Loading the orginal dataset

```r
# Set the file path for the CSV file
file_path2 <- "C:/Users/jivko/Documents/Data Analytics, Big Data, and Predictive Analytics/Pe
rsonal Project/Sentiment Analysis for Mental Health/Combined Data.csv"


# Read the CSV file into a dataframe
sentiment_analysis <- read.csv(file_path2, header = TRUE)
```

Removing redundant X column

```r
sentiment_analysis_use <- sentiment_analysis[, !names(sentiment_analysis) %in% c("X")]
```

Matching each status count to median (3888)

```r
# Load the libraries
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.4.2
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.4.2
```

```
## Loading required package: ggplot2
```

```
## Warning: package 'ggplot2' was built under R version 4.4.2
```

```
## Loading required package: lattice
```

```r
# Assuming your dataset is called sentiment_analysis
# Get the distribution of the classes in the sentiment_analysis dataset
class_counts <- table(sentiment_analysis_use$status)

# Initialize an empty list to hold the balanced dataset
balanced_data <- list()

# Loop through each class
for (class in names(class_counts)) {
  # Subset the data for the current class
  class_data <- sentiment_analysis_use %>% filter(status == class)

  # If the class has fewer than 3888 samples, oversample
  if (nrow(class_data) < 3888) {
    # Oversample with replacement
    class_data <- class_data[sample(1:nrow(class_data), 3888, replace = TRUE), ]
  }

  # If the class has more than 3888 samples, undersample
  else if (nrow(class_data) > 3888) {
    # Undersample to 3888 samples
    class_data <- class_data[sample(1:nrow(class_data), 3888), ]
  }

  # Add the balanced class data to the list
  balanced_data[[class]] <- class_data
}

# Combine the balanced data
balanced_data <- do.call(rbind, balanced_data)

# Check the distribution of the balanced data
balanced_class_counts <- table(balanced_data$status)
print(balanced_class_counts)
```

```
##
##              Anxiety               Bipolar            Depression
##                 3888                  3888                  3888
##               Normal  Personality disorder                Stress
##                 3888                  3888                  3888
##             Suicidal
##                 3888
```

Setting a fixed sample size per class (15% of 3888)

```
fixed_sample_size <- 583  # Round down to ensure consistency across classes

# Perform stratified sampling
sampled_balanced_data <- do.call(rbind, lapply(split(balanced_data, balanced_data$status), fu
nction(class_data) {
  class_data[sample(1:nrow(class_data), fixed_sample_size), ]
}))

# Check the new distribution
table(sampled_balanced_data$status)
```

```
##
##              Anxiety              Bipolar            Depression
##                  583                  583                   583
##               Normal Personality disorder                Stress
##                  583                  583                   583
##             Suicidal
##                  583
```

