Final Project Report - Business Analytics

Course: ISM 4420

Professor: Dr. Hemang Subramanian

Authors: Anh Hoang, Christopher Tsiomakidis

I. Problem Statement

The primary goal of this project is to assist the marketing team in creating a targeted and data-driven promotional strategy for mobile applications listed on the Google Play Store.
As an application analyst, we aim to classify the apps into distinct groups based on features such as installs, reviews, ratings, size, and price; uncover patterns through the clustering model. These clusters will reveal patterns in app popularity, user feedback, and monetization strategies. The business objectives are actionable insights for sales and advertising planning.

Based on the feedback from our last class, I refined the project scope to emphasize interpretable segmentation and practical business impact. Instead of overfitting or exploring many clusters, I finalized the number of clusters to three, which balances statistical quality and marketing usability.

Objectives:

- Use K-means clustering to classify apps into meaningful groups.
- Understand and describe characteristics of each group using visual and statistical methods.
- Recommend a promotional plan (with assumed dollar value) tailored to the behavioral and commercial traits of each cluster.

Values:

This classification enables the marketing team to:

- Identify which apps are ready for scale vs. which need optimization.
- Allocate marketing budgets based on performance patterns and potential ROI.

- Tailor promotional channels and messaging to app characteristics.

II. Dataset Cleaning

The dataset contains 10,841 entries across 13 columns. Following is the column name and description of each column:

- App: App name
- Category: Category of the app
- Rating: Average user rating (scale from 1 to 5)
- Reviews: Number of user reviews
- Size: App size
- Installs: Number of app installs
- Type: Free or Paid
- Price: App price
- Content Rating: Age-based target audience (everyone, adult, teen…)
- Genres: Sub-categories, specific app genre
- Last Updated: Date when the app last updated
- Current Ver: Current version of the app
- Android Ver: Minimum Android version required to run the app

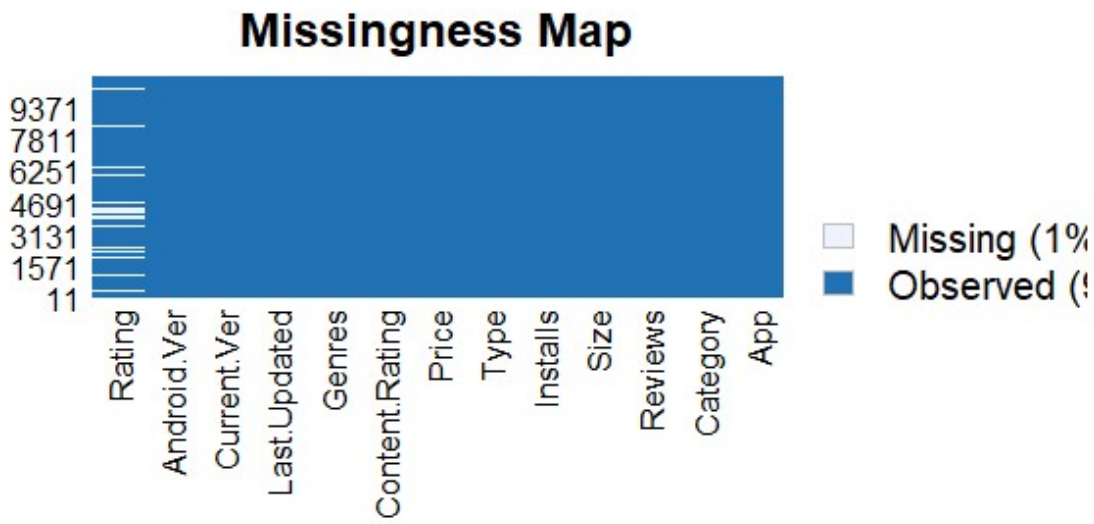
```
> head(apps)
                                                App       Category Rating Reviews Size     Installs Type Price
1     Photo Editor & Candy Camera & Grid & ScrapBook ART_AND_DESIGN    4.1     159  19M      10,000+ Free     0
2                               Coloring book moana ART_AND_DESIGN    3.9     967  14M     500,000+ Free     0
3 U Launcher Lite - FREE Live Cool Themes, Hide Apps ART_AND_DESIGN    4.7   87510 8.7M   5,000,000+ Free     0
4                                 Sketch - Draw & Paint ART_AND_DESIGN    4.5  215644  25M 50,000,000+ Free     0
5               Pixel Draw - Number Art Coloring Book ART_AND_DESIGN    4.3     967 2.8M     100,000+ Free     0
6                         Paper flowers instructions ART_AND_DESIGN    4.4     167 5.6M      50,000+ Free     0
  Content.Rating                     Genres     Last.Updated         Current.Ver  Android.Ver
1       Everyone               Art & Design  January 7, 2018               1.0.0 4.0.3 and up
2       Everyone Art & Design;Pretend Play January 15, 2018               2.0.0 4.0.3 and up
3       Everyone               Art & Design   August 1, 2018               1.2.4 4.0.3 and up
4           Teen               Art & Design     June 8, 2018 Varies with device   4.2 and up
5       Everyone   Art & Design;Creativity    June 20, 2018                 1.1   4.4 and up
6       Everyone               Art & Design   March 26, 2017                 1.0   2.3 and up
```

