DATA ANALYTICS MINI PROJECT

Title : Mall Customer Segmentation & Analytics

Name : SUSHANT GOVIND SHETTI

Roll No.: 22B-CO-062

1. Overview

The primary objective of this project is to develop an interactive dashboard that analyzes customer demographics and spending behavior. By employing unsupervised machine learning (K-Means Clustering), the project aims to identify distinct customer segments (target groups) to aid marketing strategies and business decision-making.

2. Dataset Description

Source: Kaggle ("Mall Customer Segmentation Data")

https://www.kaggle.com/datasets/abdallahwagih/mall-customers-segmentation

Size: 200 records

Attributes:

The dataset contains the following 5 variables:

1. CustomerID: Unique ID assigned to the customer.
2. Gender: Gender of the customer (Male/Female).
3. Age: Age of the customer.
4. Annual Income (k$): Annual income of the customer in thousands of dollars.
5. Spending Score (1-100): A score assigned by the mall based on customer behavior and spending nature (1 = Low spender, 100 = High spender).

3. Technology Stack & Libraries Used

The project is built using the R Programming Language within the RStudio environment. The following libraries were utilized to achieve specific functionalities:

Library	Purpose
shiny	The core framework used to build the interactive web application (UI and Server).
bslib	Used to upgrade the User Interface (UI) to a modern, responsive "Dashboard" style (Bootstrap 5).
bsicons	Provides modern icons for the Value Boxes (KPI indicators).
dplyr	Used for data manipulation, cleaning (renaming columns), and filtering.
ggplot2	The primary engine for creating static visualization charts (Histograms, Boxplots, Scatter plots).
DT	Renders the raw data as an interactive, searchable HTML table.
corrplot	Generates the graphical correlation matrix to visualize relationships between variables.
thematic	Automatically applies the dashboard's CSS theme/colors to the R plots for visual consistency.

4. Analytical Techniques & Objectives

A. Exploratory Data Analysis (EDA)

Before complex modeling, we perform EDA to understand the data distribution.

- Technique: Histograms and Boxplots.
- Objective:
  - To visualize the frequency distribution of Age, Income, and Spending Score.
  - To analyze gender-based differences (e.g., "Do females generally have a higher spending score than males?").

B. Correlation Analysis

- Technique: Pearson Correlation Coefficient (r).
- Visual: Heatmap Matrix.
- Objective: To quantify the linear relationship between numerical variables.
  - *Example:* We check if there is a strong correlation between Age and Spending Score. (A negative correlation would imply that as customers get older, they spend less).
  - *Key Insight:* In this specific dataset, Correlation Analysis often reveals that Annual Income is *not* linearly correlated with Spending Score, necessitating the use of Clustering.

C. K-Means Clustering (Unsupervised Machine Learning)

This is the core advanced analytics feature of the project.

- Technique: K-Means Algorithm.
- Objective: To segment customers into distinct groups based on two factors simultaneously: Annual Income and Spending Score.
- How it works:
  - The user selects k (number of clusters) via a slider.
  - The algorithm initiates k centroids.
  - It assigns every customer to the nearest centroid based on Euclidean distance.
  - It iteratively updates the centroids to minimize variance within groups.
- Business Value: This allows the business to identify specific segments, such as:
  - *Careful Spenders:* Low Income, Low Spending.
  - *Standard Customers:* Average Income, Average Spending.
  - *Target Customers (VIPs):* High Income, High Spending.

5. User Interface (UI) Features

The application moves beyond standard layouts by utilizing a Modern Dashboard Framework:

1. Value Boxes: Top-level KPIs (Key Performance Indicators) displaying Total Customers, Average Income, and Average Spending Score for immediate insights.
2. Card-Based Layout: Visualizations are contained within distinct "cards" with headers and borders for better organization.
3. Interactive Controls:
   - Dropdowns: To switch between variables dynamically.
   - Sliders: To adjust the k value in K-Means clustering in real-time.
4. Responsive Design: The layout automatically adjusts for different screen sizes (using the flatly Bootstrap theme).

