R Exercise Tasks

Seminar 5

Instructor: Prof. Lee, Gun-woong
Nanyang Business School

Make sure to clear memory before starting

```
> rm(list=ls())          # It will  clean up memory!
```

Mobile App Data

- Description
  - Describe the Characteristics of Data
    - Source, Collection Date, Sample Size, and Structure & Type

"Data for our analyses was collected for the top 300 paid mobile applications (apps) from U.S. Apple App Store on 19th July, 2013. A total of 1,200 apps appeared on the charts in Business, Finance, Games, and Health, categories are included in the data. A set of app-level variables such as ranking, released date, price, seller, developer, number of screenshots, size, user ratings, and version are recorded. We utilize the balanced cross-sectional data. Each app category equally contains individual 300 apps in the top charts…"

Plants vs. Zombies

By PopCap

Open iTunes to buy and download apps.

Game Center

View More By This Developer

View In iTunes

$0.99

Category: Games
Updated: Oct 10, 2012
Version: 1.9.7
Size: 75.6 MB
Languages: English, French, German, Italian, Spanish
Seller: PopCap
© 2009 PopCap Games, Inc.
Rated 9+ for the following:
Infrequent/Mild Cartoon or Fantasy Violence
Infrequent/Mild Horror/Fear Themes

Requirements: Compatible with iPhone, iPod touch, and iPad. Requires iOS 4.3 or later.

Customer Ratings

Current Version:
★★★★★ 58439 Ratings
All Versions:
★★★★★ 423521 Ratings

Top In-App Purchases

1. Dr. Zomboss' Pig... $1.99
2. Digger's Diamond... $3.99
3. Sunflower's Golde... $0.99
4. Stinky's Secret Sta... $0.99
5. Crazy Dave's Lost ... $0.99
6. Marigold's Coin Pot $0.99

Description

Get ready to soil your plants as a mob of fun-loving zombies is about to invade your home. Use your arsenal of 49 zombie-zapping plants — peashooters, wall-nuts, cherry bombs and more — to mulchify 26 types of zombies before they break down your door.

PopCap Web Site ▸ Plants vs. Zombies Support ▸ Application License Agreement ▸ ...More

What's New in Version 1.9.7

– Fix for orientation bug in iOS 6 on iPad
– Other minor bug fixes

iPhone Screenshots

Customer Reviews

Variables

Name	Description
AppID	An App's unique 9-digit identification number
Category	An App's category
Rank	An App's daily sales ranking in the top chart
Title	An App's title
Price	An App's price
Screenshots	Number of screenshots in an App's product description page
Rating_Score	Averaged user review score for an App (0 to 5 scale)
Rating_Num	Accumulated number of user review ratings for an App

Data Preparation

- Data Import
  - Download the data files and save them into your project folder
    - Apps.csv and Apps_Ratings.csv
  - Import the data files from the project folder
    - Apps_data <- read.csv('Apps.csv')
    - Ratings_data <- read.csv('Apps_Ratings.csv')
- Merging Data Sets
  - Understand the datasets

```
> head(Apps_data)
     AppID Category Rank                                  Title Price Screenshots
1 281656475    Games  212                               PAC-MAN  4.99           5
2 283035434  Finance  168    SplashMoney - Personal Finance Manager  4.99           5
3 283281224  Finance  160                                 Budget  0.99           2
4 283494170  Finance   74 PocketMoney - checkbook, budgets, expenses  4.99           5
5 284428991 Business   30                               Recorder  0.99           4
6 284650141  Finance  261                                   Tips  0.99           3
```

```
> head(Ratings_data)
      AppID Rating_Score Rating_Num
1 342548956          4.5      11302
2 294934058          3.5       2392
3 428974099          3.5       5124
4 347803339          4.5       2623
5 307868751          4.0       7039
6 577499909          4.0        148
```

  - Combine the two datasets by *AppID*

```
> head(Apps)
     AppID Category Rank                                  Title Price Screenshots Rating_Score Rating_Num
1 281656475    Games  212                               PAC-MAN  4.99           5          4.0      11360
2 283035434  Finance  168    SplashMoney - Personal Finance Manager  4.99           5          2.5       1071
3 283281224  Finance  160                                 Budget  0.99           2          3.0       3323
4 283494170  Finance   74 PocketMoney - checkbook, budgets, expenses  4.99           5          3.5       3678
5 284428991 Business   30                               Recorder  0.99           4          3.5      11389
6 284650141  Finance  261                                   Tips  0.99           3          3.5        282
```

