report

Linear Regression - by Bryan Alcorn

Abstract

We will learn to combine our previous skills and make a simple report from a MakeFile.

Notes from the Professor So far we have been focused on introducing and learning the basic tools typically used in computational reproducible workflows (e.g. bash, git, github, Make, markdown, pandoc, and some text editor). However, we haven't done any statistical data analysis… yet.

The purpose of this assignment is to give you the opportunity to start applying the computational toolkit (plus R) to reproduce a simple regression analysis. More specifically, the idea is to reproduce the analysis from Section 3.1 (pages 59 to 71), of *Chapter 3. Linear Regression*, from the book "An Introduction to Statistical Learning" (by James et al):

http://www-bcf.usc.edu/~gareth/ISL/

The data set is in the `Advertising.csv` file available here:

http://www-bcf.usc.edu/~gareth/ISL/Advertising.csv

Introduction

We want to be able to replicate this result, possibly if the data changes or anthing else changes, we can call upon our MakeFile to change everything for us.

The analysis involves carrying out a simple linear regression of `TV` advertising on `Sales` of a particular product. The overall idea is to write a report in which you are able to replicate.

Data

Along with the description of the Advertising data set bellow by the professor, we also utilize an RData file with the regression results in it and another text file with the statistical summaries of the data.

From the Professor The Advertising data set consists of the Sales (in thousands of units) of a particular product in 200 different markets, along with advertising budgets (in thousands of dollars) for the product in each of those markets for three different media: TV, Radio, and Newspaper.

Methodology

Utilizing the function lm (linear model), we get the coefficients for a regression line. Using the method of least squares, we estimate coefficients for the equation y = mx + b

Sales = m(TV) + b

Results

```r
load("../data/regression.Rdata")
summary = summary(adModel)
summary
```

```
##
## Call:
## lm(formula = adData$Sales ~ adData$TV)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.3860 -1.9545 -0.1913  2.0671  7.2124
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 7.032594   0.457843   15.36   <2e-16 ***
## adData$TV   0.047537   0.002691   17.67   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.259 on 198 degrees of freedom
## Multiple R-squared:  0.6119, Adjusted R-squared:  0.6099
## F-statistic: 312.1 on 1 and 198 DF,  p-value: < 2.2e-16
```

The relationships seems to be corrolated. the R-Squared value is above .6 which indicates a relationship, but not a strongly corrolated one.

Bellow are some plots to illustrate the data provided in Advertising.csv

> The data has an interesting trend. The spread increases over time. As the TV variables increases, the data points become less concentrated around the sales data.

