basic generalized linear models

Ben Bolker

Linear models

- foundation for (G)LM(M)s, other complex models
- flexible, robust, computationally efficient, standard
- includes (multiple) regression, ANOVA, ANCOVA, ...
- natural ways to express dependence, interactions

Linear models: assumptions

- response variables:
  - Gaussian (normally distributed)
  - independent
  - *conditionally* homoscedastic (equal variance)
  - univariate
- predictor variables
  - numeric or categorical (nominal)

Linear models: math

$$z = a + bx + cy + \epsilon$$

or (more predictor variables)

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \epsilon$$

or (more flexible distribution syntax)

$$y \sim \text{Normal}(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots, \sigma^2)$$

or (more complex sets of predictors)

$$\begin{aligned} \boldsymbol{\mu} &= \mathbf{Xfi} \\ y_i &\sim \text{Normal}(\mu_i, \sigma^2) \end{aligned}$$

what does "linear" mean?

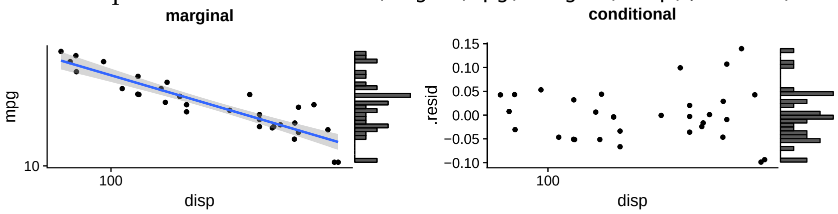
- y is a linear function of the *parameters* ($\partial^2 y / \partial^2 \beta_i = 0$)
- e.g. polynomials: $y = a + bx + cx^2 + dx^3$
- or sinusoids: $y = a \sin(x) + b \cos(x)$
- but **not**: power-law (ax^b), exponential ($a \exp(-bx)$)

marginal vs. conditional distributions

- common mistake: worry about the overall distribution of the response,
  rather than the *conditional* distribution (i.e., residuals)
- if only categorical predictors, can mean-correct each group, then look at residuals
- otherwise have to fit the model first!

example

MPG vs displacement for cars: `lm(log10(mpg)~log10(disp),mtcars)`



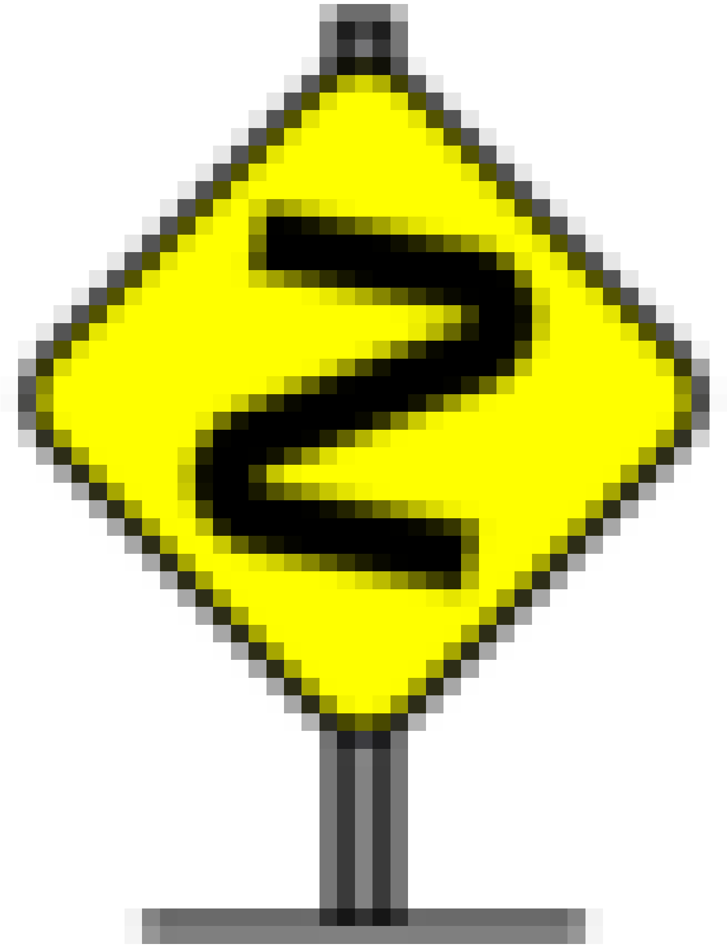
categorical predictors

- how do categorical predictors fit into this scheme?
- *dummy variables*: convert to 0/1 values
- R does this automatically with formula syntax
- e.g. for two levels:

```
dd <- data.frame(flavour = rep(c("chocolate",
    "vanilla"), c(2, 3)))
model.matrix(~flavour, dd)

##   (Intercept) flavourvanilla
## 1           1              0
## 2           1              0
## 3           1              1
## 4           1              1
## 5           1              1
## attr(,"assign")
## [1] 0 1
## attr(,"contrasts")
## attr(,"contrasts")$flavour
## [1] "contr.treatment"
```

