**PROBABILITY AND STATISTICS SEMESTER PROJECT SPRING 2023**

**PROJECT DETAILS**

1. Choose data of your interest.
2. Use Statistical Methods to analyze that data using R language.
3. Statistical analysis should consist of the appropriate:

* Graphical and tabular data representation
* Descriptive Statistical Measure
* Probability methods/Distribution
* Regression Modeling and Predictions
* Confidence interval of Descriptive measures and Regression estimates.

1. Integrate your results on a web page and will be in well-presentable form.
2. Write a report on the whole task using the default pattern available at the end of this document.
3. Submission Date of the Project will be 15 May 2023.

**GROUPS DETAILS**

1. A team will be consisting of at most 4 members
2. Choose a leader and a name/logo for the team.
3. Select a large real-world data set for your work.

**EVALUATION SCHEME**

* Evaluation criteria :

*How carefully was the above-mentioned instruction followed?*

*How many appropriate statistical tools were used?*

*How attractive and intelligent your interface is?*

*How is the Project report made?*

* *Failing to meet the required criteria will result in mark deduction.*

**General Instructions**

1. Viva of your submitted work can be conducted.
2. If your data, code, or application is similar to any other group it causes zero marks.
3. Late submissions will not be entertained.
4. If any of your team members do not play a significant role in teamwork, the leader will intimate before submission.
5. If the team leader will not coordinate properly, the majority may change it and intimate it before submission.
6. Paste all codes after giving them proper titles on a Word file using font size 9 and line space 1, so a minimum space can be managed.
7. Paste screenshots of your interface with different features/results on a Word file using the default format.
8. You will submit both hard and soft files of your Project till the submission date.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*DRAFT OF SUBMISSION\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

*First Page:*

**Netflix Content Analysis: Trends & Preferences**



*Next Pages:*

**1. Problem Statement**

The goal is to study the Netflix dataset to understand what users like and how they behave. Then, use this knowledge to suggest movies or TV shows that users are more likely to enjoy, making them happier with the service.

**2. Objective**

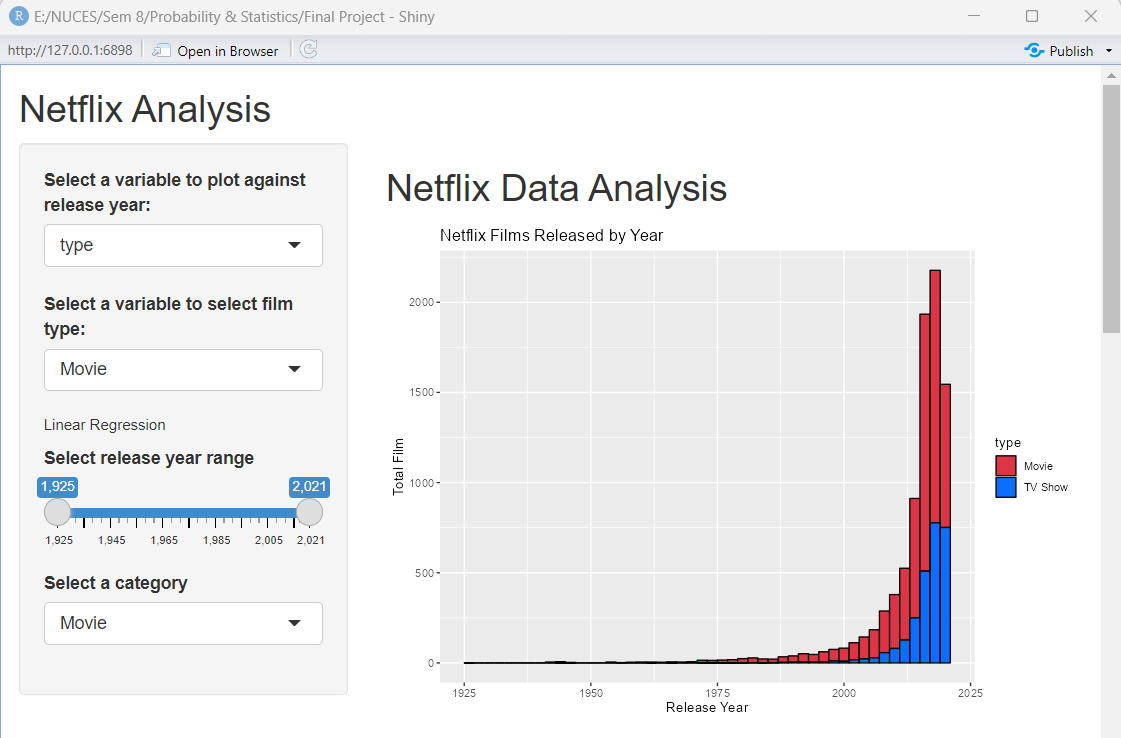
To gain insights into the content and user preferences of Netflix to improve user engagement and satisfaction and to identify trends in user preferences and behavior, specifically related to movies and TV shows.

