**CHAPTER 1**

**INTRODUCTION**

**Data mining** is the computing process of discovering patterns in large data sets involving methods at the intersection of machine learning, statistics, and database systems. An essential process where intelligent methods are applied to extract data patterns. It is an interdisciplinary subfield of computer science. The overall goal of the data mining process is to extract information from a data set and transform it into an understandable structure for further use.

The actual data mining task is the semi-automatic or automatic analysis of large quantities of data to extract previously unknown, interesting patterns such as groups of data records (cluster analysis), unusual records (anomaly detection), and dependencies (association rule mining, sequential pattern mining). This usually involves using database techniques such as spatial indices. These patterns can then be seen as a kind of summary of the input data, and may be used in further analysis or, for example, in machine learning and predictive analytics

I was tasked with exploring present innovations and industry practices in the area of Data Mining and using Survey on Mental Health in the Tech Workplace in 2014 and 2016 By Open Source Mental Illness. This knowledge would be used measure and visualize attitudes towards mental health and frequency of mental health disorders in the tech workplace along with analysis of What are the strongest predictors of mental health illness or certain attitudes towards mental health in the workplace.

**1.1 Process and Methodology**

The **knowledge discovery in databases (KDD) process** is commonly defined with the stages:

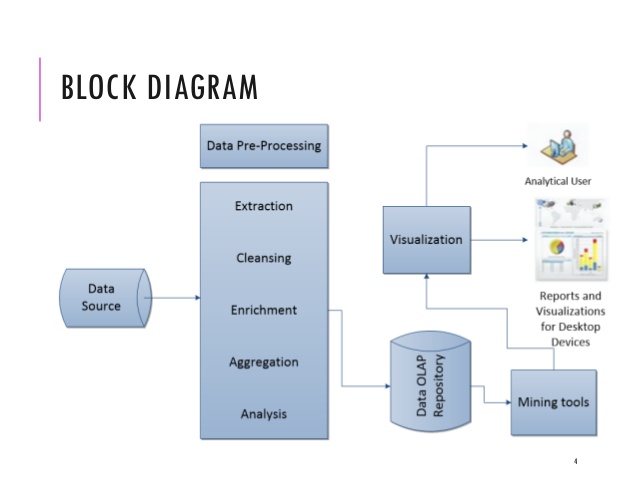
(1) Selection of appropriate Data

(2) Pre-processing of Data

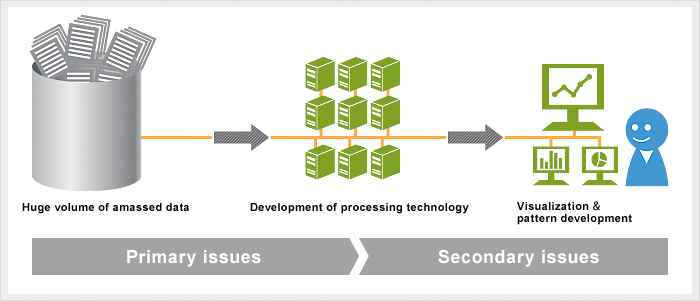
(3) Transformation and Analysis of Data

(4) Predictions using Machine Learning Tools and Libraries

(5) Result interpretation and visualisation



**Figure 1:** Block Diagram of working of Data Mining Process

****

**Figure 2:**  An overview of a typical Data mining system, where Huge Data and existing records are processed for visualization and pattern development.

**CHAPTER 2**

**Data and Business Intelligence**

**2.1 The Info-Engineering model**

**Knowledge**

**Knowledge is what we know**. Think of this as the map of the World we build inside our brains. Like a physical map, it helps us know *where*things are – but it contains more than that. It also contains our beliefs and expectations. “If I do this, I will probably get that.” Crucially, the brain links all these things together into a giant network of ideas, memories, predictions, beliefs, etc.

We **can’t currently store knowledge** in anything other than a brain, because a brain connects it all together. Everything is inter-connected in the brain. Computers are not artificial brains. They don’t **understand** what they are processing, and can’t make **independent decisions** based upon what you tell them

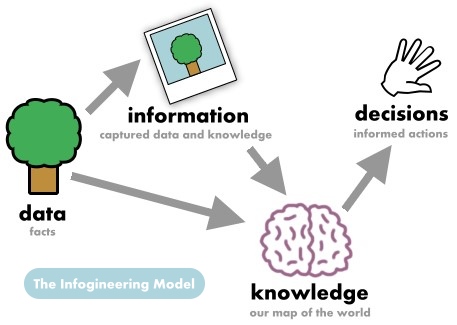
**Data**

Data is/are the **facts of the World**. For example, take yourself. You may be 5ft tall, have brown hair and blue eyes. All of this is “data”. You have brown hair whether this is written down somewhere or not. In many ways, data can be thought of as a **description of the World**. We can perceive this data with our senses, and then the brain can process this

Human beings have used data as long as we’ve existed to form knowledge of the world. Until we started using information, all we could use was data directly. If you wanted to know how tall I was, you would have to come and look at me. Our knowledge was limited by our direct experience

**Information**

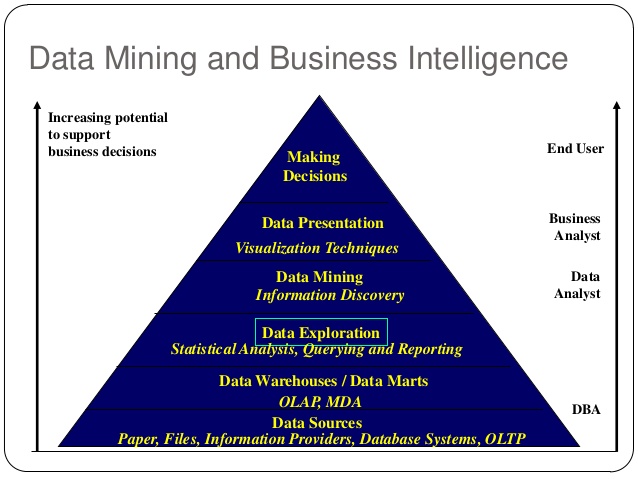
Information allows us to expand our knowledge beyond the range of our senses. We can capture data in information, then move it about so that other people can access it at different times.If I take a picture of you, the photograph is information. But what you look like is data.



**2.2 Business Intelligence**

Business Intelligence (BI) comprises the strategies and technologies used by enterprises for the data analysis of business information. BI technologies provide historical, current and predictive views of business operations. Common functions of business intelligence technologies include reporting, online analytical processing, analytics, data mining, process mining, complex event processing, business performance management, benchmarking, text mining, predictive analytics and prescriptive analytics. BI technologies can handle large amounts of structured and sometimes unstructured data to help identify, develop and otherwise create new strategic business opportunities. They aim to allow for the easy interpretation of these big data. Identifying new opportunities and implementing an effective strategy based on insights can provide businesses with a competitive market advantage and long-term stability.

Business intelligence can be used by enterprises to support a wide range of business decisions - ranging from operational to strategic. Basic operating decisions include product positioning or pricing. Strategic business decisions involve priorities, goals and directions at the broadest level. In all cases, BI is most effective when it combines data derived from the market in which a company operates (external data) with data from -company sources internal to the business such as financial and operations data (internal data).

****

**Figure 3:**  Data Mining for business intelligence.

**CHAPTER 3**

**SOFTWARE REQUIREMENT ANALYSIS**

**3.1 Purpose of the project**

The main objective of this project, Improving the work environment by predicting the importance of factor involving mental illness and providing a sustainable model for companies to measure and acquire more efficient work force in accordance to the type of work

**3.2 Scope of the Project**

A crucial aspect of a healthy and productive workplace is management’s understanding of the importance of mental health, especially in fast-paced or high-growth sectors of the economy. Given the tech industry’s rapid growth over the past few decades, I believe it would be valuable to examine the industry’s employee access to mental health resources and their understanding of these resources. I believe if we choose an analytical approach to this problem and try to map out the basic predictors of stress specifically for tech employees it would increase their working capacity and at the same time help improving the profits of the company with higher retention and loyalty of the employees.

**3.3 Perspective of the Project**

I believe this Project would be valuable to examine the industry’s employee access to mental health resources and their understanding of the resources.

