable) - predictor variable tells us if the interviewed employee is male or a female. Age(continuous variable) - predictor variable tells the age of the employee. Family history (binary variable) - predicotr variable, tells if a person has a family history of mental health disorder. no_employees (categorical varible) - predictor variable, tells about the totol number of employees in the company.The new data comprising the targeta nd predicotr variables includes 1259 rows and 6 columns.

```
survey <- survey_data %>% select(treatment, Age, Gender, family_history, no_employees, tech_company)
# checking structure of data including selected variables
str(survey)
```

```
## 'data.frame':    1259 obs. of  6 variables:
##  $ treatment     : Factor w/ 2 levels "No","Yes": 2 1 1 2 1 1 2 1 2 1 ...
##  $ Age           : num  37 44 32 31 31 33 35 39 42 23 ...
##  $ Gender        : Factor w/ 49 levels "A little about you",..: 16 24 30 30 30 30 16 24 16 30 ...
##  $ family_history: Factor w/ 2 levels "No","Yes": 1 1 1 2 1 2 2 1 2 1 ...
##  $ no_employees  : Factor w/ 6 levels "1-5","100-500",..: 5 6 5 3 2 5 1 1 2 3 ...
##  $ tech_company  : Factor w/ 2 levels "No","Yes": 2 1 2 2 2 2 2 2 2 2 ...
```

Full Logistic Regression Model

We built the logistic regression model considereing 80% of the data to review our selection for target and predictor variables. Though we are aware, some variables in the logistic regression model below are not related to our project goal, we wanted to see the p values and re think on our variable selection approach.

```
#Logistic Regression model considereing 80% of the variables
lm1 <- glm(treatment ~ Age + Gender + Country + state + self_employed + family_history + work_interfere
```

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

```
summary(lm1)
```

```
##
## Call:
## glm(formula = treatment ~ Age + Gender + Country + state + self_employed +
##     family_history + work_interfere + no_employees + remote_work +
##     tech_company + benefits + care_options + leave + mental_vs_physical +
##     coworkers + seek_help, family = "binomial", data = survey_data)
```

```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.9311  -0.2528   0.1980   0.5235   2.9153
##
## Coefficients: (1 not defined because of singularities)
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  13.63472 6522.63952   0.002  0.99833
## Age                           0.01735    0.01925   0.901  0.36739
## GenderCis Female              1.45073 9224.40440   0.000  0.99987
## Gendercis male              -30.88367 9224.40445  -0.003  0.99733
## GenderCis Male               -0.05009 7723.62498   0.000  0.99999
## Gendercis-female/femme       -0.42559 9224.40445   0.000  0.99996
## Genderf                     -17.16651 6522.63930  -0.003  0.99790
## GenderF                     -14.01133 6522.63930  -0.002  0.99829
## Genderfemail                -30.45694 9224.40456  -0.003  0.99737
## Genderfemale                -14.59716 6522.63926  -0.002  0.99821
## GenderFemale                -15.59578 6522.63925  -0.002  0.99809
## GenderFemale                -15.67163 6522.63950  -0.002  0.99808
## GenderFemale (cis)          -39.25535 9224.40449  -0.004  0.99660
## GenderFemale (trans)         -1.34955 7642.99055   0.000  0.99986
## GenderGenderqueer           -35.16751 9224.40452  -0.004  0.99696
## Genderm                     -14.32608 6522.63932  -0.002  0.99825
## GenderM                     -15.32047 6522.63925  -0.002  0.99813
## GenderMail                  -33.92378 9224.40448  -0.004  0.99707
## GenderMake                  -15.97251 6522.63943  -0.002  0.99805
## Gendermale                  -16.60765 6522.63924  -0.003  0.99797
## GenderMale                  -16.16671 6522.63923  -0.002  0.99802
## GenderMale                   -0.33331 9224.40453   0.000  0.99997
## GenderMale-ish               -1.24308 9224.40447   0.000  0.99989
## Gendermsle                  -35.69347 9224.40452  -0.004  0.99691
## GenderNah                    -0.20668 9224.40443   0.000  0.99998
## Gendernon-binary             -1.31104 9224.40444   0.000  0.99989
## Genderp                      -1.89636 9224.40455   0.000  0.99984
## Genderqueer/she/they          1.01957 9224.40442   0.000  0.99991
## GenderTrans woman             0.33640 9224.40451   0.000  0.99997
## GenderTrans-female          -36.11677 9224.40449  -0.004  0.99688
## Genderwoman                  -0.47206 9224.40450   0.000  0.99996
## GenderWoman                   0.46348 9224.40447   0.000  0.99996
## CountryBulgaria              16.22962 6522.63877   0.002  0.99801
## CountryIsrael               -17.82985 6522.63881  -0.003  0.99782
## CountryUnited States               NA         NA      NA       NA
## stateAZ                      13.73299 2466.16770   0.006  0.99556
## stateCA                      -1.12556    1.91415  -0.588  0.55652
## stateCO                      -0.77183    2.19176  -0.352  0.72473
## stateCT                      16.77403 3983.77781   0.004  0.99664
## stateDC                      -3.30085    3.02397  -1.092  0.27502
## stateFL                      -0.77862    2.25308  -0.346  0.72966
## stateGA                      -1.40053    2.06341  -0.679  0.49730
## stateIA                      18.97153 2814.33202   0.007  0.99462
## stateID                      12.56215 6522.63897   0.002  0.99846
## stateIL                      -2.05252    2.00624  -1.023  0.30627
## stateIN                      -1.20481    2.03709  -0.591  0.55423
## stateKS                     -17.20523 4362.22004  -0.004  0.99685
```

```
## stateKY                    -2.96117    2.46015  -1.204  0.22872
## stateLA                    14.39330 6522.63893   0.002  0.99824
## stateMA                    -1.99307    2.08526  -0.956  0.33918
## stateMD                    -2.57294    2.31102  -1.113  0.26557
## stateME                    16.21379 6522.63892   0.002  0.99802
## stateMI                    -2.48460    1.97974  -1.255  0.20948
## stateMN                    -0.52173    2.09552  -0.249  0.80338
## stateMO                    -2.34524    2.24090  -1.047  0.29530
## stateMS                    17.54374 6522.63894   0.003  0.99785
## stateNC                    -2.97774    2.05458  -1.449  0.14725
## stateNE                    -1.01610    3.31750  -0.306  0.75939
## stateNH                    -1.42537    2.39316  -0.596  0.55144
## stateNJ                    -2.41625    2.32953  -1.037  0.29963
## stateNV                    19.36500 4011.18209   0.005  0.99615
## stateNY                    -1.85905    1.97318  -0.942  0.34611
## stateOH                    -1.39305    1.97489  -0.705  0.48057
## stateOK                    -1.72430    2.35459  -0.732  0.46398
## stateOR                    -1.33901    1.99011  -0.673  0.50105
## statePA                    -3.19282    1.98459  -1.609  0.10766
## stateSC                    -0.38563    4.24629  -0.091  0.92764
## stateSD                    -3.16887    2.45747  -1.289  0.19723
## stateTN                    -2.79183    1.98465  -1.407  0.15951
## stateTX                    -1.36600    1.95130  -0.700  0.48390
## stateUT                    -1.75363    2.14135  -0.819  0.41282
## stateVA                    -1.02665    2.10939  -0.487  0.62647
## stateVT                   -17.17166 6522.63892  -0.003  0.99790
## stateWA                    -2.07018    1.93737  -1.069  0.28527
## stateWI                    -2.19468    2.09417  -1.048  0.29464
## stateWV                   -20.64340 6522.63892  -0.003  0.99747
## stateWY                    -2.74447    2.63801  -1.040  0.29817
## self_employedYes           -0.81888    0.64636  -1.267  0.20519
## family_historyYes           1.29549    0.30422   4.258 2.06e-05 ***
## work_interfereOften         5.30379    0.68481   7.745 9.56e-15 ***
## work_interfereRarely        3.99329    0.52603   7.591 3.16e-14 ***
## work_interfereSometimes     4.13910    0.47539   8.707  < 2e-16 ***
## no_employees100-500         0.38940    0.70072   0.556  0.57841
## no_employees26-100          1.04564    0.68554   1.525  0.12719
## no_employees500-1000        2.11384    1.06248   1.990  0.04664 *
## no_employees6-25            0.27100    0.61371   0.442  0.65880
## no_employeesMore than 1000 -0.25742    0.68827  -0.374  0.70840
## remote_workYes             -0.33697    0.35244  -0.956  0.33901
## tech_companyYes            -0.16051    0.41354  -0.388  0.69792
## benefitsNo                  0.78924    0.54506   1.448  0.14762
## benefitsYes                 1.13519    0.40603   2.796  0.00518 **
## care_optionsNot sure       -0.51135    0.36403  -1.405  0.16011
## care_optionsYes             1.14839    0.40193   2.857  0.00427 **
## leaveSomewhat difficult     0.50880    0.53607   0.949  0.34255
## leaveSomewhat easy          0.04277    0.39406   0.109  0.91357
## leaveVery difficult         0.89472    0.61253   1.461  0.14410
## leaveVery easy              0.13732    0.48006   0.286  0.77484
## mental_vs_physicalNo       -0.13935    0.38006  -0.367  0.71387
## mental_vs_physicalYes       0.13820    0.39902   0.346  0.72908
## coworkersSome of them      -0.50426    0.36201  -1.393  0.16363
## coworkersYes                0.17410    0.53545   0.325  0.74507
```

```
## seek_helpNo                   -0.89249    0.39989  -2.232  0.02562 *
## seek_helpYes                  -1.17107    0.45235  -2.589  0.00963 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 745.24  on 590  degrees of freedom
## Residual deviance: 374.94  on 489  degrees of freedom
##   (668 observations deleted due to missingness)
## AIC: 578.94
##
## Number of Fisher Scoring iterations: 17
```

We got an AIC score of 578.94 and significant variables with lowest p values to be family_history, work_interefere, no_employees, care_options and seek_help. We conclude that care_options , work_interefere and seek_help variables are not closely tied to our project goal , hence we move forward with our selection for predictor variables.

Treatment, Family history and tech company

Inspecting Variables in detail

```
#Studying each variable in detail
table(survey$treatment)
```

```
##
##  No Yes
## 622 637
```

```
table(survey$family_history)
```

```
##
##  No Yes
## 767 492
```

```
table(survey$tech_company)
```

```
##
##   No  Yes
##  228 1031
```

Treatment, family history and tech company are all binary variables with no missing data or outliers. Out of the total interviewed people in the survey, 637 have sought mental health treatment and 622 have not. Thus we see that more than 50% of the people in the survey data have sought mental health treatment.Family history is also a binary variable with no missing data. The output shows more than 60% of the people in the survey do not have a family history of having a mental disorder.The survey comprised of 1031 tech companies and 228 non tech companies.

Number of Employees

Number of employees is not an ordered variable.We order this variable which will help us visualize if there is a trend in the company size and the employees seeking mental health treatment.We are interested to see if the size of the company has a direct relation to seeking mental health care. As there is a general notion that employees working in a startup have more responsibilites and work pressure than the ones working in a large company.

