

ST2304 Exercises Week 9: Interactions and the like

Bob O'Hara

7 March 2018

Interactions

We can use the cake example to look at interactions (bottom line: there aren't any). First, read in the data:

```
library(lme4) # use install.packages("lme4") if this doesn't work
```

```
## Loading required package: Matrix
```

```
data("cake")
cake$tempF <- factor(cake$temp)
```

The details of the experiment are here: <http://bit.ly/2GNb4dA>. Briefly, cakes were baked with different recipes and at different temperatures, and then tests for how well they held together by holding one end and lifting up the other until the cake broke. The angle at which the cake broke is the response. I assume that a lower angle is better (just think about a cake that can be bent by 63°). For practical reasons, cakes were baked on different days: these are referred to as replicates.

We are interested in the effects of temperature and recipe: replicate (i.e. date) has an effect (as I hope you all discovered in the last exercise) but we are not directly interested in this. But here we want to look to see if there is an interaction between temperature and recipe. So we want this model:

```
mod.int <- lm(angle~tempF*recipe + replicate, data=cake)
round(coef(mod.int), 2)
```

```
##      (Intercept)      tempF185      tempF195      tempF205
##      43.84         2.40         1.67         4.40
##      tempF215      tempF225      recipeB      recipeC
##      9.53         5.93         -2.27         -1.20
##      replicate2    replicate3    replicate4    replicate5
##      -1.28        -9.89        -13.56        -14.44
##      replicate6    replicate7    replicate8    replicate9
##      -18.11       -19.50       -19.44       -19.50
##      replicate10   replicate11   replicate12   replicate13
##      -18.00       -16.94       -15.89       -14.94
##      replicate14   replicate15   tempF185:recipeB tempF195:recipeB
##      -18.94       -20.22         0.13         3.20
## tempF205:recipeB tempF215:recipeB tempF225:recipeB tempF185:recipeC
##      0.87         -1.93         2.47         -1.40
## tempF195:recipeC tempF205:recipeC tempF215:recipeC tempF225:recipeC
##      2.13         -1.47         -3.07         1.87
```

There are a lot of parameters there! We only want to look at the ones involved in tempF and recipe. We can remove the replicate parameters (question: if we ignore those, the parameters are still valid, but for what data?). We can do this by finding the coefficients which have “replicate” in their name. We do that with the grepl() function: grep() is a function that searches for a string (or something more complicated called a regular expression) and returns the indices (i.e. numbers) where the string is present. grepl() also searches for a string, but returns a logical vector with TRUE when the string is present and FALSE when it is not. We can reverse the TRUE and FALSE with the ! (“not”) operator:

```

Coefs <- coef(mod.int) # get all coefficients
Coefs <- Coefs[!grepl("replicate", names(Coefs))]
CI <- confint(mod.int) # Get confidence intervals
CI <- CI[!grepl("replicate", rownames(CI)),]

