Lab 9: Fitting Models to Data

Statistical Computing, 36-350

Week of Monday October 22, 2018

Name:
Andrew ID:
Collaborated with:

This lab is to be done in class (completed outside of class if need be). You can collaborate with your classmates, but you must identify their names above, and you must submit **your own** lab as an knitted HTML file on Canvas, by Sunday 11:59pm, this week.

This week's agenda: exploratory data analysis, cleaning data, fitting linear/logistic models, and using associated utility functions.

Prostate cancer data

Recall the data set on 97 men who have prostate cancer (from the book The Elements of Statistical Learning). Reading it into our R session:

```
pros.df =
  read.table("http://www.stat.cmu.edu/~ryantibs/statcomp/data/pros.dat")
dim(pros.df)
```

```
## [1] 97  9
```

```
head(pros.df, 3)
```

```
##       lcavol  lweight age      lbph svi       lcp gleason pgg45       lpsa
## 1 -0.5798185 2.769459  50 -1.386294   0 -1.386294       6     0 -0.4307829
## 2 -0.9942523 3.319626  58 -1.386294   0 -1.386294       6     0 -0.1625189
## 3 -0.5108256 2.691243  74 -1.386294   0 -1.386294       7    20 -0.1625189
```

Simple exploration and linear modeling

- **1a.** Define `pros.df.subset` to be the subset of observations (rows) of the prostate data set such the `lcp` measurement is greater than the minimum value (the minimum value happens to be `log(0.25)`, but you should not hardcode this value and should work it out from the data). As in lecture, plot histograms of all of the variables in `pros.df.subset`. Comment on any differences you see between these distributions and the ones in lecture.
- **1b.** Also as in lecture, compute and display correlations between all pairs of variables in `pros.df.subset`. Report the two highest correlations between pairs of (distinct) variables, and also report the names of the associated variables. Are these different from answers that were computed on the full data set?
- **Challenge.** Produce a heatmap of the correlation matrix (which contains correlations of all pairs of variables) of `pros.df.subset`. For this heatmap, use the full matrix (not just its upper triangular part). Makes sure your heatmap is displayed in a sensible way and that it's clear what the variables are in the plot. For full points, create your heatmap using base R graphics (hint: the `clockwise90()` function from the "Plotting tools" lecture will be handy); for partial points, use an R package.

- **1c.** Compute, using `lm()`, a linear regression model of `lpsa` (log PSA score) on `lcavol` (log cancer volume). Do this twice: once with the full data set, `pros.df`, and once with the subsetted data, `pros.df.subset`. Save the results as `pros.lm.` and `pros.subset.lm`, respectively. Using `coef()`, display the coefficients (intercept and slope) from each linear regression. Are they different?
- **1d.** Let's produce a visualization to help us figure out how different these regression lines really are. Plot `lpsa` versus `lcavol`, using the full set of observations, in `pros.df`. Label the axes appropriately. Then, mark the observations in `pros.df.subset` by small filled red circles. Add a thick black line to your plot, displaying the fitted regression line from `pros.lm`. Add a thick red line, displaying the fitted regression line from `pros.subset.lm`. Add a legend that explains the color coding.
- **1e.** Compute again a linear regression of `lpsa` on `lcavol`, but now on two different subsets of the data: the first consisting of patients with SVI, and the second consistent of patients without SVI. Display the resulting coefficients (intercept and slope) from each model, and produce a plot just like the one in the last question, to visualize the different regression lines on top of the data. Do these two regression lines differ, and in what way?

Reading in, exploring wage data

- **2a.** A data table of dimension 3000 x 11, containing demographic and economic variables measured on individuals living in the mid-Atlantic region, is up at http://www.stat.cmu.edu/~ryantibs/statcomp/data/wage.csv. (This has been adapted from the book An Introduction to Statistical Learning.) Load this data table into your R session with `read.csv()` and save the resulting data frame as `wage.df`. Check that `wage.df` has the right dimensions, and display its first 3 rows. Hint: the first several lines of the linked file just explain the nature of the data; open up the file (either directly in your web browser or after you download it to your computer), and count how many lines must be skipped before getting to the data; then use an appropriate setting for the `skip` argument to `read.csv()`.
- **2b.** Identify all of the factor variables in `wage.df`, set up a plotting grid of appropriate dimensions, and then plot each of these factor variables, with appropriate titles. What do you notice about the distributions?
- **2c.** Identify all of the numeric variables in `wage.df`, set up a plotting grid of appropriate dimensions, and then plot histograms of each these numeric variables, with appropriate titles and x-axis labels. What do you notice about the distributions? In particular, what do you notice about the distribution of the `wage` column? Does it appear to be unimodal (having a single mode)? Does what you see make sense?

