Exercise: Facebook Statistics

Joe Schmo
ECO 230: Business and Economics Research and Communication
Instructor: James M. Murray, Ph.D.

March 22, 2018

Required Packages

The following code requires the packages in the `tidyverse` and the package, `psych`. The `tidyverse` actually contains many packages that allow you to organize, summarize, and plot data. The package `psych` is used to perform statistics related to the median. If you have not already done so, download and install the libraries (needed only once per computer) by running the following code in your console:

```
install.packages("psych") # This only needs to be executed once for your machine
```

```
install.packages("tidyverse") # This only needs to be executed once for your machine
```

First, let us load the libraries that we will use. What follows is an R-chunk. Normally, it execute the code, and show in your PDF document the output from the code, along with any messages and warnings. We suppress the warnings and messages with the optional parameters, `message=FALSE, warning=FALSE`.

```
library("tidyverse")
library("psych")
```

Data Set

The data set comes from the following study on Facebook marketing and performance metrics:

Moro, S., Rita, P. and Vala, B., (2016) "Predicting Social Media Performance Metrics and Evaluation of the Impact on Brand Building: A Data Mining Approach" *Journal of Business Research*, Vol. 68, pp. 3341-3351. Available at http://www.sciencedirect.com/science/article/pii/S0148296316000813

Download and load into memory the data set:

```
load(url("http://murraylax.org/datasets/facebook.RData"))
```

The data set includes statistics from 500 Facebook posts in 2014 related to the marketing of a globally known cosmetic brand. Facebook marketing is an important part of many businesses marketing strategy. Facebook interaction can help businesses build their brand and market new products. Marketing executives such statistics to better understand the effectiveness of their Facebook marketing.

The data set includes the following variables:

1. **Type**: Scale / Class: Nominal / Factor. Type of post. Possible outcomes are "Link", "Photo", "Status", and "Video"
2. **Month**: Scale / Class: Ordinal / Ordered factor. Month of the year for the post.
3. **Weekday**: Scale / Class: Ordinal / Ordered factor. Day of the week for the post.
4. **Hour**: Scale / Class: Ratio / Integer. Hour of the day - between 0 (12:00AM) and 23 (11:00PM)
5. **Paid**: Scale / Class: Binary / Integer. Dummy variable equal to 1 if a paid post, 0 if a free or unsolicited post.

6. **Reach**: Scale / Class: Ratio / Integer. Number of unique individuals who saw the post appear on their news feeds.
7. **Impressions**: Scale / Class: Ratio / Integer. Number of times the post appeared on people's news feeds (some individuals may have had the post appear more than once)
8. **EngagedUsers**: Scale / Class: Ratio / Integer. Number of unique individuals that clicked anywhere in the post.
9. **Comments**: Scale / Class: Ratio / Integer. Number of comments on the post.
10. **Likes**: Scale / Class: Ratio / Integer. Number of likes for the post
11. **Shares**: Scale / Class: Ratio / Integer. Number of shares for the post
12. **Interactions**: Scale / Class: Ratio / Integer. The sum, Comments + Likes + Shares.
13. **Weekday.Int**: Scale / Class: Ordinal / Integer: Number associated with day of the week in **Weekday**
14. **Month.Int**: Scale / Class: Ordinal / Integer: Number associated with month in **Month**

Exercises

1. Compute the mean and standard deviation for the number of engaged users per post. You can use the functions `mean()` and `sd()`

```
# Type your R-code here.
# RMarkdown will execute the code and show the code and output in your PDF file.
```

2. Compute the median and interpolated median for the number of comments per post.

```
# Type your R-code here.
# RMarkdown will execute the code and show the code and output in your PDF file.
```

3. Compute the frequencies of Facebook posts by day of the week. What two days have the most number of Facebook posts?

```
# Type your R-code here.
# RMarkdown will execute the code and show the code and output in your PDF file.
```

Saturday and Sunday have the most number of Facebook posts.

4. Compute the median and interpolated median for the number of comments by day of week. What day results in the most number of comments? What day of the week results in the least number of comments on average? Hint: Use `group_by()` and `summarize()`

```
# Type your R-code here.
# RMarkdown will execute the code and show the code and output in your PDF file.
```

In addition to using R to compute the necessary statistics, be sure to also type an answer the question!! Replace this text with a description of your answer.

5. Compute and interpret a 95% confidence interval for the mean number of shares of a Facebook post

```
# Type your R-code here.
# RMarkdown will execute the code and show the code and output in your PDF file.
```

In addition to using R to compute the necessary statistics, be sure to also type an answer the question!! Replace this text with a description of your answer.

6. Compute the mean and interpret a 95% confidence interval for the mean for the number of shares generated by a Facebook post created on a Monday

```
# Type your R-code here.
# RMarkdown will execute the code and show the code and output in your PDF file.
```

In addition to using R to compute the necessary statistics, be sure to also type an answer the question!! Replace this text with a description of your answer.

7. Report the median number of engaged users from paid posts versus unpaid posts. Based on these sample results, do paid posts result in more engaged users than unpaid posts?

```
# Type your R-code here.
# RMarkdown will execute the code and show the code and output in your PDF file.
```

In addition to using R to compute the necessary statistics, be sure to also type an answer the question!! Replace this text with a description of your answer.

8. Test the hypothesis that there is a difference in the median number of engaged users from paid posts versus unpaid posts. Type out explicitly each necessary step of the hypothesis testing procedure.

Replace this text with your answer. Include R-chucks as necessary.

9. Compare Facebook posts of photos versus videos. Is there statistical evidence that one results in more interactions than the other? Test the appropriate hypotheses.

To do this problem, it is first necessary to create a subset of the data, where `Type` is equal to only `"Photo"` or `"Video"`. You can verify with a call to `levels()` that `Type` currently takes on four possible values:

```
levels(df$Type)
```

```
## [1] "Link"   "Photo"  "Status" "Video"
```

We can use the `filter()` function, to select only rows where type is equal to "Photo" *OR* "Video". In the code below, the vertical pipe, `|`, is an *OR operator*.

```
testdata <- filter(df, Type=="Photo" | Type=="Video")
```

The data frame `testdata` now only contains the rows where `Type` is "Photo" or "Video". We need to do one more step, though, before we can compare the groups. If you call `levels()` again, you can see that R still thinks the other levels "Link" and "Status" are possible, even if they are not included in our new data frame. We can drop the levels that are not present in the data frame with a call to `droplevels()`

```
testdata <- droplevels(testdata)
```

I did the code above for you to prepare the data frame to conduct the hypothesis test, but you will need to do this on your own in the future.

Now, test the appropriate hypothesis. Type out explicitly each necessary step of the hypothesis testing procedure. Include text and R chunks as necessary.