

## Text Mining

```
library(textmineR)

## Warning: package 'textmineR' was built under R version 3.4.4

## Loading required package: Matrix

## textmineR v3.0 is coming with major changes that WILL break things!
## Please see https://github.com/TommyJones/textmineR/tree/3.0 for the
## development version. Expected release is October or November of 2018.
## Please submit questions or requests to
https://github.com/TommyJones/textmineR/issues
```

```
library(tidyverse)

## Warning: package 'tidyverse' was built under R version 3.4.4

## -- Attaching packages -----
## ----- tidyverse 1.2.1 -----

## v ggplot2 3.0.0      v purrr  0.2.5
## v tibble  1.4.2      v dplyr  0.7.6
## v tidyr   0.8.1      v stringr 1.3.1
## v readr   1.1.1      v forcats 0.3.0

## Warning: package 'ggplot2' was built under R version 3.4.4
## Warning: package 'tibble' was built under R version 3.4.4
## Warning: package 'tidyr' was built under R version 3.4.4
## Warning: package 'readr' was built under R version 3.4.4
## Warning: package 'purrr' was built under R version 3.4.4
## Warning: package 'dplyr' was built under R version 3.4.4
## Warning: package 'stringr' was built under R version 3.4.4
## Warning: package 'forcats' was built under R version 3.4.4

## -- Conflicts -----
## ----- tidyverse_conflicts() -----
## x tidyr::expand() masks Matrix::expand()
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(factoextra)

## Warning: package 'factoextra' was built under R version 3.4.4
```

```
## Welcome! Related Books: `Practical Guide To Cluster Analysis in R` at  
https://goo.gl/13EFCZ
```

```
library(cluster)  
library(NbClust)  
library(fpc)
```

```
## Warning: package 'fpc' was built under R version 3.4.4
```

```
library(wordcloud)
```

```
## Warning: package 'wordcloud' was built under R version 3.4.4
```

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

```
library(dendroextras)
```

```
## Warning: package 'dendroextras' was built under R version 3.4.4
```

```
##  
## Attaching package: 'dendroextras'
```

```
## The following object is masked from 'package:dplyr':  
##  
##      slice
```

```
library(dendextend)
```

```
## Warning: package 'dendextend' was built under R version 3.4.4
```

```
##  
## -----  
## Welcome to dendextend version 1.8.0  
## Type citation('dendextend') for how to cite the package.  
##  
## Type browseVignettes(package = 'dendextend') for the package vignette.  
## The github page is: https://github.com/talgalili/dendextend/  
##  
## Suggestions and bug-reports can be submitted at:  
https://github.com/talgalili/dendextend/issues  
## Or contact: <tal.galili@gmail.com>  
##  
## To suppress this message use:  
suppressPackageStartupMessages(library(dendextend))  
## -----
```

```
##  
## Attaching package: 'dendextend'
```

```
## The following object is masked from 'package:dendroextras':  
##  
##      labels<-
```

```
## The following object is masked from 'package:stats':
##
##      cutree

library(mclust)

## Warning: package 'mclust' was built under R version 3.4.4

## Package 'mclust' version 5.4.1
## Type 'citation("mclust")' for citing this R package in publications.

##
## Attaching package: 'mclust'

## The following object is masked from 'package:purrr':
##
##      map

library(dbscan)

## Warning: package 'dbscan' was built under R version 3.4.4

##
## Attaching package: 'dbscan'

## The following object is masked from 'package:fpc':
##
##      dbscan

library(dplyr)
library(e1071)

## Warning: package 'e1071' was built under R version 3.4.4

library(seriation)

## Warning: package 'seriation' was built under R version 3.4.4

library(arules)

## Warning: package 'arules' was built under R version 3.4.4

##
## Attaching package: 'arules'

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

## The following objects are masked from 'package:base':
##
##      abbreviate, write
```

```

library(ggplot2)
library(RColorBrewer)
library(tm)

## Warning: package 'tm' was built under R version 3.4.4

## Loading required package: NLP

## Warning: package 'NLP' was built under R version 3.4.4

##
## Attaching package: 'NLP'

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

##
## Attaching package: 'tm'

## The following object is masked from 'package:arules':
##
##   inspect

library(DT)

## Warning: package 'DT' was built under R version 3.4.4

library(arulesViz)

## Warning: package 'arulesViz' was built under R version 3.4.4

## Loading required package: grid

library(arulesCBA)

## Warning: package 'arulesCBA' was built under R version 3.4.4

## Loading required package: discretization

library(dplyr)
bible<-
read.csv("https://raw.githubusercontent.com/vigneshjmurali/Statistical-
Predictive-Modelling/master/Datasets/bible_asv.csv")
dim(bible)

## [1] 31103      8

bible_var=aggregate(Testaments~Books,data=bible,FUN = unique,collapse="" )
bible_var$Testaments=as.factor(ifelse(bible_var$Testaments==bible_var$Testame
nts[1],1,2))

levels(bible$Sections)

```

```
## [1] "Apostles" "Gospels" "History" "Law" "Paul" "Prophets"
## [7] "Wisdom"

bible_books=aggregate(Sections~Books, data=bible, FUN = unique, collapse="")
bible_books$Sections<-
ordered(bible_books$Sections,levels=c('Apostles','Gospels','History','Law','Paul','Prophets','Wisdom'))

bible_chap=aggregate(Testaments~Chapters,data=bible,FUN=unique, collapse="")
bible_chap$Testaments=as.factor(ifelse(bible_chap$Testaments==bible_chap$Testaments[1],1,2))
bible_chas=aggregate(Sections~Chapters,data=bible,FUN=unique, collapse="")
bible_chas$Sections<-
ordered(bible_chas$Sections,levels=c('Apostles','Gospels','History','Law','Paul','Prophets','Wisdom'))

bible_ver=bible[,c('Testaments','Verses')]
bible_ver$Testaments=as.factor(ifelse(bible_ver$Testaments==bible_ver$Testaments[1],1,2))
bible_verse=bible[,c('Sections','Verses')]
bible_verse$Sections<-
ordered(bible_verse$Sections,levels=c('Apostles','Gospels','History','Law','Paul','Prophets','Wisdom'))

bible_test=aggregate(Testaments~text,data=bible,FUN=unique, collapse="")
bible_test$Testaments=as.factor(ifelse(bible_test$Testaments==bible_test$Testaments[1],1,2))

bible_sect=aggregate(Sections~text,data=bible,FUN=unique, collapse="")
```

All the texts from the verses are collapsed into a common book which makes it easier to perform the analysis.

```
attach(bible)
text.Book=c()
for (i in 1:66){
  text.Book[i]=paste(text[Books==as.character(unique(Books)[i])],collapse="")
}

text.Chapters=c()
for (i in 1:1189){

text.Chapters[i]=paste(text[Chapters==as.character(unique(Chapters)[i])],collapse = "")
}

bible_col=data.frame(Books=unique(Books),text=text.Book)
bible_chapters=data.frame(Chapters=unique(Chapters),text=text.Chapters)
bible_verses=bible
dim(bible_col);dim(bible_chapters);dim(bible_verses)

## [1] 66 2
```

```
## [1] 1189    2
## [1] 31103    8
```

In order to get better results, we should convert all the characters into lower cases, remove the punctuations, numbers and whitespace.

```
my_words<-stopwords("en")
my_stopwords1 = my_words
my_stopwords2 =
c('thou','thee','thy','ye','shall','shalt','lo','unto','hath','thereof','hast',
',','set','thine','art','yea','midst','wherefore','wilt','thysself')

Testaments=c(rep('OT',39),rep('NT',27))
Sections=c(rep('Law',5),
rep('History',12),rep('Wisdom',5),rep('Prophets',17),rep('Gospels',5),rep('Paul',13),rep('Apostles',9))
bible_new
=data.frame(Books=unique(Books),Testaments=as.factor(c(rep("OT",39),rep("NT",
27)))),

Sections=as.factor(c(rep("Law",5),rep("History",12),rep("Wisdom",5),rep("Prophets",17),rep("Gospels",5),rep("Paul",13),rep("Apostles",9))),
text=text.Book)
```

## Clustering Analysis:

```
dtm_b <- CreateDtm(bible_col$text,doc_names = bible_col$Books,ngram_window =
c(1, 7),
stopword_vec =
c(tm::stopwords("english"),tm::stopwords("SMART"),
my_stopwords1, my_stopwords2),
lower = TRUE, remove_punctuation = TRUE, remove_numbers = FALSE)
```

```
##
|
|=====| 11%
|=====| 21%
|=====| 32%
|=====| 42%
|=====| 53%
|=====| 64%
|=====| 74%
|=====| 85%
```

```

===== | 95%
===== | 100%
##
===== | 11%
===== | 21%
===== | 32%
===== | 42%
===== | 53%
===== | 64%
===== | 74%
===== | 85%
===== | 95%
===== | 100%

```

```
tf <- TermDocFreq(dtm_b)
```

```
vocabulary <- tf$term[tf$term_freq>2 & tf$doc_freq>1]
```

```
dtm_b <- dtm_b[, vocabulary]
```

```
csim_b <- dtm_b / sqrt(rowSums(dtm_b*dtm_b))
```

```
csim_b <- csim_b %*% t(csim_b)
```

```
dist.mtx_b <- 1-csim_b
```

```
Testaments=c(rep('OT',39),rep('NT',27))
```

```
Sections=c(rep('Law',5),
```

```
rep('History',12),rep('Wisdom',5),rep('Prophets',17),
```

```
rep('Gospels',5),rep('Paul',13),rep("Apostles",9))
```

