Skytrax Airline Ratings and Reviews Analysis

Akhil Mathew

March 9, 2016

Introduction

This is an exploratory analysis on the airline ratings and reviews. Source of the data is http://www.airlinequality.com/. Airlines are segmented based on the ratings. Reviews are used to find out the reasons why certain airlines are under-rated. Various clustering techniques are used to segment the airlines. Topic modelling and Latent Dirichlet Allocation are used the text reviews.

Data Preparation

Loading Packages

```
suppressPackageStartupMessages({
require(stats)
library(dbscan)
library(plyr)
library(arm)
library(reshape2)
library(fpc)
library(dplyr)
library(Amelia)
library(cluster)
library(tm)
})
```

Read Data

```
airline <- read.csv("airline.csv", header = TRUE, na.strings = c("","NA"))
seat <- read.csv("seat.csv", header = TRUE)
```

Data Wrangling

```
airline <- tbl_df(airline)
airlines <- table(airline$airline_name)
aairlines <- data.frame(airlines)
colnames(aairlines) <- c("airline_name","repeats")
merge <- left_join(airline,aairlines,by="airline_name")

responses <- airline %>%
  group_by(airline_name) %>%
  summarise(response = length(which(overall_rating!="NA")))

merged <- left_join(merge,responses,by="airline_name")
merged_airline <- merged %>%
  mutate(percentage = response/repeats)
```

```
aairline <- merged_airline %>%
  filter(percentage >= 0.9,repeats >= 100)
airline_rating <- aairline[,c(1,12:16,19)]
```

Imputing Missing Values using EM Algorithm

```
aairline_imputed <- amelia(airline_rating[,c(2:7)])
```

```
## Warning: There are observations in the data that are completely missing.
##          These observations will remain unimputed in the final datasets.
## -- Imputation 1 --
##
##   1  2  3  4
##
## -- Imputation 2 --
##
##   1  2  3  4
##
## -- Imputation 3 --
##
##   1  2  3  4
##
## -- Imputation 4 --
##
##   1  2  3  4
##
## -- Imputation 5 --
##
##   1  2  3  4
```

```
aairline_imp <- aairline_imputed$imputations[[1]]
aairline_rating <- bind_cols(aairline[,c(1)],aairline_imp)
```

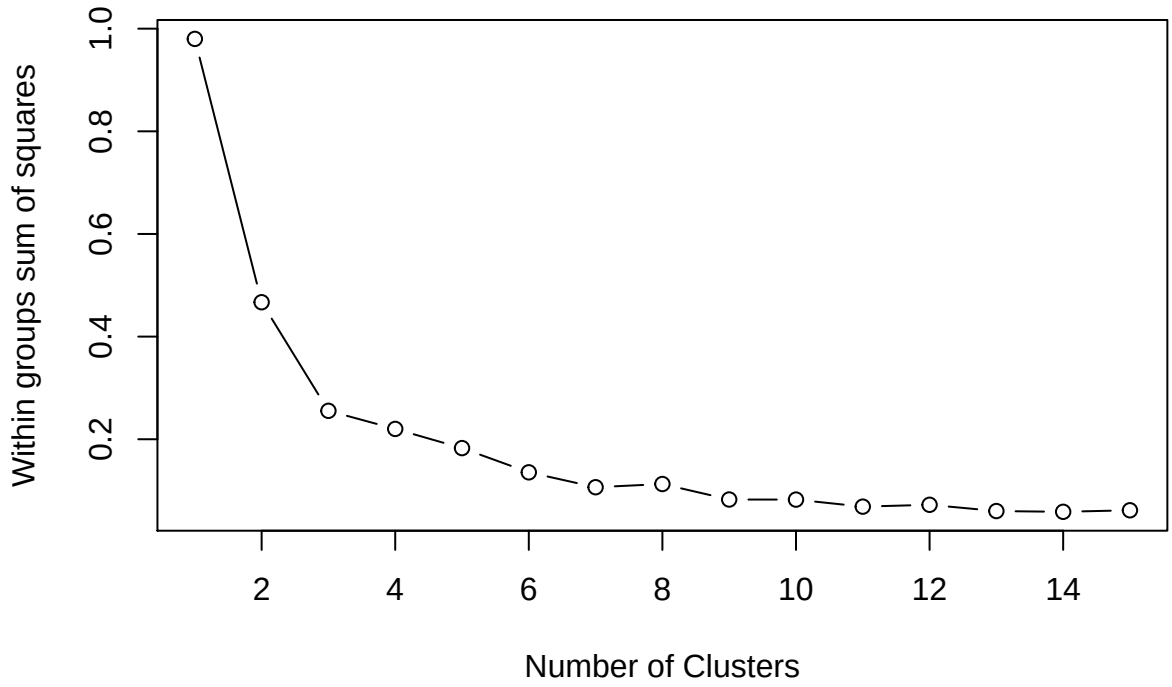
Normalization

```
mmnormalize <- function(a){
  m <- max(a,na.rm = TRUE)
  n <- min(a,na.rm = TRUE)
  mmnormalized <- a
  mmnormalized <- (mmnormalized-n)/(m-n)
  return (mmnormalized)
}

aairline_normalized <- mmnormalize(aairline_rating[,c(2:7)])
airline_proc <- bind_cols(aairline_rating[,c(1)],aairline_normalized)
airline_proc <- airline_proc[complete.cases(airline_proc),]

airline_data <- airline_proc %>%
  group_by(airline_name) %>%
  summarise_each(funs(mean))
```

