USE CASE STUDY REPORT

Group No.: Group 13

Student Names: Namita Kiran Mahendrakar and Anirudh Kishore Polisetty

Executive Summary:

Mental health has been a pressing issue especially in today's world, with all the work stress and pressure, it's become tough to maintain a sane life. Through this project we would like to know how useful can surveys prove in terms of detecting if people need to consider mental health treatment. With the help of information obtained from some survey questions conducted across different tech organizations, the main agenda is to see if a person needs to seek mental health treatment or not. Based on the results, certain changes can be made to the organizations and individuals in order to maintain work life balance and lead a life peacefully. To succeed in our project, we have utilized Machine Learning algorithms lime Logistic Regression, KNN Classification, Neural Networks, Radom Forest, Naïve Bayes and Classification Tress to analyze our data. We have evaluated our models using ROC curves and gains chart.

I. Background and Introduction:

Problem:

Mental health, like physical health, is important and needs to be taken care of. According to CDC, 1 in 5 US adults aged 18 or above have been reported to have mental illness as of 2016[1]. In Information Technology industries, the employees work with a lot of responsibilities, experience stress from the upper management and handle deadlines. Every employer needs to look after its employees. Many-a-times, we can see people with mental illness also have physical illness. The costs that a company incurs due to mental and physical illness is 2 to 3 times greater than with single illness. If the mental health of employees is taken care of then the employers can reduce on treatments and improve their productivity.

Goal:

Mental Health has always been a pressing issue in all walks of life. It has become dominant over the years especially in tech industry, probably due to the kind of work pressure and stress that employees undergo. This stress has taken a toll on employees' mental health and in turn is affecting their physical health as well. The objective is to provide measures to the employees in a way to make them healthy both mentally and physically. The problem statement focuses on predicting if an employee is mentally healthy or not.

Possible Solution:

To come up with different algorithm techniques to see if a person needs mental health treatment or not. We will focus on Logistic Regression and Regression Trees the most and evaluate using AUC and RMSE values.

II. Data Exploration and Visualization:

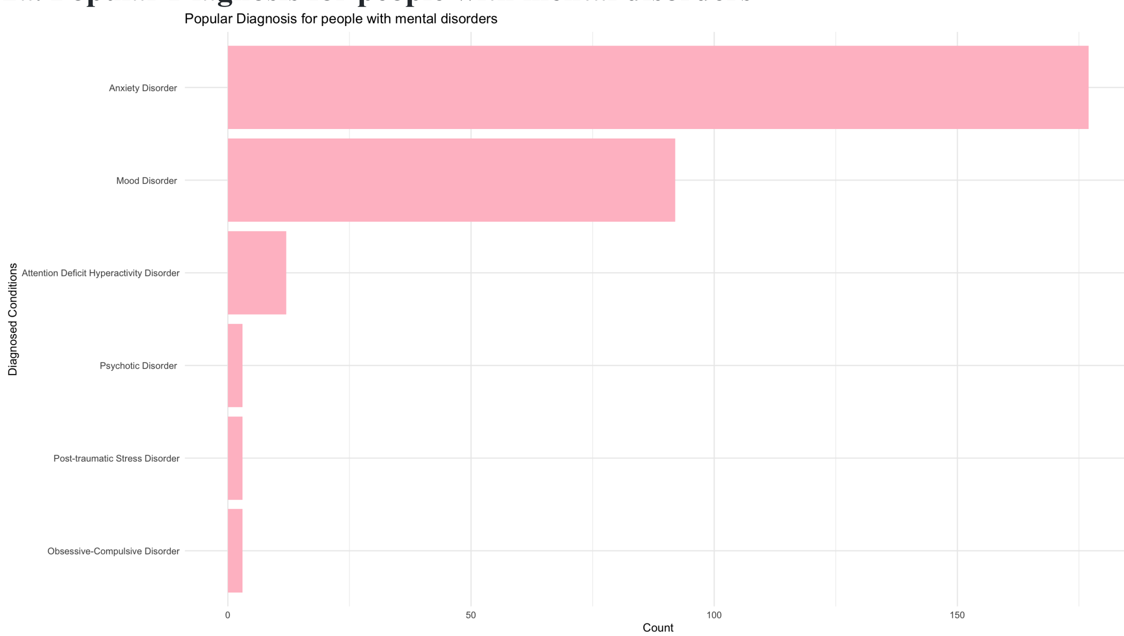
1. Data Summary

The data information is as follows. To keep a check of the employees in the IT industry, regarding their mental health, a survey was conducted in 2016. In the dataset [2], mental-heath-in-tech-2016_20161114.csv, there are 1433 rows and 63 columns. The dataset consists of survey questions answered from various different places. The data attributes are answers to survey questions. Some of the survey questions are:

- Do you have a family history of mental illness?
- Have you sought treatment for a mental health condition?
- If you have a mental health condition, do you feel that it interferes with your work?
- How many employees does your company or organization have?
- Do you work remotely (outside of an office) at least 50% of the time?
- Is your employer primarily a tech company/organization?
- Does your employer provide mental health benefits?
- Do you know the options for mental health care your employer provides?
- Has your employer ever discussed mental health as part of an employee wellness program?
- Does your employer provide resources to learn more about mental health issues and how to seek help?
- Is your anonymity protected if you choose to take advantage of mental health or substance abuse treatment resources?
- How easy is it for you to take medical leave for a mental health condition?
- Do you think that discussing a mental health issue with your employer would have negative consequences?
- Do you think that discussing a physical health issue with your employer would have negative consequences?
- Would you be willing to discuss a mental health issue with your coworkers?
- Would you be willing to discuss a mental health issue with your direct supervisor(s)?
- Would you bring up a mental health issue with a potential employer in an interview?
- Would you bring up a physical health issue with a potential employer in an interview?
- Do you feel that your employer takes mental health as seriously as physical health?
- Have you heard of or observed negative consequences for coworkers with mental health conditions in your workplace?

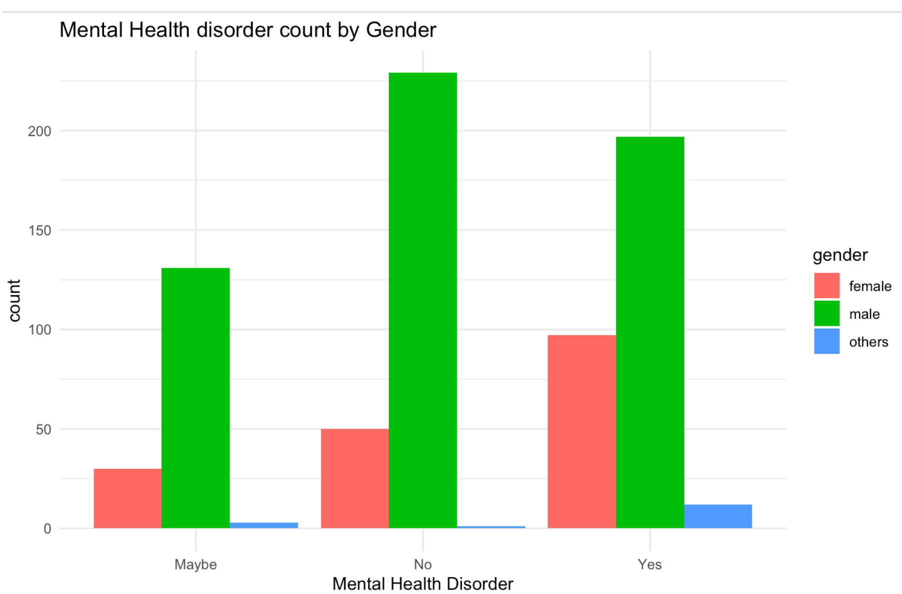
2. Data Visualizations:

2a. Popular Diagnosis for people with mental disorders



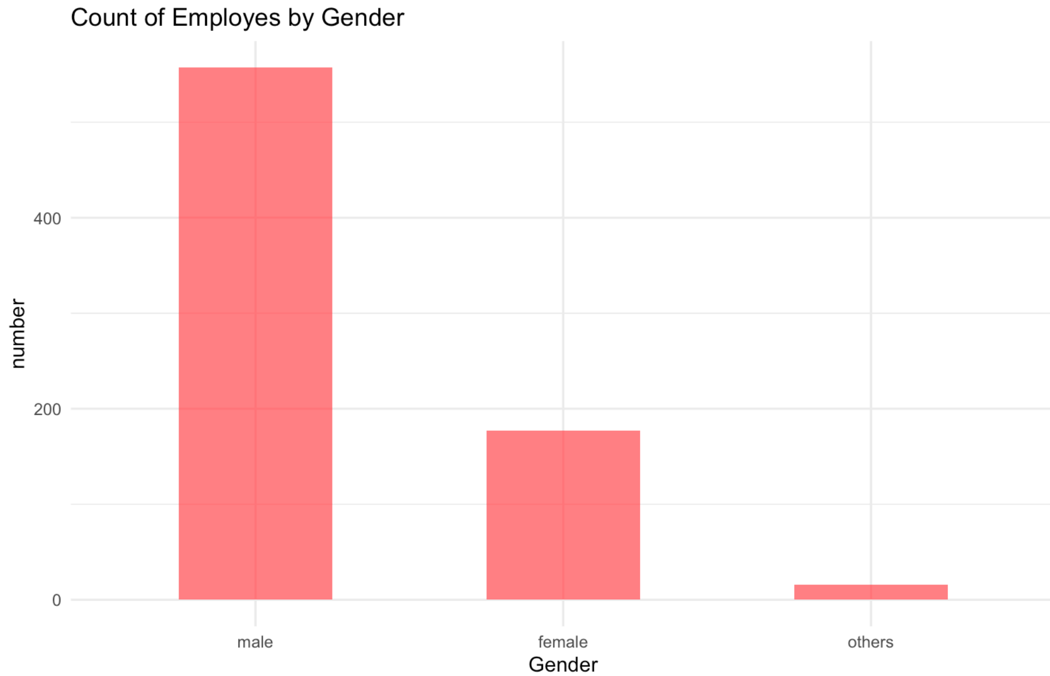
For people who have been diagnosed with mental disorders, the main diagnosis was Anxiety disorder, Mood disorder, OCD, Attention deficit disorder, Psychotic disorder, Post-traumatic stress disorder. These can be considered the major contributing factors for mental illness.

2b. Mental Health disorder count by Gender



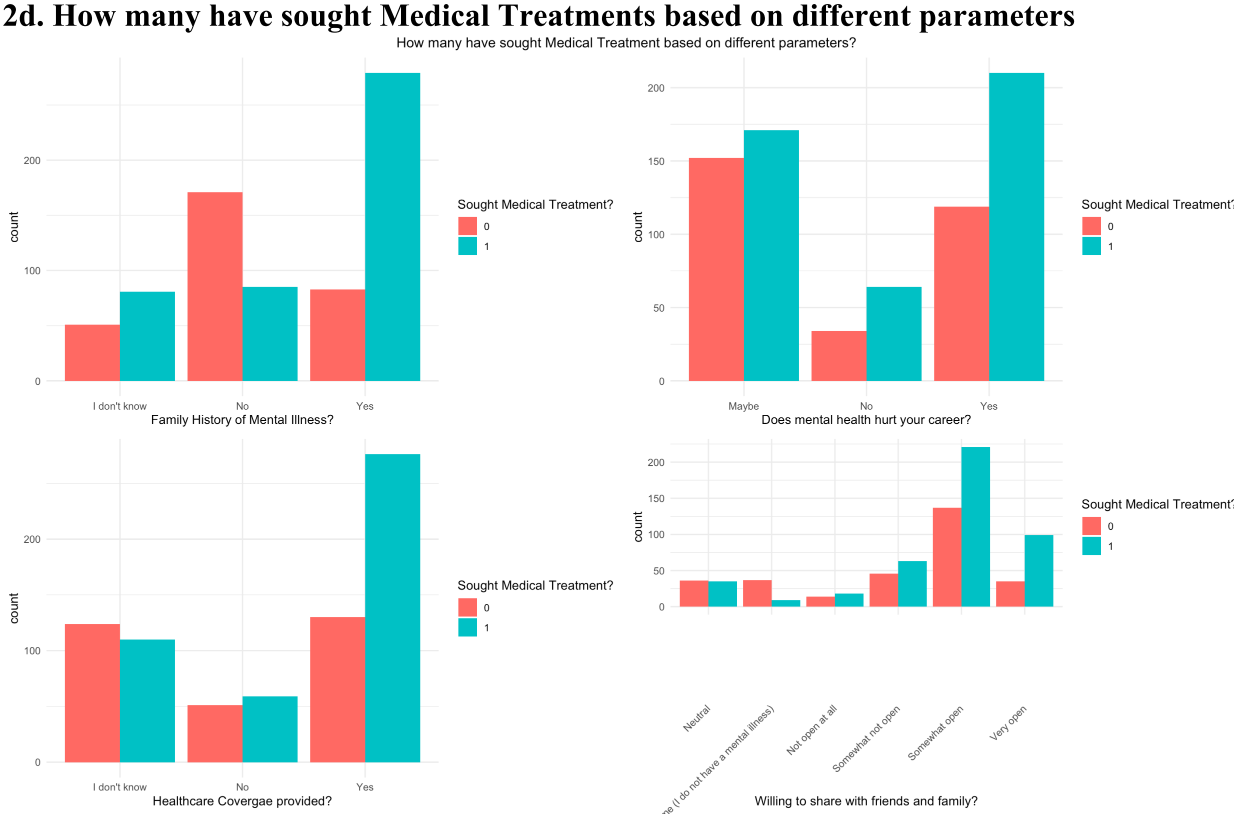
Male have been diagnosed the highest having mental health disorders followed by female and others.

2c. Count of Employees by Gender



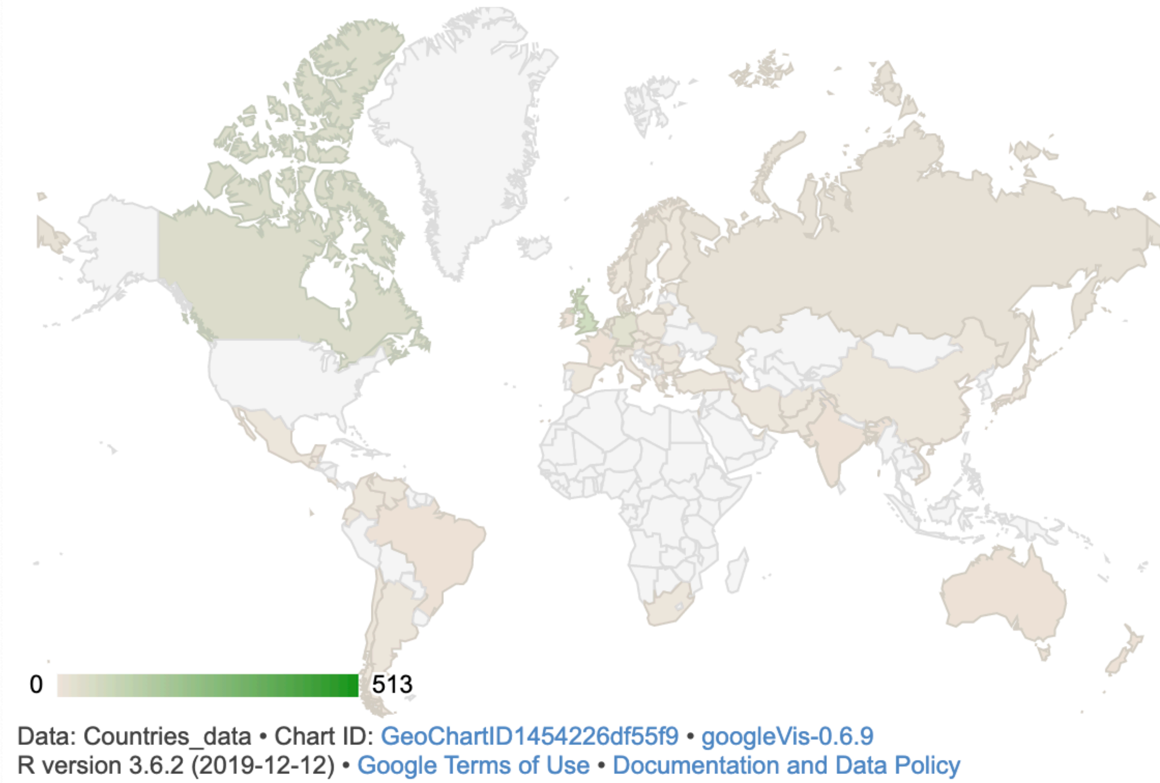
The number of employees that took part in the survey are dominated by male, followed by female and others.