## PCA

```
m_b<-as.matrix(dtm_b)
```

```
dtm_b.pca=prcomp(m_b)
```

```
dtm_b.pca$rotation[1:5,1:5]
```

```
##          PC1          PC2          PC3          PC4
## death_hand -3.580728e-04  0.0008654491 -0.0001652069 -0.0003291659
## round_cut  -1.639124e-04  0.0008628898  0.0006513863  0.0001668109
## jehovah_god_die -4.040029e-04 -0.0001670719 -0.0004160397  0.0003339312
## worthy_unloose  2.483106e-05 -0.0002439001 -0.0001839567 -0.0009217738
```

```
## saul_meet      -1.672111e-04 -0.0009880575  0.0004396606 -0.0001791222
##                                     PC5
## death_hand     -6.772558e-04
## round_cut       3.457596e-05
## jehovah_god_die -2.607624e-04
## worthy_unloose  7.968094e-04
## saul_meet      -6.381335e-04

dim(dtm_b.pca$x)

## [1] 66 66

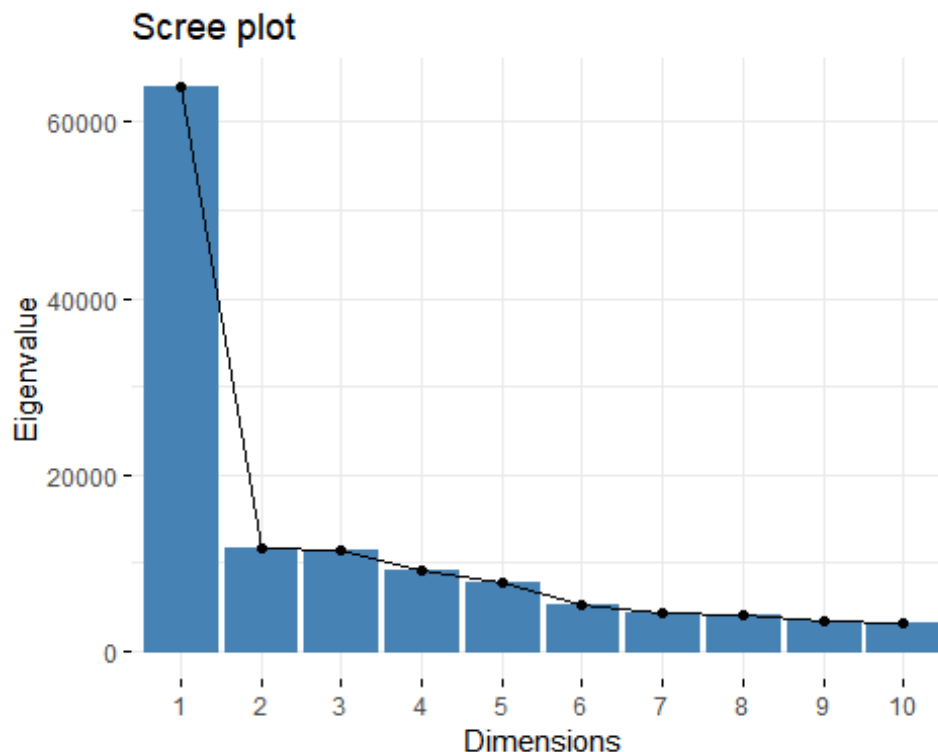
dtm_b.sd=dtm_b.pca$sdev
dtm_b.var=dtm_b.pca$sdev^2
dtm_b.var[1:5]

## [1] 63990.369 11716.451 11492.751  9245.405  7825.913

pve=dtm_b.var/sum(dtm_b.var) ; cumsum(pve[1:10])

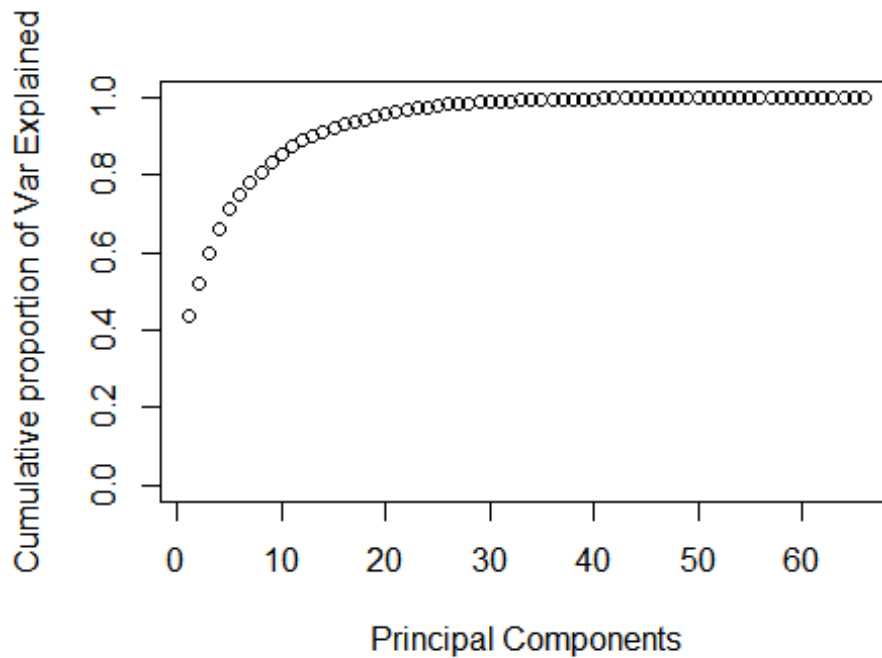
## [1] 0.4376956 0.5178364 0.5964470 0.6596858 0.7132152 0.7495426 0.7800858
## [8] 0.8083959 0.8321576 0.8548969

fviz_screplot(dtm_b.pca,np=10,choice="eigenvalue")
```



```
plot(cumsum(pve),xlab="Principal Components", ylab="Cumulative proportion of
Var Explained", ylim=c(0,1),type='b')
```





```
which.max(cumsum(pve)[cumsum(pve)<0.90])
```

```
## [1] 12
```

```
dtm_bnew=as.data.frame(dtm_b.pca$x[,1:12])
```

```
dtm_bnew1=dtm_b.pca$x[,1:12]
```

## K Means:

```
set.seed(2)
```

```
km_2.fit=kmeans(dtm_bnew,2,nstart=30)
```

```
attributes(km_2.fit)
```

```
## $names
```

```
## [1] "cluster"      "centers"      "totss"      "withinss"
```

```
## [5] "tot.withinss" "betweenss"   "size"      "iter"
```

```
## [9] "ifault"
```

```
##
```

```
## $class
```

```
## [1] "kmeans"
```

```
y_k2=table(km_2.fit$cluster, bible_var$Testaments) ; y_k2
```

```
##
```

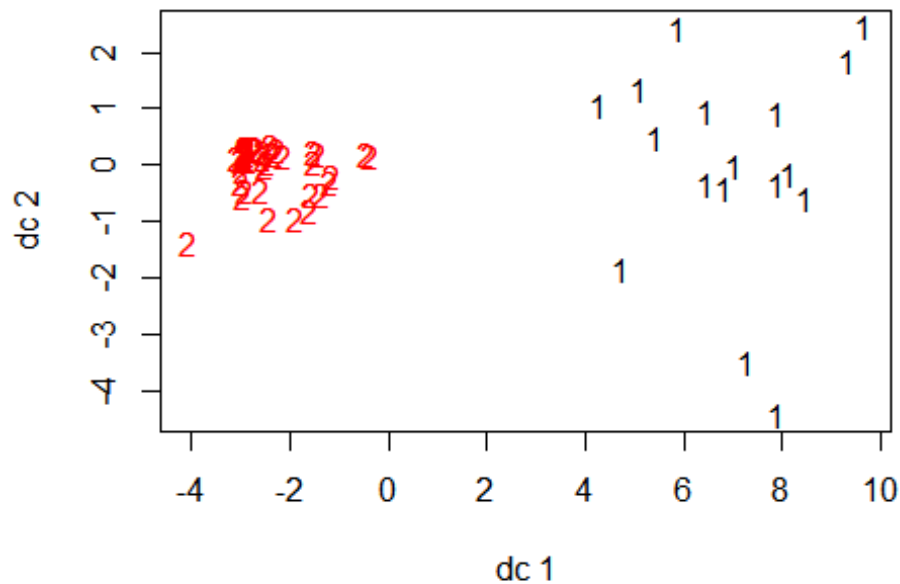
```
##      1  2
```

```
##    1  9  8
```

```
##    2 30 19
```

```
mean(km_2.fit$cluster==bible_var$Testaments)
```

```
## [1] 0.4242424
misrate_k2<-1-sum(diag(y_k2))/sum(y_k2) ; misrate_k2
## [1] 0.5757576
plotcluster(dtm_bnew,km_2.fit$cluster)
```



```
set.seed(4)
km_7.fit=kmeans(dtm_bnew,7,nstart = 30)
attributes(km_7.fit)

## $names
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
##
## $class
## [1] "kmeans"

y_k7=table(km_7.fit$cluster,bible_books$Sections) ; y_k7

##
##    Apostles  Gospels  History  Law  Paul  Prophets  Wisdom
## 1          1         0         0   0    0          1         0
## 2          1         0         3   0    1          0         0
## 3          0         0         0   1    1          0         1
## 4          4         4         4   4    9        15         4
```

```
##      5      1      0      2  0      1      0      0
##      6      1      0      1  0      1      0      0
##      7      1      1      2  0      0      1      0

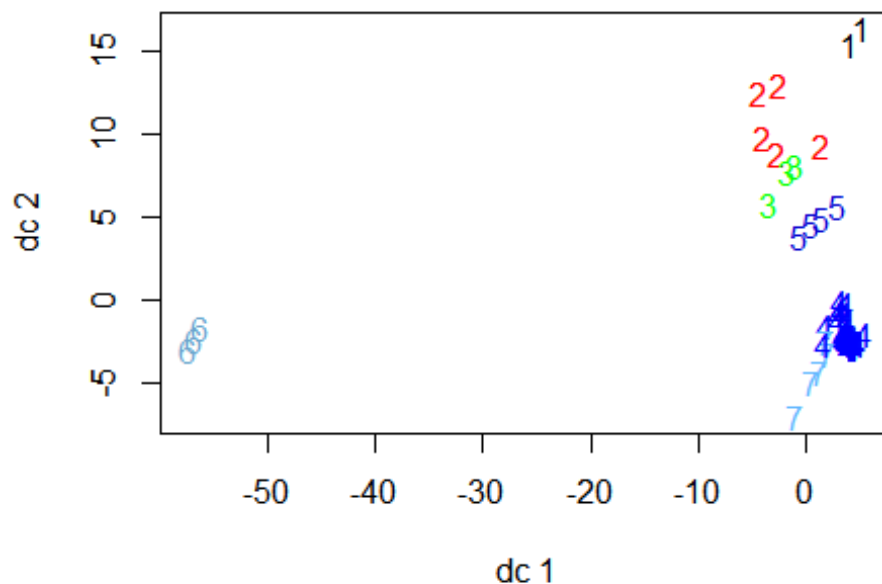
mean(km_7.fit$cluster == bible_books$Sections)

## [1] 0

misrate_k7<-1-sum(diag(y_k7))/sum(y_k7) ; misrate_k7

## [1] 0.9090909

plotcluster(dtm_bnew, km_7.fit$cluster)
```



## Hierarchical Clustering:

```
par(mfrow=c(1,2))
hc.ward=hclust(dist(dtm_bnew, method = "euclidean"), method="ward.D2")
plot(hc.ward,main="Complete Linkage", xlab="", sub="", cex=.9)

rect.hclust(hc.ward,k=2,border="red")
groups2=cutree(hc.ward,2)
y_h2<-table(groups2,bible_var$Testaments) ;y_h2

##
## groups2  1  2
##          1 31 19
##          2  8  8
```

```

mean(groups2 ==bible_var$Testaments)

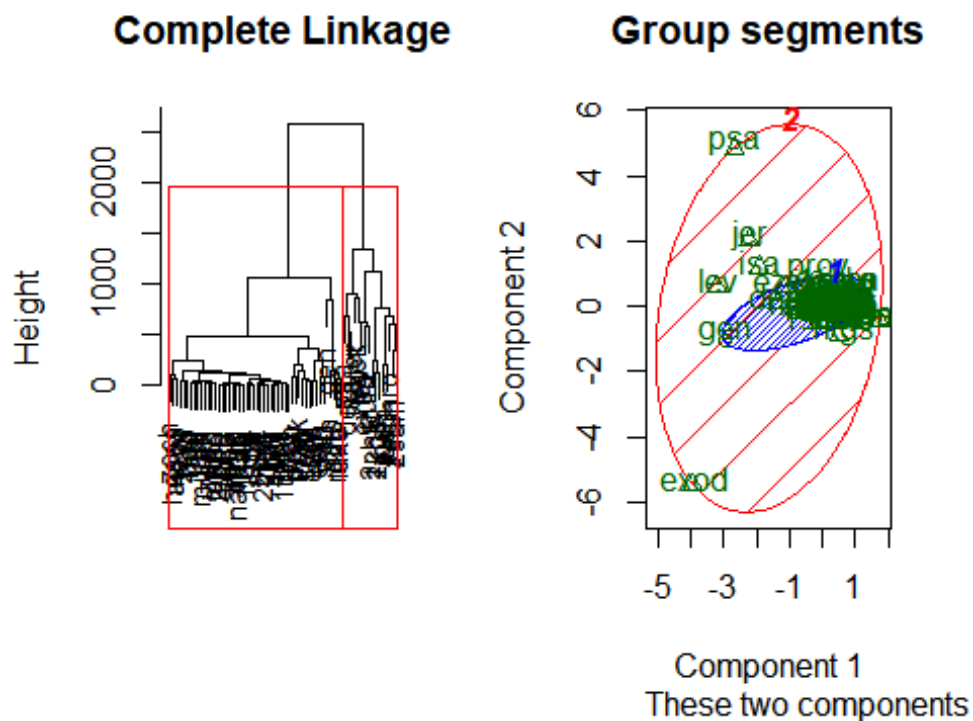
## [1] 0.5909091

misrate_h2<-1-sum(diag(y_h2))/sum(y_h2) ; misrate_h2

## [1] 0.4090909

clusplot(dtm_bnew, groups2, color=TRUE, shade=TRUE,
          labels=2, lines=0, main= 'Group segments')