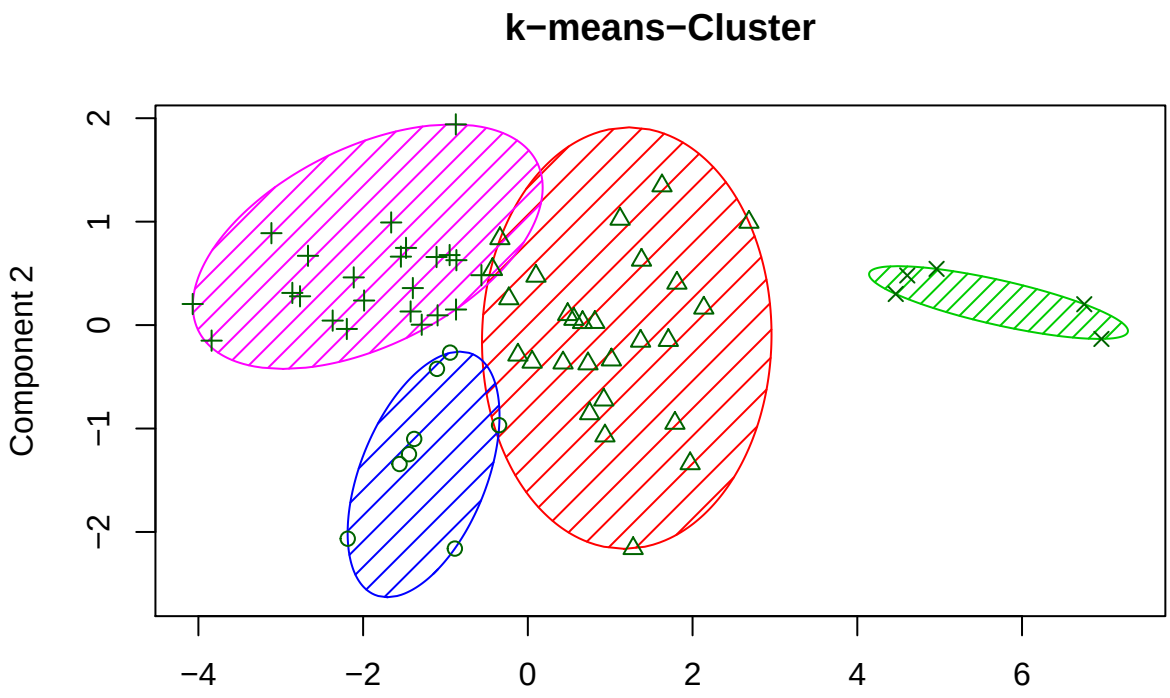
K-means Clustering

SSE Curve - Elbow curve gives us the idea about the number of clusters in the data.

```
wss <- (nrow(airline_data)-1)*sum(apply(airline_data[,c(2:7)],2,var))
for (i in 2:15) wss[i] <- sum(kmeans(airline_data[,c(2:7)],
                                     centers=i)$withinss)
plot(1:15, wss, type="b", xlab="Number of Clusters",
     ylab="Within groups sum of squares")
```




```
km <- kmeans(airline_data[,c(2:7)],4)
clusplot(airline_data[,c(2:7)],
         km$cluster,
         color = TRUE,
         shade = TRUE,
         lines = 47,
         main = "k-means-Cluster")
```



These two components explain 94.43 % of the point variability.

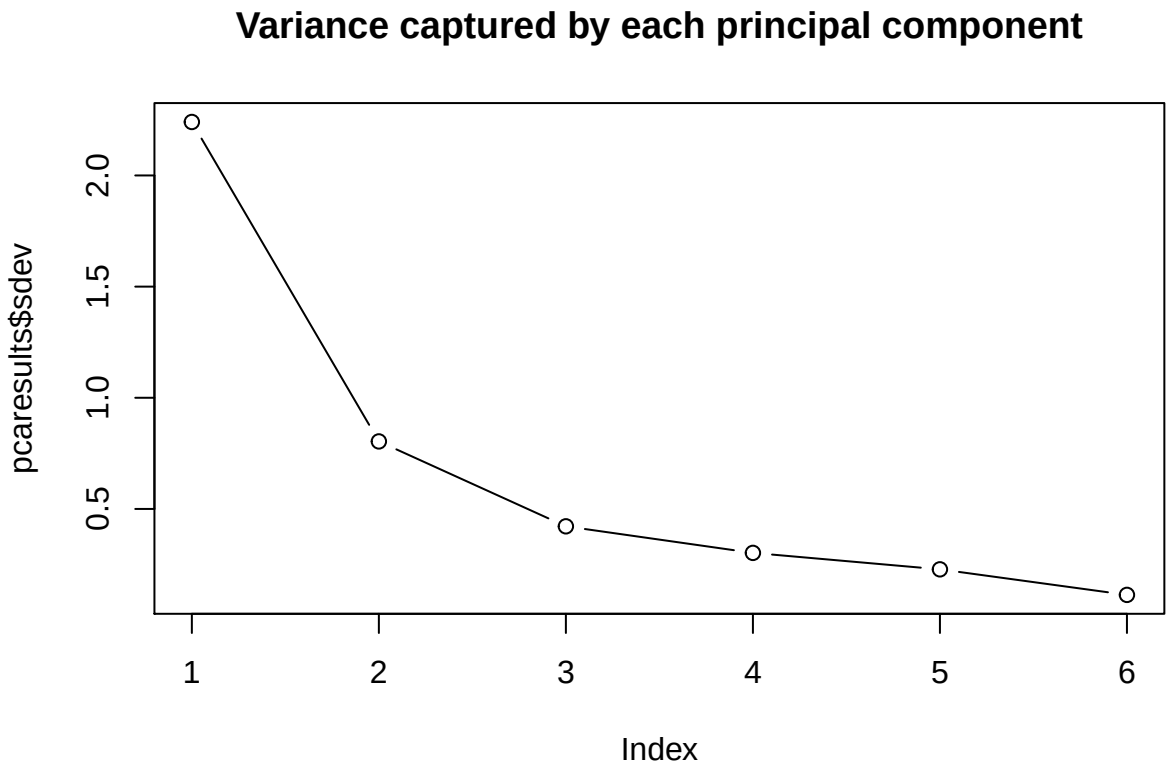
Principal Component Analysis

```
# =======
# Run PCA
# =======

pcaresults <- prcomp(airline_data[,c(2:7)], center = TRUE, scale. = TRUE)
#print(pcaresults)
summary(pcaresults)
```