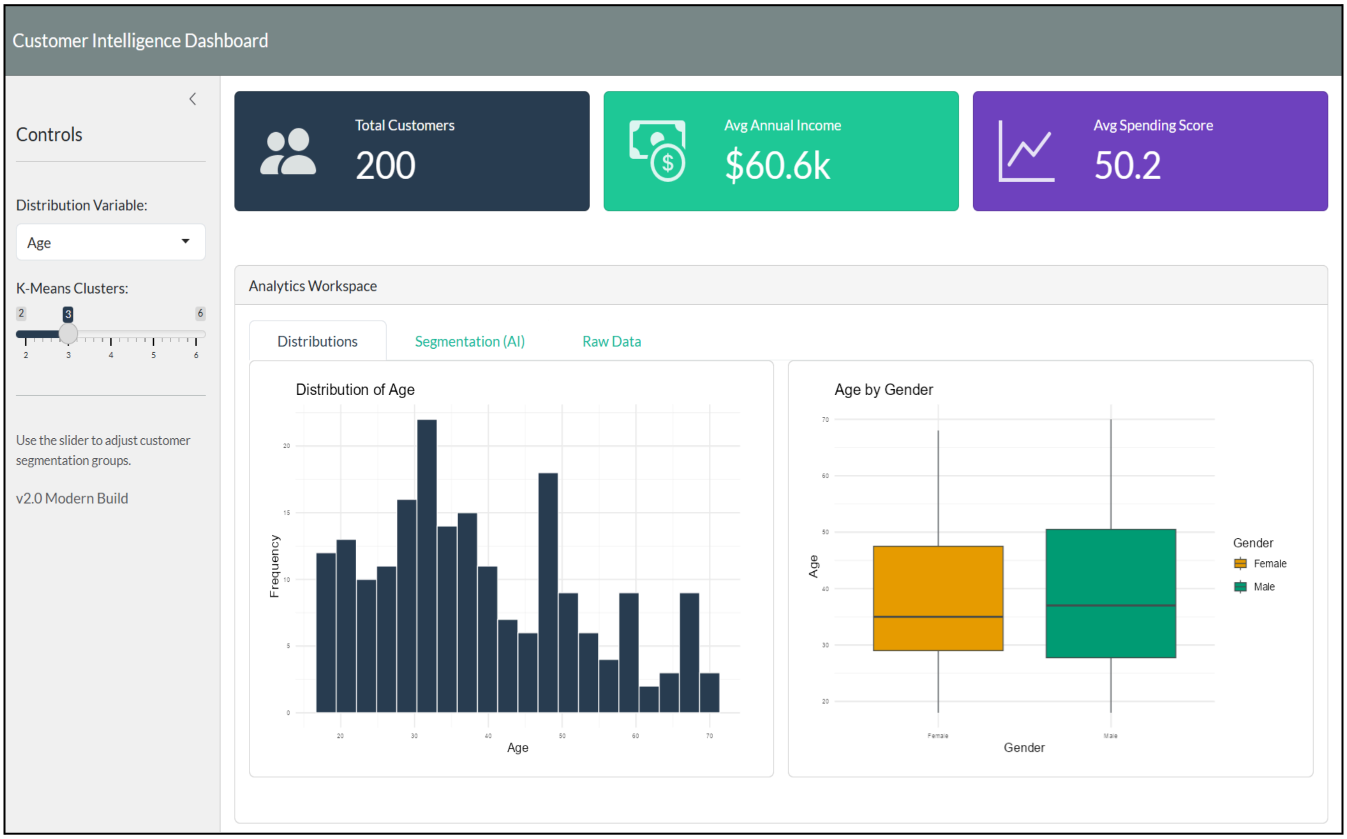
6. CODE :

