Lab 6: Simple linear regression in R

Prof. Adam Loy

Due: Friday, November 4, 2016 by 4 p.m. on Moodle

0. Intro

Welcome to Lab 6!

Today's lab will explore how to fit simple linear regression models in R. In addition, it will explore how we can use the bootstrap to create confidence intervals for the y-intercept and slope, and how we can create prediction intervals for a new observation.

Setup

You can download the .Rmd file for this lab and the data from the course webpage as a zip file. If you are using a Mac, then you will need to use a browser other than Safari for the download.

If you are using the R Studio server, then you can upload the entire zip folder directly onto the server.

We will use the following packages during this lab. Make sure that you have downloaded all of them before running the commands.

```r
library(ggplot2)
library(mosaic)
library(car)
```

1. The data

How much is an extra square foot of space worth? In this lab, we investigate this question for homes in Saratoga County, New York. The data set contains the prices of 1,728 homes. To begin we must load the data set.

```r
saratoga <- read.csv("data/saratoga.csv")
```

1.1. Exploring the data

Before jumping into a regression analysis, it's a good idea to plot the variables involved to determine whether there is linear relationship between them.

Question 1. Create a histogram of the price of the homes and describe the distribution.

Question 2. Create a histogram of the living ares of the homes and describe the distribution.

Question 3. Create a scatterplot of price vs. living area and describe the relationship. (Look at the column names carefully!)

In addition to exploratory plots, we can quantify the strength and direction of the linear association between the price and living area using the correlation. To calculate the correlation in R we use the `cor` command from `mosaic` package. The basic syntax is `cor(y ~ x, data = data_frame)` where `y` is the response variable and `x` is the explanatory variable. For our data set this becomes

```r
cor(Price ~ Living.Area, data = saratoga)
```

```
## [1] 0.7123902
```

2. Fitting a simple linear regression model in R

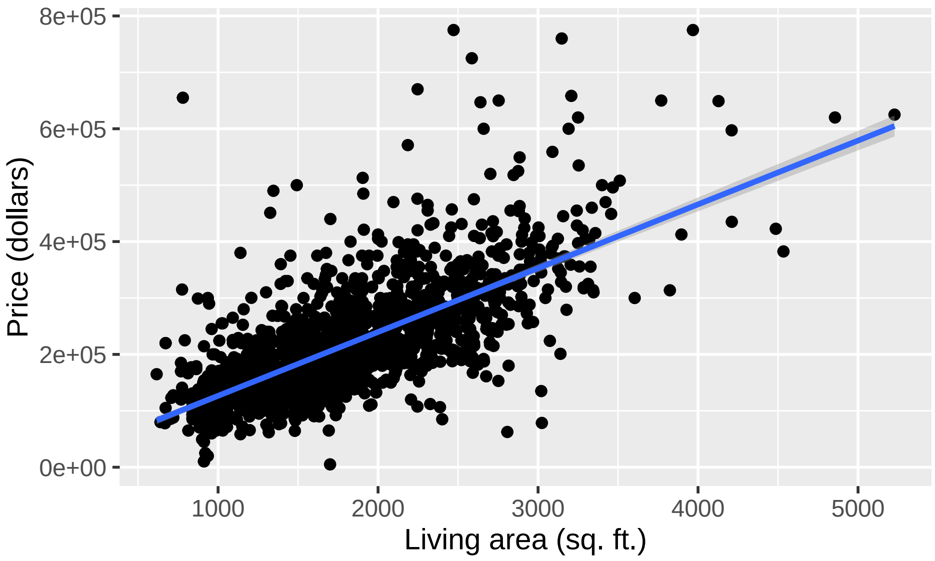
Having verified that a linear relationship is plausible between price and living area, we fit a regression model to further explore this relationship. In R, we use the `lm` function to fit a simple linear regression. The basic syntax follows the same pattern as before: `lm(y ~ x, data = data_frame)` where `y` is the response variable and `x` is the explanatory variable. For our data set this becomes

```
mod <- lm(Price ~ Living.Area, data = saratoga)
summary(mod)
```

```
##
## Call:
## lm(formula = Price ~ Living.Area, data = saratoga)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -277022  -39371   -7726   28350  553325
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 13439.394   4992.353   2.692  0.00717 **
## Living.Area   113.123      2.682  42.173  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 69100 on 1726 degrees of freedom
## Multiple R-squared:  0.5075, Adjusted R-squared:  0.5072
## F-statistic:  1779 on 1 and 1726 DF,  p-value: < 2.2e-16
```

You can also easily overlay the fitted regression line on a scatterplot by adding a layer:

```
ggplot(data = saratoga, mapping = aes(x = Living.Area, y = Price)) +
  geom_point() +
  geom_smooth(method = "lm") +
  labs(x = "Living area (sq. ft.)", y = "Price (dollars)")
```



Question 4. Report the fitted least squares regression equation.

Question 5. Interpret the estimate of the slope within the context of the problem.

Question 6. Interpret the estimate of the y-intercept within the context of the problem. Does this interpretation make sense in this context?

