Progressive Education Society's

Modern College of Engineering, Pune

**MCA Department A.Y.2024-25**

# (410907) Big Data Analytics Laboratory

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Class: SY-MCA Shift / Div : S3 / A Roll Number : 52062

Name: Laxman Shinde Assignment No: 2 Date of Implementation: 06 / 09 / 2024

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Q.2. i.Implement a content-based recommendation system for a movie dataset.**

**Use movie genres and descriptions to recommend similar movies.**

**Code :**

if (!require("tm")) install.packages("tm", dependencies=TRUE)

if (!require("textstem")) install.packages("textstem", dependencies=TRUE)

if (!require("proxy")) install.packages("proxy", dependencies=TRUE)

library(tm)

library(textstem)

library(proxy)

movies <- data.frame(

title = c("Inception", "Titanic", "The Dark Knight", "La La Land", "The Conjuring",

"Interstellar", "The Social Network", "Mad Max: Fury Road", "The Godfather", "The Lord of the Rings"),

genre = c("Action Sci-Fi", "Romance Drama", "Action Crime Thriller",

"Romance Musical", "Horror", "Science Fiction",

"Drama Biography", "Action Adventure", "Crime Drama", "Fantasy Adventure"),

description = c(

"A mind-bending thriller where a skilled thief enters dreams to steal secrets.",

"A romantic tale of love and loss aboard the ill-fated RMS Titanic.",

"Batman faces off against the Joker in a dark and thrilling crime saga.",

"A jazz musician and an aspiring actress fall in love while pursuing their dreams.",

"A family is haunted by malevolent spirits in their farmhouse.",

"Explorers travel through a wormhole in space to find a new habitable planet.",

"The story of Mark Zuckerberg and the creation of Facebook.",

"In a post-apocalyptic world, Max teams up with Furiosa to escape a ruthless warlord.",

"The aging patriarch of a crime family transfers control of his empire to his son.",

"A hobbit and his companions embark on a journey to destroy a powerful ring."

)

)

movies$content <- paste(movies$genre, movies$description)

preprocess\_text <- function(text) {

# Convert to lower case

text <- tolower(text)

# Remove punctuation

text <- removePunctuation(text)

text <- removeWords(text, stopwords("en"))

text <- lemmatize\_strings(text)

text <- stripWhitespace(text)

return(text)

}

movies$content <- sapply(movies$content, preprocess\_text)

corpus <- Corpus(VectorSource(movies$content))

dtm <- DocumentTermMatrix(corpus, control = list(weighting = weightTfIdf))

dtm\_matrix <- as.matrix(dtm)

similarity\_matrix <- as.matrix(simil(dtm\_matrix, method = "cosine"))

recommend\_movies <- function(movie\_index, top\_n = 3) {

similarity\_scores <- similarity\_matrix[movie\_index, ]