app.R

```r
library(shiny)
library(ggplot2)
library(dplyr)
library(DT)
library(corrplot)
library(bslib)      # For modern UI themes and cards
library(bsicons)    # For icons in the value boxes
library(thematic)   # To make plots match the theme automatically

# --- 1. DATA LOADING & CLEANING ---
df <- tryCatch({
  data <- read.csv("Mall_Customers.csv")
  colnames(data) <- c("CustomerID", "Gender", "Age", "Annual_Income_k", "Spending_Score")
  data
}, error = function(e) {
  # Fallback data if file is missing (so app doesn't crash during testing)
  data.frame(
    CustomerID = 1:200,
    Gender = sample(c("Male", "Female"), 200, replace = TRUE),
    Age = sample(18:70, 200, replace = TRUE),
    Annual_Income_k = sample(15:137, 200, replace = TRUE),
    Spending_Score = sample(1:100, 200, replace = TRUE)
  )
})

# Enable thematic for auto-styling plots to match the dashboard
thematic_shiny()

# --- 2. UI SECTION ---
ui <- page_sidebar(

  # Set a modern theme (try "flatly", "minty", "journal", or "darkly")
  theme = bs_theme(bootswatch = "flatly"),

  title = "Customer Intelligence Dashboard",

  # The Sidebar
  sidebar = sidebar(
    title = "Controls",
    selectInput("hist_var", "Distribution Variable:",
                choices = c("Age", "Annual_Income_k", "Spending_Score")),

    sliderInput("clusters", "K-Means Clusters:",
                min = 2, max = 6, value = 5),

    hr(),
    helpText("Use the slider to adjust customer segmentation groups."),
    p(class = "text-muted", "v2.0 Modern Build")
  ),

  # The Main Dashboard Area

  # ROW 1: Key Performance Indicators (Value Boxes)
  layout_columns(
    fill = FALSE,
    value_box(
      title = "Total Customers",
      value = nrow(df),
      showcase = bs_icon("people-fill"),
      theme = "primary"
    ),
    value_box(
      title = "Avg Annual Income",
      value = paste0("$", round(mean(df$Annual_Income_k), 1), "k"),
      showcase = bs_icon("cash-coin"),
      theme = "teal"
    ),
    value_box(
      title = "Avg Spending Score",
      value = round(mean(df$Spending_Score), 1),
      showcase = bs_icon("graph-up"),
      theme = "purple"
    )
  ),

  br(), # Little bit of spacing

  # ROW 2: The Main Tabs inside a Card
  card(
    card_header("Analytics Workspace"),
    tabsetPanel(

      # Tab 1: Visualization
      tabPanel("Distributions",
               layout_columns(
                 col_widths = c(6, 6),
                 card(full_screen = TRUE, plotOutput("dist_plot")),
                 card(full_screen = TRUE, plotOutput("box_plot"))
               )
      ),

      # Tab 2: Advanced Analysis
      tabPanel("Segmentation (AI)",
