Demo 4: Correlations and regression

Experimental and Statistical Methods in Biological Sciences I

October 6, 2014

1 Data preparations and descriptive statistics

1.1 Description of the dataset

In this dataset, we are attempting to identify predictors of student performance. Based on theory and past research, we anticipate that most of the variables identified here will be associated with performance in one way or another. We do not know, however, whether they all contribute separately, or whether one (or more) predictors explain the contribution of the others.

The variables are:

- Performance = an exam mark - higher values indicate a better score
- Hours = hours of class missed - higher score indicates poorer attendance
- Educ = years of education prior to taking this course
- Rating = student rating of the course upon completion
- Entry = score on an exam taken in the first week of the course
- Extra = additional work which did not contribute to the final mark - amount of additional work was recorded, but not marked
- Stress = self-reported stress level - higher values indicate greater stress

1.2 Import and review the data

Load in the data from `http://becs.aalto.fi/~heikkih3/students`. This time the data is in a .txt file instead of the usual .csv format we have seen so far, so now you cannot use the `read.csv` function. Instead, use `read.table` like this:

```
students <- read.table('http://becs.aalto.fi/~heikkih3/students', header=T, sep="",
            na.strings="NA")
```

Take a quick look at the data with `head` and `summary`.

Question 1 Can you identify any problems with the variables?

Look at the data for outliers. Try using `which, boxplot`.

Question 2 Can any of the values be considered errors, or an attempt to code values as NA?

Recode the extreme cases as NA if appropriate, but save to a new data frame "students2".

Question 3 What is the advantage of saving to a new data frame, rather than rewriting the original?

Question 4 Try plotting the data visually to examine trends. Use scatterplots, boxplots, and histograms.

Question 5 When looking at the plots, what associations can you see between "Performance", and other variables?

2 Correlations

2.1 Pearson

Correlations between two variables can be tested by using `cor.test`:

```
cor.test(Performance, Hours)
```

We see that performance and hours of classes missed are correlated ($r(221) = -.33, p < .001$). A scatterplot might clarify the relationship (but the direction of correlation already tells the same story):

```
plot(Performance, Hours)
```

You can also examine the whole correlation matrix between all variables in the data set with `corr.test` (from package 'psych'). (Note that this is also a very useful summary when examining the associations between variables before running regression.)

```
library(psych)
corr.test(students2)
```

Question 6 Do all the variables correlate with each other significantly? If no, which don't?

Question 7 What is our maximum and minimum sample size? Why does this matter?

Look at the non-significant correlations. Remember, we expected all of our variables to associate significantly with "Performance".

Question 8 Why might these variables not associate significantly with performance in our sample? Hint: visual inspection of the variables through plots may help.

Question 9 Are all the correlations positive?

2.2 Spearman

By default, all the correlation functions use Pearson correlation coefficient. If we want to take correlations using Spearman's correlation coefficient instead, we simply need to add a named argument "method":

```
cor.test(Performance, Hours, method="spearman")
```

Question 10 Take Spearman correlation matrix for correlations between all continuous variables.

Question 11 Do results Pearson and Spearman give different results? Take both matrices and look at the p values. Tip: you can save the results from correlations tests as objects and observe them. Try:

```
# Save the correlation tests:
pearson <- corr.test(students2[c(1:2,4:7)])
spearman <- corr.test(students2[c(1:2,4:7)], method="spearman")
# Observe the structure of the object:
str(pearson)
# Take, e.g., the p values from both objects:
pearson$p
# ... too many decimals for me, try this instead:
round(pearson$p,2)
round(spearman$p,2)
```

2.3 Partial correlations

If you want to calculate partial correlations, you can use the package 'ppcor'. For instance, you can calculate the partial correlation between Performance and Entry while controlling for Hours:

```
install.packages('ppcor')
library(ppcor)
pcor.test(Performance, Entry, Hours)
```

3 Regression

3.1 Running a basic model

Past work suggests that attendance predicts performance very well. Create a regression model with "Performance" as the predicted variable and "Hours" as the predictor. Store the model in an object.

```
model1 <- lm(Performance ~ Hours)
```

We have a lot of information stored in the model (if you don't believe it, you can always try `str(model1)`. Use `summary` to examine the model. You can also try `coefficients, confint`.

We see that performance is significantly predicted by attendance ($b = -3.3, t(221) = -5.17, p < .001$) (you might usually want to present this in a table). The model fit is good ($F(1, 221) = 26.67, p < .001$, adjusted $R^2 = .10$).

You can interpret the direction of the trend by looking at the beta values. In this case, better performance is predicted by fewer missed hours.

3.2 Evaluating assumptions

Multicollinearity

Look at the correlation matrix with `corr.test`. See whether any correlations are high (e.g., above .6).

Normality

Look at he regression plots using `plot`. (Remember to adjust the layout if you wish.) You can use the Q-Q plot to check for normality, or use histograms or normality tests.

Homoscedasticity

Variance should be homogenous. You can use either the "Residuals vs Fitted" plot or `ncvTest` from package 'car'.

Influential cases

You can examine the "Residuals vs Leverage" plot. In the case of model1, you can see that no values have a Cook's distance in excess of .5. Note that usually a dotted red line indicates the accepted limit - in this case the line is missing because no values are close to exceeding these limits.

Alternatively, you can use `influence.measures` to examine all cases.

3.3 Multiple regression

We want to start adding predictors to our initial model to assess the independent contribution of each. Based on past evidence, we expect that performance at the start of the course ("Entry") is a good choice. We expect this to be relatively independent of "Hours".

```
model2 <- lm(Performance ~ Hours + Entry)
```

Question 12 Is the new predictor significant?