Data Understanding

- **Understand the Data Structure and Variables**
  - Identify the Structure of Data

```
> str(Apps)
'data.frame':   1202 obs. of  8 variables:
 $ AppID       : int  281656475 283035434 283281224 283494170 284428991 284650141 284652710 284815117 284918921 284938955 ...
 $ Category    : Factor w/ 4 levels "Business","Finance",..: 3 2 2 2 1 2 1 3 1 2 ...
 $ Rank        : int  212 168 160 74 30 261 293 83 298 292 ...
 $ Title       : Factor w/ 1199 levels "???¡§?¢â?¢Ó???¡¾???????? ???¡x?¢â?¢æ??¢â¨¡¡¯?¢â?¢Ó?¢â?????¨£",..: 746 984 139 799 849 1081 976 915 1101 503 ...
 $ Price       : num  4.99 4.99 0.99 4.99 0.99 0.99 1.99 1.99 2.99 1.99 ...
 $ Screenshots : int  5 5 2 5 4 3 5 5 5 4 ...
 $ Rating_Score: num  4 2.5 3 3.5 3.5 3.5 3.5 4 3 4 ...
 $ Rating_Num  : int  11360 1071 3323 3678 11389 282 2782 96800 1061 124 ...
```

  - Present the summary statistics for the App variables

```
> summary(Apps[,c('Rank','Price','Screenshots','Rating_Score','Rating_Num')])
      Rank            Price          Screenshots     Rating_Score     Rating_Num
 Min.   :  1.0   Min.   : 0.990   Min.   :0.000   Min.   :0.000   Min.   :     0.0
 1st Qu.: 75.0   1st Qu.: 0.990   1st Qu.:5.000   1st Qu.:3.500   1st Qu.:    50.2
 Median :150.0   Median : 1.990   Median :5.000   Median :4.000   Median :   276.5
 Mean   :150.3   Mean   : 3.145   Mean   :4.656   Mean   :3.786   Mean   :  8201.7
 3rd Qu.:225.0   3rd Qu.: 3.990   3rd Qu.:5.000   3rd Qu.:4.500   3rd Qu.:  2181.0
 Max.   :300.0   Max.   :89.990   Max.   :5.000   Max.   :5.000   Max.   :823547.0
                 NA's   :25
```

Data Cleaning

- **Variable Conversion**
  - Identify the value type of *Title*
  - Convert the values to character strings

```
> str(Apps)
'data.frame':   1202 obs. of  8 variables:
 $ AppID       : int  281656475 283035434 283281224 283494170 284428991 284650141 284652710 284815117 284918921 284938955 ...
 $ Category    : Factor w/ 4 levels "Business","Finance",..: 3 2 2 2 1 2 1 3 1 2 ...
 $ Rank        : int  212 168 160 74 30 261 293 83 298 292 ...
 $ Title       : chr  "PAC-MAN" "SplashMoney - Personal Finance Manager" "Budget" "PocketMoney - checkbook, budgets, expenses" ...
 $ Price       : num  4.99 4.99 0.99 4.99 0.99 0.99 1.99 1.99 2.99 1.99 ...
 $ Screenshots : int  5 5 2 5 4 3 5 5 5 4 ...
 $ Rating_Score: num  4 2.5 3 3.5 3.5 3.5 3.5 4 3 4 ...
 $ Rating_Num  : int  11360 1071 3323 3678 11389 282 2782 96800 1061 124 ...
```

Data Cleaning

- **Missing Values**
  - Impute the missing values in *Price*
    - *Identify the missing values (NA) in Price (Hint: is.na(Apps$Price))*
      - *How many missing values in Price?? Which Apps have missing prices??*
    - *Replace the missing values with the mean of Price (i.e., mean(Apps$Price, na.rm=TRUE))*

```
> summary(Apps[,c('Rank', 'Price','Screenshots','Rating_Score','Rating_Num')])
      Rank            Price          Screenshots     Rating_Score     Rating_Num
 Min.   :  1.0   Min.   : 0.990   Min.   :0.000   Min.   :0.000   Min.   :     0.0
 1st Qu.: 75.0   1st Qu.: 0.990   1st Qu.:5.000   1st Qu.:3.500   1st Qu.:    50.2
 Median :150.0   Median : 1.990   Median :5.000   Median :4.000   Median :   276.5
 Mean   :150.3   Mean   : 3.145   Mean   :4.656   Mean   :3.786   Mean   :  8201.7
 3rd Qu.:225.0   3rd Qu.: 3.990   3rd Qu.:5.000   3rd Qu.:4.500   3rd Qu.:  2181.0
 Max.   :300.0   Max.   :89.990   Max.   :5.000   Max.   :5.000   Max.   :823547.0
```