```

Coefs

```

##      (Intercept)      tempF185      tempF195      tempF205
##      43.8444444      2.4000000      1.6666667      4.4000000
##      tempF215      tempF225      recipeB      recipeC
##      9.5333333      5.9333333      -2.2666667      -1.2000000
## tempF185:recipeB tempF195:recipeB tempF205:recipeB tempF215:recipeB
##      0.1333333      3.2000000      0.8666667      -1.9333333
## tempF225:recipeB tempF185:recipeC tempF195:recipeC tempF205:recipeC
##      2.4666667      -1.4000000      2.1333333      -1.4666667
## tempF215:recipeC tempF225:recipeC
##      -3.0666667      1.8666667

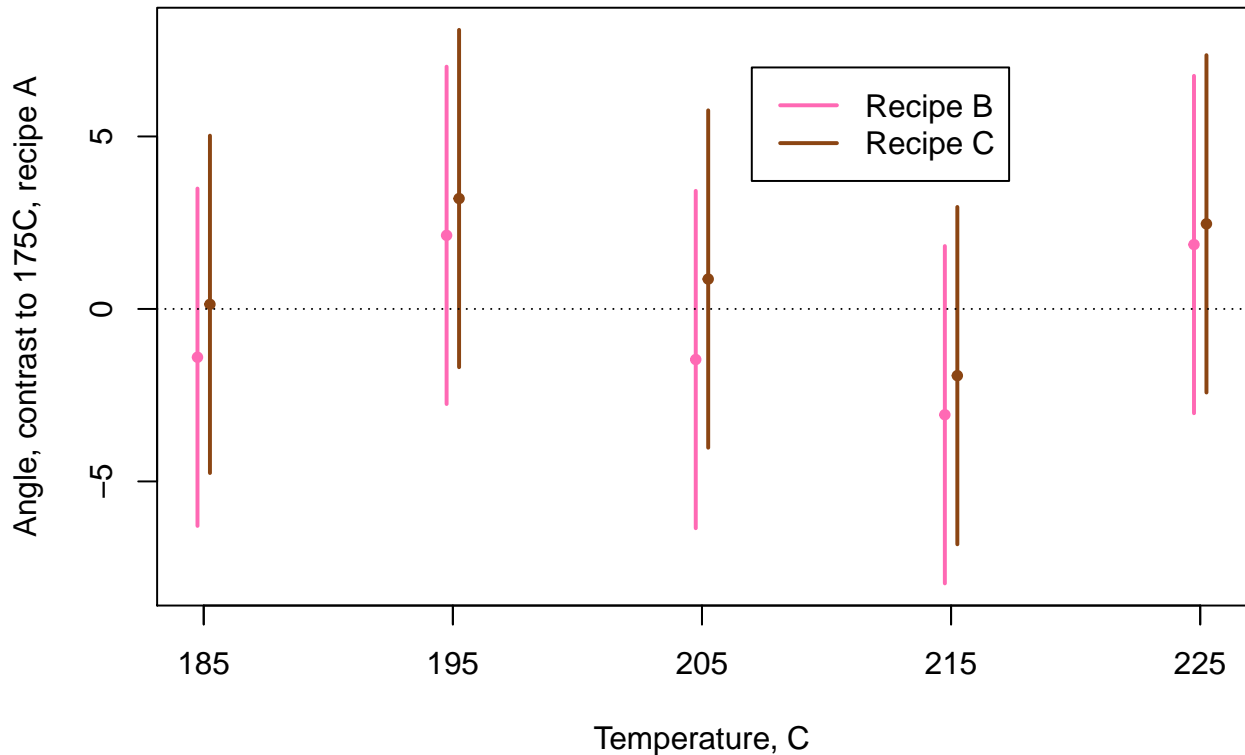
```

We can then plot the interactions (letting R do the work of sorting out the data: this coding will not be on the exam!):

```

Coefs.int <- Coefs[grepl(":", names(Coefs))] # only interactions have a :
CI.int <- CI[grepl(":", rownames(CI)),]
Col.int <- c("hotpink", "saddlebrown")[1+grepl("recipeB", names(Coefs.int))]
At.temp <- gsub(":recipe.", "", gsub("tempF", "", names(Coefs.int)))
At.int <- as.numeric(At.temp) + (0.5*grepl("recipeB", names(Coefs.int)))-0.25
plot(At.int, Coefs.int, col=Col.int, pch=20, ylim=range(CI.int),
     xaxt="n", xlab="Temperature, C",
     ylab="Angle, contrast to 175C, recipe A")
segments(At.int, CI.int[,1], At.int, CI.int[,2], col=Col.int, lwd=2)
axis(1, at=as.numeric(At.temp))
abline(h=0, lty=3)
legend(207, 7, c("Recipe B", "Recipe C"), col=c("hotpink", "saddlebrown"),
      lty=1, lwd=2)

