Multivariate Statistics in Ecology and Quantitative Genetics
Generalized Linear Models (GLMs)

Dirk Metzler & Noémie Becker

http://evol.bio.lmu.de/_statgen

25. Juni 2014

Contents

Contents

Contents

```
> daph <- read.table("daphnia_justina.csv",h=T)
> mod1 <- lm(counts~foodlevel+species,data=daph)
> mod2 <- lm(counts~foodlevel*species,data=daph)
> anova(mod1,mod2)
Analysis of Variance Table

Model 1: counts ~ foodlevel + species
Model 2: counts ~ foodlevel * species
  Res.Df    RSS Df Sum of Sq      F   Pr(>F)
1      9 710.00
2      8 176.67  1    533.33 24.151 0.001172 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
>
```

```
> daph
   counts foodlevel species
1      68      high   magna
2      54      high   magna
3      59      high   magna
4      24      high galeata
5      27      high galeata
6      16      high galeata
7      20       low   magna
8      18       low   magna
9      18       low   magna
10      5       low galeata
11      8       low galeata
12      9       low galeata
```

The normal distribution $\mathcal{N}(\mu, \sigma^2)$ is a continuous distribution and thus not suitable to model distributions on small numbers.

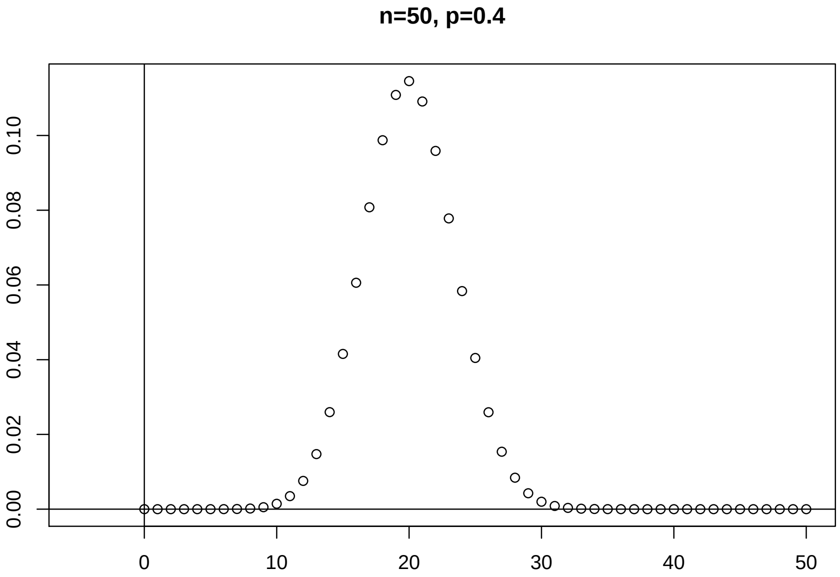
The normal distribution $\mathcal{N}(\mu, \sigma^2)$ is a continuous distribution and thus not suitable to model distributions on small numbers.

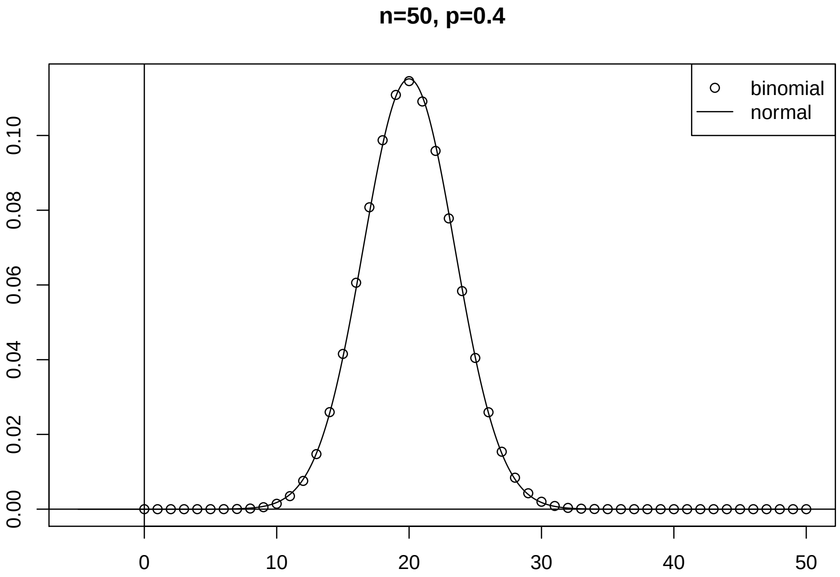
The Poisson distribution Pois(λ) is a distribution on $\{0, 1, 2, 3, \dots\}$.

