Lab Week 4

STAT5003

Dr. Justin Wishart

Semester 2, 2022

Contents

Preparation and assumed knowledge

- High dimensional viz content in Module 4.
- Listen to the Week 4 lecture pre-recording.
- Data files
  - `movielens_top40.csv` from Canvas
  - `author_count.csv` from Canvas

Aims

- Explore decompositions of data using
  - different PCA calibrations
  - different clustering calibrations using k-means and hierarchical clustering.
  - different representations of the data using t-SNE and MDS
- Create a visualizations using PCA, t-SNE and basic MDS
- Understand the difference between clustering algorithms and data visualization.

1 Movie ratings data

We will be analysing the MovieLens dataset which contains movie ratings of 58,000 movies by 280,000 users. The entire dataset is too big for us to work with in this lab. It has been preprocessed with only a small

subset of the data being considered. If you want to do more exploration yourself, the entire dataset can be downloaded here.

This part of the lab is based on a chapter in an online book by Rafael Irizarry. You can find it here. There are lots of examples in this book to show you how to use R for data science.

1.1 Data processing [optional]

This part of the code is for interested students only. You do not need this for the lab.

```
# Here is the code used to preprocess the data (taken from the Irizarry lab):
library(dplyr)
library(tidyr)
ratings <- read.csv("ml-latest-small/ratings.csv", header = TRUE)
movies <- read.csv("ml-latest-small/movies.csv", header = TRUE)
movielens <- left_join(movies, ratings)

top <- movielens %>%
  group_by(movieId) %>%
  summarize(n=n(), title = first(title)) %>%
  top_n(40, n) %>%
  pull(movieId)

x <- movielens %>%
  filter(movieId %in% top) %>%
  group_by(userId) %>%
  filter(n() >= 20) %>%
  ungroup() %>%
  select(title, userId, rating) %>%
  spread(userId, rating)
x <- as.data.frame(x)
rownames(x) <- x$title
x$title <- NULL
colnames(x) <- paste0("user_", colnames(x))


write.table(x, row.names = TRUE, col.names = TRUE, sep = ",", file = "movielens_top40.csv")
```

1.2 Data input and IDA

Load the data `movielens_top40.csv` into R. It contains the top 40 movies with the most ratings and users who rated at least 20 out of the 40 movies. Note, IDA refers to initial data analysis. This is important component for all data analytics.

```
movielens <- read.csv("movielens_top40.csv", header = TRUE)
dim(movielens)
```

```
## [1]  40 153
```

```
range(movielens, na.rm = TRUE)
```

```
## [1] 0.5 5.0
```

In this case, the data is structured in the opposite of a typical data layout. whereby the variables of interest are the movies and they appear on the rows and the user response values appear as the columns. This is done somewhat intentionally for the distance calculations coming soon that computes the pairwise distances, where the pairing is done by row.

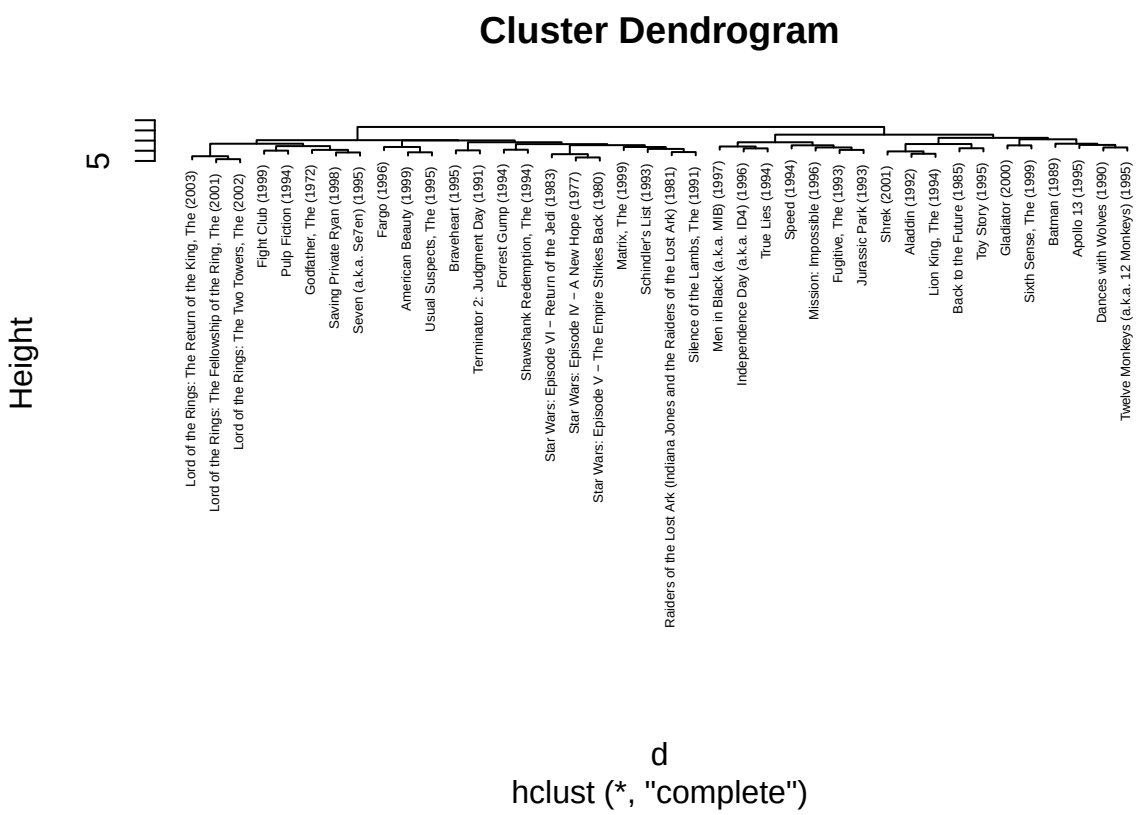
1.3 Hierarchical clustering

Given the large amount of variables, a natural high-dimensional visualization method is to cluster the movies based on different user ratings. We will look at how to do this in R.

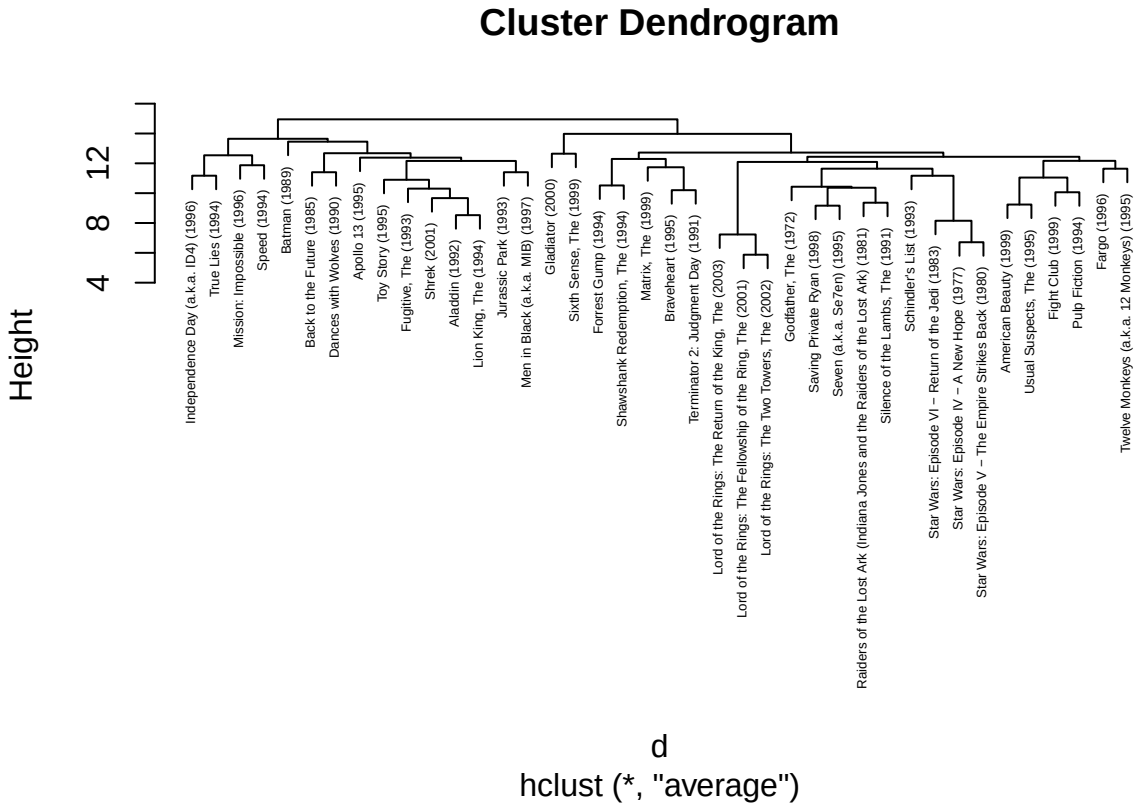
1. Basic `hclust` usage

Perform hierarchical clustering using the `hclust()` function and plot the resulting dendrogram. Try it with the `average`, `complete` and `single` methods.