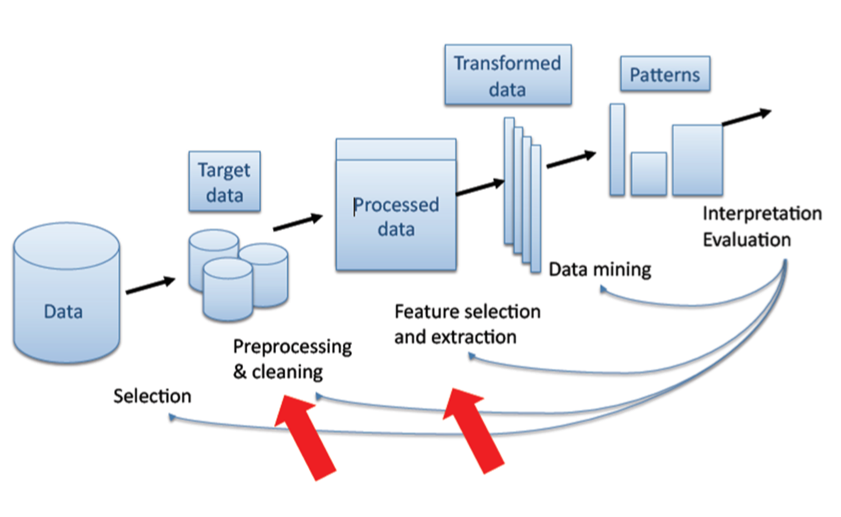
The project will also be helpful in improving the work environment by predicting the importance of factor involving mental illness and providing a sustainable model for companies to measure and acquire more efficient work force in accordance to the type of work

**3.4 Functionality and Analysis**

I break down my analysis into three areas, which align closely with the variables provided in the dataset:

* + Whether there is a need for mental health resources in the workplace,
  + Whether the employer is perceived to recognize the importance of mental health, and
  + whether the employer offers resources to employees for mental health issues.
  1. **System Architecture**

1. Data Loading and cleaning
2. Pre-processing of Data and Feature Engineering
3. Visualisation of Data
4. Analysis Of Data
5. Predictions using Machine Learning Tools and Libraries
6. Result interpretation and visualisation



**Figure 4:** Architecture of System

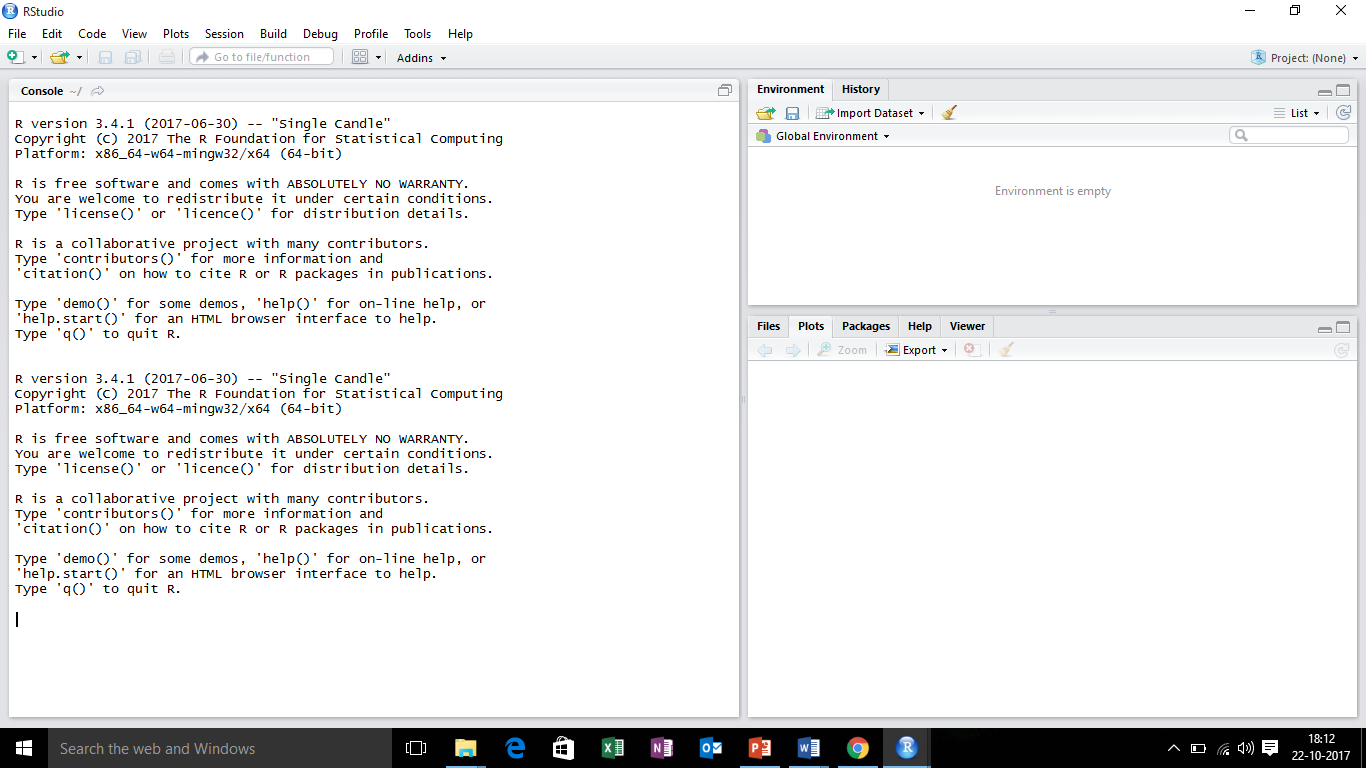
* 1. **Language Used**

R is a language and environment for statistical computing and graphics. It is a GNU project which is similar to the S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. R can be considered as a different implementation of S. There are some important differences, but much code written for S runs unaltered under R.

R provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, …) and graphical techniques, and is highly extensible. The S language is often the vehicle of choice for research in statistical methodology, and R provides an Open Source route to participation in that activity.

* 1. **Software Used**

**RStudio** is a [free and open-source](https://en.wikipedia.org/wiki/Free_and_open-source) [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) for [R](https://en.wikipedia.org/wiki/R_(programming_language)), a [programming language](https://en.wikipedia.org/wiki/Programming_language) for [statistical computing](https://en.wikipedia.org/wiki/Statistical_computing) and graphics. RStudio was founded by [JJ Allaire](https://en.wikipedia.org/wiki/Joseph_J._Allaire),  creator of the programming language [ColdFusion](https://en.wikipedia.org/wiki/ColdFusion_Markup_Language). [Hadley Wickham](https://en.wikipedia.org/wiki/Hadley_Wickham) is the Chief Scientist at RStudio



**Figure 5:** Rstudio Environment

**CHAPTER 4**

**Data pre-processing**

**4.1 Data Source**

My Major data source has been a Survey on Mental Health in the Tech Workplace in 2014 and 2016 By Open Source Mental Illness in united stare of America on various tech employees and their families’ Further data for analyzation is obtained from the ongoing 2016 Open Source Mental Illness survey.

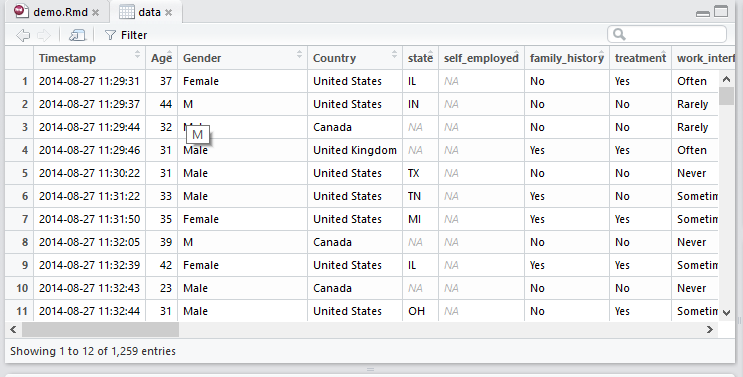
I also took some crucial data from respective heath organisation throughout the world like WHO, Pan American health association, world health assembly and Indian government websites like data.gov.in

**Benefits of Survey Data**

As this dataset includes thoughts from the workers’ perspectives, it provides a closer look into gauging the need for mental health resources and whether resources currently offered are effective. To fully answer the question I propose, feedback from the employers would also be needed to compare the resources that are perceived to be available and which resources are actually available. With the limited data, however, I am still able to at least understand which resources employees perceive to be available. Additionally, the data provide insight as to whether an employee perceives that an employer recognizes the importance of mental health.

The dataset contains following factors in data:

* Timestamp
* Age
* Gender
* Country
* state
* self-employed
* family history
* Treatment
* work interfere and 19 other factors.