You can compare two models using `anova`: which one has a better fit?

Question 13 Is the new, less parsimonious model better than our first model?

Question 14 If we wanted to continue evaluating predictors, would we keep "Entry" in the model or remove it?

Question 15 Why would it be a bad idea to include "Extra" in the model? Try running a model with it included if you are not sure.

Question 16 Try using "Hours" and "Rating" as predictors, in that order. Remember to save it as a new model for comparison.

Question 17 Is this new predictor significant?

Question 18 Is this model significantly better than our first model?

Question 19 What value do you want to examine to understand how much of the variance in Performance is accounted for by the model?

Question 20 How much of the variance is explained when using only "Hours" and "Rating" as predictors?

3.4 Interactions

We want to explore potential interactions between "Hours" and "Rating". Create a model that tests for the main effect of each and an interaction between them.

```
model1 <- lm(Performance ~ Hours * Rating)
```

Question 21 Which should be entered first, main effects or interactions?

Question 22 Do the predictors interact significantly?

Question 23 Does keeping the interaction in the model improve it?

3.5 Nonlinear effects

We want to investigate whether "Rating" has nonlinear effects. Create a new variable in the model: "Rating2". To do this, multiply "Rating" by itself. Make sure you tell R the variable ought to be located in your dataset.

Question 24 What sort of shape does a quadratic function take?

Try running a model with just "Rating" and "Rating2" included.

Question 25 Does "Rating2" predict performance? What are the implications of this result?

Try running a model with all possible predictors (including interactions) in the model.

Question 26 Are the R^2 and adjusted R^2 values higher or lower compared to a model with just "Hours" and "Rating" as predictors?

Question 27 Why?

4 Contrasts

Last week, we learned about ANOVA. The topic we did not go through was that of building contrasts, i.e., planned comparisons.

In a linear modeling context, it is useful to recode the variables to test specific hypotheses. This can be effectively done using effects coding.

In our current data set, the only factor is Education, so we will use that as an example.

For instance, Education has three different levels (years 12, 13, 14). We might want to use ANOVA to see whether years of education have an effect on some other variable, for example on the performance. Running ANOVA might give us the main effect, but we still do not know whether it is the differences between years 12 and 13, years 13 and 14, or years 12 and 14 that cause the differences. Last week we learned about post-hoc comparisons. However, the preferred method are contrasts which can be done using effects coding in R.

Building up contrasts

Start by checking the levels of the categorical variable.

```
levels(Educ)
```

Define the contrasts by adding contrast vector to the variable. Let's hypothesize that years of education 14 differs from both of the groups with less education, and that years of education 12 and 13 differ from each other.

```
contrasts(Educ) <- cbind(c(-0.5,-0.5,1),c(1,-1,0))
```

Finally, let's see how your contrasts were stored:

```
contrasts(Educ)
```

Running the comparisons

Running ANOVA or linear model with a variable with contrasts will automatically include the contrasts in the model.

```
# ANOVA:
A1 <- aov(Performance ~ Educ)
# Linear model:
M1 <- lm(Performance ~ Educ)
summary(M1)
```

5 Exercises

We will again turn to the "naming" dataset, in the wide format that we used last week. The following variables are included:

- iq = intelligence (continuous)
- hrs = hours spent on reading (continuous)
- sex (female, male)
- ms.regular = average time taken to read regular words (continuous)
- ms.exception = average time taken to read exceptional words (continuous)
- reading_time = reading category based on hours spent on reading (high, low, medium)

You can find the dataset here: `http://becs.aalto.fi/~heikkih3/naming_wide.csv` . Load the dataset into a data frame in R using read.csv. Remember to check for factors and missing values first! Correct these if necessary using guidelines from previous weeks.

Once your data frame is in its final format, it is useful to attach it. First, remove other attached datasets with `detach`, then attach naming instead (`attach`).

Question 28 Examine the correlations between the variables in the data set. Use scatterplot and correlation matrix.

Question 29 Do you have equal sample size in all variables?

Question 30 What significant correlations you can find in the data? Are these positive and negative? How do you interpret the significant correlations verbally?

Question 31 Start with a simple model where you predict "ms.regular" with "iq". How would you formulate the hypothesis you are testing?

Question 32 Save and run the model for Question 31. Report the results.

Question 33 Next we will add "hrs" to the model. How would you formulate the hypothesis you are testing? Save and run the model.

Question 34 Compare the models from Questions 32 and 33. Which one fits the data better? What are the implications?

Question 35 Next, we will add the interaction effect of intelligence and hours spent on reading. Save and run a new model with the interaction effect. Compare it with the best model from Question 34. Does adding the interaction effect improve the model?

Question 36 Next, we will turn to comparison between groups of categorical variables using contrasts, i.e. effect coding in R. Our first target categorical variable is reading_time. The comparison we want to make is between high and low reading groups. First, create a new variable "ec_1" and assign zeros to it. Second, add the effects coding: note that coding for reading_time as high = 1, medium = 0, and low = -1 contrasts the effect of high with low while minimizing the effect of medium.

Question 37 Save and run an anova model (using `aov`) where you include your contrast from previous Question. Are the results significant? How do you interpret the results?

Question 38 Next, add also the comparison between high and medium reading groups. Create again a new variable "ec_2" and assign zeros to it. Using the same logic as before, add the effects coding that codes for the contrast you want to make.

Question 39 Save and run an anova model (using `aov`) where you include your contrast from previous Question. Are the results significant? How do you interpret the results?