MSG500/MVE190
Linear Models - Lecture 10

Rebecka Jörnsten
Mathematical Statistics
University of Gothenburg/Chalmers University of Technology

November 27, 2012

1 Dummy variables, Polynomial regression and Interactions

In this lecture we will discuss how to include categorical covariates in our analysis. In addition, we will discuss models that include interactions or combinations of variables.

2 Categorical covariates

We have already encountered "dummy variables" in the Demos, e.g. when I used a 0/1 variable to encode the smoking status of a patient in the South African heart disease data set. The estimated coefficient for the smoking dummy variable in the backward selected model (Lecture 8) was approximately 0.15. Since the dummy variable only takes on two values: 0 and 1, we can interpret this coefficient as the excess ldl we expect to see among the smokers compared with the nonsmokers.

When our categorical variable (here smoking status) only takes on two values, the dummy variable is not much different than any other numerical variable in the model. Things get more complicated when we have more than one level: e.g. non-smoker, smokes at parties, smokes on weekends, 3 cigarettes a day, 1 pack a day, more than 1 pack a day. Here I hypothesize that we had a smoking variable with 6 levels. We can always encode such a variable with 6 dummy variables, one for each level. We thus estimate 6 coefficients, one for each level of smoking, and interpret the coefficient value as the excess ldl expected for this level of smoking. Note, we can't include 6 dummy variables if we also include an intercept in the model, since the combined role of all 6 levels is that of an intercept (the column sum of all the dummy variables is a vector of 1s). If we think of one of the levels of our categorial variables as "baseline" we can choose to let that level take the role of the intercept and include 5 dummy variables to encode all the other levels. We can make other restrictions in terms of contrasts between levels, but I refer you to the classes on experimental design for a deeper study of these options.

Note, in the above example the order of the levels of the categorical variable has a meaning. When this is the case, spend some time to explore the option of turning the variable into a numerical one. This saves a lot of parameters! In our above example, perhaps you need to keep a dummy variable for non-smokers/smoker but might be able to turn the levels of smoking into a numerical variable. This is something you have to look at on a case by case basis.

We will now turn to a data example: an anorexia data set consisting of weight data for 72 young women. We have a record of the women's weight prior to the study, `Prewt`, and after the study, `Postwt`. I have also added the weight gain = Post-Pre, `WtGain`, such that a positive number for this variable means that the women gained weight. Note, the weight is in *pounds* not kilograms. One pound is around 0.45 kilos, so these women are severely underweight (mean weight 37 kg). The study itself consisted of dividing the women into three groups; one control group, one group that underwent cognitive behavior treatment (where the women meet with a therapist), and a group that underwent family therapy (where the women's parents are instructed to interrupt the destructive behavior when they observe it. The groups are encoded 1, 2 and 3 respectively. (I was unable to retrieve information about the length of the study unfortunately.)

```
> anorexia <- read.table("anorexia.dat", header = T)
> boxplot(anorexia$WtGain ~ anorexia$Treat, names = c("Ctrl", "CBT",
+     "FT"))
> abline(h = 0, lty = 2)
```

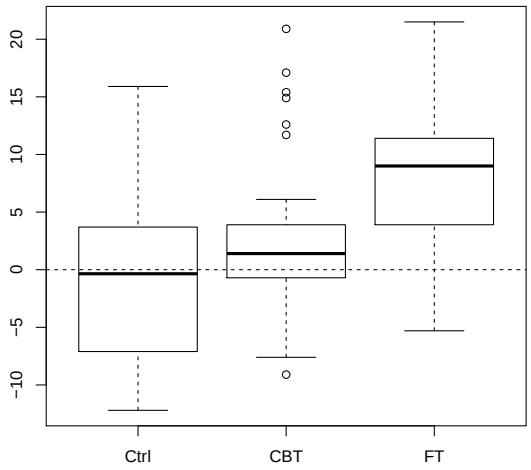

Figure 1: Weight gain as a function of treatment (Control, Cognitive behavior treatment, and Family treatment).

From Figure 1 you can see that the control group essentially gains no weight, whereas both treatment groups appear to experience some weight gain. Now, we could approach this problem as a (one-way) ANOVA (analysis of variance) - comparing the mean weight gain between the treatment groups.

```
> anovatest <- aov(WtGain ~ as.factor(Treat), data = anorexia)
```

The `as.factor()` is an R command that tells R that this variable should be interpreted as a categorical feature, not a numerical one. In Table 1 I show the outcome of the ANOVA test of the effect of treatment

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
as.factor(Treat)	2	614.64	307.32	5.42	0.0065
Residuals	69	3910.74	56.68		

Table 1: ANOVA summary

on the group weight gain. As you can see from the table, we reject the hypothesis that the mean weight gain is the same in all treatment groups. Strictly speaking, the underlying assumptions of the ANOVA test as executed above is equal sample variance in each treatment group and normally distributed errors. ANOVA is fairly robust to departures from these assumptions as long as the sample sizes for the groups aren't too different paired with unequal sample variance or skewed distributions. We can compare the outcome of a non-parametric test (Kruskal-Wallis):

```
> kruskal.test(anorexia$WtGain, as.factor(anorexia$Treat))

        Kruskal-Wallis rank sum test

data:  anorexia$WtGain and as.factor(anorexia$Treat)
Kruskal-Wallis chi-squared = 9.0475, df = 2, p-value = 0.01085
```

As you can see, the test still comes out significant in this case.