Note: No Missing Values in *Price*

Data Cleaning

- **Handling Outliners**
  - Identify the outliers in *Rating_Num*
    - *Any values over 3 SDs from the mean are outliers*
    - ✓ *Hint: Outliers if Rating_Num > Mean+3*SD*
      *or Rating_Num < Mean-3*SD*
  - Remove the outlier values in *Rating_Num*
    - ✓ *Hint: Apps$Rating_Num[i] <- NA*
  - Identify the number missing values in *Rating_Num*
    - *# outliers = # NA's*

```
> summary(Apps[,c('Rank','Price','Screenshots','Rating_Score','Rating_Num')])
      Rank            Price          Screenshots     Rating_Score     Rating_Num
 Min.   :  1.0   Min.   : 0.990   Min.   :0.000   Min.   :0.000   Min.   :     0
 1st Qu.: 75.0   1st Qu.: 0.990   1st Qu.:5.000   1st Qu.:3.500   1st Qu.:    48
 Median :150.0   Median : 1.990   Median :5.000   Median :4.000   Median :   256
 Mean   :150.3   Mean   : 3.145   Mean   :4.656   Mean   :3.786   Mean   :  4076
 3rd Qu.:225.0   3rd Qu.: 3.990   3rd Qu.:5.000   3rd Qu.:4.500   3rd Qu.:  1997
 Max.   :300.0   Max.   :89.990   Max.   :5.000   Max.   :5.000   Max.   :132339
                                                                  NA's   :14
```

Data Pre-processing

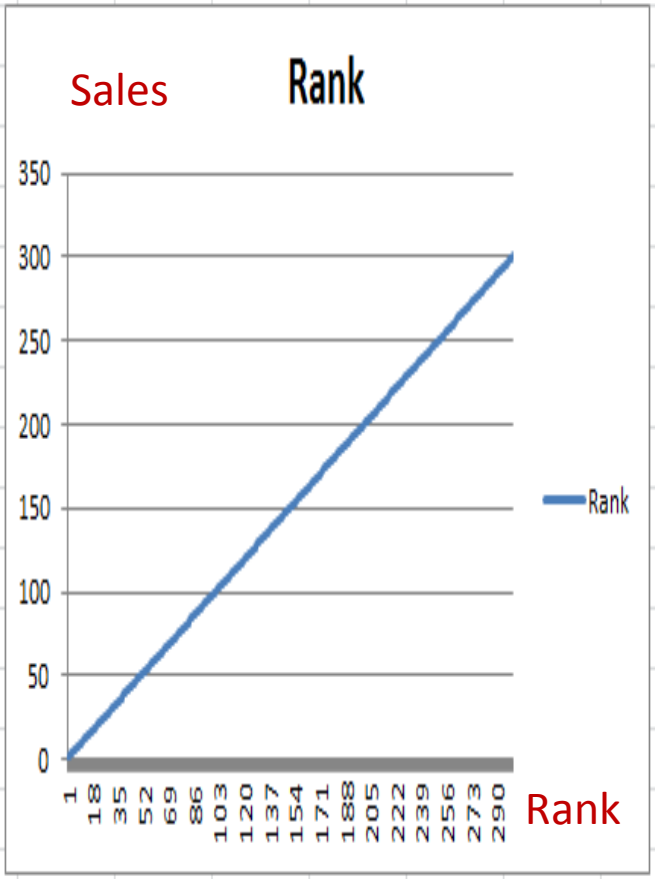
- **Normalization**
  - Normalize the values in *Rating_Num*
    - Create a Normalize() function taking a column vector *x* of numeric values (e.g., *Review_Num*). The function should return the normalized values.

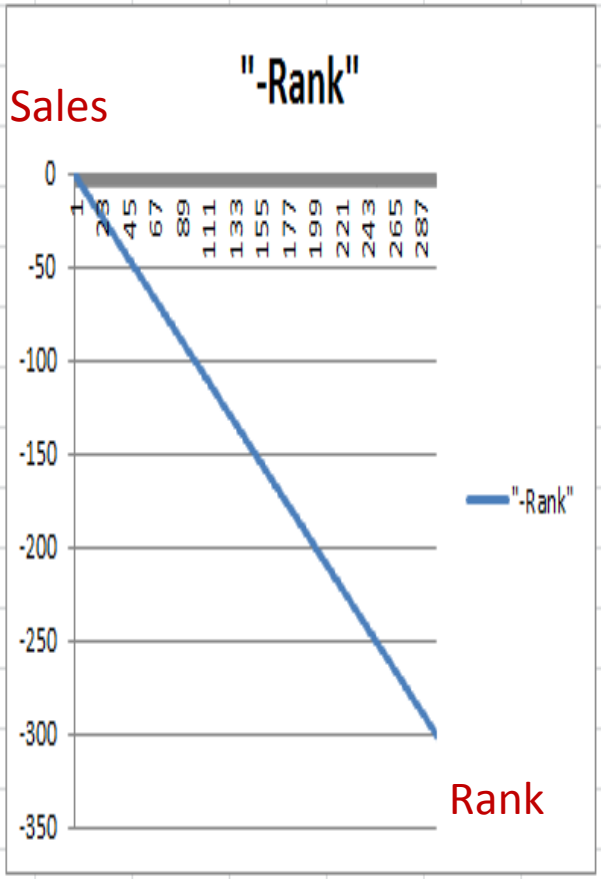
✓ Hint: Return($\frac{x-\min(x,\ na.rm=TRUE)}{\max(x,\ na.rm=TRUE)-\min(x,\ na.rm=TRUE)}$)