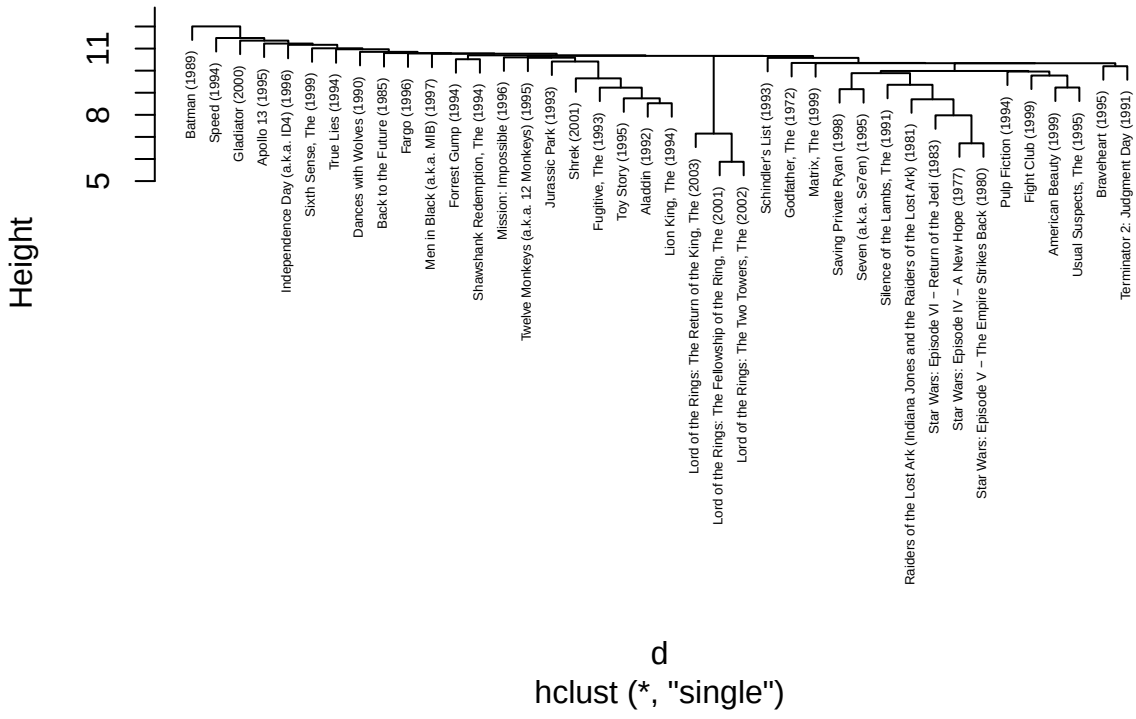
```
d <- dist(movielens)
h <- hclust(d)
plot(h, cex = 0.4)
```




```
h_avg <- hclust(d, method = "average")
plot(h_avg, cex = 0.4)
```




```
h_single <- hclust(d, method = "single")
plot(h_single, cex = 0.4)
```



2. Form clusters in `hclust`

Use the `cutree()` function on the output of `hclust()` (with default settings) to separate the movie titles into four clusters. Can you extract the movies in cluster 1? We can also cut the tree by defining a height at which the tree should be cut. Can you find the value of `h` to cut the tree into four clusters?

```
movie_groups <- cutree(h, k = 4)
head(movie_groups)
```

```
##            Aladdin (1992)    American Beauty (1999)          Apollo 13 (1995)
##                         1                         2                         3
## Back to the Future (1985)             Batman (1989)         Braveheart (1995)
##                         1                         3                         2
```

```
split(names(movie_groups), movie_groups)
```

```
## $`1`
## [1] "Aladdin (1992)"            "Back to the Future (1985)"
## [3] "Lion King, The (1994)"     "Shrek (2001)"
## [5] "Toy Story (1995)"
##
## $`2`
##  [1] "American Beauty (1999)"
##  [2] "Braveheart (1995)"
##  [3] "Fargo (1996)"
##  [4] "Fight Club (1999)"
##  [5] "Forrest Gump (1994)"
##  [6] "Godfather, The (1972)"
##  [7] "Lord of the Rings: The Fellowship of the Ring, The (2001)"
##  [8] "Lord of the Rings: The Return of the King, The (2003)"
##  [9] "Lord of the Rings: The Two Towers, The (2002)"
## [10] "Matrix, The (1999)"
## [11] "Pulp Fiction (1994)"
## [12] "Raiders of the Lost Ark (Indiana Jones and the Raiders of the Lost Ark) (1981)"
## [13] "Saving Private Ryan (1998)"
## [14] "Schindler's List (1993)"
## [15] "Seven (a.k.a. Se7en) (1995)"
## [16] "Shawshank Redemption, The (1994)"
## [17] "Silence of the Lambs, The (1991)"
## [18] "Star Wars: Episode IV - A New Hope (1977)"
## [19] "Star Wars: Episode V - The Empire Strikes Back (1980)"
## [20] "Star Wars: Episode VI - Return of the Jedi (1983)"
## [21] "Terminator 2: Judgment Day (1991)"
## [22] "Usual Suspects, The (1995)"
##
## $`3`
## [1] "Apollo 13 (1995)"
## [2] "Batman (1989)"
## [3] "Dances with Wolves (1990)"
## [4] "Gladiator (2000)"
## [5] "Sixth Sense, The (1999)"
## [6] "Twelve Monkeys (a.k.a. 12 Monkeys) (1995)"
##
## $`4`
## [1] "Fugitive, The (1993)"
## [2] "Independence Day (a.k.a. ID4) (1996)"
```

```
## [3] "Jurassic Park (1993)"
## [4] "Men in Black (a.k.a. MIB) (1997)"
## [5] "Mission: Impossible (1996)"
## [6] "Speed (1994)"
## [7] "True Lies (1994)"
```

```
table(movie_groups)
```

```
## movie_groups
##  1  2  3  4
##  5 22  6  7
```