**Figure 6:**  Data Snippet

**4.2 Library downloading and installation**

*# Function: installing and loading of packages*

install\_load <- **function** (packages) {

*# Start loop to determine if each package is installed*

**for**(package **in** packages){

*# If package is installed locally, load*

**if**(package %in% rownames(installed.packages()))

do.call('library', list(package))

*# If package is not installed locally, download, then load*

**else** {

install.packages(package, dependencies = TRUE)

do.call("library", list(package))

}

} }

*# Generic libraries loading*

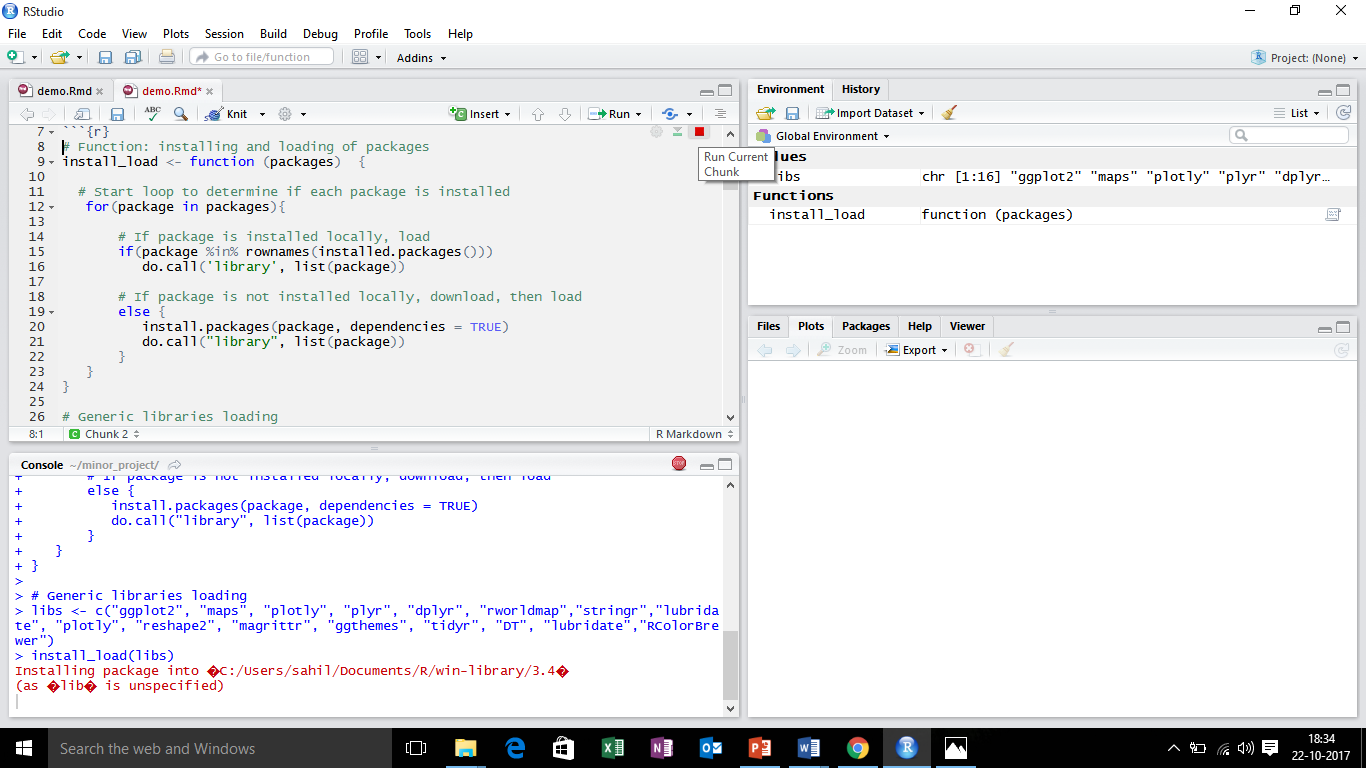
libs <- c("ggplot2", "maps", "plotly", "plyr", "dplyr", "rworldmap","stringr","lubridate", "plotly", "reshape2", "magrittr", "ggthemes", "tidyr", "DT", "lubridate","RColorBrewer")

install\_load(libs)

*# Specific methods libraries loading*

libs.methods <- c("C50", "lattice", "caret", "nnet", "e1071","Matrix", "foreach","glmnet","C50","randomForest","ipred","rpart")

install\_load(libs.methods)

****

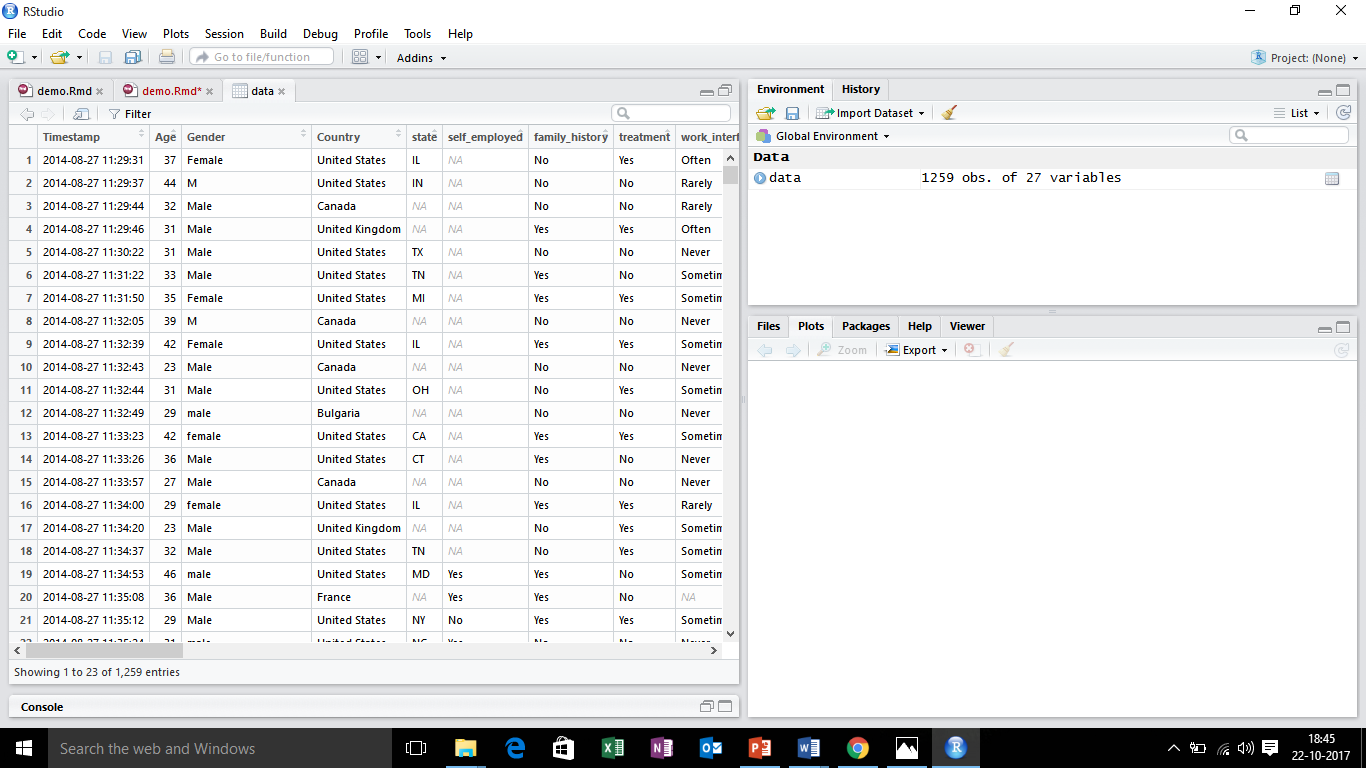
**Figure 7:** Downloading and installing the library

**4.3 Data Loading and Viewing**

*# Data loading*

data <- read.csv("../input/survey.csv")

View(data)



**Figure 8:** data for processing

# **4.4  Data cleaning and feature extraction**

*# To delete no important elements*

data <- data[ , !(names(data) %in% "state")]

data <- data[ , !(names(data) %in% "Timestamp")]

data <- data[ , !(names(data) %in% "comments")]

data <- data[ , !(names(data) %in% "self\_employed")]

*# Gender unification.*

data$Gender %<>% str\_to\_lower()

male\_str <- c("male", "m", "male-ish", "maile", "mal", "male (cis)", "make", "male ", "man","msle", "mail", "malr","cis man", "cis male")

trans\_str <- 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" )

female\_str <- c("cis female", "f", "female", "woman", "femake", "female ","cis-female/femme", "female (cis)", "femail")

data$Gender <- sapply(as.vector(data$Gender), **function**(x) **if**(x %in% male\_str) "male" **else** x )

data$Gender <- sapply(as.vector(data$Gender), **function**(x) **if**(x %in% female\_str) "female" **else** x )

data$Gender <- sapply(as.vector(data$Gender), **function**(x) **if**(x %in% trans\_str) "trans" **else** x )

data %<>% filter(Gender != "a little about you")

data %<>% filter(Gender != "guy (-ish) ^\_^")

data %<>% filter(Gender != "p")

*# Age categorization*

data$Age<-cut(data$Age, c(-Inf,20,35,65,Inf))