```



```

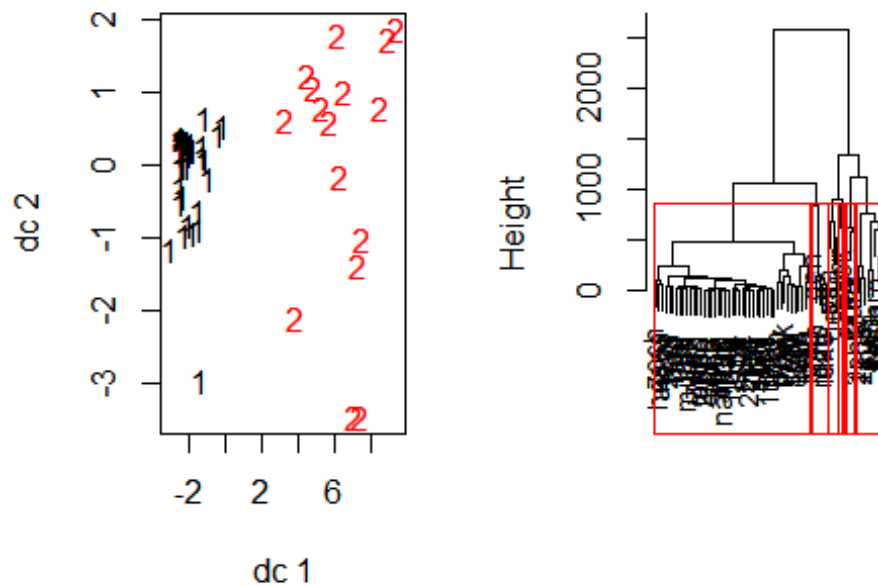
plotcluster(dtm_bnew, groups2)

plot(hc.ward,main="Complete Linkage", xlab="", sub="", cex=.9)

rect.hclust(hc.ward,k=7,border="red")

```

## Complete Linkage



```
groups7=cutree(hc.ward,7)
y_h7<-table(groups7,bible_books$Sections) ;y_h7

##
## groups7 Apostles Gospels History Law Paul Prophets Wisdom
##      1      1      1      3  0      0      0      0
##      2      1      0      1  0      1      0      0
##      3      1      0      0  0      0      1      1
##      4      2      0      4  0      2      0      0
##      5      4      4      4  4      9      16      4
##      6      0      0      0  0      1      0      0
##      7      0      0      0  0      0      0      0

mean(groups7 ==bible_books$Sections)

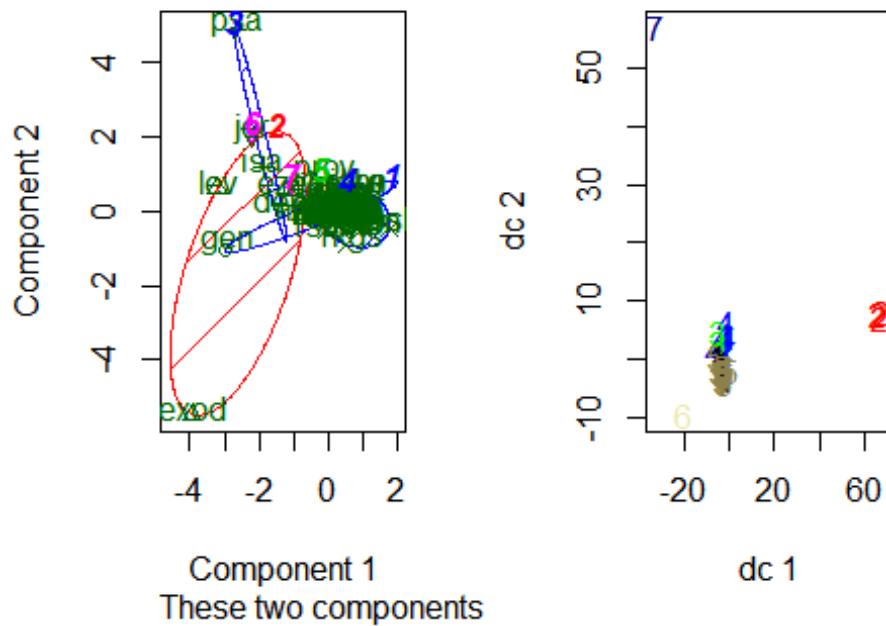
## [1] 0

misrate_h7<-1-sum(diag(y_h7))/sum(y_h7) ; misrate_h7

## [1] 0.8484848

clusplot(dtm_bnew, groups7, color=TRUE, shade=TRUE,
          labels=2, lines=0, main= 'Group segments')
plotcluster(dtm_bnew, groups7)
```

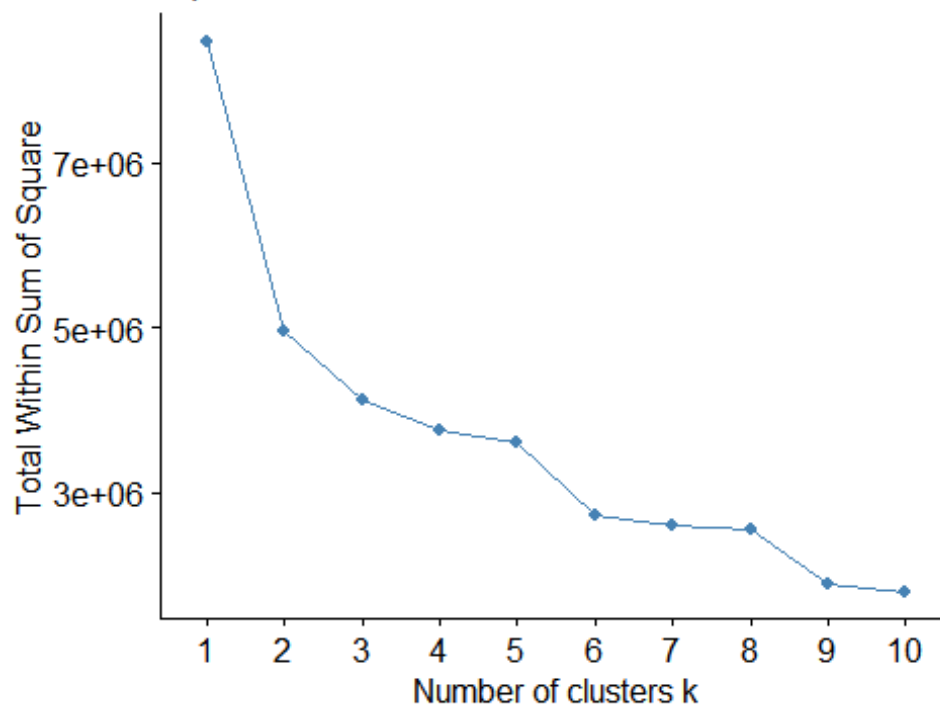
## Group segments



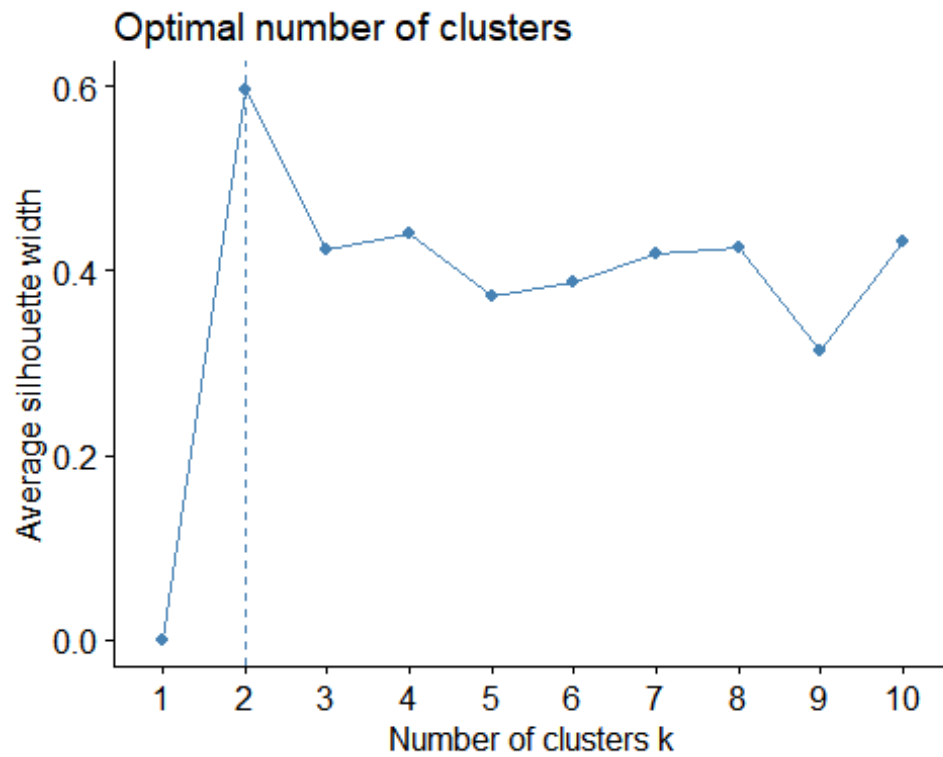
## NB Clust:

```
par(mfrow=c(2,2))  
fviz_nbclust(dtm_bnew1,kmeans,method="wss")
```

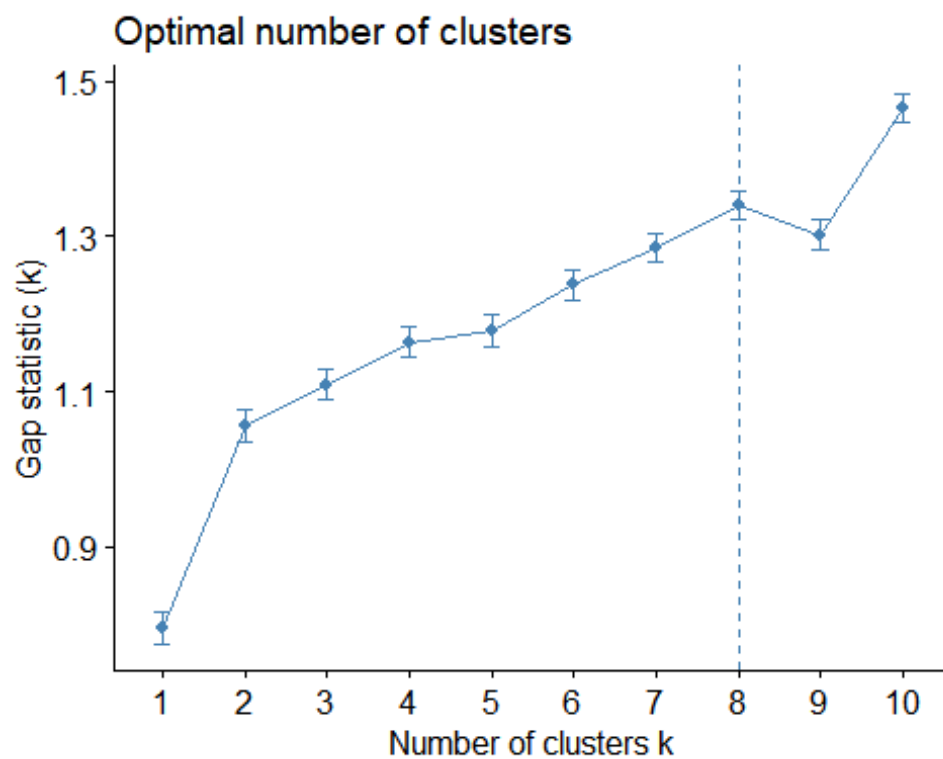
## Optimal number of clusters



```
fviz_nbclust(dtm_bnew1, kmeans, method="silhouette")
```