We will now approach this problem using regression instead. We first create dummy variables for each of the groups:

```
> dctrl <- rep(0, 72)
> dcbt <- rep(0, 72)
> dft <- rep(0, 72)
> dctrl[anorexia$Treat == 1] <- 1
> dcbt[anorexia$Treat == 2] <- 1
> dft[anorexia$Treat == 3] <- 1
> anorexia2 <- cbind(anorexia, dctrl, dcbt, dft)
> names(anorexia2) <- c(names(anorexia), "ctrl", "CBT", "FT")
> regmod <- lm(WtGain ~ CBT + FT, data = anorexia2)
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-0.4500	1.4764	-0.30	0.7614
CBT	3.4569	2.0333	1.70	0.0936
FT	7.7147	2.3482	3.29	0.0016

Table 2: Regression summary

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	69	3910.74				
2	71	4525.39	-2	-614.64	5.42	0.0065

Table 3: Regression summary

Note, in `regmod` I only include the two dummy variables for the treatments, letting control act as baseline (intercept). Tables 2 and 3 summarize the results. If you compare Tables 3 you find the F goodness-of-fit test and comparing this with Table 1 you see that the p-value for the ANOVA and goodness-of-fit F-tests are identical. Recall that the F goodness-of-fit test in regression tests if any of the variables (`dcbt, dft`) are related to weight gain (differently from `dctrl`) against the null that the intercept is sufficient (meaning all groups exhibit the same mean weight gain), exactly what ANOVA does.

The regression summary in Table 2 includes the t-tests for the individual parameters (CBT versus control, and FT versus control). We can access similar results from the ANOVA as well using post-hoc pairwise comparisons between means in the groups. In R you can use Tukey's Honest Significant Difference post-hoc adjustment:

```
> TukeyHSD(anovatest)

  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = WtGain ~ as.factor(Treat), data = anorexia)

$`as.factor(Treat)`
        diff       lwr       upr     p adj
2-1 3.456897 -1.413483  8.327276 0.2124428
3-1 7.714706  2.090124 13.339288 0.0045127
3-2 4.257809 -1.250554  9.766173 0.1607461
```

Note, in contrast to the above, the regression summary does not test the pair comparison CBT versus FT. Of course, you can do that in regression by using one of the treatments as baseline instead of the control, or specifically test that the coefficients of CBT and FT are equal, etc. Note that the Tukey test adjust for making three comparisons, whereas the regression coefficient t-tests do not (as we have

discussed previously). (Compare the regression p-values to the Tukey p-values. What if you adjust by a factor of 3 (Bonferroni correction) for the coefficients?)

We don't have to code up dummy variables manually as above since the `as.factor()` command also works with the regression commands:

```
> regmod <- lm(WtGain ~ as.factor(Treat), data = anorexia)
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-0.4500	1.4764	-0.30	0.7614
as.factor(Treat)2	3.4569	2.0333	1.70	0.0936
as.factor(Treat)3	7.7147	2.3482	3.29	0.0016

Table 4: Regression summary - factor command

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	69	3910.74				
2	71	4525.39	-2	-614.64	5.42	0.0065

Table 5: Regression summary - factor command

Comparing Tables 4 and 5 to Tables 2 and 3 you see that the results are identical.

3 Interactions with numerical variables

The anorexia data set also contains pre-study weight data.

```
> plot(anorexia$Prewt, anorexia$WtGain, pch = anorexia$Treat, col = anorexia$Treat)
> abline(h = 0, lty = 2)
> legend(91, 20, c("Ctrl", "CBT", "FT"), pch = c(1, 2, 3), col = c(1,
+     2, 3))
```

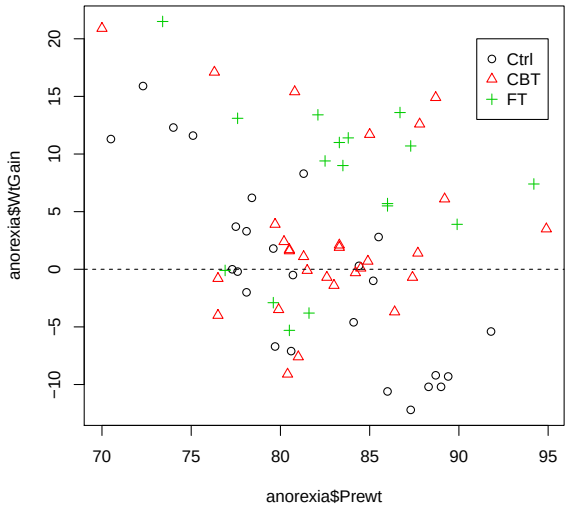

Figure 2: Weight gain as a function of pre-study weight and treatment (Control, Cognitive behavior treatment, and Family treatment).