```

```r
               layout_columns(
                 col_widths = c(5, 7), # 5 parts correlation, 7 parts clustering
                 card(
                   card_header("Correlation Matrix"),
                   plotOutput("corr_plot")
                 ),
                 card(
                   card_header("K-Means Cluster Map"),
                   plotOutput("cluster_plot")
                 )
               )
      ),

      # Tab 3: Data Table
      tabPanel("Raw Data",
               DTOutput("raw_table")
      )
    )
  )
)

# --- 3. SERVER SECTION ---
server <- function(input, output) {

  # 1. Render Data Table
  output$raw_table <- renderDT({
    datatable(df, options = list(pageLength = 10, scrollX = TRUE),
              style = "bootstrap4") # Modern table style
  })

  # 2. Render Distribution Histogram
  output$dist_plot <- renderPlot({
    req(input$hist_var)
    ggplot(df, aes_string(x = input$hist_var)) +
      geom_histogram(bins = 20, fill = "#2C3E50", color = "white") + # Matching Flatly theme dark blue
      labs(title = paste("Distribution of", input$hist_var), y = "Frequency") +
      theme_minimal(base_size = 14)
  })

  # 3. Render Boxplot
  output$box_plot <- renderPlot({
    req(input$hist_var)
    ggplot(df, aes_string(x = "Gender", y = input$hist_var, fill = "Gender")) +
      geom_boxplot() +
      labs(title = paste(input$hist_var, "by Gender")) +
      theme_minimal(base_size = 14)
  })

  # 4. Render Correlation Plot
  output$corr_plot <- renderPlot({
    num_data <- df %>% select(Age, Annual_Income_k, Spending_Score)
    M <- cor(num_data)
    corrplot(M, method = "color", type = "upper",
             addCoef.col = "black",
             tl.col = "black", tl.srt = 45,
             diag = FALSE)
  })

  # 5. Render Clustering (K-Means)
  output$cluster_plot <- renderPlot({
    cluster_data <- df %>% select(Annual_Income_k, Spending_Score)
    set.seed(123)
    kmeans_result <- kmeans(cluster_data, centers = input$clusters)
    cluster_data$Cluster <- as.factor(kmeans_result$cluster)

    ggplot(cluster_data, aes(x = Annual_Income_k, y = Spending_Score, color = Cluster)) +
      geom_point(size = 5, alpha = 0.8) +
      stat_ellipse(aes(fill = Cluster), geom = "polygon", alpha = 0.2) + # Adds fancy circles around clusters
      theme_minimal(base_size = 14) +
      labs(title = paste("Customer Segments (k =", input$clusters, ")"),
           x = "Annual Income (k$)",
           y = "Spending Score (1-100)") +
      scale_color_brewer(palette = "Set1") +
      scale_fill_brewer(palette = "Set1")
  })
}

# --- 4. RUN APP ---
shinyApp(ui = ui, server = server)
```