Preprocessing text data

```
# Load required libraries
library(textstem)
```

```
## Warning: package 'textstem' was built under R version 4.4.2
```

```
## Loading required package: koRpus.lang.en
```

```
## Warning: package 'koRpus.lang.en' was built under R version 4.4.2
```

```
## Loading required package: koRpus
```

```
## Warning: package 'koRpus' was built under R version 4.4.2
```

```
## Loading required package: sylly
```

```
## Warning: package 'sylly' was built under R version 4.4.2
```

```
## For information on available language packages for 'koRpus', run
##
##    available.koRpus.lang()
##
## and see ?install.koRpus.lang()
```

```r
library(tm)
```

```
## Warning: package 'tm' was built under R version 4.4.2
```

```
## Loading required package: NLP
```

```
##
## Attaching package: 'NLP'
```

```
## The following object is masked from 'package:ggplot2':
##
##     annotate
```

```
##
## Attaching package: 'tm'
```

```
## The following object is masked from 'package:koRpus':
##
##     readTagged
```

```r
library(textclean)  # Ensure this library is loaded for replace_contraction()
```

```
## Warning: package 'textclean' was built under R version 4.4.2
```

```r
library(quanteda)   # For tokenization and n-grams
```

```
## Warning: package 'quanteda' was built under R version 4.4.2
```

```
## Package version: 4.1.0
## Unicode version: 15.1
## ICU version: 74.1
```

```
## Parallel computing: 8 of 8 threads used.
```

```
## See https://quanteda.io for tutorials and examples.
```

```
##
## Attaching package: 'quanteda'
```

```
## The following object is masked from 'package:tm':
##
##     stopwords
```

```
## The following objects are masked from 'package:NLP':
##
##     meta, meta<-
```

```
## The following objects are masked from 'package:koRpus':
##
##     tokens, types
```

```r
# Replace stemming with lemmatization
corpus <- Corpus(VectorSource(sampled_balanced_data$statement))

# Apply preprocessing steps
corpus <- tm_map(corpus, content_transformer(tolower))        # Convert to lowercase
```

```
## Warning in tm_map.SimpleCorpus(corpus, content_transformer(tolower)):
## transformation drops documents
```

```r
corpus <- tm_map(corpus, content_transformer(replace_contraction))  # Correctly use textclean
's function
```

```
## Warning in tm_map.SimpleCorpus(corpus,
## content_transformer(replace_contraction)): transformation drops documents
```

```r
corpus <- tm_map(corpus, removePunctuation)                    # Remove punctuation
```

```
## Warning in tm_map.SimpleCorpus(corpus, removePunctuation): transformation drops
## documents
```

```r
corpus <- tm_map(corpus, removeNumbers)                        # Remove numbers
```

```
## Warning in tm_map.SimpleCorpus(corpus, removeNumbers): transformation drops
## documents
```

```r
corpus <- tm_map(corpus, stripWhitespace)                      # Remove extra whitespaces
```

```
## Warning in tm_map.SimpleCorpus(corpus, stripWhitespace): transformation drops
## documents
```

```r
# Remove non-alphanumeric characters (optional)
corpus <- tm_map(corpus, content_transformer(function(x) gsub("[^[:alnum:] ]", "", x)))
```

```
## Warning in tm_map.SimpleCorpus(corpus, content_transformer(function(x)
## gsub("[^[:alnum:] ]", : transformation drops documents
```

```r
# Remove URLs (optional)
corpus <- tm_map(corpus, content_transformer(function(x) gsub("http[s]?://\\S+", "", x)))
```

```
## Warning in tm_map.SimpleCorpus(corpus, content_transformer(function(x)
## gsub("http[s]?://\\S+", : transformation drops documents
```

```r
# Remove mentions and hashtags (optional, useful for social media data)
corpus <- tm_map(corpus, content_transformer(function(x) gsub("@\\S+|#\\S+", "", x)))
```

```
## Warning in tm_map.SimpleCorpus(corpus, content_transformer(function(x)
## gsub("@\\S+|#\\S+", : transformation drops documents
```

```r
# Remove stopwords
corpus <- tm_map(corpus, removeWords, stopwords("en"))
```

```
## Warning in tm_map.SimpleCorpus(corpus, removeWords, stopwords("en")):
## transformation drops documents
```

```r
# Correct spelling mistakes (optional, if needed)
# Use textclean or hunspell for spell correction if applicable

# Apply lemmatization
cleaned_statements <- data.frame(statement = lemmatize_strings(sapply(corpus, as.character)),
                                 status = sampled_balanced_data$status)

# Remove short texts (optional)
cleaned_statements <- cleaned_statements[nchar(as.character(cleaned_statements$statement)) >
3, ]

# View a sample of the cleaned data
head(cleaned_statements)
```