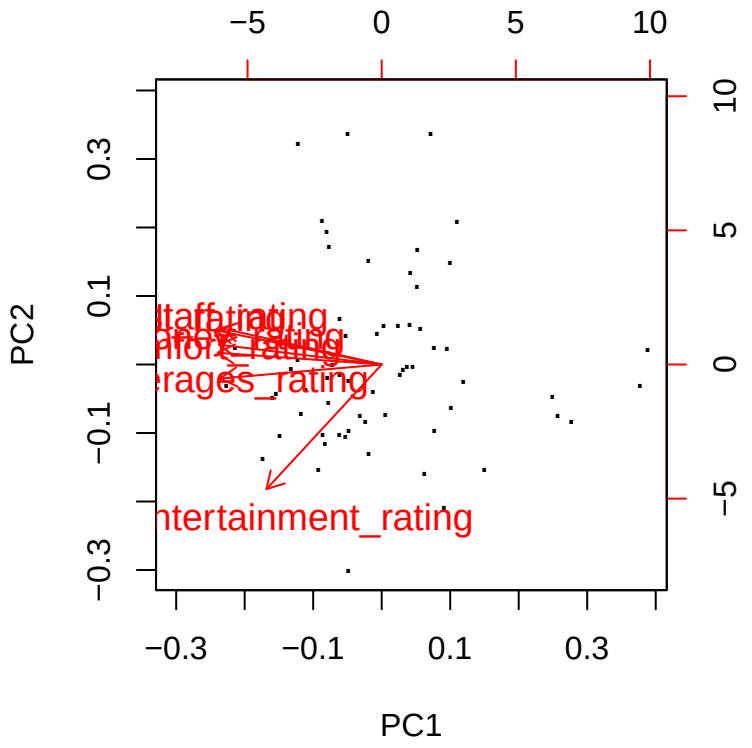
```
## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5     PC6
## Standard deviation     2.2405 0.8037 0.42178 0.30222 0.22827 0.11339
## Proportion of Variance 0.8366 0.1077 0.02965 0.01522 0.00868 0.00214
## Cumulative Proportion  0.8366 0.9443 0.97395 0.98917 0.99786 1.00000
```

```
pcadata <- as.data.frame(pcaresults$x)
#print(pcadata)
#summary(pcadata)

# =============
# Visualize PCA
```

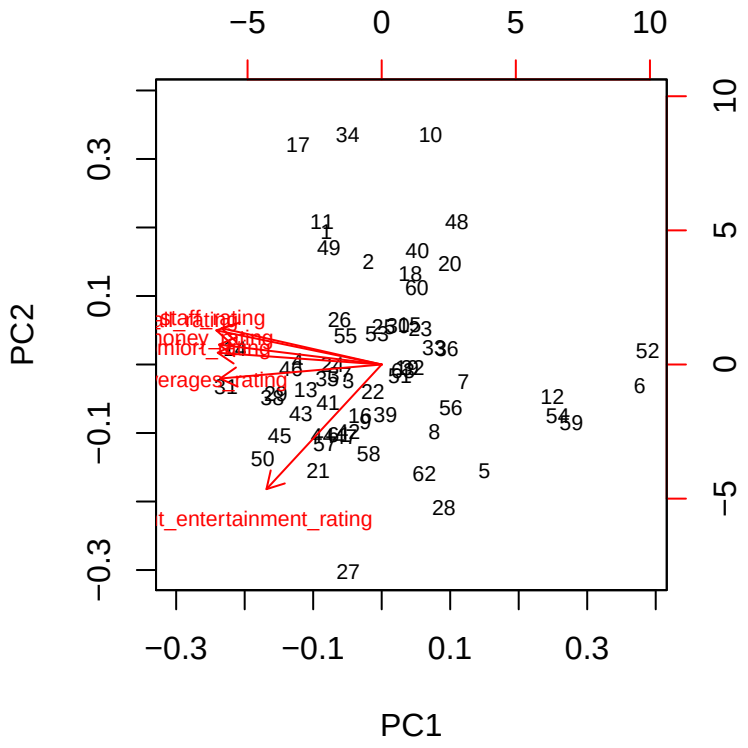
```r
# =============

# Show variances captured by principal components
# Type = "b" (stands for "both") shows both lines and points
plot(pcaresults$sdev, type="b",
     main = "Variance captured by each principal component")
```




```r
# Main idea: Use main two principal components
#   to visualize multi-dimensional data in one plot

# Use biplot functionality
labels <- rep(".", nrow(airline_data))
biplot(pcaresults, cex=1.2, xlabs=labels)
```




```
# Alternative labels:
labels <- 1:nrow(airline_data)
biplot(pcaresults, cex=0.7, xlabs=labels)
```




```
# Use ggplot2 for more advanced graphs
require(ggplot2)
```