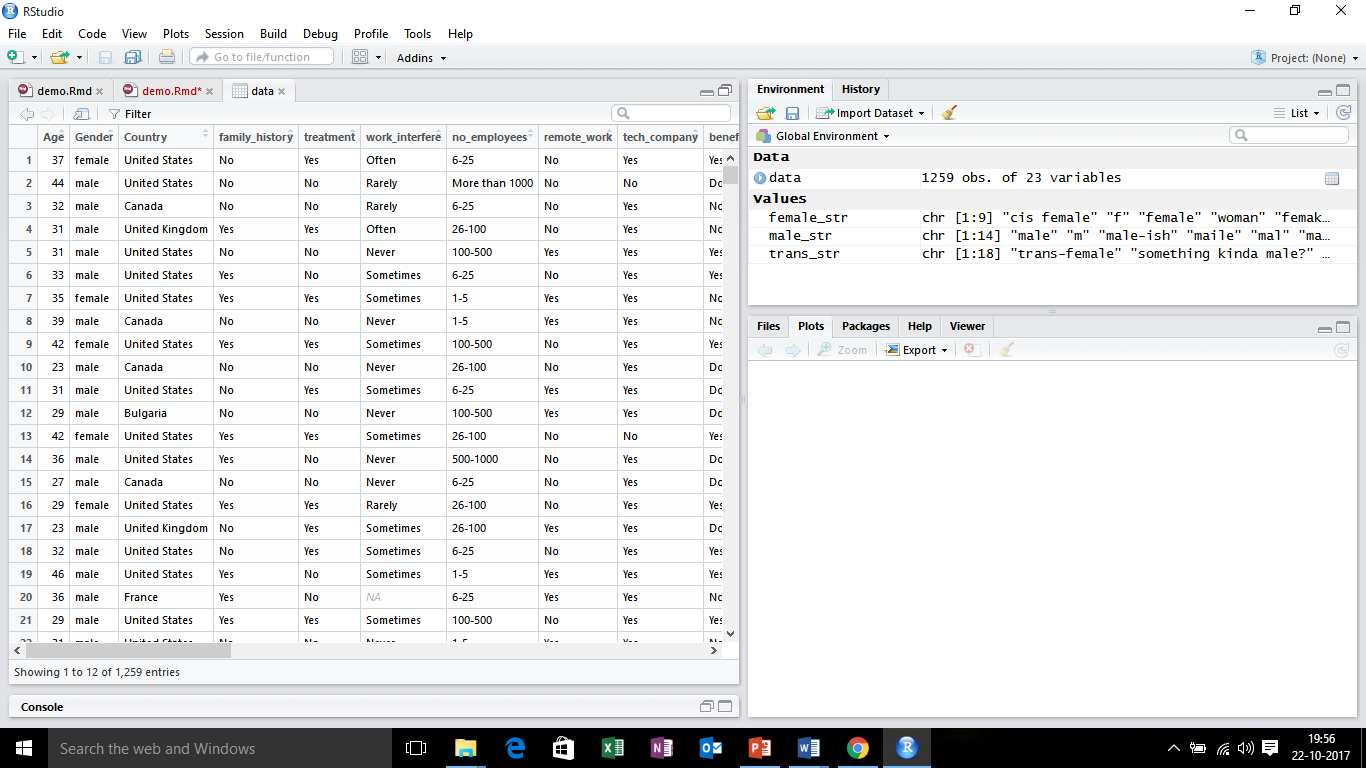
*# NA values detection and deleting the row.*

sapply(data, **function**(x) sum(is.na(x)))

data <- data[!is.na(data$work\_interfere),]

*# Saving the original data with all importants variables*

data.origin <- data



**Figure 9:** **Cleaned data after Gender unification**

**4.5 Variability comparison between categories of variables**

**Matrix of covariances**

**for**(i **in** 1:length(data)){

aux <- prop.table(table(data$treatment, data[,i]), 1)\*100

percent <- round(max(abs(aux[1,]-aux[2,])), digits = 2)

**if**(percent > 10 & percent < 99){

*# Data preparing to visualization*

aux <- prop.table(table(data$treatment, data[,i]), 1)\*100

nom <- colnames(aux)

type <- c(rep("Yes",ncol(aux)),rep("No",ncol(aux)))

val <- append(aux[1,], aux[2,])

data.aux<-data.frame(nom=nom,type=type ,val=val

*# Use of the library ggplot2 to data visualization*

g <- ggplot() + geom\_bar(data=data.aux,aes(x=nom, y=val,fill=type),stat='identity',position='dodge')+

coord\_flip() +

labs(

x = "Importance",

y = "",

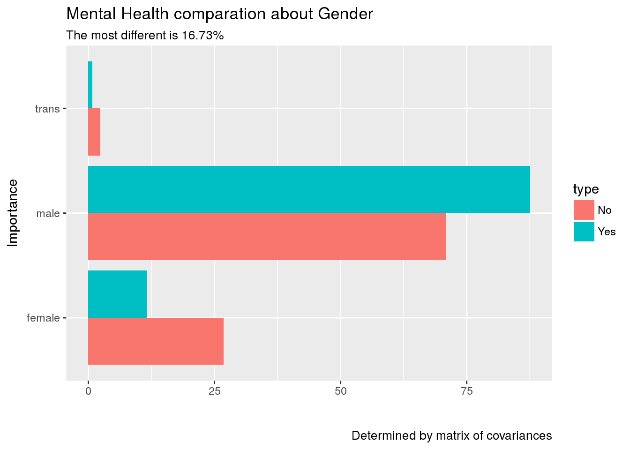
title = paste("Mental Health comparation about ", names(data[i]), sep=""),

subtitle = paste("The most different is ", percent, "%", sep=""),

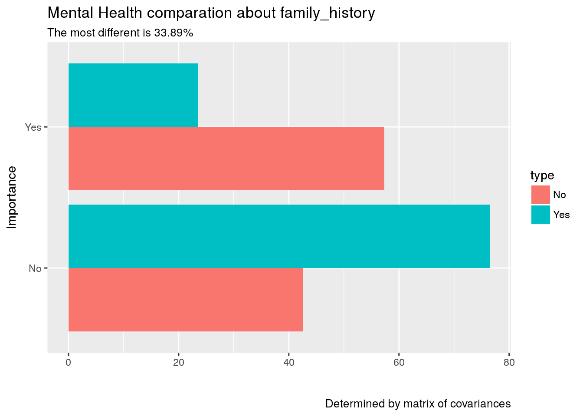
caption = "\nDetermined by matrix of covariances"

) %>% suppressWarnings()

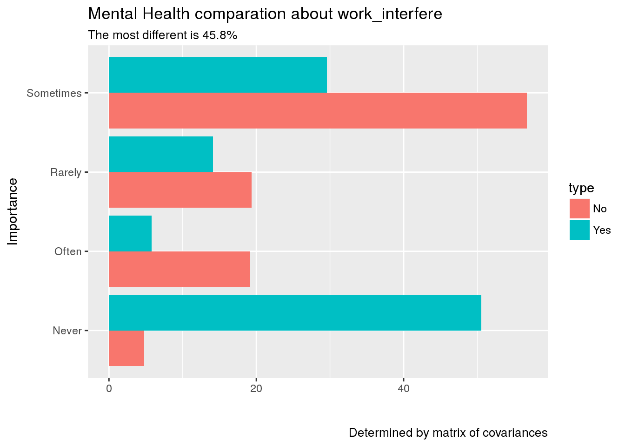
print(g)}



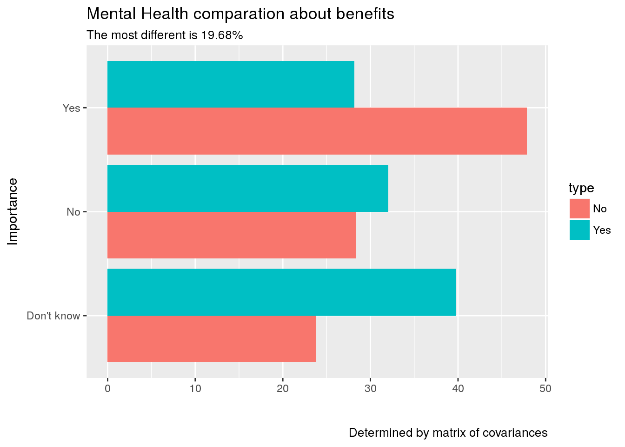
**Figure 10:** **Mental Health Comparation about Gender**

0

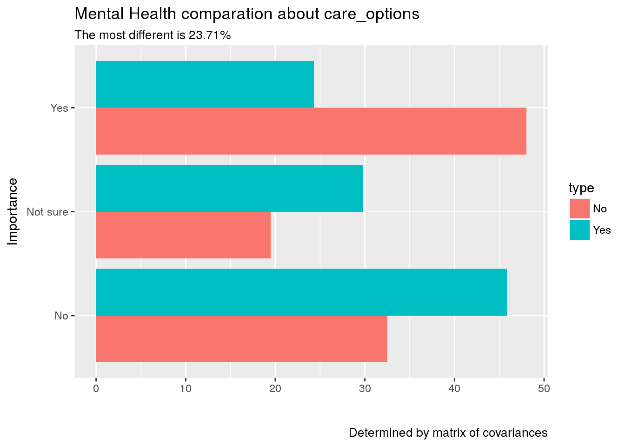
**Figure 11:** **Mental Health Comparation about Family\_history**