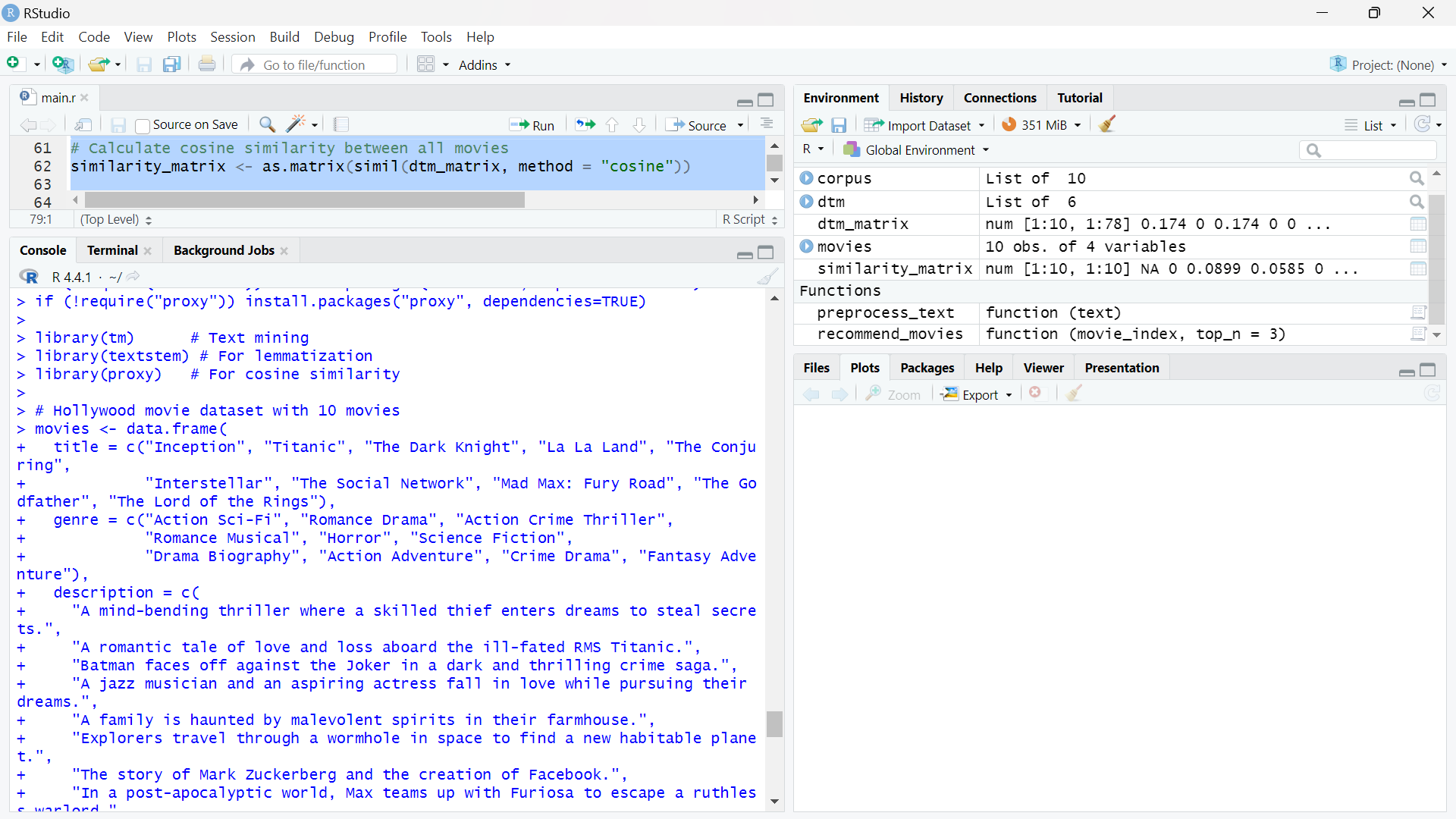
similar\_movies <- order(similarity\_scores, decreasing = TRUE)[2:(top\_n+1)]