2d. How many have sought Medical Treatments based on different parameters



People with a family history of mental illness, people who think mental health will hurt their career, who have healthcare coverage and who are somewhat open to share with family and friends have sought medical treatment

2e. Number of tech employees working by country on a global scale



This is an interactive chart the displays countries that have been actively participating in these mental health surveys. Majority of Employees working in USA, UK, Canada & Germany seem to be taking part in Tech Organizations. Therefore, it can be considered, mental health is given importance in these Countries.

Var1	Freq
<fctr>	<int>
United States of America	513
United Kingdom	60
Canada	45
Germany	29

III. Data Preparation and Pre-Processing:

Performed pre-processing to clean the data and structured the data for further analysis Performed Exploratory Data Analysis on Mental Health data to derive relationships between variables and observe distribution of the entire dataset.

1) Data Pre-processing:

Renaming Columns:
Since the column names were questions from the survey and were too long to comprehend, we have renamed the columns for convenience.

Handling Missing Data:
We have removed missing information (null, N/A values) and have omitted irrelevant columns. Since we have 63 columns and most of them were categorical, we retained the columns that suits our best interests for this project.

We have modified the data for few of columns for further better analysis and by removing null values and grouping data with same meaning but different values. Some of those attributes are number of employees, anonymity_protected, mental_vs_physical health, mental_health_consequences and offer_benefits.

Handling Gender column:
Gender column has a lot of values that significantly can be categorized down to 3 major categories, male, female, others.

Filtering Data:
Since, this is a survey conducted only in tech organization, filtered the data to reflect only tech related organization's records.
Also, filtered out the data who are self-employed and retained records of those only who aren't
After pre-processing the dataset was reduced from 1433 rows to 749 rows

Number of rows after data cleaning and preprocessing = 749
data loss = (1433-749)/1433 = 0.477 = 47%
Therefore, we lost 47% of unnecessary data after preprocessing.

VI. Data Mining Techniques and Implementation:

We divided the dataset into train and test sets in a ratio of 80:20 and performed different data mining techniques on it.

Data Mining Models/ Methods:

1. KNN

Performed KNN regression age (numerical) as predictor variable and sought_treatment as output variable. Results of our analysis are as follows:

Accuracy Measures after standardizing predictor variables

For k=1

```
                 ME      RMSE       MAE       MPE     MAPE
Test set 0.01538671 0.5184478 0.4951855 -3.371757 86.03842
```

For k=3

```
                 ME      RMSE       MAE       MPE     MAPE
Test set 0.01538671 0.5184478 0.4951855 -3.371757 86.03842
```

For k=7

```
                 ME      RMSE       MAE      MPE     MAPE
Test set 0.01312564 0.5170298 0.4945405 -3.59398 86.26065
```

For k=9

```
                  ME      RMSE       MAE      MPE     MAPE
Test set 0.004478766 0.5130024 0.4950607 -4.50831 87.17498
```

Correlation coefficients after standardizing predictor variables

```
[1] "Correlation coefficient for k=1 is: -0.113835130041248"
[1] "Correlation coefficient for k=3 is: -0.113835130041248"
[1] "Correlation coefficient for k=7 is: -0.106155268183294"
[1] "Correlation coefficient for k=9 is: -0.111362429609935"
```

We decided to stop at k=9 since RMSE value was declining from 9, and the correlation coefficient isn't showing any significance with the target variable. This might not be our best model, so we decided to move forward with Naïve Bayes

2.Naïve Bayes

The probability values of the classifiers are as follows:

```
Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)
```

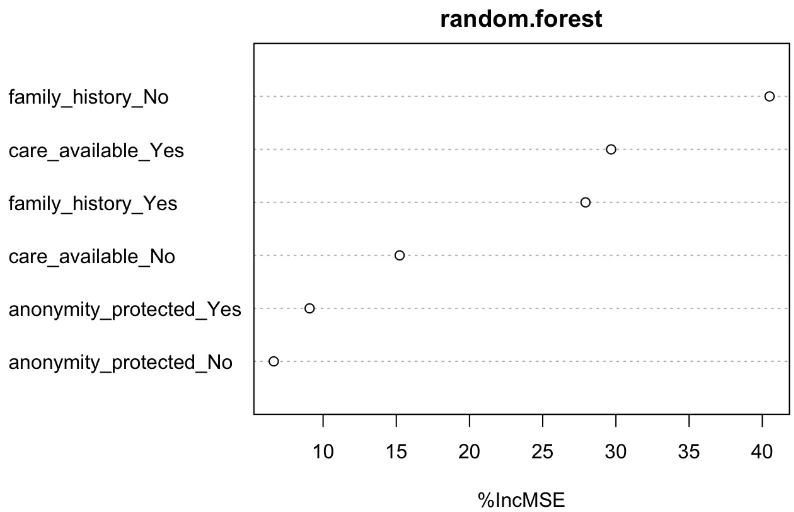
```
A-priori probabilities:
Y
    0     1
0.425 0.575

Conditional probabilities:
   care_available
Y    Not sure        No       Yes
  0 0.3568627 0.4823529 0.1607843
  1 0.3304348 0.2985507 0.3710145

   phy_health_interview
Y       Maybe        No       Yes
  0 0.4313725 0.3098039 0.2588235
  1 0.4231884 0.3333333 0.2434783

   mental_health_interview
Y        Maybe         No        Yes
  0 0.34509804 0.53725490 0.11764706
  1 0.25217391 0.69275362 0.05507246

   family_history
Y   I don't know        No       Yes
  0    0.1568627 0.5647059 0.2784314
  1    0.1884058 0.1913043 0.6202899

   have_mhd
Y       Maybe        No       Yes
  0 0.2156863 0.6784314 0.1058824
  1 0.2289855 0.1826087 0.5884058

   Have.you.been.diagnosed.with.a.mental.health.condition.by.a.medical.professional.
Y           No        Yes
  0 0.93333333 0.06666667
  1 0.20000000 0.80000000

   gender
Y        female        male      others
  0 0.156862745 0.835294118 0.007843137
  1   0.286956522 0.681159420 0.031884058
```