OUTPUT (Screenshots):



Customer Intelligence Dashboard

Controls

Distribution Variable: Age

K-Means Clusters: 3

Use the slider to adjust customer segmentation groups.

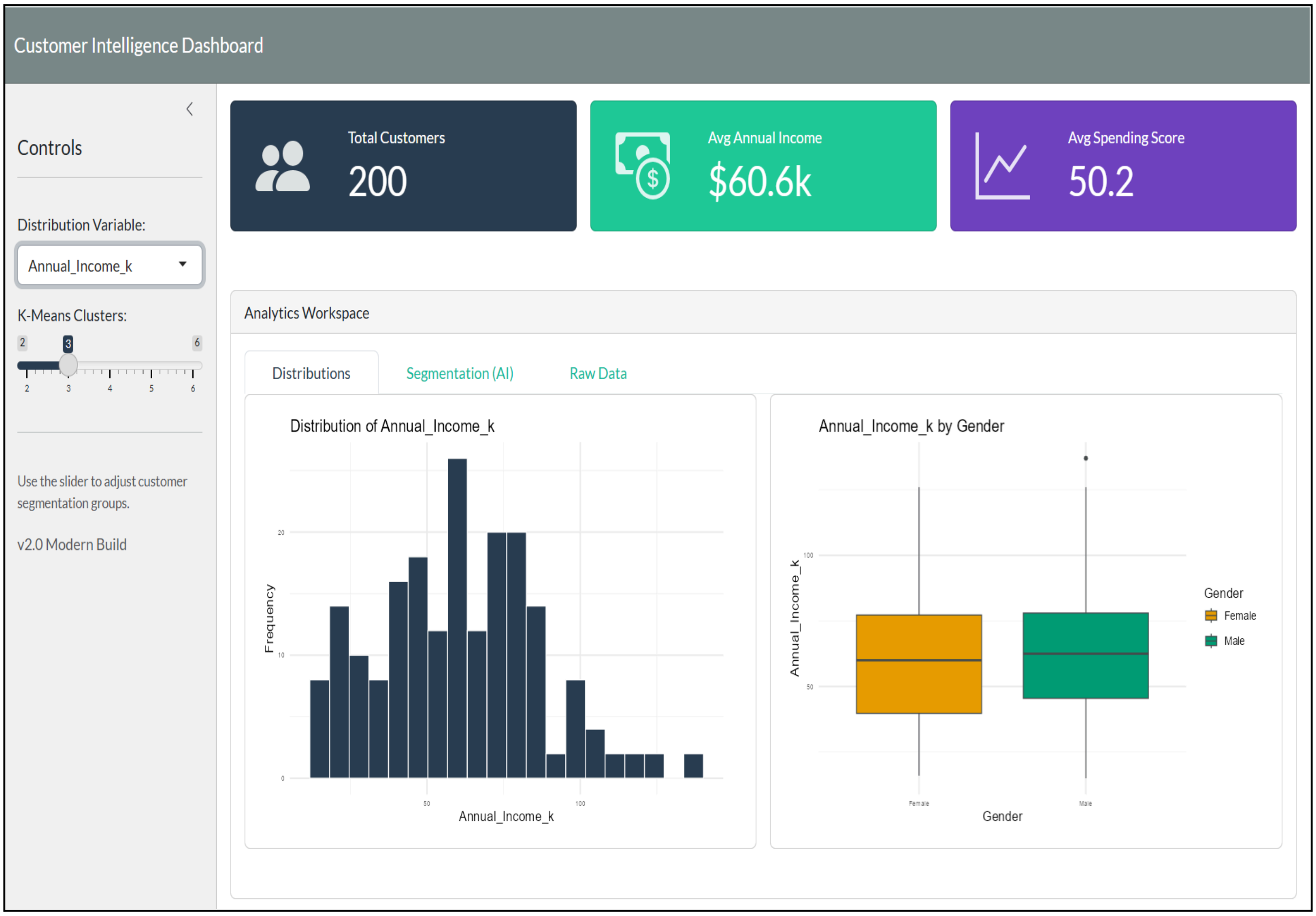
v2.0 Modern Build

Total Customers 200

Avg Annual Income $60.6k

Avg Spending Score 50.2

Analytics Workspace



Customer Intelligence Dashboard

Controls

Distribution Variable: Annual_Income_k

K-Means Clusters: 3

Use the slider to adjust customer segmentation groups.