In Figure 2 I depict the Weight gain as a function of the pre-study weight and the treatment. In the control group it seems clear that the lower pre-study weight individuals tend to gain more weight during the study, whereas the women with slightly higher pre-study weight even go on to lose some weight. In the treatment groups the pattern is less clear. The family treatment group appear to gain weight on average, and the amount of weight gain seems to be independent of the pre-study weight. For the CBT group, we appear to have one group of patients that don't lose or gain weight, and some patients who gain quite a lot of weight (these were the 'outliers' in the boxplot above).

Let us try some modeling. The first model I will use is a so-called *additive* model. An additive model assumes that both pre-study weight and treatment may affect weight gain. In addition, an additive model assumes that the relationship between pre-study weight and weight gain is the same for all treatment groups - the treatment merely changes the intercept of the model.

```
> regmod <- lm(WtGain ~ Prewt + as.factor(Treat), data = anorexia)
> plot(anorexia$Prewt, anorexia$WtGain, pch = anorexia$Treat, col = anorexia$Treat)
> abline(h = 0, lty = 2)
> legend(91, 20, c("Ctrl", "CBT", "FT"), pch = c(1, 2, 3), col = c(1,
+     2, 3))
> lines(c(30, 100), regmod$coef[1] + regmod$coef[2] * c(30, 100))
> lines(c(30, 100), regmod$coef[1] + regmod$coef[3] + regmod$coef[2] *
+     c(30, 100), col = 2)
> lines(c(30, 100), regmod$coef[1] + regmod$coef[4] + regmod$coef[2] *
+     c(30, 100), col = 3)
```

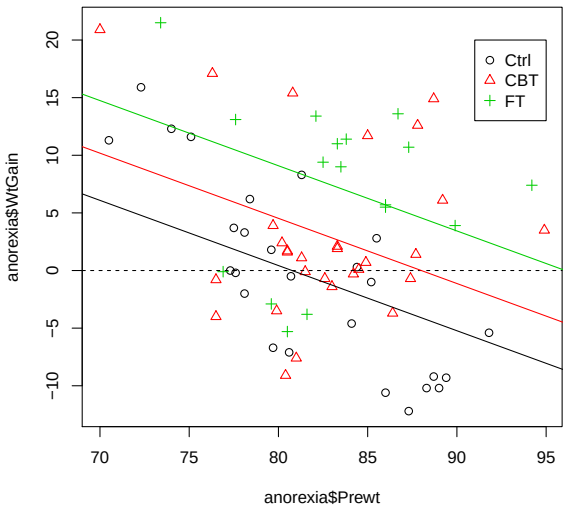

Figure 3: Additive model fit.

In Figure 3 I depict the additive model fit.

```
> xtable(summary(regmod), caption = "Additive model", label = "tab:addfit")
```

Tables 6 and 7 summarize the additive fit. As you can see, both the pre-study weight and the treatment have significant impact on the weight gain. However, we need to perform some diagnostics to assess the overall suitability of the model.

In Figure 4 we can clearly see that the error variance is larger in the two treatment groups compared with control. We will come back to this is a later lecture (Weighted Least Squares). For now, we will go ahead despite this problem. Do see see any trends in the residuals? In Figure 5 we compare look at the residuals more closely. There is some indication that there is a trend in the residuals for the control group. Now, the scatter plot above we saw a strong linear trend in the data for the control group but

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	45.6740	13.2167	3.46	0.0009
Prewt	-0.5655	0.1612	-3.51	0.0008
as.factor(Treat)2	4.0971	1.8935	2.16	0.0340
as.factor(Treat)3	8.6601	2.1931	3.95	0.0002

Table 6: Additive model

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	68	3311.26				
2	71	4525.39	-3	-1214.12	8.31	0.0001

Table 7: Additive model - F-test

not for the treatment groups. It is possible that because we forced this model to use the same slope for all treatment groups, the estimated slope is insufficient to capture the relationship between pre-study weight and the weight gain for the control group. We will address this using *interactions*. We will thus allow for a separate slope parameter for each treatment group, meaning that the relationship between pre-study weight and weight gain is different for the three treatment groups.