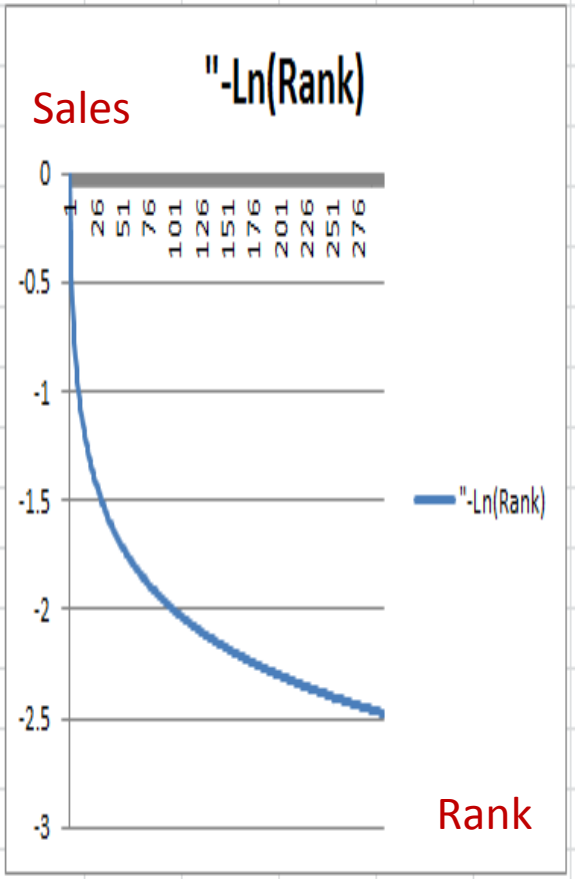
```
> summary(Apps[,c('Rank','Price','Screenshots','Rating_Score','Rating_Num')])
      Rank            Price          Screenshots     Rating_Score     Rating_Num
 Min.   :  1.0   Min.   : 0.990   Min.   :0.000   Min.   :0.000   Min.   :0.000000
 1st Qu.: 75.0   1st Qu.: 0.990   1st Qu.:5.000   1st Qu.:3.500   1st Qu.:0.000363
 Median :150.0   Median : 1.990   Median :5.000   Median :4.000   Median :0.001934
 Mean   :150.3   Mean   : 3.145   Mean   :4.656   Mean   :3.786   Mean   :0.030803
 3rd Qu.:225.0   3rd Qu.: 3.990   3rd Qu.:5.000   3rd Qu.:4.500   3rd Qu.:0.015088
 Max.   :300.0   Max.   :89.990   Max.   :5.000   Max.   :5.000   Max.   :1.000000
                                                                  NA's   :14
```

From Rankings to Sales

- Variable Transformation
  - How to measure an App's sales from ranking information?







Data Pre-processing

- **Variable Transformation**
  - Transform *Rank* to –ln(*Rank*)
    - Create a variable, *Sales,* storing –log(*Rank*)

```
> head(Apps)
      AppID Category Rank                                    Title Price Screenshots Rating_Score  Rating_Num     Sales
1 281656475    Games  212                                  PAC-MAN  4.99           5          4.0 0.085840153 -5.356586
2 283035434  Finance  168      SplashMoney - Personal Finance Manager  4.99           5          2.5 0.008092852 -5.123964
3 283281224  Finance  160                                   Budget  0.99           2          3.0 0.025109756 -5.075174
4 283494170  Finance   74 PocketMoney - checkbook, budgets, expenses  4.99           5          3.5 0.027792261 -4.304065
5 284428991 Business   30                                 Recorder  0.99           4          3.5 0.086059287 -3.401197
6 284650141  Finance  261                                     Tips  0.99           3          3.5 0.002130891 -5.564520
```

