STAT 215: Lab 5

Simple Linear Regression

Last Revised October 6, 2017

Note: This lab is a modification by Colin Dawson from source material by Andrew Bray, Mine Çetinkaya-Rundel, and the UCLA statistics department which accompanies the OpenIntro statistics textbooks. This handout as well as the source material is covered by a CreativeCommons Attribution-ShareAlike 3.0 Unported license.

Lab Summary The movie Moneyball focuses on the "quest for the secret of success in baseball". It follows a low-budget team, the Oakland Athletics, who believed that underused statistics, such as a player's ability to get on base, better predict the ability to score runs than typical statistics like home runs, RBIs (runs batted in), and batting average. Obtaining players who excelled in these underused statistics turned out to be much more affordable for the team.

The goal of this lab is to explore various simple linear regression models to predict the number of runs scored by baseball teams in a season, using a variety of common team level measures of a team's offensive performance.

What to Turn In You only need to turn in written answers to the questions at the end, in the section titled "Homework". You are encouraged, but not required, to use Markdown to prepare your writeup.

The Data

We will use a dataset from the 2011 Major League Baseball season. In addition to runs scored (`Runs`), there are seven traditionally used variables in the data set: `AtBats`, `Hits`, `HomeRuns`, `BattingAvg`, `Strikeouts`, `StolenBases`, and `Wins`. There are also three newer variables: on base percentage (OBP), slugging percentage (SLG), and on-base plus slugging (OPS). For the first portion of the analysis we'll consider the seven traditional variables. At the end of the lab, you'll work with the newer variables on your own.

The data is located at http://colinreimerdawson.com/data/mlb11.csv Load the mosaic package and read the data in with read.file() as MLB11:

```r
library("mosaic")
MLB11 <- read.file("http://colinreimerdawson.com/data/mlb11.csv")
```

If you are using Markdown you should also run the lines above in the console, as we will be using some interactive graphs with this data that will not work in a Markdown document.

Exercise 1 What type of plot would you use to display the relationship between Runs and one of the other quantitative variables? Plot this relationship using the variable AtBats as the predictor. Looking at your plot, describe the relationship between these two variables. Make sure to discuss the form, direction, and strength of the relationship as well as any unusual observations. Does the relationship look linear? If you knew a team's AtBats, would you be comfortable using a linear model to predict the number of runs? If the relationship does look linear, quantify the strength of the relationship by computing the correlation coefficient.

Finding the Best-Fit Line

Type the following **at the console** to produce an interactive plot that will let you draw your own regression line and then will show you the residuals associated with it.

```r
library("oilabs")
plot_ss(y = Runs, x = AtBats, data = MLB11)
```

You may need to expand the plot window in the lower right of RStudio to see the whole graph.

After running this command, you'll be prompted to click two locations anywhere in the plotting plane to define a line. Once you've done that, the line you specified will be shown in black and the residuals in blue. Note that there are 30 residuals, one for each of the 30 observations. Recall that the residuals are the difference between the observed values and the values predicted by the line.

The most common way to do linear regression is to select the line that minimizes the sum of squared residuals. To visualize the squared residuals, you can rerun the plot command and add the argument `showSquares = TRUE`.

```
plot_ss(y = Runs, x = AtBats, data = MLB11, showSquares = TRUE)
```

Note that the sum of squared residuals is displayed in the console when you choose your line.

Exercise 2 Re-run the line above a few times (at the console), selecting different lines, and see how small you can get the sum of squared residuals (SSR) to be. Write down the prediction equation for your best line (in the form $\hat{y} = \hat{a} + \hat{b}x$).

It is rather cumbersome to try to get the correct least squares line, i.e. the line that minimizes the sum of squared residuals, through trial and error. As we discussed in class, the best line is the solution to a multivariable calculus problem; but we can use the `lm()` function in R to fit the linear model (a.k.a. regression line).

We will want to use the resulting regression model later, so we'll save the result to a named R object.

```
AtBatsModel <- lm(Runs ~ AtBats, data = MLB11)
```

You can display the model coefficients (that is, intercept and slope) by calling the `coef()` function on the model

```
coef(AtBatsModel)
```

Exercise 3 Write down the prediction equation for the best fit line found by `lm()`. Did you get close with your trial-and-error approach? Plot the best fit line over the data using `xyplot()` as we have done before (using `type = c("p","r")`) to visualize it. If a team manager saw the least squares regression line and not the actual data, how many runs would they predict for a team with 5,578 at-bats?

Exercise 4 Fit a new model that uses `HomeRuns` to predict `Runs`. Using the estimates from the R output, write the equation of the regression line. What does the slope tell us in the context of the relationship between success of a team and its home runs?

Measuring fit with R^2

Definition: R^2 One measure of how well the model fits the data is the sum of the squared residuals. Another is called the **coefficient of determination**, denoted by R^2. This is a number that ranges from 0 to 1 which indicates what proportion of the total variability in the response variable is linearly related to the explanatory variable. The proportion of variability that is *not* linearly related; that is the "random" part that's not explained by the model, is represented by the ratio between the variance of the residuals and the variance of the response variable by itself, and R^2 is 1 minus this proportion:

$$R^2 = 1 - s^2_{residuals}/s^2_y$$

It turns out that it can also be computed by squaring the correlation coefficient.