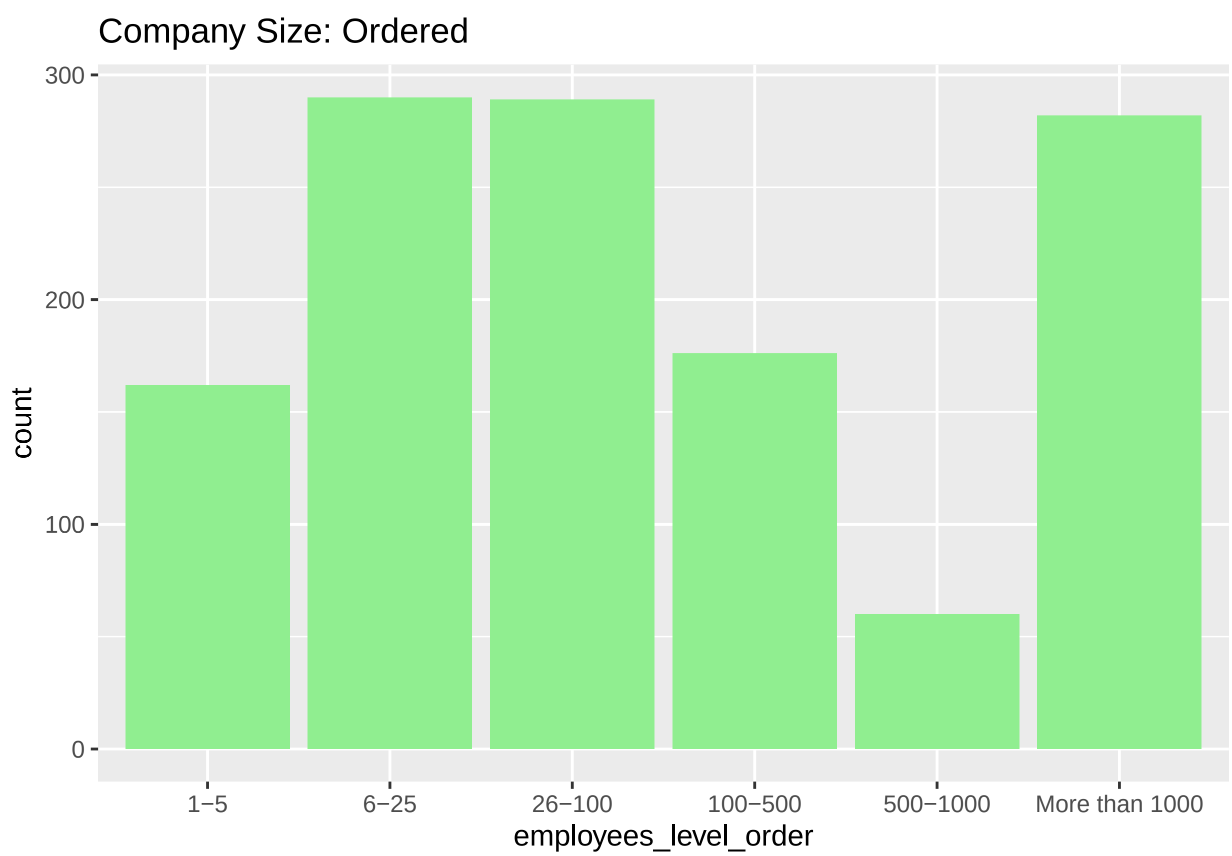
```
#no_employees is not an ordered variable.
#Ordering the no_employees variable
summary(survey$no_employees)
```

```
##            1-5        100-500         26-100       500-1000           6-25
##            162            176            289             60            290
## More than 1000
##            282
```

```
employees_level_order <- factor(survey$no_employees, levels = c("1-5","6-25","26-100","100-500", "500-1(
```

There are 6 groups in the no_employees category.

```
# Company distribution graph
survey %>% ggplot(aes(x=employees_level_order))+
  geom_bar(fill = "lightgreen") + ggtitle("Company Size: Ordered")
```



The above graph tells us that there is no specific trend with the size of the comapny and the employees seeking treatment. However, we can say that small companies and large companies with more than 1000 employees do have more people seeking mentak health treatment.Companies sized 6-25 and 26-100 have similar rate of seeking the treatment.

GENDER

To further explore and inspect the selected variables, we are interested to see if gender has somthing to do with seeking the treatment. Are female employees more in number to seek the treatment than male?

Gender variable has outliers and includes values like 'Malr','m','F','Genderqueer' and so on. So we decide to categories the gender variable in three categories as "Male", "Female" and "Queer".We first list the three categories in the code and then categories the values in the list to their respective categories.

```
# Listing categories

Male <- c("Male ","Cis Man", "Malr", "Male", "male", "M", "m", "Male-ish", "maile", "Mal", "Male (CIS)"
Female <- c("Female ","femail","Female (cis)","female","Female","F","Woman","f","Femake","woman","Femal
Queer <-c ("ostensibly male, unsure what that really means","p","A little about you","queer","Neuter","
```

```
#categorizing gender variable
survey$Gender <- sapply(
  as.vector(survey$Gender),
  function(x) if(x %in% Male) "Male" else x )

survey$Gender <- sapply(
  as.vector(survey$Gender),
  function(x) if(x %in% Female) "Female" else x )

survey$Gender <- sapply(
  as.vector(survey$Gender),
  function(x) if(x %in% Queer) "Queer" else x )
survey$Gender <- as.factor(survey$Gender)
```

Let's view the results and the number of employees in each of the categories

```
# Records in each category
table(survey$Gender)
```

```
##
## Female   Male  Queer
##    251    991     17
```

```
table(survey$Gender)/length(survey$Gender) #studying the relative frequency of the gender variable
```

```
##
##     Female       Male      Queer
## 0.19936458 0.78713264 0.01350278
```

We see that there is significantly a number of Male population in the survey. This is obvious as number of male employees in tech companies are more as compared to female employees. There is a low number in 'Queer' population.We also wanted to fetch the results for the relative frequency of the population of females, males and queer. We observe that male consi=titutes a large number in the survey.

```r
#library(ggplot2)
# Visualize the number of subjects in each gender type
#Gender_df <- ggplot(gender_diversity, aes(x = Gender, y = count, fill = Gender)) +
 # geom_bar(stat = "identity", alpha = 0.5) +
  #xlab("Gender Diversity") +
  #ylab("Number of People") +
  #ggtitle("Gender Diversity in Tech Survey")

#Gender_df
```

The above graph is a result of categorizing the gender variable and helps us to visualize the ratio of female, male and queer in the data.

AGE

Age variable has outliers as previously seen in the summary. It has negative values along with extreemly high values. We need to handle the outliers in the age variable and replace it with the median values to keep the data integrity. In the following code, we replaced the outliers with the median values.The summary of the transformed variable is shown in the output

```r
# Age Variable --Outlier management: replacing with median value
survey$Age[which(survey$Age<0)]<- median(survey$Age)
survey$Age[which(survey$Age>100)]<- median(survey$Age)
Age_one <- survey$Age
Age_one
```

```
##    [1] 37 44 32 31 31 33 35 39 42 23 31 29 42 36 27 29 23 32 46 36 29 31 46 41
##   [25] 33 35 33 35 34 37 32 31 30 42 40 27 29 38 50 35 24 35 27 18 30 38 28 34
##   [49] 26 30 22 33 31 32 28 27 32 24 26 33 44 26 27 26 35 40 23 36 31 34 28 34
##   [73] 23 38 33 19 25 31 32 28 38 23 30 27 33 31 39 34 29 32 31 40 34 18 25 29
##   [97] 24 31 33 30 26 44 25 33 29 35 35 28 34 32 22 28 45 32 28 26 21 27 18 35
##  [121] 29 25 33 36 27 27 27 32 31 19 33 32 27 38 24 39 28 39 29 22 38 37 35 31
##  [145] 30 37 24 23 30 29 19 32 28 36 37 25 27 26 27 25 36 25 31 26 33 27 34 42
##  [169] 23 24 26 31 22 23 34 31 28 32 45 33 29 26 28 45 43 37 24 26 23 35 38 28
##  [193] 28 35 32 31 35 26 27 28 27 34 41 37 34 32 21 30 24 26 40 37 26 32 32 27
##  [217] 30 31 29 41 34 33 28 28 23 24 32 34 24 26 36 41 38 38 30 25 37 34 37 28
##  [241] 22 34 33 25 27 40 21 29 32 29 23 28 31 27 24 29 23 42 24 25 27 27 30 29
##  [265] 43 32 41 32 37 32 30 23 30 34 38 33 34 28 28 23 22 27 18 35 25 27 26 18
##  [289] 38 26 30 35 45 32 56 24 30 60 33 37 23 31 26 28 37 26 30 26 25 27 25 35
##  [313] 36 26 27 30 29 25 22 29 41 29 32 24 25 25 30 25 30 33 24 25 31 45 29 46
##  [337] 30 29 24 29 35 33 27 36 33 25 23 54 22 25 29 27 30 26 25 31 33 34 34 29
##  [361] 33 34 26 32 31 28 35 36 21 21 22 41 55 32 21 45 27 25 34 26 41 27 31 25
##  [385] 26 27 42 29 25 33 31 40 31 26 24 29 48 35 32 29 26 28 23 35 29 26 33 33
##  [409] 22 30 33 31 21 31 26 30 30 23 34 55 28 26 28 32 28 21 24 26 23 24 28 24
##  [433] 33 34 27 28 26 20 23 29 26 36 41 33 23 39 34 26 24 37 43 40 30 34 27 36
##  [457] 27 35 32 37 29 33 28 26 27 38 57 28 26 42 31 58 29 39 34 57 27 23 18 30
##  [481] 23 43 18 29 48 43 28 30 26 33 31 30 27 24 25 23 36 25 54 34 38 40 32 25
##  [505] 35 46 42 32 47 22 33 25 29 39 38 43 46 38 33 34 62 23 35 25 36 41 24 51
##  [529] 29 31 27 31 27 23 21 27 39 26 27 22 26 31 32 28 28 23 30 36 21 30 25 32
##  [553] 29 21 27 32 34 33 22 24 65 27 33 36 40 28 39 32 31 38 23 42 27 26 50 37
##  [577] 23 33 29 34 41 50 29 35 27 40 27 29 31 43 34 29 19 41 29 23 24 31 43 31
```

```
##  [601] 29 35 33 30 27 32 50 24 27 27 32 42 37 30 29 30 35 35 38 22 24 22 31 23
##  [625] 31 28 37 34 32 28 24 56 31 34 35 28 36 30 35 49 36 35 29 57 31 37 25 30
##  [649] 26 22 39 29 54 34 32 25 29 32 30 31 20 27 32 26 30 30 22 24 26 43 26 23
##  [673] 26 26 35 28 22 29 29 45 33 38 19 29 21 23 33 49 28 27 23 29 30 28 32 32
##  [697] 37 39 31 29 30 33 37 23 43 32 26 32 37 29 34 27 30 29 32 31 25 37 29 27
##  [721] 33 30 29 25 33 31 21 30 29 43 37 24 29 31  5 33 43 33 27 36 37 32 39 31
##  [745] 36 30 28 32 35 19 33 42 37 40 36 29 38 26 34 21 31 37 37 38 27 39 33 27
##  [769] 36 28 39 33 32 28 37 39 43 32 27 31 43 33 34 33 25 25 32 25 37 39 29 33
##  [793] 37 35 22 38 32 28 27 35 29 23 39 30 32 28 40 36 27 41 29 29 35 28 36 39
##  [817] 39 44 26 35 40 35 38 34 43 48 20 40 29 35 29 40 29 29 34 44 24 47 43 36
##  [841] 43 36 31 35 33 37 34 36 40 40 42 23 21 26 31 25 51 24 33 32 32 26 23 33
##  [865] 46 34 35 39 32 43 56 32 41 39 37 30 31 29 23 31 29 30 37 36 35 41 31 38
##  [889] 26 39 42 32 29 30 40 51 33 34 50 24 25 43 25 24 51 49 30 25 36 48 48 53
##  [913] 24 33 25 30 30 34 31 22 28 35 28 42 33 29 43 29 25 31 35 34 43 38 26 38
##  [937] 42 33 32 44 28 40 31 32 28 39 45 43 35 40 34 24 61 36 38 33 30 34 26 33
##  [961] 32 25 35 24 55 33 26 25 45 33 43 30 40 49 29 26 38 27 26 28 40 37 34 28
##  [985] 27 29 39 28 23  8 38 19 30 28 20 35 39 31 32 27 25 42 34 26 35 34 38 34
## [1009] 39 44 40 33 24 38 31 23 26 46 30 25 19 30 32 32 37 42 25 19 40 34 26 31
## [1033] 40 31 36 35 26 44 34 35 28 33 40 26 29 26 33 28 41 39 26 23 35 36 42 39
## [1057] 27 33 31 28 29 27 44 25 24 25 34 26 48 34 39 43 41 25 31 40 43 27 37 32
## [1081] 25 29 30 34 32 37 41 38 32 28 11 43 32 25 37 36 24 40 29 43 29 26 33 35
## [1105] 45 25 50 26 33 30 33 29 37 25 40 24 40 46 38 34 32 44 33 45 35 26 20 31
## [1129] 37 28 42 32 36 27 27 27 25 41 23 21 26 29 28 27 23 26 38 39 35 32 32 26
## [1153] 38 34 39 32 37 31 30 51 29 31 31 26 46 32 29 34 26 32 29 30 40 23 20 38
## [1177] 26 29 40 25 32 38 72 35 28 27 56 38 31 40 44 34 37 38 27 34 35 34 32 25
## [1201] 28 28 31 24 34 32 34 23 33 29 24 45 34 31 33 28 27 42 28 38 46 46 41 23
## [1225] 24 23 39 32 25 39 23 24 25 23 24 23 60 28 28 30 31 31 28 43 32 22 32 36
## [1249] 41 30 30 36 29 36 26 32 34 46 25
```