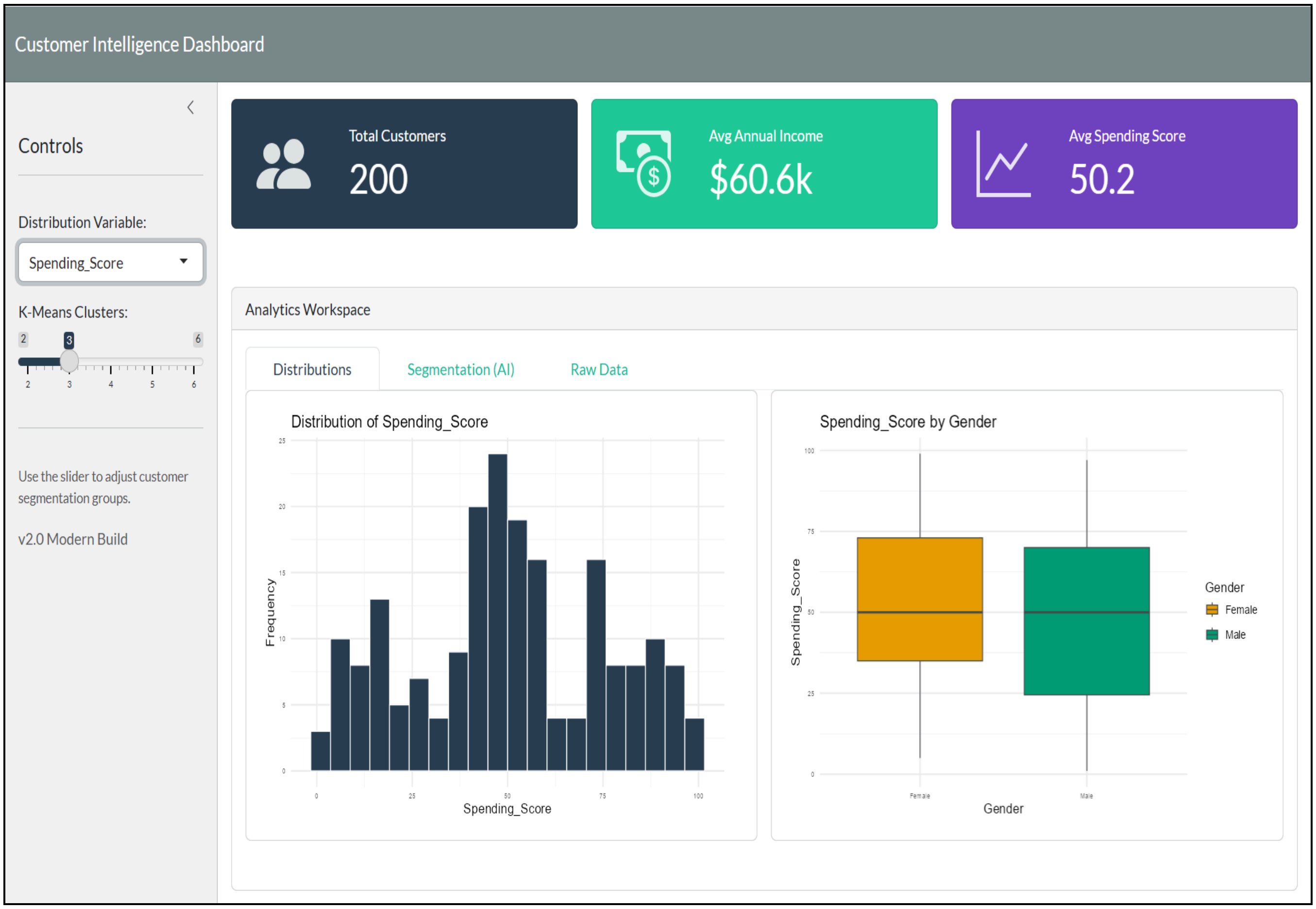
v2.0 Modern Build

Total Customers 200

Avg Annual Income $60.6k

Avg Spending Score 50.2

Analytics Workspace



Customer Intelligence Dashboard

Controls

Distribution Variable:

Spending_Score

K-Means Clusters:

Use the slider to adjust customer segmentation groups.