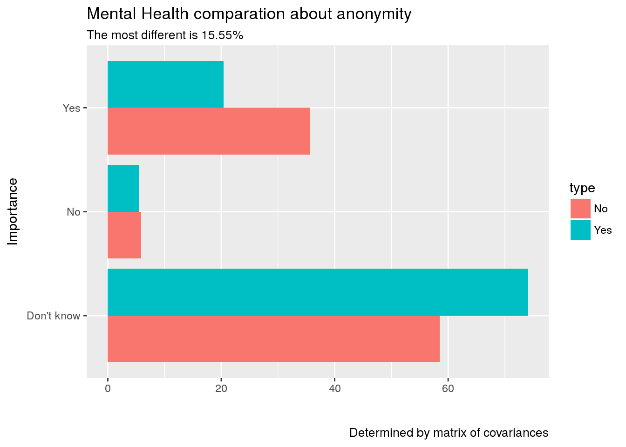
**Figure 12:** **Mental Health Comparation about Work\_interfare**



**Figure 13:** **Mental Health Comparation about benefits.**



**Figure 14:** **Mental Health Comparation about care\_options**



**Figure 15:** **Mental Health Comparation about anonamilty**

**4.6 Selection of variables with higher variability**

data <- data.frame(gender= data$Gender, family\_history= data$family\_history, work\_interfere= data$work\_interfere, benefits= data$benefits, care\_options= data$care\_options, anonymity= data$anonymity, treatment=data$treatment)

**4.7 Data training and testing**

set.seed(101)

n <- nrow(data)

data.index <- sample(1:n , size=round(n\*0.7))

train <- data[data.index,]

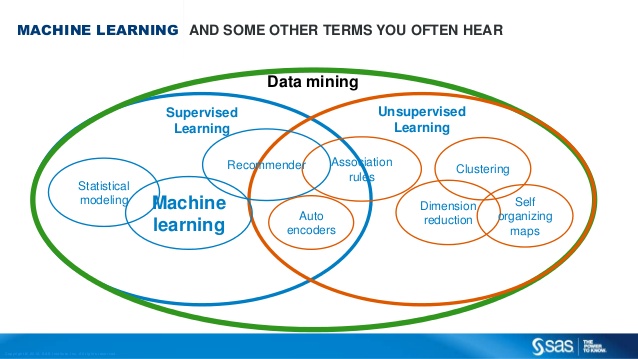
test <- data[-data.index,]

**CHAPTER 5**

**Machine Learning**

**5.1 Machine Learning Introduction**

Machine learning is closely related to (and often overlaps with) computational statistics, which also focuses on prediction-making through the use of computers. It has strong ties to mathematical optimization, which delivers methods, theory and application domains to the field. Machine learning is sometimes conflated with data mining,where the latter subfield focuses more on exploratory data analysis and is known as unsupervised learning. Machine learning can also be unsupervised and be used to learn and establish baseline behavioural profiles for various entitiesand then used to find meaningful anomalies.

Within the field of data analytics, machine learning is a method used to devise complex models and algorithms that lend themselves to prediction; in commercial use, this is known as predictive analytics. These analytical models allow researchers, data scientists, engineers, and analysts to "produce reliable, repeatable decisions and results" and uncover "hidden insights" through learning from historical relationships and trends in the data. 

**Figure 16:**  Machine Learning overview

**5.2 Supervised learning**

Supervised learning  is the [machine learning](https://en.wikipedia.org/wiki/Machine_learning) task of inferring a function from labelled training data.[[1]](https://en.wikipedia.org/wiki/Supervised_learning#cite_note-1) The [training data](https://en.wikipedia.org/wiki/Training_set) consist of a set of training examples. In supervised learning, each example is a pair consisting of an input object (typically a vector) and a desired output value (also called the supervisory signal). A supervised learning algorithm analyses the training data and produces an inferred function, which can be used for mapping new examples. An optimal scenario will allow for the algorithm to correctly determine the class labels for unseen instances.

**Types of Supervised Learning Used :**

* Stochastic Gradient Boosting
* Trees Classifier
* Neuronal Network
* Random Forest
* Bagging

**Preparing regression function for the use in other methods**

regresion <- treatment~

gender+

family\_history+

work\_interfere+

benefits+

care\_options+

anonymity

*# Saving prediction percentage of each method*

percent <- data.frame(methods=c("Trees Classifier", "Neuronal Network","Randon Forest","Bagging"), value=c(0,0,0,0))

**CHAPTER 6**

**Machine Learning Models and Predictions**

**6.1 Stochastic Gradient Boosting**

Gradient boosting is a machine learning technique for regression and classification -problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. It builds the model in a stage-wise fashion like other boosting methods do, and it generalizes them by allowing optimization of an arbitrary differentiable loss function.

The idea of gradient boosting originated in the observation by Leo Breiman that boosting can be interpreted as an optimization algorithm on a suitable cost function. Explicit regression gradient boosting algorithms were subsequently developed by Jerome H. Friedman simultaneously with the more general functional gradient boosting perspective of Llew Mason, Jonathan Baxter, Peter Bartlett and Marcus Frean. The latter two papers introduced the abstract view of boosting algorithms as iterative functional gradient descent algorithms. That is, algorithms that optimize a cost function over function space by iteratively choosing a function (weak hypothesis) that points in the negative gradient direction. This functional gradient view of boosting has led to the development of boosting algorithms in many areas of machine learning and statistics beyond regression and classification.

*# Preparation of training scheme*

control <- trainControl(method = "repeatedcv", number = 3, repeats = 3)

set.seed(27)

regresion.origin <- treatment~.

caret\_gbm <- caret::train(regresion.origin,

data = data.origin,

method = "gbm",

preProcess = NULL,

trControl = control)

*# Importance estimation of variable category*

importance <- varImp(caret\_gbm, scale=TRUE)

*# Data preparing to visualization*

data.imp <- data.frame(Overall=importance$importance$Overall, group= rownames(importance$importance))

data.imp <- data.imp[with(data.imp, order(-data.imp$Overall)), ] *# Orden inverso*

aux<- data.imp[1:10,]

data.imp<- data.frame(Overall=aux$Overall, group= aux$group)

data.imp1 <- data.imp

data.imp0 <- data.frame(Overall=0, group= data.imp$group)

data.imp <- rbind(data.imp,data.imp0)

*# Use of the library ggplot2 to data visualization*

ggplot() +

geom\_point(data = data.imp1, aes(x = Overall, y = group, color = group), size = 4) +

geom\_path(data = data.imp, aes(x = Overall, y = group, color = group, group = group), size = 2) +