3. Logistic Regression

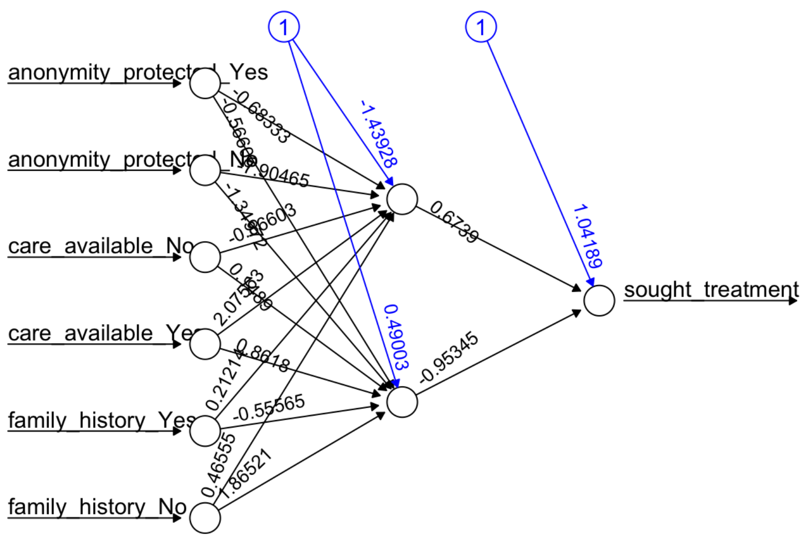
```
Call:
glm(formula = sought_treatment ~ ., family = "binomial", data = logit.train.df)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.7993  -0.4563   0.2173   0.3019   2.4302

Coefficients:
```

```
                                      Estimate Std. Error z value        Pr(>|z|)
(Intercept)                           -0.27179    0.49040  -0.554        0.579422
care_availableNo                       0.20534    0.32177   0.638        0.523357
care_availableYes                      0.03409    0.35833   0.095        0.924215
anonymity_protectedNo                  0.56740    0.50678   1.120        0.262878
anonymity_protectedYes                 0.63421    0.31489   2.014        0.044004 *
family_historyNo                      -0.51814    0.37725  -1.373        0.169610
family_historyYes                      0.05110    0.35591   0.144        0.885831
mhd.pastNo                            -1.37882    0.38242  -3.606        0.000312
***
mhd.pastYes                            0.99501    0.37093   2.683        0.007307
**
have_mhdNo                            -0.13756    0.36104  -0.381        0.703202
have_mhdYes                            0.05669    0.43272   0.131        0.895771
diagnosed.by.a.medical.professionalYes  2.86166    0.39154   7.309 0.00000000000027
***
gendermale                            -0.62719    0.33328  -1.882        0.059858 .
genderothers                          12.46786  793.20308   0.016        0.987459
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 808.42  on 599  degrees of freedom
Residual deviance: 372.11  on 586  degrees of freedom
AIC: 400.11
Number of Fisher Scoring iterations: 15
Setting levels: control = 0, case = 1
Setting direction: controls < cases

Call:
roc.default(response = logit.train.df$sought_treatment, predictor =
logit.mental.health$fitted.values,     percent = TRUE, plot = TRUE, legacy.axes =
TRUE, xlab = "False Positive Percentage",     ylab = "True Postive Percentage", col =
"#377eb8", lwd = 2,     main = "ROC Curve")

Data: logit.mental.health$fitted.values in 241 controls
(logit.train.df$sought_treatment 0) < 359 cases (logit.train.df$sought_treatment 1).
```

4. Random Forest

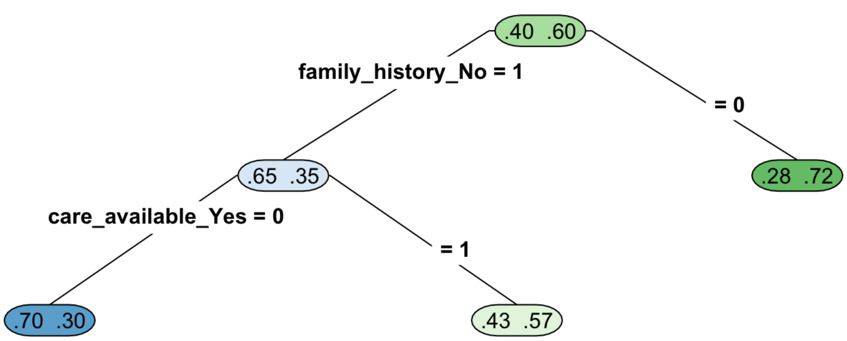


5. Neural Networks



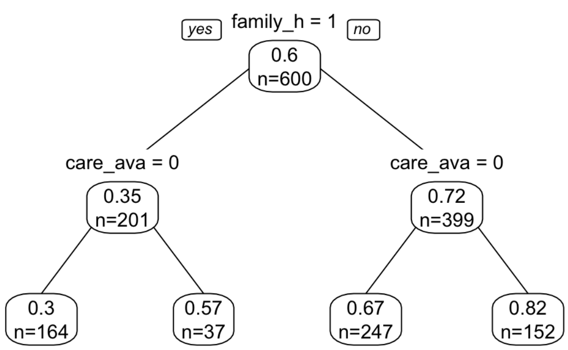
```
Confusion matrix (absolute):
          Actual
Prediction   0   1 Sum
       0    54   1  55
       1    89   6  95
       Sum 143   7 150

Confusion matrix (relative):
          Actual
Prediction    0    1  Sum
       0   0.36 0.01 0.37
       1   0.59 0.04 0.63
       Sum 0.95 0.05 1.00

Accuracy:
0.4 (60/150)

Error rate:
0.6 (90/150)

Error rate reduction (vs. base rate):
-11.8571 (p-value = 1)
```

6. Classification Tree



7. Regression Tree

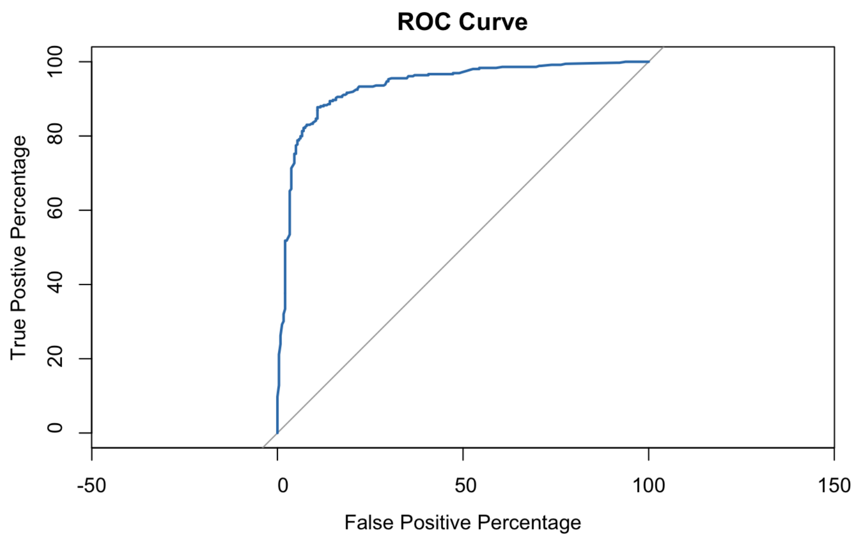
```
Setting levels: control = 0, case = 1
Setting direction: controls < cases

Call:
roc.default(response = reg.tree.train.df$sought_treatment, predictor =
mental.health.regression$where,     percent = TRUE, plot = TRUE, legacy.axes = TRUE,
xlab = "False Positive Percentage",     ylab = "True Postive Percentage", col =
"#377eb8", lwd = 2,     main = "ROC Curve")

Data: mental.health.regression$where in 241 controls
(reg.tree.train.df$sought_treatment 0) < 359 cases (reg.tree.train.df$sought_treatment
1).
```