**3. Data Description**

[Netflix Dataset Link](https://www.kaggle.com/datasets/shivamb/netflix-shows?resource=download)

Netflix is one of the most popular media and video streaming platforms. This tabular dataset consists of listings of all the movies and tv shows available on Netflix, along with details such as - cast, directors, ratings, release year, duration, etc.

**4. Results**



A screen shot of a diagram

Description automatically generated with low confidence

A screenshot of a computer

Description automatically generated with medium confidence

A screen shot of a graph

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

**5. R codes**

# Load the necessary libraries

library(shiny)

library(tidyverse)

library(readr)

library(ggplot2)

library(ggpubr)

# Define UI

ui <- fluidPage(

# Application title

titlePanel("Netflix Analysis"),

# Sidebar with a select input for the variable to plot against release year

sidebarLayout(

sidebarPanel(

selectInput("var",

"Select a variable to plot against release year:",

choices = c("type", "rating", "listed\_in", "country")

),

selectInput("film\_type",

"Select a variable to select film type:",

choices = c("Movie", "TV Show")

),

h6("Linear Regression"),

sliderInput(inputId = "year\_range", label = "Select release year range",

min = min(netflix\_titles$release\_year),

max = max(netflix\_titles$release\_year),

value = c(min(netflix\_titles$release\_year),

max(netflix\_titles$release\_year))),

selectInput(inputId = "category", label = "Select a category",

choices = c("Movie", "TV Show"), selected = "Movie")

),

# Show a histogram and a pie chart of film types

mainPanel(

h2("Netflix Data Analysis"),

plotOutput("histogram"),

titlePanel("Type of Netflix Films Comparison"),

plotOutput("piechart"),

titlePanel("Top 10 Countries with the Most Movie Productions on Netflix"),

DT::dataTableOutput("table"),

plotOutput('film\_comparision'),

#Linear Regression

titlePanel("Linear Regression"),

plotOutput(outputId = "scatterplot"),

titlePanel("Linear regression model Results"),

verbatimTextOutput(outputId = "model\_summary")

)

)

)

# Define server logic

server <- function(input, output) {

# Load the Netflix dataset

netflix\_titles <- read\_csv("netflix\_data.csv", show\_col\_types = 'FALSE')

#Cleaning the loaded data Function

clean\_netflix\_data<- function(netflix\_data) {

# Drop any rows with missing values

netflix\_data <- netflix\_data[complete.cases(netflix\_data),]

# Convert release\_year to numeric

netflix\_data$release\_year <- as.numeric(netflix\_data$release\_year)

# Convert duration to numeric (in minutes)

netflix\_data$duration <- as.numeric(gsub(" min", "", netflix\_data$duration))

# Remove leading/trailing whitespace from strings

netflix\_data$title <- trimws(netflix\_data$title)

netflix\_data$director <- trimws(netflix\_data$director)

netflix\_data$cast <- trimws(netflix\_data$cast)

netflix\_data$country <- trimws(netflix\_data$country)

netflix\_data$listed\_in <- trimws(netflix\_data$listed\_in)

netflix\_data$description <- trimws(netflix\_data$description)