- Removed duplicate:

  apps <- distinct(apps)

- Missing data: approximately 99% of the dataset is observed, about 1% of the total data is missed (primarily concentrated in the 'Rating' column)

```
install.packages("Amelia")
library(Amelia)
missmap(apps)
```



- Handling missing data: Since missing data concentrated in “Rating” column, we created 2 subsets of data with and without rating. After further analysis, apps with 0 reviews and missing ratings were removed, as they provided no user feedback. Besides, Reviews has statistical correlation with Installs. Keeping them could have introduced noise into the clustering and reduced model quality.
- apps_w_rating: drop all rows that contain missing values. Dataset now has 9366 observations.

```
install.packages("tidyverse")
library(tidyverse)
apps_w_rating <- drop_na(apps)
str(apps_w_rating)
```

- Cleaning data:

- Converted character data (char) into numeric data type (num) in these columns: reviews, installs, price, size, rating
- Kept data cleaned, same format by delete non-numeric sign like $, M, K, +
- Converted size to same unit (MB)
- Replaced value 0+ value in Installs to 1 to separate more than 0 installation from no installation at all.
- Replaced “Varies with device” observations in Size column to average of app sizes in each genre.

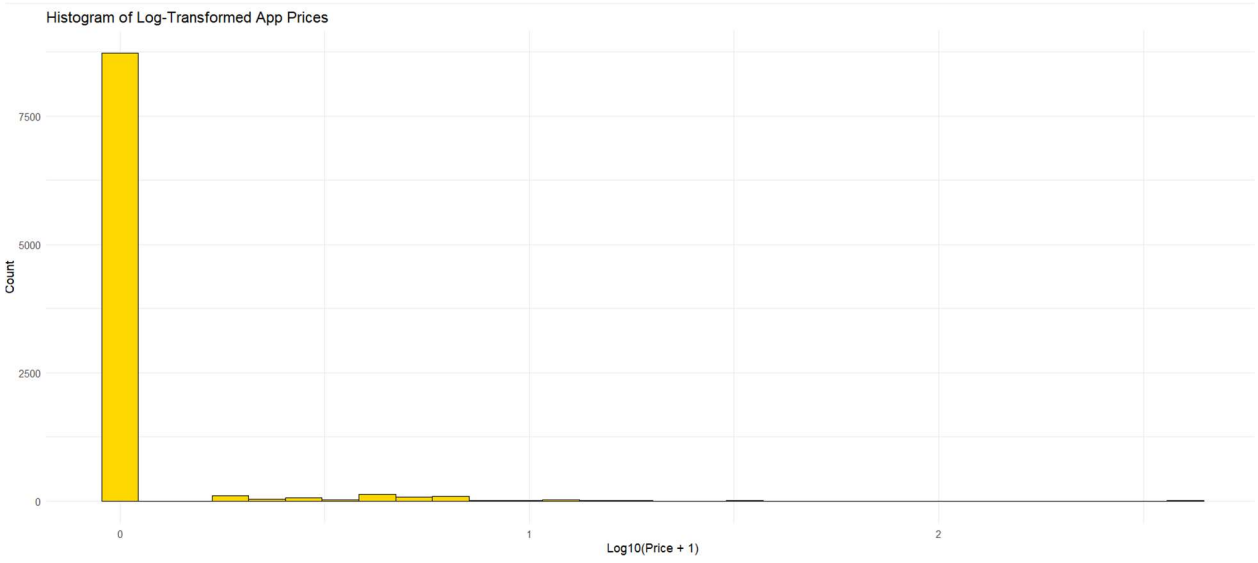
R code:

```
#cleaning Reviews and Installs
apps_w_rating <- apps_w_rating %>%
  mutate(
    Reviews = as.numeric(Reviews),
    Installs = ifelse(Installs == "0+", "1", Installs),
    Installs = str_remove_all(Installs, "[+,]"),
    Installs = as.numeric(Installs)
  )
#cleaning Price
apps_w_rating <- apps_w_rating %>%
  filter(Price != "Everyone") %>%
  mutate(
    Price = str_remove(Price, "\\$"),
    Price = as.numeric(Price)
  )
#cleaning Size
apps_w_rating <- apps_w_rating %>%
```

```
  mutate(Size = as.character(Size)) %>%
  mutate(
    Size = case_when(
      str_detect(Size, "M") ~ as.numeric(str_replace(Size, "M", "")),
      str_detect(Size, "k") ~ as.numeric(str_replace(Size, "k", "")) / 1024,
      TRUE ~ NA_real_
    )
  )
# Replaced Size$Varies_with_device
apps_w_rating <- apps_w_rating %>%
  group_by(Genres) %>%
  mutate(Size = ifelse(is.na(Size_MB), mean(Size_MB, na.rm = TRUE), Size)) %>%
  ungroup()
```