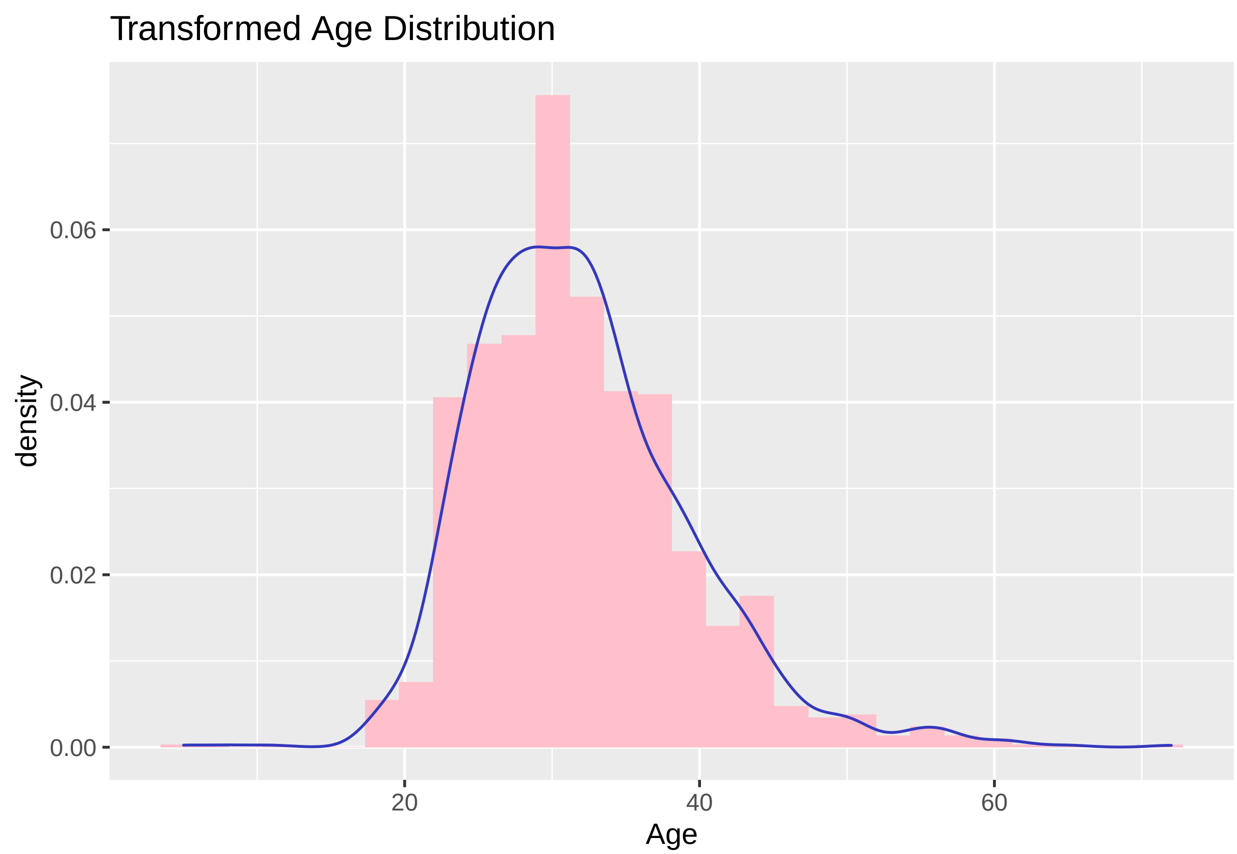
```r
# Summary Age
summary(survey$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    5.00   27.00   31.00   32.02   36.00   72.00
```

Plotting the histogram to see if the distribution of the transformed age variable.

```r
g2 <- ggplot(survey,aes(x=Age))+geom_histogram(aes(y=..density..), fill="pink")+geom_density(col="#3438
g2
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



The histogram shows that the outliers are taken care of and that the data is without anomalies.

After handing the outliers, we can categorize the age data so it will be great to understand the age group of people seeking mental health care. We categorize the age in four groups namely, 'Fresh' including the age group of - 0 to 16, 'Junior' in the age group of 17 to 34, 'Senior' in the age group of 35 to 60 and 'Super' in the age group of 61 to 70.

```
# Age variable categorization
survey$Age<-cut(survey$Age, breaks = c(0, 16, 34, 60, 75), labels = c('Fresh', 'Junior', 'Senior', 'Supe
table(survey$Age)
```

```
##
##  Fresh Junior Senior  Super
##      3    868    384      4
```

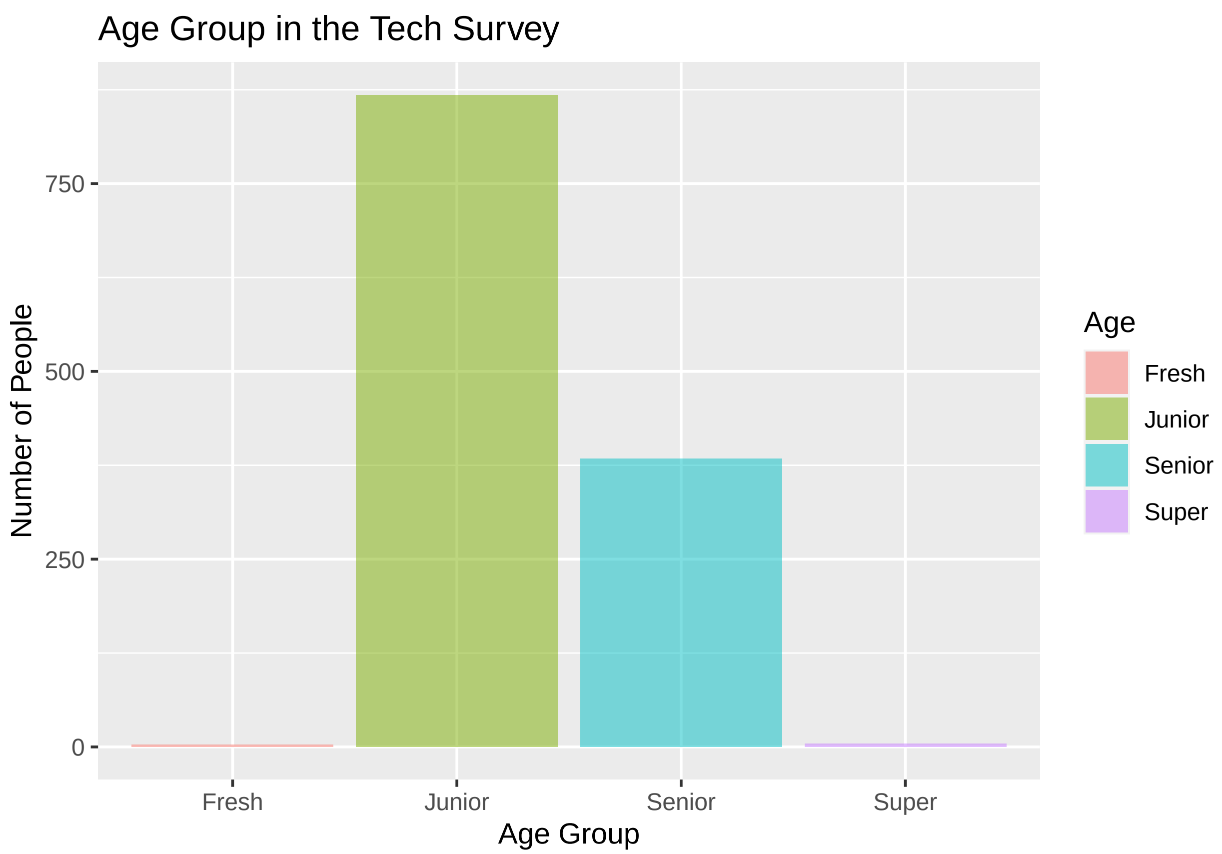
The code below is for grouping the data in each age category.

```
# Group by Age Group and count each group
age_group <- survey %>%
  group_by(Age) %>%
  dplyr::summarize(count = n())
age_group
```

```
## # A tibble: 4 x 2
##   Age    count
##   <fct>  <int>
```

```
## 1 Fresh      3
## 2 Junior   868
## 3 Senior   384
## 4 Super      4
```

```
g3 <- ggplot(age_group, aes(x = Age, y = count, fill = Age)) +
  geom_bar(stat = "identity", alpha = 0.5) +
  xlab("Age Group") +
  ylab("Number of People") +
  ggtitle("Age Group in the Tech Survey")

g3
```



The above graph shows that the large number of employees in the survey data are from the age category Junior and Senior.It is relatable as people in the age group of 0 to 16 hardly work in the tech companies as they are still in the process of education and people above the age of 61 generally retire from the tech comapnies.

Let's look at the final data frame we have.

```
summary(survey)
```

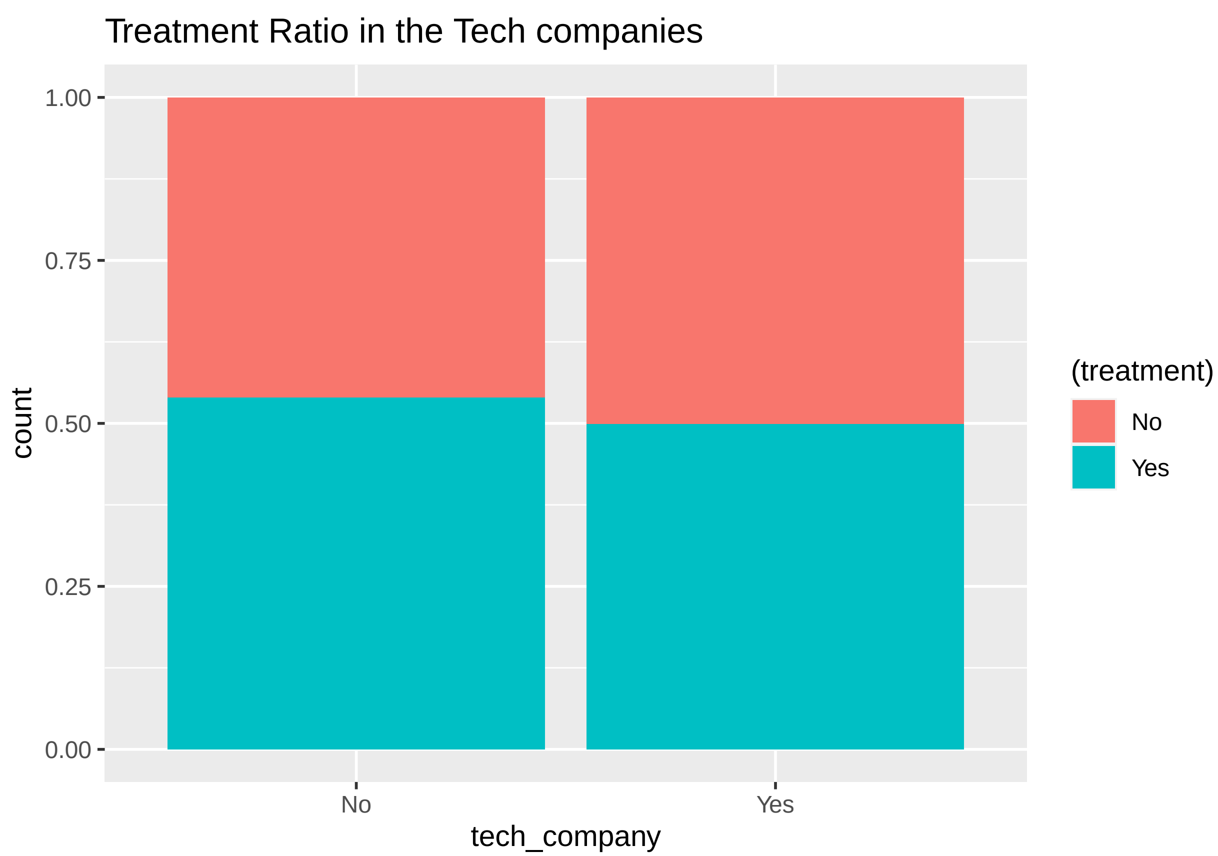
```
##   treatment      Age          Gender     family_history         no_employees
##   No :622    Fresh :  3   Female:251   No :767        1-5            :162
##   Yes:637    Junior:868   Male  :991   Yes:492        100-500        :176
##              Senior:384   Queer : 17                  26-100         :289
```

```
##           Super :  4                                        500-1000      : 60
##                                                             6-25          :290
##                                                             More than 1000:282
##  tech_company
##  No : 228
##  Yes:1031
##
##
##
##
```

Studying the relationships between target and predictor variables

We want to have a closer look at the variables we selected as predictors and see if they have a strong relationship with our target variable.The following graph shows the relation between tech_company variable and the treatment variable.