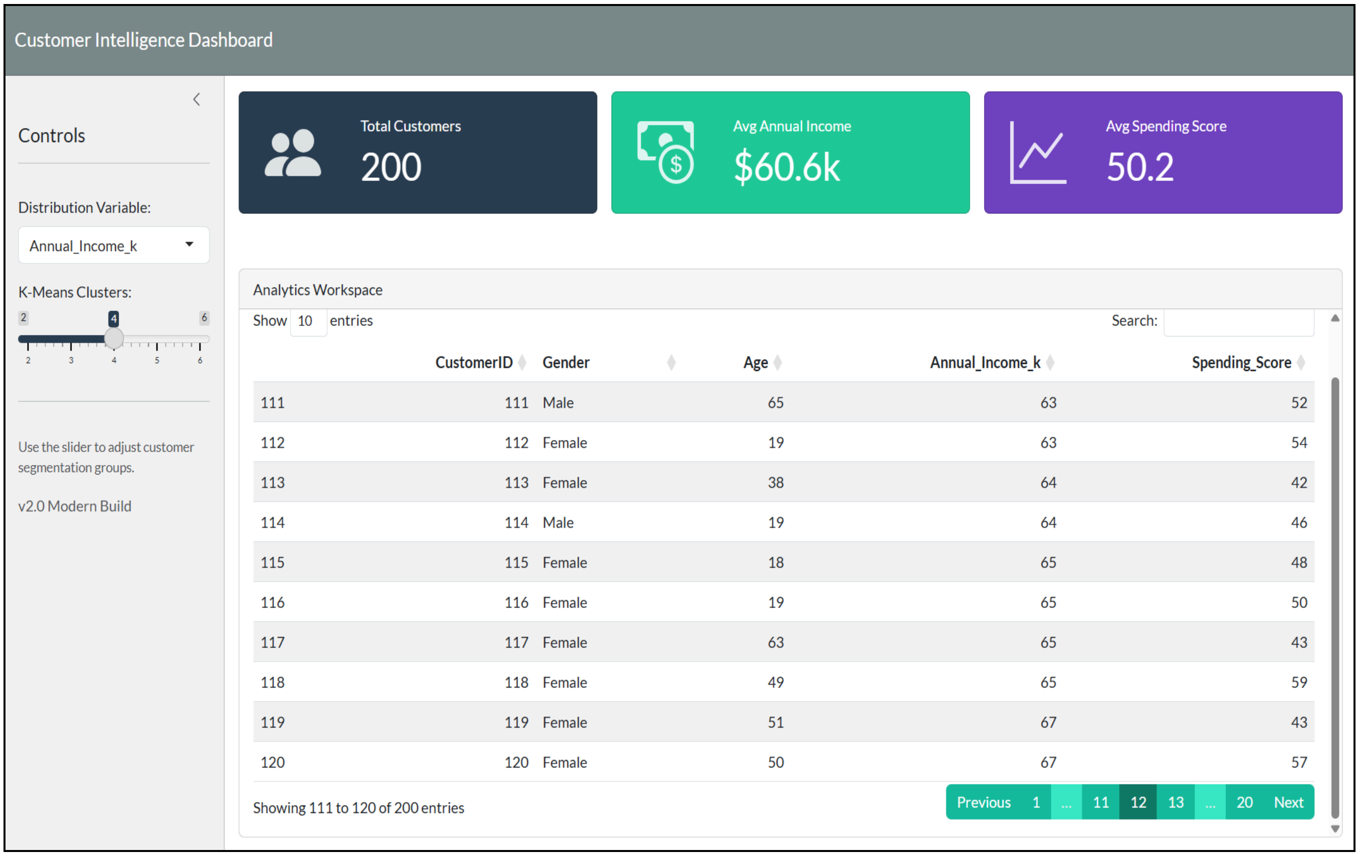
v2.0 Modern Build

Total Customers 200

Avg Annual Income $60.6k

Avg Spending Score 50.2

Analytics Workspace



Customer Intelligence Dashboard

Controls

Distribution Variable:

Annual_Income_k

K-Means Clusters:

Use the slider to adjust customer segmentation groups.

