In-class Discussion and Problem Set (after lecture 1)

LSA 2013, LI539

Mixed Effect Models

T. Florian Jaeger

Brain and Cognitive Sciences
University of Rochester

July 11, 2013

Getting Help

- Subscribe to ling-R-lang: `https://mailman.ucsd.edu/mailman/listinfo/ling-r-lang-l`
- In R: try `?foo` or `help(foo)` first
- Ask your neighbor (during problem sets) or the class.

The linear predictor

- Recall that for the **LM**:

$$\eta = I(\mu) = E(\mathbf{y}) = \alpha + \beta_1 \mathbf{x}_1 + \cdots + \beta_n \mathbf{x}_n$$

...and for **GLM**, more generally, η is a linear combination of the predictors $\{\mathbf{x_i}\}$:

$$\eta = \alpha + \beta_1 \mathbf{x_1} + \cdots + \beta_n \mathbf{x_n} \qquad \text{(linear predictor)}$$

The linearity assumption

$$\boxed{\mathbf{y} = \overbrace{\beta_0 \mathbf{x_0} + \ldots + \beta_n \mathbf{x_n}}^{\text{Predicted Mean}} + \overbrace{\epsilon}^{\text{Noise}\sim N(0,\sigma_\epsilon)}}$$

Questions

- What does the linearity assumption made by **AN(C)OVA** and the **LM** mean?

The linearity assumption

$$\boxed{\mathbf{y} = \overbrace{\beta_0 \mathbf{x_0} + \ldots + \beta_n \mathbf{x_n}}^{\text{Predicted Mean}} + \overbrace{\epsilon}^{\text{Noise} \sim N(0,\sigma_\epsilon)}}$$

Questions

- What does the linearity assumption made by **AN(C)OVA** and the **LM** mean?
- When we use these models, do we assume that the output is linear in the input variables (i.e., that the input variables are linearly related to the outcome)? Do we assume that the outcome is linear in the parameter estimates?

The linearity assumption

$$\mathbf{y} = \overbrace{\beta_0 \mathbf{x_0} + \ldots + \beta_n \mathbf{x_n}}^{\text{Predicted Mean}} + \overbrace{\epsilon}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

Questions

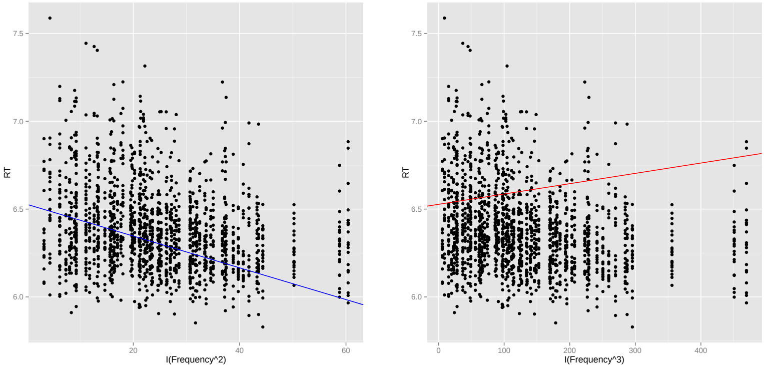
- Are the following, well-formed linear models?

$$\begin{aligned}
\mathbf{RT} &= \alpha \\
\mathbf{RT} &= \alpha + \beta_1 \mathbf{Frequency} \\
\mathbf{RT} &= \beta_1 \mathbf{Frequency} \\
\mathbf{RT} &= \beta_1 \log_{10} \mathbf{Frequency} \\
\mathbf{RT} &= \alpha + \beta_1 \mathbf{Frequency}^2 + \beta_2 \mathbf{Frequency}^3 \\
\mathbf{RT} &= \alpha + \beta_1 \mathbf{Frequency} * \mathbf{DerivEntropy} \\
\mathbf{RT} &= \alpha + \beta_1 \mathbf{Frequency}^{\beta_2} \\
\mathbf{RT} &= \frac{\beta_1}{\mathbf{Frequency}}^{2/3}
\end{aligned}$$

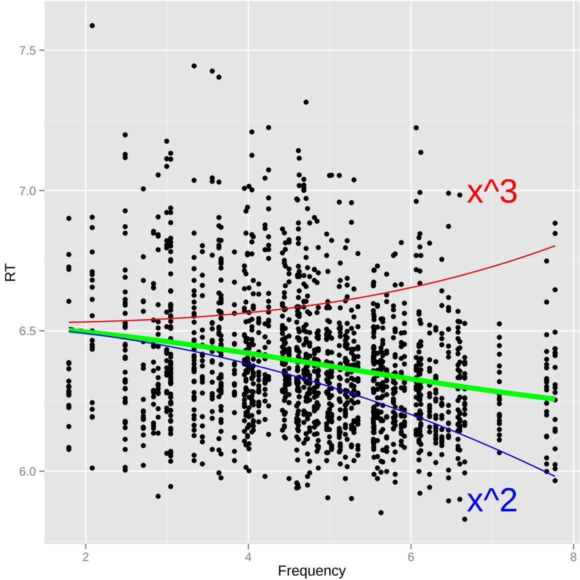
Visualization

```
library(languageR)
data(lexdec)

g = glm(RT ~ I(Frequency^2) + I(Frequency^3), data=lexdec)
summary(g)$coefficients

##                    Estimate  Std. Error t value Pr(>|t|)
## (Intercept)      6.5271408  0.0259644 251.388 0.000000
## I(Frequency^2) -0.0090349  0.0031069  -2.908 0.003686
## I(Frequency^3)  0.0005876  0.0003969   1.480 0.138936


library(ggplot2)
ggplot(lexdec, aes(x=I(Frequency^2), y=RT)) +
  geom_point() +
  geom_abline(
    intercept = coef(g)['(Intercept)'],
    slope = coef(g)['I(Frequency^2)'],
    color = "blue")

ggplot(lexdec, aes(x=I(Frequency^3), y=RT)) +
  geom_point() +
  geom_abline(
    intercept = coef(g)['(Intercept)'],
    slope = coef(g)['I(Frequency^3)'],
    color = "red")
```