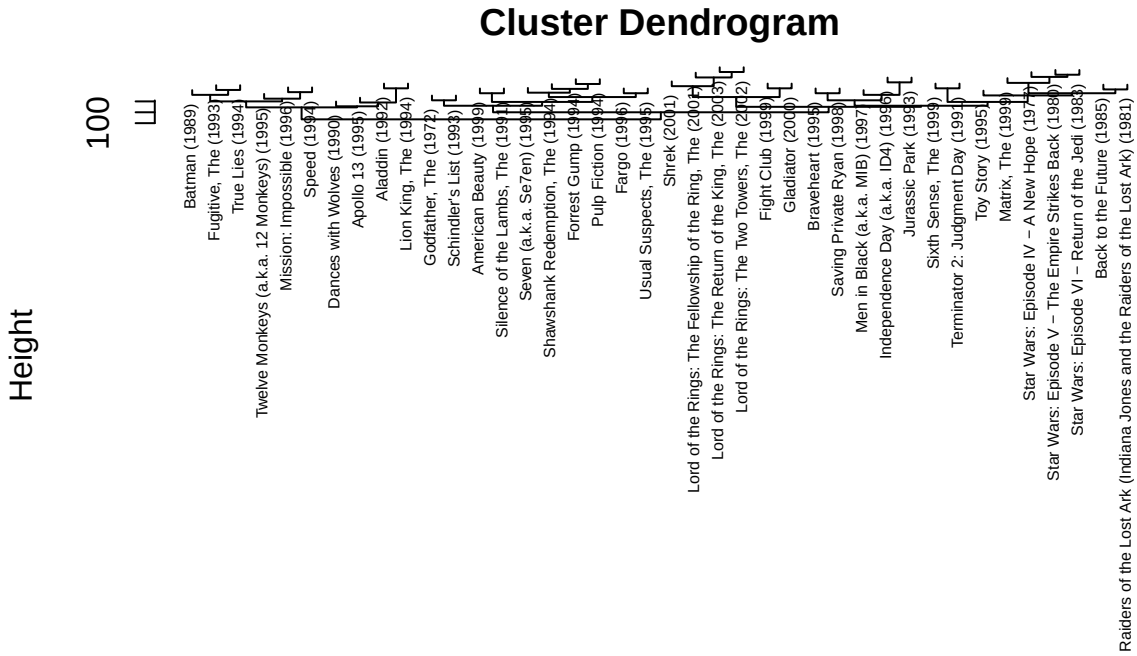
```
cutree(h, h = 16)
```

```
##                                                                 Aladdin (1992)
##                                                                              1
##                                                        American Beauty (1999)
##                                                                              2
##                                                             Apollo 13 (1995)
##                                                                              3
##                                                     Back to the Future (1985)
##                                                                              1
##                                                                  Batman (1989)
##                                                                              3
##                                                           Braveheart (1995)
##                                                                              2
##                                                     Dances with Wolves (1990)
##                                                                              3
##                                                                 Fargo (1996)
##                                                                              2
##                                                           Fight Club (1999)
##                                                                              2
##                                                         Forrest Gump (1994)
##                                                                              2
##                                                       Fugitive, The (1993)
##                                                                              4
##                                                          Gladiator (2000)
##                                                                              3
##                                                       Godfather, The (1972)
##                                                                              2
##                                          Independence Day (a.k.a. ID4) (1996)
##                                                                              4
##                                                        Jurassic Park (1993)
##                                                                              4
##                                                       Lion King, The (1994)
##                                                                              1
##                 Lord of the Rings: The Fellowship of the Ring, The (2001)
##                                                                              2
##                     Lord of the Rings: The Return of the King, The (2003)
##                                                                              2
##                              Lord of the Rings: The Two Towers, The (2002)
##                                                                              2
##                                                           Matrix, The (1999)
##                                                                              2
##                                               Men in Black (a.k.a. MIB) (1997)
```

```
##                                                                                 4
##                                                       Mission: Impossible (1996)
##                                                                                 4
##                                                              Pulp Fiction (1994)
##                                                                                 2
## Raiders of the Lost Ark (Indiana Jones and the Raiders of the Lost Ark) (1981)
##                                                                                 2
##                                                       Saving Private Ryan (1998)
##                                                                                 2
##                                                          Schindler's List (1993)
##                                                                                 2
##                                                      Seven (a.k.a. Se7en) (1995)
##                                                                                 2
##                                                  Shawshank Redemption, The (1994)
##                                                                                 2
##                                                                  Shrek (2001)
##                                                                                 1
##                                                  Silence of the Lambs, The (1991)
##                                                                                 2
##                                                          Sixth Sense, The (1999)
##                                                                                 3
##                                                                  Speed (1994)
##                                                                                 4
##                                       Star Wars: Episode IV - A New Hope (1977)
##                                                                                 2
##                             Star Wars: Episode V - The Empire Strikes Back (1980)
##                                                                                 2
##                                 Star Wars: Episode VI - Return of the Jedi (1983)
##                                                                                 2
##                                               Terminator 2: Judgment Day (1991)
##                                                                                 2
##                                                                 Toy Story (1995)
##                                                                                 1
##                                                                 True Lies (1994)
##                                                                                 4
##                                       Twelve Monkeys (a.k.a. 12 Monkeys) (1995)
##                                                                                 3
##                                                     Usual Suspects, The (1995)
##                                                                                 2
```

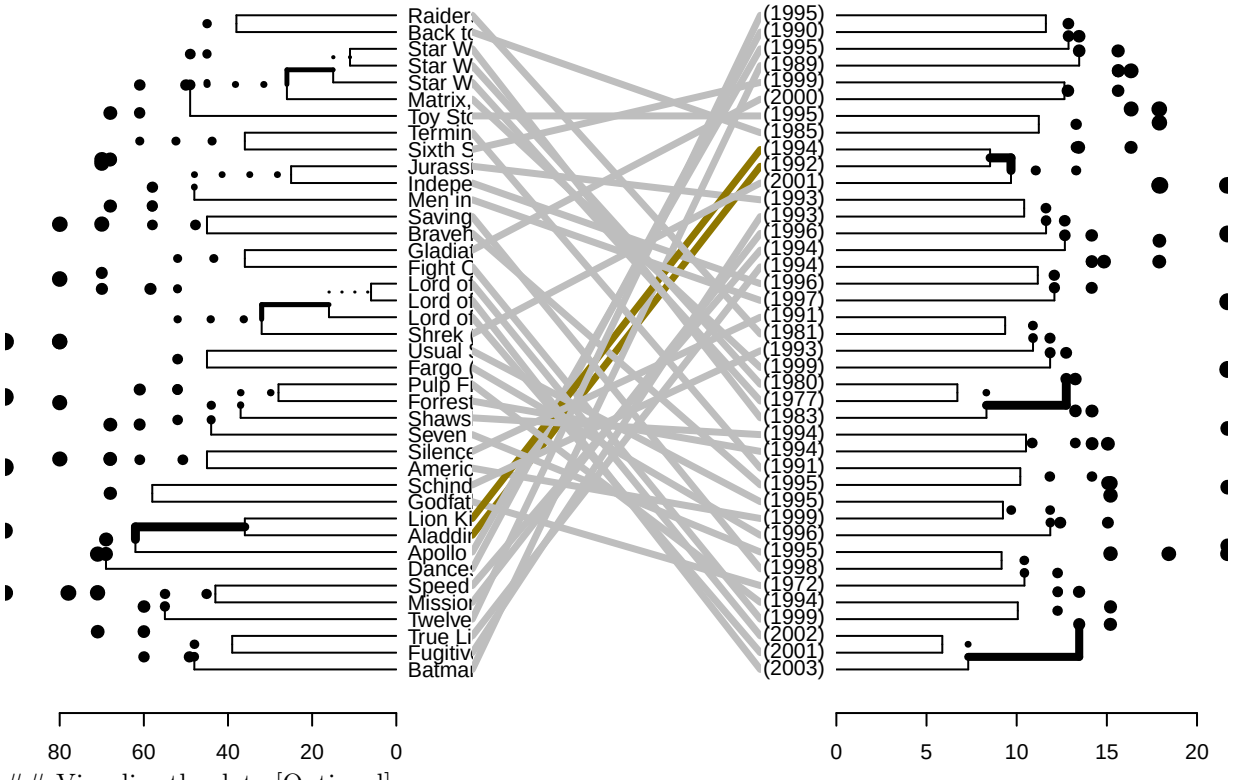
3. You may have noticed that not every movie has a rating by every user. This makes sense since no one could have possibly watched every movie. One question you may ask is whether the clustering result is based on the actual number in the rating (of 1 to 5 stars), or whether it's clustering for the existence of a rating. Make a new dataset by replacing all missing ratings (ie. the NAs) with 0, and all the ratings (regardless of value) with 1. And then repeat the hierarchical clustering, but this time use the **Manhattan distance**. Use `cutree` to find 4 clusters and compare to your result in the previous question.

```
movielens_mat <- as.matrix(movielens)
movielens_mat[is.na(movielens_mat)] <- 0
movielens_mat[movielens_mat > 0] <- 1

d_man <- dist(movielens_mat, method = "manhattan")
h_man <- hclust(d_man)
plot(h_man, cex = 0.5)
```



d_man
hclust (*, "complete")

```
movie_groups_man <- cutree(h_man, k = 4)
tanglegram(as.dendrogram(h_man), as.dendrogram(h))
```

