

POLS 706
Exercises To Turn In.
Feb 24, 2010

I do not have all exercises & assignments ready at this time, but I have most of them. I will make updated versions of this handout available as the semester proceeds. All students are required to do all of these exercises, most are graded “pass” or “fail”. A-B-C grades will be assigned for a few of the assignments. Those are starred (*). Assignments should be submitted before class on the date indicated. Late submissions will be accepted only until 5pm on the day after the assignment is due.

About Email Submissions If you email your assignments to me, please make my life easy by putting the Subject of your email “PS706 lastname Ex 1,” where you replace your last name and the Exercise number. I can then easily spot your emails and filter them into a folder.

1. Due: Jan. 20. Join stat-1, post a brief message to introduce yourself. Give your name and brief information on “where you came from.” Add one paragraph that describes some kind of data analysis project you would like to do. What kind of data do you wish you had?
2. Due: Jan. 27. I need proof that you started R, explored `help.start()`, and fiddled around. Below you find a sample program. If you run R inside Emacs, it should be easy enough to just save the R session in a text file and print it out. If you run R in some other way, it might be harder to prove you actually did this. I’d accept a “screenshot” of a running R session or the output of the following R commands as evidence of your progress.

```
> set.seed(9)
> myFunkyVariable <- rnorm(100)
> mean(myFunkyVariable)
> sd(myFunkyVariable)
```

3. Due: Feb. 3. There are 2 objectives. 1) create plots, and 2) save them into various formats. For R, pdf is now the default format, but there is still a special place in my heart for encapsulated postscript (eps, the old favorite). You can use any data you want, or make it up.
 - (a) Create any barchart, save it into an “encapsulated postscript (eps)” file. Set the width at 5 inches and the height at 6 inches. Email me the output file. In the Subject of the email, put this “PS706 lastname Ex 3a filename.eps” Replace “lastname” with your name, of course, and “filename” with the file name.
 - (b) Create a pdf output of the same barchart. To make this a bit more interesting, create the pdf output in a funny shape. Lets say you make it 7 wide by 4 high. Email that one to me as “PS706 lastname Ex 3b filename.pdf”. Take care that the pdf you create does not have that huge white space at the top. As I recall, you can get there by specifying the height, width, paper=”special”, onefile=F.

- (c) Create any histogram, save it into a “png” file. Set the width at 1000 pixels and the height at 1200. Email me the png file. In the Subject of the email, put this “PS706 lastname Ex. 3c filename.png” .
 - (d) For extra credit, save a file using some other device. (type ?device to see the possibilities on your system). The xfig device is available, I think that’s fun. You can edit that in the xfig program (that’s for drawing line art). It may be your system has a device to make svg files (scalable vector graphics; run ?svg to see). Inkscape can edit svg files.
4. Due: Feb. 5. Little LyX Exercise as described in the Stuff Worth Knowing Handout (Chapter 6.3). Hopefully, I will have a more polished set of guidelines by the time we get there.
5. Due: Feb. 12. Become a useR.
- (a) Get a gmail email account and subscribe to the r-help email list from that account. Do that from gmail because the volume of traffic on r-help would crush your ordinary KU email account.
 - (b) To keep your inbox from filling up, within your gmail account, create a filter that will label your mail from r-help. In my account, the filter is set with only the TO option specified, and the value is “r-help*”. Then the default action is to archive emails and skip the inbox. Otherwise your inbox becomes cluttered.
6. * Due: Tuesday February 17. Create a document that has (at least) these elements. Use any data you want. You may use any of the example datasets you have been using as you work on the exercises in Verzani or you may use any other data you collected in other courses, including POLS 705. When I say “write one paragraph” I mean act as though you are presenting this in a paper, so try to interpret it as best you can.
- (a) Create a histogram, present it in your paper, and write one paragraph about it. This might look nicer if you include the “density” curve with it, but it is not a requirement.
 - (b) Create a cross tabulation table that is consistent with The Iron Law of Crosstabs and present it in your paper and write one paragraph about it.
 - (c) Create a barplot, present it in your paper and write one paragraph about it.
 - (d) Create a boxplot, present it in your paper and write one paragraph about it.
 - (e) Create a scatterplot, present it in your paper, and write one paragraph about it.
 - (f) This is an Extra Credit component. Explore variations on the scatterplot.
 - i. Choose different plotting characters by changing the pch option.
 - ii. Replace the plotting character with letters that represent something informative.

