Simple Linear Regression

Nura Kawa

October 7, 2016

Abstract

In this report we reproduce the main results displayed in section 3.1 "Simple Linear Regression" in the book *An Introduction to Statistical Learning*.

Introduction

Our goal is to provide advice on how to improve sales of the particular product. To do this, we look at the impact of advertising on sales and model the association between them. We then develop a simple linear regression model to predict sales on the basis of advertising via three media.

Data

The Advertising dataset consists of *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*.

Below is the first ten rows of the Advertising Dataset:

```
{r dataset, echo=F} envpath <- "C:/Users/Nura/Desktop/Fall 2016/Stat
159/stat159-fall2016-hw02/stat159-fall2016-hw02/" setwd(envpath)
Advertising <- read.csv(paste0(envpath, "data/Advertising.csv"))
Advertising <- Advertising[,-1] head(Advertising, 10)
```

The *Sales* variable reveals the following distribution:

Looking specifically at the *TV* advertising budget, we see this distribution over the 200 markets:

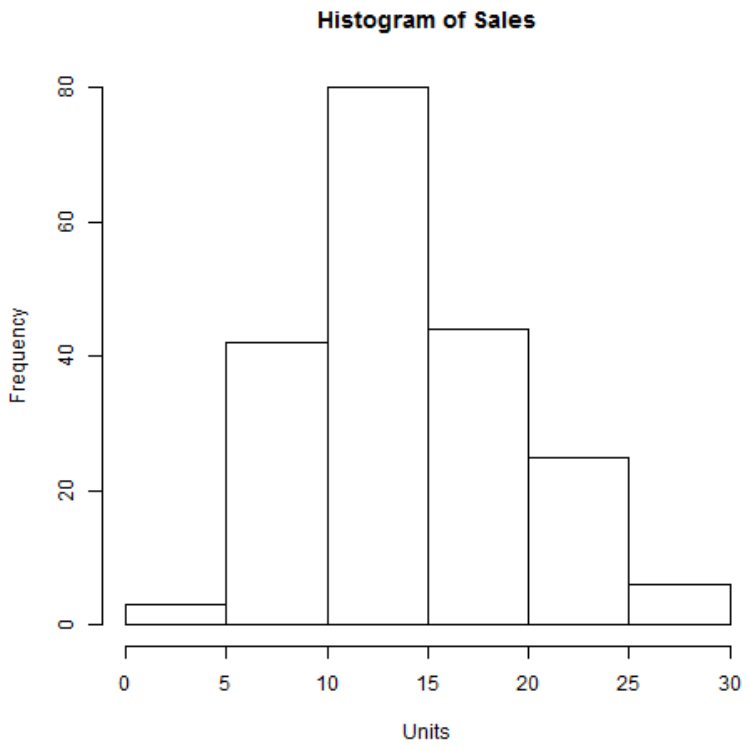


Figure 1: Histogram of Sales

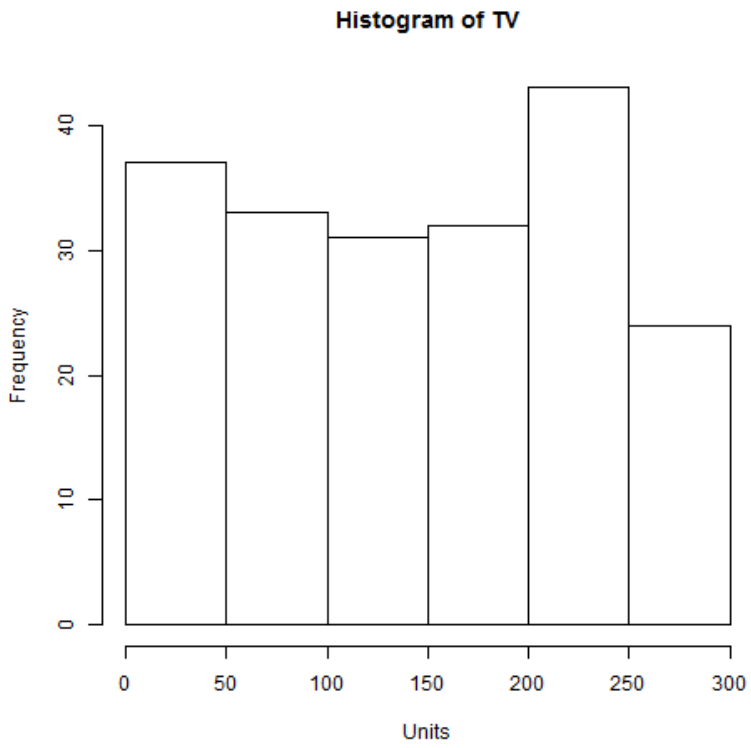


Figure 2: Histogram of TV Budget

Methodology

We use Simple Linear Regression to model the association between **Sales** and **TV**. Our method predicts **Sales** using **TV**, using the following linear model:

$$Y = \beta_0 + \beta_1 X$$

Here, our **Y** is **Sales** and our **X** is **TV**.