theme(legend.position = "none",

axis.text.x = element\_text(angle = 0, vjust = 0.5, hjust = 0.5)) +

labs(

x = "Importance",

y = "",

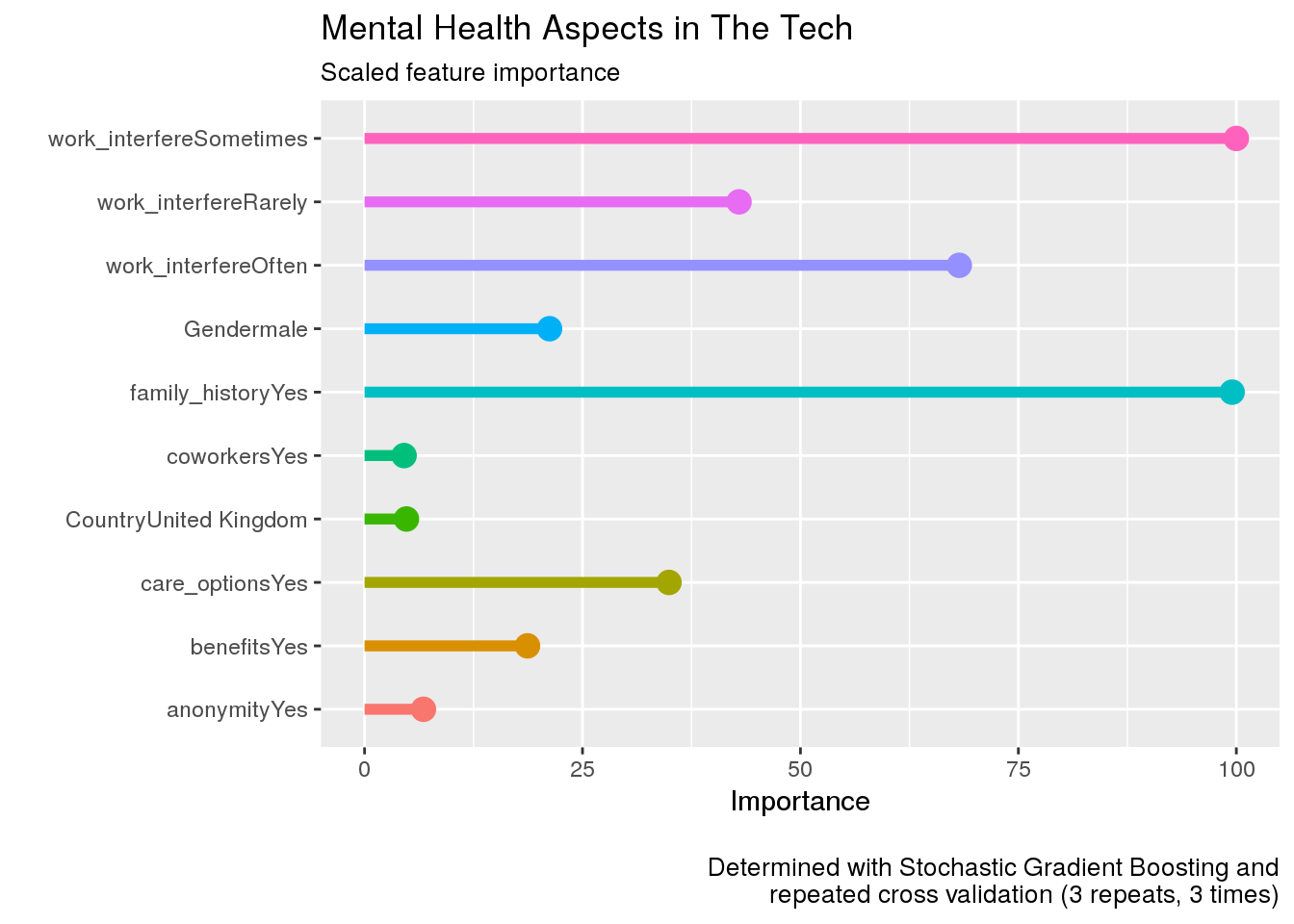
title = "Mental Health Aspects in The Tech",

subtitle = "Scaled feature importance",

caption = "\nDetermined with Stochastic Gradient Boosting and

repeated cross validation (3 repeats, 3 times)"

) %>% suppressWarnings()



**Figure 17:** **Stochastic Gradient Boosting**

**6.2 Prediction of mental health**

## TREES CLASSIFIER C5.0

*# Executing model C5.0*

model <- C5.0( treatment ~ . , data = train)

*# Prediction*

prediction <- predict(model,newdata=test)

*# Confussion matrix*

( mc <- table(prediction, test$treatment) )

##

## prediction No Yes

## No 64 18

## Yes 47 169

*# Succesful percentage of clasification*

( percent$value[1] <- sum(diag(mc)) / sum(mc) \* 100 )

## [1] 78.18792

## NEURONAL NETWORK

*# Calculation of size and decay parameters*

*# size: number of intermediate hidden unit*

*# decay: avoiding overfitting*

parameter <- train( treatment ~ . , data=train, method="nnet", trace=F)

size <- parameter$bestTune$size

decay <- parameter$bestTune$decay

*#parameter$bestTune*

*# Neuronal Network model*

model <- nnet(treatment ~ ., size=size, decay=decay, trace=F, data=train)

*# Prediction. Creating a dataframe with the probabilities*

predict <- data.frame(predict(model, test), treatment=predict(model,test, type="class"))

*# Confussion matrix*

( mc <- table(predict$treatment,test$treatment, dnn = c("Asignado","Real")) )

## Real

## Asignado No Yes

## No 71 30

## Yes 40 157

*# Succesful percentage of clasification*

( percent$value[2] <- sum(diag(mc)) / nrow(test) \* 100 )

## [1] 76.51007

## RANDOM FOREST

*# Random Forest model*

model <- randomForest(treatment ~ ., data=train)

*# Prediction. Creating a dataframe with the probabilities*

predict <- predict(model, test)

*# Confussion matrix*

( mc <- with(test, table(predict, treatment)) )

## treatment

## predict No Yes

## No 68 22

## Yes 43 165

*# Succesful percentage of clasification*

( percent$value[3] <- sum(diag(mc)) / sum(mc) \* 100 )

## [1] 78.18792

## BAGGING

*# Bagging model*

model <- bagging(treatment ~ ., data=train)

*# Prediction. Creating a dataframe with the probabilities*

predict <- predict(model, test)

*# Confussion matrix*

( mc <- with(test, table(predict, treatment)) )

## treatment

## predict No Yes

## No 68 28

## Yes 43 159

*# Succesful percentage of clasification*

( percent$value[4] <- sum(diag(mc)) / sum(mc) \* 100 )

## [1] 76.1745

## 6.3 Predicting capacity comparison of methods

percent$methods <- paste(percent$methods, " - " , round(percent$value,digits = 2) , "%" , sep = "")

visualize <- data.frame(valor=percent$value, group= percent$methods)

visualize2 <- rbind(visualize,data.frame(valor=50, group= visualize$group))

ggplot() +

geom\_point(data = visualize, aes(x = valor, y = group, color = group), size = 4) +

geom\_path(data = visualize2, aes(x = valor, y = group, color = group, group = group), size = 2) +

theme(legend.position = "none",

axis.text.x = element\_text(angle = 0, vjust = 0.5, hjust = 0.5)) +

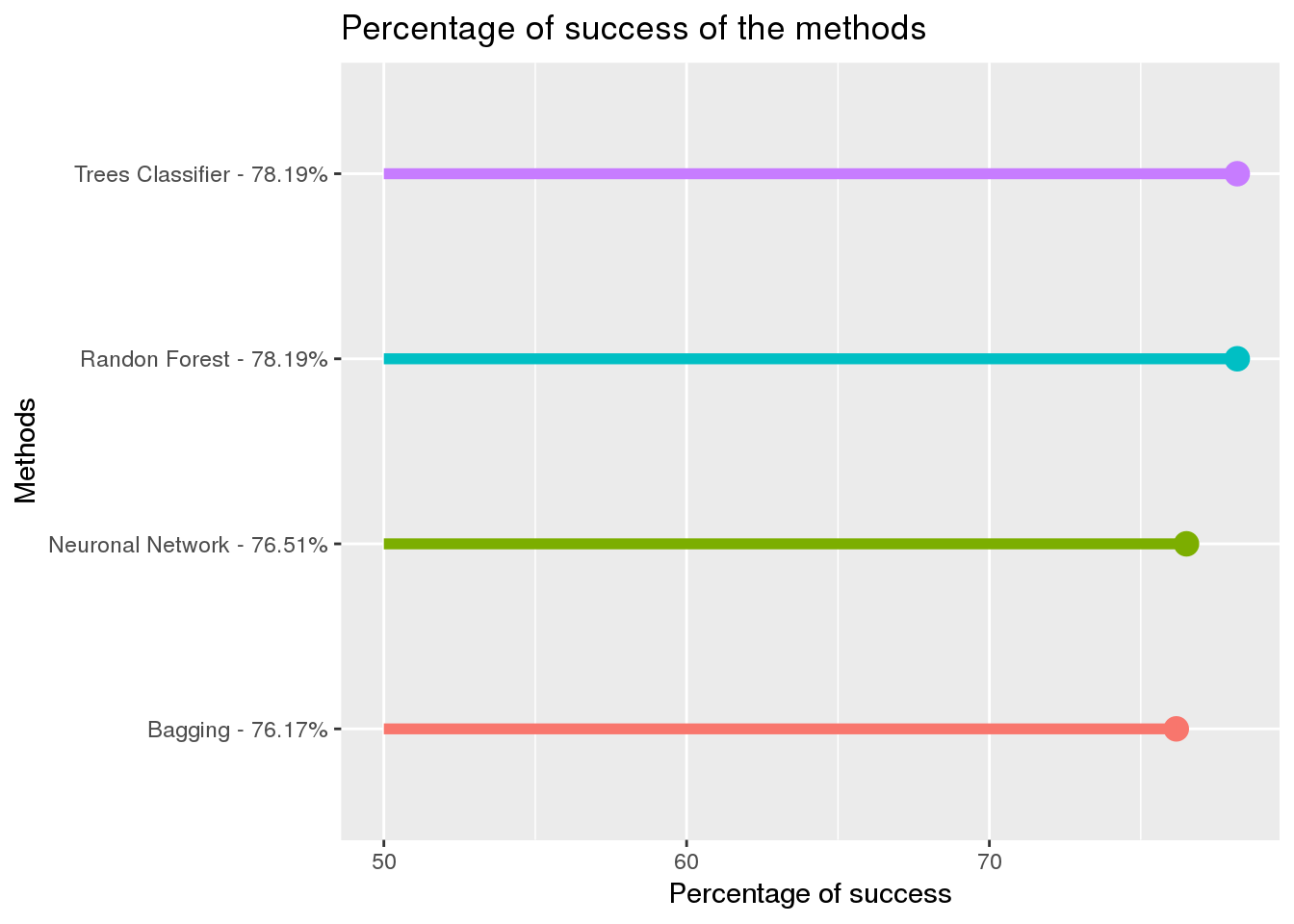
labs(

x = "Percentage of success",

y = "Methods",

title = "Percentage of success of the methods"