You should make a pdf file and you should email me with the Subject “PS706 lastname Ex4 myfile.pdf”. I do not need to see your lyx document or the other files that might be incorporated in it. The pdf file will give all the proof I need.

7. Feb. 19. Email me your best effort on the problems at the end of the “summation” chapter in Stuff Worth Knowing.
8. Feb. 24. Email me your best effort on the problems at the end of the “plotting lines” chapter in Stuff Worth Knowing.
9. February 26. Matrix algebra assignment. There are some “paper and pencil” matrix exercises at the end of my chapter on matrices in Stuff Worth Knowing. Write those up and hand them in. I’m not correcting them, but if you have any uncertainty about any of them, you should make sure you learn what to do. You don’t have to type these, you can write by hand. Give me some paper, either way. I’ll give you the correct answers in return.
10. March 3. Prove you can download and import data in either text and Stata data form.

I will upload these 2 files:

<http://pj.freefaculty.org/stat/ps706/practiceData.txt>

<http://pj.freefaculty.org/stat/ps706/practiceData.dta>

practiceData.txt is a text file that you can just look at and see it has a header row and that the symbol | is used as the separator. You use "read.table" to access that data, with the appropriate separator and header options specified, of course.

The Stata data set can be accessed if you load the "foreign" package with the read.dta function.

- (a) The first problem you face when you import data is to address the question, “what do I have?” “What sorts of variables are they? (integers, real-valued numbers, ordered factor variables, unordered factors)” “Are there any typographical errors in this data?”

So, consider each data set. How would you go about finding about these variables?

- (a) For each one, tell me

- which variables are numeric? (either integer or real-valued). Functions like “str” and “is.numeric” or “is.factor” or “is.integer” “attributes” might help.
- which are factors, and for factors, what are the “levels”? I’d suggest creating a numeric version with “as.numeric” and then making a table that compares the original and the new variables. Also, remember to run the “attributes” and “levels” functions to see the details.

- (a) For one of the numeric variables, calculate a “central tendency” indicator and a “dispersion” indicator. Create a histogram (with density curve) and write the

indicators in the figure. Round your numbers after the second decimal value. Write 1) one paragraph that explains elements in the figure and 2) one paragraph that discusses the relevance of the indicators to the observed display.

- (b) For one of the factor variables, create a table that illustrates the observed distribution of cases. What happens if you try to apply the “hist” function to that factor variable? What happens if you try to apply the “sd” function to a factor variable? Can you suggest any ways you might want to summarize central tendency and dispersion in a factor variable?
- (c) Just for fun, create one scatterplot. Put a predictive “regression line” (of any sort you like) on the plot. Label your variables. I like this plot based on the text data set, which I called “dat”. You could fancy it up by fiddling with the symbols if you want.

```
mean(dat)
hist(dat$inc)
plot(inc ~ ed, type="n", data=dat)
text(dat$ed, dat$inc, labels=letters[dat$grp])
### This adds a legend
legend("bottomright", legend=c("a=group1", "b=group2", "c=group3"))
```

- (d) I’ve mentioned the “by” command in class. This demonstrates one use of it. It also shows one of the ways to draw several small graphics into a larger figure. I’m not completely sure this is the best way to do it, partly because R has more sophisticated layout functions, but also because, if I were making a publication, I would save the separate small graphics in separate files and integrate them within L^AT_EX.

```
by(dat, dat$race, function(theData) mean(theData[, c("wealth", "school")], na.rm=T))
### I've not seen you placing several plots on one page before.
### here's one way to do that: 3 figures in 1 column
### the alternative is the "layout" function, which is newer.
par(mfcol=c(3,1))
by(dat, dat$race, function(theData) hist(theData$wealth, xlab="wealth", main=levels(theData$race)[theData$race[1]]))
## This re-sets the plotter to do one plot per page
par(mfcol=c(1,1))
```

11. March 5. I need to make up some “re-coding” data exercises for you to do. I am certain I had them, but can’t find them.

12. March 31. Lets make illustrations of some statistical distributions. When this is finished, I want you to email me an R program that generates the figures and a pdf file that has the figures included in it with pleasant titles and such. In your R program, include a comment at the top with your name, the date, and a brief title or description of the project. (You'll notice that when I circulate R code, I try to remember to put that at the top.) I want to be able run your code line by line to generate the same results that you have. That means you should give me a clean file, no mistakes remaining. Don't forget to put a title on your email like "ps706 lastname probability exercise".