To fit interaction models to the data we create new variables that are products of dummy variables and the numerical variable. This gives as an adjustment to the slope between treatment groups.

```
> intctrl <- dctrl * anorexia$Prewt
> intcbt <- dcbt * anorexia$Prewt
> intft <- dft * anorexia$Prewt
> anorexia3 <- cbind(anorexia2, intctrl, intcbt, intft)
> names(anorexia3) <- c(names(anorexia2), "intctrl", "intCBT",
+     "intFT")
> regmod <- lm(WtGain ~ CBT + FT + intctrl + intCBT + intFT, data = anorexia3)
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	92.0515	18.8085	4.89	0.0000
CBT	-76.4742	28.3470	-2.70	0.0089
FT	-77.2317	33.1328	-2.33	0.0228
intctrl	-1.1342	0.2301	-4.93	0.0000
intCBT	-0.1520	0.2561	-0.59	0.5547
intFT	-0.0908	0.3272	-0.28	0.7823

Table 8: Regression summary - Interaction model

In Tables 8 and 9 I summarize the interaction model results. As you can see, the slope parameter is only significant for the control group. In the above, the intercept plays the role of the baseline control group. However, I fit a separate slope parameter to each group. An alternative model formulation is to let the control group slope be the baseline slope and fit the contrast slope parameters for the other groups. This is what the `*as.factor()` does as default (`*` is how R denotes interactions):

```
> regmodb <- lm(WtGain ~ as.factor(Treat) * Prewt, data = anorexia)
```

Compare the entries in Table 10 for the slope estimates to those of Table 8. The slope parameters for the CBT and FT groups in Table 8 are equal to `Prewt` slope plus the contrasts `as.factor(Treat)2:Prewt` and `as.factor(Treat)3:Prewt` in Table 10. Table 8 is good for analyzing the groups separately, whereas the model formulation of Table 10 is best for making group comparisons.

We plot the interaction model fits:

```
> plot(anorexia$Prewt, anorexia$WtGain, pch = anorexia$Treat, col = anorexia$Treat)
> abline(h = 0, lty = 2)
```

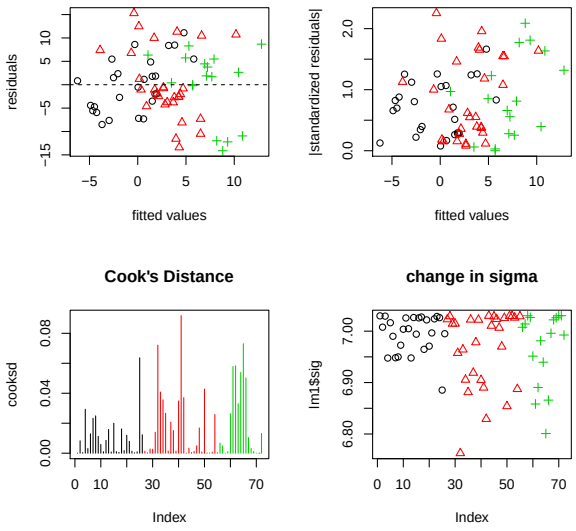

Figure 4: Diagnostics of additive model.

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	66	2844.78				
2	71	4525.39	-5	-1680.60	7.80	0.0000

Table 9: Regression summary - Interaction model

```
> legend(91, 20, c("Ctrl", "CBT", "FT"), pch = c(1, 2, 3), col = c(1,
+     2, 3))
> lines(c(30, 100), regmod$coef[1] + regmod$coef[4] * c(30, 100))
> lines(c(30, 100), regmod$coef[2] + regmod$coef[1] + regmod$coef[5] *
+     c(30, 100), col = 2)
> lines(c(30, 100), regmod$coef[3] + regmod$coef[1] + regmod$coef[6] *
+     c(30, 100), col = 3)
```

In Figure 6 we see that the slopes for the two treatment groups are almost 0, whereas the control group slope is negative. We look at the model diagnostics: In Figure 7 we see that the trends in the residuals are no longer present, but of course we still have the problem with higher error variance in the treatment groups. There are also at least two outliers present in the data (one in each treatment group). (Repeat the above analysis without the outliers - use code from previous lecture to help you identify them.)

I will repeat the analysis with a slight reformulation of my variables:

```
> anorexia4 <- anorexia3
> anorexia4$CBT <- anorexia3$CBT + anorexia3$FT
> anorexia4$intCBT <- anorexia3$intCBT + anorexia3$intFT
> names(anorexia4) <- c(names(anorexia), c("ctrl", "Therapy", "FTvCBT",
+     "intctrl", "intTherapy", "intFTvCBT"))
```

Here, I create a dummy variable `Therapy` that stands for either CBT or FT, and similarly for the interaction variable `intTherapy`. The variable `FTvCBT` is now a contrast between CBT and FT. The estimated coefficient for this variable will tell us about the expected excess weight gain using family therapy over cognitive behavior therapy. Similarly, `intFTvCBT` will tell us how the relationship between pre-study weight and weight gain differs between family therapy and CBT. This model formulation is good for testing both a) treatment effect compared with control and b) differences between treatments. We fit this model to the data:

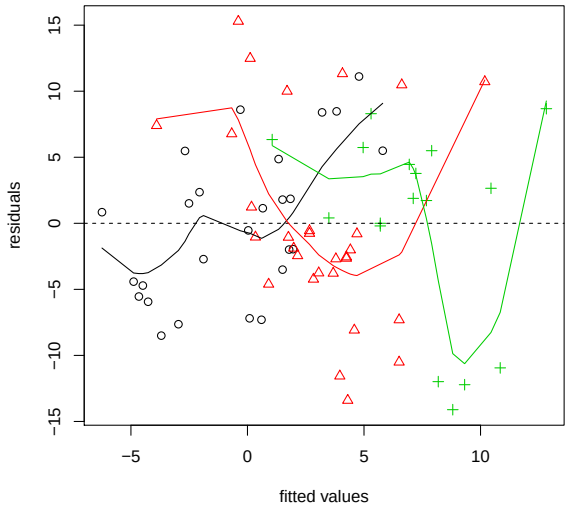

Figure 5: Residual plot -additive fit

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	92.0515	18.8085	4.89	0.0000
as.factor(Treat)2	-76.4742	28.3470	-2.70	0.0089
as.factor(Treat)3	-77.2317	33.1328	-2.33	0.0228
Prewt	-1.1342	0.2301	-4.93	0.0000
as.factor(Treat)2:Prewt	0.9822	0.3442	2.85	0.0058
as.factor(Treat)3:Prewt	1.0434	0.4000	2.61	0.0112

Table 10: Regression summary - Interaction model - contrasts

```
> regmodc <- lm(WtGain ~ Therapy + FTvCBT + intctrl + intTherapy +
+     intFTvCBT, data = anorexia4)
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	92.0515	18.8085	4.89	0.0000
Therapy	-76.4742	28.3470	-2.70	0.0089
FTvCBT	-0.7575	34.5516	-0.02	0.9826
intctrl	-1.1342	0.2301	-4.93	0.0000
intTherapy	-0.1520	0.2561	-0.59	0.5547
intFTvCBT	0.0612	0.4155	0.15	0.8833

Table 11: Regression summary - FT vs CBT

In Tables 11 and 12 this new model is summarized. As you can see, there is a Therapy effect on Weight Gain, but this effect is not different between the two treatment groups (p-value for `FTvCBT` is 0.98). The slope parameter is only significant for the control group.

4 Cautionary remarks

Let us try stepwise model selection on this last model:

```
> selectmodel <- step(regmodc, trace = F)
> summary(selectmodel)
```

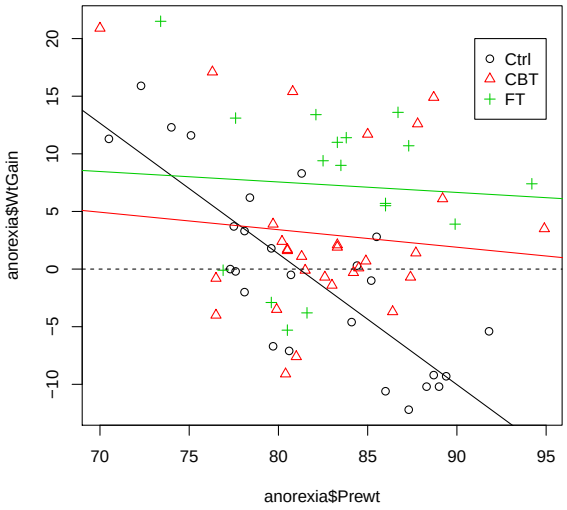

Figure 6: Interaction model fit

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	66	2844.78				
2	71	4525.39	-5	-1680.60	7.80	0.0000

Table 12: Regression summary - FT vs CBT

```
Call:
lm(formula = WtGain ~ Therapy + intctrl + intFTvCBT, data = anorexia4)

Residuals:
     Min       1Q   Median       3Q      Max
-12.3870  -3.7554  -0.9766   3.8224  17.8696

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  92.05147   18.60548   4.948 5.21e-06 ***
Therapy     -89.02102   18.64445  -4.775 9.96e-06 ***
intctrl      -1.13418    0.22759  -4.983 4.55e-06 ***
intFTvCBT     0.05039    0.02377   2.120   0.0377 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.494 on 68 degrees of freedom
Multiple R-squared: 0.3662,        Adjusted R-squared: 0.3383
F-statistic:  13.1 on 3 and 68 DF,  p-value: 7.6e-07
```

```
> plot(anorexia$Prewt, anorexia$WtGain, pch = anorexia$Treat, col = anorexia$Treat)
> abline(h = 0, lty = 2)
> legend(90, 20, c("Ctrl", "Therapy", "FTvCBT"), pch = c(1, 2,
+     3), col = c(1, 2, 3))
> lines(c(30, 100), selectmodel$coef[1] + selectmodel$coef[3] *
+     c(30, 100))
> lines(c(30, 100), selectmodel$coef[2] + selectmodel$coef[1] +
+     0 * c(30, 100), col = 2)
> lines(c(30, 100), selectmodel$coef[2] + selectmodel$coef[1] +
```