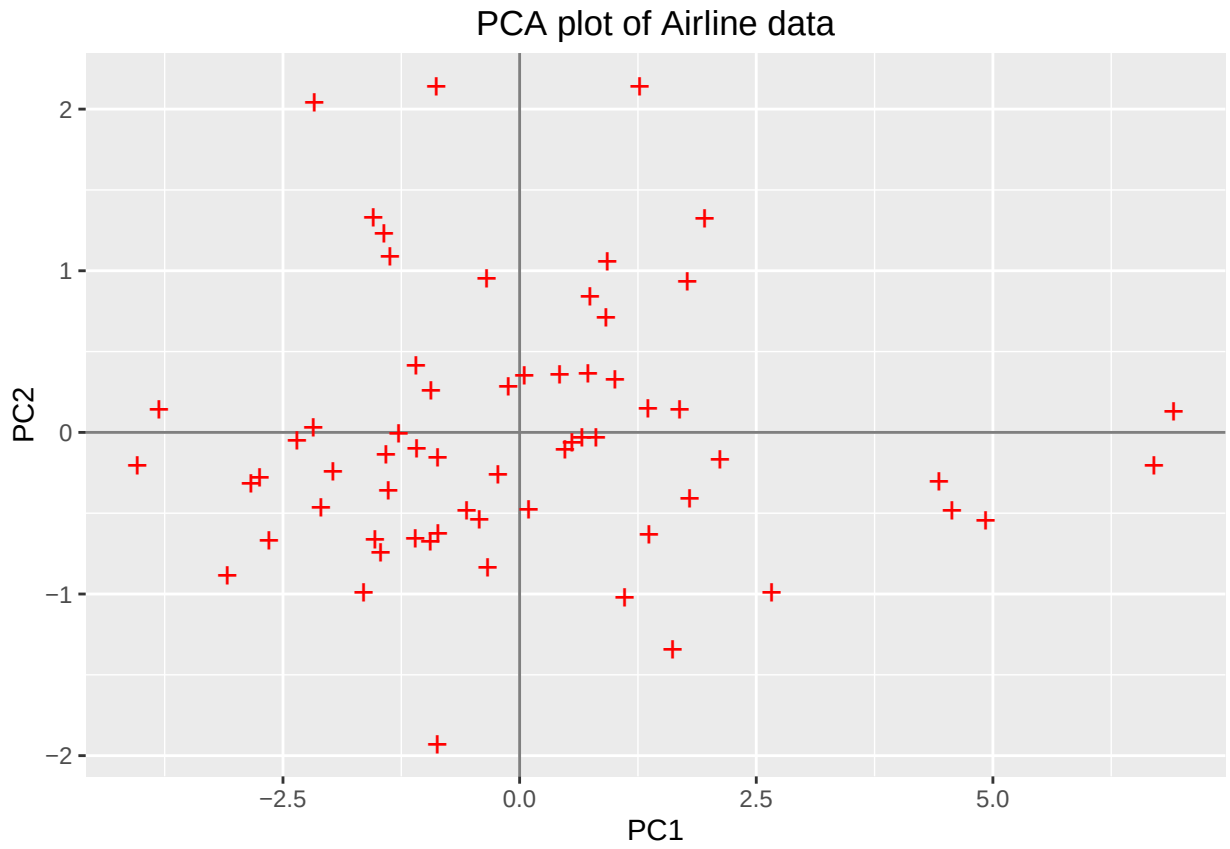
```
## Loading required package: ggplot2
```

```
## Warning: package 'ggplot2' was built under R version 3.2.3
```

```
##
## Attaching package: 'ggplot2'
```

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

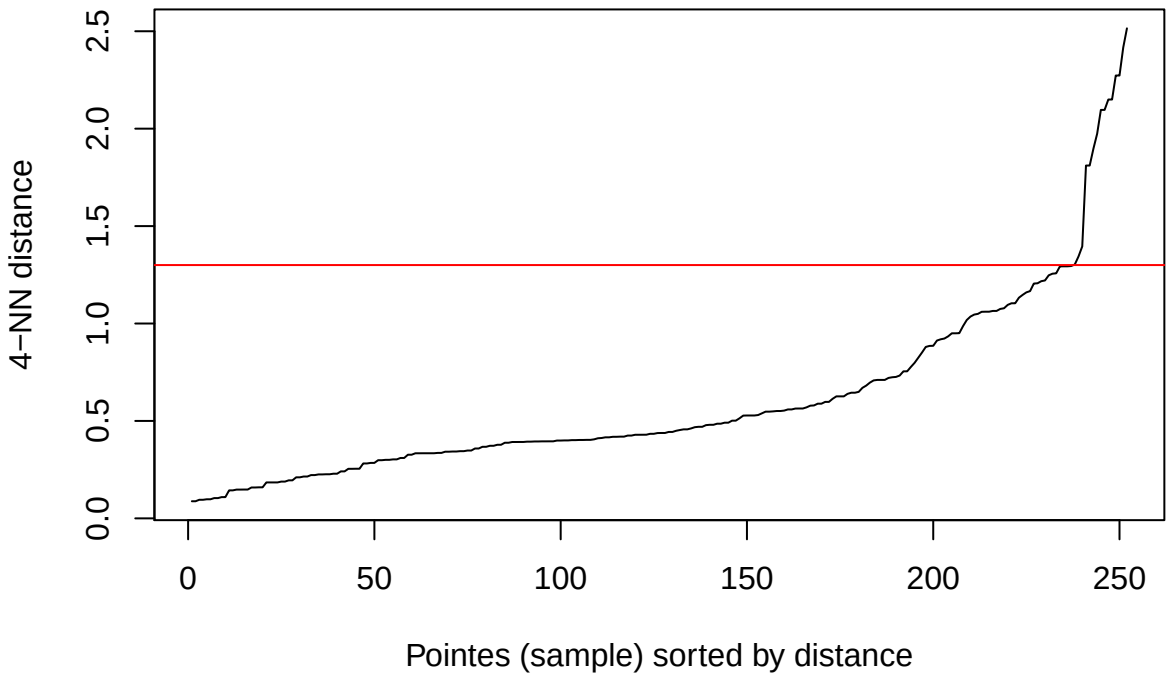
```
ggplot(data = pcadata, aes(x = PC1, y = PC2, label = "+")) +
  geom_hline(yintercept = 0, colour = "gray50") +
  geom_vline(xintercept = 0, colour = "gray50") +
  geom_text(colour = "red", size = 5) +
  ggtitle("PCA plot of Airline data")
```