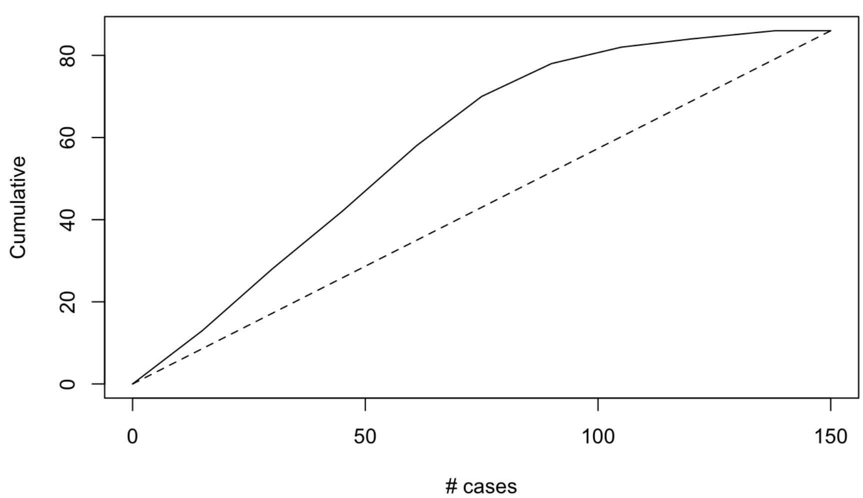


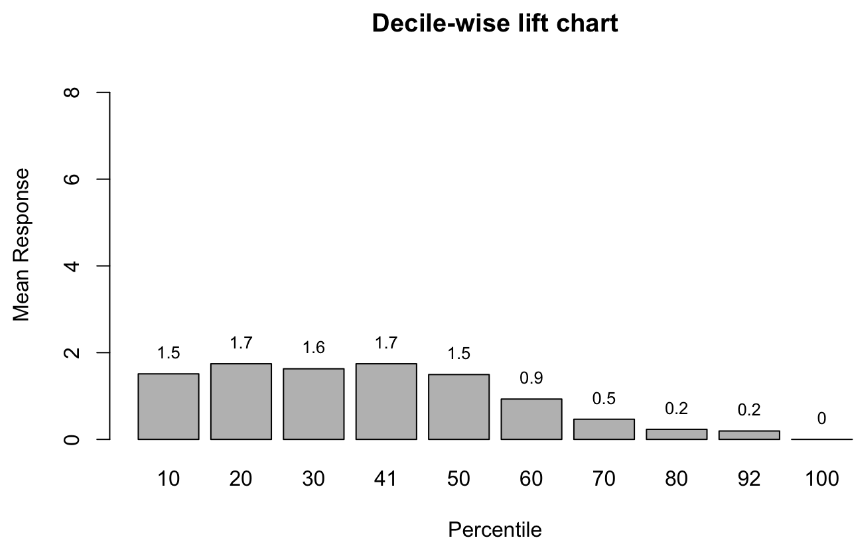
V. Performance Evaluation:

For Logistic Regression

```
Area under the curve: 93.51%
```

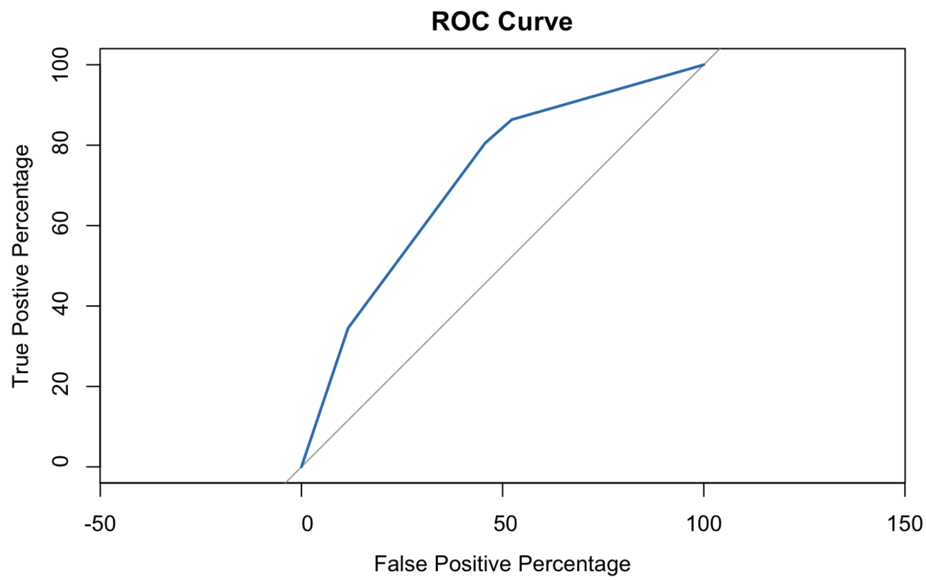






For Regression Tree

```
Area under the curve: 71.58%
```



We have evaluated our models using RMSE and ROC, Gains and Decile-wise lift plots. Based on our analysis, Logistic Regression and Regression Tree seem to be the best fit models for identifying if a person needs mental health treatment or not.

VI. Discussion and Recommendation:

The predicted logistic regression and regression tree models potentially help us determine whether the employee is prone to take treatment for mental illness or not. The results will help the organizations to focus on taking extreme measures to curb this issue. This technique can be used in other walks of life not just tech industries because mental illness is prevailing everywhere. It could provide more awareness as to how important it is to openly talk about mental health issues and use these results to reduce the number of mental health victims globally.

VII. Summary:

We conclude that mental health is still a pressing issue and needs to be addressed not just in tech industry or not just for a working professional but for every person. Based on our analysis, we found that although most of the people are aware about the options for mental health treatment options, we can still do a lot in terms of awareness and potentially improving people's lives.

References:

1. https://www.cdc.gov/workplacehealthpromotion/tools-resources/workplace-health/mental-health/index.html

2. https://www.kaggle.com/osmi/mental-health-in-tech-2016

Appendix: R Code for use case study

~~~~
```{r}
library(readxl)
library(dplyr)
library(tidyverse)

# Load Data

data <- read.csv("mental-heath-in-tech-2016_20161114.csv")
head(data)
```

```{r Renaming Columns}
#Rename Columns

colnames(data)[1]<-"self_employed"
colnames(data)[2]<-"no_of_employees"
colnames(data)[3]<-"tech_company"
colnames(data)[5]<-"offer_benefits"
colnames(data)[6]<-"care_available"
colnames(data)[7]<-"wellness_campaign"
colnames(data)[8]<-"offer_help"
colnames(data)[9]<-"anonymity_protected"
colnames(data)[11]<-"mental_health_consequence"
colnames(data)[12]<-"phy_health_consequence"
colnames(data)[13]<-"discuss_coworkers"
colnames(data)[14]<-"discuss_supervisor"
colnames(data)[15]<-"mental_vs_physical"
colnames(data)[16]<-"obs_consequence"
colnames(data)[37]<-"phy_health_interview"
colnames(data)[39]<-"mental_health_interview"
colnames(data)[46]<-"family_history"
colnames(data)[48]<-"have_mhd"
colnames(data)[53]<-"sought_treatment"
colnames(data)[56]<-"age"
colnames(data)[57]<-"gender"
colnames(data)[60]<-"country"
colnames(data)[63]<-"remote_work"
```

```{r}
#Data Cleaning
#Removing employees who are self Employeed
~~~~

~~~~
data1 <- data
data1 <- data1 %>%
        filter(self_employed == 0)

#Removing employees who are not working in tech organization
data1 <- data1 %>%
        filter(tech_company == 1)

#Drop empty and irrelevant columns
empty_columns = c(
          "Do.you.know.local.or.online.resources.to.seek.help.for.a.mental.health.disorder.",

"If.you.have.been.diagnosed.or.treated.for.a.mental.health.disorder..do.you.ever.reveal.this.to.cli
ents.or.business.contacts.",

"If.you.have.revealed.a.mental.health.issue.to.a.coworker.or.employee..do.you.believe.this.has.i
mpacted.you.negatively.",

"If.you.have.been.diagnosed.or.treated.for.a.mental.health.disorder..do.you.ever.reveal.this.to.co
workers.or.employees.",

"If.you.have.revealed.a.mental.health.issue.to.a.client.or.business.contact..do.you.believe.this.has
.impacted.you.negatively.",
          "Do.you.believe.your.productivity.is.ever.affected.by.a.mental.health.issue.",

"If.yes..what.percentage.of.your.work.time..time.performing.primary.or.secondary.job.functions.
.is.affected.by.a.mental.health.issue.",

"Do.you.have.medical.coverage..private.insurance.or.state.provided..which.includes.treatment.of
..mental.health.issues.")

irrelevent_columns = c("self_employed",
          "What.US.state.or.territory.do.you.live.in.",
         "What.US.state.or.territory.do.you.work.in.",
         "What.country.do.you.live.in.",
         "Why.or.why.not..1",
         "Why.or.why.not.")
data2 = data1

data2 <- data2 %>%
  select(-irrelevent_columns, -empty_columns)
```

```{r Male/Female/Others}
#Data preprocessing
data2$gender <- data2$gender %>% str_to_lower()
~~~~