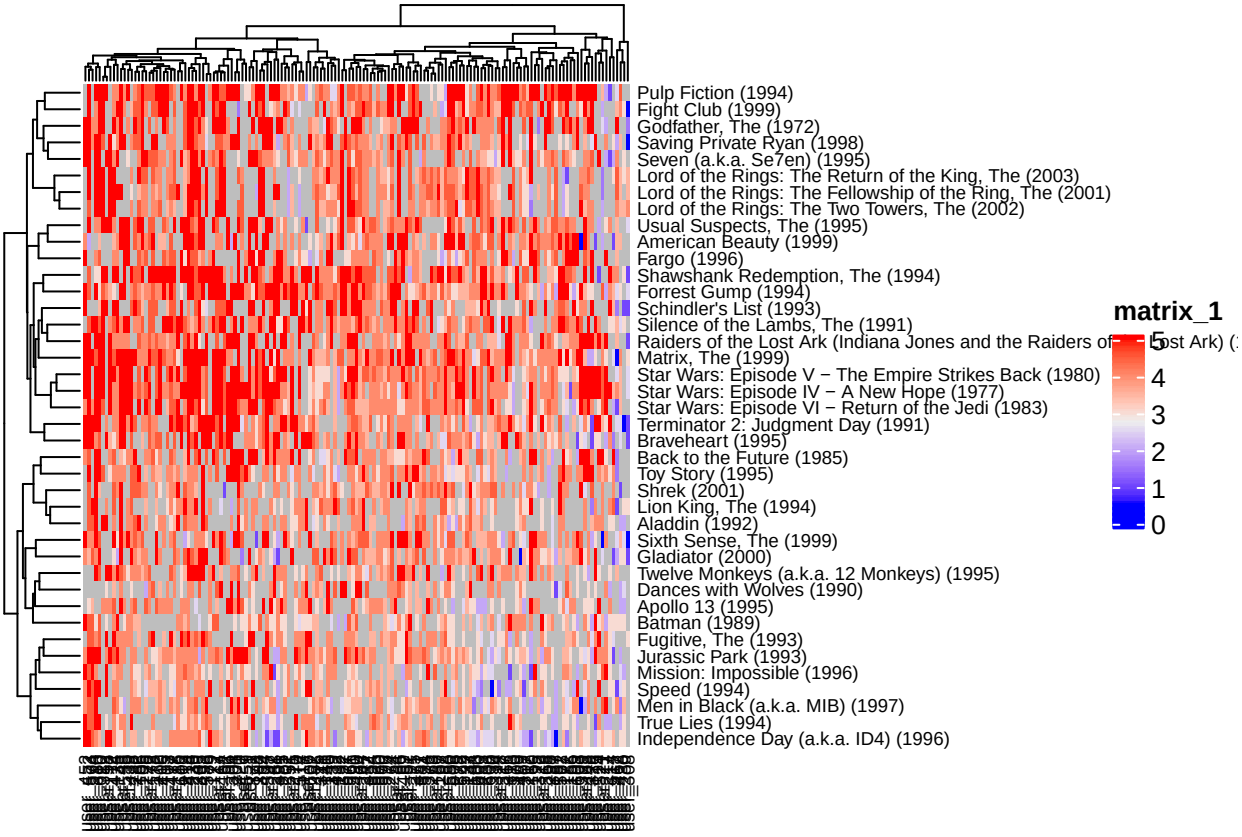


Visualize the data [Optional]

R also offers a number of packages that enable the user to visualize the data together with the clustering tree. We call these visualizations "heatmaps" of the data matrix. Download and install the package `ComplexHeatmap` using the code provided below and we will need to ensure the input is a matrix as expected by the function `Heatmap`. The arguments `row_names_gp` and `column_names_gp` enable us to reduce the font size.

```
# BiocManager::install("ComplexHeatmap")
# BiocManager::install("shape")
library(ComplexHeatmap)
movielens_matrix <- as.matrix(movielens)
```

```
movielens_matrix <- as.matrix(movielens)
library(ComplexHeatmap)
movielens_matrix <- as.matrix(movielens)
Heatmap(movielens_matrix,
        row_names_gp = gpar(fontsize = 7),
        column_names_gp = gpar(fontsize = 7))
```

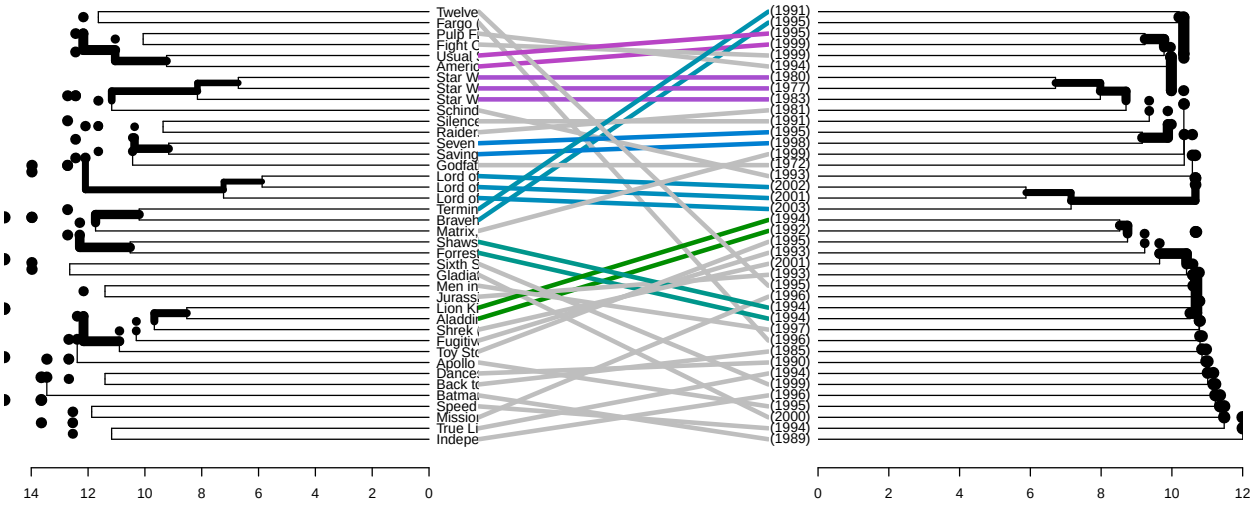


1.4 Comparing trees [Optional]

Suppose we like to compare the effect of two trees and visualize it. `R` has a package called `dendextend` that compare two dendrograms, it has the following key functions - `untangle()`: finds alignment, - `tanglegram()`: visualise the two dendrograms, - `entanglement()`: computes the quality of the alignment.

```
library(dendextend)

# Create two dendrograms
h_avg <- hclust(d, method = "average")
h_single <- hclust(d, method = "single")
```

```
dend1 <- as.dendrogram (h_avg)
dend2 <- as.dendrogram (h_single)

# Create a list to hold dendrograms
dend_list <- dendlist(dend1, dend2)

# Compare the two trees
tanglegram(dend_list)
```



1.5 k-means

1. Basic k-means usage

Next, let's explore the `kmeans` method. Go back to the original movies dataset with ratings between 1 to 5 and missing values, let's now make a new dataset replacing all the NAs with 0 but keep the ratings. We are doing this because the `kmeans` function cannot handle missing values. In a later module, we will look at how to handle missing values. Use `kmeans` to cluster the movies into four clusters. How many movies are in each cluster?

```
movielens_mat <- as.matrix(movielens)
movielens_mat[is.na(movielens_mat)] <- 0
kmeans_res <- kmeans(movielens_mat, centers = 4)
table(kmeans_res$cluster)
```

```
##
##  1  2  3  4
##  9 10 15  6
```