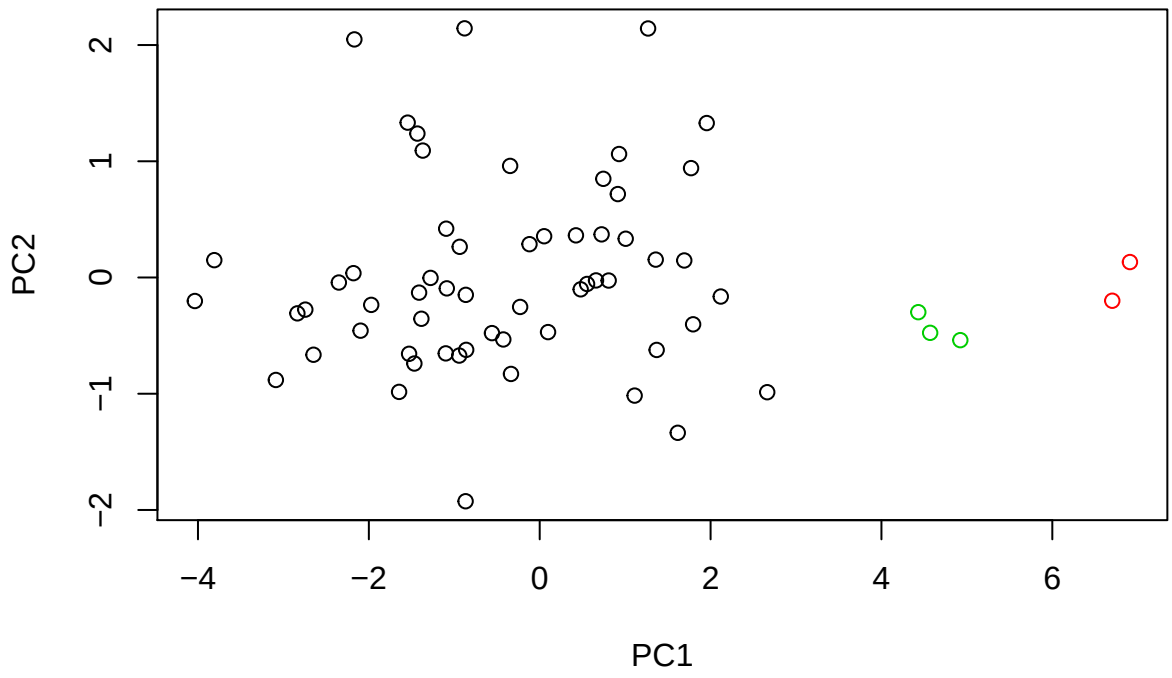

```
dbscandata <- pcadata[,c(1:2)]
```

DBSCAN Clustering

```
kNNdistplot(dbscandata, k = 4)
abline(h=1.3, col="red")
```




```
#run DBScan
db <- dbscan(dbscandata, eps=1.5, MinPts=2)
plot(dbscandata, col=db$cluster)
```



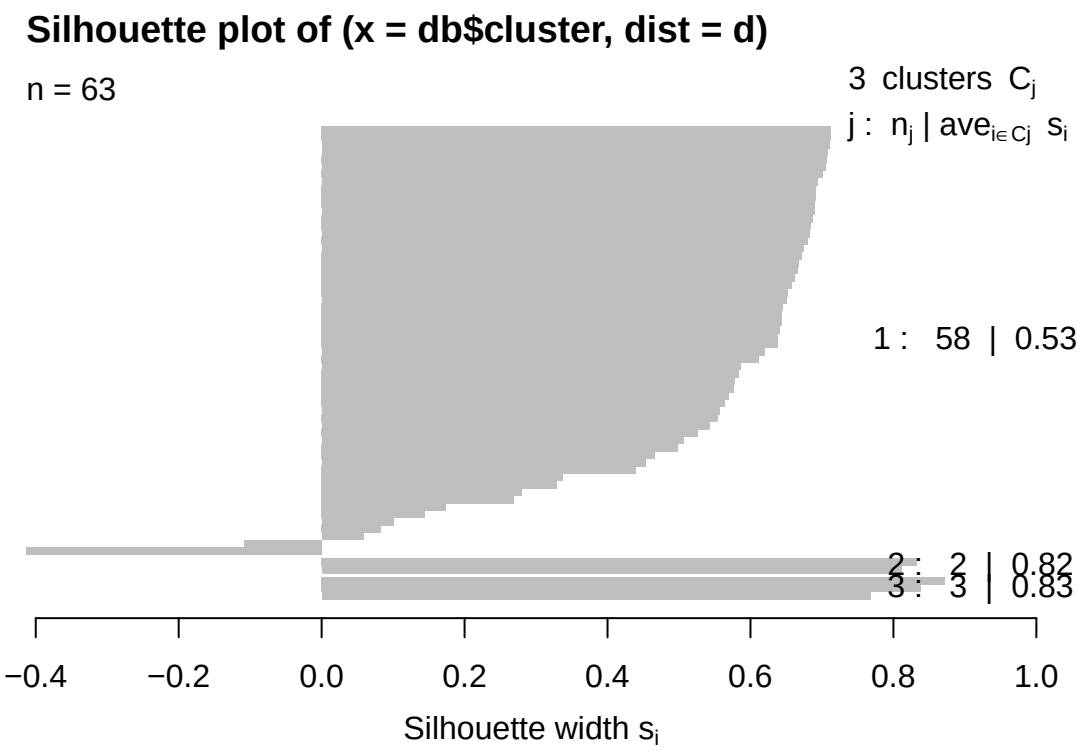

```
#play around with eps

#examine cluster1
#dbscandata[db$cluster==1,]

d <- dist(dbscandata)
#metrics
cluster.stats(d, db$cluster)
```

```
## $n
## [1] 63
##
## $cluster.number
## [1] 3
##
## $cluster.size
## [1] 58  2  3
##
## $min.cluster.size
## [1] 2
##
## $noisen
## [1] 0
##
## $diameter
## [1] 6.7456246 0.3920113 0.5477490
```