Data Summary

- Present the summary statistics of the variables
  - Before

```
> summary(Apps[,c('Rank','Price','Screenshots','Rating_Score','Rating_Num')])
      Rank           Price          Screenshots     Rating_Score     Rating_Num
 Min.   :  1.0   Min.   : 0.990   Min.   :0.000   Min.   :0.000   Min.   :     0.0
 1st Qu.: 75.0   1st Qu.: 0.990   1st Qu.:5.000   1st Qu.:3.500   1st Qu.:    50.2
 Median :150.0   Median : 1.990   Median :5.000   Median :4.000   Median :   276.5
 Mean   :150.3   Mean   : 3.145   Mean   :4.656   Mean   :3.786   Mean   :  8201.7
 3rd Qu.:225.0   3rd Qu.: 3.990   3rd Qu.:5.000   3rd Qu.:4.500   3rd Qu.:  2181.0
 Max.   :300.0   Max.   :89.990   Max.   :5.000   Max.   :5.000   Max.   :823547.0
                 NA's   :25
```

  - After

```
> summary(Apps[,c('Rank', 'Sales','Price','Screenshots','Rating_Score','Rating_Num')])
      Rank           Sales              Price          Screenshots     Rating_Score     Rating_Num
 Min.   :  1.0   Min.   :-5.704   Min.   : 0.990   Min.   :0.000   Min.   :0.000   Min.   :0.000000
 1st Qu.: 75.0   1st Qu.:-5.416   1st Qu.: 0.990   1st Qu.:5.000   1st Qu.:3.500   1st Qu.:0.000363
 Median :150.0   Median :-5.011   Median : 1.990   Median :5.000   Median :4.000   Median :0.001934
 Mean   :150.3   Mean   :-4.709   Mean   : 3.145   Mean   :4.656   Mean   :3.786   Mean   :0.030803
 3rd Qu.:225.0   3rd Qu.:-4.317   3rd Qu.: 3.990   3rd Qu.:5.000   3rd Qu.:4.500   3rd Qu.:0.015088
 Max.   :300.0   Max.   : 0.000   Max.   :89.990   Max.   :5.000   Max.   :5.000   Max.   :1.000000
                                                                                    NA's   :14
>
```