The R^2 value of a model can be found as follows:

```
rsquared(AtBatsModel)
```

Verify that you get the same thing if you square the correlation coefficient:

```
r <- cor(Runs ~ AtBats, data = MLB11)
r^2
```

and that we get the same thing if we look at ratios of variances:

```
s2.residuals <- residuals(AtBatsModel) %>% var()
s2.y <- var(~Runs, data = MLB11)
1 - s2.residuals / s2.y
```

Note: It is possible you will get a cryptic warning when you use the `var()`

function as above. You can ignore this; it is caused by an odd interaction between packages, but it doesn't hurt anything.

Exercise 5 Compare the R^2 value for the `AtBats` model to the R^2 value for the `HomeRuns` model. Which explanatory variable does a better job of accounting for the number of runs scored?

Assessing Model Quality

As we have seen, not every linear model is appropriate, even if the residuals are small. We should check at least two things:

- **Linearity:** Is there a "leftover" pattern in the residuals which is associated with the explanatory variable or with the predicted values? If so, the relationship is likely not linear.
- **Approximate Normality:** Are the residuals approximately bell-shaped (Normally distributed)? If not, the best fit line may not be reliable, due to skew or outliers.

To check for linearity, we should plot the residuals against the explanatory or fitted values:

```
## Plotting residuals against explanatory variable
xyplot(residuals(AtBatsModel) ~ AtBats, data = MLB11, type = c("p", "r"))

## Plotting residuals against fitted (i.e., predicted response) values
xyplot(residuals(AtBatsModel) ~ fitted.values(AtBatsModel), type = c("p", "r"))
## Note that we can omit the data= argument in the second case, since
## everything we need is stored with the model.
```

Do you see any pattern?

To check for Normality (bell-shapedness) we can create a histogram of the residuals, with an overlaid Normal curve:

```
## We can again omit data= since we are using only the residuals
histogram(~residuals(AtBatsModel), fit = "normal")
```

An alternative plot we can use to assess Normality is called a `Quantile-Quantile` plot (or QQ Plot). It plots the quantiles of a theoretical Normal distribution against the actual quantiles of the residuals. If the fit is normal, the residuals should fall on a straight line. If the values are very curved or form an S-shape, that is a sign that the residual distribution is skewed, or has values that are more extreme than expected.

```
## QQ Plot
plot(AtBatsModel, which = 2)
```

It is notoriously difficult to get one's head around the precise meaning of the axes in a QQ Plot, and in practice we do not really need to: it suffices to be able to interpret a few standard shapes. You can play with QQ plots and see the shapes for different residual distributions here: `https://xiongge.shinyapps.io/QQplots/`.

QQ Plots: The Nitty Gritty Details If you are interested, here is what is really happening: We are taking our set of residuals, and turning each one into a quantile (think percentile), so if we have 10 residuals, the smallest (i.e., largest negative) value is the 10th percentile, the next smallest is the 20th percentile, etc. On the x-axis we are turning those percentiles into values in a Standard Normal (e.g., the 2.5th percentile is at about -2 in a Standard Normal). On the y-axis we are plotting the standardized residual (i.e., the z-score of the residual) at that percentile, by taking the raw residual, subtracting the mean residual, and dividing by the standard deviation of the residuals.

If the residuals are perfectly Normal, the 2.5th percentile will have a z-score of -2, the 16th percentile will have a z-score of -1, the 50th percentile will be equal to the mean (z-score $= 0$), etc., *just like in the Normal*, and so the residuals will fall on a perfect diagonal, with the z-score of -2 in the actual distribution lining up with -2 in the standard Normal, etc. But if the distribution is skewed, or has "fat tails" (or "light tails"), we will see curvature. For example, if the distribution is right-skewed, the lowest percentiles will be closer to the center than they should be for a Normal — that is, the z-scores will be closer to zero than they should be, and the points in the QQ plot will be above the line. For the highest percentiles, the residuals will be *farther* from center than expected, so the z-scores will be more positive, and again the points will fall above the line. Overall we will have an upward curve. Left-skew similarly results in a downward curve. If *both* tails have more extreme values than expected, then the left-hand points will be below the line and the right-hand points will be above, producing a sort of vertically stretched-out N-shape (think a tan function if you are a trig buff). “Light” tails yield an S shape; etc.

Exercise 6 Produce residual diagnostic plots for the `HomeRuns` model you created above. Does the fit look roughly linear? Are the residuals roughly Normal?

1 Homework

1. Choose another traditional variable from `MLB11` that you think might be a good predictor of `Runs`. Produce a scatterplot of the two variables and fit a linear model. At a glance, does there seem to be a linear relationship?
2. How does this relationship compare to the relationship between `Runs` and `AtBats`? Use the R^2 values from the two model summaries to compare. Does your variable seem to predict `Runs` better than `AtBats`? How can you tell?
3. Now that you can summarize the linear relationship between two variables, investigate the relationships between `Runs` and each of the other four traditional variables: `Hits`, `BattingAvg`, `Strikeouts`, and `StolenBases`. Which variable best predicts `Runs`? Support your conclusion using the graphical and numerical methods we’ve discussed (for the sake of conciseness, only include output for the best variable, not all five).
4. Now examine the three newer variables: on-base percentage (`OBP`), slugging percentage (`SLG`) and on-base-plus-slugging (`OPS`). These are the statistics used by the author of *Moneyball* to predict a teams success. In general, are they more or less effective at predicting runs than the old variables? Explain using appropriate graphical and numerical evidence. Of all ten variables you’ve analyzed, which seems to be the best predictor of runs? Does the model using that variable satisfy the conditions of linearity and near-normality?