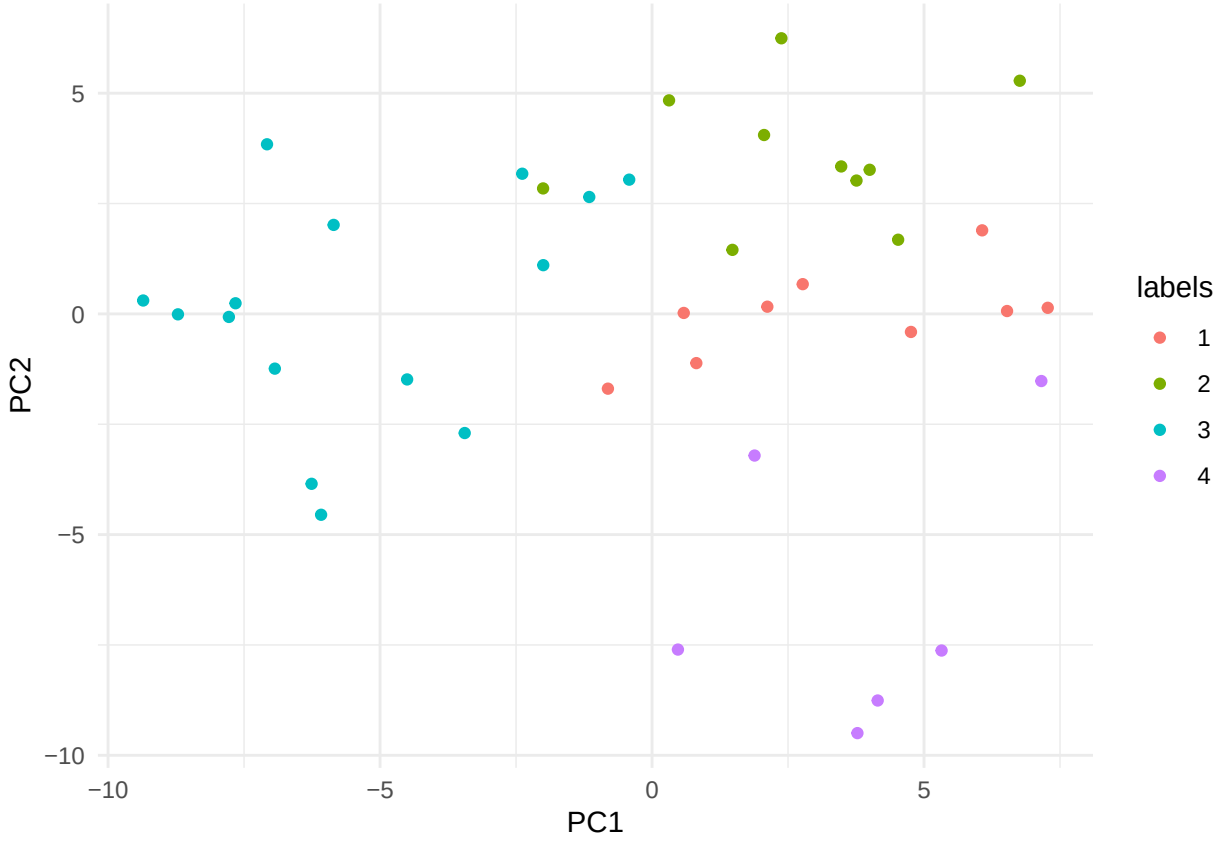
2. To visualize results from the `kmeans` clustering, use a dimension reduction technique such as PCA.

```
movie_pc = prcomp(movielens_mat, scale = TRUE)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

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

```
movie.df = data.frame(PC1 = movie_pc$x[,1], PC2 = movie_pc$x[,2],
                      labels = factor(kmeans_res$cluster))
ggplot(movie.df, aes(PC1, PC2, colour = labels)) + geom_point() + theme_minimal()
```

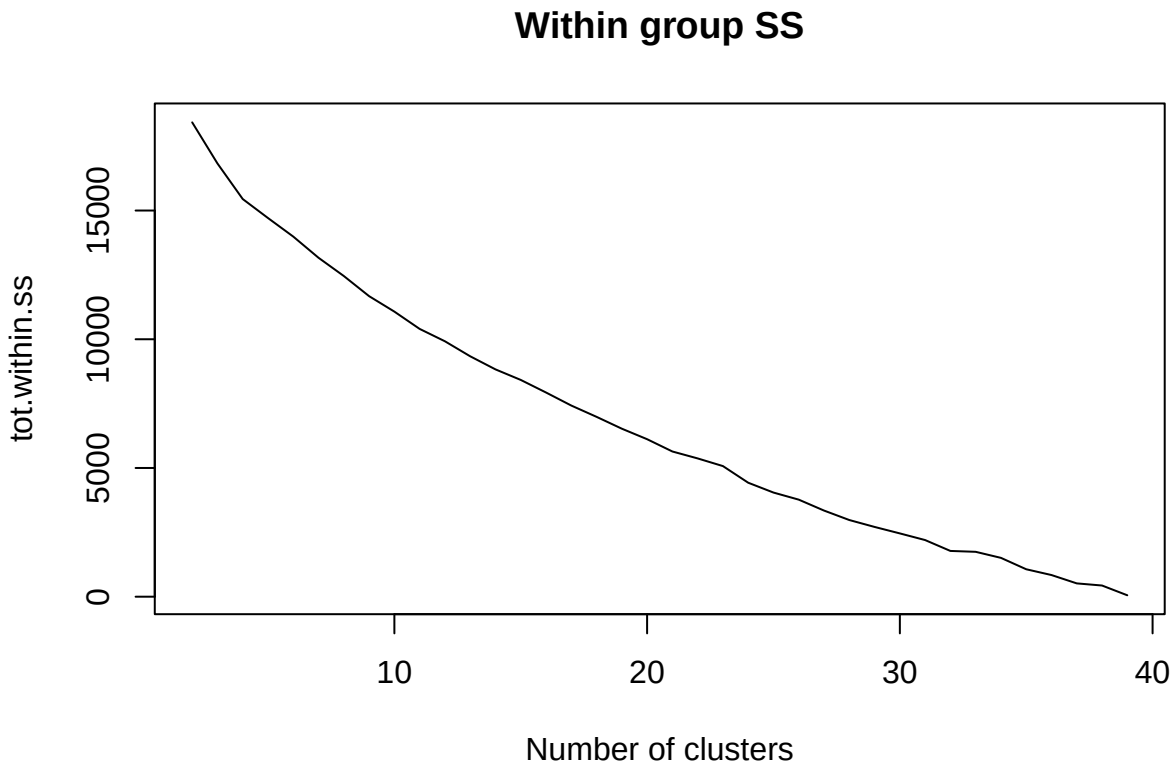


1.6 Cluster statistics

Let's now look at the cluster statistics. Can you plot the total within group sum of squares for k = 2, 3, 4, 5, 6 from `kmeans()`. The `tot.withinss` is part of the output value of `kmeans`. Repeat for between group sum of squares (`betweenss`). Do the plots hint at what is the best `k`?

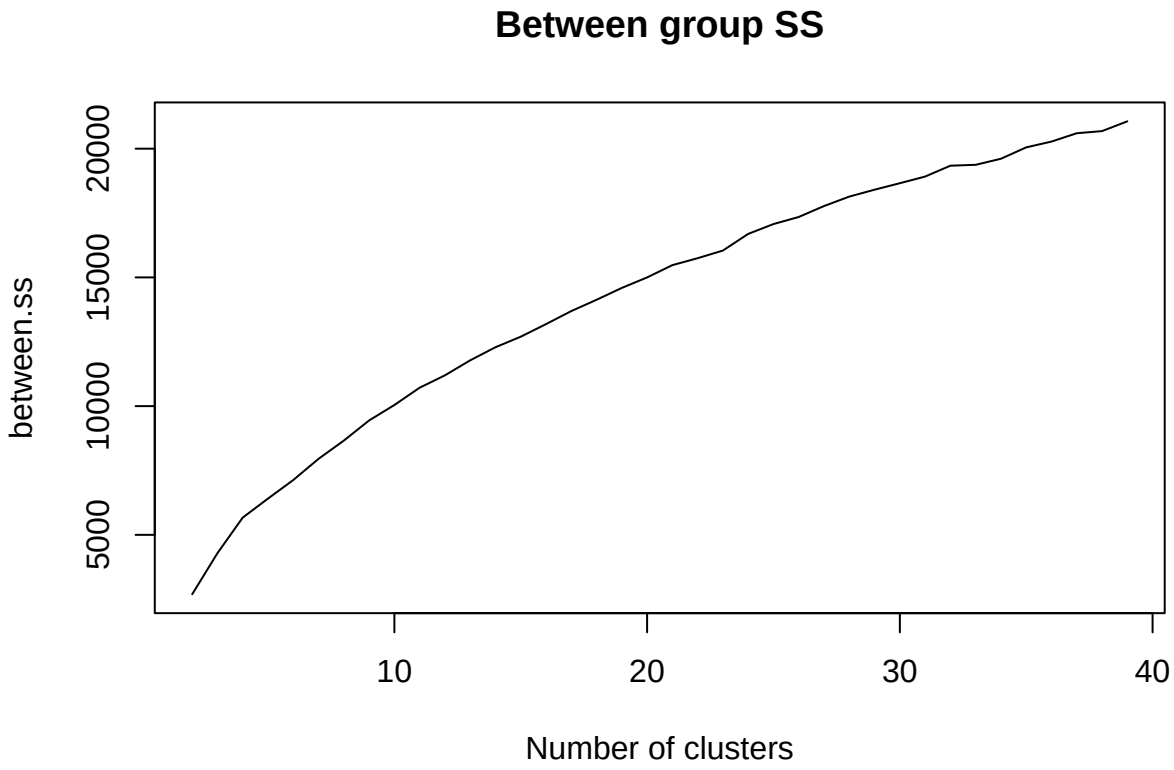
```
set.seed(5003)
kmeans_2 <- kmeans(movielens_mat, centers = 2)
kmeans_3 <- kmeans(movielens_mat, centers = 3)
kmeans_4 <- kmeans(movielens_mat, centers = 4)
kmeans_5 <- kmeans(movielens_mat, centers = 5)
kmeans_6 <- kmeans(movielens_mat, centers = 6)