```
# treatment ratio for tech companies
survey %>% ggplot(aes(x=tech_company, fill = (treatment))) +
  geom_bar(position = "fill") + ggtitle("Treatment Ratio in the Tech companies")
```



The above graph shows that there is a strong realtion between treatment and tech comapny variable. This also shows that even in the non tech comapny, the ratio of seeking mental health care is similar to that of the tech comapny.

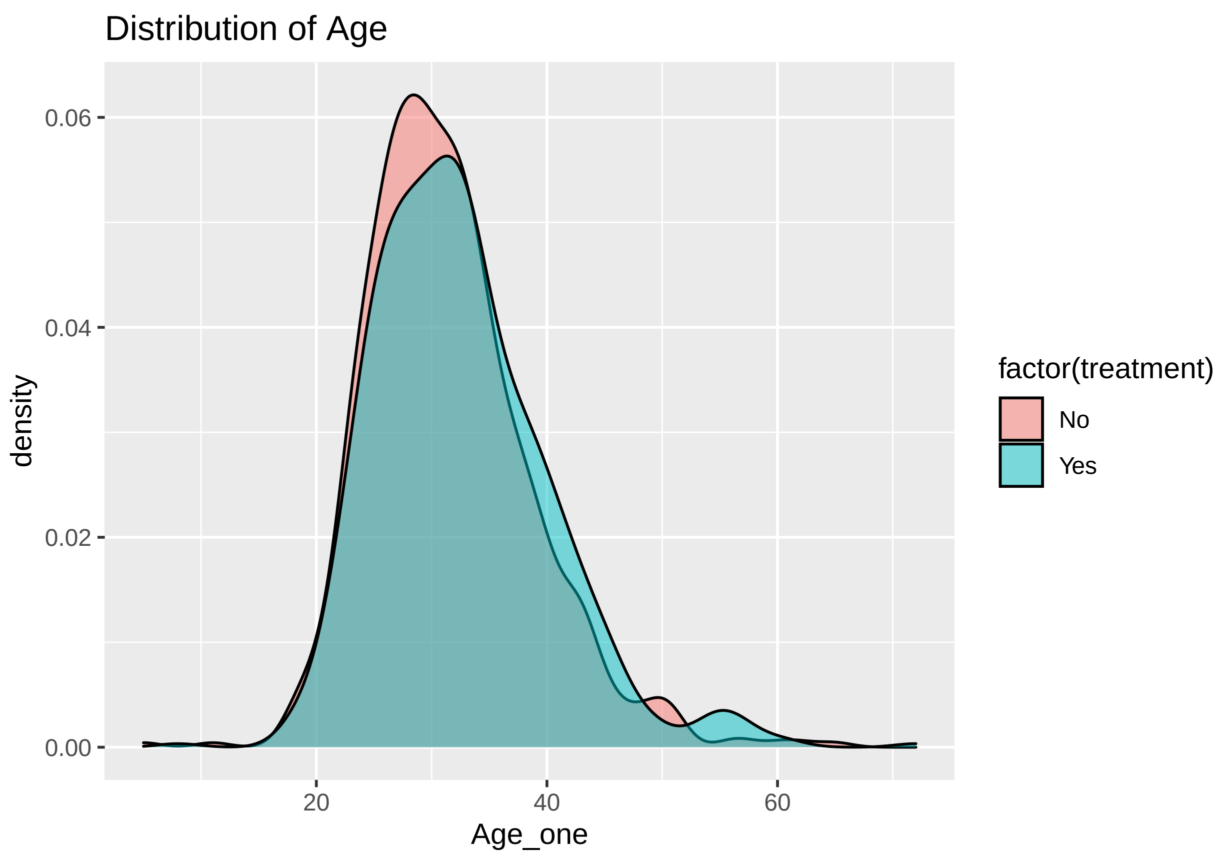
As our focus is on tech comapnies, we will filter the data to include results for only the tech companies.

```r
# Focusing only on the data related to the Tech company
Tech <- survey %>% select(treatment, Age, Gender, family_history, no_employees, tech_company) %>% filte
summary(Tech)
```

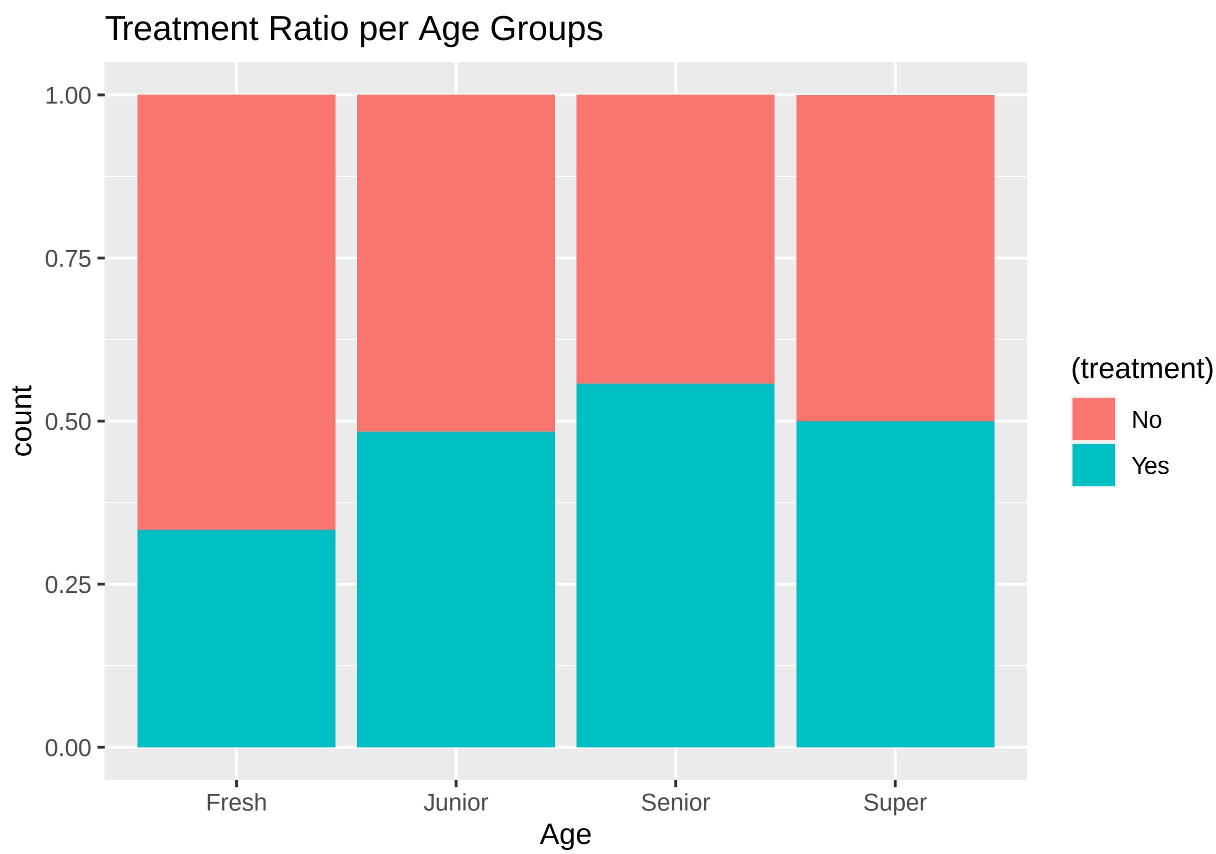
```
##  treatment     Age          Gender    family_history         no_employees
##  No :517   Fresh :  3   Female:190   No :639        1-5           :152
##  Yes:514   Junior:731   Male  :827   Yes:392        100-500       :136
##            Senior:295   Queer : 14                  26-100        :242
##            Super :  2                               500-1000      : 41
##                                                     6-25          :268
##                                                     More than 1000:192
##  tech_company
##  No :   0
##  Yes:1031
##
##
##
##
```

The following graphs show the results of treatment in each of the age groups focused only on tech industry.This will give a clear picture of more susceptible age groups seeking mental health care in the tech industry.

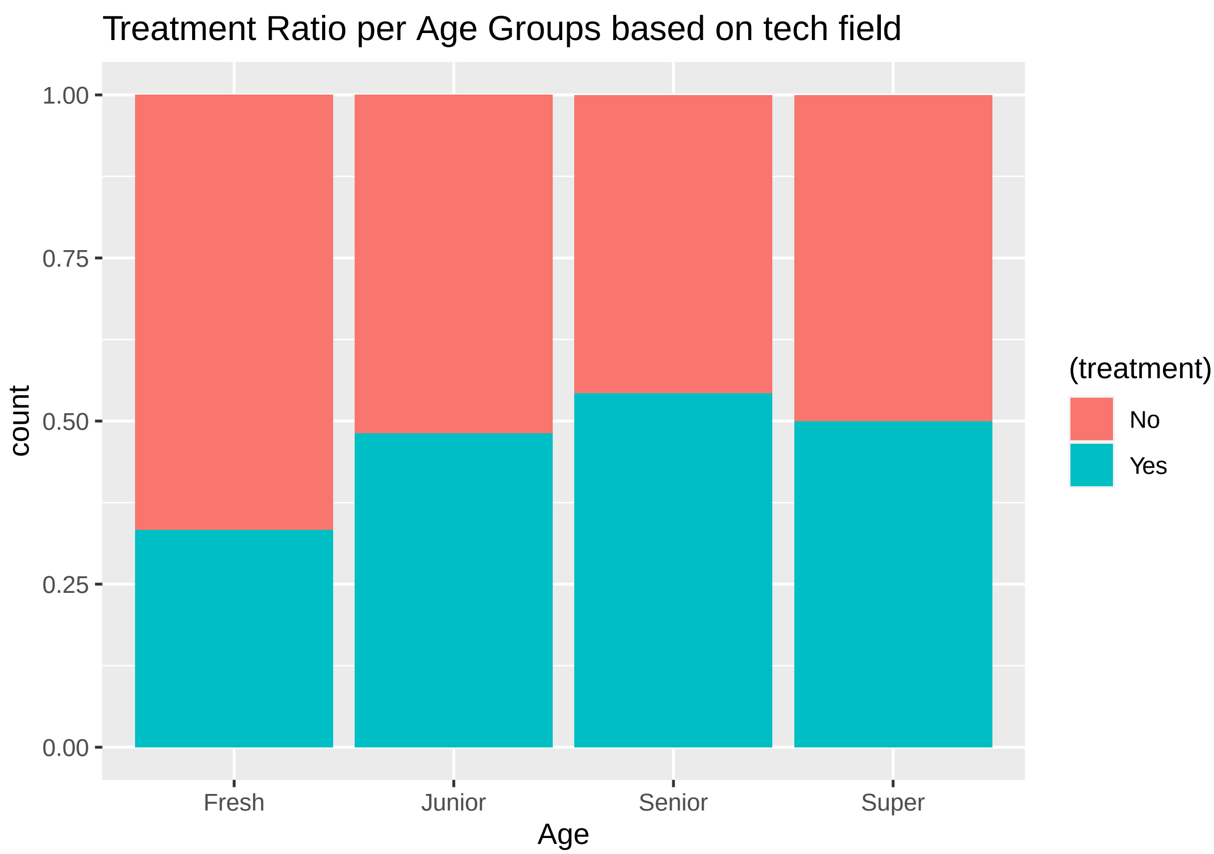
```r
# Age Distribution graph
Age_1 <- survey %>% ggplot(aes(x=Age_one, fill = factor(treatment))) +
  geom_density(alpha = 0.5) + ggtitle("Distribution of Age")
Age_1
```