)



**Figure 18:** **Percent of Scucess of all the meathods**

**CHAPTER 7**

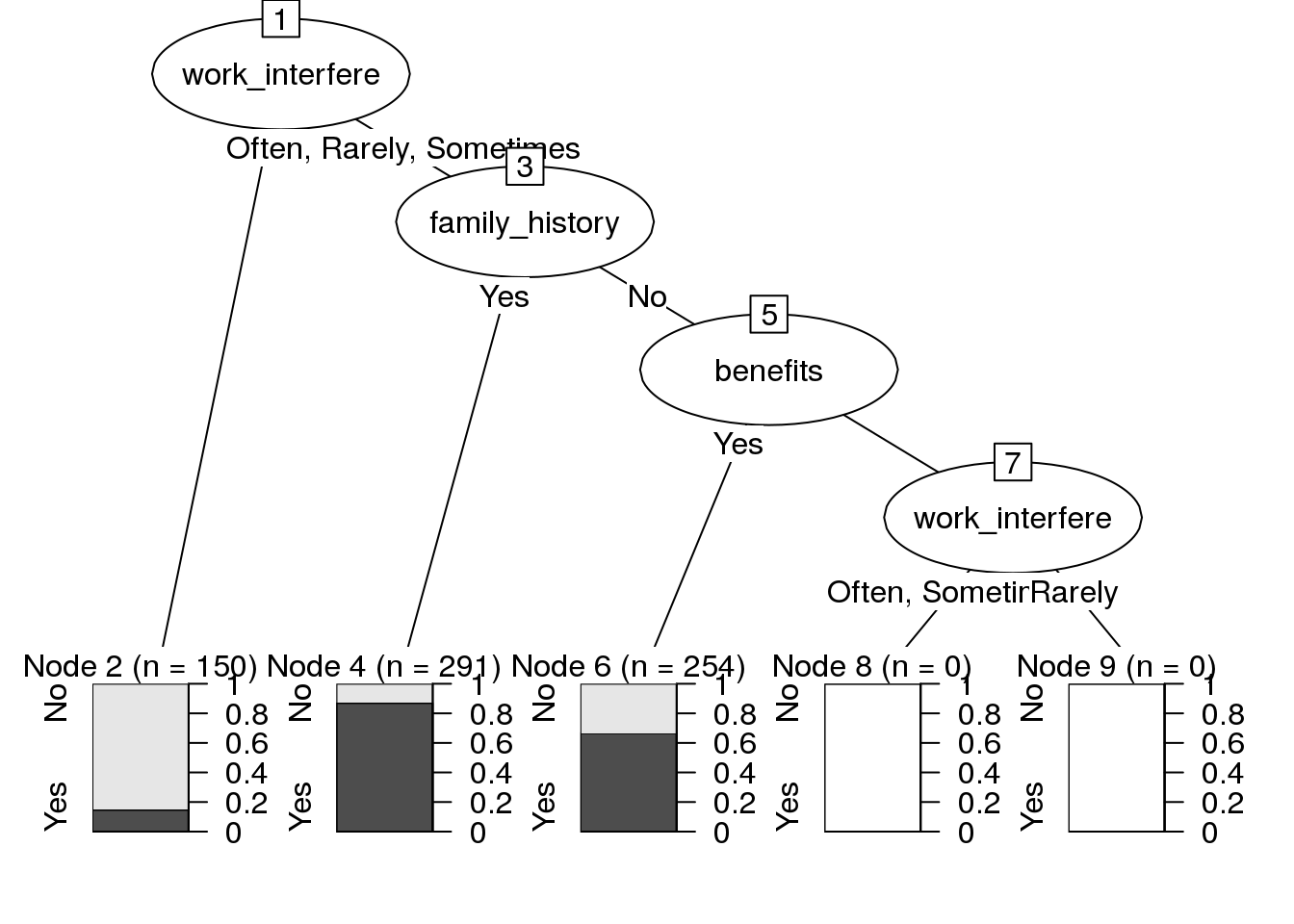
**Result interpretation**

## 7.1 TREES CLASSIFIER C5.0

*# Executing model C5.0*

model <- C5.0( treatment ~ . , data = train)

plot(model)



**Figure 19:** **Tree Classifier Model.**

## RECURSIVE PARTITIONING

mental\_all\_var <- treatment ~

gender+

family\_history+

work\_interfere+

benefits+

care\_options+

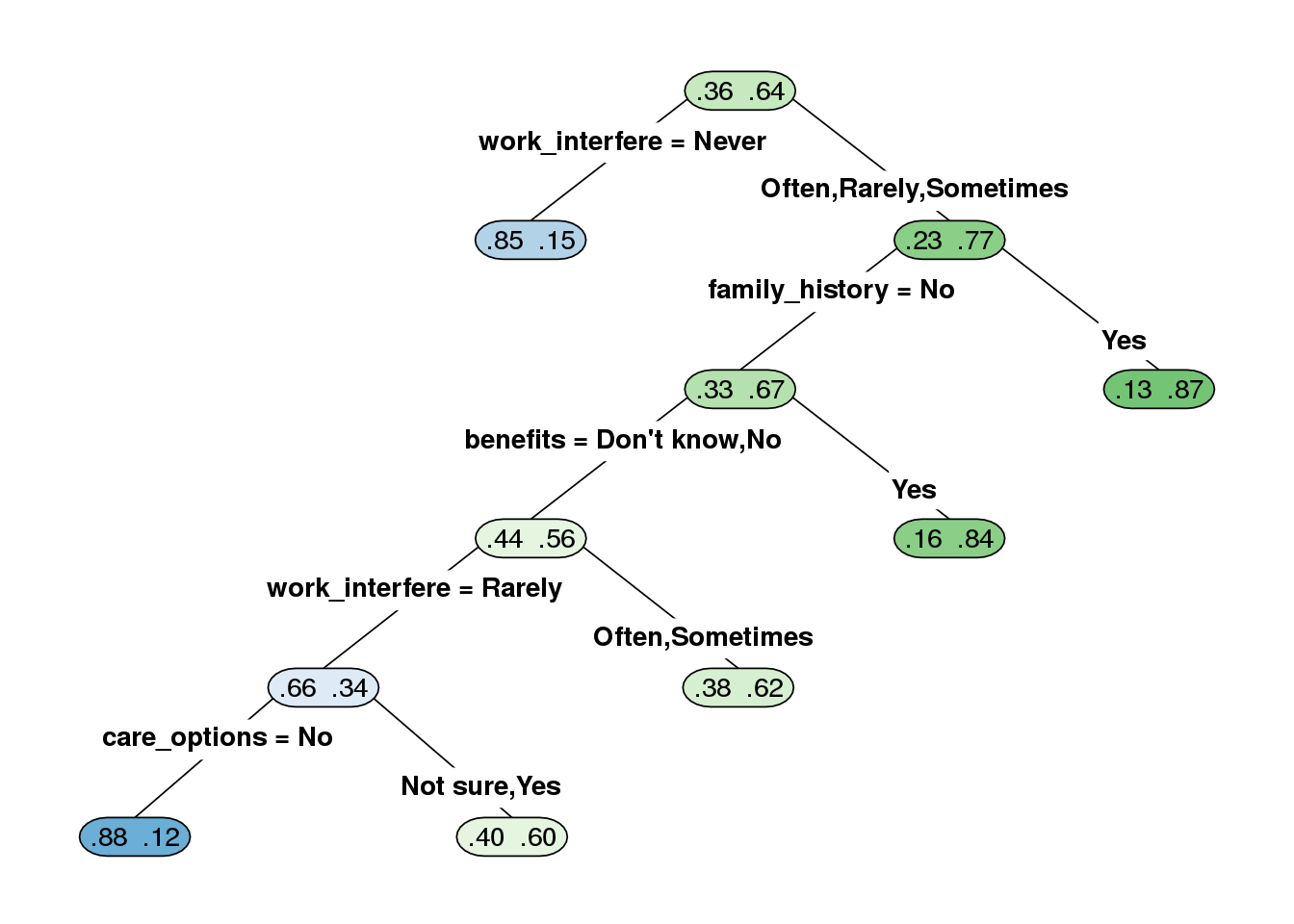
anonymity

rpart\_all\_variables <- rpart(mental\_all\_var,

data=train,

method = "class")

rpart.plot::rpart.plot(rpart\_all\_variables, type = 4, fallen.leaves = FALSE, extra = 5)