tot.withinss <- c(kmeans_2$tot.withinss, kmeans_3$tot.withinss,
                  kmeans_4$tot.withinss, kmeans_5$tot.withinss,
                  kmeans_6$tot.withinss)

betweenss <- c(kmeans_2$betweenss, kmeans_3$betweenss,
               kmeans_4$betweenss, kmeans_5$betweenss,
               kmeans_6$betweenss)

# or more directly using the apply suite over a larger range
set.seed(5003)
```

```
center.seq <- 2:39
kmeans <- lapply(center.seq, function(x) kmeans(movielens_mat, centers = x))
tot.within.ss <- sapply(kmeans, "[[", "tot.withinss")
between.ss <- sapply(kmeans, "[[", "betweenss")


plot(center.seq, tot.within.ss,
     xlab = "Number of clusters", main = "Within group SS",
     type = 'l')
```



Notice the total within sum of squares decreasing monotonically as the number of clusters increases

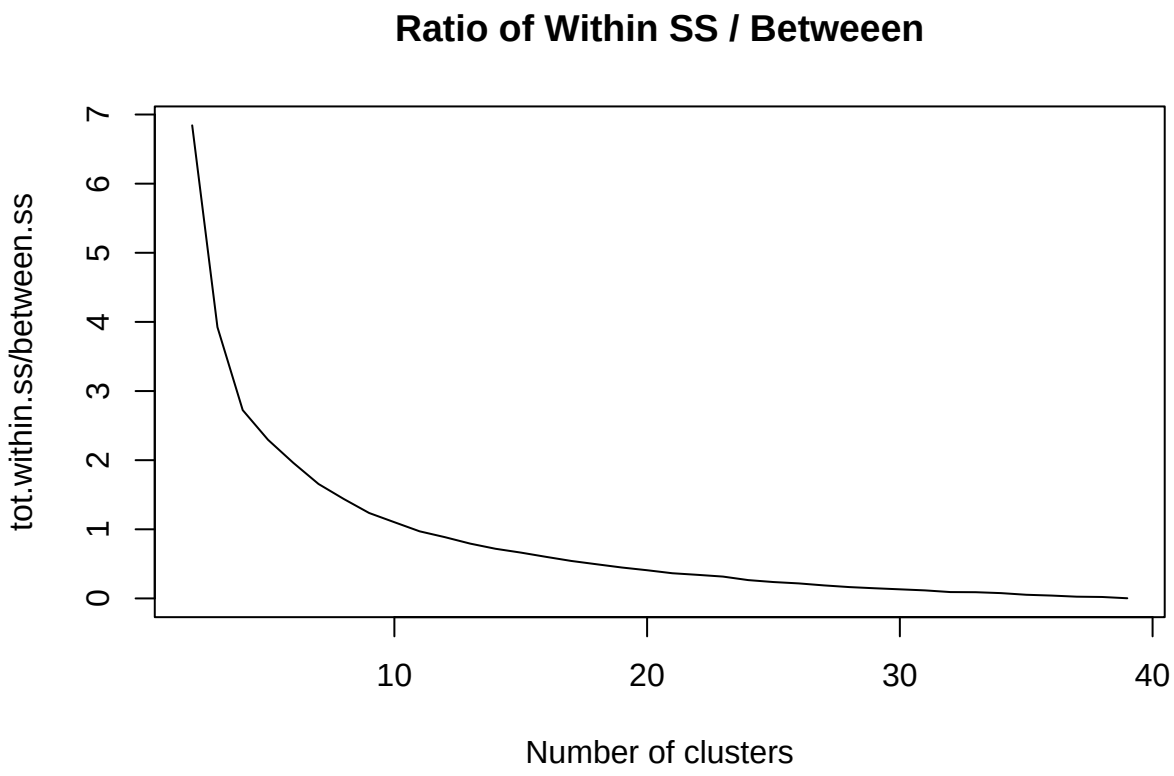
```
plot(center.seq, between.ss,
     xlab = "Number of clusters", main = "Between group SS",
     type = 'l')
```



Also the between sum of squares increases as the number of clusters increases

The appropriate metric to consider for assessing the appropriateness of the number of clusters is the ratio of within to between SS