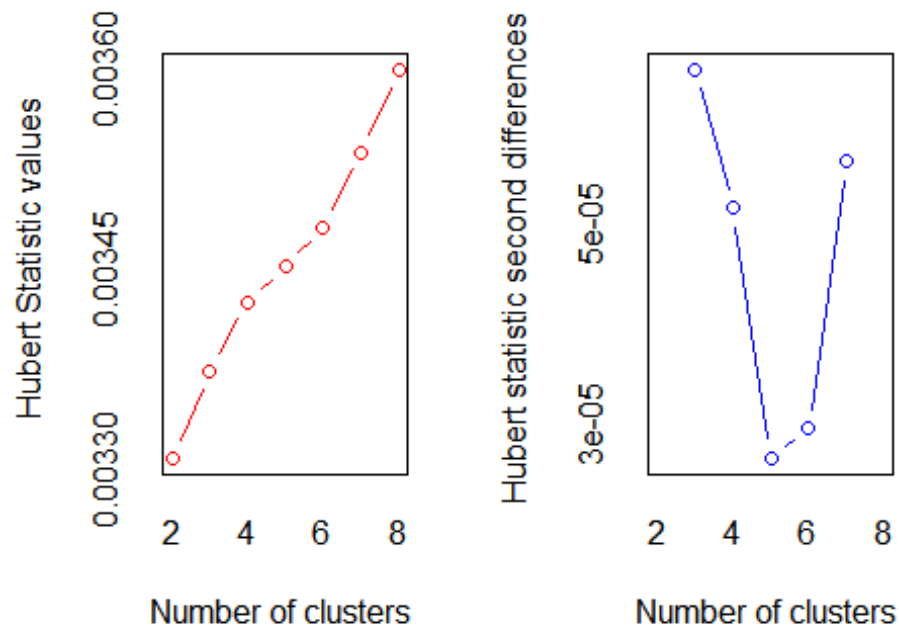
```
fviz_nbclust(dtm_bnew1, kmeans, method="gap_stat")
```



```

mito.nbclust<-dtm_bnew1 %>%
  scale() %>%
  NbClust(distance="euclidean",min.nc=2,max.nc=8,method="single",index="all")

```

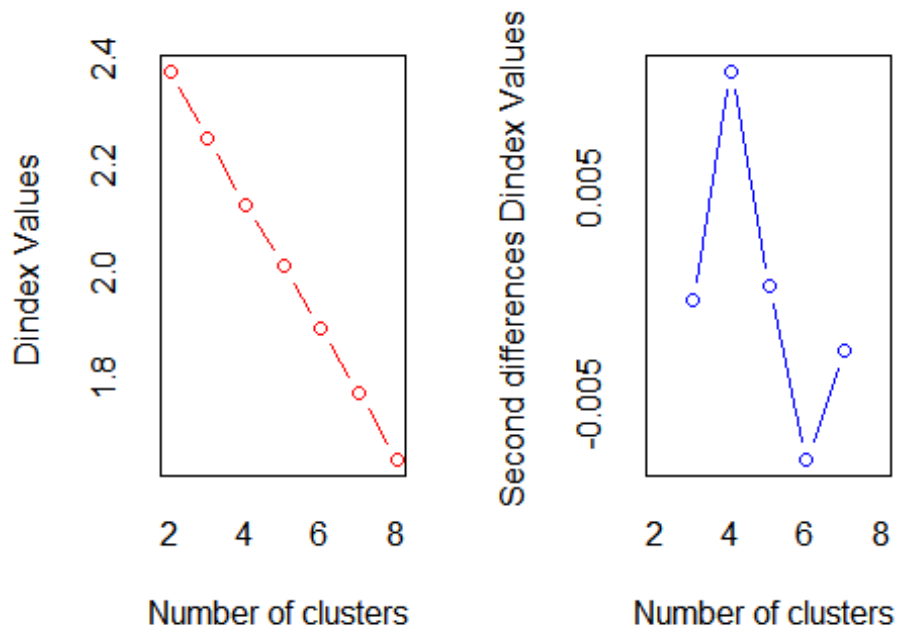


```

## *** : The Hubert index is a graphical method of determining the number of
clusters.
##           In the plot of Hubert index, we seek a significant knee
that corresponds to a
##           significant increase of the value of the measure i.e the
significant peak in Hubert
##           index second differences plot.
##

```





```
## *** : The D index is a graphical method of determining the number of
clusters.
##           In the plot of D index, we seek a significant knee (the
significant peak in Dindex
##           second differences plot) that corresponds to a significant
increase of the value of
##           the measure.
##
## *****
## * Among all indices:
## * 6 proposed 2 as the best number of clusters
## * 3 proposed 3 as the best number of clusters
## * 2 proposed 4 as the best number of clusters
## * 1 proposed 6 as the best number of clusters
## * 11 proposed 8 as the best number of clusters
##
##           ***** Conclusion *****
##
## * According to the majority rule, the best number of clusters is  8
##
## *****
```

```
par(mfrow=c(1,2))
mb.fit <- Mclust(dtm_bnew)
summary(mb.fit)
```

```
par(mfrow=c(1,2))
mb.fit <- Mclust(dtm_bnew)
summary(mb.fit)

## -----
## Gaussian finite mixture model fitted by EM algorithm
## -----
##
## Mclust VEI (diagonal, equal shape) model with 6 components:
##
##      log.likelihood   n df          BIC          ICL
##      -3490.608  66 94  -7375.043  -7376.014
##
## Clustering table:
##      1  2  3  4  5  6
##     14 19  3 10 15  5

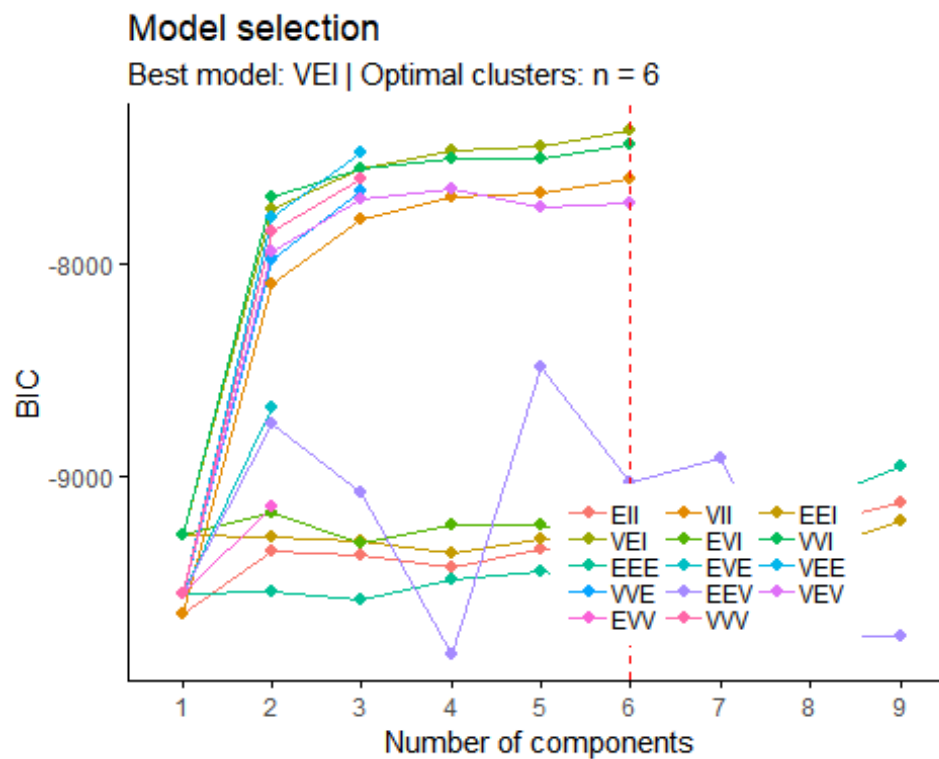
mb.fit$modelName

## [1] "VEI"

mb.fit$G

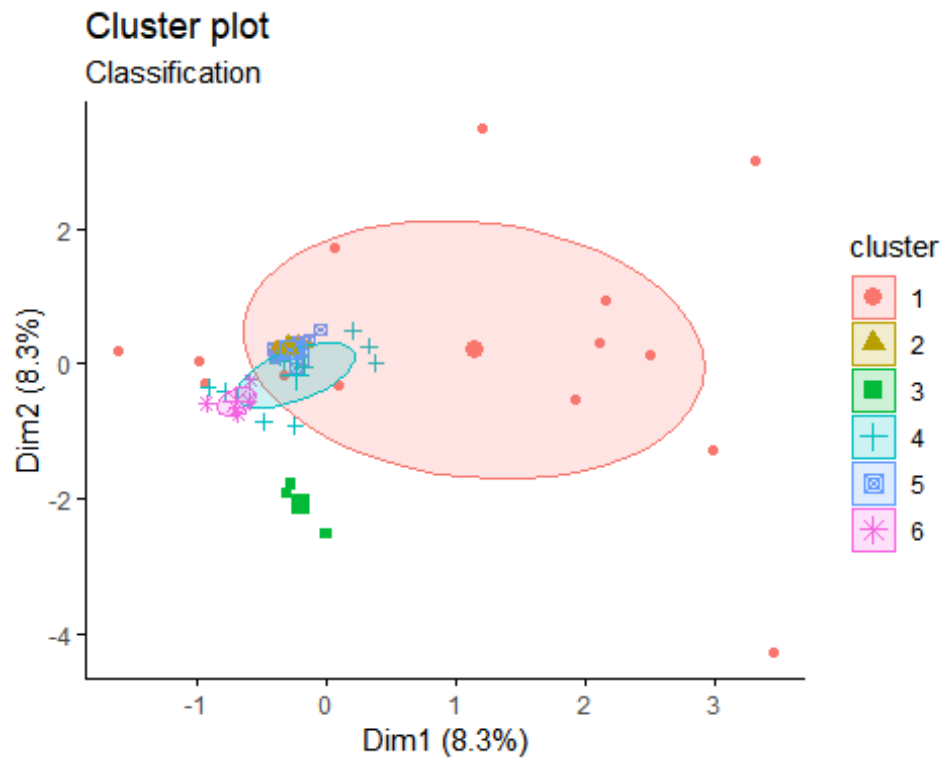
## [1] 6

fviz_mclust(mb.fit, "BIC", palette = "futuruma")
```



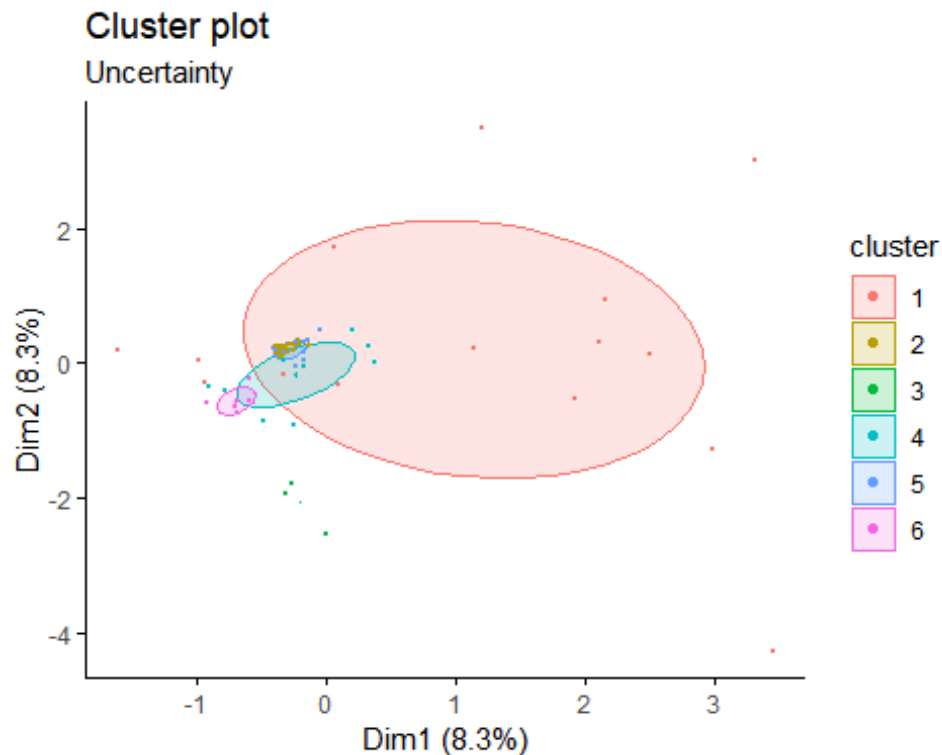
```
fviz_mclust(mb.fit, "classification", geom = "point", pointsize = 1.5,  
palette = "futuruma")
```

```
## Too few points to calculate an ellipse
```



```
fviz_mclust(mb.fit, "uncertainty", palette = "futuruma")
```

```
## Too few points to calculate an ellipse
```



```
bible.group_sections<-data.frame(dtm_bnew,km_7.fit$cluster)
bible.group_testaments<-data.frame(dtm_bnew,km_2.fit$cluster)

corpus1<-Corpus(VectorSource(bible_sect$text))
text_corpus1 <- tm_map(corpus1,removeWords,my_stopwords1)

## Warning in tm_map.SimpleCorpus(corpus1, removeWords, my_stopwords1):
## transformation drops documents

text_corpus1 <- tm_map(corpus1,removeWords,my_stopwords2)

## Warning in tm_map.SimpleCorpus(corpus1, removeWords, my_stopwords2):
## transformation drops documents

text_corpus1 <- tm_map(corpus1, stripWhitespace)

## Warning in tm_map.SimpleCorpus(corpus1, stripWhitespace): transformation
## drops documents

text_corpus1 <- tm_map(corpus1, content_transformer(tolower))

## Warning in tm_map.SimpleCorpus(corpus1, content_transformer(tolower)):
## transformation drops documents

text_corpus1 <- tm_map(corpus1, removeWords, stopwords("english"))

## Warning in tm_map.SimpleCorpus(corpus1, removeWords,
stopwords("english")):
## transformation drops documents
```

```

text_corpus1 <- tm_map(corpus1, stemDocument)

## Warning in tm_map.SimpleCorpus(corpus1, stemDocument): transformation
drops
## documents

text_corpus1 <- tm_map(corpus1, removeNumbers)

## Warning in tm_map.SimpleCorpus(corpus1, removeNumbers): transformation
## drops documents

text_corpus1 <- tm_map(corpus1, removePunctuation)

## Warning in tm_map.SimpleCorpus(corpus1, removePunctuation): transformation
## drops documents

dtm_b2<-DocumentTermMatrix(text_corpus1); dim(dtm_b2)

## [1] 30722 12765

dtm_b221<-removeSparseTerms(dtm_b2,sparse=0.95); dim(dtm_b221)

## [1] 30722 48

dtmr1 <-DocumentTermMatrix(text_corpus1, control=list(wordLengths=c(2, 20),
bounds = list(global = c(2,45)))) ;dim(dtmr1)

## [1] 30722 7454

freq<-sort(colSums(as.matrix(dtmr1)),decreasing = TRUE); head(freq,10)

## nakedness      redeem appearance      eateth      apart      tables
##          58          56          56          55          54          54
## vessel        salute      sockets      esther
##          52          52          52          52

wf1<-data.frame(word=names(freq),freq=freq); head(wf1) ; head(wf1,10)

##          word freq
## nakedness nakedness 58
## redeem      redeem  56
## appearance appearance 56
## eateth       eateth  55
## apart        apart  54
## tables       tables  54

##          word freq
## nakedness nakedness 58
## redeem      redeem  56
## appearance appearance 56
## eateth       eateth  55
## apart        apart  54
## tables       tables  54