Linear in the coefficients (and transformed predictors)



Non-linear in the input variables



Centering a continuous variable

Task

- Analyze the effects of `Frequency` and `Trial` on `RT`s in the `lexdec` data set.

```
summary(glm(RT ~ Frequency + Trial, data= lexdec))
```

- Then center both predictors (i.e., remove their mean out of them). Save these new predictors under a new name in `lexdec`, e.g.

```
lexdec$cFrequency = lexdec$Frequency - mean(lexdec$Frequency)
summary(lexdec[,c('Frequency','cFrequency')])

##    Frequency       cFrequency
##  Min.   :1.79   Min.   :-2.9594
##  1st Qu.:3.95   1st Qu.:-0.7999
##  Median :4.75   Median : 0.0025
##  Mean   :4.75   Mean   : 0.0000
##  3rd Qu.:5.65   3rd Qu.: 0.9014
##  Max.   :7.77   Max.   : 3.0208
```

- Rerun the model with the new centered predictors. What changes and what doesn't change?

Centering a continuous variable

Task

- Now, let's do the same when there's also an interaction in the model:

```
summary(glm(RT ~ Frequency * Trial, data= lexdec))$coefficients

##                       Estimate Std. Error t value  Pr(>|t|)
## (Intercept)          6.6761129  5.403e-02 123.568 0.000e+00
## Frequency           -0.0544937  1.103e-02  -4.939 8.635e-07
## Trial               -0.0008211  4.609e-04  -1.781 7.502e-02
## Frequency:Trial      0.0001084  9.416e-05   1.152 2.496e-01
```

- Compare the models with uncentered and centered predictors.

```
# Beyond the coefficients, another interesting thing to compare is
# the models predictions (the predicted RTs)
g1 =  glm(RT ~ Frequency * Trial, data= lexdec)
g2 =  glm(RT ~ cFrequency * cTrial, data= lexdec)

# if the models make the same prediction the difference in their
# predictions should be zero
predict(g1) - predict(g2)
```

Treatment-coded 2x2

Task

- Analyze the effects of `NativeLanguage`, `Sex`, and their interaction on `RT`s in the `lexdec` data set.

```
summary(glm(RT ~ NativeLanguage * Sex, data= lexdec))$coefficients
```

- Now change the ordering of the levels for `NativeLanguage`, so the model compares 'native' against 'non-natives', rather than the other way around. Rerun the model and compare.

```
# by default R assumes alphanumeric ordering of factor levels
# but we can always change that by explicitly stating what
# order we would like:
lexdec$NativeLanguage =
  factor(lexdec$NativeLanguage, levels=c('Other','English'))

# Btw, c() is simply referring to a vector in R. They can
# contain numbers, strings, or any (combination) of other
# objects, e.g. c(4, "seven", "Hans the Wurst", c(log(2), "hai"))
```

- Let's go through what just happened.

ANOVA-coded 2x2

Task

- Now, use sum/effect/ANOVA/contrast-coding for the two predictors (`NativeLanguage` and `Sex`) and repeat the analysis

```
# confirm that the default contrast was treatment-coding, e.g.
contrasts(lexdec$Sex)

##   M
## F 0
## M 1


# change contrasts
contrasts(lexdec$Sex) = contr.sum(2)
contrasts(lexdec$Sex)

##   [,1]
## F    1
## M   -1

# notice that default sum-coding in R assigns 1 to the
# alphanumerically first level and -1 to the second level

# let's label the contrast so that we don't get confused
# (cbind() makes a matrix out of multiple vectors; in this
# case 1 vector, but it allows us to label the columns/vectors)
contrasts(lexdec$Sex) = cbind("Male" = c(-1, 1))
```

ANOVA-coded 2x2

Task

- Compare the output of a 2x2 GLM under a) default treatment-coding, b) default sum-coding, c) halved sum-coding:

```
contrasts(lexdec$Sex) = contr.sum(2) / 2
contrasts(lexdec$Sex)

##    [,1]
## F   0.5
## M -0.5


contrasts(lexdec$NativeLanguage) = contr.sum(2) / 2
contrasts(lexdec$NativeLanguage)

##          [,1]
## Other     0.5
## English -0.5
```