##
statement
1
hand fall asleep past day ive wake night hand tingly asleep shake finger wake try fall asleep can feel light tingle hand hasnt throughout day less notice wonder anyone else symptom like
2 double ibuprofen aspirinacetaminophen good start say history headache near front side he adI sure exactly cause I consider see doctor point post yesterday one horrible headache take mg ibuprofen hour late mg acetaminophen aspirinalso mg caffeine doubt thatd effect ampxb upon wake today headache yesterday take medicine take every morning much list interfere either als o take mg acetaminophen aspirin minute late headache still horribleit bad forget take medicin e take mg ibuprofen ampxb generally try double upor triple acetaminophen aspirin one pill gen erally limit mg ibuprofen every hour less case like yesterday today headache generally bad go mg possibly even mg original mg ibuprofen work hour really try avoid I stomach issue take muc h go away take normal dose even just take mg ibuprofen try take aspirin time forget just much pain ampxb likely migraine combine sinus bad weather outside can cause headache luckily medic ine help quite bite know can side effect take together oftenespecially dose take ampxb point post want make sure I immediate danger anything take mg acetaminophen aspirin mg ibuprofen on cenot ask future plan take much just want make sure I safe ampxb tldr forget take mg acetamin ophen aspirin pillssame pain killer headache just wake take mg ibuprofen minute late just wan t make sure I risk ampxb plan double like future mistake much pain combine tire just wake
3
can tell shortness breath chest discomfort anxiety something else constantly feel like diffic ult breathe sometimes go away come back cant go doctor know theyll just tell tell two year ag o im finetheres nothing wrong issue get annoy
4
tongue bump go go brush tooth realize plaque buildup back tongue go grow brush realize bump r eddish color tongue back tongue make stupid choice look now im really worry every outcome see m horrific legitimatly diagnose bad anxiety cant fall asleep friday morning post cant fall as leep worry ehat bump may say treat body hygene try take really good care bejng unable afford dentist appointment go last year ago get temporary cap put emergency root canal actually use tooth eat start hurt eat around time th birthday long history problematic mouth due past mist ake brush every night bed drink plently waterstay away sugary drink do know bump magically ap pear tongue serious csn sk come
5
can believe doctor tell be super frustrate today cardiologist appointment bc palpitation some times heart race randomly acid reflux cardiologist blood pressurepulse check rule pot say fin e front ekg come back great say heart excellent shape even ask order take home monitor just p eace mind okay echocardiogram do year ago different cardiologist everything come back great n ow get home doctor anxious brain get plot now worry order another echocardiogram miss somethi ng rational brain believe trust doctor much anxiety part brain just want create worry reason can find can anyone give advice insight even medical reason need 2 echocardiogram year late a ssume detect defect probably birth 2 one probably pointless hard tell anxious brain just want believe doctor give positive news believe I fine edit note aware health anxiety currently the rapy already still bug though
6
walk crowd room anyone feel super awkward simply walk crowd room even just lightly populate a wkward like look walk normal people look narcissistic way low self confidence way ive issue e ntire life simple somewhat crowd gym complete set designate area walk across room grab paper sanitizer spray wipe station walk past park car people different level anxiety other experien ce
status

```
## 1 Anxiety
## 2 Anxiety
## 3 Anxiety
## 4 Anxiety
## 5 Anxiety
## 6 Anxiety
```

```
# Optional: Convert cleaned data to Document-Term Matrix (DTM)
dtm <- DocumentTermMatrix(Corpus(VectorSource(cleaned_statements$statement)))
```

Word Cloud - All Statuses

```
library(wordcloud)
```

```
## Warning: package 'wordcloud' was built under R version 4.4.2
```

```
## Loading required package: RColorBrewer
```

```
wordcloud(words = unlist(strsplit(as.character(cleaned_statements$statement), " ")),
          min.freq = 125, scale=c(4,0.8), colors=brewer.pal(8, "Dark2"))
```

```
## Warning in tm_map.SimpleCorpus(corpus, tm::removePunctuation): transformation
## drops documents
```

```
## Warning in tm_map.SimpleCorpus(corpus, function(x) tm::removeWords(x,
## tm::stopwords())): transformation drops documents
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : symptom could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : something could
## not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : anything could
## not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : extremely could
## not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : conversation
## could not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : panic could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : basically could
## not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : medication could
## not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : college could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : drive could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : bad could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : constant could
## not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : effect could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : manic could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : depression could
## not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : episode could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : recently could
## not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : phone could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : mother could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : stay could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : come could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : put could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : call could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : remember could
## not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : disorder could
## not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : without could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : way could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : girl could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : thought could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : hope could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : seem could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : control could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : point could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : pain could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : write could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : together could
## not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : care could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : people could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : meet could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : cope could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : night could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : pretty could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : play could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : attempt could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : thank could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : leave could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : year could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : change could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : doctor could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : suicide could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : hour could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : move could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : think could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : hand could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : life could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : two could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : manage could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : attack could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : try could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : even could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : diagnose could
## not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : stress could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : know could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : walk could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : therapy could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : feel could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : share could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : back could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : say could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : due could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : car could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : maybe could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : couple could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : mean could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : easy could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : happen could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : however could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : make could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : another could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : really could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : drink could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : small could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : guess could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : bring could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : support could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : believe could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : fall could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : cant could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : abuse could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : first could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : especially could
## not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : dream could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : handle could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : much could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : advice could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : alone could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : hard could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : tell could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : never could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : spend could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : fight could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : man could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : forget could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : want could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : sort could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : group could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : health could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : sure could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : send could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : chest could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : great could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : show could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : heart could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : past could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : look could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : wake could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : may could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : hate could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : exist could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : morning could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : always could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : someone could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : almost could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : feeling could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : everything could
## not be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : long could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : mental could not
## be fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : yes could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : week could not be
## fit on page. It will not be plotted.
```

```
## Warning in wordcloud(words =
## unlist(strsplit(as.character(cleaned_statements$statement), : side could not be
## fit on page. It will not be plotted.
```