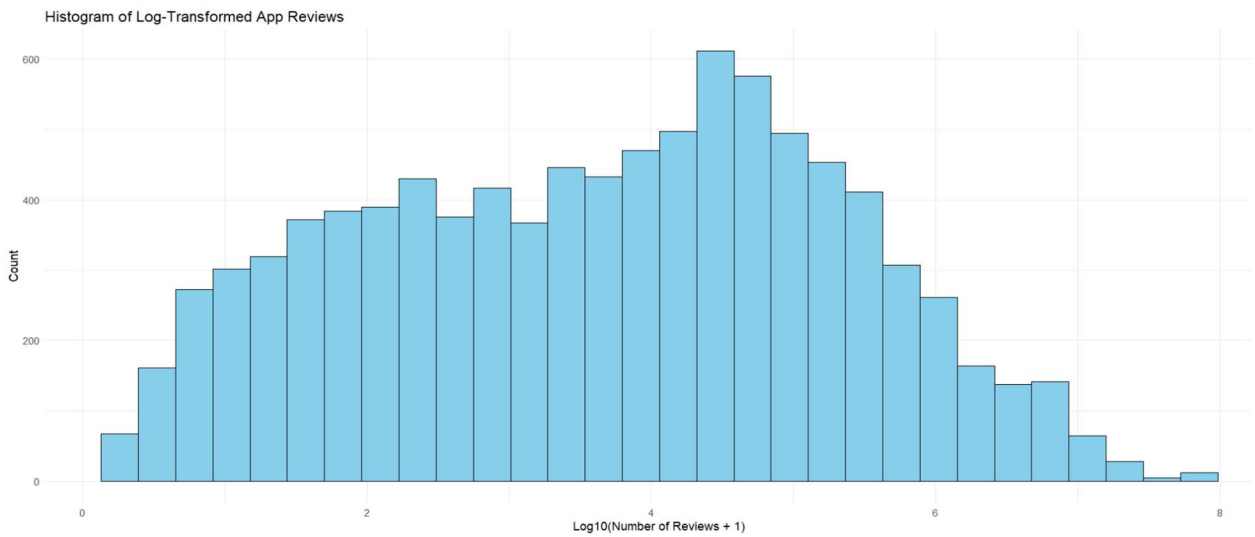
- Histogram graph:

- Price:

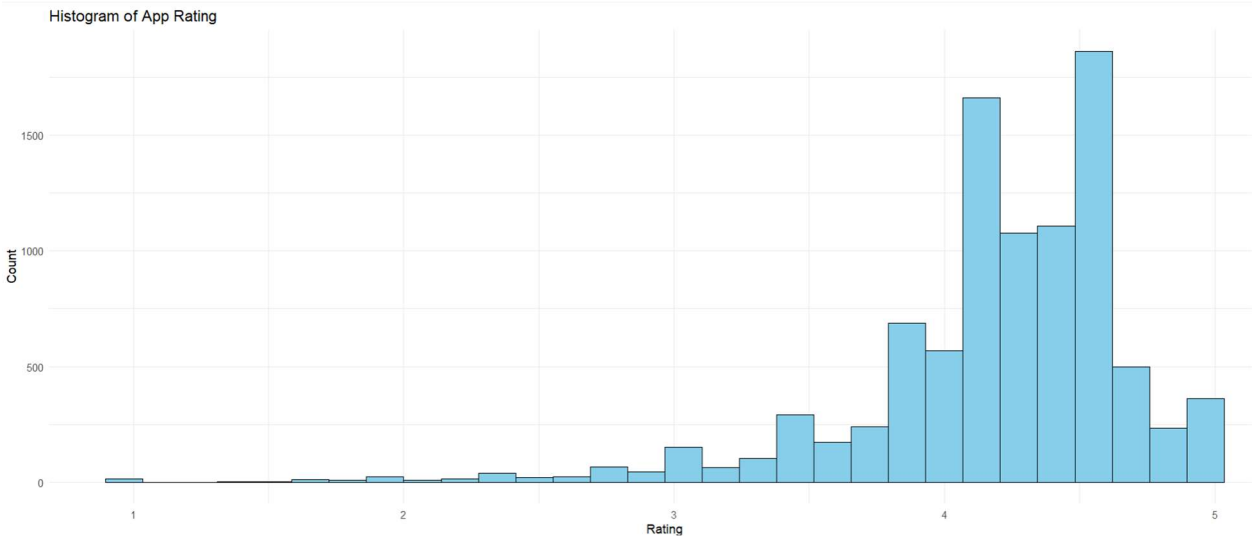


Majority of apps are free.

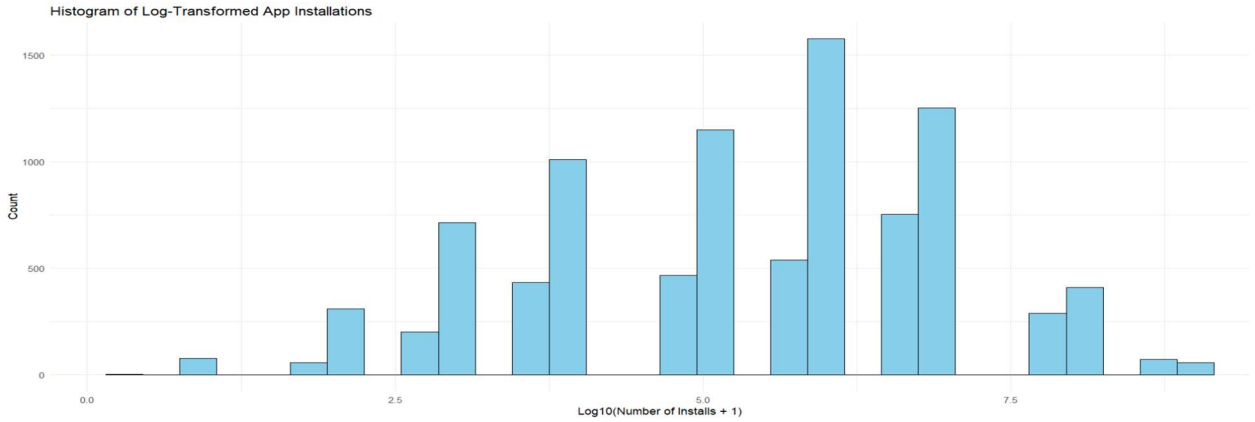
- Review:



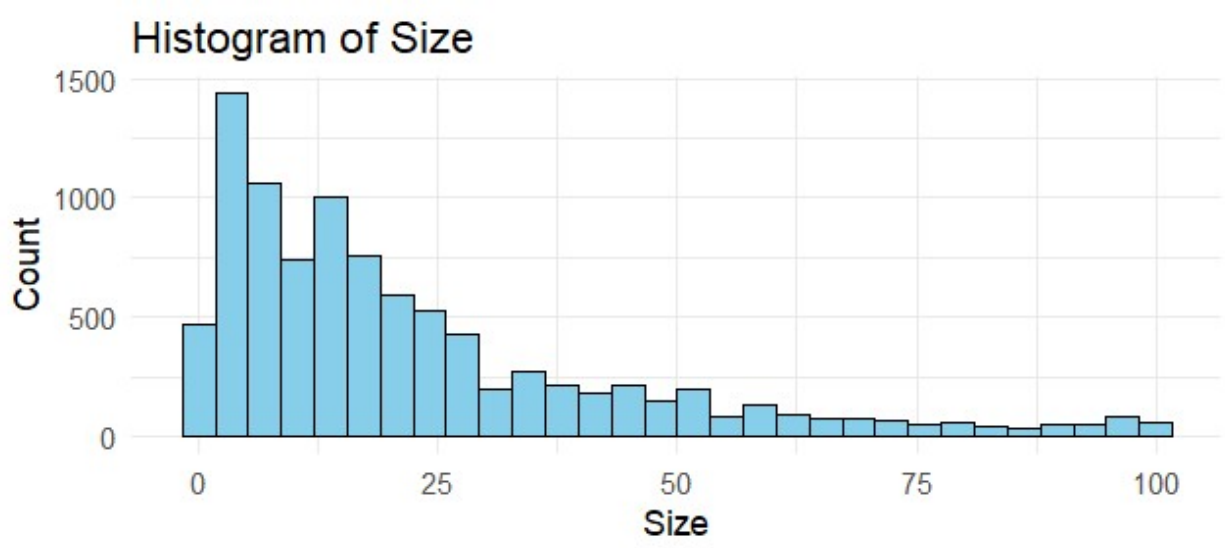
- Rating:



- Installs:



- Size:



- Summary:

```
> summary(apps_w_rating$Rating)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   4.000   4.300   4.192   4.500   5.000
> summary(apps_w_rating$Reviews)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
       1      186     5930   514050    81533 78158306
> summary(apps_w_rating$Size)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
  0.0083   6.3000  15.0000  22.2989  30.0000 100.0000
> summary(apps_w_rating$Installs)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
1.00e+00 1.00e+04 5.00e+05 1.79e+07 5.00e+06 1.00e+09
> summary(apps_w_rating$Price)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
  0.0000   0.0000   0.0000   0.9609   0.0000 400.0000
```

- Standard Deviation

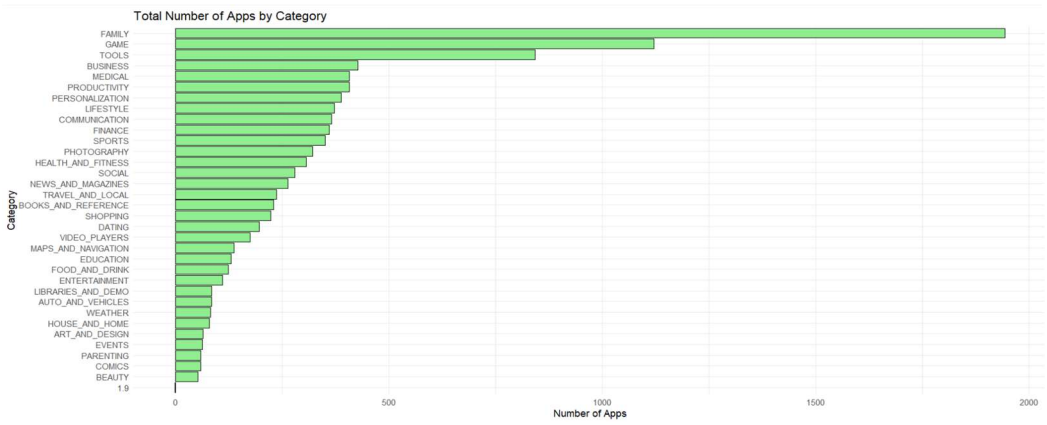
```
> sd(apps_w_rating$Size)
[1] 21.89883
> sd(apps_w_rating$Installs)
```

```
[1] 86376000
> sd(apps_w_rating$Price)
[1] 16.18934
> sd(apps_w_rating$Reviews)
[1] 2905052
> sd(apps_w_rating$Rating)
[1] 0.5223767
```

III. Dataset Description

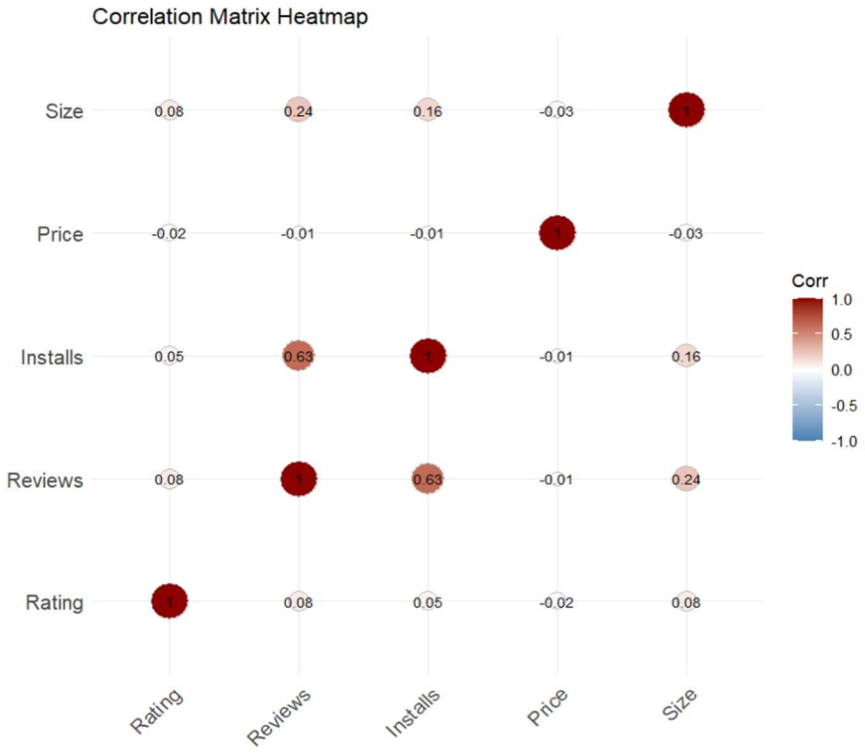
1. After data cleaning (removing duplicates, converting formats, and handling missing values)
    - Number of rows remaining: 8892
    - Number of columns used for analysis: 5
        - Price
        - Rating
        - Reviews
        - Installs
        - Size
2. Description Analysis:

- Counting apps in each category:



We analyzed the distribution of app categories by counting the number of apps in each segment to identify both highly competitive and more niche markets. Categories like FAMILY, GAME, and TOOLS have the highest volume of apps, indicating strong saturation and intense competition in those areas.

- Correlation Analysis:
    - Heat map matrix shows the strongest positive relationship is between Reviews and Installs (r = 0.63).
    - Insight: User engagement (reviews) is a strong proxy for popularity.



- Linear Regression Model:

```
> model <- lm(Installs ~ Reviews + Rating + Price + Size, data = regression_data)
> summary(model)

Call:
lm(formula = Installs ~ Reviews + Rating + Price + Size, data = regression_data)

Residuals:
       Min         1Q     Median         3Q        Max
-749798993   -8879735   -7398159   -4500276  975474658

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.385e+06  5.743e+06   0.589  0.55556
Reviews      1.889e+01  2.461e-01  76.751  < 2e-16 ***
Rating       1.456e+06  1.364e+06   1.068  0.28567
Price       -2.989e+04  4.380e+04  -0.682  0.49499
Size        -8.499e+04  3.268e+04  -2.600  0.00933 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 66820000 on 8887 degrees of freedom
Multiple R-squared:  0.4018,    Adjusted R-squared:  0.4015
F-statistic:  1492 on 4 and 8887 DF,  p-value: < 2.2e-16
```

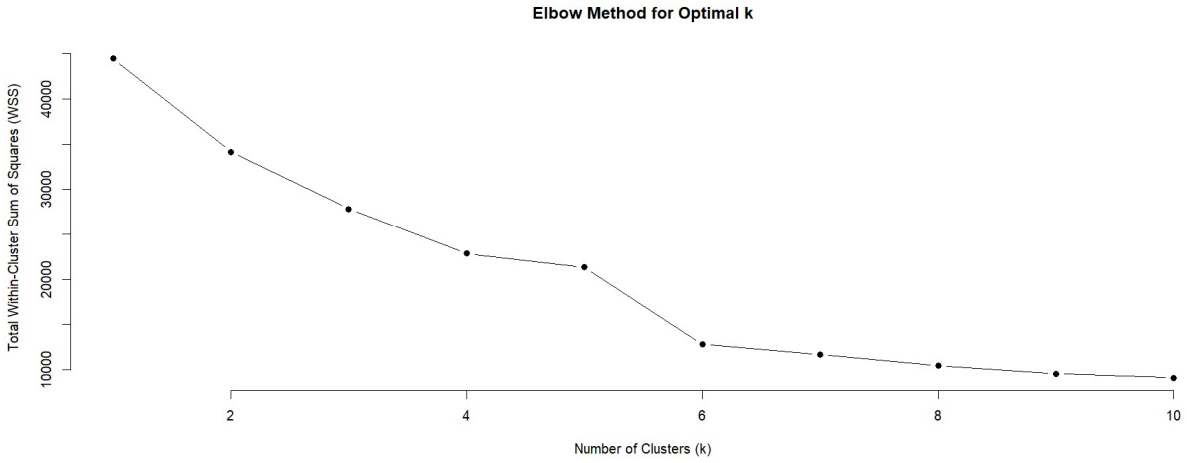
We did a linear regression model to predict Installs based on Reviews, Rating, Price, and Size suggesting the number of users (Installs). Reviews is a strong and significant predictor of app installs. On average, every additional review is associated with about 16 more installs.