```

```
## vessel          vessel    52
## salute          salute    52
## sockets         sockets    52
## esther           esther     52

set.seed(142)
wordcloud(names(freq),freq,min.freq=20,max.words = 50,random.order =
FALSE,rot.per = .1,
          random.color=TRUE)

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : vessel could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : sockets could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : esther could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : talents could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : trumpets could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : ephah could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : next could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : haman could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : towns could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : ephod could not be fit on page. It will not be
## plotted.
```

```
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : weep could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : forgiven could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : shepherds could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : azariah could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : multiply could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : spear could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : joash could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : distress could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : rehoabam could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : looking could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : kind could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : weight could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : witnesses could not be fit on page. It will not be  
## plotted.
```

```
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : lions could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : job could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : dwelleth could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : touched could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : thorns could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : pharaohs could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : rejected could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : vengeance could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : teeth could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : recompense could not be fit on page. It will not
be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : steps could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : chamber could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : ishmael could not be fit on page. It will not be
## plotted.
```



```
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : boat could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : countries could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : jealousy could not be fit on page. It will not be  
## plotted.
```

laban  
gift apart just  
appearance  
nakedness  
redeem  
eateth pillar  
tables salute

```
wordcloud(names(freq),freq,min.freq=20,max.words = 50,random.order =  
FALSE,rot.per = .35,  
          random.color=TRUE)  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : redeem could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : vessel could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : sockets could not be fit on page. It will not be  
## plotted.
```

```
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : talents could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : trumpets could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : laban could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : ephah could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : next could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : haman could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : towns could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : ephod could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : weep could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : forgiven could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : shepherds could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : azariah could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : multiply could not be fit on page. It will not be  
## plotted.
```

```
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : joash could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : distress could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : rehoboam could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : looking could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : kind could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : weight could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : witnesses could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : lions could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : dwelleth could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : touched could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : thorns could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : pharaohs could not be fit on page. It will not be
## plotted.

## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,
## random.order = FALSE, : rejected could not be fit on page. It will not be
## plotted.
```

```
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : vengeance could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : teeth could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : steps could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : chamber could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : sinners could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : ishmael could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : boat could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : jesse could not be fit on page. It will not be  
## plotted.  
  
## Warning in wordcloud(names(freq), freq, min.freq = 20, max.words = 50,  
## random.order = FALSE, : countries could not be fit on page. It will not be  
## plotted.
```

pillar apart of  
eateth s  
nakedness  
appearance  
, , ,

```
corpus<-Corpus(VectorSource(bible_col$text))
text_corpus <- tm_map(corpus,removeWords,my_stopwords1)

## Warning in tm_map.SimpleCorpus(corpus, removeWords, my_stopwords1):
## transformation drops documents

text_corpus <- tm_map(corpus,removeWords,my_stopwords2)

## Warning in tm_map.SimpleCorpus(corpus, removeWords, my_stopwords2):
## transformation drops documents

text_corpus <- tm_map(corpus, stripWhitespace)

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

text_corpus <- tm_map(corpus, content_transformer(tolower))

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

text_corpus <- tm_map(corpus, removeWords, stopwords("english"))

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

text_corpus <- tm_map(corpus, stemDocument)
```

```

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

text_corpus <- tm_map(corpus, removeNumbers)

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

text_corpus <- tm_map(corpus, removePunctuation)

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

dtm_b2<-DocumentTermMatrix(text_corpus) ;dim(dtm_b2)

## [1]    66 27727

dtm_b22<-removeSparseTerms(dtm_b2,sparse=0.95) ; dim(dtm_b22);

## [1]    66 5269

dtmr <-DocumentTermMatrix(text_corpus, control=list(wordLengths=c(4, 20),
bounds = list(global = c(5,45))))
dim(dtmr) ;

## [1]    66 3965

freq<-sort(colSums(as.matrix(dtmr)),decreasing = TRUE); head(freq,20)

##      jehovah      king      israel      land      david      pass      moses
##      5870      2166      2150      1579      972      843      769
##      took      jesus      judah      fathers      jerusalem      spake      kings
##      751      737      723      634      630      614      590
##      thine      hundred      egypt      thus      voice      thousand
##      547      541      492      487      487      477

wf<-data.frame(word=names(freq),freq=freq); head(wf) ; head(wf,10)

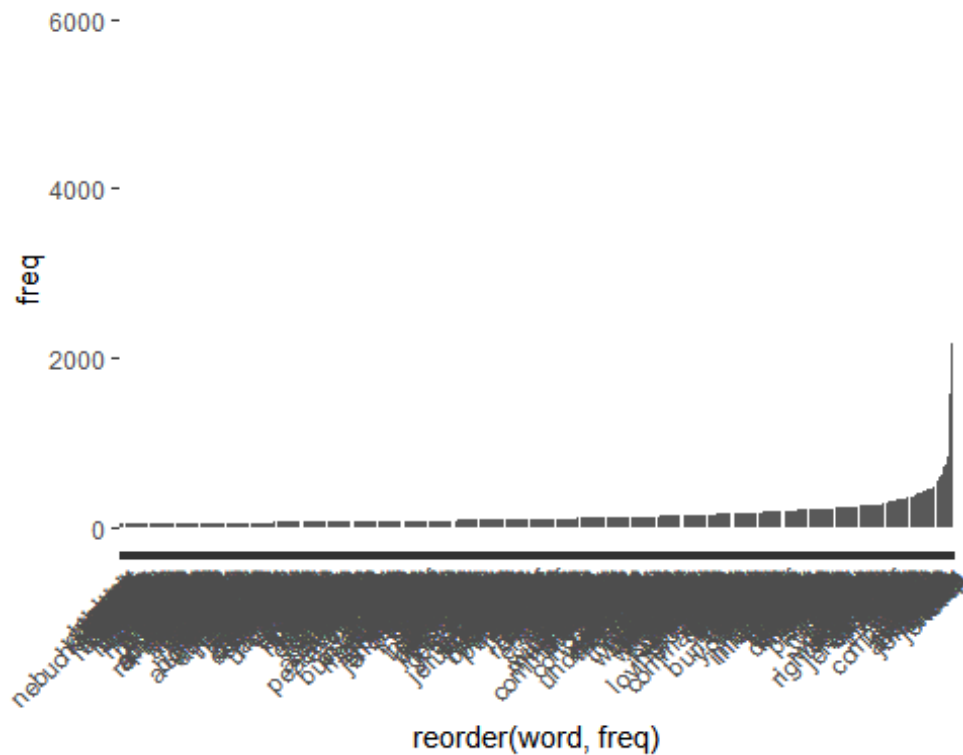
##           word freq
## jehovah jehovah 5870
## king      king 2166
## israel   israel 2150
## land     land 1579
## david    david  972
## pass     pass  843

##           word freq
## jehovah jehovah 5870
## king      king 2166
## israel   israel 2150
## land     land 1579
## david    david  972

```

```
## pass      pass  843
## moses     moses 769
## took      took  751
## jesus     jesus 737
## judah     judah 723
```

```
p<-
ggplot(subset(wf,freq>50),aes(x=reorder(word,freq),y=freq))+geom_bar(stat="id
entity")+
  theme(axis.text.x=element_text(angle=45,hjust=1))
p ; set.seed(150)
```



```
wordcloud(names(freq),freq,min.freq=100,max.words = 10,random.order =
FALSE,rot.per = .1,
  random.color=TRUE)
```



```
wordcloud(names(freq),freq,min.freq=100,max.words = 10,random.order =  
FALSE,rot.per = .35,  
          random.color=TRUE)
```





```

corpus<-Corpus(VectorSource(bible_test$text))
text_corpus <- tm_map(corpus,removeWords,my_stopwords1)

## Warning in tm_map.SimpleCorpus(corpus, removeWords, my_stopwords1):
## transformation drops documents

text_corpus <- tm_map(corpus,removeWords,my_stopwords2)

## Warning in tm_map.SimpleCorpus(corpus, removeWords, my_stopwords2):
## transformation drops documents

text_corpus <- tm_map(corpus, stripWhitespace)

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

text_corpus <- tm_map(corpus, content_transformer(tolower))

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

text_corpus <- tm_map(corpus, removeWords, stopwords("english"))

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

text_corpus <- tm_map(corpus, stemDocument)

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

text_corpus <- tm_map(corpus, removeNumbers)

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

text_corpus <- tm_map(corpus, removePunctuation)

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

dtm_b2<-DocumentTermMatrix(text_corpus);dim(dtm_b2)

## [1] 30722 12765

dtm_b22<-removeSparseTerms(dtm_b2,sparse=0.95);dim(dtm_b22)

## [1] 30722 48

dtmr <-DocumentTermMatrix(text_corpus, control=list(wordLengths=c(2, 20),
bounds = list(global = c(2,45))));dim(dtmr)

## [1] 30722 7454

```

```
freq<-sort(colSums(as.matrix(dtmr)),decreasing = TRUE); head(freq,25)
```

```
## nakedness      redeem appearance      eateth      apart      tables
##          58          56          56          55          54          54
## vessel         salute      sockets      esther      pillar      talents
##          52          52          52          52          52          51
## trumpets       laban        ephah        gift        next        haman
##          51          51          50          50          50          50
## towns          ephod        weep        just       forgiven    shepherds
##          50          50          49          49          49          49
## azariah
##          49
```

```
wf<-data.frame(word=names(freq),freq=freq); head(wf); head(wf,100)
```