- first alphabetical level (`chocolate`) used as default



- *ordered factors* are handled differently

R formulas

- "Wilkinson-Rogers" notation
- `response ~ predictor1 + predictor2 + ...`
- numeric variables used "as is"
- categorical variables (factors) converted to dummy variables
- intercept added automatically (`1+ ...`)
- interaction: `:` *multiplies* relevant columns
- `a*b`: main effect plus interactions
- `model.matrix(formula, data)`

Formulas, continued

- `y~f`: 1-way ANOVA

- `y~f+g`: 2-way ANOVA (additive)
- `y~f*g`: 2-way ANOVA (with interaction)
- `y~x`: univariate regression
- `y~f+x`: ANCOVA (parallel slopes)
- `y~f*x`: ANCOVA (with interaction, non-parallel slopes)
- `y~x1+x2`: multivariate regression (additive)
- `y~x1*x2`: multiv. regression with interaction

If confused, (1) try to write out the equation; (2) `model.matrix()`

Contrasts

- Machinery for translating categorical variables to dummy (0/1) variables
- **treatment** contrasts (default):
  - β_1 = intercept = expected value of first level (by default, "aardvark")
  - β_i = difference between level $i+1$ and baseline
- **sum-to-zero** contrasts:
  - β_1 = intercept = unweighted mean of all levels
  - β_i = difference between level i and mean; last level not included (!)

too many ways to change contrasts (globally via `options()`; as attribute of factor; `contrasts` argument in `lm()`)

Example 1 (treatment contrasts)

Data on ant colonies from Gotelli and Ellison (2004):

```
ants <- data.frame(place = rep(c("field", "forest"),
    c(6, 4)), colonies = c(12, 9, 12, 10, 9, 6,
    4, 6, 7, 10))
aggregate(colonies ~ place, data = ants, FUN = mean)
```

```
##    place colonies
## 1  field 9.666667
## 2 forest 6.750000
```

```
pr <- function(m) printCoefmat(coef(summary(m)),
    digits = 3, signif.stars = FALSE)
pr(lm1 <- lm(colonies ~ place, data = ants))
```

```
##             Estimate Std. Error t value
## (Intercept)    9.667      0.958   10.09
## placeforest   -2.917      1.515   -1.92
##             Pr(>|t|)
## (Intercept)    8e-06
## placeforest     0.09
```

Ants: change to sum-to-zero contrasts

```
pr(lm2 <- update(lm1, contrasts = list(place = contr.sum)))
```

```
##             Estimate Std. Error t value
## (Intercept)    8.208      0.758   10.83
## place1         1.458      0.758    1.92
##             Pr(>|t|)
## (Intercept)  4.7e-06
## place1          0.09
```

```
data(lizards, package = "brglm")
```

Scaling and centering

- Schielzeth

LM diagnostics

- fitted vs. residual: pattern in mean?
- scale-location: pattern in variance?
- Q-Q plot: Normal distribution?
- leverage/Cook's distance: influential points?

Diagnostics

- statisticians: "don't use p-values to evaluate LM assumptions"
- everyone else: "so what should I do?"
- statisticians: "look at pictures"
- everyone else: "how do I decide whether to worry?"
- statisticians: "…"

From LM to GLM

- assumptions of LMs do break down sometimes
- count data: discrete, non-negative
- proportion data: discrete counts, $0 \leq x \leq N$

  https://twitter.com/thedavidpowell/status/984432764215754753

GLMs in action

- vast majority of GLMs
  - *logistic regression* (binary/Bernoulli data)
  - *Poisson regression* (count data)

link functions

- transform *prediction*, not response
- e.g. rather than $\log(\mu) = \beta_0 + \beta_1 x$, use $\mu = \exp(\beta_0 + \beta_1 x)$
- in this case log is the **link function**, exp is the **inverse link** function

families

- link function plus variance function
- typical defaults
  - Poisson: log (exponential)
  - binomial: logit/log-odds (logistic)

logit link

estimation basics

- iteratively re-weighted least-squares

testing hypotheses and interpreting results

- something here

References

Gotelli, Nicholas J., and Aaron M. Ellison. 2004. *A Primer of Ecological Statistics*. Sunderland, MA: Sinauer.