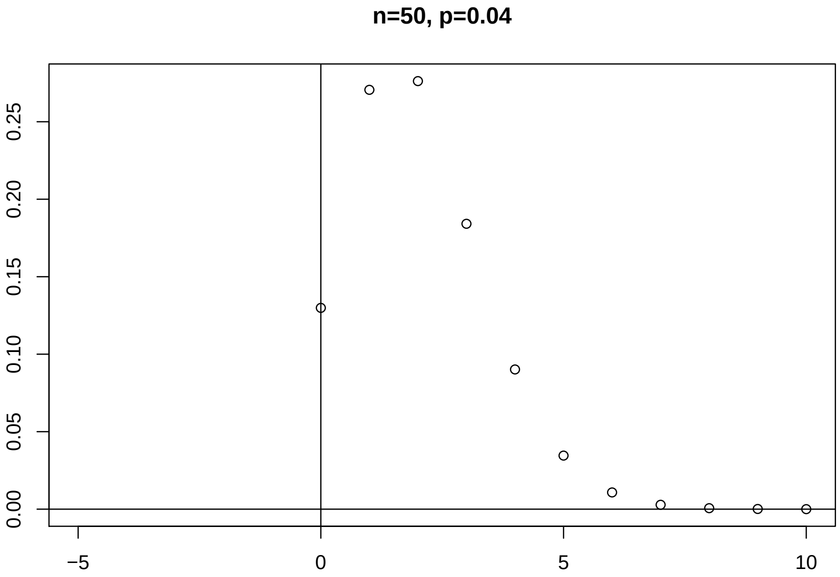
The normal distribution $\mathcal{N}(\mu, \sigma^2)$ is a continuous distribution and thus not suitable to model distributions on small numbers.

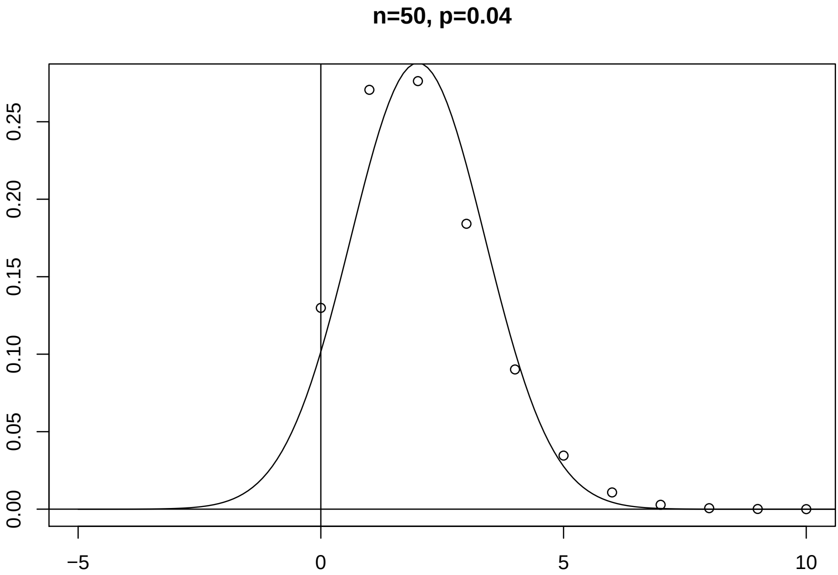
The Poisson distribution Pois(λ) is a distribution on $\{0, 1, 2, 3, \dots\}$.