Wage linear regression modeling

- **3a.** Fit a linear regression model, using `lm()`, with response variable `wage` and predictor variables `year` and `age`, using the `wage.df` data frame. Call the result `wage.lm`. Display the coefficient estimates, using `coef()`, for `year` and `age`. Do they have the signs you would expect, i.e., can you explain their signs? Display a summary, using `summary()`, of this linear model. Report the standard errors and p-values associated with the coefficient estimates for `year` and `age`. Do both of these predictors appear to be significant, based on their p-values?
- **3b.** Save the standard errors of `year` and `age` into a vector called `wage.se`, and print it out to the console. Don't just type the values in you see from `summary()`; you need to determine these values programmatically. Hint: define `wage.sum` to be the result of calling `summary()` on `wage.lm`; then figure out what kind of R object `wage.sum` is, and how you can extract the standard errors.
- **3c.** Plot diagnostics of the linear model fit in the previous question, using `plot()` on `wage.lm`. Look at the "Residuals vs Fitted", "Scale-Location", and "Residuals vs Leverage" plots—are there any groups

of points away from the main bulk of points along the x-axis? Look at the "Normal Q-Q" plot—do the standardized residuals lie along the line $y = x$? Note: don't worry too if you're generally unsure how to interpret these diagnostic plots; you'll learn a lot more in your Modern Regression 36-401 course; for now, you can just answer the questions we asked. **Challenge**: what is causing the discrepancies you are (should be) seeing in these plots? Hint: look back at the histogram of the `wage` column you plotted above.

- **3d.** Refit a linear regression model with response variable `wage` and predictor variables `year` and `age`, but this time only using observations in the `wage.df` data frame for which the `wage` variable is less than or equal to 250 (note, this is measured in thousands of dollars!). Call the result `wage.lm.lt250`. Display a summary, reporting the coefficient estimates of `year` and `age`, their standard errors, and associated p-values. Are these coefficients different than before? Are the predictors `year` and `age` still significant? Finally, plot diagnostics. Do the "Residuals vs Fitted", "Normal Q-Q", "Scale-location", and "Residuals vs Leverage" plots still have the same problems as before?
- **3e.** Use your fitted linear model `wage.lm.lt250` to predict: (a) what a 30 year old person should be making this year; (b) what President Trump should be making this year; (c) what you should be making 5 years from now. Comment on the results—which do you think is the most accurate prediction?

Wage logistic regression modeling

- **4a.** Fit a logistic regression model, using `glm()` with `family="binomial"`, with the response variable being the indicator that `wage` is larger than 250, and the predictor variables being `year` and `age`. Call the result `wage.glm`. Note: you can set this up in two different ways: (i) you can manually define a new column (say) `wage.high` in the `wage.df` data frame to be the indicator that the `wage` column is larger than 250; or (ii) you can define an indicator variable "on-the-fly" in the call to `glm()` with an appropriate usage of `I()`. Display a summary, reporting the coefficient estimates for `year` and `age`, their standard errors, and associated p-values. Are the predictors `year` and `age` both significant?
- **4b.** Refit a logistic regression model with the same response variable as in the last question, but now with predictors `year`, `age`, and `education`. Note that the third predictor is stored as a factor variable, which we call a **categorical variable** (rather than a continuous variable, like the first two predictors) in the context of regression modeling. Display a summary. What do you notice about the predictor `education`: how many coefficients are associated with it in the end? **Challenge**: can you explain why the number of coefficients associated with `education` makes sense?
- **4c.** In general, one must be careful fitting a logistic regression model on categorial predictors. In order for logistic regression to make sense, for each level of the categorical predictor, we should have observations at this level for which the response is 0, and observations at this level for which the response is 1. In the context of our problem, this means that for each level of the `education` variable, we should have people at this education level that have a wage less than or equal to 250, and also people at this education level that have a wage above 250. Which levels of `education` fail to meet this criterion? Let's call these levels "incomplete", and the other levels "complete".
- **4d.** Refit the logistic regression model as in Q3b, with the same response and predictors, but now throwing out all data in `wage.df` that corresponds to the incomplete education levels (equivalently, using only the data from the complete education levels). Display a summary, and comment on the differences seen to the summary for the logistic regression model fitted in Q3b. Did any predictors become more significant, according to their p-values?

Wage generalized additive modeling (optional)

- **5a.** Install the `gam` package, if you haven't already, and load it into your R session with `library(gam)`. Fit a generalized additive model, using `gam()` with `family="binomial"`, with the response variable being the indicator that `wage` is larger than 250, and the predictor variables being `year`, `age`, and `education`; as in the last question, only use observations in `wage.df` corresponding to the complete education levels. Also, in the call to `gam()`, allow for `age` to have a nonlinear effect by using `s()` (leave `year` and `education` alone, and they will have the default—linear effects). Call the result `wage.gam`. Display a summary with `summary()`. Is the `age` variable more or less significant, in terms of its p-value, to what you saw in the logistic regression model fitted in the last question? Also, plot the fitted effect for each predictor, using `plot()`. Comment on the plots—does the fitted effect make sense to you? In particular, is there a strong nonlinearity associated with the effect of `age`, and does this make sense?
- **5b.** Using `wage.gam`, predict the probability that a 30 year old person, who earned a Ph.D., will make over $250,000 in 2018.
- **5c.** For a 32 year old person who earned a Ph.D., how long does he/she have to wait until there is a predicted probability of at least 20% that he/she makes over $250,000 in that year? Plot his/her probability of earning at least $250,000 over the future years—is this strictly increasing?