Biostatistics 208 Lab #3 1/22/15

The purpose of this lab is to give you practice in dealing with categorical predictors and log transformations in linear regression models. Note that many of the commands covered will carry over to applications in logistic and Cox regression models without change. We will use baseline data from the 2763 post-menopausal women with heart disease and without diabetes who participated in the HERS trial. You can download `lab3.dta` from the course website. We will focus on the following variables in the dataset:

1. `bmi` – body mass index (BMI) in kg/m^2
2. `physact` – relative physical activity by self-report, coded 1-5
3. `age10` – age in units of 10 years
4. `smoking` – an indicator of current smoking (1 = yes, 0 = no).
5. `drnkspwk` – average drinks per week.

First, we focus on `physact`, the 5-level self-reported physical activity variable discussed in lecture, as an ordinal categorical predictor of BMI. Start off by summarizing this variable using the `tabulate` command with the `summarize()` option to see how many women there are in each category, and at the same time obtain the mean and SD of BMI in each group:

```
tab physact, sum(bmi)
```

Now we'll use the `i.` command prefix to regress `bmi` on `physact` as a five-level categorical predictor, or *factor* variable. Recall from lecture that this prompts Stata to create the indicator variables needed to represent the distinct levels of `physact`. Only four of the five are needed (due to the inclusion of the intercept term); Stata omits the indicator for the first level by default. Note that categorical variables need to have non-negative integer values (e.g. 0, 1, 2, …) to be used as factors in regression models. Non-numeric variables or this with negative numeric values need to be recoded for this purpose You can verify that `physact` will work by looking at its values with the `codebook` command.

```
reg bmi i.physact
```

You can easily see that the coefficient estimate for the intercept (`_cons`) is the sample mean for BMI among "much less active" women (`physact=1`). Now fit the model again, this time specifying the `baselevels` option:

```
reg bmi i.physact, baselevels
```

This option lists the reference level of factor variables included in the model using `i.`. The coefficients of the regression model for `physact` summarize the relationship with BMI as differences in mean BMI for each of the activity levels relative to the lowest (reference) level.

Use the Stata code on the next page to loop through the `physact` groups, and compute the group BMI means and 95% confidence intervals based on the appropriate regression coefficients. The commands must be typed exactly as written, on three lines, including the brackets; the opening single quote in the second line is a back quote (also an accent grave), and must be typed using the key usually located on the upper left of the keyboard (between the tab and esc keys). The closing single quote is an apostrophe (left of return key).

```
forvalues i = 1/5 {
lincom _cons + `i'.physact
}
```

Stata help for the `forvalues` command (and the related command `foreach`) has other examples of how to run loops in Stata, which can make some repetitive data management tasks much easier. We could also get the group-specific BMI means using the `margins` command:

```
margins physact
```

The `margins` command was introduced in Stata 11, and can be used to estimate the *marginal* mean outcome levels associated with a predictor of (e.g. `physact`). Note that `margins` is a fairly general command with a number of options. In this case with a single predictor, it performs the same task as the `lincom` and `tab` statements issued above. With additional predictors in the model, `margins` performs the same task, but produces marginal mean estimates accounting for the presence of other variables in the model. (We'll see an example of this later in the lab.)

Recall from lecture that assessing the overall significance of a categorical predictor is complicated when the *t*-tests for the individual coefficients are considered (i.e. this raises the issue of multiple comparisons as well as failing to provide a global summary of the contribution of physical activity). For `physact`, we see that the highest two activity levels are significantly different that the lowest level. However, if we wanted to report on the overall significance of `physact` as a predictor, what would we report? Note that if `physact` made no contribution to predicting BMI, we would expect that average BMI would be constant across levels of activity. Equivalently, the differences in mean BMI between levels (i.e. the regression coefficients) would not differ significantly from zero. This observation suggests that we would like to evaluate the *homogeneity hypothesis* that the average outcome levels are the same across categories of activity. Taking this as the null hypothesis to be disproved, the alternative implies *heterogeneity* of group-specific averages. Note that in a model including only `physact` as a single categorical predictor, the overall *F* test provided in the default regression output addresses exactly this hypothesis (i.e. the default *F* test compares the fitted model to a model excluding predictors). However, when other predictors are included, the default test provided does not address the effect of `physact` separately. In this case, the desired test can be performed using the `testparm` command:

```
testparm i.physact
```

This command must follow the `regress` command that fits the desired model. In this example, your result should be identical to the default *F* test provided in the regression model output. The `testparm` command (and the companion command `test`) tests a variety of hypotheses about coefficients from a previously fitted model. In the syntax given above, it evaluates the null hypothesis that the true coefficients for the effect of activity are all zero (which is equivalent to the homogeneity hypothesis discussed above).