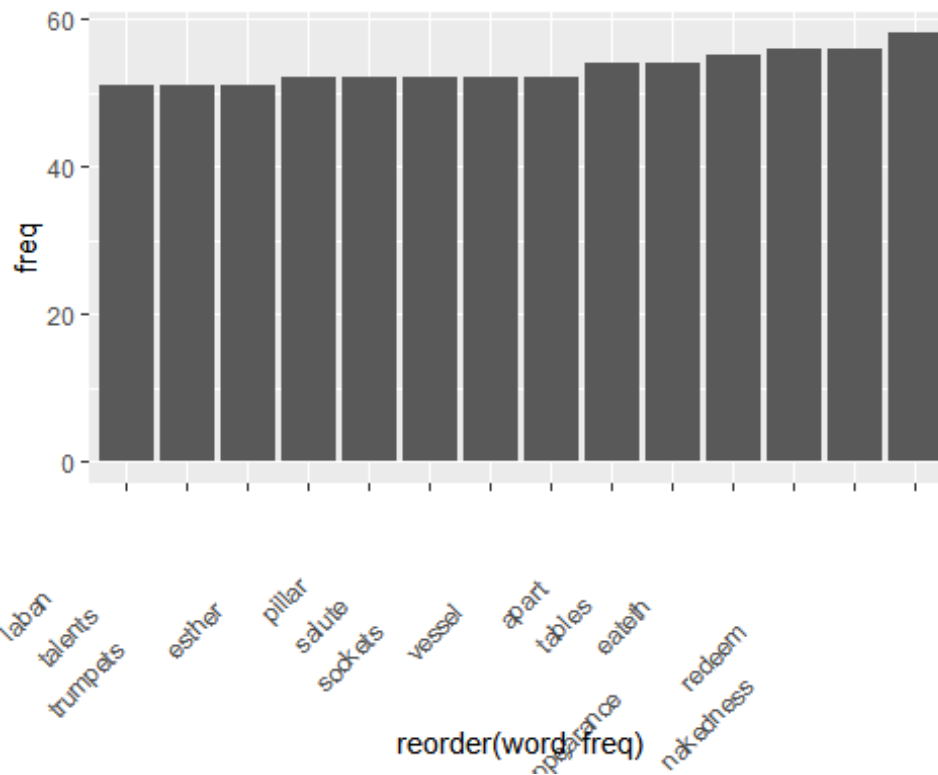
```
##          word freq
## nakedness  nakedness  58
## redeem      redeem    56
## appearance appearance  56
## eateth      eateth    55
## apart       apart     54
## tables      tables    54
```

```
##          word freq
## nakedness  nakedness  58
## redeem      redeem    56
## appearance appearance  56
## eateth      eateth    55
## apart       apart     54
## tables      tables    54
## vessel      vessel    52
## salute      salute    52
## sockets     sockets    52
## esther      esther    52
## pillar      pillar    52
## talents     talents    51
## trumpets    trumpets   51
## laban       laban     51
## ephah       ephah     50
## gift        gift      50
## next        next      50
## haman       haman     50
## towns       towns     50
## ephod       ephod     50
## weep        weep      49
## just        just      49
## forgiven    forgiven   49
## shepherds   shepherds  49
## azariah     azariah    49
## multiply    multiply   49
## spear       spear      49
```

## joash	joash	49
## distress	distress	49
## rehoboam	rehoboam	49
## looking	looking	48
## kind	kind	48
## weight	weight	48
## witnesses	witnesses	48
## lions	lions	48
## job	job	48
## dwelleth	dwelleth	48
## touched	touched	48
## thorns	thorns	48
## pharaohs	pharaohs	48
## rejected	rejected	48
## vengeance	vengeance	48
## teeth	teeth	47
## recompense	recompense	47
## steps	steps	47
## chamber	chamber	47
## sinners	sinners	47
## ishmael	ishmael	47
## boat	boat	47
## jesse	jesse	47
## countries	countries	47
## jealousy	jealousy	47
## gifts	gifts	46
## gladness	gladness	46
## removed	removed	46
## images	images	46
## dismayed	dismayed	46
## sixth	sixth	46
## asaph	asaph	46
## array	array	46
## veil	veil	46
## simeon	simeon	46
## flour	flour	46
## wouldest	wouldest	46
## doest	doest	46
## conceived	conceived	46
## glorified	glorified	46
## raiment	raiment	46
## interpretation	interpretation	46
## sackcloth	sackcloth	46
## profit	profit	46
## created	created	46
## staves	staves	46
## abideth	abideth	45
## bars	bars	45
## vineyards	vineyards	45
## instruments	instruments	45

## asher	asher	45
## herself	herself	45
## smoke	smoke	45
## sojourn	sojourn	45
## indignation	indignation	45
## naked	naked	44
## salt	salt	44
## pleasant	pleasant	44
## masters	masters	44
## sitteth	sitteth	44
## changed	changed	44
## building	building	44
## lies	lies	44
## satisfied	satisfied	44
## low	low	44
## forgotten	forgotten	44
## repaired	repaired	44
## ahaz	ahaz	44
## spirits	spirits	44
## valor	valor	44
## appoint	appoint	44
## issachar	issachar	44
## eastward	eastward	44

```
p<-
ggplot(subset(wf,freq>50),aes(x=reorder(word,freq),y=freq))+geom_bar(stat="id
entity")+
  theme(axis.text.x=element_text(angle=45,hjust=3))
p ; set.seed(150)
```



```
freq<-sort(colSums(as.matrix(dtm_b2)),decreasing = TRUE); head(freq,15)

##      the      and      that      unto      for      shall      his      they jehovah
## 58738 51682 13502  9096  9085  9071  8084  7569   6612
##      him      not      them      with      all      thou
## 6586  6543  6370  5960  5570  5477

wf<-data.frame(word=names(freq),freq=freq); head(wf)

##      word  freq
## the      the 58738
## and      and 51682
## that     that 13502
## unto     unto 9096
## for      for 9085
## shall   shall 9071

head(wf,100)

##      word  freq
## the      the 58738
## and      and 51682
## that     that 13502
## unto     unto 9096
## for      for 9085
## shall   shall 9071
## his      his 8084
## they     they 7569
```

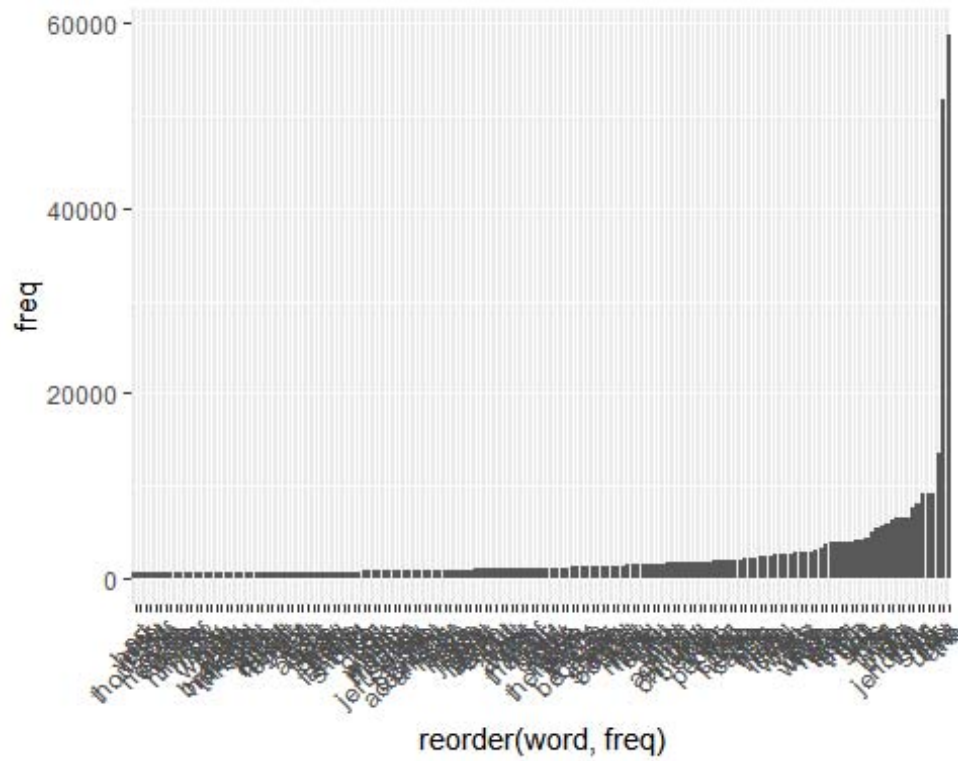
## jehovah	jehovah	6612
## him	him	6586
## not	not	6543
## them	them	6370
## with	with	5960
## all	all	5570
## thou	thou	5477
## thy	thy	4919
## was	was	4423
## will	will	4101
## god	god	4063
## but	but	3975
## their	their	3897
## said	said	3872
## from	from	3843
## thee	thee	3829
## have	have	3669
## are	are	3223
## which	which	2944
## upon	upon	2764
## were	were	2731
## out	out	2728
## this	this	2702
## when	when	2634
## you	you	2553
## israel	israel	2549
## man	man	2530
## there	there	2360
## son	son	2334
## hath	hath	2312
## king	king	2238
## one	one	2062
## came	came	2038
## house	house	1940
## into	into	1916
## come	come	1892
## had	had	1855
## her	her	1837
## people	people	1830
## your	your	1780
## then	then	1779
## before	before	1753
## land	land	1748
## children	children	1724
## day	day	1670
## men	men	1647
## against	against	1602
## shalt	shalt	1588
## also	also	1518
## who	who	1516

## let	let	1496
## even	even	1454
## hand	hand	1450
## made	made	1446
## now	now	1437
## went	went	1380
## lord	lord	1348
## behold	behold	1339
## saith	saith	1312
## saying	saying	1298
## these	these	1257
## our	our	1178
## because	because	1176
## sons	sons	1173
## things	things	1167
## every	every	1155
## down	down	1149
## therefore	therefore	1143
## make	make	1093
## after	after	1092
## may	may	1089
## david	david	1079
## say	say	1073
## over	over	1054
## thereof	thereof	1028
## forth	forth	1014
## she	she	1006
## what	what	985
## away	away	984
## hast	hast	976
## did	did	970
## put	put	958
## earth	earth	956
## name	name	939
## father	father	936
## great	great	935
## give	give	922
## jesus	jesus	917
## days	days	874
## take	take	873
## pass	pass	863
## heart	heart	860

p<-

```
ggplot(subset(wf, freq>500), aes(x=reorder(word, freq), y=freq))+geom_bar(stat="identity")+
  theme(axis.text.x=element_text(angle=45, hjust=1))
```