```
male <- c("male", "m", "male-ish", "maile", "mal", "male (cis)", "make", "male ", "man","msle",
"mail", "malr","cis man", "cis male", "male.", "sex is male", "i'm a man why didn't you make this
a drop down question. you should of asked sex? and i would of answered yes please. seriously
how much text can this take? ", "m|", "cisdude")

female <- c("cis female", "cis female ", "f", "female", "woman",  "femake", "female ","cis-
female/femme", "female (cis)", "femail","i identify as female.", "fm", "female/woman",
"cisgender female", "fem", "female (props for making this a freeform field, though)", " female",
"cis-woman", "           f", "female assigned at birth ")

others <- c("trans-female", "something kinda male?", "queer/she/they", "non-binary","nah", "all",
"enby", "fluid", "genderqueer", "androgyne", "agender", "male leaning androgynous", "guy (-ish)
^_^", "trans woman", "neuter", "female (trans)", "queer", "ostensibly male, unsure what that
really means", "bigender", "transitioned, m2f", "genderfluid (born female)",
"other/transfeminine", "female or multi-gender femme", "androgynous", "male 9:1 female,
roughly", "other", "nb masculine", "genderfluid", "genderqueer woman", "mtf",
"male/genderqueer", "nonbinary", "unicorn", "male (trans, ftm)", "transgender woman", "female-
bodied; no feelings about gender", "genderflux demi-girl", "afab" )


data2$gender <- sapply(as.vector(data2$gender), function(x) if(x %in% male) "male" else if (x
%in% female) "female" else if (x %in% others) "others" )
# Modifying data

data2<-data2[!data2$care_available%in%c("N/A","" ),]
data2$care_available<-as.factor(as.character(data2$care_available))
levels(data2$care_available)[levels(data2$care_available)=="I am not sure"]<-"Not sure"

data2<-data2[!data2$no_of_employees%in%c(""),]
data2$no_of_employees<-as.factor(as.character(data2$no_of_employees))
levels(data2$no_of_employees)[levels(data2$no_of_employees)=="More than 1000"]<-">1000"

data2<-data2[!data2$anonymity_protected%in%c(""),]
data2$anonymity_protected<-as.factor(as.character(data2$anonymity_protected))


data2<-data2[!data2$mental_vs_physical%in%c(""),]
data2$mental_vs_physical<-as.factor(as.character(data2$mental_vs_physical))

data2<-data2[!data2$mental_health_consequence%in%c(""),]
data2$mental_health_consequence<-as.factor(as.character(data2$mental_health_consequence))

data2<-data2[!data2$offer_benefits%in%c("","Not eligible for coverage / N/A"),]
data2$offer_benefits<-as.factor(as.character(data2$offer_benefits))
```

```
```

```{r train test}
# drop NA values

data3 <- data2 %>% select(-Is.your.primary.role.within.your.company.related.to.tech.IT.)
data3 <- data3 %>% drop_na()

nrow(data3)
#Data Split to train : 80%

set.seed(2)
train.index <- sample(c(1:dim(data3)[1]), dim(data3)[1]*0.8)
train.df <- data3[train.index, ]
test.df <- data3[-train.index, ]
```

Number of rows after data cleaning and preprocessing = 750
data loss = (1433-749)/1433 = 0.477 = 47%
Therefore, we lost 47% of unnecessary data after preprocessing.

```{r Data Visulizations}

# Barplot for Popular top 10 Mental Health disorders vs Frequency

# Separate Multiple disorders which are seperated by "|"
data_disorder = data3
#unique(data_disorder$If.yes..what.condition.s..have.you.been.diagnosed.with.)
temp1 = data_disorder %>% separate(If.yes..what.condition.s..have.you.been.diagnosed.with.,
sep = '\\|', c('mhdisorder_1', 'mhdisorder_2', 'mhdisorder_3', 'mhdisorder_4', 'mhdisorder_5',
'mhdisorder_6', 'mhdisorder_7', 'mhdisorder_8', 'mhdisorder_9'), fill = 'right')

temp2 = temp1 %>%
  select(matches('mhdisorder_[1-9]')) %>% mutate_all(.funs = 'as.factor')
temp2 %>% select(matches('mhdisorder_[1-9]')) %>% str()

# Column bind the new generated columns to the data_disorder
data_disorder = cbind(data_disorder, temp2)

# Consider only the disorder name before "("
temp1 = table(data_disorder$mhdisorder_1)
data_disorder$mhdisorder_1 = factor(data_disorder$mhdisorder_1, levels =
names(temp1[order(temp1, decreasing = TRUE)]))
levels(data_disorder$mhdisorder_1) = sapply(strsplit(levels(data_disorder$mhdisorder_1), split
= "\\("), `[`, 1)
```

```
#Popular Diagnosis
v1 <- data_disorder %>%
  select(have_mhd,mhdisorder_1) %>%
  filter(have_mhd == "Yes") %>%
  group_by(mhdisorder_1) %>%
  dplyr::summarise(count=n()) %>%
  arrange(desc(count)) %>%
  top_n(5)

# Drop NA Values
v1 <- v1 %>% drop_na()

#Remove irrelevent record
v1 = subset(v1, mhdisorder_1 != "I haven\'t been formally diagnosed, so I felt uncomfortable
answering, but Social Anxiety and Depression.")

#v1 <- data %>%
 #
select(Do.you.currently.have.a.mental.health.disorder.,If.yes..what.condition.s..have.you.been.di
agnosed.with.) %>%
  #filter(Do.you.currently.have.a.mental.health.disorder. == "Yes") %>%
  #group_by(If.yes..what.condition.s..have.you.been.diagnosed.with.) %>%
  #summarise(count=n()) %>%
  #arrange(desc(count)) %>%
  #top_n(10)
# Barplot
ggplot(v1, aes(reorder(mhdisorder_1,count), count)) + geom_col(fill = 'pink') + coord_flip() +
labs(y="Count",x="Diagnosed Conditions", title = "Popular Diagnosis for people with mental
disorders") + theme_minimal()
```
```

```{r Data Visualization with Gender}
library(ggplot2)
data_disorder = data3
data_disorder$gender <- data_disorder$gender %>% str_to_lower()

data_disorder = filter(data_disorder, gender != "NULL")
data_disorder$gender <- as.factor(unlist(data_disorder$gender))

#  Barplot for diagnozed employees by gender
ggplot(data_disorder, aes(x = have_mhd)) +
 geom_bar(aes(fill = gender), position = "dodge") +
 theme(legend.position = "top") + theme_minimal() + labs(x="Mental Health Disorder", title =
"Mental Health disorder count by Gender")

#unique(data_disorder$have_mhd)
```
```

```

```{r}
# Count of population who work in tech companies

library (plyr)
percentage_of_male_female <- data_disorder%>%
  filter(tech_company == 1) %>%
  group_by(gender)%>%
  dplyr::summarize(number=n())%>%
  mutate(percent=signif(number/sum(number),3))
percentage_of_male_female

ggplot(percentage_of_male_female, aes(x=reorder(gender,-number),y = number, alpha = 0.6)) +
 geom_col(fill = "red", width = 0.5) +
 #theme(legend.position = "top")
 theme_minimal() + labs(x="Gender", title = "Count of Employes by Gender" ) +
theme(legend.position="none")

```
Male popultion is more in tech industry than femal population.

```{r}
# World map for Number of tech employes working in Country

library(googleVis)

Countries_data <- data.frame(table(data_disorder$country))
Countries_data <- Countries_data %>%
                  arrange(desc(Freq))
Countries_data
geoMap <- gvisGeoChart(Countries_data,locationvar="Var1",colorvar="Freq",
                     options=list(dataMode="regions"))
plot(geoMap)
```
Majority of Employes working in USA, UK, Canada & Germany seem to be taking part in Tech Organizations.
Therefore, it can be considered, mental health is given importaance in these Countries.

```{r}
#sought treatment vs other parameters
library(gridExtra)

p1<-ggplot(data_disorder, aes(x=family_history,fill = as.factor(sought_treatment))) +