3. How the Descriptive Analysis Helped Shape the Plan

- Cluster Analysis Decision:
    - K-means was chosen for its efficiency with large datasets and clear group separation.
    - Log-transformation was used to normalize skewed variables like Reviews, Price and Installs.
- Plan Relevance: The variables directly support marketing decisions, like:
    - Which apps are already popular?
    - Which apps need a promotional push?
    - Which apps are potentially undervalued or overexposed?

IV. K-mean clustering Model

- K-means partitions the dataset into K distinct groups (clusters) based on the similarity of input variables.
    - Initializes K centroids randomly
    - Assigns each data point to the nearest centroid (using a distance metric)
    - Recalculates centroids and repeats until convergence (no significant movement)
- Variable Selection Rationale:
    - Rating, Reviews, Installs, Size, and Price for clustering.
    - These variables represent app performance, visibility, engagement, and monetization potential.
- Choosing the Number of Clusters (K = 3)
    - Elbow method to examine total within-cluster sum of squares (WSS)



R code:

```
cluster_data <- apps_w_rating %>%
  select(Rating, Reviews, Installs, Price, Size) %>%
  drop_na() %>%
```

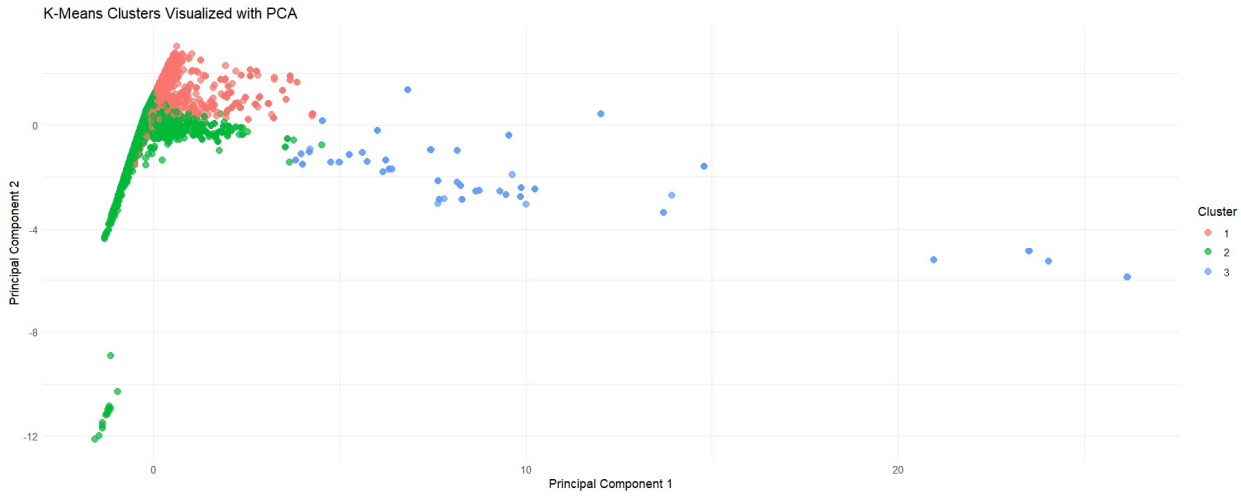
```
  mutate(across(everything(), as.numeric))

cluster_data_scaled <- scale(cluster_data)

head(cluster_data_scaled)

wss <- vector()

for (k in 1:10) {

  kmeans_result <- kmeans(cluster_data_scaled, centers = k, nstart = 10)

  wss[k] <- kmeans_result$tot.withinss

}



wss <- numeric(10)



for (k in 1:10) {

  set.seed(123)  # for reproducibility

  kmeans_result <- kmeans(cluster_data_scaled, centers = k, nstart = 10)

  wss[k] <- kmeans_result$tot.withinss

}

plot(1:10, wss, type = "b",

     pch = 19, frame = FALSE,
```

```
  xlab = "Number of Clusters (k)",

  ylab = "Total Within-Cluster Sum of Squares (WSS)",

  main = "Elbow Method for Optimal k")
```

- 3 clusters was optimal because:
    - Distinct, interpretable groupings
    - Good separation and compactness
    - Practical business relevance for marketing decisions
- Cluster Characteristics:

```
> print(cluster_summary)
# A tibble: 3 × 7
  Cluster Avg_Rating Avg_Reviews Avg_Installs Avg_Price Avg_Size Count
  <fct>        <dbl>       <dbl>        <dbl>     <dbl>    <dbl> <int>
1 1             4.29     694002.     14973747.     0.319     58.8  1767
2 2             4.16     165967.      5844129.     1.14      13.0  7010
3 3             4.33   15775619.    688695652.     0         29.4   115
>
```

We visualized them using Principal Component Analysis (PCA) to visualize multiple variables in 2-D dimension.