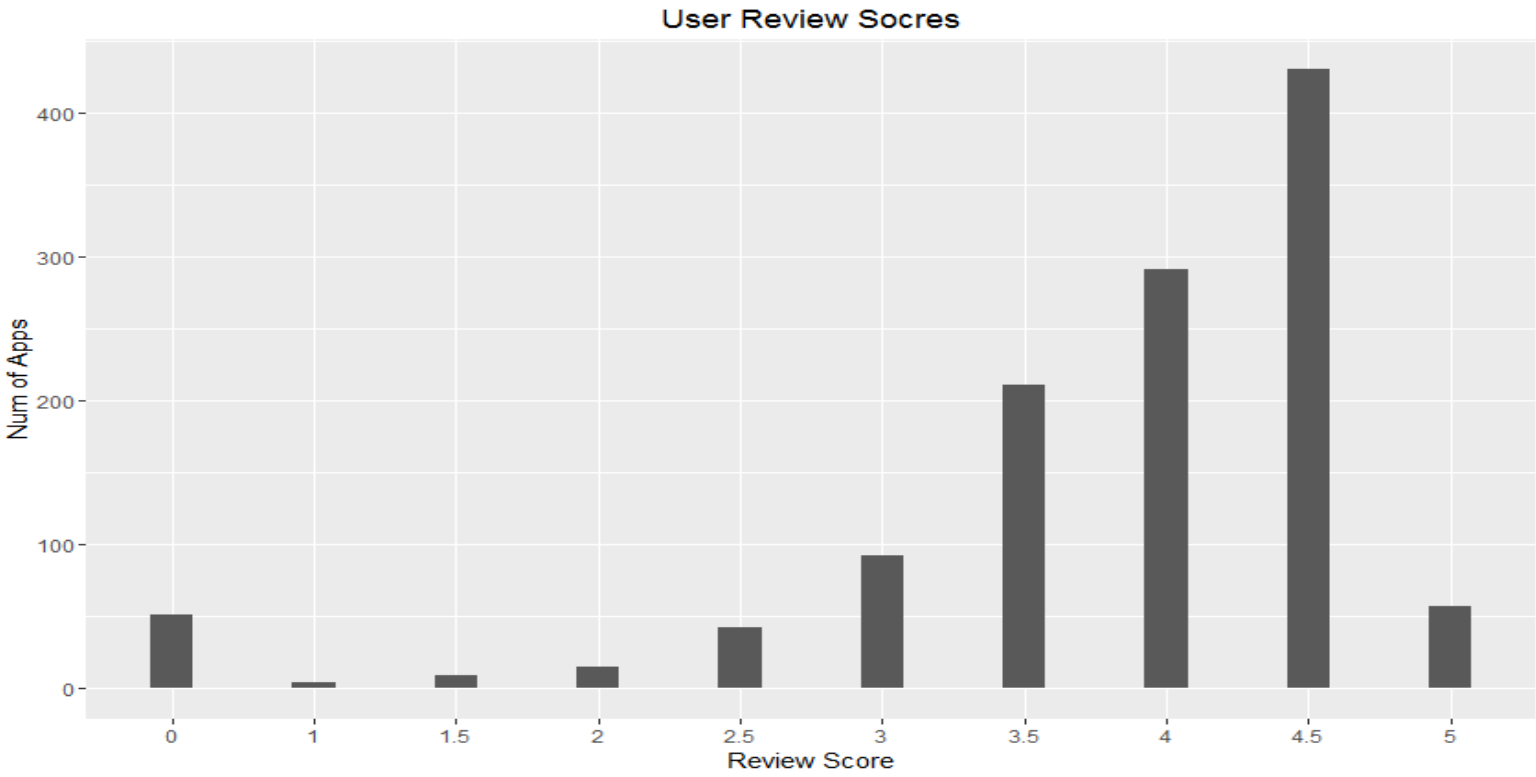
Data Summary

- Present a correlation table for the variables

```
> cor(Apps[,c('Rank', 'Sales','Price','Screenshots','Rating_Score','Rating_Num')], use="complete.obs")
                    Rank       Sales       Price  Screenshots Rating_Score  Rating_Num
Rank         1.000000000 -0.87564341  0.002042587 -0.124909018  -0.19276245 -0.11964210
Sales       -0.875643415  1.00000000 -0.018931930  0.105033013   0.18176958  0.12432577
Price        0.002042587 -0.01893193  1.000000000 -0.000770228  -0.08203855 -0.06084986
Screenshots -0.124909018  0.10503301 -0.000770228  1.000000000   0.19143140  0.04398633
Rating_Score -0.192762445  0.18176958 -0.082038554  0.191431401   1.00000000  0.14065733
Rating_Num  -0.119642096  0.12432577 -0.060849859  0.043986335   0.14065733  1.00000000
```

Visualization

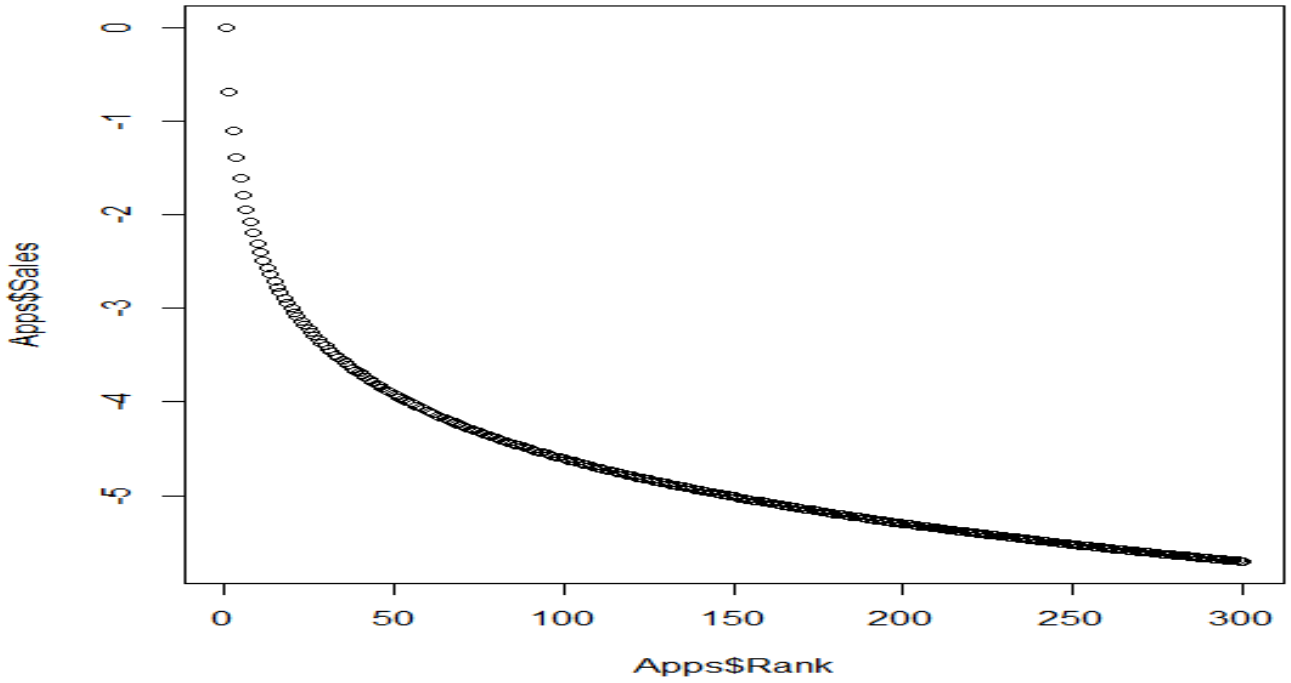
- **Histogram**
  - Generate a histogram for *Rating_Score* (using ggplot())