The parameters β_0 and β_1 are respectively the intercept and slope of our linear model (also called a regression line) fitted to our data.

More about the Linear Model

There exist several regression coefficients that allow us to asses at the accuracy of our linear model. Specifically, these coefficients give us an idea of how well our particular model allows us to predict **Sales** from **TV**.

Residual Standard Error (RSE)

The RSE is an estimate of the standard deviation of the error in our model. This shows how far our data will deviate from the generated regression line. The equation for RSE from page 69 of *Introduction to Statistical Learning* is:

double subscript error replace

where (y_i) is a data point of our response variable **Sales** *as predicted by our model*, and $(yhat_i)$ is our actual data point. Our goal always is to generate a model with a minimal RSE.

R-Squared

The R-Squared measures the goodness of fit of our model. It measures the proportion of variance explained by our model. Thus it has the range [0,1] and is independent of Y's scale.

$$R^2 = (TSS - RSS)/TSS$$

Where TSS measures the total sum of squares; the total variance inherent in the response before performing our regression. RSS meausres the amount of variability explained by our regression. Thus the above formula measures the proportion of variability in Y that is explained using X.

R Squared near 1: A large proportion of variability in the response explained by the regression; i.e. our model fits the data well. R Squared near 0: The regression did not explain much variability in the response; i.e. our regressin performed poorly. Although R Squared has the advantage of being more interpretable than RSE (due to it's restriction between 0 and 1), it is still problem dependent.

Results

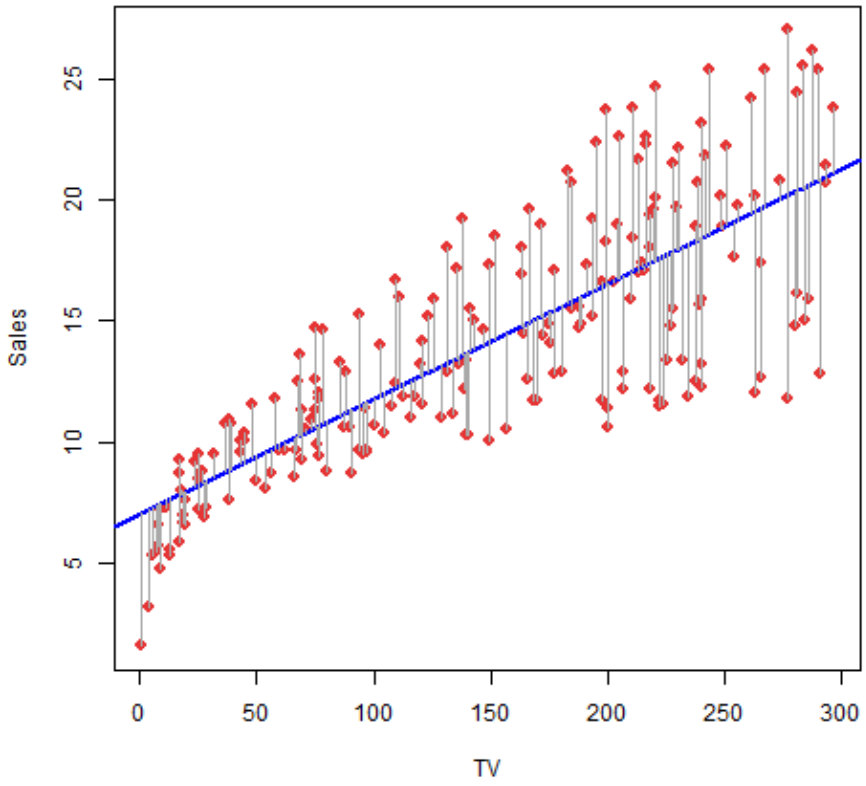
We compute the regression coefficients:

"'{r produce table, eval = T, warning = F, message=F, results="asis", echo=F} load(paste0(envpath, "/data/regression.RData"))

R_square <- regression_summary$r.squared$F_stat <- regression_summary$fstatistic[1]; names(F_stat) = NULL RSE <- 1234 "'

More information about the least-squares model is given below:

```
{r regression output xtable, results="asis", message=F, warning=F,
echo=F} library(xtable) reg_table <- xtable(regression_object)
print(reg_table, type="latex", comment = F)
```



{width = “400px”}

Conclusions