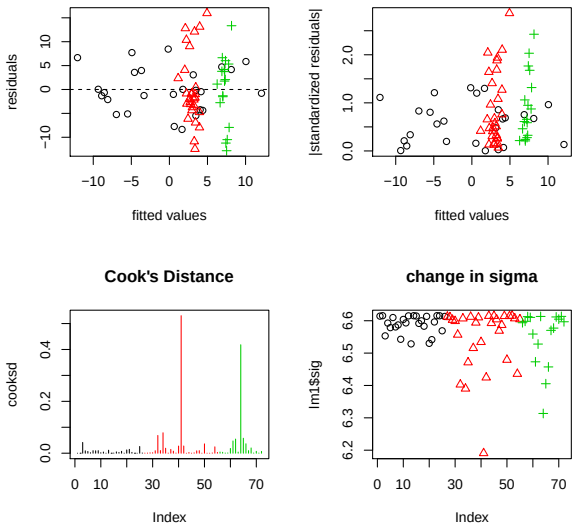

Figure 7: Diagnostics of interaction model.

```
+     selectmodel$coef[4] * c(30, 100), col = 3)
```

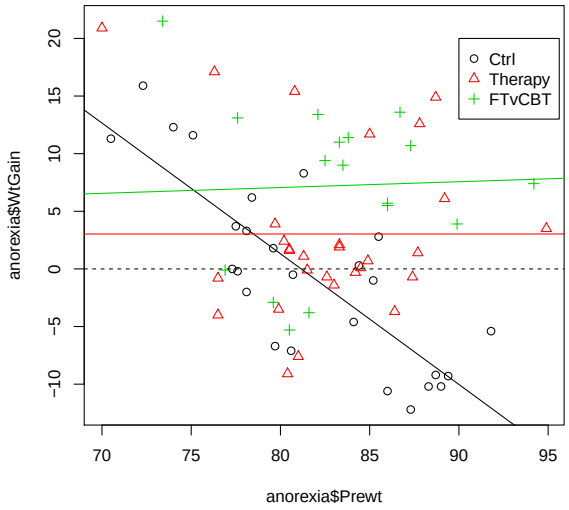

Figure 8: Backward selection model.

The backward selection includes; intercept (control group), therapy effect, control group's relationship between pre-study weight and weight gain and a separate slope for the family therapy group (see Figure 8).

What conclusions can we draw? For the control group, we see a significant dependency of weight gain on pre-study weight, where women whose weight was lower tended to gain more. For both therapy groups we saw an increased expected weight gain compared with the control group. In addition, for the the family therapy group there is some indication that there is a positive correlation between pre-study weight and weight gain. However, this is not significant if one adjust for multiple comparison.

Was the above selection step reasonable? We have to be very careful not to read too much into an interaction term when its so-called *main effect* is not present (here `FTvCBT`). First of all, the main effects variable and the interaction variable are often highly correlated, and especially if there is no strong relationship between the numerical variable and the outcome. We know that such collinearities often result in weird fits. It is common, in fact, common practise to only include interaction variables if their main effects are also in the model. Instead of automating the model selection, we therefore try to eliminate interaction terms first. If we proceed in this fashion, using backward F (first eliminating `intFTvCBT`, the `intTherapy` and finally trying to eliminate `FTvCBT` which is rejected at level $\alpha = 5\%$ (p-value 0.35), we arrive at the model:

```
> regmodd <- lm(WtGain ~ Therapy + FTvCBT + intctrl, data = anorexia4)
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	92.0515	18.5901	4.95	0.0000
Therapy	-89.0446	18.6291	-4.78	0.0000
FTvCBT	4.2578	1.9821	2.15	0.0353
intctrl	-1.1342	0.2274	-4.99	0.0000

Table 13: Regression summary - FT vs CBT - selected

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	68	2863.29				
2	71	4525.39	-3	-1662.09	13.16	0.0000

Table 14: Regression summary - FT vs CBT - selected

From Tables 13 and 14 we conclude that both treatments have an effect on weight gain. Note, the intercept for the CBT group is now obtained from the intercept (control) plus the contrast `Therapy` whereas the intercept for the family therapy group is obtained from the intercept (control) plus the `Therapy` effect plus the contrast`FTvsCBT`. The weight gain is therefore larger in the family group than the CBT. In addition, there is no relationship between pre-study weight and weight gain in either of the treatment groups. We summarize the fit in a graph:

```
> plot(anorexia$Prewt, anorexia$WtGain, pch = anorexia$Treat, col = anorexia$Treat)
> abline(h = 0, lty = 2)
> legend(91, 20, c("Ctrl", "CBT", "FT"), pch = c(1, 2, 3), col = c(1,
+     2, 3))
> lines(c(30, 100), regmodd$coef[1] + regmodd$coef[4] * c(30, 100))
> lines(c(30, 100), regmodd$coef[2] + regmodd$coef[1] + 0 * c(30,
+     100), col = 2)
> lines(c(30, 100), regmodd$coef[3] + regmodd$coef[2] + regmodd$coef[1] +
+     0 * c(30, 100), col = 3)
```

In Figure 9 I depict the model from Table 13.

5 Polynomial regression, Interactions between numerical variables

You can now generalize the above to other settings. For example, you can create interactions between numerical variables by including products of these variables in the model. Similarly, you can model second- or third-order models by including products of the variables themselves.

Creating product variables is easy enough, and in R you include them in your models as you did above the dummy variables or interaction variables. You can also use the `*` in the model formulation to include interactions between numerical variables. There are two things we have to discuss before concluding this segment of the lectures; (1) how can we detect the need for an interaction? and (2) how do we deal with the collinearities that might show up?

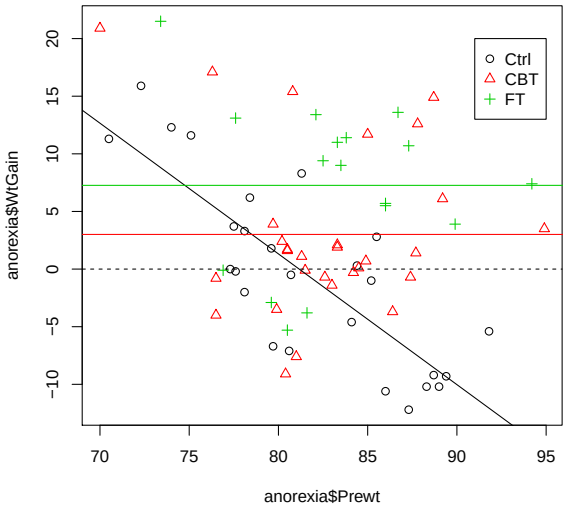

Figure 9: Final fitted model.

5.1 Detecting interactions

Above we saw that we can detect interactions between an outcome and numerical variable and a categorical variable by coloring the scatter plot using the levels of the categorical variable. If the colored points seem to capture a different x-y relationship (above: negative slope for control group, slope equal to zero for the treatment group), an interaction should be included in the model.

When we have two numerical variables we use so-called conditioning plots `coplot()`. Here is an example:

```
> x1 <- rnorm(mean = 2, 250)
> x2 <- rnorm(mean = 1, 250)
> x3 <- rnorm(mean = 1, 250)
> y <- 2 + 3 * x1 - 2 * x2 + 4 * x1 * x2 + rnorm(250, sd = 3)
```

I create a data set with an interaction term `x1*x2`. Let us first look at some pairwise plots:

```
> par(mfrow = c(1, 1))
> plot(x1, y)
```

```
> par(mfrow = c(1, 1))
> plot(x2, y)
```

In Figure 10 you see that both x-variables are related to y.

The conditional plots proceeds as follow: we pick one variable to condition on (below it's $x2$). We bin the data into groups corresponding to different ranges of $x2$ and plot a scatter plots of y versus $x2$ for each bin of data. If these scatter plots look the same for all bins, then we don't need the interaction $x1 * x2$ since the relationship between y and $x1$ is independent of $x2$, and vice versa.