Visualization

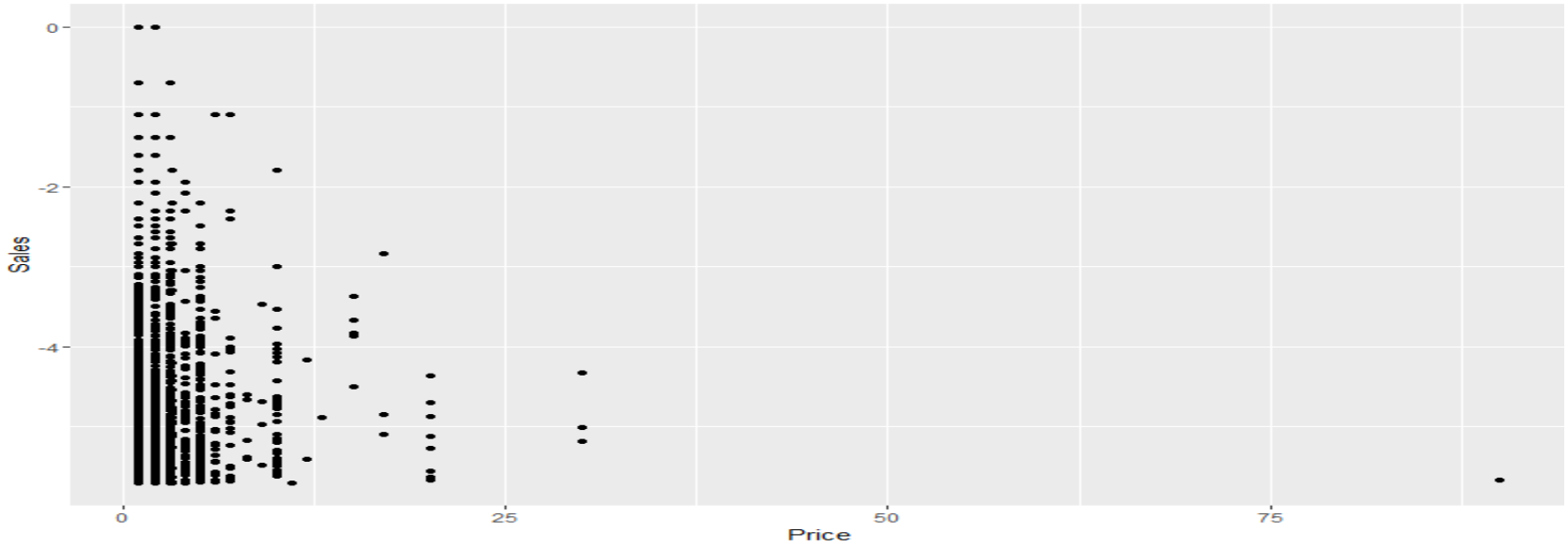
- **Scatter Plots**
  - **Ranking vs. Sales**

```
> plot(Apps$Rank, Apps$Sales)
```