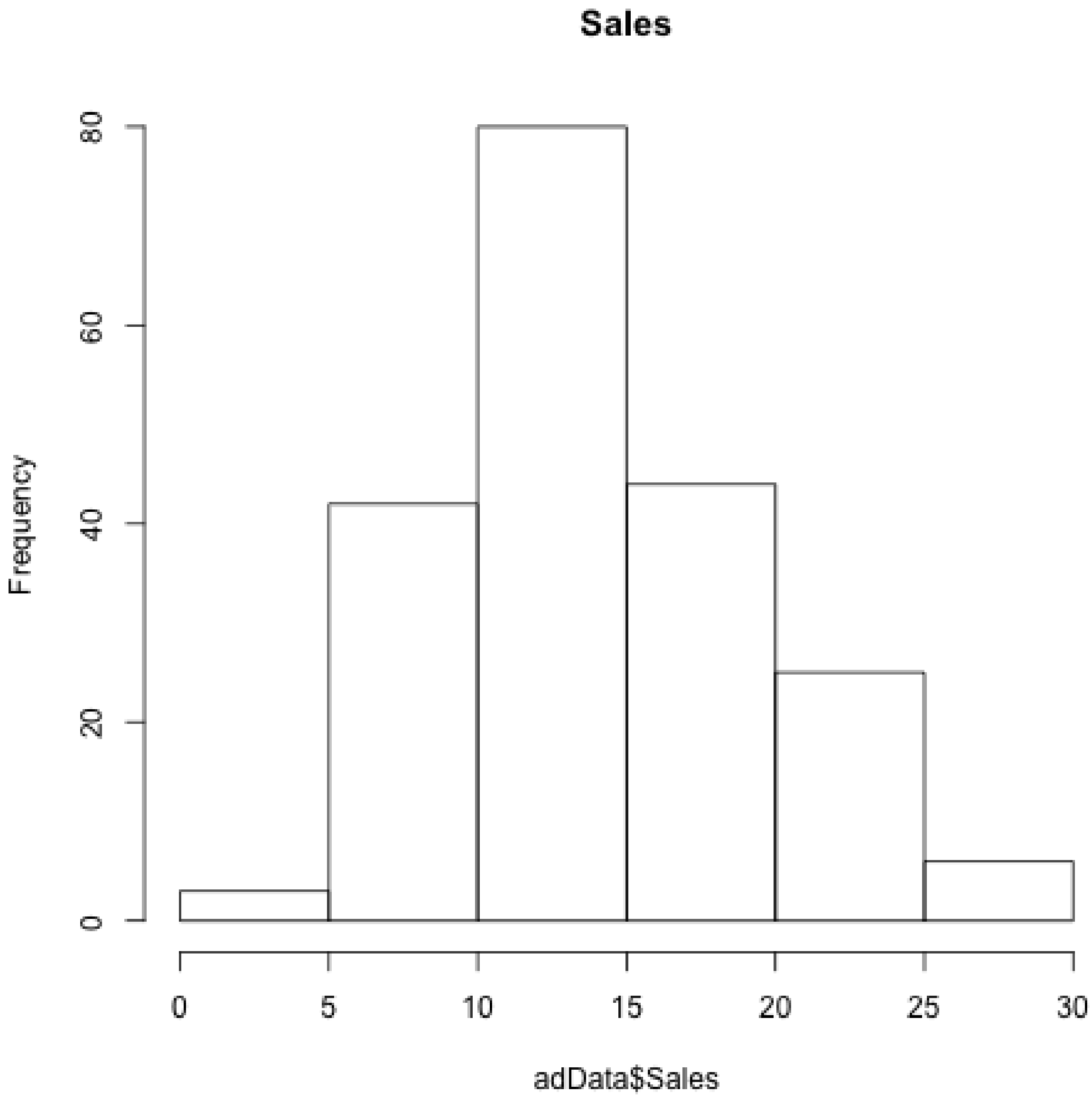

Figure 1: Histogram Sales

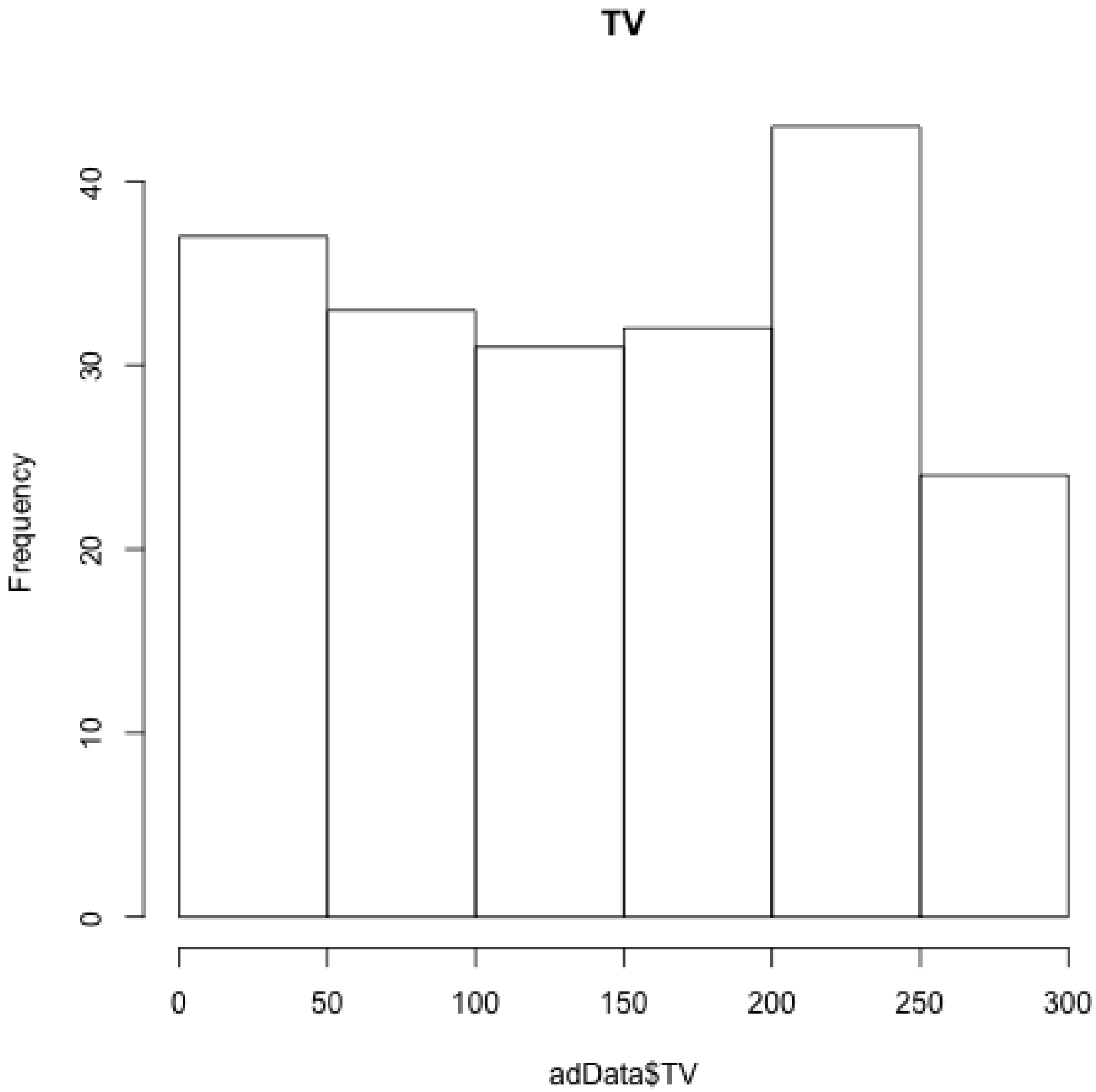

Figure 2: Histogram Data

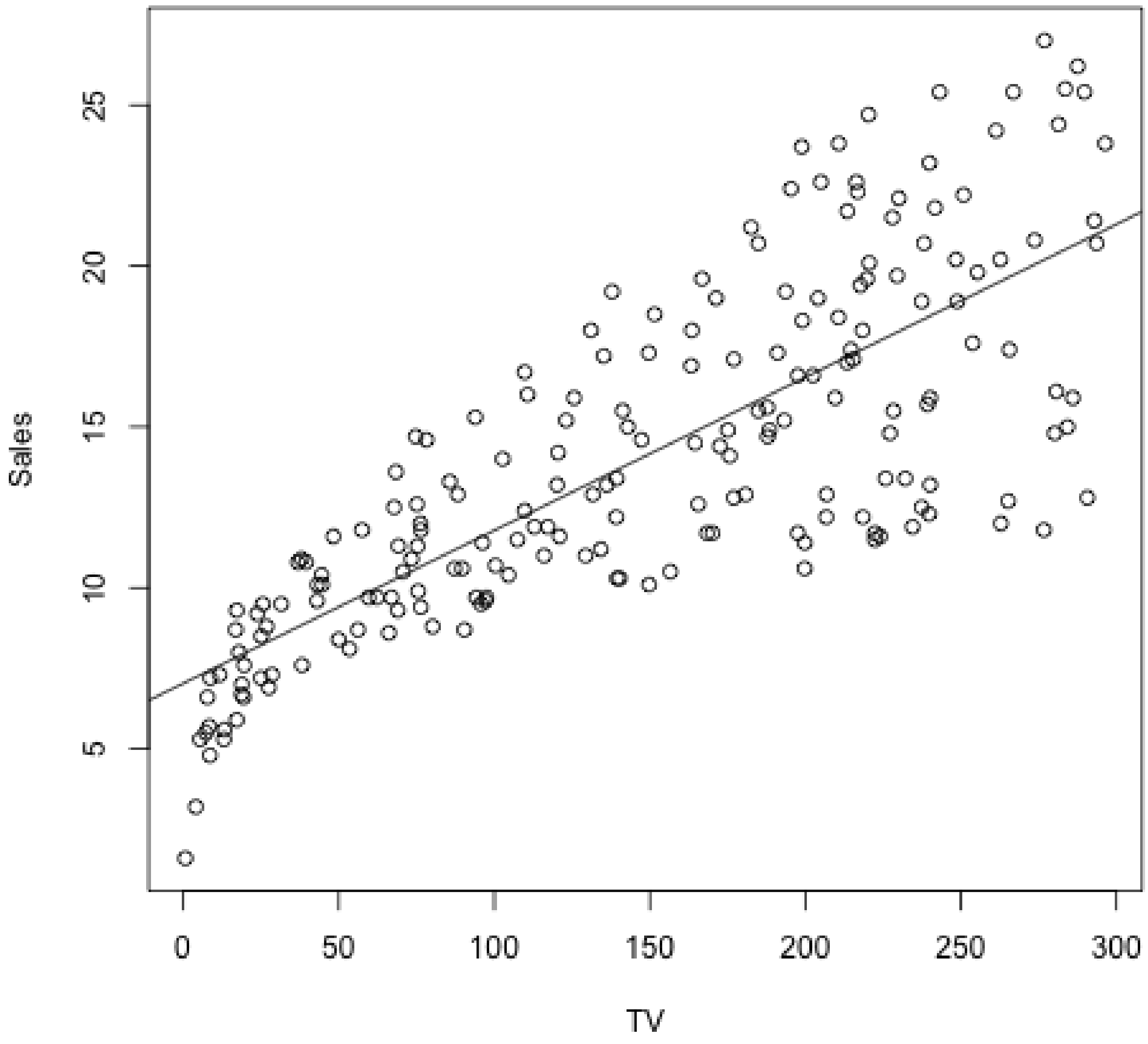

Figure 3: Scatterplot TV vs. Sales

Conclusions

To summarize, in this proejct I automated my workflow to gather data and produce the images and the data for a report. Then the RMD file synthesizes that information and uses its dynamic capabilities to include the images and code in the report.