```
plot(center.seq, tot.within.ss/between.ss, xlab = "Number of clusters",
     main = "Ratio of Within SS / Betweeen", type = 'l')
```



2 Author by word count

The next dataset `author_count.csv` shows the counts of common words appearing in documents by four authors, Jane Austen, Jack London, William Shakespeare and John Milton. We like to investigate whether clustering based word characterstics is able to split the four authors apart. Here the first column shows the author, the remaining columns show the counts of each word.

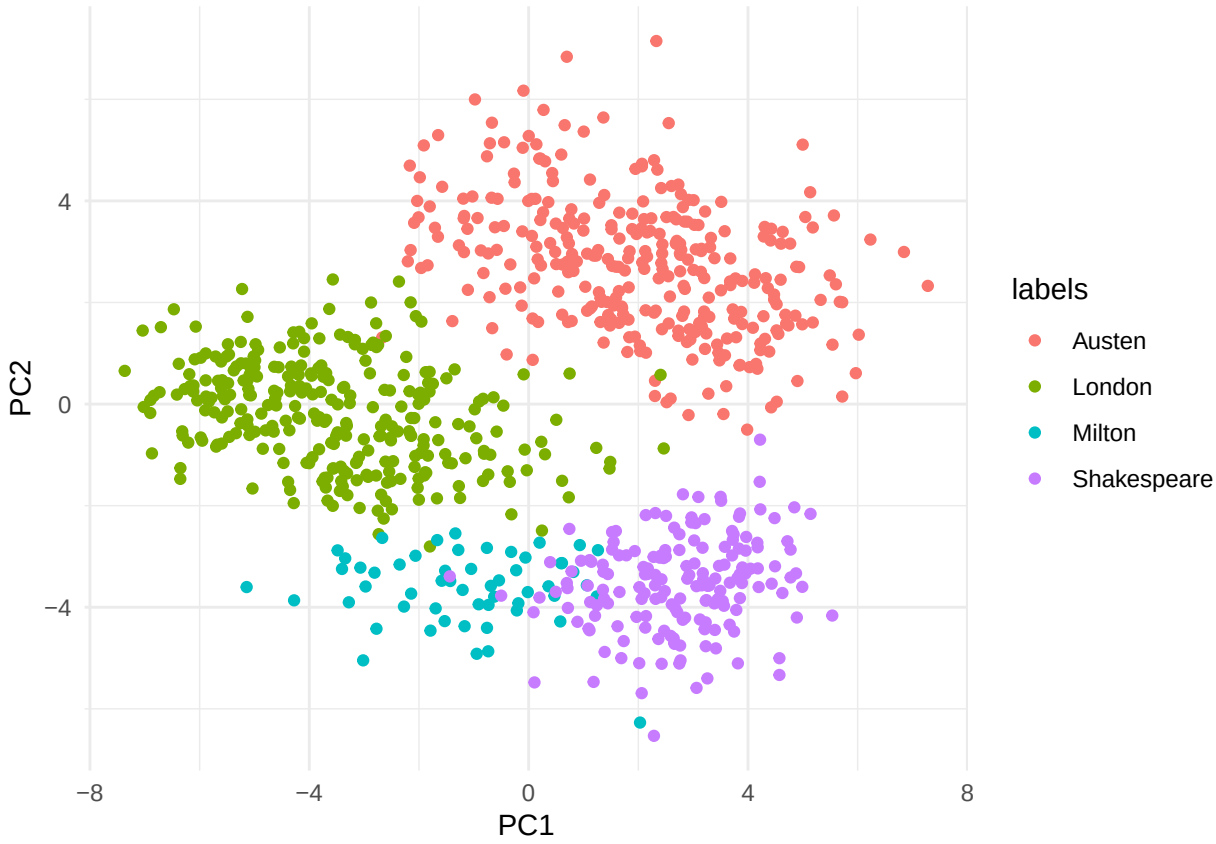
2.1 Data input

```
author.dat <- read.csv("author_count.csv", header = TRUE)
numeric.dat <- author.dat[-1]
authors <- factor(author.dat[[1]])
```

2.2 PCA

Compute the PCA and visualize the output.

```
pca.scaled <- prcomp(numeric.dat, scale = TRUE)
library(gridExtra)
author.df <- data.frame(PC1 = pca.scaled$x[,1], PC2 = pca.scaled$x[,2],
                        PC3 = pca.scaled$x[,3], labels = authors)
pca.plot <- ggplot(author.df, aes(PC1, PC2, col = labels)) + geom_point() + theme_minimal()
pca.plot
```

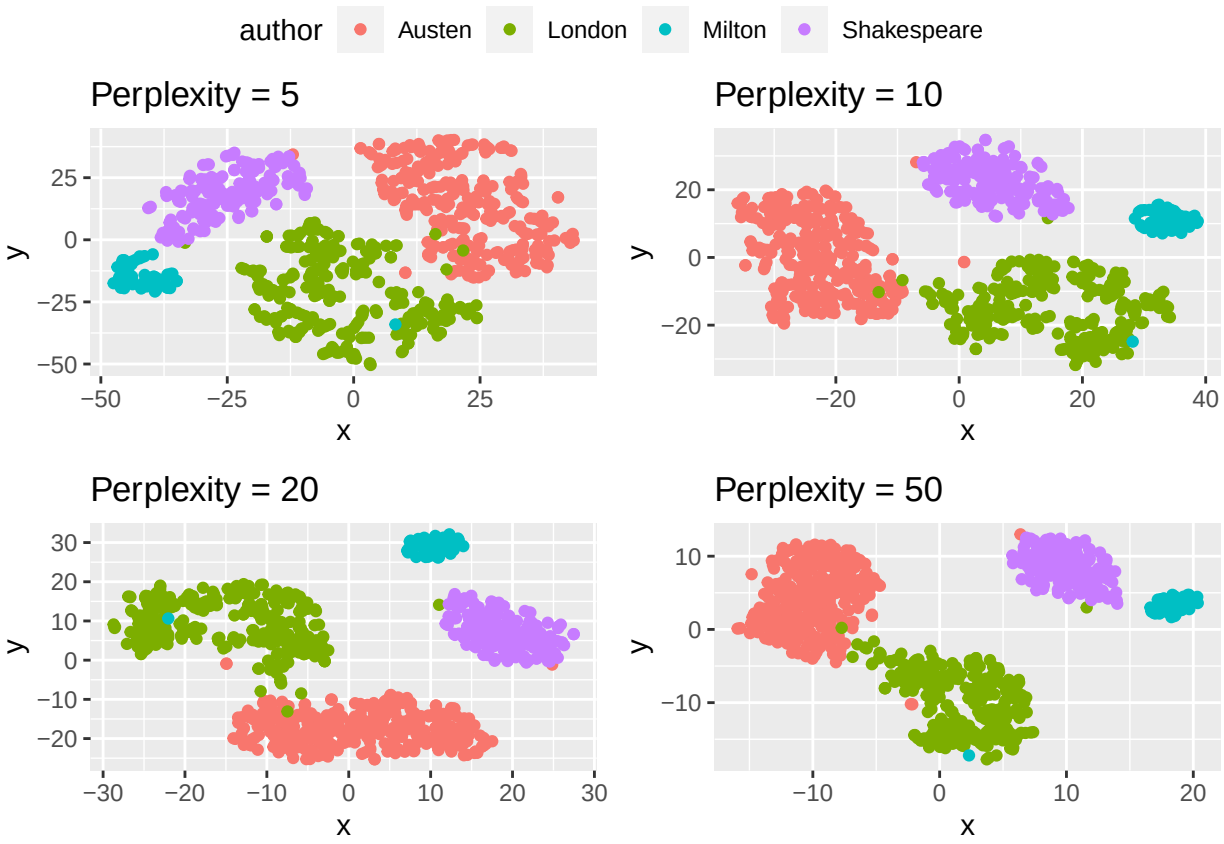


2.3 t-SNE

Compute and view the t-SNE plots for various perplexity levels for this dataset. Here you will need to consider adjusting the perplexity values.

Solution

```
library(Rtsne)
library(ggpubr)
set.seed(5003)
perplexity <- c(5, 10, 20, 50)
rtsne <- lapply(perplexity, function(x) {
  y <- Rtsne(numeric.dat, dims = 2, perplexity = x)$Y
  attr(y, "perplexity") <- x
  y
  })

tsne.plots <- lapply(rtsne, function(dat) {
  perplexity <- attr(dat, "perplexity")
  dat <- as.data.frame(dat)
  names(dat) <- c("x", "y")
  dat[["author"]] <- authors
  ggplot(dat) + geom_point(aes(x = x, y = y, colour = author)) +
    ggtitle(paste0("Perplexity = ", perplexity))
})