```
geom_bar(aes(group = as.factor(sought_treatment)), position = "dodge") +
guides(fill=guide_legend(title="Sought Medical Treatment?"))+ labs(x="Family History of
Mental Illness?") + theme_minimal()

data_disorder$Do.you.feel.that.being.identified.as.a.person.with.a.mental.health.issue.would.hurt
.your.career.<-
as.factor(as.character(data_disorder$Do.you.feel.that.being.identified.as.a.person.with.a.mental.h
ealth.issue.would.hurt.your.career.))
levels(data_disorder$Do.you.feel.that.being.identified.as.a.person.with.a.mental.health.issue.wou
ld.hurt.your.career.)[levels(data_disorder$Do.you.feel.that.being.identified.as.a.person.with.a.me
ntal.health.issue.would.hurt.your.career.)=="Yes, it has"]<-"Yes"

levels(data_disorder$Do.you.feel.that.being.identified.as.a.person.with.a.mental.health.issue.wou
ld.hurt.your.career.)[levels(data_disorder$Do.you.feel.that.being.identified.as.a.person.with.a.me
ntal.health.issue.would.hurt.your.career.)=="Yes, I think it would"]<-"Yes"

levels(data_disorder$Do.you.feel.that.being.identified.as.a.person.with.a.mental.health.issue.wou
ld.hurt.your.career.)[levels(data_disorder$Do.you.feel.that.being.identified.as.a.person.with.a.me
ntal.health.issue.would.hurt.your.career.)=="No, it has not"]<-"No"

levels(data_disorder$Do.you.feel.that.being.identified.as.a.person.with.a.mental.health.issue.wou
ld.hurt.your.career.)[levels(data_disorder$Do.you.feel.that.being.identified.as.a.person.with.a.me
ntal.health.issue.would.hurt.your.career.)=="No, I don't think it would"]<-"No"

p2<-ggplot(data_disorder,
aes(x=Do.you.feel.that.being.identified.as.a.person.with.a.mental.health.issue.would.hurt.your.ca
reer.,fill = as.factor(sought_treatment))) +
  geom_bar(aes(group = as.factor(sought_treatment)), position = "dodge") +
guides(fill=guide_legend(title="Sought Medical Treatment?"))+ labs(x="Does mental health hurt
your career?") + theme_minimal()


p3<-  ggplot(data_disorder,aes(x=offer_benefits,fill = as.factor(sought_treatment))) +
geom_bar(aes(group = as.factor(sought_treatment)), position = "dodge") +
guides(fill=guide_legend(title="Sought Medical Treatment?"))+ labs(x="Healthcare Covergae
provided?") + theme_minimal()

p4<-ggplot(data_disorder,
aes(x=How.willing.would.you.be.to.share.with.friends.and.family.that.you.have.a.mental.illness.,
fill = as.factor(sought_treatment))) + geom_bar(aes(group = as.factor(sought_treatment)),
position = "dodge") + guides(fill=guide_legend(title="Sought Medical Treatment?"))+
labs(x="Willing to share with friends and family?") + theme_minimal() + theme(axis.text.x =
element_text(angle = 45, vjust = 0.5, hjust=1))

grid.arrange(p1,p2,p3,p4, ncol=2, top = "How many have sought Medical Treatment based on
different parameters?")
```

````
```

```{r KNN}
library(caret)
library(class)
library(forecast)

# Lets consider the respones variable for KNN classification as sought_treatment and the other variables are the predictor variables.

gender <- data3$gender
data3 <- data3[,-c(45)]
data3$gender<-(unlist(gender))


data4 <- data3
set.seed(2)
train.index <- sample(c(1:dim(data4)[1]), dim(data4)[1]*0.8)
train.df <- data4[train.index, ]
test.df <- data4[-train.index, ]

#KNN Regression
data.knn_1 <- knnreg(sought_treatment~age, train.df, k = 1)
data.knn_3 <- knnreg(sought_treatment~age, train.df, k = 3)
data.knn_7 <- knnreg(sought_treatment~age, train.df, k = 7)
data.knn_9 <- knnreg(sought_treatment~age, train.df, k = 9)
data.knn_1
data.knn.pred_1 <- predict(data.knn_1, test.df)
data.knn_3
data.knn.pred_3 <- predict(data.knn_3, test.df)
data.knn_7
data.knn.pred_7 <- predict(data.knn_7, test.df)
data.knn_9
data.knn.pred_9 <- predict(data.knn_9, test.df)

data.knn.pred_1
data.knn.pred_3
data.knn.pred_7
data.knn.pred_9

#to find accuracy
#
accuracy(test.df$sought_treatment, data.knn.pred_9)
#
cor(test.df$sought_treatment, data.knn.pred_9)
````

```
#Standardising
data4$age <- (data4$age - mean(data4$age))/sd(data4$age)
#
train.index <- sample(c(1:dim(data4)[1]), dim(data4)[1]*0.8)
train.df <- data4[train.index, ]
test.df <- data4[-train.index, ]

standardised.data.knn_1 <- knnreg(sought_treatment~age, train.df, k = 1)
standardised.data.knn_3 <- knnreg(sought_treatment~age, train.df, k = 3)
standardised.data.knn_7 <- knnreg(sought_treatment~age, train.df, k = 7)
standardised.data.knn_9 <- knnreg(sought_treatment~age, train.df, k = 9)

#
standardised.data.knn.pred_1 <- predict(standardised.data.knn_1, test.df)
standardised.data.knn.pred_3 <- predict(standardised.data.knn_3, test.df)
standardised.data.knn.pred_7 <- predict(standardised.data.knn_7, test.df)
standardised.data.knn.pred_9 <- predict(standardised.data.knn_9, test.df)


#Finding accuracy after standardizing the parameter:age
accuracy(test.df$sought_treatment, standardised.data.knn.pred_1)
accuracy(test.df$sought_treatment, standardised.data.knn.pred_3)
accuracy(test.df$sought_treatment, standardised.data.knn.pred_7)
accuracy(test.df$sought_treatment, standardised.data.knn.pred_9)


#
paste("Correlation coefficient for k=1 is:" ,cor(test.df$sought_treatment,
standardised.data.knn.pred_1))
paste("Correlation coefficient for k=3 is:" ,cor(test.df$sought_treatment,
standardised.data.knn.pred_3))
paste("Correlation coefficient for k=7 is:" ,cor(test.df$sought_treatment,
standardised.data.knn.pred_7))
paste("Correlation coefficient for k=9 is:" ,cor(test.df$sought_treatment,
standardised.data.knn.pred_9))
```

KNN - Regression with k-value(1, 3, 7, 9)

Performed KNN regression age (numerical) as predictor varible and sought_treatment as output variable.

As we cane see, RMSE value for KNN-9 has the least value with test data.

```
```{r Naive Bayes}
#Naive Bayes

# Let us consider sought_treatment as the response variable and gender, diagnosed by a medical professional, have mental health disorder, family history, care available, mental health interview, physical health interview as the predictor vaiables.

library(e1071)

colnames(data3)[48]
colnames(data3)[39]
colnames(data3)[36]
colnames(data3)[34]
colnames(data3)[4]
colnames(data3)[27]
colnames(data3)[28]
colnames(data3)[41]
data3$care_available
selected.var.naive <- c(4,27,28,34,36,39,41,48)
set.seed(1)

train.index <- sample(c(1:dim(data3)[1]), dim(data3)[1]*0.8)
naive.train.df <- data3[train.index, selected.var.naive]
naive.test.df <- data3[-train.index, selected.var.naive]


# run naive bayes
naive.mental.health <- naiveBayes(sought_treatment ~ ., data = naive.train.df)
naive.mental.health

## predict probabilities
pred.prob <- predict(naive.mental.health, newdata = naive.test.df, type = "raw")
pred.prob

library(gains)
gain <- gains(naive.test.df$sought_treatment, pred.prob[,1], groups=100)

plot(c(0,gain$cume.pct.of.total*sum(naive.test.df$sought_treatment==1))~c(0,gain$cume.obs),
    xlab="# cases", ylab="Cumulative", main="", type="l")
lines(c(0,sum(naive.test.df$sought_treatment==1))~c(0, dim(naive.test.df)[1]), lty=2)