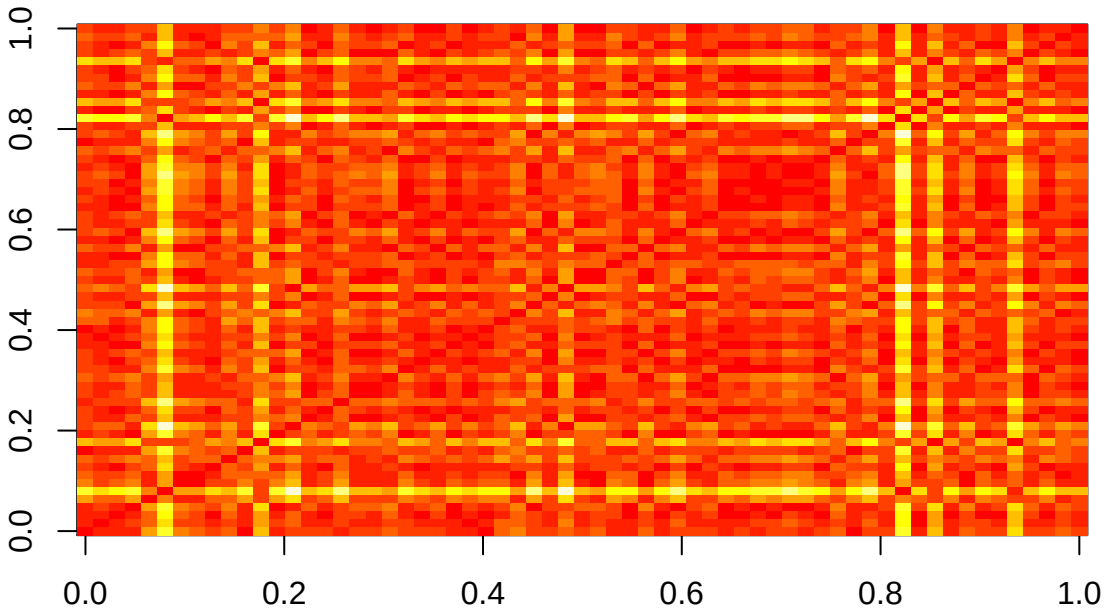
```
##
## $average.distance
## [1] 2.2023946 0.3920113 0.3773136
##
## $median.distance
## [1] 2.0649376 0.3920113 0.3586146
##
## $separation
## [1] 1.897351 1.810811 1.810811
##
## $average.toother
## [1] 6.060495 7.079694 5.113017
##
## $separation.matrix
##          [,1]     [,2]     [,3]
## [1,] 0.000000 4.114649 1.897351
## [2,] 4.114649 0.000000 1.810811
## [3,] 1.897351 1.810811 0.000000
##
## $ave.between.matrix
##          [,1]     [,2]     [,3]
## [1,] 0.000000 7.331566 5.213115
## [2,] 7.331566 0.000000 2.210182
## [3,] 5.213115 2.210182 0.000000
##
## $average.between
## [1] 5.982448
##
## $average.within
## [1] 2.197998
##
## $n.between
## [1] 296
##
## $n.within
## [1] 1657
##
## $max.diameter
## [1] 6.745625
##
## $min.separation
## [1] 1.810811
##
## $within.cluster.ss
## [1] 180.3458
##
## $clus.avg.silwidths
##         1         2         3
## 0.5310276 0.8219948 0.8261878
##
## $avg.silwidth
## [1] 0.5543199
##
## $g2
```

```
## NULL
##
## $g3
## NULL
##
## $pearsongamma
## [1] 0.7103333
##
## $dunn
## [1] 0.2684423
##
## $dunn2
## [1] 1.003536
##
## $entropy
## [1] 0.3306296
##
## $wb.ratio
## [1] 0.3674077
##
## $ch
## [1] 28.43443
##
## $cwidegap
## [1] 1.2172544 0.3920113 0.3586146
##
## $widestgap
## [1] 1.217254
##
## $sindex
## [1] 1.914471
##
## $corrected.rand
## NULL
##
## $vi
## NULL
```

```
plot(silhouette(db$cluster, d))
```