```



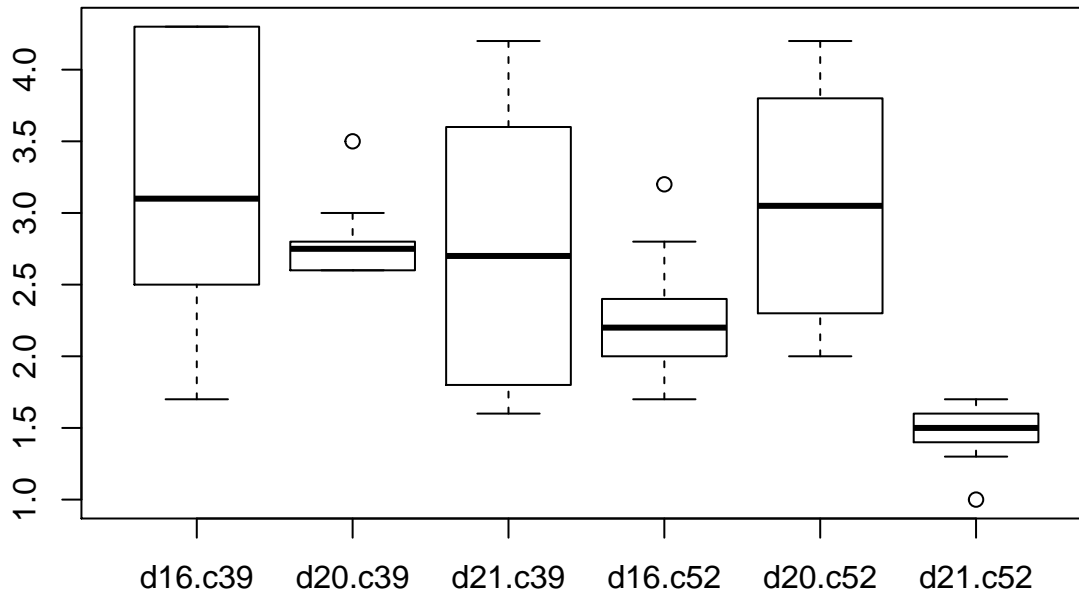
We can see that all of the confidence intervals overlap 0, which might suggest that there is no interaction, i.e. any difference between recipes A and B, and between recipes A and C do not vary with temperature. But an alternative explanation is that there are large effects, but we can't estimate them with enough precision to see them in the data.

How can we tell the difference? The problem is whether the model is estimating effects which are large. Deciding that is ultimately not a statistical issue: it depends on the biology (or in this case the cooking!). But we can get some idea by looking at what else the model is estimating. The confidence intervals for the interaction go from -8° to 8.1° , so it is unlikely that any interaction effects are beyond these. Thus, the question is whether these values (which are probably over-estimates of the largest effects) could be considered "big". If you don't have intimate knowledge of cake bending, then you can look at the effects of other covariates, e.g. the replicate contrasts go down to -20, so any effect of the interaction is modest compared to this (to be fair, all effects are modest compared to this!). We can also look at the unexplained variation: the residual standard deviation is `round(summary(mod.int)$sigma,1)`, so the confidence limits is a bit bigger than this, in other words if an effect was at the range of likely values, it would change the angle by less than one and a half residual standard deviations. This is probably not a big effect, and the most likely values (i.e. the estimates) are much smaller than this. Thus, at best the effects are moderate, and are probably much smaller, and thus unlikely to be a major influence.

Problem 1: Interactions

The data comes from an experiment on cabbages, and the effects of data of planting and cultivar (i.e. genotype) on the yield (weight and vitamin C concentration). You can get the data and take a look with this:

```
library(MASS)
data("cabbages")
boxplot(HeadWt~Date*Cult, data=cabbages)
```



The problems:

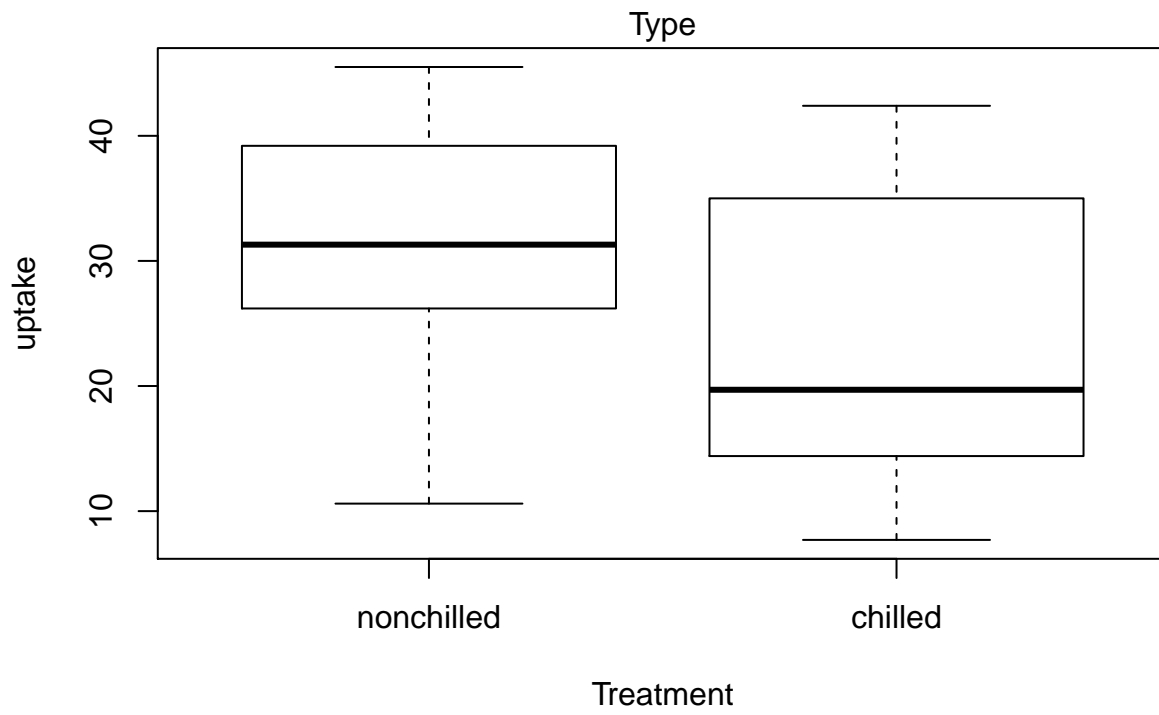
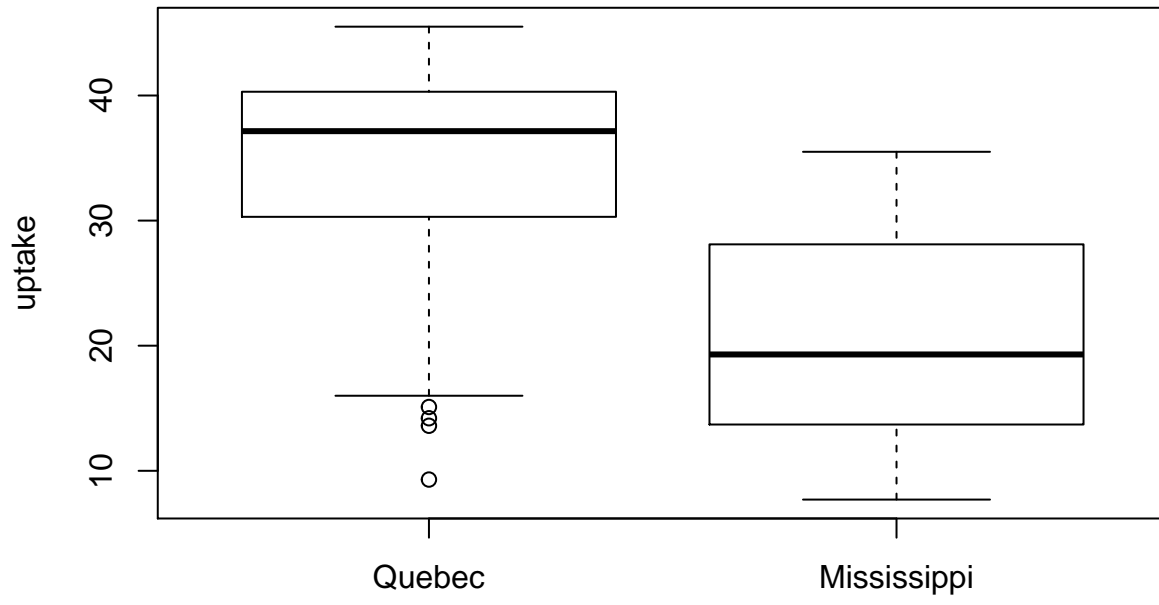
1. are there effects of cultivar ('Cult') and planting date ('Date') on yield ('HeadWt')? Does the effect of cultivar change depending on the planting date?
2. are there effects of cultivar and planting date on vitamin C concentration('VitC')? Does the effect of cultivar change depending on the planting date? For each of these, summarise how well the model explains the data and if there are any problems with the model fit.

Problem 2: CO₂ uptake in a grass

The C02 data is from an experiment on cold tolerance in a C4 grass. Plants were taken from two geographic locations (Mississippi and Quebec), and grown in different CO₂ concentrations: some were also chilled before growth. The experimenters were interested in looking at the effect of chilling on CO₂ uptake, and whether this varied depending on the population (something which might reflect local adaptation, as Mississippi doesn't really have a winter).

1. Fit a model with the source population("Type") and treatment ("Treat") as effects. Do they have an effect, and do their effects interact? If so, how? How well does this model explain the data?
2. Now add log-transformed CO₂ concentration to the model, just as a main effect. Does this change your conclusions from the first model?
3. Optional: If you are feeling brave, look to see if there is an effect of the interactions between log(concentration) and the other variables. Can you interpret what is going on?

```
data(C02)
plot(uptake~Type*Treatment, data=C02)
```



Ana