Word Cloud - Anxiety

```r
library(wordcloud)
library(RColorBrewer)

# Anxiety Word Cloud
anxiety_data <- cleaned_statements[cleaned_statements$status == "Anxiety", ]
wordcloud(words = unlist(strsplit(as.character(anxiety_data$statement), " ")),
          min.freq = 100,
          scale=c(4,0.8),
          colors=brewer.pal(8, "Dark2"),
          main = "Word Cloud for Anxiety")
```

```
## Warning in tm_map.SimpleCorpus(corpus, tm::removePunctuation): transformation
## drops documents
```

```
## Warning in tm_map.SimpleCorpus(corpus, function(x) tm::removeWords(x,
## tm::stopwords())): transformation drops documents
```


Word Cloud - Bipolar

```r
# Bipolar Word Cloud
bipolar_data <- cleaned_statements[cleaned_statements$status == "Bipolar", ]
wordcloud(words = unlist(strsplit(as.character(bipolar_data$statement), " ")),
          min.freq = 100,
          scale=c(4,0.8),
          colors=brewer.pal(8, "Dark2"),
          main = "Word Cloud for Bipolar")
```

```
## Warning in tm_map.SimpleCorpus(corpus, tm::removePunctuation): transformation
## drops documents
```

```
## Warning in tm_map.SimpleCorpus(corpus, function(x) tm::removeWords(x,
## tm::stopwords())): transformation drops documents
```


Word Cloud - Depression

```r
# Depression Word Cloud
depression_data <- cleaned_statements[cleaned_statements$status == "Depression", ]
wordcloud(words = unlist(strsplit(as.character(depression_data$statement), " ")),
          min.freq = 100,
          scale=c(4,0.8),
          colors=brewer.pal(8, "Dark2"),
          main = "Word Cloud for Depression")
```

```
## Warning in tm_map.SimpleCorpus(corpus, tm::removePunctuation): transformation
## drops documents
```

```
## Warning in tm_map.SimpleCorpus(corpus, function(x) tm::removeWords(x,
## tm::stopwords())): transformation drops documents
```


Word Count - Normal

```
# Normal Word Cloud
normal_data <- cleaned_statements[cleaned_statements$status == "Normal", ]
wordcloud(words = unlist(strsplit(as.character(normal_data$statement), " ")),
          min.freq = 15,
          scale=c(4,0.8),
          colors=brewer.pal(8, "Dark2"),
          main = "Word Cloud for Normal")
```

```
## Warning in tm_map.SimpleCorpus(corpus, tm::removePunctuation): transformation
## drops documents
```

```
## Warning in tm_map.SimpleCorpus(corpus, function(x) tm::removeWords(x,
## tm::stopwords())): transformation drops documents
```
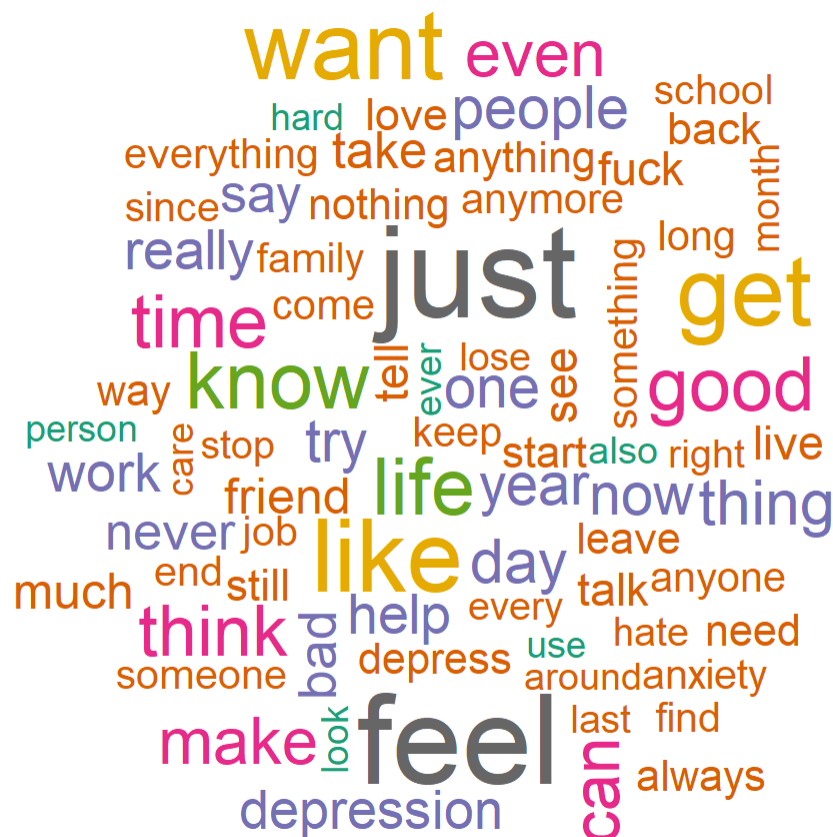