v2.0 Modern Build

Total Customers 200

Avg Annual Income $60.6k

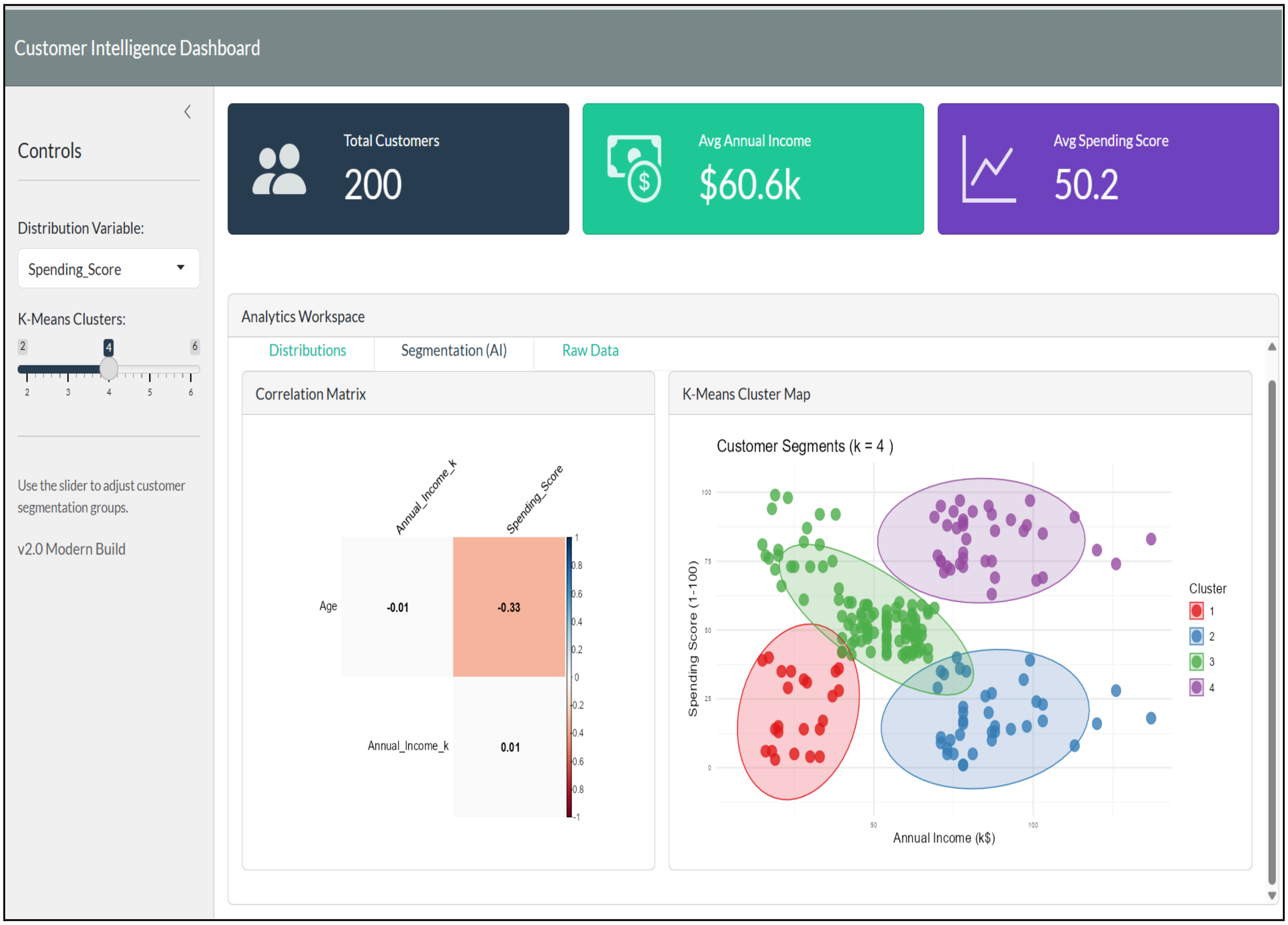
Avg Spending Score 50.2

Analytics Workspace

Show 10 entries Search:

	CustomerID	Gender	Age	Annual_Income_k	Spending_Score
111	111	Male	65	63	52
112	112	Female	19	63	54
113	113	Female	38	64	42
114	114	Male	19	64	46
115	115	Female	18	65	48
116	116	Female	19	65	50
117	117	Female	63	65	43
118	118	Female	49	65	59
119	119	Female	51	67	43
120	120	Female	50	67	57

Showing 111 to 120 of 200 entries



Customer Intelligence Dashboard

Controls

Distribution Variable: Spending_Score

K-Means Clusters: 4

Use the slider to adjust customer segmentation groups.

v2.0 Modern Build

Total Customers 200

Avg Annual Income $60.6k

Avg Spending Score 50.2

Analytics Workspace

8. Conclusion

This project successfully transforms raw transactional data into actionable business insights. By integrating statistical analysis with machine learning (Clustering) in an interactive Shiny dashboard, we provide a tool that allows stakeholders to not only visualize past data but also identify high-value customer segments for targeted marketing campaigns.