```


```{r Logistic Regression}
```
```

```
#Logistic Regression

library(fastDummies)
library(pROC)
library(gains)

colnames(data3)[2]  #tech company
colnames(data3)[4]  #care available
colnames(data3)[7]  #anonymity portected
colnames(data3)[15] #previous employers
colnames(data3)[34] #family history
colnames(data3)[35] #mhd past
colnames(data3)[36] #have_mhd
colnames(data3)[39] #diagnosed by a medical professional
colnames(data3)[41] #sought treatment
colnames(data3)[44] #age
colnames(data3)[48] #gender


is.factor(data3$anonymity_protected)

is.factor(data3$care_available)

is.factor(data3$Have.you.had.a.mental.health.disorder.in.the.past.)

is.factor(data3$have_mhd)

is.factor(data3$Have.you.been.diagnosed.with.a.mental.health.condition.by.a.medical.profession
al.)

factor(data3$gender)

data3.logit <- data3[c(4,7,34,35,36,39,41,48)]

set.seed(2)
train.index <- sample(c(1:dim(data3.logit)[1]), dim(data3.logit)[1]*0.8)
logit.train.df <- data3.logit[train.index, ]
logit.test.df <- data3.logit[-train.index, ]

logit.mental.health <- glm(sought_treatment ~ ., data = logit.train.df, family = "binomial")
options(scipen=999)
summary(logit.mental.health)

# use predict() with type = "response" to compute predicted probabilities.
logit.mental.health.pred <- predict(logit.mental.health, logit.test.df[, -7], type = "response")
```

```
# first 5 actual and predicted records
data.frame(actual = logit.test.df$sought_treatment[1:5], predicted =
logit.mental.health.pred[1:5])


roc(logit.train.df$sought_treatment, logit.mental.health$fitted.values, plot=TRUE,
legacy.axes=TRUE, percent=TRUE, xlab="False Positive Percentage", ylab="True Postive
Percentage",col="#377eb8", lwd=2, main="ROC Curve")

#AUC 93.51%

gain <- gains(logit.test.df$sought_treatment, logit.mental.health.pred, groups=10)

# plot lift chart
plot(c(0,gain$cume.pct.of.total*sum(logit.test.df$sought_treatment))~c(0,gain$cume.obs),
    xlab="# cases", ylab="Cumulative", main="", type="l")
lines(c(0,sum(logit.test.df$sought_treatment))~c(0, dim(logit.test.df)[1]), lty=2)

# compute deciles and plot decile-wise chart
heights <- gain$mean.resp/mean(logit.test.df$sought_treatment)
midpoints <- barplot(heights, names.arg = gain$depth, ylim = c(0,9),
                xlab = "Percentile", ylab = "Mean Response", main = "Decile-wise lift chart")

# add labels to columns
text(midpoints, heights+0.5, labels=round(heights, 1), cex = 0.8)
```



```{r Regression Tree: rpart package}

library(rpart)
library(rpart.plot)

#create dummy variables

colnames(data3)[2]  #tech company
colnames(data3)[4]  #care available
colnames(data3)[7]  #anonymity portected
colnames(data3)[15] #previous employers
colnames(data3)[34] #family history
colnames(data3)[35]<-"mhd.past"
colnames(data3)[36] #have_mhd
colnames(data3)[39]<-"diagnosed.by.a.medical.professional"
colnames(data3)[41] #sought treatment
```

```
colnames(data3)[44] #age
colnames(data3)[48] #gender
colnames(data3)[colnames(data3) ==
"If.you.have.a.mental.health.issue..do.you.feel.that.it.interferes.with.your.work.when.being.treate
d.effectively."] <- "interferes.when.treated"

colnames(data3)[colnames(data3) ==
"If.you.have.a.mental.health.issue..do.you.feel.that.it.interferes.with.your.work.when.NOT.being
.treated.effectively."] <- "interferes.when.not.treated"

regression.tree.data3 <- dummy_cols(data3, select_columns =
c('anonymity_protected','care_available','family_history','mhd.past','have_mhd','diagnosed.by.a.m
edical.professional','gender'))

regression.tree.data3 <- regression.tree.data3[,c(-7,-4,-34,-35,-36,-39,-48)]

regression.tree.data3$sought_treatment <- data3$sought_treatment
regression.tree.data3$interferes.when.treated <- data3$interferes.when.treated
regression.tree.data3$interferes.when.not.treated <- data3$interferes.when.not.treated

set.seed(2)
train.index <- sample(c(1:dim(regression.tree.data3)[1]), dim(regression.tree.data3)[1]*0.8)
reg.tree.train.df <- regression.tree.data3[train.index, ]
reg.tree.test.df <- regression.tree.data3[-train.index, ]

mental.health.regression <- rpart::rpart(sought_treatment ~
anonymity_protected_Yes+anonymity_protected_No+care_available_No+care_available_Yes+`
care_available_Not sure`+`family_history_I don't
know`+family_history_Yes+family_history_No, data =reg.tree.train.df, method = "anova",
control = rpart.control(maxdepth = 3))

printcp(mental.health.regression)


summary(mental.health.regression)

prp(mental.health.regression, type = 1, extra = 1, split.font = 0.1, varlen = -8)

roc(reg.tree.train.df$sought_treatment, mental.health.regression$where, plot=TRUE,
legacy.axes=TRUE, percent=TRUE, xlab="False Positive Percentage", ylab="True Postive
Percentage",col="#377eb8", lwd=2, main = "ROC Curve")

#AUC 71.58

reg.tree.train.pred <- predict(mental.health.regression, newdata = reg.tree.train.df)
RMSE(pred = reg.tree.train.pred, obs = reg.tree.train.df$sought_treatment)
```

reg.tree.test.pred <- predict(mental.health.regression, newdata = reg.tree.test.df)
RMSE(pred = reg.tree.test.pred, obs = reg.tree.test.df$sought_treatment)

RMSE is greater for the lower for the validation dataset than the training dataset for Regression trees.

```

```{r Random Forest}

#install.packages('randomForest')
library(randomForest)


## random forest
random.forest <- randomForest(sought_treatment ~ anonymity_protected_Yes+anonymity_protected_No+care_available_No+care_available_Yes+family_history_Yes+family_history_No, data =reg.tree.train.df, ntree = 500, mtry = 4, nodesize = 5, importance = TRUE)



## variable importance plot
varImpPlot(random.forest, type = 1)

## confusion matrix
random.forest.pred <- predict(random.forest, reg.tree.test.df)

```



```{r Classfication Tree}

mental.health.classification <- rpart::rpart(sought_treatment ~ anonymity_protected_Yes+anonymity_protected_No+care_available_No+care_available_Yes+`care_available_Not sure`+`family_history_I don't know`+family_history_Yes+family_history_No, data =reg.tree.train.df, method = "class")

rpart.plot::rpart.plot(mental.health.classification, type = 4, fallen.leaves = FALSE, extra = 5)
```
```

```{r Neural Net}
library(neuralnet)
library(OneR)

neural.net<- neuralnet(sought_treatment ~
anonymity_protected_Yes+anonymity_protected_No+care_available_No+care_available_Yes+f
amily_history_Yes+family_history_No, data =reg.tree.train.df, hidden = 2, threshold = 0.5,
linear.output = T, algorithm = "rprop+",stepmax = 1e7)

neural.net$result.matrix

plot(neural.net)

compute(neural.net, reg.tree.train.df[,-c(3)])

neural.net.train.pred <- predict(neural.net, newdata = reg.tree.train.df)
RMSE(pred = neural.net.train.pred, obs = reg.tree.train.df$sought_treatment)

neural.net.test.pred <- predict(neural.net, newdata = reg.tree.test.df)
RMSE(pred = neural.net.test.pred, obs = reg.tree.test.df$sought_treatment)

prediction <- round(compute(neural.net, reg.tree.test.df)$net.result)
eval_model(prediction, reg.tree.test.df)

#The RMSE value for the test data is greater than the RMSE value of the training data, as the
data was trained on the training dataset.```
```