3. Checking model fit

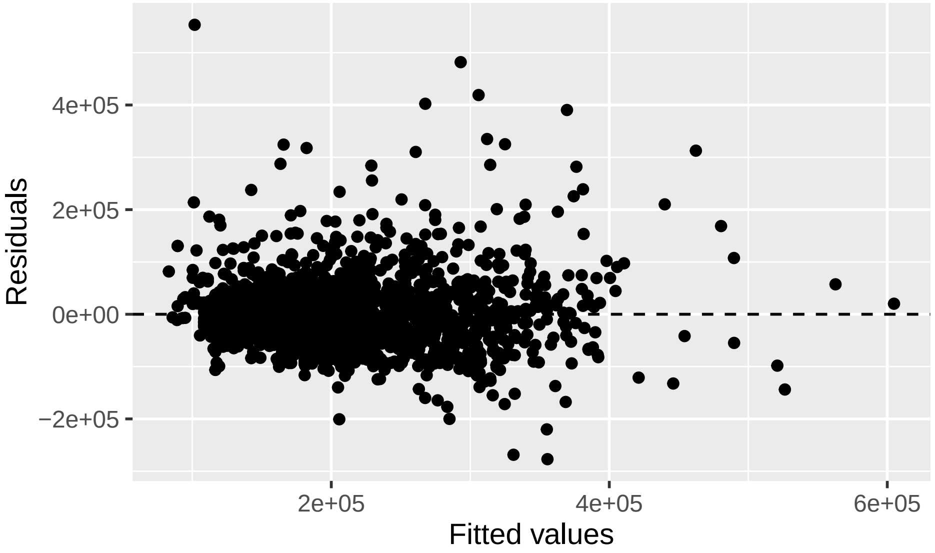
After fitting any regression model it is a good idea to double-check that our linear model fits the data appropriately. This can be accomplished by examining plots of the residuals to check for (1) linearity, (2) constant variability, and (3) nearly normal residuals.

3.1 Linearity

While we have already checked linearity using a plot of the response vs. explanatory variables, it should be double-checked using a residual plot (a plot of the residuals vs. the fitted values). The residual plot for a "good" regression model will look like a formless cloud with no pattern, and all of the residuals will be centered around 0. Nonlinear patterns in a residual plot are indicative of a nonlinear relationship between the explanatory and response variable.

A residual plot for our regression model is created below. Notice that we are able to use our model, `mod`, as our data set.

```r
ggplot(data = mod, mapping = aes(x = .fitted, y = .resid)) +
  geom_hline(yintercept = 0, linetype = "dashed") +
  geom_point() +
  labs(x = "Fitted values", y = "Residuals")
```



Question 7. Is there any apparent pattern in the residual plot? What does this indicate about the linearity of the relationship between price and living area?

Constant variability

We want the relationship between the response and explanatory variable to be linear, but we also want the spread of the observations around the least squares regression line to be approximately constant. This assumption will be important later when we discuss how to conduct inference for regression using the t distribution, but it's a good idea to get used to checking this assumption now.

Question 8. Based on the residual plot, does the constant variability condition appear to be met?

Bell-shaped residual distribution

Finally, we want the residuals to be symmetrically distributed about 0. While this can be checked from the above residual plot, it is often useful to create a histogram of the residuals to double-check this.

Question 9. Create a histogram of the residuals. (Remember to use `mod` as your data frame and `.resid` as your variable.) Is the distribution of the residuals approximately bell shaped?

4. Bootstrapping regression models

Up to this point we have simply described what our simple linear regression model reveals about the 1,728 houses contained in our sample; however, if we drew a new sample from the population it may give a different fitted least squares regression equation. In order to draw inferences about this linear relationship back to the population of all houses in Saratoga County, New York we must either construct confidence intervals for the slope and intercept (also called regression coefficients), or we must run hypothesis tests. In this lab we will focus on constructing confidence intervals for regression coefficients using the bootstrap. Recall that confidence intervals can be used to test hypotheses as long as we keep track of how confidence and significance levels are related.

In order to bootstrap a regression model we must take the following steps (you should understand these steps, but we will not manually implement them). Assuming that there are n rows in our data set

Step 1. Draw a random sample with replacement of size n from the rows of the data set.

Step 2. Fit the regression model to this new data set and save the values of the estimated coefficients or other summary statistics.

Step 3. Repeat steps 1 and 2 R times.

Then we build confidence intervals either using the plug-in or percentile method.

To bootstrap regression models in R, we use the `Boot` function found in the `car` package. The basic syntax is `Boot(model, R)` where `model` is the name of the original regression model fit to our data and `R` is the number of bootstrap samples to collect. To bootstrap our regression model we run the following code:

```r
mod_boot <- Boot(mod, R = 999)
summary(mod_boot)
```

```
##               R original   bootBias    bootSE  bootMed
## (Intercept) 999 13439.39 108.986261 5707.3947 13628.59
## Living.Area 999   113.12  -0.088441    3.4846   112.96
```

Question 10. Create a 95% plug-in confidence interval for the slope.

Question 11. Interpret the interval you just created in the context of the problem.

Question 12. Based on your interval, is there statistically significant evidence that the slope is not 0 (i.e. that there is some linear relationship between the variables)? If so, at what significance level?

5. Wrapping Up

Congratulations, you just just bootstrapped your first regression model in R! Be sure to review the steps and logic behind the bootstrap before next class.