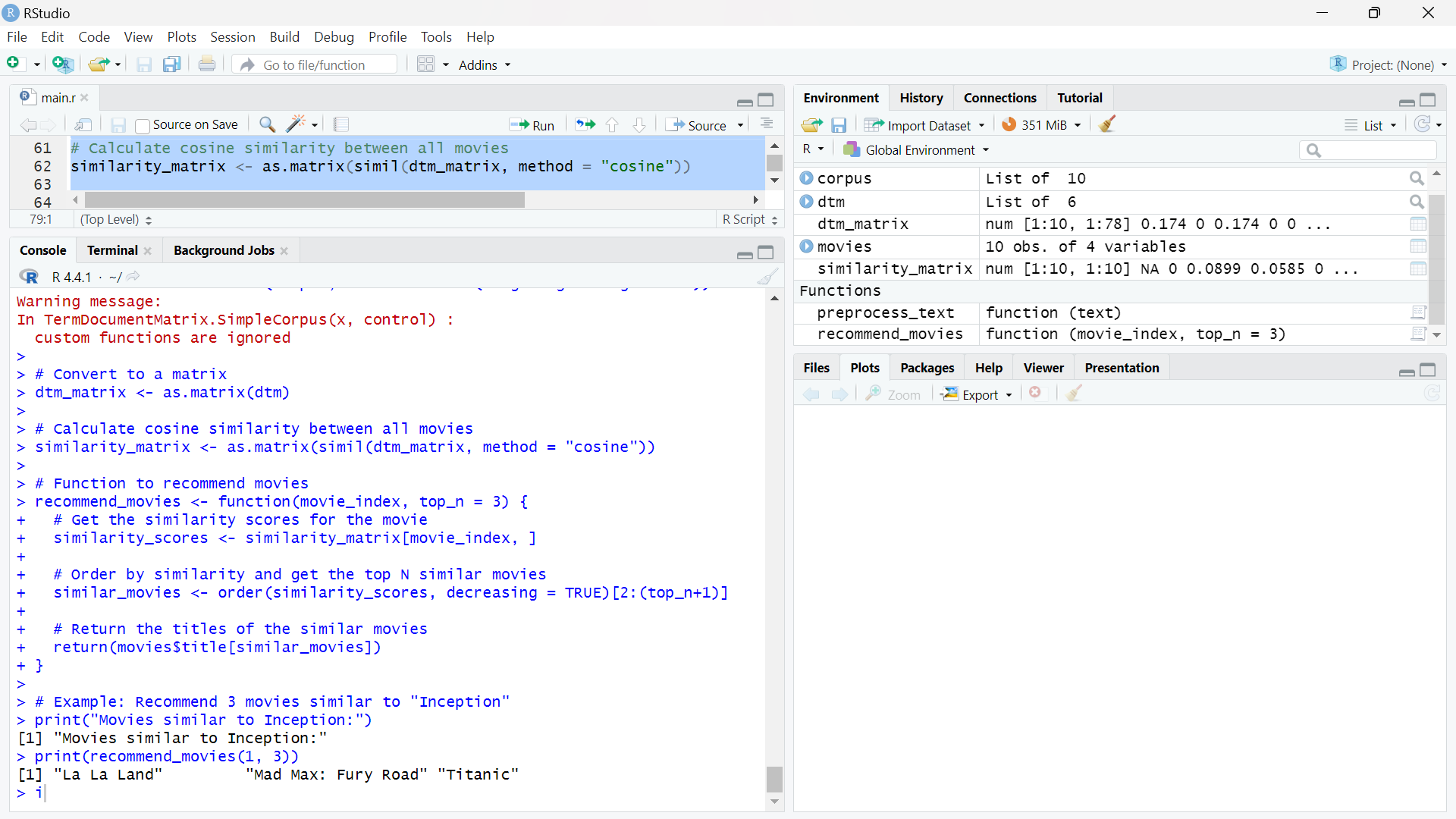
return(movies$title[similar\_movies])

}

print("Movies similar to Inception:")

print(recommend\_movies(1, 3))





**Q.2. ii. Create a graph representing a social network. Visualize the graph.**

Code :

if (!require("igraph")) install.packages("igraph", dependencies=TRUE)

library(igraph)

edges <- c("Alice", "Bob",

"Alice", "Charlie",

"Alice", "David",

"Bob", "Eve",

"Charlie", "David",

"David", "Eve",

"Eve", "Frank",

"Frank", "George",

"George", "Alice")

g <- graph(edges, directed = FALSE)

plot(g,

vertex.color = "lightblue", # Color of the vertices (people)

vertex.size = 30, # Size of the vertices

vertex.label.color = "black", # Label color

vertex.label.cex = 1.2, # Label size

edge.color = "gray", # Color of the edges (relationships)

main = "Social Network Graph"

)

**Output :**