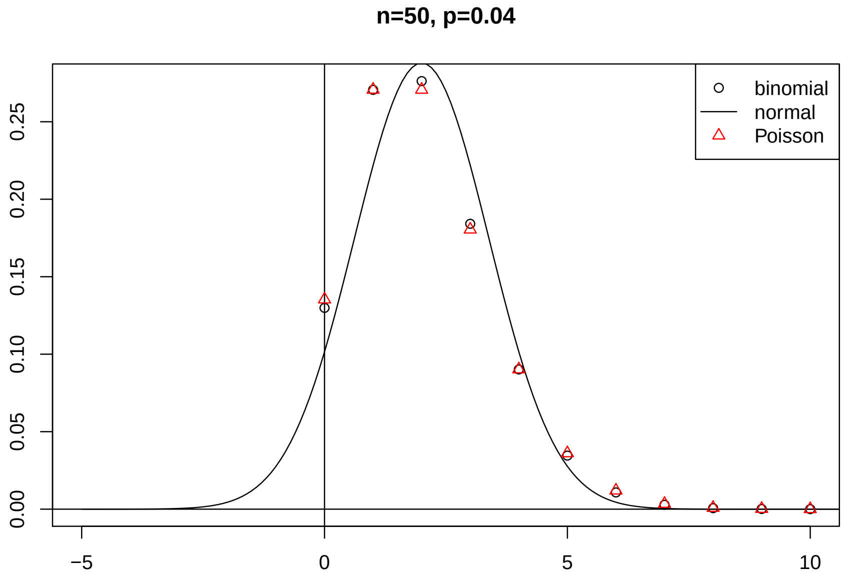
$\mathcal{N}(\mu = n \cdot p, \sigma^2 = n \cdot p \cdot (1 - p))$ approximates the binomial distribution Bin(n,p) if $n \cdot p \cdot (1 - p)$ is not too small (rule of thumb: $n \cdot p \cdot (1 - p) > 9$), Pois($\lambda = n \cdot p$) gives a better approximation when p is small.


n=50, p=0.4
0.00
0.02
0.04
0.06
0.08
0.10
0
10
20
30
40
50


n=50, p=0.4
binomial
normal
0.00
0.02
0.04
0.06
0.08
0.10
0
10
20
30
40
50


n=50, p=0.04
0.00
0.05
0.10
0.15
0.20
0.25
−5
0
5
10


n=50, p=0.04
0.00
0.05
0.10
0.15
0.20
0.25
−5
0
5
10


n=50, p=0.04
binomial
normal
Poisson
0.00
0.05
0.10
0.15
0.20
0.25
−5
0
5
10

If Y is Pois(λ)-distributed, then

$$\begin{aligned} \Pr(Y = k) &= \frac{\lambda^k}{k!} \cdot e^{-\lambda} \qquad \text{for } k = 0, 1, 2, \ldots \\ \mathbb{E}Y &= \lambda \\ \mathrm{Var}(Y) &= \lambda \end{aligned}$$

If Y is Pois(λ)-distributed, then

$$\begin{aligned} \Pr(Y = k) &= \frac{\lambda^k}{k!} \cdot e^{-\lambda} \qquad \text{for } k = 0, 1, 2, \ldots \\ \mathbb{E}Y &= \lambda \\ \mathrm{Var}(Y) &= \lambda \end{aligned}$$

Is there a linear model with Pois(λ) instead of $\mathcal{N}(\mu, \sigma^2)$?

If Y is Pois(λ)-distributed, then

$$\begin{aligned} \Pr(Y=k) &= \frac{\lambda^k}{k!}\cdot e^{-\lambda} \qquad \text{for } k=0,1,2,\ldots \\ \mathbb{E}Y &= \lambda \\ \mathrm{Var}(Y) &= \lambda \end{aligned}$$

Is there a linear model with Pois(λ) instead of $\mathcal{N}(\mu,\sigma^2)$?

Yes, the **Generalized Linear Model (GLM) of type Poisson**.

Remeber the normal linear model:

$$Y_i = b_0 + b_1 \cdot X_{1,i} + \cdots + b_k \cdot X_{k,i} + \varepsilon_i \qquad \text{with } \varepsilon_i \sim \mathcal{N}(0, \sigma^2)$$

Remeber the normal linear model:

$$Y_i = b_0 + b_1 \cdot X_{1,i} + \cdots + b_k \cdot X_{k,i} + \varepsilon_i \qquad \text{with } \varepsilon_i \sim \mathcal{N}(0, \sigma^2)$$

or equivalently:

$$\begin{aligned} \eta_i &= b_0 + b_1 \cdot X_{1,i} + \cdots + b_k \cdot X_{k,i} \\ Y_i &\sim \mathcal{N}(\eta_i, \sigma^2) \end{aligned}$$

η is called the *linear predictor*.