```
# Comparing treatment ratio in Age groups
Age_2 <- survey %>% ggplot(aes(x=Age, fill = (treatment))) +
  geom_bar(position = "fill") + ggtitle("Treatment Ratio per Age Groups")
Age_2
```



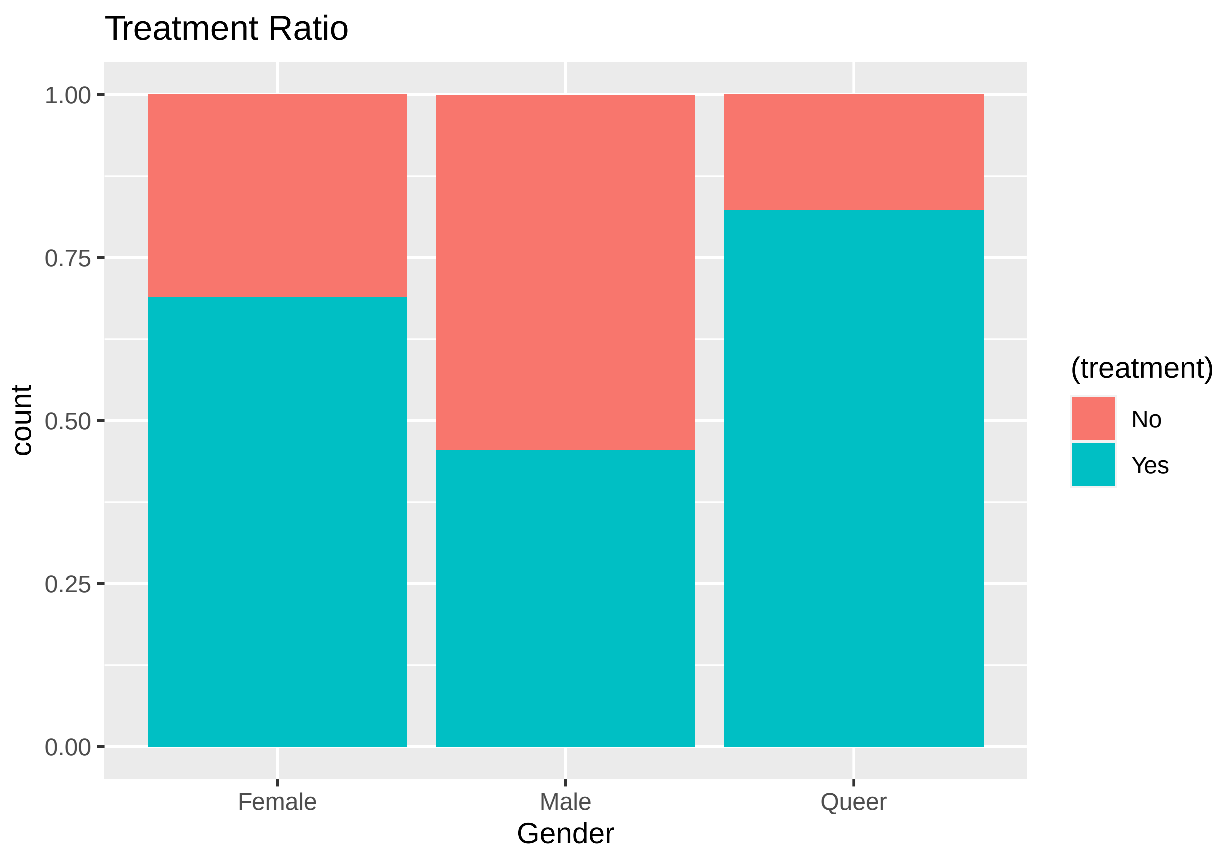

```r
# Comparing treatment ratio in Age groups focusing on tech field
Age_3 <- Tech %>% ggplot(aes(x=Age, fill = (treatment))) +
  geom_bar(position = "fill") + ggtitle("Treatment Ratio per Age Groups based on tech field")
Age_3
```



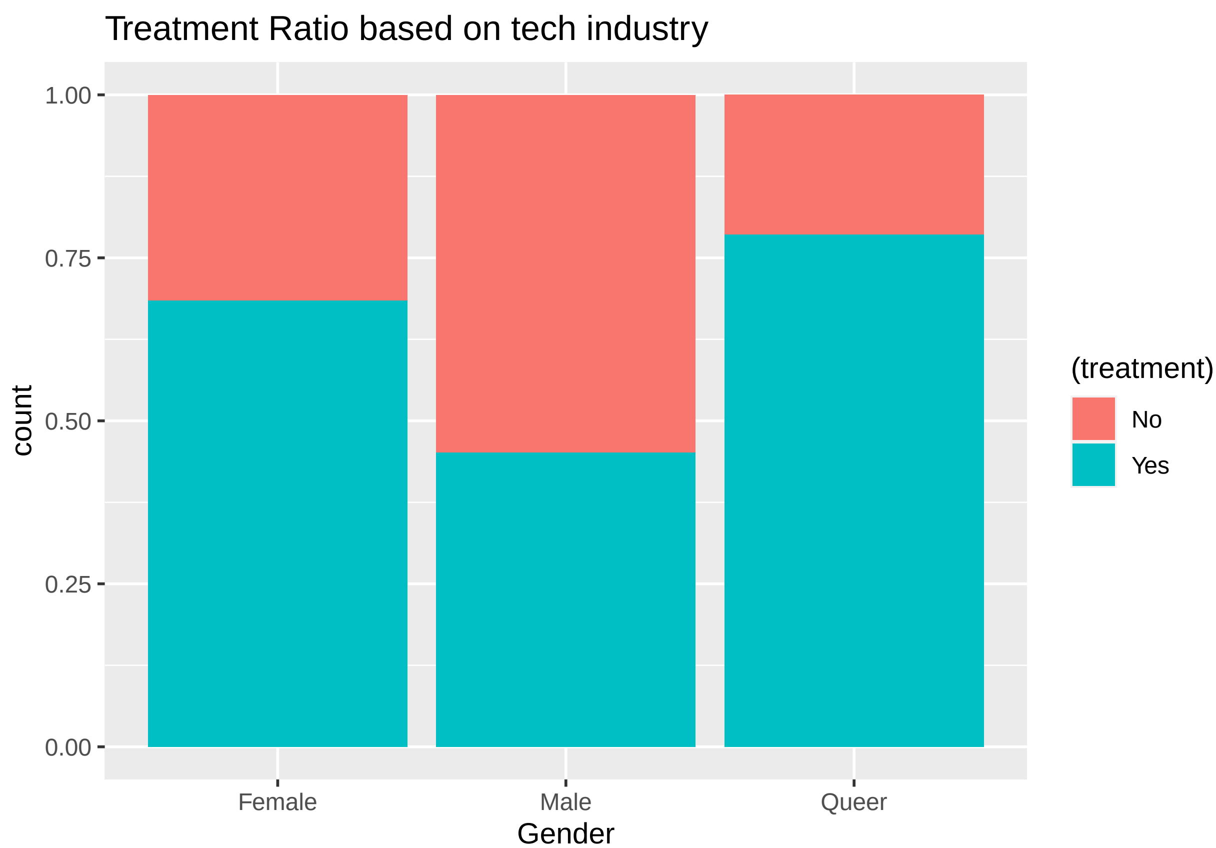
We get that the junior and the senior are the two groups in the tech industry who often seek mental health care treatment.

In the process of studying the relationship between the target and the predictor variable, we do observe a strong relationship in the variables till now. Let's explore the realtionship between the gender variable and the treatment (target variable). We do that by plotting the below graphs.

```
# Comparing treatment ratio in Gender groups
g1 <- survey %>% ggplot(aes(x=Gender, fill = (treatment))) +
  geom_bar(position = "fill") + ggtitle("Treatment Ratio")
g1
```



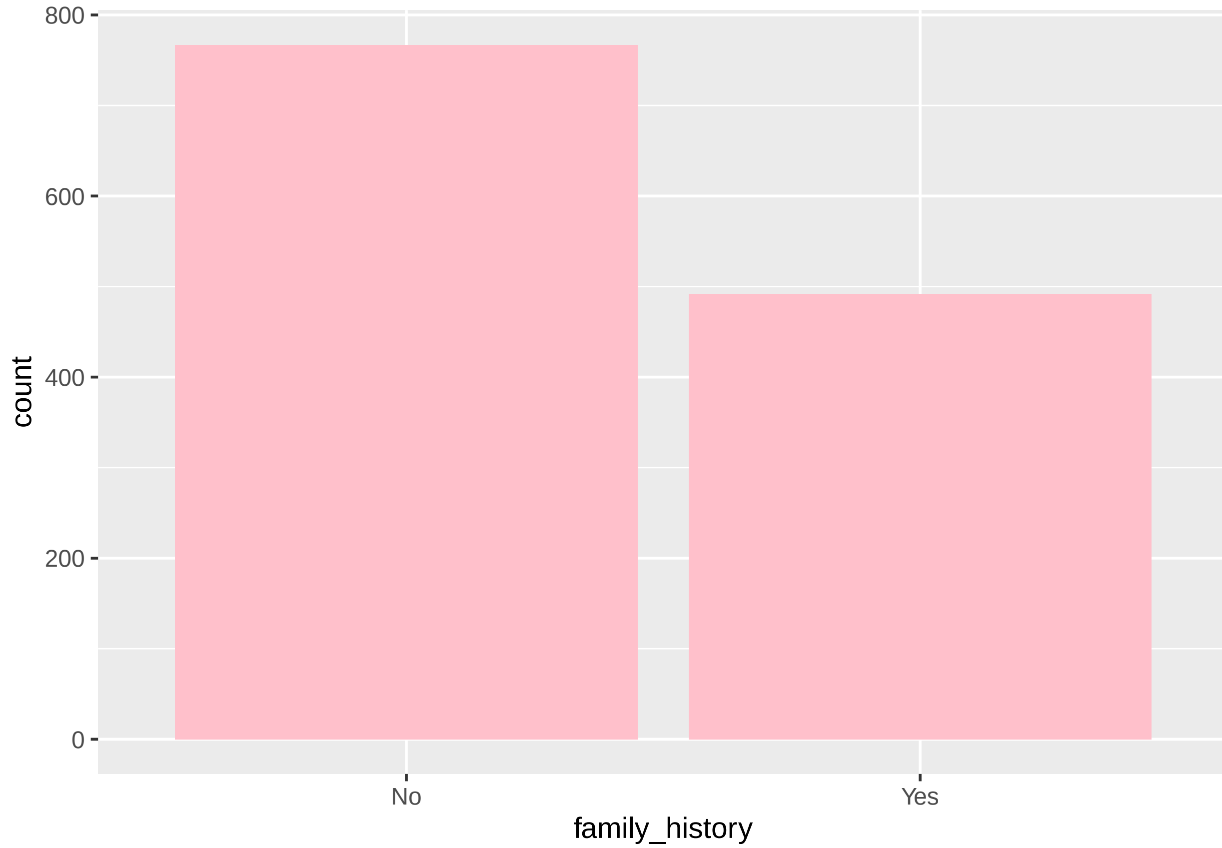