- Here is the PCA a matrix of loadings. It tells how the original variables contribute to each principal component (PC).

```
> pca_result$rotation
                PC1        PC2         PC3
Rating     0.14649337  0.6363593  0.33350214
Reviews    0.69100971 -0.1285973 -0.03193939
Installs   0.67948259 -0.1986270 -0.07160475
Price     -0.03350858 -0.4314637  0.90096155
Size       0.19551436  0.5940505  0.26626537
```

    - PC1 is mainly a combination of Reviews and Installs → indicates popularity.
    - PC2 is driven by Rating and Size → may reflect user satisfaction or app complexity.

Summary:

Cluster 1: These are popular, well-rated apps with high install and review counts, likely offering strong user experience and broad appeal. The pricing is mostly free or very low, and app sizes are relatively large, indicating rich features.

Cluster 2: This cluster includes paid or freemium apps with moderate installs and reviews. Lower app size suggests simpler interfaces or utility-based use. These apps are decent in quality (highest average price), may be newly leased or more niche.

Cluster 3: These are top-tier, viral apps with extremely high reach and reviews, mostly for everyone, apps like TikTok, Instagram, Google or big games. All are free, which helps explain their huge adoption. They sit far right in the PCA plot (PC1 = popularity).

V. Planning

Based on our cluster analysis, we propose three differentiated promotional strategies, each tailored to the behavioral patterns, monetization models, and marketing readiness of the app groups.

Cluster 1 – "Solid Performers"

Traits:

- Moderate to high installs and reviews
- Good ratings
- Mixed monetization (free and low-cost apps)
- Rich in features (larger app size)

Strategy:

These apps already demonstrate user trust and traction. A **scaling strategy** will enhance discoverability and increase ROI. Use **online ads** (Google Ads, Meta Ads) with targeted campaigns based on user interests and lookalike audiences.

Plan:

- Monthly ad spend: **$3,000**
- Channel: Google Ads (search + display), YouTube pre-rolls
- KPI focus: Installs, cost per install (CPI), user retention
- Messaging: “Join 1M+ users enjoying this app” / “Top-rated app for your needs”

Expected ROI:

Given the solid foundation, we estimate a 15–20% increase in installs with targeted visibility over a 4-week campaign window.

Cluster 2 – "Niche/Newcomers"

Traits:

- Moderate installs and reviews
- Freemium or paid structure
- Smaller size, often simpler in function
- Highest average price (monetization focus)

Strategy:

These apps need **credibility building**. Use **influencer partnerships**, **PR blogs**, and **micro-targeting** to communicate value. Supplement with small-budget ads to build traction and user trust.

Plan:

- Monthly ad spend: **$2,500**
- Channel: Instagram/TikTok influencer partnerships, Tech blog mentions, Reddit Ads
- KPI focus: Conversion rate, review volume, trial rate
- Messaging: “Try before you buy” / “Solve [specific problem] in seconds”

Expected ROI:

Conversion rate could double due to niche targeting. Success depends on effective storytelling and third-party validation.

Cluster 3 – "Market Leaders"

Traits:

- Massive installs and reviews
- All free apps
- Strong brand recognition
- Viral potential (e.g., social, entertainment)

Strategy:

These apps don't need awareness—they need **continued engagement** and **retention marketing**. Traditional ads offer minimal ROI here. Instead, invest in **in-app promotions**, **feature rollouts**, and **TV ad placements** during major events for brand reinforcement.

Plan:

- Monthly media budget: **$10,000**
- Channel: National TV ads (primetime), App Store front page promotion, co-branding deals
- KPI focus: Daily Active Users (DAU), engagement duration, ad revenue per user
- Messaging: "New feature just dropped—have you tried it?" / "Join the experience 1B+ users love!"

Expected ROI:

Given the size of the user base, even a 1% engagement lift can translate to significant ad revenue. Brand reinforcement builds long-term loyalty

To Summarize Our Plan:

These promotional plans were designed to align with each cluster's unique strengths and market behavior. High-visibility, free apps receive the largest allocation to sustain mass engagement, while high-priced, niche apps focus on building trust and conversions. Solid performers are scaled through targeted ads to maximize ROI and expand their reach.