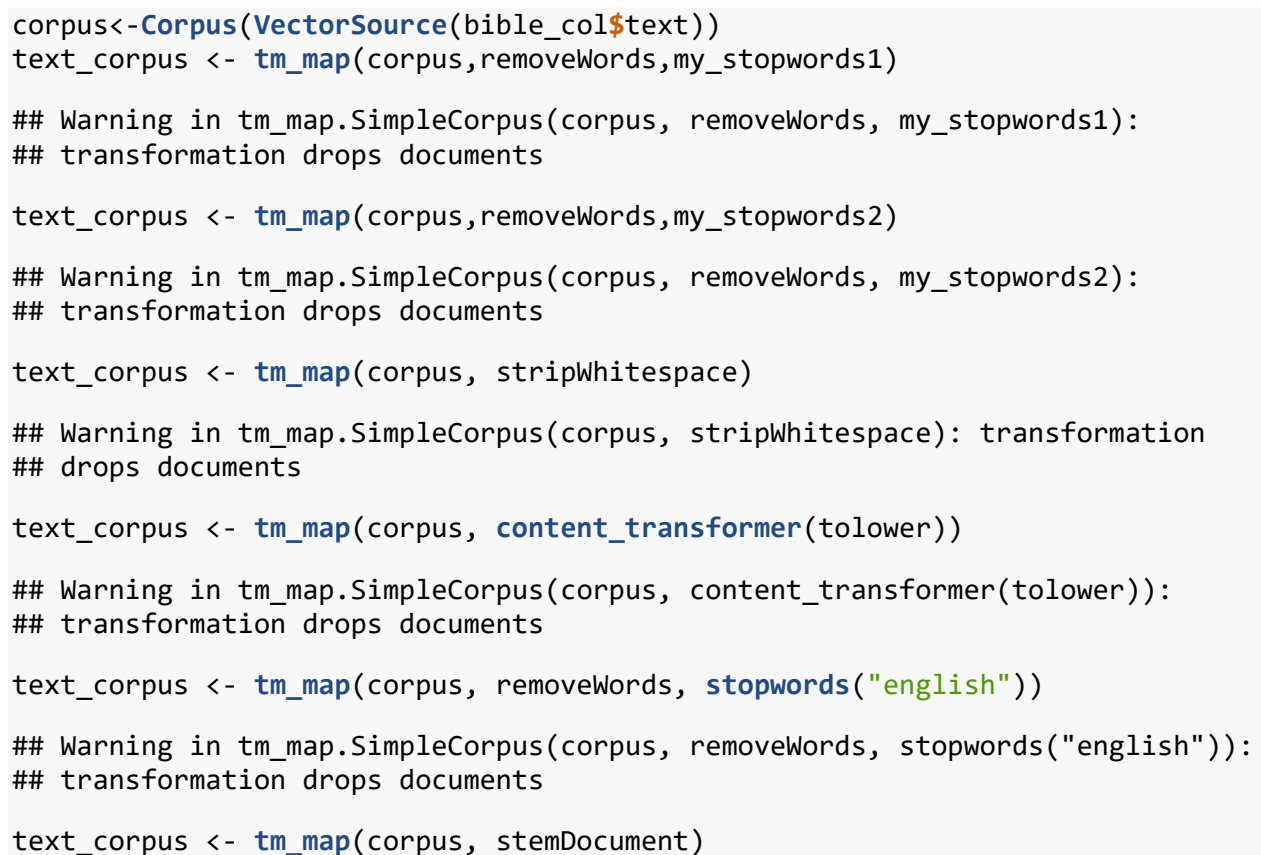
p



```
set.seed(150)
```

```
wordcloud(names(freq),freq,min.freq=100,max.words = 50,random.order =  
FALSE,rot.per = .35,  
          random.color=TRUE)
```





```

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

text_corpus <- tm_map(corpus, removeNumbers)

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

text_corpus <- tm_map(corpus, removePunctuation)

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

dtm_b2<-DocumentTermMatrix(text_corpus);dim(dtm_b2)

## [1]    66 27727

dtm_b22<-removeSparseTerms(dtm_b2,sparse=0.95);dim(dtm_b22)

## [1]    66 5269

dtmr <-DocumentTermMatrix(text_corpus, control=list(wordLengths=c(2, 20),
bounds = list(global = c(2,45))));dim(dtmr)

## [1]    66 10230

freq<-sort(colSums(as.matrix(dtmr)),decreasing = TRUE); head(freq,15)

##      jehovah      king      israel      land      david      she      pass
##      5870      2166      2150      1579      972      966      843
##      two      moses      took      jesus      judah      fathers  jerusalem
##      805      769      751      737      723      634      630
##      spake
##      614

wf<-data.frame(word=names(freq),freq=freq); head(wf); head(wf,10)

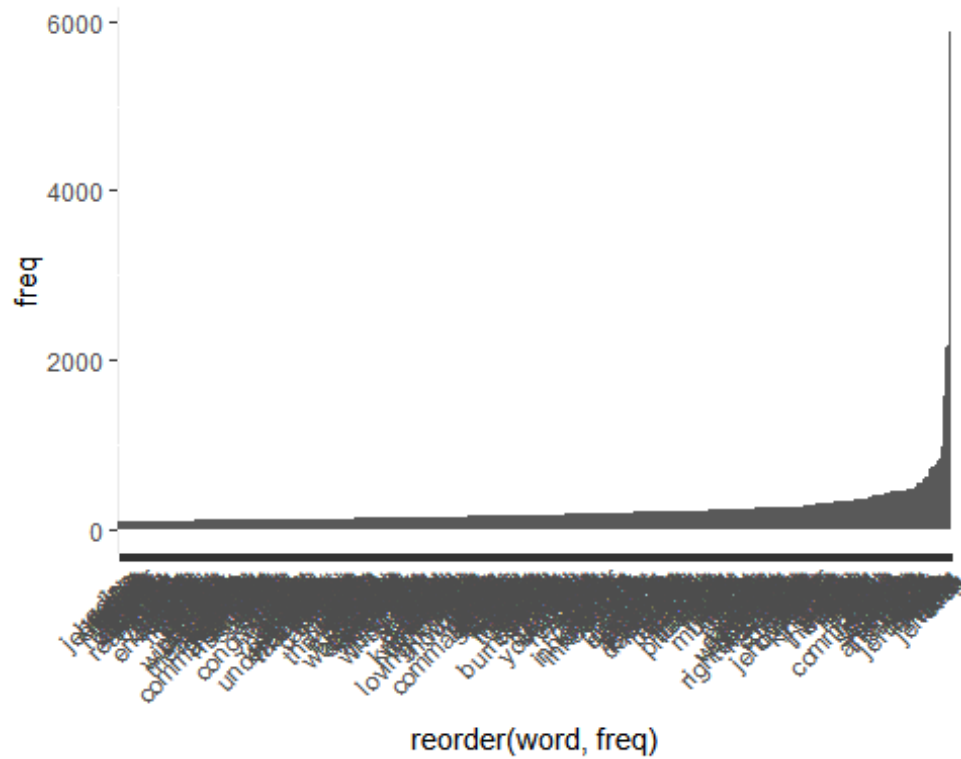
##           word freq
## jehovah jehovah 5870
## king      king 2166
## israel   israel 2150
## land     land 1579
## david    david  972
## she      she  966

##           word freq
## jehovah jehovah 5870
## king      king 2166
## israel   israel 2150
## land     land 1579
## david    david  972
## she      she  966

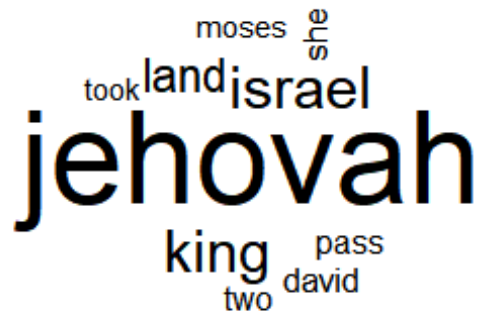
```

```
## pass      pass  843
## two       two   805
## moses     moses 769
## took      took  751
```

```
p<-
ggplot(subset(wf,freq>100),aes(x=reorder(word,freq),y=freq))+geom_bar(stat="identity")+
  theme(axis.text.x=element_text(angle=45,hjust=1))
p ; set.seed(142)
```



```
wordcloud(names(freq),freq,min.freq=50,max.words = 10,random.order =
FALSE,rot.per = .1,
  random.color=TRUE)
```



```
wordcloud(names(freq),freq,min.freq=50,max.words = 10,random.order =  
FALSE,rot.per = .35,  
          random.color=TRUE)
```



## Association Rules:

```
bible_dis<-discretizeDF(bible)
rules_bible<-apriori(bible_dis)

## Apriori
##
## Parameter specification:
## confidence minval smax arem aval originalSupport maxtime support minlen
##          0.8    0.1    1 none FALSE                TRUE         5     0.1    1
## maxlen target  ext
##          10 rules FALSE
##
## Algorithmic control:
## filter tree heap memopt load sort verbose
##       0.1 TRUE TRUE  FALSE TRUE     2     TRUE
##
## Absolute minimum support count: 3110
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[63095 item(s), 31103 transaction(s)] done [0.03s].
## sorting and recoding items ... [13 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 4 done [0.00s].
## writing ... [65 rule(s)] done [0.00s].
## creating S4 object ... done [0.02s].

summary(rules_bible)

## set of 65 rules
##
## rule length distribution (lhs + rhs):sizes
##  2  3  4
## 23 30 12
##
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      2.000  2.000  3.000  2.831  3.000  4.000
##
## summary of quality measures:
##      support      confidence      lift      count
##      Min.    :0.1452   Min.    :1   Min.    :1.344   Min.    : 4516
##      1st Qu.:0.1538   1st Qu.:1   1st Qu.:1.344   1st Qu.: 4785
##      Median :0.1881   Median :1   Median :3.000   Median : 5852
##      Mean   :0.2112   Mean    :1   Mean   :2.572   Mean    : 6568
##      3rd Qu.:0.2559   3rd Qu.:1   3rd Qu.:3.000   3rd Qu.: 7958
##      Max.   :0.3333   Max.    :1   Max.   :3.908   Max.   :10368
##
## mining info:
##      data ntransactions support confidence
## bible_dis      31103      0.1      0.8
```

```

subrules_bible<-rules_bible[quality(rules_bible)$confidence>0.5]
subrules_bible

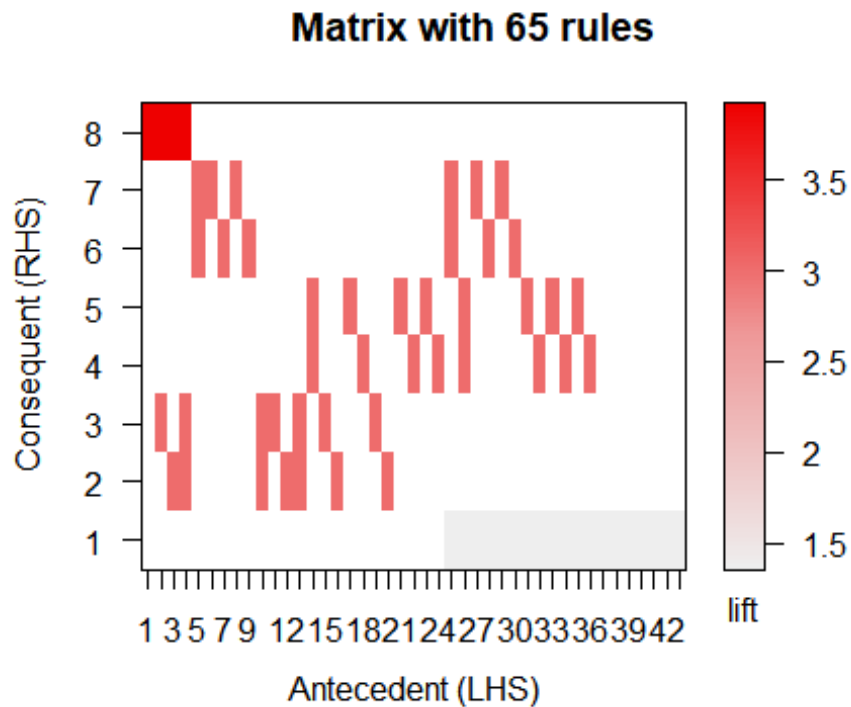
## set of 65 rules

plot(subrules_bible,method="matrix",measure = "lift")

## Itemsets in Antecedent (LHS)
## [1] "{X=[2.07e+04,3.11e+04],field=[2.6e+07,6.6e+07],Sections=Gospels}"
## [2] "{X=[2.07e+04,3.11e+04],Sections=Gospels}"
## [3] "{field=[2.6e+07,6.6e+07],Sections=Gospels}"
## [4] "{Sections=Gospels}"
## [5] "{Testaments=OT,Sections=Wisdom}"
## [6] "{X=[1.04e+04,2.07e+04),Testaments=OT}"
## [7] "{field=[1.3e+07,2.6e+07),Testaments=OT}"
## [8] "{X=[1.04e+04,2.07e+04),Testaments=OT,Sections=Wisdom}"
## [9] "{field=[1.3e+07,2.6e+07),Testaments=OT,Sections=Wisdom}"
## [10] "{Testaments=NT}"
## [11] "{X=[2.07e+04,3.11e+04]}"
## [12] "{field=[2.6e+07,6.6e+07]}"
## [13] "{Testaments=NT,Sections=Gospels}"
## [14] "{Testaments=OT,Sections=Law}"
## [15] "{X=[2.07e+04,3.11e+04),Testaments=NT}"
## [16] "{field=[2.6e+07,6.6e+07),Testaments=NT}"
## [17] "{X=[1,1.04e+04),Testaments=OT}"
## [18] "{field=[1e+06,1.3e+07),Testaments=OT}"
## [19] "{X=[2.07e+04,3.11e+04),Testaments=NT,Sections=Gospels}"
## [20] "{field=[2.6e+07,6.6e+07),Testaments=NT,Sections=Gospels}"
## [21] "{X=[1,1.04e+04),Testaments=OT,Sections=Law}"
## [22] "{field=[1e+06,1.3e+07),Testaments=OT,Sections=Law}"
## [23] "{X=[1,1.04e+04),Testaments=OT,Sections=History}"
## [24] "{field=[1e+06,1.3e+07),Testaments=OT,Sections=History}"
## [25] "{Sections=Wisdom}"
## [26] "{Sections=Law}"
## [27] "{X=[1.04e+04,2.07e+04]}"
## [28] "{field=[1.3e+07,2.6e+07]}"
## [29] "{X=[1.04e+04,2.07e+04),Sections=Wisdom}"
## [30] "{field=[1.3e+07,2.6e+07),Sections=Wisdom}"
## [31] "{X=[1,1.04e+04]}"
## [32] "{field=[1e+06,1.3e+07]}"
## [33] "{X=[1,1.04e+04),Sections=Law}"
## [34] "{field=[1e+06,1.3e+07),Sections=Law}"
## [35] "{X=[1,1.04e+04),Sections=History}"
## [36] "{field=[1e+06,1.3e+07),Sections=History}"
## [37] "{Sections=Prophets}"
## [38] "{Sections=History}"
## [39] "{X=[1.04e+04,2.07e+04),field=[1.3e+07,2.6e+07]}"
## [40] "{X=[1,1.04e+04),field=[1e+06,1.3e+07]}"
## [41] "{X=[1.04e+04,2.07e+04),field=[1.3e+07,2.6e+07),Sections=Wisdom}"
## [42] "{X=[1,1.04e+04),field=[1e+06,1.3e+07),Sections=Law}"