```r
# Comparing treatment ratio in Gender groups focusing on tech industry
g2 <- Tech %>% ggplot(aes(x=Gender, fill = (treatment))) +
  geom_bar(position = "fill") + ggtitle("Treatment Ratio based on tech industry")

g2
```



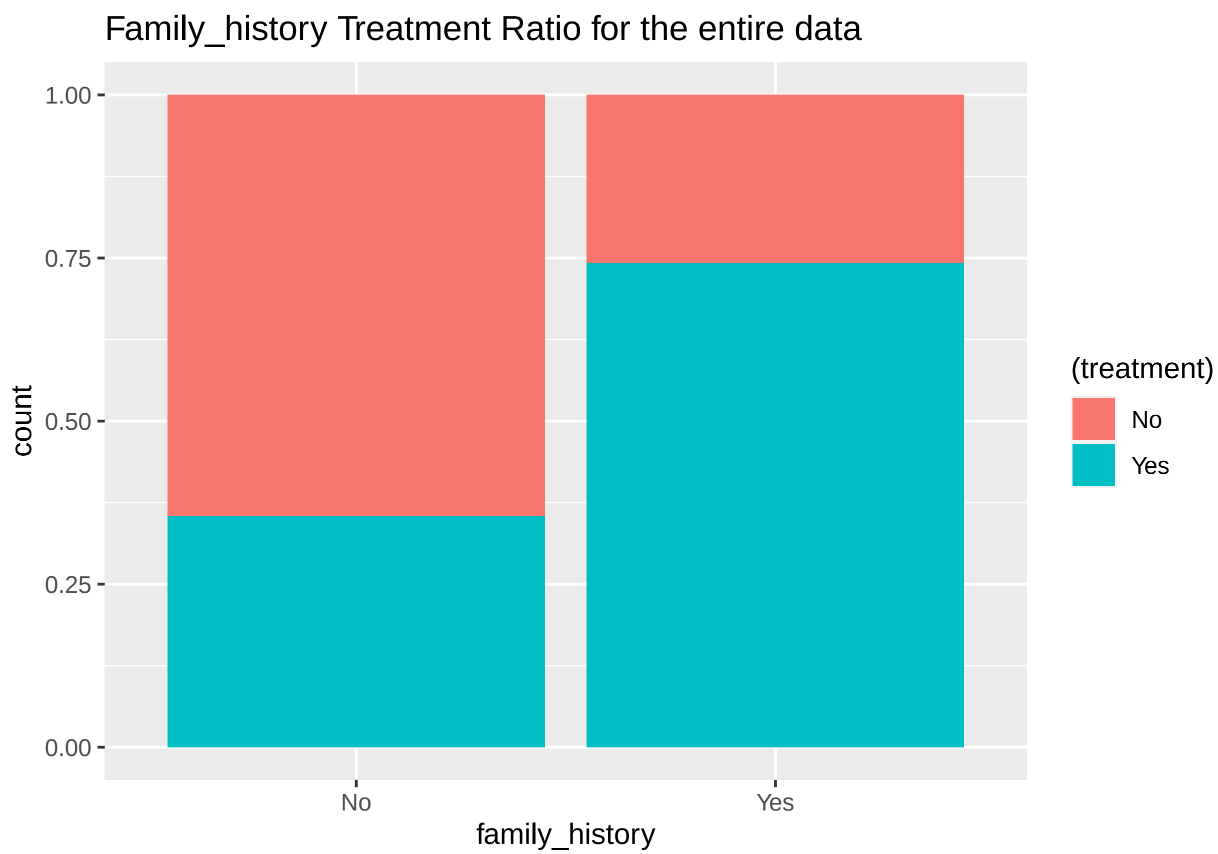
the graphs show that female and Queer employees in the tech industry sought mental health care more as compare to male employees. This also tells us that the female and queer employees might be more stressed due to increased compition in the industry and the pressure if performance.Another reason can be that, female generally have other responsibilities than work which might casue them to be more pressurized. This gives an important insight that female employees need more supervision in the tech industry regarding mental health care or treatment.

#Family history and Treatment The code below is to plot graphs to stduy the relationship or see if the fsmily history is strongly associated with the target variable 'treatment'

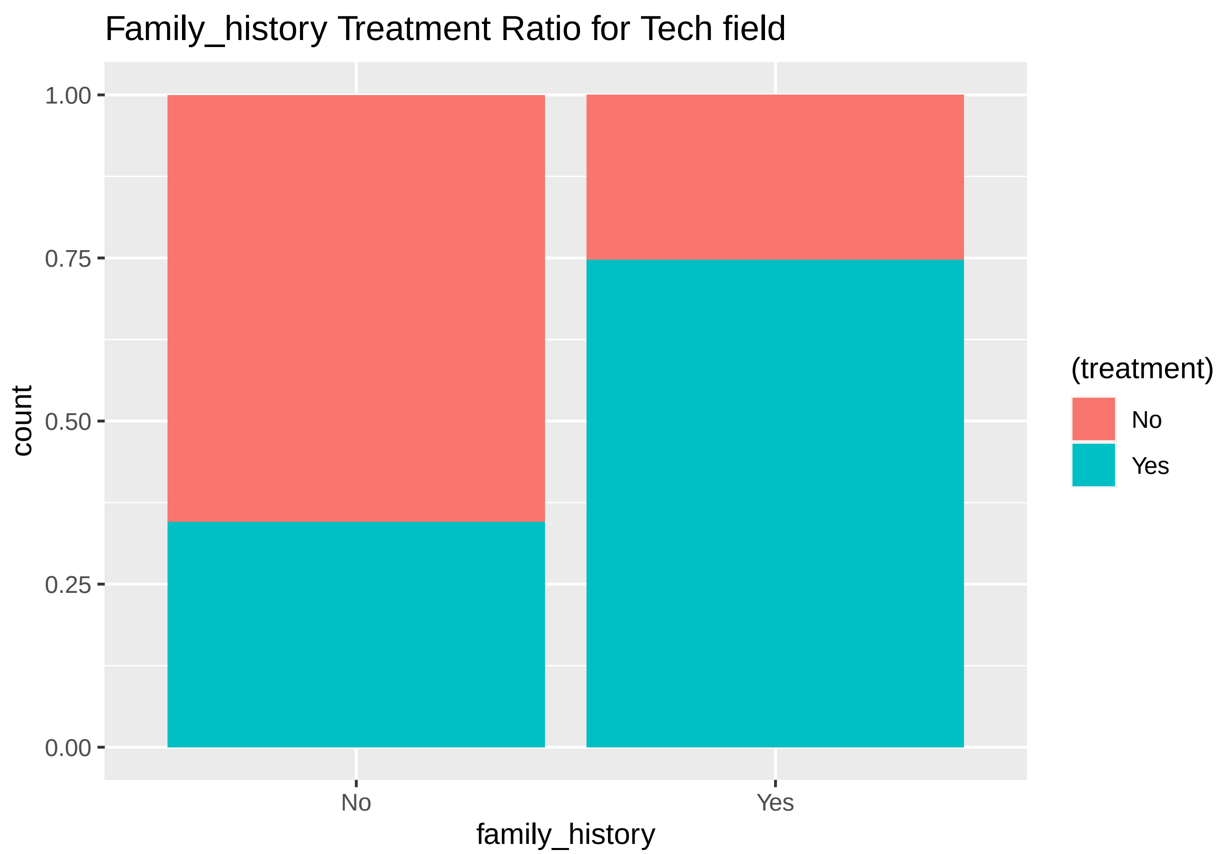
```
# studying the family_history variable
f1 <- survey %>% ggplot(aes(x=family_history)) +
  geom_bar(fill = "pink")
f1
```




```r
# Comparing Family_history treatment ratio
f2 <- survey %>% ggplot(aes(x=family_history, fill = (treatment))) +
  geom_bar(position = "fill") + ggtitle("Family_history Treatment Ratio for the entire data")
f2
```




```r
# Comparing Family_history treatment ratio focusing on tech industy
f3 <- Tech %>% ggplot(aes(x=family_history, fill = (treatment))) +
  geom_bar(position = "fill") + ggtitle("Family_history Treatment Ratio for Tech field")
f3
```

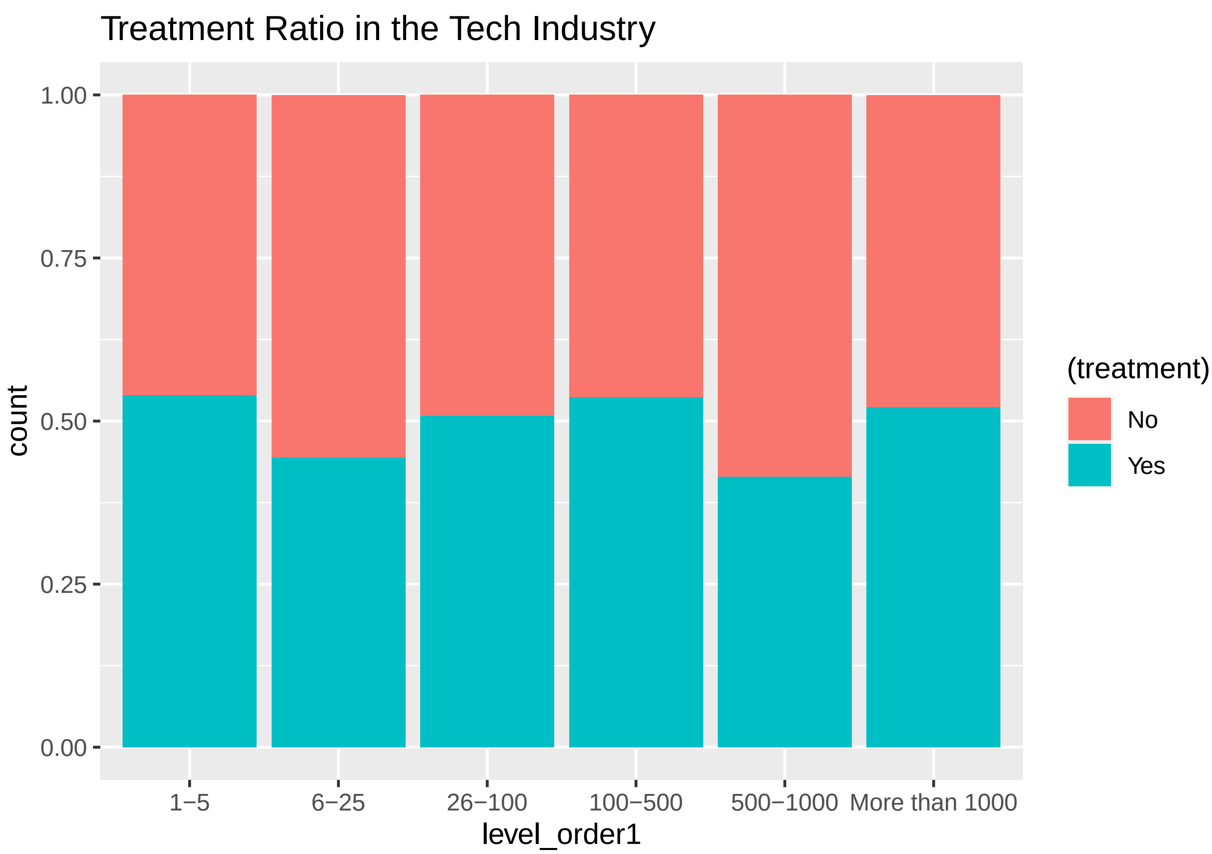


It is evident from the graph that more than 60% of the population in the survey did not have a family history if mental health disorder. However, those who did seek mental health treatment do seem to have a family history.

Treatment ratio in the tech industry

The code below is to plot graph to view the treatment ratio in each of the company sizes.

```
# level_order
level_order1 <- factor(Tech$no_employees, levels = c("1-5","6-25","26-100","100-500", "500-1000","More
#Treatment ratio in the  tech industry
z1 <- Tech %>% ggplot(aes(x=level_order1, fill = (treatment))) +
  geom_bar(position = "fill") + ggtitle("Treatment Ratio in the Tech Industry")
z1
```



The above graph bursts a myth that seeking mental health treatment does not do much with the size of the company. Generally the smaller the company or a startup, the more the stress. But this is proven to be wrong. Seeking mental health treatment is not depended on the size of the company. The above graph also shows that people belonging to companies from varied size have sought mental health treatment.

MODELING

LOGISTIC REGRESSION MODEL

Logistic regression model is mainly used for predicting discrete or categorical variables. One of the assumptions that this algorithm follows is that the target variable must be a binary variable.Moreover, logistic regression involves using a logistic function also known as sigmoid function that makes it possible to solve classification problems.We ran the logistic regression model using all the variables to cross verify our selection of the variables amongst the 27 available variables.