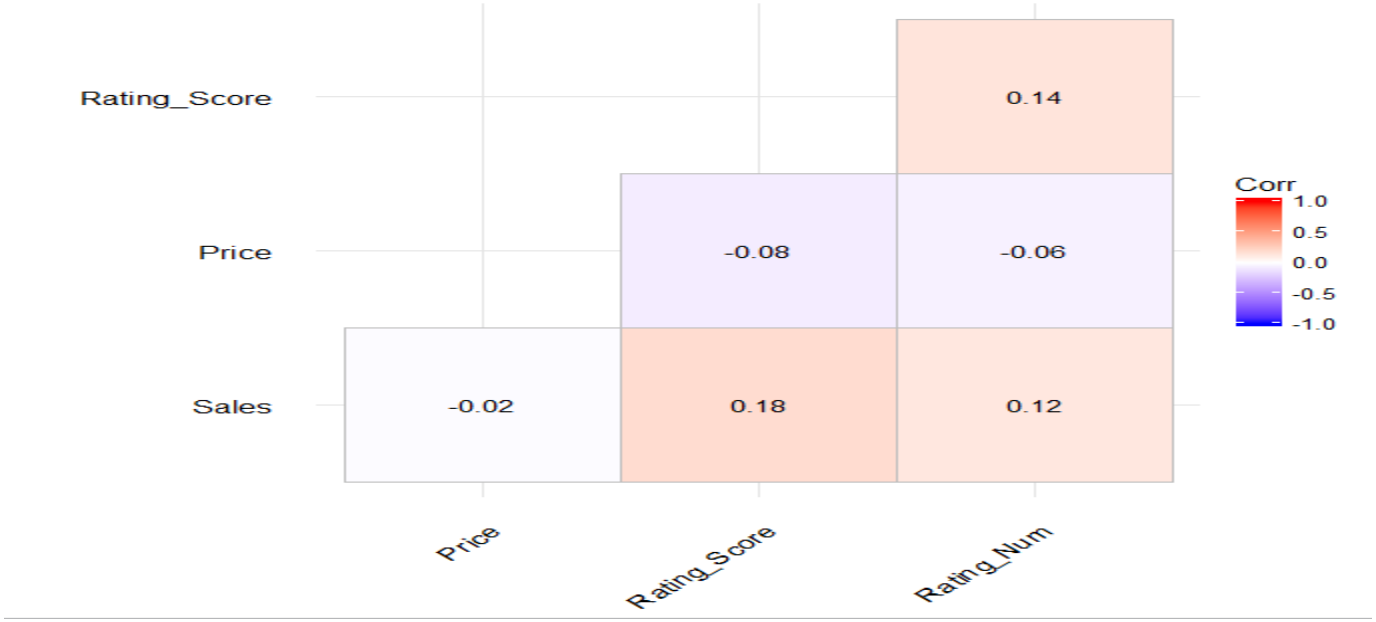
Visualization

- **Scatter Plots**
  - Generate a scatter plot presenting the relationship between *Sales* and *Price* (using ggplot())



Visualization

- **Matrix Plots**
  - Generate a matrix plot presenting the correlations among the variables: *Sales*, *Price*, *Ratings_Score*, and *Ratings_Num* (using ggcorrplot())



Data Export

Save the Apps data to the project folder for later use.

	A	B	C	D	E	F	G	H	I
1	AppID	Category	Rank	Title	Price	Screenshots	Rating_Score	Rating_Num	Sales
2	281656475	Games	212	PAC-MAN	4.99	5	4	0.085840153	-5.35659
3	283035434	Finance	168	SplashMo	4.99	5	2.5	0.008092852	-5.12396
4	283281224	Finance	160	Budget	0.99	2	3	0.025109756	-5.07517
5	283494170	Finance	74	PocketMo	4.99	5	3.5	0.027792261	-4.30407
6	284428991	Business	30	Recorder	0.99	4	3.5	0.086059287	-3.4012
7	284650141	Finance	261	Tips	0.99	3	3.5	0.002130891	-5.56452
8	284652710	Business	293	SpeakEasy	1.99	5	3.5	0.02102177	-5.68017
9	284815117	Games	83	SCRABBLE	1.99	5	4	0.731454824	-4.41884
10	284918921	Business	298	TravelTrac	2.99	5	3	0.008017289	-5.69709
11	284938955	Finance	292	iGold	1.99	4	4	0.000936988	-5.67675
12	284946069	Finance	93	Car Care -	4.99	5	3.5	0.009876151	-4.5326
13	284947174	Finance	20	iXpenseIt	4.99	5	3.5	0.065226426	-2.99573
14	284950726	Finance	286	Mini Mort	0.99	3	3.5	0.000400487	-5.65599
15	284974411	Finance	246	Day Bank -	1.99	5	3	0.009226305	-5.50533
16	284981670	Business	228	Mocha VN	5.99	5	3.5	0.012853354	-5.42935
17	285517831	Finance	247	TipCalc	0.99	4	3	0.004405353	-5.50939
18	285538312	Health	111	Ambiance	2.99	5	4	0.178737938	-4.70953
19	285538794	Health	157	Quit It 3.0	0.99	4	4	0.000680072	-5.05625
20	285539648	Finance	233	Tiptap	0.99	5	3.5	0.005969518	-5.45104
21	285750155	Business	41	iRecorder	2.99	5	3.5	0.012883579	-3.71357
22	285864916	Finance	226	Bowtie ??	0.99	2	4	0.016510628	-5.42053