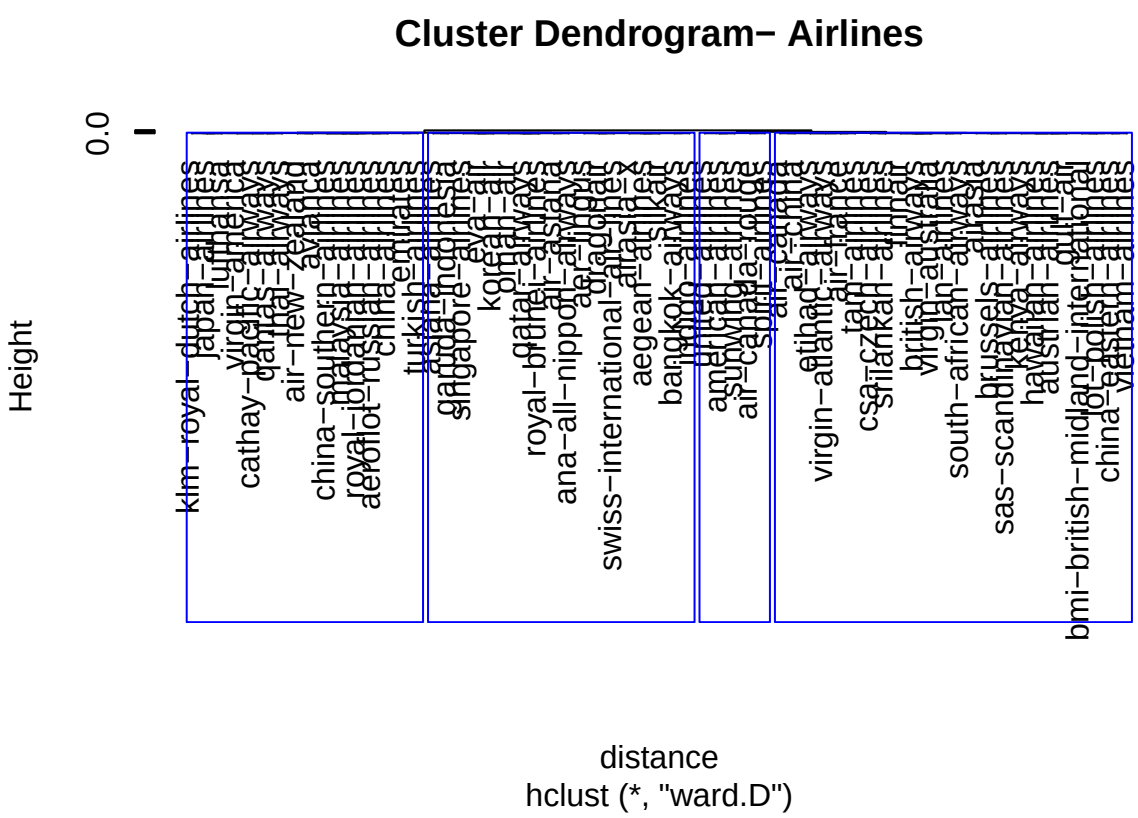



```
image(as.matrix(d))
```



Hierarchical Clustering

```
names <- as.vector(airline_data$airline_name)
distance <- dist(airline_data[,c(2:7)], method = "euclidean")
hcluster <- hclust(distance,method = "ward.D")
plot(hcluster,labels =names, main = "Cluster Dendrogram- Airlines")
rect.hclust(hcluster,k=4,border = "blue")
```



Topic Modelling

Preparing Data for Reviews Analysis

```
groups <- cutree(hcluster, k=6)
clustered_data <- cbind(airline_data[,c(1:7)], groups)
clustered <- left_join(clustered_data,aairline, by="airline_name")

g1 <- subset(clustered, groups == 1)
g2 <- subset(clustered, groups == 2)
g3 <- subset(clustered, groups == 3)
g4 <- subset(clustered, groups == 4)
g5 <- subset(clustered, groups == 5)
g6 <- subset(clustered, groups == 6)
g7 <- subset(g5, g5$airline_name == "air-canada-rouge" & g5$recommended == 0)
g8 <- subset(g5, g5$airline_name == "sunwing-airlines" & g5$recommended == 0)
g9 <- subset(g5, g5$airline_name == "spirit-airlines" & g5$recommended == 0)
g10 <- subset(g5, g5$airline_name == "american-airlines" & g5$recommended == 0)
g11 <- subset(g5, g5$airline_name == "united-airlines" & g5$recommended == 0)
```

Data Pre-processing for Topic Modelling

```
comment <- Corpus(DirSource(dirname, encoding = "UTF-8"))

meta(comment[[1]])
```

```
##   author       : character(0)
##   datetimestamp: 2016-03-10 03:36:15
##   description  : character(0)
##   heading      : character(0)
##   id           : comment1.txt
##   language     : en
##   origin       : character(0)
```

```
# The following steps pre-process the raw text documents.
# Remove punctuations and numbers because they are generally uninformative.
comment <- tm_map(comment, removePunctuation)
comment <- tm_map(comment, removeNumbers)
# Convert all words to lowercase.
comment <- tm_map(comment, content_transformer(tolower))
# Remove stopwords such as "a", "the", etc.
comment <- tm_map(comment, removeWords, stopwords("english"))

comment <- tm_map(comment, removeWords, c("flight","airline","plane"))
# Use the SnowballC package to do stemming.
library(SnowballC)
```

```
## Warning: package 'SnowballC' was built under R version 3.2.2
```

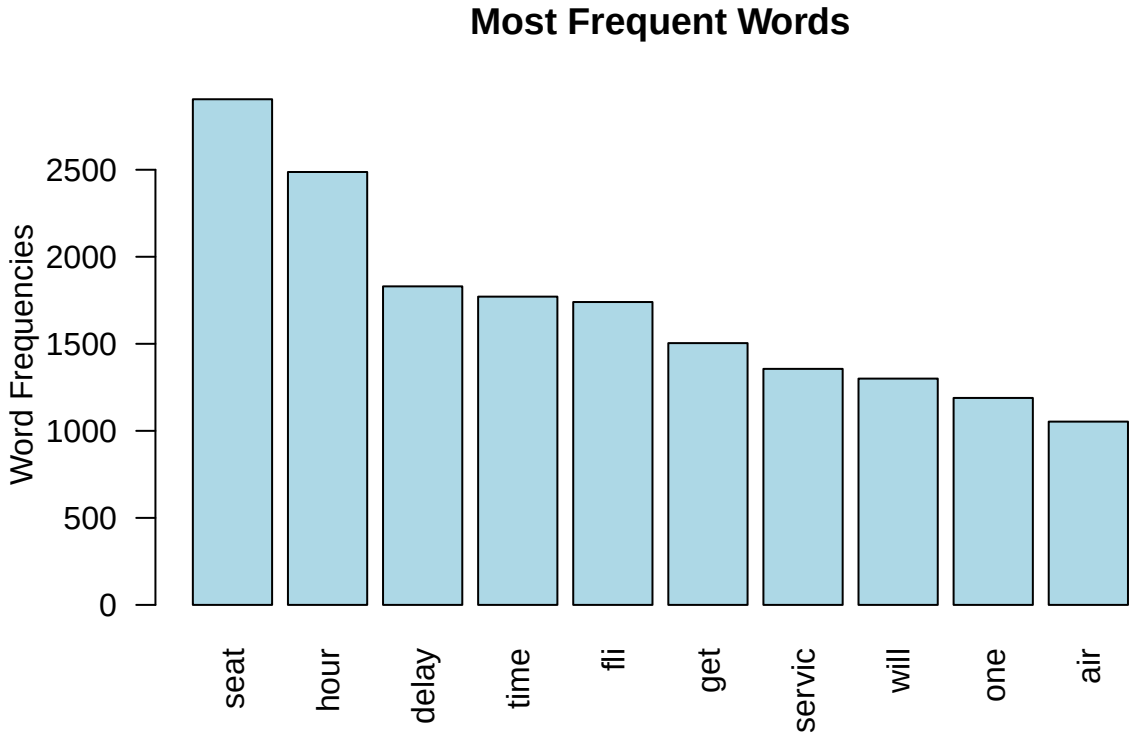
```
comment <- tm_map(comment, stemDocument)
# Remove excess white spaces between words.
comment <- tm_map(comment, stripWhitespace)
# Inspect the first document to see what it looks like.
#comment[["comment1.txt"]]$content
# Convert all documents to a term frequency matrix.
tfm <- DocumentTermMatrix(comment)
# We can check the dimension of this matrix by calling dim()
#print(dim(tfm))
```