```r
# Fit logistic model to the data required to answer the project goal
lm <- glm( treatment ~ Age + Gender + no_employees + family_history, data = Tech, family = "binomial" )
summary(lm)
```

```
##
## Call:
## glm(formula = treatment ~ Age + Gender + no_employees + family_history,
##     family = "binomial", data = Tech)
```

```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1608  -0.8801  -0.5518   0.8967   1.8068
##
## Coefficients:
##                           Estimate Std. Error z value Pr(>|z|)
## (Intercept)                -0.9074     1.5737  -0.577    0.564
## AgeJunior                   1.0402     1.5670   0.664    0.507
## AgeSenior                   1.3318     1.5712   0.848    0.397
## AgeSuper                    1.8560     2.1102   0.880    0.379
## GenderMale                 -0.8978     0.1857  -4.834 1.34e-06 ***
## GenderQueer                 0.2386     0.7272   0.328    0.743
## no_employees100-500        -0.1205     0.2631  -0.458    0.647
## no_employees26-100          0.0162     0.2303   0.070    0.944
## no_employees500-1000       -0.6498     0.4019  -1.617    0.106
## no_employees6-25           -0.2030     0.2261  -0.898    0.369
## no_employeesMore than 1000  -0.1017     0.2414  -0.421    0.673
## family_historyYes           1.6713     0.1459  11.458  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1429.3  on 1030  degrees of freedom
## Residual deviance: 1234.2  on 1019  degrees of freedom
## AIC: 1258.2
##
## Number of Fisher Scoring iterations: 4
```

```r
coef(lm)
```

```
##                (Intercept)                  AgeJunior
##                -0.90738596                 1.04025055
##                  AgeSenior                   AgeSuper
##                 1.33183163                 1.85602939
##                 GenderMale                GenderQueer
##                -0.89777921                 0.23856043
##        no_employees100-500         no_employees26-100
##                -0.12052743                 0.01620141
##       no_employees500-1000           no_employees6-25
##                -0.64976225                -0.20304359
## no_employeesMore than 1000          family_historyYes
##                -0.10172844                 1.67131834
```

We got the AIC value of 1258.2 which we will try to lower down or improve to make our model efficient. The significant variables are Gender Male and Family history__yes as they have the lowest p values and low error rate.This tells us that employees who are male, employees who have a family history of mental health care, and employees working in a company-sized between 500 to 1000 are more susceptible to having mental health issues.

We further split the data into training - 80% and test - 20% for the purpose of feature engineering.

```
#splitting the dataset to training and testing
i <- nrow(Tech)
i
```

```
## [1] 1031
```

```
train_ind <- sample(seq_len(i), size = floor(0.8*i))

Tech_training <- Tech[train_ind, ]
Tech_testing <- Tech[-train_ind, ]
```

We define a transformation function for feature engineering.We apply the function to our training and testing data separately.

```
transformations <- function(Tech) {
  # Gender
  # Create the list of three categories
  Male <- c("Male ","Cis Man", "Malr", "Male", "male", "M", "m", "Male-ish", "maile", "Mal", "Male (CIS
  Female <- c("Female ","femail","Female (cis)","female","Female","F","Woman","f","Femake","woman","Fema
  Queer <-c ("ostensibly male, unsure what that really means","p","A little about you","queer","Neuter"

  # Categorize genders
  Tech$Gender <- sapply(
    as.vector(Tech$Gender),
    function(x) if(x %in% Male) "Male" else x )

  Tech$Gender <- sapply(
    as.vector(Tech$Gender),
    function(x) if(x %in% Female) "Female" else x )

  Tech$Gender <- sapply(
    as.vector(Tech$Gender),
    function(x) if(x %in% Queer) "Queer" else x )

  # Age
  # Replacing negative values and outliers with median
  Tech$Age <- as.numeric(Tech$Age)
  Tech$Age[which(Tech$Age<0)]<- median(Tech$Age)
  Tech$Age[which(Tech$Age>100)]<- median(Tech$Age)

  # Summary Age
  summary(Tech$Age)

  # Age categorization#
  Tech$Age1 <- cut(Tech$Age, breaks = c(0, 16, 34, 60, 75), labels = c('Fresh', 'Junior', 'Senior', 'Su

  # Verify Age group
  Tech$Age1 %>% table

  # Return the transformed dataframe
  return(Tech)
}
# Feature Engineerung for Test and Train Dataset
```

```r
Tech_training <- Tech_training %>% transformations
Tech_testing <- Tech_testing %>% transformations

# Train Data
Tech_training %>% head(2)
```

```
##     treatment Age Gender family_history   no_employees tech_company  Age1
## 329       Yes   2 Female            Yes More than 1000          Yes Fresh
## 679        No   3   Male            Yes More than 1000          Yes Fresh
```

```r
# Test data
Tech_testing %>% head(2)
```

```
##   treatment Age Gender family_history no_employees tech_company  Age1
## 5        No   2   Male            Yes         6-25          Yes Fresh
## 7        No   3   Male             No          1-5          Yes Fresh
```

We train the the logistic regression model using the training dataset to make predictions on the train and test data.

```r
# Training the logistic regression model  with feature engineering
lm_train <- glm(treatment ~ Age + Gender + family_history + no_employees, data = Tech_training, family =
summary(lm_train)
```

```
##
## Call:
## glm(formula = treatment ~ Age + Gender + family_history + no_employees,
##     family = "binomial", data = Tech_training)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.11817  -0.91356  -0.08777   0.92825   1.69000
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  -0.49522    0.46544  -1.064   0.2873
## Age                           0.36736    0.16916   2.172   0.0299 *
## GenderMale                   -0.89757    0.20619  -4.353 1.34e-05 ***
## GenderQueer                   0.08017    0.73417   0.109   0.9130
## family_historyYes             1.61799    0.16278   9.940  < 2e-16 ***
## no_employees100-500          -0.34098    0.29451  -1.158   0.2469
## no_employees26-100           -0.09368    0.25408  -0.369   0.7124
## no_employees500-1000         -0.86319    0.44283  -1.949   0.0513 .
## no_employees6-25             -0.23312    0.25204  -0.925   0.3550
## no_employeesMore than 1000 -0.34684    0.26580  -1.305   0.1919
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1142.31  on 823  degrees of freedom
## Residual deviance:  988.66  on 814  degrees of freedom
```

```
## AIC: 1008.7
##
## Number of Fisher Scoring iterations: 4
```

We see that logistic regression model with feature engineering improved the value of the AIC to 1007 from 1258.2. Thought the significant variables remain the same, we got a better performing model than the previous one.

The code below makes predictions on the training and the testing set and computes the confusion matrix and the accuracy.

```
# Predictions on the training set
Tech_training$predict_probs <- predict(lm_train, Tech_training, type = "response")
Tech_training$predict <- ifelse(Tech_training$predict_probs < 0.5, "No", "Yes")
# Predictions on the test set
Tech_testing$predict_probs <- predict(lm_train, Tech_testing, type = "response")
Tech_testing$predict <- ifelse(Tech_testing$predict_probs < 0.5, "No", "Yes")
# Confusion matrix for training data
cm_train <- table(Tech_training$treatment, Tech_training$predict, dnn = c("real", "predict"))
cm_train
```

```
##      predict
## real   No Yes
##   No  310 102
##   Yes 140 272
```

```
paste('Accuracy:', round(( cm_train['Yes','Yes'] + cm_train['No','No'] ) / sum(cm_train),2))
```

```
## [1] "Accuracy: 0.71"
```

```
# Confusion matrix for testing data
cm_test <- table(Tech_testing$treatment, Tech_testing$predict, dnn = c("real", "predict"))
cm_test
```

```
##      predict
## real  No Yes
##   No  81  24
##   Yes 38  64
```

```
paste('Accuracy:', round(( cm_test['Yes','Yes'] + cm_test['No','No'] ) / sum(cm_test),2))
```

```
## [1] "Accuracy: 0.7"
```

We achieved the accuracy of 71% with this model.That also means the model indicates that 70% of the mental health treatment predictions are correct and accurate.

OPTIMIZATION - LOGISTIC REGRESSION MODEL

In order to further improve the logistic regression model with feature engineering built above, we tried optimizing the model with stepwise AIC criterion. We chose stepwise AIC as it considers all the candidate variables in each step and checks if they fall below a certain threshold value. It works by eliminating the insignificant variables and thus reduces the complexity on the model leading to better performance.

```r
library(MASS)
```

```
##
## Attaching package: 'MASS'

## The following object is masked _by_ '.GlobalEnv':
##
##     survey

## The following object is masked from 'package:dplyr':
##
##     select

## The following object is masked from 'package:plotly':
##
##     select
```

```r
#OPTIMIZATION
#STEP AIC
step.model <- lm_train %>% stepAIC(trace = FALSE)
coef(step.model)
```

```
##       (Intercept)               Age        GenderMale       GenderQueer
##        -0.6643012         0.3261735        -0.8519297         0.1769366
## family_historyYes
##         1.6141311
```

```r
#Predictions
probabilities <- predict(step.model, Tech_testing, type = "response")
predicted.classes <- ifelse(probabilities > 0.5, "Yes", "No")
cm_1 <- table(Tech_testing$treatment, predicted.classes, dnn = c("real", "predict"))
cm_1
```

```
##      predict
## real   No Yes
##   No   86  19
##   Yes 42  60
```

```r
paste('Accuracy:', round(( cm_1['Yes','Yes'] + cm_1['No','No'] ) / sum(cm_1),2))
```

```
## [1] "Accuracy: 0.71"
```

The results of the optimization model tells us that Gender Male and Queer along with family history are the most significant variables for us to predict if an employee in the tech company needs to seek a mental health treatment. Also the model yield an accuracy of 69% which is the same as the previous model.However, it gave us one more attribute 'Queer' which is significant for our prediction. It also reduced the complexity on the model by selecting the most significant ones.

KNN MODEL

KNN is one of the most commonly used supervised machine learning algorithms. It can be used for classification, regression and forecasting. We used KNN as a classifier since our project goal was to identify who needs treatment. KNN works by considering K nearest data points for predicting a class, where the classes will be 'yes' or 'no' for treatment needed or not respectively. Euclidean distance is calculated between new data points and the nearest neighbors. This algorithm has many advantages like no assumptions are made (non-parametric) , intuitive and all data is used hence we chose to implement it .

```
# BUILDING THE KNN MODEL
trControl <- trainControl(method = 'repeatedcv',
                          number = 10,
                          repeats = 10)
set.seed(333)
fit <- train(treatment ~.,
             data = Tech_training,
             tuneGrid = expand.grid(k=2),
             method = 'knn',
             trControl = trControl)

predict_knn <- predict(fit,Tech_testing)
cm_knn <- with(Tech_testing,table(predict_knn,treatment))
cm_knn
```

```
##            treatment
## predict_knn No Yes
##         No  79  39
##         Yes 26  63
```