ggarrange(plotlist = tsne.plots, common.legend = TRUE)
```



2.4 MDS

1. Consider the MultiDimensionalScaling (MDS) technique to visualize the data. Compute different distance matrices using the `dist` function for the `author_count` dataset.

Solution

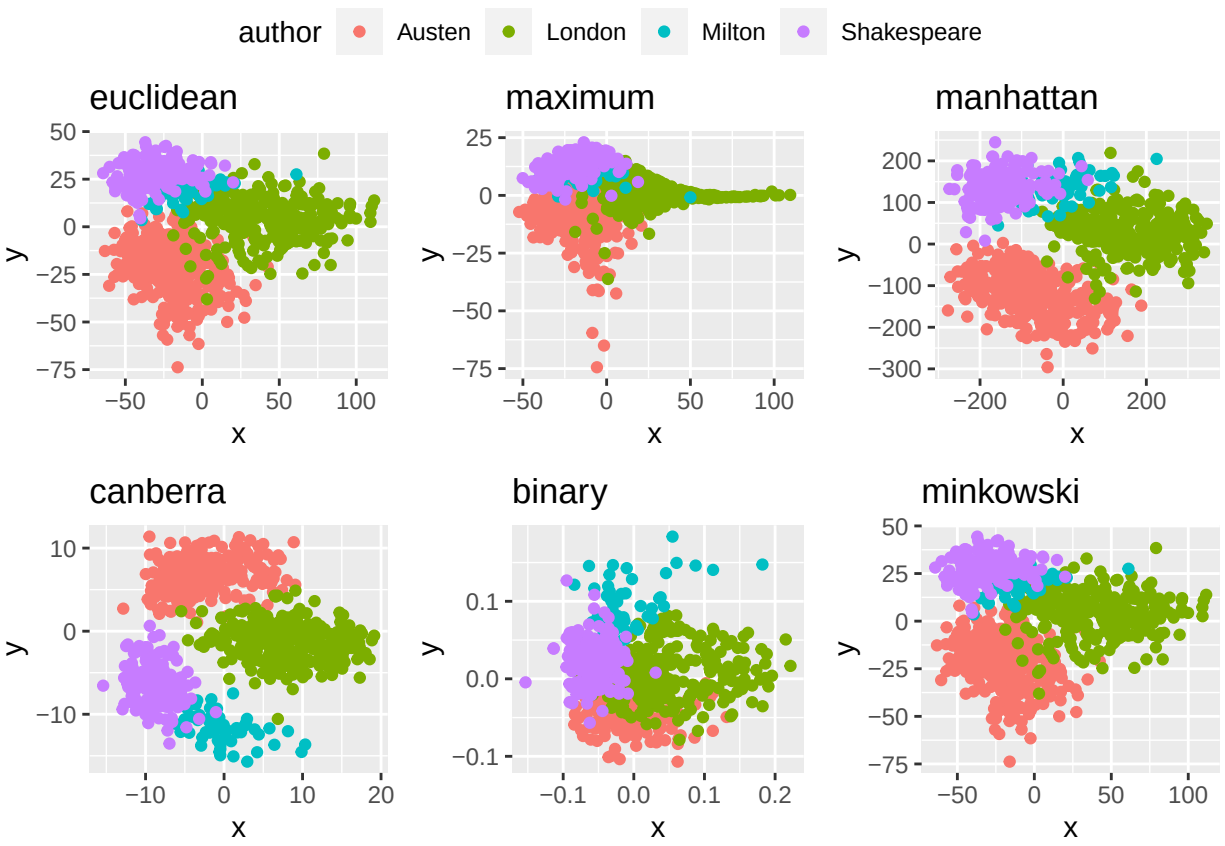
```
dist.types <- c("euclidean", "maximum", "manhattan", "canberra", "binary",  "minkowski")
dist.matrices <- lapply(dist.types, function(x) {
  y <- dist(numeric.dat, method = x)
  y
})
```

2. Create the MDS plot in 2 dimensions and colour the plot by the true author.

Solution

```
mds.out <- lapply(dist.matrices, function(x) {
  y <- cmdscale(x)
  attr(y, "method") <- attr(x, "method")
  y
})

mds.plots <- lapply(mds.out, function(dat) {
  method <- attr(dat, "method")
  dat <- as.data.frame(dat)
  names(dat) <- c("x", "y")
  dat[["author"]] <- authors
  ggplot(dat) + geom_point(aes(x = x, y = y, colour = author)) + ggtitle(method)
```

```
})

ggarrange(plotlist = mds.plots, common.legend = TRUE)
```

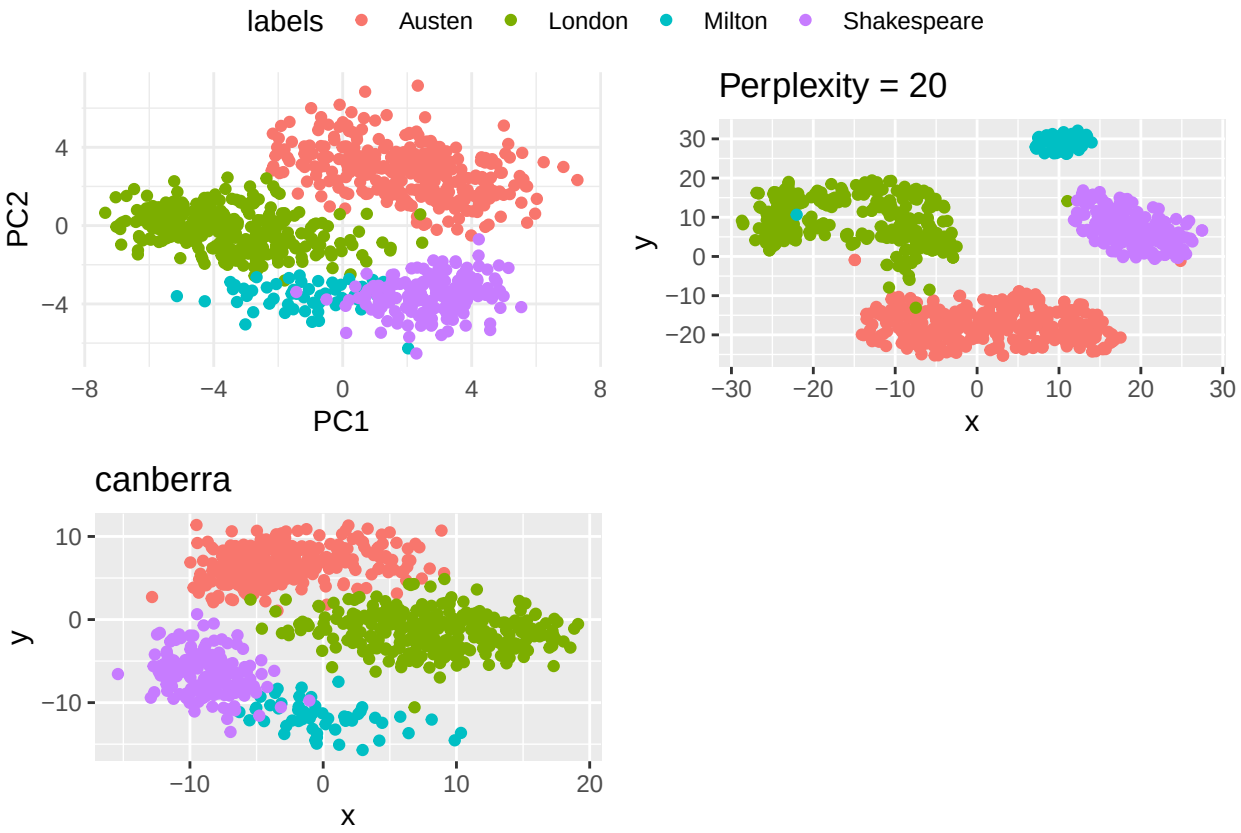


2.5 Compare and contrast

Select the best result in each case for PCA, t-SNE and MDS and compare.

Solution

```
mds.plot <- mds.plots[[which(dist.types == "canberra")]]
tsne.plot <- tsne.plots[[which(perplexity == 20)]]
ggarrange(plotlist = list(pca.plot, tsne.plot, mds.plot), common.legend = TRUE)
```



The Canberra metric appears to be the best for MDS. The t-SNE plot seems to do the best to discriminate between the authors into well separated clusters with tight grouping around their centers. However, there seems to be some rare points where t-SNE has placed the point in the wrong cluster. The MDS plot seems to not suffer from this but the clusters are not as well separated. The PCA while it does an admirable job, it the least favourable to explain the data visually.

3 Shiny app to allow the user to explore and decide

Create a shiny app for the `author_count` data which gives the user options to decide which visualization technique to use and calibrate it with any necessary parameters .