```
> coplot(y ~ x1 | x2)
```

The panels of Figure 11 are read as follows: as you see there are 6 bins of $x2$. The smallest values for $x2$ are used to form the bin whose $y \sim x1$ plot is in the lower left panel. The top right panel is the scatter plots for $x1$ and y when $x2$ is in the largest value bin. You read the panels from left to right, bottom to top as corresponding to bins of increasing values of $x2$. If you compare the panels you see a clear alteration in the relationship between $x1$ and y. In fact, the relationship is stronger for large $x2$.

You can also compare multiple variables at a time:

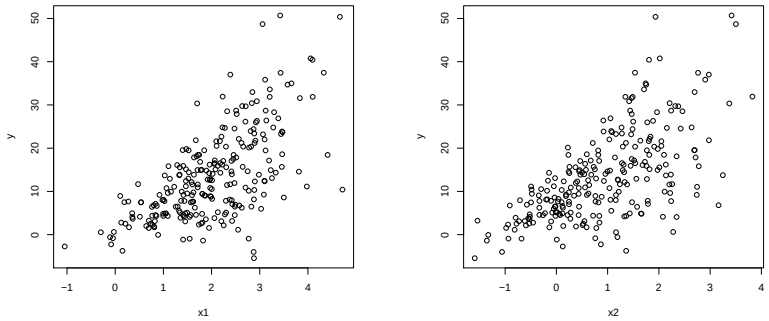

Figure 10: Scatter plots y vs x1 and y vs x2.

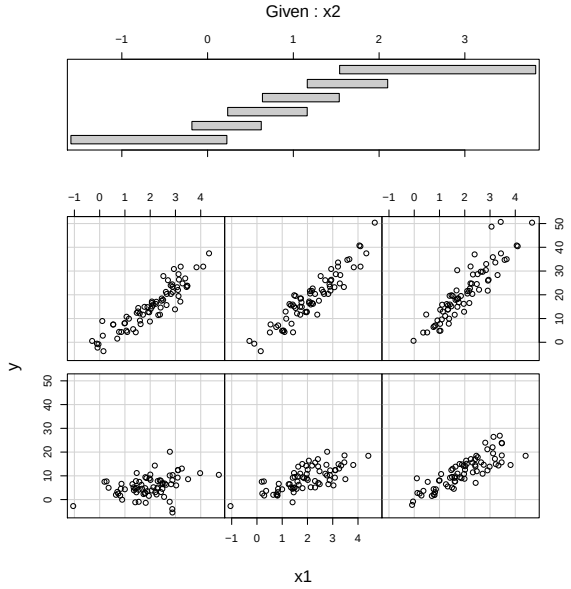

Figure 11: coplot of y on $x1$, conditioning on levels of $x2$.

```
> coplot(y ~ x1 | x2 * x3)
```

The panels of Figure 12 I condition on both $x2$ and $x3$. As you see, going from left to right, the relationship between $x1$ and y changes, meaning the levels of $x2$ matter (suggests interaction $x1 * x2$). Going from top to bottom, however, we see almost no change in the relationship, suggesting we don't need to include an interaction $x1 * x3$. If all the panels looked different this would suggest a model with a third-order interaction ($x1 * x2 * x3$).

Try at home: simulate some more models with or without interactions and use coplots to try to identify them.

5.2 Fitting models with interaction

As above, we can create interaction variables and use our standard least squares fit to estimate the parameters. There are two things to be wary of; (1) it is best to only include interaction terms if you also include the main effects; and (2) to reduce the collinearity problem, it is best to center all the variables first. Centering means that you subtract the mean from each variable prior to including them in the model, and prior to creating the interaction variables. To see why this is the case, compare the following:

```
> mod1 <- lm(y ~ x1 * x2)
```

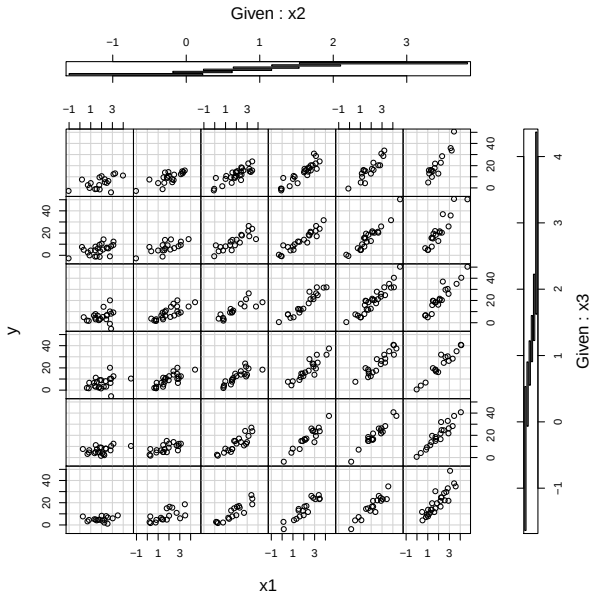

Figure 12: coplot of y on $x1$, conditioning on levels of $x2$ and $x3$.

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	1.8772	0.5875	3.20	0.0016
x1	3.0001	0.2659	11.28	0.0000
x2	-1.3050	0.4333	-3.01	0.0029
x1:x2	3.6455	0.1957	18.63	0.0000

Table 15: Regression summary - no centering

and

```
> mod2 <- lm(y ~ I(x1 - mean(x1)) * I(x2 - mean(x2)))
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	13.4216	0.1882	71.32	0.0000
I(x1 - mean(x1))	6.3701	0.1889	33.72	0.0000
I(x2 - mean(x2))	5.9921	0.1767	33.91	0.0000
I(x1 - mean(x1)):I(x2 - mean(x2))	3.6455	0.1957	18.63	0.0000

Table 16: Regression summary - centering

In Tables 15 and 16 I compare the regression summaries without and with centering, respectively. Note that the significance of the coefficients is improved after centering, demonstrating that the collinearity has been reduced by centering the data prior to modeling.

To guarantee that the main effects are included if the interactions are you have to be careful about using automated model selection schemes like `step()` above.

Try at home: play around with weaker/stronger interactions using simulated data. Try to detect interactions in some of the Demo data we have used.