Remeber the normal linear model:

$$Y_i = b_0 + b_1 \cdot X_{1,i} + \cdots + b_k \cdot X_{k,i} + \varepsilon_i \qquad \text{with } \varepsilon_i \sim \mathcal{N}(0, \sigma^2)$$

or equivalently:

$$\begin{array}{rcl} \eta_i & = & b_0 + b_1 \cdot X_{1,i} + \cdots + b_k \cdot X_{k,i} \\ Y_i & \sim & \mathcal{N}(\eta_i, \sigma^2) \end{array}$$

η is called the *linear predictor*.

This also works for the Poisson distribution:

$$\begin{array}{rcl} \eta_i & = & b_0 + b_1 \cdot X_{1,i} + \cdots + b_k \cdot X_{k,i} \\ Y_i & \sim & \mathrm{Pois}(\eta_i) \end{array}$$

(but note that the additional σ^2 is missing!)

Instead of using η directly as parameter of the Poisson distribution, it is common to apply a transformation:

$$\begin{aligned} \ell(\mu_i) = \eta_i &= b_0 + b_1 \cdot X_{1,i} + \cdots + b_k \cdot X_{k,i} \\ Y_i &\sim \mathrm{Pois}(\mu_i) \end{aligned}$$

$\ell(.)$ is called the *link function*.

Instead of using η directly as parameter of the Poisson distribution, it is common to apply a transformation:

$$\begin{aligned} \ell(\mu_i) = \eta_i &= b_0 + b_1 \cdot X_{1,i} + \cdots + b_k \cdot X_{k,i} \\ Y_i &\sim \mathrm{Pois}(\mu_i) \end{aligned}$$

$\ell(.)$ is called the *link function*.
The default link function for Poisson GLMs is log, the natural logarithm.

Instead of using η directly as parameter of the Poisson distribution, it is common to apply a transformation:

$$\begin{aligned} \ell(\mu_i) \;=\; \eta_i &= b_0 + b_1 \cdot X_{1,i} + \cdots + b_k \cdot X_{k,i} \\ Y_i &\sim \mathrm{Pois}(\mu_i) \end{aligned}$$

$\ell(.)$ is called the *link function*.
The default link function for Poisson GLMs is log, the natural logarithm.
Thus,

$$\mathbb{E}\,Y_i = \mu_i = e^{\eta_i} = e^{b_0 + b_1 \cdot X_{1,i} + \cdots + b_k \cdot X_{k,i}} = e^{b_0} \cdot e^{b_1 \cdot X_{1,i}} \cdots e^{b_k \cdot X_{k,i}}$$

and the Poisson GLM with this default link is multiplicative model rather than an additive one.

Contents

```
> pmod1 <- glm(counts~foodlevel+species,data=daph,
                                          family=poisson)
> summary(pmod1)
[...]
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)    3.1166     0.1105  28.215  < 2e-16 ***
foodlevellow  -1.1567     0.1298  -8.910  < 2e-16 ***
speciesmagna   0.9794     0.1243   7.878 3.32e-15 ***
[...]
```

Note that the Poisson model has log as its default link function. Thus, the model pmod1 assumes that the number of Daphnia in row i is Poisson distributed with mean λ_i, i.e. $\Pr(X = k) = \frac{\lambda_i^k}{k!} e^{-\lambda}$, and

$$\log(\lambda_i) \approx 3.12 - 1.15 \cdot I_{\text{lowfoodlevel}} + 0.979 \cdot I_{\text{magna}}$$

Note that the Poisson model has log as its default link function. Thus, the model pmod1 assumes that the number of Daphnia in row i is Poisson distributed with mean λ_i, i.e. $\Pr(X = k) = \frac{\lambda_i^k}{k!} e^{-\lambda}$, and

$$\log(\lambda_i) \approx 3.12 - 1.15 \cdot I_{\mathrm{lowfoodlevel}} + 0.979 \cdot I_{\mathrm{magna}}$$

or, equivalently,

$$\lambda_i \approx e^{3.12} \cdot e^{-1.15}{}^{I_{\mathrm{lowfoodlevel}}} \cdot e^{0.979}{}^{I_{\mathrm{magna}}} \approx 22.6 \cdot 0.317^{I_{\mathrm{lowfoodlevel}}} \cdot 2.66^{I_{\mathrm{magna}}}$$