Word Cloud - Personality Disorder

```
# Personality Disorder Word Cloud
pd_data <- cleaned_statements[cleaned_statements$status == "Personality disorder", ]
wordcloud(words = unlist(strsplit(as.character(pd_data$statement), " ")),
          min.freq = 100,
          scale=c(4,0.8),
          colors=brewer.pal(8, "Dark2"),
          main = "Word Cloud for Personality Disorder")
```

```
## Warning in tm_map.SimpleCorpus(corpus, tm::removePunctuation): transformation
## drops documents
```

```
## Warning in tm_map.SimpleCorpus(corpus, function(x) tm::removeWords(x,
## tm::stopwords())): transformation drops documents
```

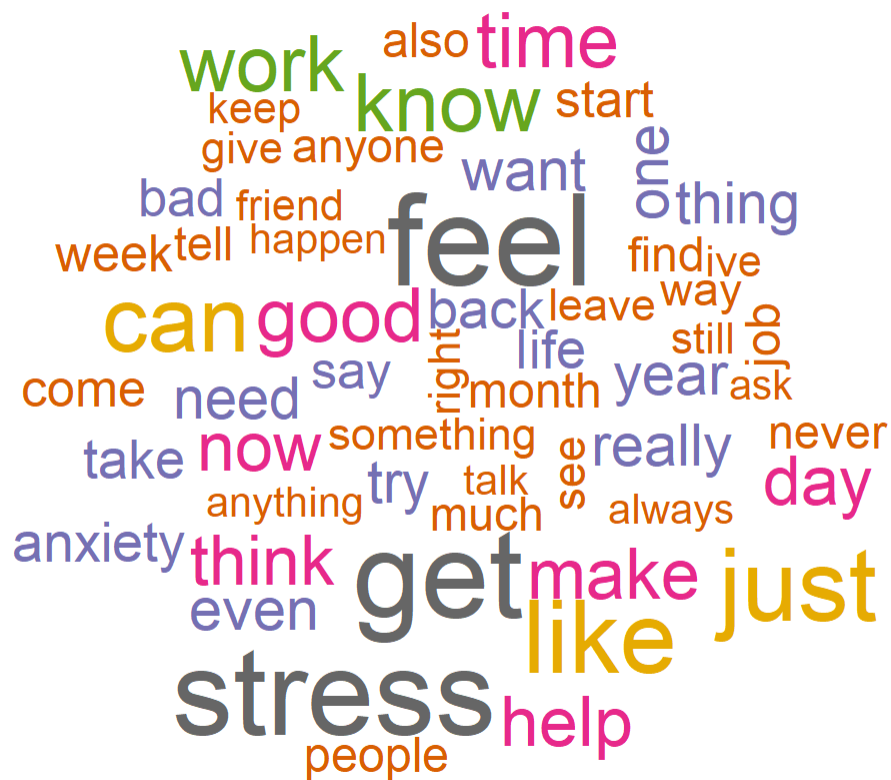

Word Cloud - Stress

```
stress_data <- cleaned_statements[cleaned_statements$status == "Stress", ]
wordcloud(words = unlist(strsplit(as.character(stress_data$statement), " ")),
          min.freq = 75,
          scale=c(4,0.8),
          colors=brewer.pal(8, "Dark2"),
          main = "Word Cloud for Stress")
```

```
## Warning in tm_map.SimpleCorpus(corpus, tm::removePunctuation): transformation
## drops documents
```

```
## Warning in tm_map.SimpleCorpus(corpus, function(x) tm::removeWords(x,
## tm::stopwords())): transformation drops documents
```


Word Cloud - Suicidal

```r
# Suicidal Word Cloud
suicidal_data <- cleaned_statements[cleaned_statements$status == "Suicidal", ]
wordcloud(words = unlist(strsplit(as.character(suicidal_data$statement), " ")),
          min.freq = 100,
          scale=c(4,0.8),
          colors=brewer.pal(8, "Dark2"),
          main = "Word Cloud for Suicidal")
```

```
## Warning in tm_map.SimpleCorpus(corpus, tm::removePunctuation): transformation
## drops documents
```

```
## Warning in tm_map.SimpleCorpus(corpus, function(x) tm::removeWords(x,
## tm::stopwords())): transformation drops documents
```