Question: Is there convincing evidence for heterogeneity in mean BMI across levels of physical activity?

The conclusion that average BMI differs significantly between levels of reported physical activity does not allow us to conclude anything about the *nature* of the observed differences. Since we can regard `physact` as an ordinal categorical variable, an evaluation of possible trend in average outcomes with increasing levels seems natural. Since a linear trend is the simplest trend to

evaluate, a test for linear trend is useful here. As discussed in lecture and in section 4.3 in the book, since the numerical levels of `physact` don't have a quantitative interpretation, treating this variable as continuous and evaluating the slope would yield a result that depends inappropriately on the numerical scale. The more general trend tests described in section 4.3.5 of the book avoid this issue.

The trend test evaluates a *contrast* of the model coefficients consistent with a pattern that would be expected in the presence of a linear trend. The appropriate contrast depends on the number of categories represented in the predictor of interest (i.e. five in the case of `physact`). The following command requests a linear trend test using the `contrast` command:

```
contrast q(1).physact, noeffects
```

The `q()` operator requests a contrast assuming evenly spaced categories of `physact`, and the 1 specified in parentheses asks for a linear contrast. (Note that unequally spaced categories can also be treated by correct choice of numerical category values and use of the `q()` operator for the `contrast` command.) The `noeffects` option suppresses additional non-useful output. The same test can be specified using the `test` command and the appropriate coefficient contrast pattern from table 4.7 as follows:

```
test -2.physact + 4.physact + 2*5.physact=0
```

Question: Is there a statistically significant linear trend?

As discussed in section 4.3.5 of the book, the finding of evidence for a linear trend in means leaves open the possibility that the trend is actually nonlinear (e.g. quadratic). The book provides a procedure to evaluate evidence for a departure from linearity that is appropriate for ordinal categorical predictors. Given evidence for linear trend, you can further test for the presence of quadratic, cubic and quartic departures from linearity using the `contrast` command again, changing the argument provided to the `q` operator accordingly:

```
contrast q(2/4).physact, noeffects
```

The results indicate evidence for some additional nonlinearity in this example (and this can be seen by examining the activity group BMI means). The `joint` entry in the `contrast` output addresses evidence for joint effects of all three types of nonlinearity. In reporting these results, you might provide the activity group-specific means and confidence intervals, along with a statement of significant evidence for a linear (decreasing) trend in average BMI with increasing self-reported activity level, as well as an indication of departure from nonlinearity.

The above analyses consider `physact` as the sole predictor. In most cases, models controlling for other explanatory predictors would also be of interest. Next, we will re-do the above analysis adjusting for age, smoking, and alcohol use, all possible confounders of the association between physical activity and BMI (but certainly not an exhaustive list).

```
reg bmi i.physact age10 smoking drnkspwk
```

This model represents the effects of activity on BMI in terms of changes from the reference level, adjusting for the effects of the other included predictors. By definition, these effects are fixed, for

any specified fixed levels of the other predictors. If we want to estimate adjusted average BMI for the different activity levels (like we did above for the unadjusted regression), we will need to specify what to do with the other predictors in the model. For example the `margins` command will estimate adjusted mean BMI in each of the 5 groups. Without specifying options, `margins` estimate group means for each activity level averaging over the observed values of all the other variables:

```
margins physact
```