```

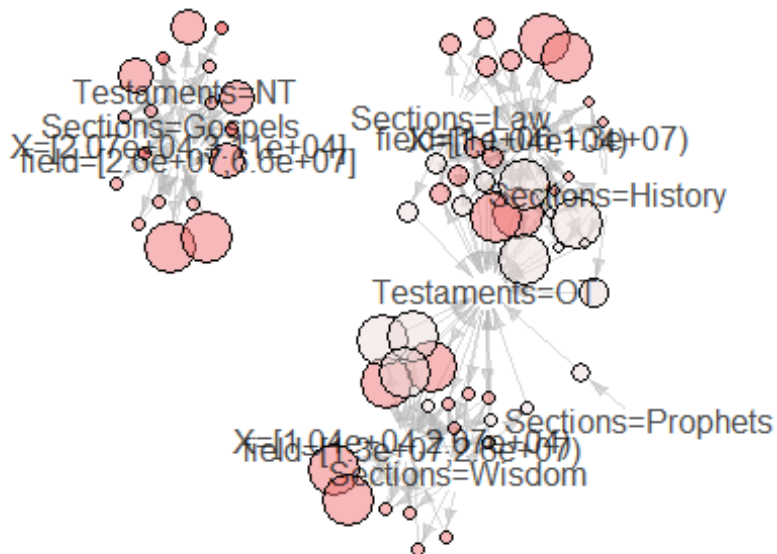
```
## [43] "{X=[1,1.04e+04),field=[1e+06,1.3e+07),Sections=History}"
## Itemsets in Consequent (RHS)
## [1] "{Testaments=OT}"           "{X=[2.07e+04,3.11e+04]}"
## [3] "{field=[2.6e+07,6.6e+07]}" "{X=[1,1.04e+04]}"
## [5] "{field=[1e+06,1.3e+07]}"   "{X=[1.04e+04,2.07e+04]}"
## [7] "{field=[1.3e+07,2.6e+07]}" "{Testaments=NT}"
```



```
subrules_bible2<-head(sort(rules_bible,by="lift"),66)
plot(subrules_bible2,method = "graph")
```

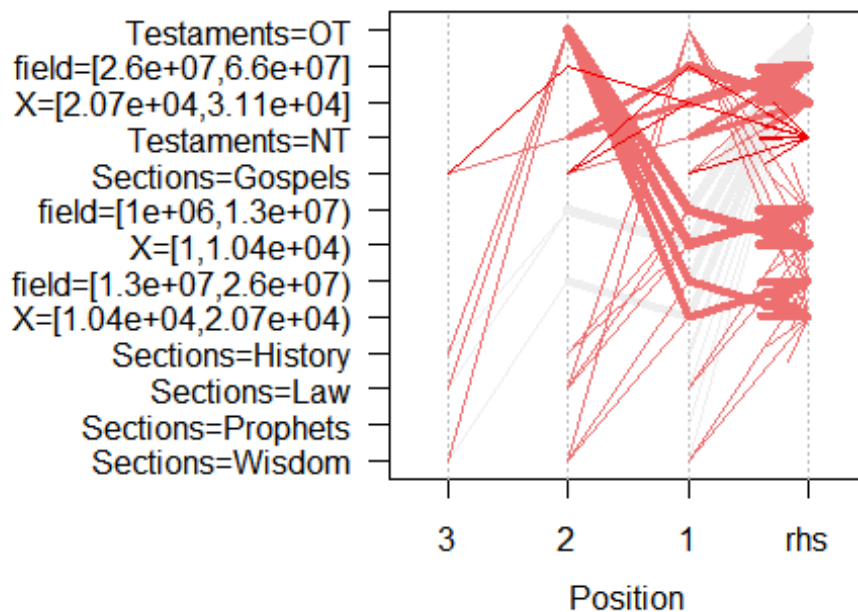
## Graph for 65 rules

size: support (0.145 - 0.333)  
color: lift (1.344 - 3.908)



```
plot(subrules_bible2, method="paracoord")
```

## Parallel coordinates plot for 65 rules

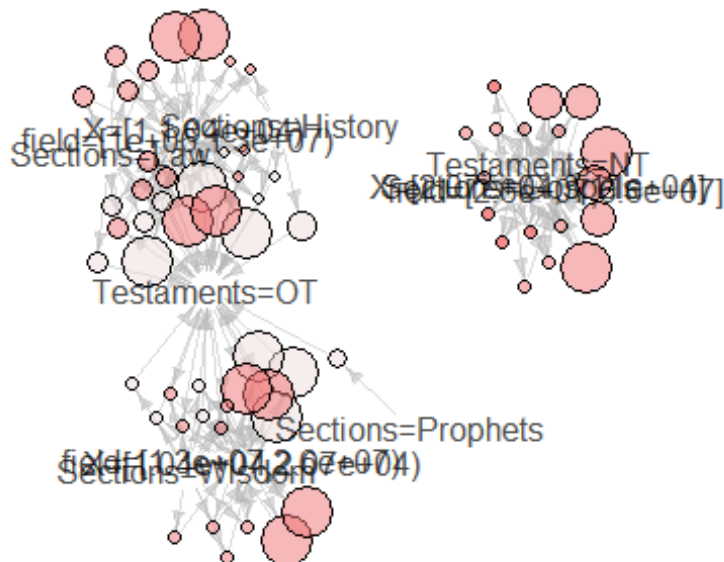


```
plot(rules_bible, method="graph")
```



## Graph for 65 rules

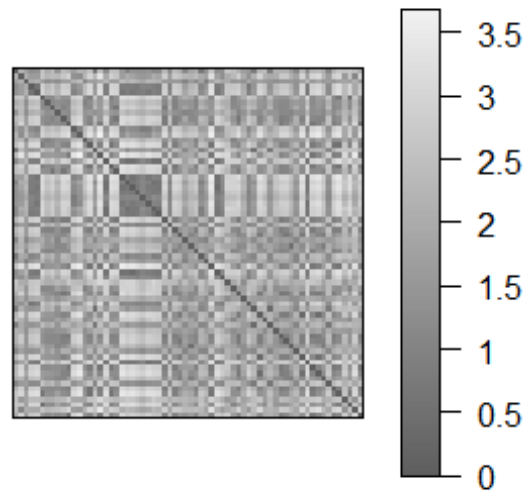
size: support (0.145 - 0.333)  
color: lift (1.344 - 3.908)



## Seriation Analysis:

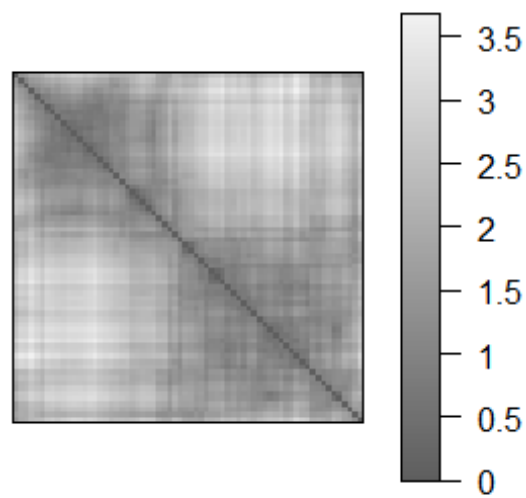
```
x<-as.matrix(csim_b)
x<-x[sample(seq_len(nrow(x))),]
d<-dist(x)
o<-seriate(d,method="OLO")
pimage(d,main="Original")
```

Original



```
pimage(d,o,main="Reordered")
```

Reordered

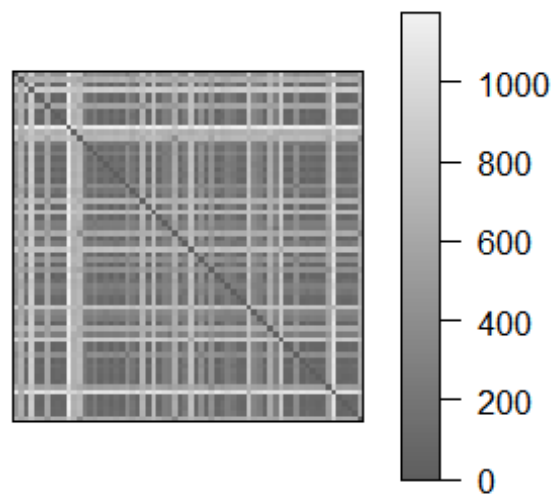


```
get_order(o)
```

```
## [1] 63 65 2 13 12 25 26 56 24 28 18 27 22 39 23 21 5 4 45 40 30 60 49
## [24] 46 16 1 36 54 37 3 33 9 53 34 59 32 52 55 47 58 51 43 10 29 20 8
## [47] 31 19 62 15 38 61 57 17 66 11 7 35 14 42 6 64 48 50 44 41
```

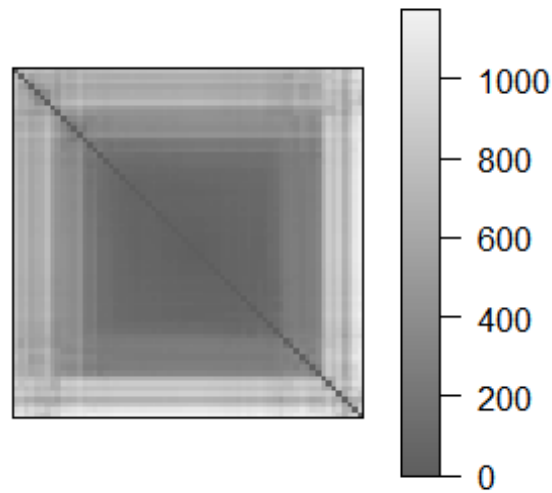
```
x1<-as.matrix(dtm_b)
x1<-x1[sample(seq_len(nrow(x1))),]
d1<-dist(x1)
o1<-seriate(d1,method="OLO")
pimage(d1,main="Original")
```

Original



```
pimage(d1,o1,main="Reordered")
```

## Reordered



```
get_order(o1)
```

```
## [1] 27 49 25 7 66 31 13 60 38 36 54 22 1 14 42 33 18 20 56 43 64 44 28  
## [24] 35 5 63 58 10 21 37 6 19 55 39 8 26 48 3 53 52 17 32 62 24 9 59  
## [47] 57 65 15 40 23 30 50 47 16 29 41 46 12 51 34 4 2 45 61 11
```

## Report

The bible was collapsed into 66 books of old and new testament. An analysis on bible was performed based on the 7 sections. From the analysis, it is evident that the words “the” is the most frequently repeated word followed by “Jehovah”.