Initial Analysis

```
dtm <- TermDocumentMatrix(comment)
m <- as.matrix(dtm)
v <- sort(rowSums(m),decreasing=TRUE)
d <- data.frame(word = names(v),freq=v)
#head(d, 10)

#findAssocs only works when there's more than one document
findAssocs(dtm, terms = "uncomfortable", corlimit = 0.95)
```

```
## $uncomfortable
## numeric(0)
```

```
barplot(d[1:10,]$freq, las = 2, names.arg = d[1:10,]$word,
        col ="lightblue", main ="Most Frequent Words",
        ylab = "Word Frequencies")
```



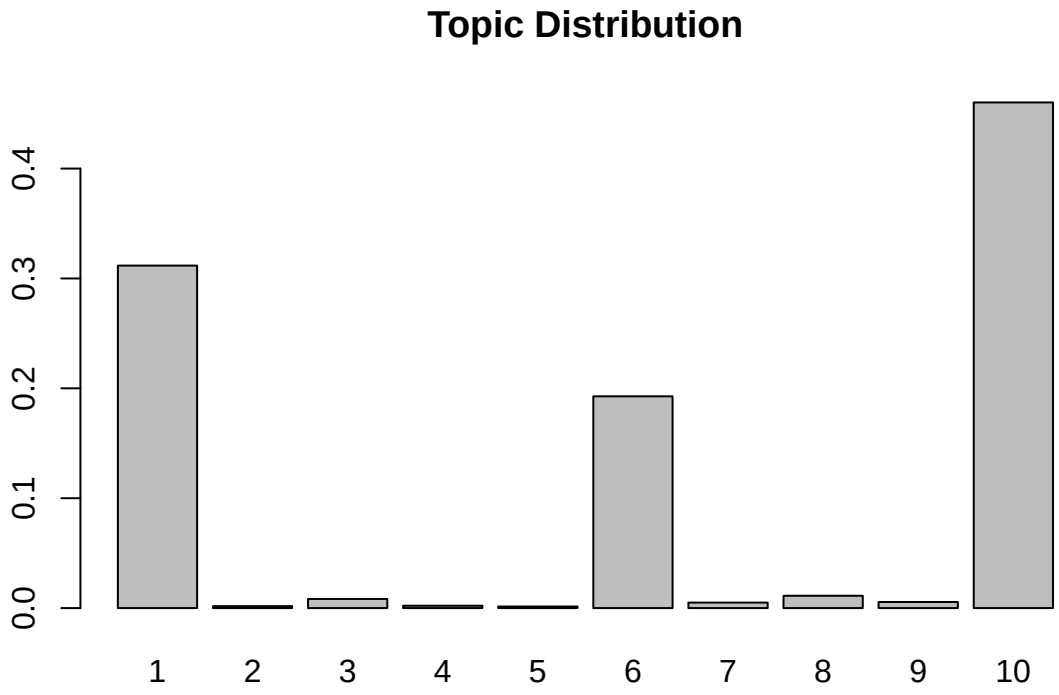
Topic Modelling

```
# Use topicmodels package to conduct LDA analysis.
library(topicmodels)
```

```
## Warning: package 'topicmodels' was built under R version 3.2.2
```

```
results <- LDA(tfm, k = 10, method = "Gibbs")
# Obtain the top five words (i.e., the 5 most probable words) for each topic.
Terms <- terms(results, 20)
# Obtain the most likely topic assignment for each document.
Topic <- topics(results, 1)
# Get the posterior probability for each document over each topic
posterior <- posterior(results)[[2]]
```

```r
#look at the posterior topic distribution for the first document and plot it visually
#posterior(results)[[2]][1,]
barplot(posterior(results)[[2]][1,],main = "Topic Distribution")
```




```r
# Calculate the entropy for each document to quantify keyword ambiguity
CalcEntropy <- function(document) {
  entropy = 0
  for (i in 1:length(document)) {
    entropy = entropy - document[i]*log(document[i])
  }
  return(entropy)
}

Entropy <- apply(posterior, 1, CalcEntropy) #posterior is matrix, 1 indicates rows
newKeywordConstruct <- data.frame(Entropy, Topic)
```

Reflections

It was a deep exploration. We have found the segments in the airlines, especially the poorly rated ones. Focusing our analysis onto those airlines pointed us to a few reasons why they are under rated. Recommendations are formuled according to these findings.Please find it in the presentation included.