**Figure 20:** **Recursive Partioning.**

# **7.2 Frequency of mental health showed by countries**

data <- data.origin

*# Data preparing*

country <- data$Country

ill <- data$treatment

data.aux <- data.frame(country,ill)

*# Data frame definition*

data.result <- ddply(data.aux,.(country,ill), nrow)

*# Frecuency accumulation of treated people*

frec <- data.frame(id = data.result$country

, value = data.result$V1)

frec <- mutate(group\_by(frec,id), cumsum=cumsum(value))

data.all <- data.frame(id=frec$id,num=frec$cumsum)

data.all <- data.all[with(data.all, order(-data.all$num)), ]

data.all <- data.all[!duplicated(data.all$id),]

data.all <- data.all[with(data.all, order(data.all$id)), ]

*# Putting treated people in negative way*

**for** (i **in** 1:length(data.result$country)){

**if**(data.result$ill[i] =="Yes"){

data.result$V1[i] <- data.result$V1[i]\*-1

}

}

*# Frecuency accumulation of treated people*

frec <- data.frame(id = data.result$country

, value = data.result$V1)

frec <- mutate(group\_by(frec,id), cumsum=cumsum(value))

data.aux <- data.frame(id=frec$id,num=frec$cumsum)

data.aux <- data.aux[with(data.aux, order(-data.aux$num)), ]

data.aux <- data.aux[!duplicated(data.aux$id),]

data.aux <- data.aux[with(data.aux, order(data.aux$id)), ]

*# Making relative treated people*

data <- data.frame(id=data.all$id)

data$id <- data.aux$id

data$num <- (data.aux$num / data.all$num )

data$category <- data$num

*# Categorization of treated people by geographic location*

data$category <- cut(data$num, breaks=c(-Inf, -0.75, -0.5, -0.35, 0, 0.35, 0.5, 0.75, Inf))

levels(data$category) <- c("-70%","-50%","-30%","-15%","+15%","+30%","+50%","+70%")

d <- data.frame( country=data$id, value=data$cate)

n <- joinCountryData2Map(d, joinCode="NAME", nameJoinColumn="country")

## 39 codes from your data successfully matched countries in the map

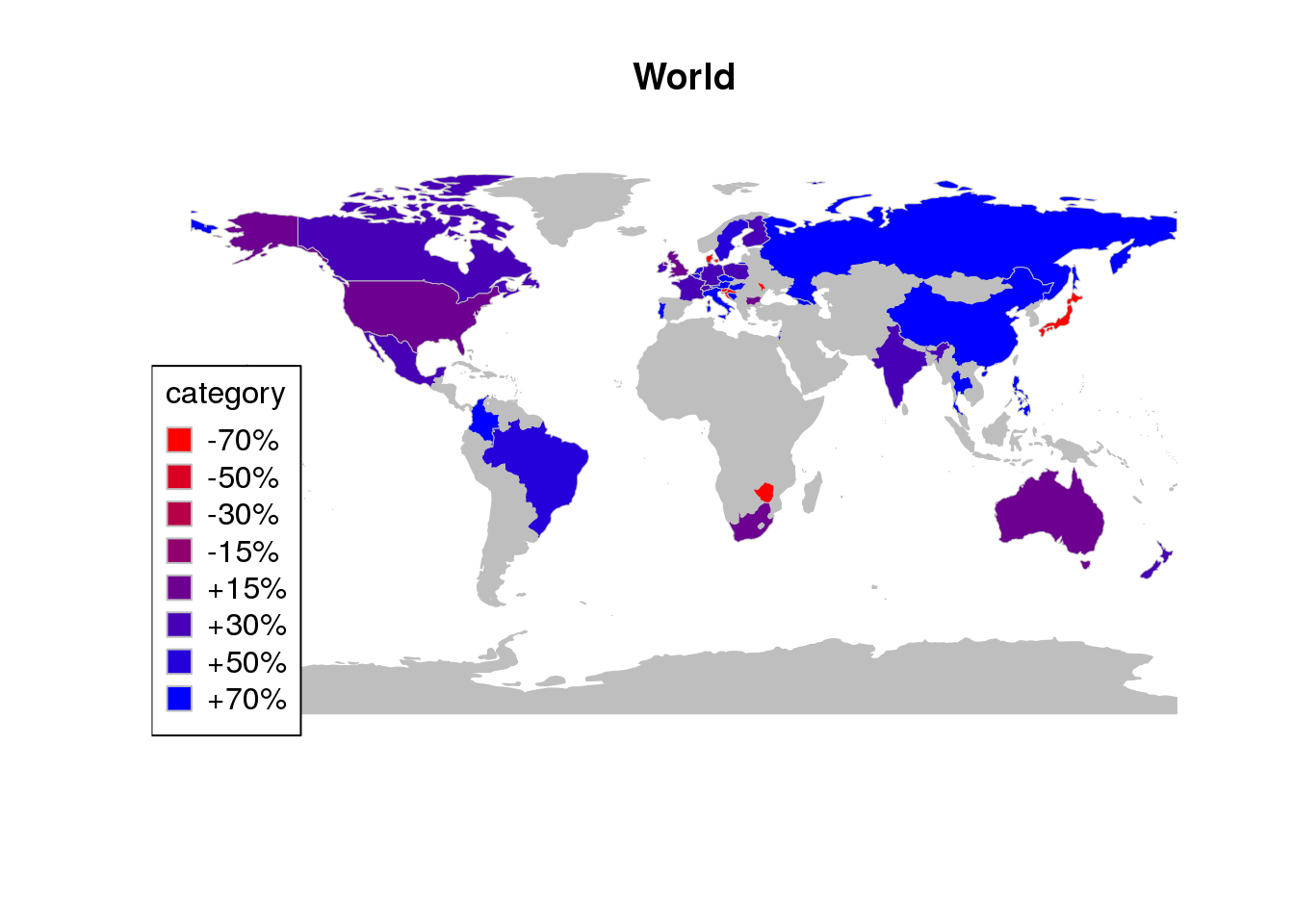
## 0 codes from your data failed to match with a country code in the map

## 204 codes from the map weren't represented in your data

*# Visualization of treated people by geographic location*

mapCountryData(n, nameColumnToPlot="value", mapTitle="World" ,catMethod="categorical",

colourPalette=c('red','blue'),missingCountryCol="grey", aspect =0)

-

**Figure 21:** **Frequency of mental health showed by countries**

**CHAPTER 8**

**FUTURE SCOPE**

A crucial aspect of a healthy and productive workplace is management’s understanding of the importance of mental health, especially in fast-paced or high-growth sectors of the economy. Given the tech industry’s rapid growth over the past few decades, I believe it would be valuable to examine the industry’s employee access to mental health resources and their understanding of these resources.

Improving the work environment by predicting the importance of factor involving mental illness and providing a sustainable model for companies to measure and acquire more efficient work force in accordance to the type of work

**REFERENCES AND BIBLIOGRAPHY**

**Books**

Bezdek, J. C., & Pal, S. K. (1992). *Fuzzy models for pattern recognition: Methods that search for structures in data*. New York: IEEE Press

Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., & Uthurusamy, R. (Eds.). (1996). *Advances in knowledge discovery and data mining.* AAAI/MIT Press.

Han, J., & Kamber, M. (2000). *Data mining: Concepts and techniques*: Morgan Kaufmann.

Hastie, T., Tibshirani, R., & Friedman, J. H. (2001). *The elements of statistical learning: Data mining, inference, and prediction:*New York: Springer.

Jain, A. K., & Dubes, R. C. (1988). *Algorithms for clustering data*. New Jersey: Prentice Hall.

Jensen, F. V. (1996). *An introduction to bayesian networks*. London: University College London Press.

Kaufman, L., & Rousseeuw, P. J. (1990). *Finding groups in data: An introduction to cluster analysis*. New York: John Wiley.

Michie, D., Spiegelhalter, D. J., & Taylor, C. C. (1994). *Machine learning, neural and statistical classification*: Ellis Horwood.

**Articles**

Agrawal, R., Imieliski, T., & Swami, A. (1993). Mining association rules between sets of items in large databases, *Proceedings of the 1993 ACM SIGMOD international conference on Management of data*. Washington, D.C., United States: ACM Press.

Bauer, E., & Kohavi, R. (1999). An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. *Machine Learning, 36*(1 - 2), 105.

Bittner, T. (2001). Rough sets in spatio-temporal data mining, *Proceedings of the First International Workshop on Temporal, Spatial, and Spatio-Temporal Data Mining-Revised Papers*: Springer-Verlag.