- (a) Create a figure that illustrates the pdf of a Normal distribution for which the mean is not 0 and the standard deviation is not 1.

Hint 1: create an x variable with seq and then use dnorm to find out how likely each point might be.

Hint 2: to draw, use plot(x,y, type="n"); lines(x,y); to draw a big empty figure and add a line into it.

Be sure to label the components of your figure appropriately. Use an R text command to write the mean and standard deviation of the distribution on the plot. Label the curve. I'll be entertained if you use R's arrow function to point from your label to the curve.

- (b) Create a figure that illustrates the pdfs of 2 Normal distributions. One should be the same distribution as the previous question. For the second Normal, choose one with the same mean, but set a new value for the standard deviation. Please choose a line type that is different for this second line and include a "legend" to explain which line represents which distribution.

- (c) Draw random samples of 500 observations from each of those 2 distributions. Then create histograms for the 2 samples. These should show proportions (probabilities). I think it will work best if you either use R's par(mfcol=c(2,1)) to put the 2 plots into one graphic image, or use L^AT_EX's subfigure thing to put 2 separate histograms on the same page. Either way will be OK with me, as long as we can compare the two samples. You should include density curves in your histograms. Oh, and also include a line that represents the pdf—the true probability value—in each graph.

- (d) Extra Credit. This part is difficult, but valuable. R has a more-or-less L^AT_EX like ability to take text and convert it to mathematical symbols. They call that "plotmath" and "?plotmath" shows the help page, but, weirdly, there is no "plotmath" function. plotmath is ubiquitous, ever-present in all text plotting in R. Run these commands to see the basic approach:

```
plot( 1:10, 1:10, type="n")
text ( 4, 5, expression(paste( alpha , " is alpha")))
text ( 7, 3, expression(paste( beta * alpha, " is beta times alpha")))
text (2, 6, expression(paste(frac(beta, alpha), " is frac(beta,alpha)")))
text(2,9, expression(paste( y[i] == beta[0]+beta[1]*x[1]+hat(e))))
```

The aim is to create an equation in the white space of the figure that represents the formula of the pdf. If you can't figure how to make R write the equation for you, then you have to do something else that is tedious and horrible. Maybe you learn to write equations in some other program, and you make a tiny picture of the equation, and then learn how to put that into the R plot (hassle!). Maybe you save your R plot as an Xfig document, and then use xfig to edit the drawing and insert the equation (hassle!). Maybe you get a pdf editor and add the equation in after your project is done (hassle!). So please try `?plotmath` and `example(plotmath)` and know the joy.

13. *April 2. Repeat the process that you went through in the previous assignment with the Normal, but now do the same thing for a different probability model. You can pick any nonuniform distribution for continuous data, such as Beta, exponential, Gamma, Dirchelet, or whatever else you can find in R or any of the packages available for R. In my web pages, under "stat", you should see a folder called Distributions. That has essays I wrote about the distributions with some students.

- (a) Create one figure that illustrates the pdf when the essential parameters are set at some standard value.
- (b) Create one drawing that illustrates the change in the pdf that results when you change one parameter in the distribution. Be sure to label the curves so I can tell which one represents which value (maybe use a legend for that).
- (c) Illustrate the histograms of 2 different samples from your chosen distributions (put onto same figure in output, either by manipulating R to force them into same graph or using subfigures in L^AT_EX).
- (d) Email me your R program and a pdf file that includes your Figures. Remember to use an informative email subject and to put name/date/title as comments in your R code.
- (e) Extra Credit activity. Run this code to create a Figure that illustrates the probability mass function of a Binomial distribution. Choose the drawing that you think works "best", put it in your document and write a few sentences about why it is the best figure.

```
x<- seq(0, 10)
probx <- dbinom(x, size=50, prob=0.1, log=F)
###experiment with plot types, see the differences
plot(x, probx, type="p")
plot(x, probx, type="l")
plot(x, probx, type="s")
plot(x, probx, type="h")
points(x, probx, pch=16)
```

- (f) Extra Credit 2. Load the "plotrix" package (install if you don't have it, of course) and run some of their examples. Print out any graph that uses plotrix, include it in your document, and include a little explanation of what is supposed to be neat about the figure..