In this (default) version of `margins`, the mean for each group is estimated from the entire sample used in fitting the model, by fixing physical activity at a specified level for all individuals, and using the model to estimate the mean BMI for everyone, allowing the values of other predictors to remain at their observed values. These estimates can be interpreted as adjusted average predicted average BMI at the different activity levels. This approach is also used in producing *average causal effect* estimates, as illustrated in the second lecture. You can use the following commands to manually calculate the marginal means using the estimate coefficients (the "`if e(sample)`" qualifier insures that your estimates are based on the data used in the regression model):

```
gen y1 = _b[_cons] + _b[1.physact]*1 + _b[age10 ]*age10 + _b[smoking]*smoking +
_b[drnkspwk]*drnkspwk if e(sample)
gen y2 = _b[_cons] + _b[2.physact]*1 + _b[age10 ]*age10 + _b[smoking]*smoking +
_b[drnkspwk]*drnkspwk if e(sample)
gen y3 = _b[_cons] + _b[3.physact]*1 + _b[age10 ]*age10 + _b[smoking]*smoking +
_b[drnkspwk]*drnkspwk if e(sample)
gen y4 = _b[_cons] + _b[4.physact]*1 + _b[age10 ]*age10 + _b[smoking]*smoking +
_b[drnkspwk]*drnkspwk if e(sample)
gen y5 = _b[_cons] + _b[5.physact]*1 + _b[age10 ]*age10 + _b[smoking]*smoking +
_b[drnkspwk]*drnkspwk if e(sample)
summ y1 y2 y3 y4 y5
```

As an alternative, you can use `margins` to estimate the group means holding the other predictors fixed at their sample mean values:

```
margins physact, atmeans
```

Notice that the group means in the last two examples depended on what we specify about the other predictors in the fitted model. However, the differences between group means relative to the reference activity level are fixed, and equal the estimated regression coefficients in both examples, as required by the model. Check this.

Next, repeat the assessments of heterogeneity and trend in the adjusted model.

```
testparm i.physact
```

```
contrast q(1).physact, noeffects
```

Question: Using the output, make sure you can you can:

- Interpret the regression coefficients for variables `_cons`, `physact` (levels 2-5), `age10`, and `drnkspwk`.
- Interpret the results of the tests for heterogeneity and trend in the light of the pattern in the adjusted means produced by the `margins` command.

- Relate the differences between the adjusted group means given by the `margins` command to the coefficient estimates for variables `_cons` and `physact` (levels 2-5). Recall that the coefficients for the four `physact` indicators estimate adjusted differences in mean BMI with respect to the reference group (level 1 of `physact`).

Now we briefly review interpretation of models including (natural) log-transformed predictors and outcomes, using the examples considered in lecture.

First, let's consider log creatinine (`lncreat`) as an outcome. Fit the model considered in lecture as follows:

```
regress lncreat bmi age
```

You can use kdensity or qnorm to check that residuals from a regression model based on the untransformed version of creatinine seem less normally distributed than those from the log-transformed version.

The resulting coefficients have the interpretation of the estimated change in average log creatinine level associated with a one-unit increase in the predictor (say, `bmi`), holding the other predictor fixed. Since the difference between the means of the log values is approximately the same as the log of the ratio of the means of the untransformed values, the coefficients can also be viewed as giving the log of relative change in average creatinine associated with the one-unit increase in the predictor. Thus, exponentiating the coefficients (using the `eform` option) results in effects interpretable as the relative increase in average creatinine associated with a unit increase in the predictor.

```
regress lncreat bmi age, eform("exp(beta)")
```

The text in quotes provided to `eform` specifies a label for the coefficient column in the output. Now, the coefficient for BMI gives the relative change (in this case, increase) in average creatinine associated with a unit increase in BMI (adjusted for age). This can be expressed as a percentage as follows:

```
nlcom 100*(exp(_b[bmi])-1)
```

Now consider log creatinine as a predictor, again using the model displayed in lecture. Using a log transformation again in this context makes sense due to the right skewed distribution (e.g. check out a box plot), with some extreme values that could constitute outliers.

```
regress sbp lncreat age diabetes
```

The coefficient for `lncreat` has the interpretation of the increase in SBP associated with a 1 log increase in creatinine. This corresponds to an approximate 2.7 fold increase in creatinine. Here's a way to get the estimated change in average SBP associated with a 25% increase in creatinine:

```
nlcom _b[lncreat]*log(1.25)
```

This is generally more interpretable, and, as shown in lecture, the chosen % increase is easy to change based on your desired interpretation.