```
paste('Accuracy:', sum(diag(cm_knn)) / sum(cm_knn) * 100 )
```

```
## [1] "Accuracy: 68.5990338164251"
```

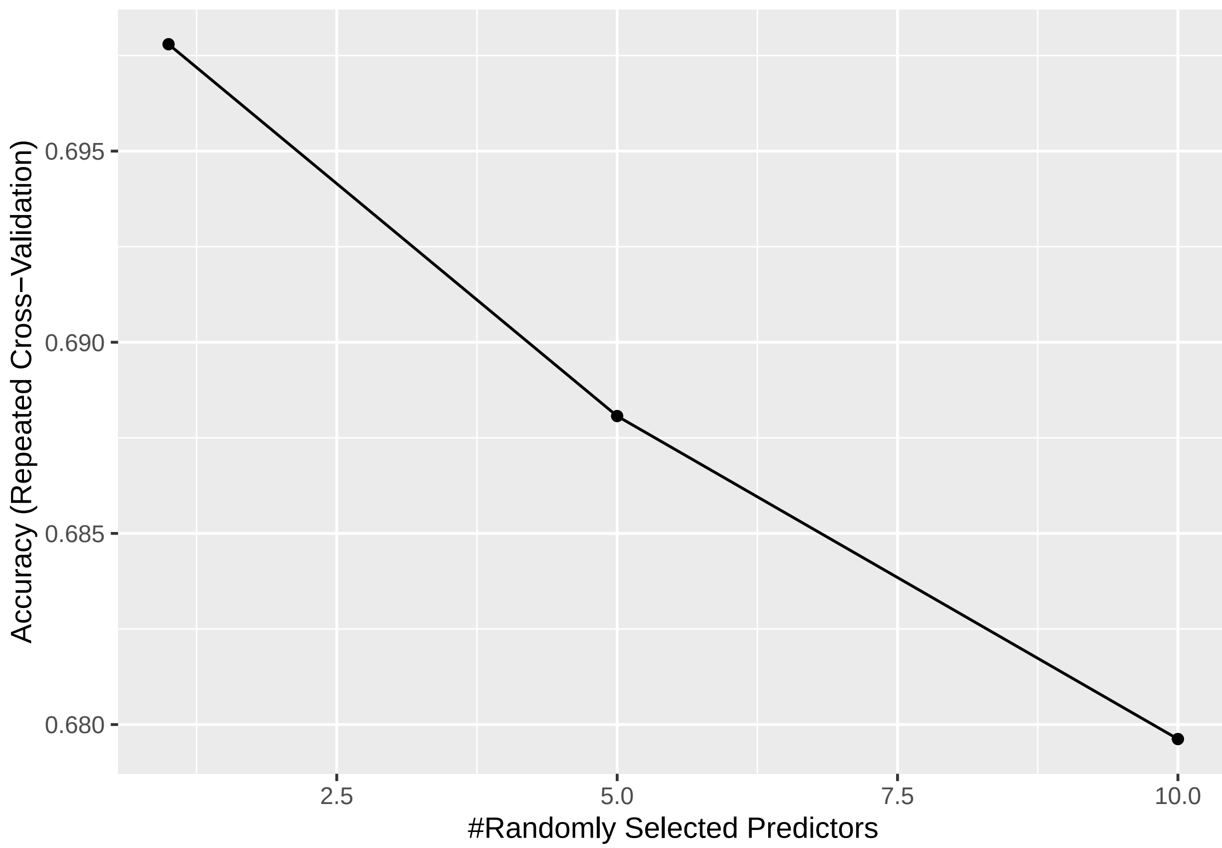
We got an accuracy of 67.32% for this model by setting a few parameters. We used repeated cross-validation, so the dataset is split randomly and divided into k folds of equal length and reiterate on all the folds. We set the value of k=2. By using trainControl() we repeated the steps for 10 times. There were 72 and 68 employees that were classified correctly by the model who did not need treatment and later who needed to treatment.We got a better accuracy with KNN model.

RANDOM FOREST MODEL

Random Forest can be defined as a model which can be defined as the model that combines together multiple decision trees of different depths in predicting the model. Here we have used random forest for improving our accuracy of the model as it reduces the overall complexity of the model that is being built. Random Forest helps in building the model that gives us information about the relationships between models and its classification. The goal in building this model is to get good predictions on the unseen data.

```
control <- trainControl(method="repeatedcv", number = 5)
grid <- data.frame(mtry = c(1, 5, 10))
```

```
train_rf <-  train(treatment ~., Tech_training,
                   method = "rf",
                   ntree = 500,
                   trControl = control,
                   tuneGrid = grid,
                   nSamp = 5000)

ggplot(train_rf)
```




```
predict <- predict(train_rf,Tech_testing)
cm <- with(Tech_testing,table(predict,treatment))
cm
```

```
##        treatment
## predict No Yes
##     No  81  38
##     Yes 24  64
```

```
paste('Accuracy:', sum(diag(cm)) / sum(cm) * 100 )
```

```
## [1] "Accuracy: 70.048309178744"
```

Here we used random forest with repeated cross-validation which helps us in error estimation in the problem and also introduces a bias in the data thereby strengthening its prediction on the unseen data. The ntree (number of decision trees) was set to 500 and parameter tuneGrid was set to grid . Whereas nSamp was set to 5000.The accuracy of the model predicted is 70.53%. 79 and 67 are the correct predictions made by the model for people not seeking treatnment and people seeking treatment.

XgBoost

XGboost is a decision tree based model which is known to improve speed and performance. This model can also be used as a regressor and classifier. It is an ensemble model since it creates new models based on errors of the previous one to improve performance. This process is carried on until no changes or improvements can be made. As the name says gradient boosting it uses gradient descent to reduce the cost and reach to convergence point (optimal value/point). When the learning rate is set to a low value it will take a lot of time to reach the optimal point and if it is set to a large value then it may never reach the optimal value .For learning rate the value can range from 0 to 1 which is also known as shrinkage(eta) . Parameter max_depth controls the depth of the tree also it was observed that as the length of the tree increases and the complexity of the model increases leading to overfitting of the model. The gamma parameter is responsible for regularization and preventing overfitting.

```
library(xgboost)
```

```
#library(xgboost)
#library(readr)
#library(ggplot2)
#library(GGally)
#library(caret) # models
#library(DALEX) # explain models
#library(DescTools) # plots
#library(doParallel) # parellel processing
#library(dplyr) # syntax
#library(inspectdf) # data overview
#library(readr) # quick load
#library(sjPlot) # contingency tables
#library(tabplot) # data overview
#library(tictoc) # measure time
#library(inspectdf) # data overview
#library(readr) # quick load
#library(randomForest)
#library(GGally)
#library(caret) # models
#library(corrplot) # correlation plots
```

```
parameterGrid <-  expand.grid(eta = 0.1, # shrinkage (learning rate)
                              colsample_bytree = c(0.5,0.7), # subsample ration of columns
                              max_depth = c(5,7), # max tree depth. model complexity
                              nrounds = 10, # boosting iterations
                              gamma = 1, # minimum loss reduction
                              subsample = 0.8, # ratio of the training instances
                              min_child_weight = 1) # minimum sum of instance weight

model_xgb <- train(treatment ~ .,
                   data = Tech_training,
                   method = "xgbTree",
                   trControl = trainControl(),
```

```
                    tuneGrid=parameterGrid)
model_xgb
```

```
## eXtreme Gradient Boosting
##
## 824 samples
##   8 predictor
##   2 classes: 'No', 'Yes'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 824, 824, 824, 824, 824, 824, ...
## Resampling results across tuning parameters:
##
##   max_depth  colsample_bytree  Accuracy   Kappa
##   5          0.5               0.6866003  0.3730130
##   5          0.7               0.6832437  0.3661863
##   7          0.5               0.6847474  0.3692339
##   7          0.7               0.6833860  0.3664336
##
## Tuning parameter 'nrounds' was held constant at a value of 10
## Tuning
##  'min_child_weight' was held constant at a value of 1
## Tuning
##  parameter 'subsample' was held constant at a value of 0.8
## Accuracy was used to select the optimal model using the largest value.
## The final values used for the model were nrounds = 10, max_depth = 5, eta
##  = 0.1, gamma = 1, colsample_bytree = 0.5, min_child_weight = 1 and subsample
##  = 0.8.
```

```
predict1 <- predict(model_xgb,Tech_testing)
cm1 <- with(Tech_testing,table(predict1,treatment))
cm1
```

```
##         treatment
## predict1 No Yes
##      No  81  38
##      Yes 24  64
```

```
paste('Accuracy:', sum(diag(cm1)) / sum(cm1) * 100 )
```

```
## [1] "Accuracy: 70.048309178744"
```

Using this model we achieved an accuracy of 68.11%. Multiple parameters are used to tune the XGBoost model like learning rate , gamma , sub-sample ratio.A learning rate of 0.1 was selected to reach optimal value and reduce the overfitting of the data .The parameter nrounds was set as 10 .Subsample ratio was set as 0.8 to randomly select 80% of training data. The sum of weights for child nodes was considered as 1. A combination of tuning parameters were tried for better performance.

SUPPORT VECTOR MACHINE MODEL

Support vector machines (SVM) can be used as a regressor as well as classifier. For our project we used SVM as a classifier since we needed to predict whether a person needs treatment or not.Support Vector Machine works by creating a margin between classes and a maximum marginal boundary is selected to separate classes from each other. Here the concept of support vector (data points) is used for maximizing the margin . Support vectors are responsible for positioning the hyperplane margins . One of the advantages of using this algorithm is that it uses less memory because of subsetting training data .There are various kernel options available to model the data linear , polynomial and radial basis function .

```
library(e1071)
```

```
model_svm<-svm(treatment~.,data=Tech_training,kernel='linear',gamma= 1,cost=100)
model_svm
```

```
##
## Call:
## svm(formula = treatment ~ ., data = Tech_training, kernel = "linear",
##     gamma = 1, cost = 100)
##
##
## Parameters:
##    SVM-Type:  C-classification
##  SVM-Kernel:  linear
##        cost:  100
##
## Number of Support Vectors:  515
```

```
test_pred <- predict(model_svm, newdata = Tech_testing)
test_pred
```

```
##   5   7  12  16  27  29  31  32  35  36  47  49  55  56  60  62
## Yes  No Yes  No  No Yes Yes  No  No  No Yes  No Yes Yes Yes  No
##  63  69  81  83  85  97 102 107 112 115 122 125 135 138 148 157
##  No  No Yes  No  No Yes  No  No Yes  No  No  No  No  No  No  No
## 159 160 161 168 170 171 173 185 191 205 211 221 231 235 236 242
##  No Yes  No  No  No Yes  No Yes  No  No Yes  No Yes  No  No  No
## 243 249 250 260 261 262 264 271 282 283 295 300 303 304 308 309
##  No Yes  No  No  No  No Yes Yes Yes  No  No  No Yes Yes Yes Yes
## 310 313 331 338 346 353 354 362 364 377 380 383 385 393 405 411
##  No Yes Yes  No Yes Yes  No Yes Yes  No Yes  No Yes  No  No Yes
## 412 417 422 423 439 446 448 450 451 455 462 464 477 479 482 493
## Yes  No  No Yes  No Yes Yes  No  No Yes Yes  No  No  No  No  No
## 496 504 512 527 532 536 541 546 547 555 556 559 563 566 569 570
## Yes Yes Yes  No Yes Yes  No  No Yes Yes Yes  No  No  No  No  No
## 575 580 589 594 596 602 603 611 613 615 616 617 620 621 630 646
## Yes  No  No  No Yes  No  No  No Yes  No Yes  No  No  No  No Yes
## 653 654 655 660 665 666 667 669 681 683 695 703 707 711 712 714
##  No Yes  No Yes Yes Yes  No  No  No Yes  No  No  No  No  No  No
## 717 720 723 736 743 747 767 768 772 777 780 786 787 788 794 810
##  No  No Yes  No Yes  No Yes Yes  No Yes  No Yes Yes  No  No  No
## 817 818 830 840 842 846 848 851 853 857 861 883 886 900 903 904
```

```
## Yes No No Yes Yes Yes No No Yes Yes Yes Yes Yes Yes No No
## 905 916 923 929 932 935 939 941 943 947 952 955 958 959 974 975
## Yes No No Yes Yes No Yes No Yes Yes No No No No Yes Yes
## 983 985 986 988 992 1001 1010 1011 1012 1018 1020 1022 1026 1028 1029
## No No Yes No Yes No No Yes No Yes No No Yes No Yes
## Levels: No Yes
```

```
cm2 <- with(Tech_testing,table(test_pred,treatment))
cm2
```

```
##          treatment
## test_pred No Yes
##       No  81  38
##       Yes 24  64
```

```
paste('Accuracy:', sum(diag(cm2)) / sum(cm2) * 100 )
```

```
## [1] "Accuracy: 70.048309178744"
```

```
#confusionMatrix(test_pred, Tech_testing$treatment )
```

We got accuracy of 70.53% using this model. There were few parameters that we tuned while building the model like cost which was set to 100.We used the kernel as 'linear' because classification of only two classes had to be done.

Conclusion

SVM, random forest and logistic regression model with feature engineering gave the best performance amongst all the models that we ran in our dataset. Highest accuracy was observed for this model that means it correctly classified most of the data as treatment needed 'yes' as 'yes' and treatment not needed to be 'no' as 'no' . Moreover, the logistic model helped us identify significant attributes that the tech industry should have focused on so that they can help the employees who are in dire need of treatment.Logistic regression model also performed with an accuracy of

Furthermore,based on the results we can say that gender and family history plays an important role in the determination of seeking mental health care. The number of men in the tech industry is relatively higher than the number of females, which may create gender baisity in the work environment leading to stress. Family history also plays a major role in mental health as a sound mind helps in giving a better performance in the workplace. When considered managing both family and work simultaneously which can be burdened and disturb the work-life balance leading to more susceptible to having mental health problems.

It can be summarized that tech companies should have schemes so that people can seek mental health care. Gender is one of the prominent variables determining mental health care so, companies can try maniniting the gender ratio. Companies should focus on employees and their mental health problems and should have a seperate care mental health department or a counselor to address their issues.

For further analysis we should have a detailed survey which includes the number of hours worked weekly, the stress level of each employee, workload etc., needs to be considered and also the attributes other than feature variables that are affecting mental health needs to be taken into consideration for more clarity and better precise prediction and arrive at a more prominent conclusion.