14. April 7. Do Verzani Exercises: 6.3, 6.4, 6.9, 6.10, 6.11
15. April 9. Do Verzani Exercises: 7.10 , 7.17, 7.18, 7.31
16. April 14. Do Verzani Exercises: 8.9, 8.10, 8.12, 8.16, 8.31, 8.33
17. April 23. Regression Exercises. Do Verzani Exercises 10.1. For each one, I want you to present a fitted model in this format

$$\widehat{output\ variable}_i = \hat{b}_0 + \hat{b}_1 \cdot input\ variable_i$$

where you replace the $\hat{b}_j, j \in \{0, 1\}$ with the OLS estimates and replace the words “output variable” and “input variable” with meaningful names.

Then provide the predicted values that Verzani asks for. I want you to try to do this in 2 different ways. Here is a sketch of the first approach. You have to replace “whatever.data,” “whatever.x” and “whatever.y”, of course.

```
mymod1 <- lm ( whatever.y ~ whatever.x, data= whatever.data)
p1 <- predict(mymod)
newWhatever.data <- cbind ( whatever.data, p1)
```

Here’s a second way to get predicted values. Rather than getting predictions for all cases, let’s just check the quantiles.

```
myq <- quantile(whatever.data$whatever.x)
p2 <- predict(mymod, newdata= data.frame( whatever.x= myq ) )
```

You cannot add p2 to the data frame “whatever.data.” You see why, don’t you?

Once you see that p2 is meaningful, then you can insert other values of interest into your newdata. For example,

```
myq <- c( quantile(whatever.data$whatever.x), 752, mean(whatever.data$whatever,
p3 <- predict(mymod, newdata= data.frame( whatever.x= myq ) )
```

There is some trouble if whatever.x is a factor variable, because the legal values of factors are limited to values displayed in the attributes of variables. There is plenty of information in my R writeups about “working with factors” so that you can see how this should be done. But as long as the input variable is numeric, it is simple.

18. * May 5. Regression Exercises.
 - (a) At <http://pj.freefaculty.org/stat/ps706>, I uploaded a data frame called “mydf_example.txt”. There is one predictor, x, and there are 3 output variables, y1, y2, and y3. Download that data, use read.table to bring it into R, and then do the following.

- i. Make a scatterplot of each output variable as a function of x . On each one, draw an “eyeball” regression line and figure out the intercept and slope of your eyeball line. Put those eyeball estimates into this format

$$\widehat{y}_i = \hat{b}_0 + \hat{b}_1 x_i$$

where you replace the $\hat{b}_j, j \in \{0, 1\}$ with your eyeball estimates.

- ii. Use the R `lm` function to fit a linear regression for each of the output variables. Use the summary function to view the results. For example,

```
mod1 <- lm(y1~x, data=mydf)
summary(mod1)
```

- iii. Create one table that summarizes your three regression models. I want you to include the estimates of the intercepts, slopes, standard error of the b 's, the residual standard error, and the R^2 .

Write one paragraph about the estimates that are similar in the 3 models.

Write one paragraph about the estimates that are different in the 3 models.

- iv. You can use the following approach to create a scatterplot and draw the fitted regression line on the plot. Supposing your data frame is called `mydf`:

```
mod1 <- lm(y1~x, data=mydf)
plot(y1~x, data=mydf, main="A regression of y1 on x")
abline(mod1)
```

I want you to make 3 plots like that, one for each output variable. I don't want to see them printed separately, however. Create three plots on one page. Please take care that each of the scatterplots uses the same range for the output variable.

- (b) In the “car” package, there's a data frame called `Prestige`. It has 6 variables. Let's see if we can predict income from education.

- i. Make a scatterplot of the relationship between education and income.
- ii. If it appears to you that a linear model might offer an acceptable fit, estimate one with the “lm” function. (Hint: I think it is acceptable, so if you say it is not, you better have a good reason).
- iii. Copy and paste the results as they appear in your printout into your homework. With a ball point pen, I want you to circle and label the estimates of the following values then write one or two sentences to explain what each one means.
 - A. Estimate of the intercept
 - B. Estimate of the slope of income as it depends on education
 - C. The Standard Error of the estimate of the Intercept.
 - D. The Standard Error of the estimate of the slope of income as it depends on education.
 - E. The Residual Standard Error.

- iv. Suppose your model object is called “mymod”. Run this command to create a new graph:

```
termplot(mymod, partial.resid=T, se=T)
```

Present that graph in your paper. What do you think the hourglass shaped area represents?

- v. EXTRA CREDIT. Use my “outreg” function or a similar function in “memisc” or “apsrtable” to create a publication quality regression table. Include that table in your homework.
- (c) [PLACEHOLDER] I’m looking for one more data set on which to base regression examples.