# Return cleaned dataset

return(netflix\_data)

}

#Data Cleaned

netflix\_titles <- clean\_netflix\_data(netflix\_data)

# Separate "listed\_in" column into three "category" column, and take only the first.

netflix\_titles <- netflix\_titles %>% separate(listed\_in, c("Category1", "Category2", "Category3"), sep = ",")

netflix\_titles <- netflix\_titles %>% select(c(-"Category2", -"Category3"))

# Separate "date\_added" column into "date\_added" and "year\_added" column.

netflix\_titles <- netflix\_titles %>% separate(date\_added, c("date\_added", "year\_added"), sep = ",")

# Create a new data frame of film types

Film\_Types <- netflix\_titles %>%

group\_by(type) %>%

count() %>%

ungroup() %>%

mutate(perc = `n` / sum(`n`)) %>%

arrange(perc) %>%

mutate(labels = scales::percent(perc))

# Define a histogram of film release years

output$histogram <- renderPlot({

ggplot(netflix\_titles, mapping = aes(x=release\_year, fill = !!sym(input$var))) +

geom\_histogram(color = "black", binwidth=2)+

labs(title="Netflix Films Released by Year", x="Release Year", y="Total Film")+

scale\_fill\_manual(values = c("Movie" = "#dc3545","TV Show" = "#0d6efd"))

})

# Define a pie chart of film types

output$piechart <- renderPlot({

ggplot(Film\_Types, aes(x = "", y = perc, fill = type)) +

geom\_col() +

geom\_text(aes(label = labels), position = position\_stack(vjust = 0.5)) +

coord\_polar(theta = "y") +

labs(title="") +

theme\_void() +

scale\_fill\_manual(values = c("Movie" = "#dc3545","TV Show" = "#0d6efd"))

})

#Filter Table Data

# Count the number of productions for each country

country\_counts <- netflix\_titles %>%

separate\_rows(country, sep = ",") %>%

group\_by(country) %>%

summarize(count = n()) %>%

arrange(desc(count)) %>%

top\_n(10, count)

output$table <- renderDataTable({

country\_counts

})

# Film Trends plot

output$film\_trend\_plot <- renderPlot({

df <- netflix\_titles

if (input$film\_type != "All") {

df <- filter(df, type == input$film\_type)

}

ggplot(df, mapping = aes(x=release\_year, fill = type)) +

geom\_histogram(color = "black", binwidth=2)+

labs(title="Netflix Films Released by Year", x="Release Year", y="Total Film")+

scale\_fill\_manual(values = c("Movie" = "#F6AE2D","TV Show" = "#00b22d"))

})

# Run a linear regression on the filtered data

# Filter the data based on the user's input

observe({

# Get the year range from the input

year\_range <- input$year\_range

data\_subset <- netflix\_titles %>%

filter(release\_year >= input$year\_range[1] &

release\_year <= input$year\_range[2] &

type == input$category)

# Create a scatterplot of release year vs. duration

output$scatterplot <- renderPlot({

ggplot(data\_subset, mapping = aes(x = release\_year, y = duration)) +

geom\_point() +

labs(title = "Netflix Film Duration by Release Year", x = "Release Year", y = "Duration (min)") +

theme\_bw()

})

# Run a linear regression model on the data and display the results

output$model\_summary <- renderPrint({

lm\_model <- lm(duration ~ release\_year, data = data\_subset)

summary(lm\_model)

})

})

}

# Run the application

shinyApp(ui = ui, server = server)

**6. Conclusion**

After analyzing the Netflix dataset, we found that most of the content on Netflix consists of movies, followed by TV shows. We also saw that the number of movie releases is higher than TV shows every year. The United States is the largest producer of movies on Netflix, followed by India and the United Kingdom. The linear regression shows that as the years have progressed, movies have tended to become longer in duration. This trend can be seen in the scatterplot generated by the model, which shows a general upward trend in duration as the years increase. These insights can help us understand user preferences on Netflix and improve user satisfaction by developing a better recommendation system.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*END\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*