Thus, this Poisson model assumes multiplicative effects.

```
> pmod1 <- glm(counts~foodlevel+species,
                             data=daph,family=poisson)
> pmod2 <- glm(counts~foodlevel*species,
                             data=daph,family=poisson)
> anova(pmod1,pmod2,test="F")
```

```
> pmod1 <- glm(counts~foodlevel+species,
                               data=daph,family=poisson)
> pmod2 <- glm(counts~foodlevel*species,
                               data=daph,family=poisson)
> anova(pmod1,pmod2,test="F")

Analysis of Deviance Table

Model 1: counts ~ foodlevel + species
Model 2: counts ~ foodlevel * species
  Resid. Df Resid. Dev Df Deviance      F Pr(>F)
1         9     6.1162
2         8     6.0741  1 0.042071 0.0421 0.8375
Warning message:
F-Test not appropriate for family 'poisson'
```

Note:

- The anova command gives us an “analysis of deviance” instead of an analysis of variance!

Note:

- The anova command gives us an “analysis of deviance” instead of an analysis of variance!
- What is a deviance?

Note:

- The anova command gives us an “analysis of deviance” instead of an analysis of variance!
- What is a deviance?
- There is a Warning “F-Test not appropriate for family ’poisson’ ”.

Note:

- The anova command gives us an “analysis of deviance” instead of an analysis of variance!
- What is a deviance?
- There is a Warning “F-Test not appropriate for family ’poisson’ ”.
- Why?

Note:

- The anova command gives us an “analysis of deviance” instead of an analysis of variance!
- What is a deviance?
- There is a Warning “F-Test not appropriate for family ’poisson’ ”.
- Why?
- Which test should we apply?

What is the deviance?

Let $\widehat{b_0}, \ldots, \widehat{b_k}$ be our fitted model coefficients and

$$\widehat{\mu}_i = \ell^{-1}\left(\widehat{b_0} + \widehat{b_1} X_{1i} + \cdots + \widehat{b_k} X_{ki}\right)$$

be the predicted means for all observations. The Likelihood of the fitted parameter values is the probability of the observations assuming the fitted parameter values:

$$L(\widehat{\mu}) = \frac{\widehat{\mu_1}^{Y_1}}{Y_1!} e^{-\widehat{\mu_1}} \cdot \frac{\widehat{\mu_2}^{Y_2}}{Y_2!} e^{-\widehat{\mu_2}} \cdots \frac{\widehat{\mu_k}^{Y_k}}{Y_k!} e^{-\widehat{\mu_k}}$$

Now we compare this to a *saturated* Poisson GLM model, i.e. a model with so many parameters such that we can get a perfect fit of $\widetilde{\mu}_i = Y_i$. This leads to the highest possible likelihood $L(\widetilde{\mu})$.

What is the deviance?

Let $\widehat{b_0}, \ldots, \widehat{b_k}$ be our fitted model coefficients and

$$\widehat{\mu_i} = \ell^{-1}\left(\widehat{b_0} + \widehat{b_1} X_{1i} + \cdots + \widehat{b_k} X_{ki}\right)$$

be the predicted means for all observations. The Likelihood of the fitted parameter values is the probability of the observations assuming the fitted parameter values:

$$L(\widehat{\mu}) = \frac{\widehat{\mu_1}^{Y_1}}{Y_1!} e^{-\widehat{\mu_1}} \cdot \frac{\widehat{\mu_2}^{Y_2}}{Y_2!} e^{-\widehat{\mu_2}} \cdots \frac{\widehat{\mu_k}^{Y_k}}{Y_k!} e^{-\widehat{\mu_k}}$$

Now we compare this to a *saturated* Poisson GLM model, i.e. a model with so many parameters such that we can get a perfect fit of $\widetilde{\mu}_i = Y_i$. This leads to the highest possible likelihood $L(\widetilde{\mu})$.In practice such a model is not desirable because it leads to overfitting.

What is the deviance?

$$
\begin{aligned}
\text{our model: } L(\widehat{\mu}) &= \frac{\widehat{\mu_1}^{Y_1}}{Y_1!}e^{-\widehat{\mu_1}} \cdot \frac{\widehat{\mu_2}^{Y_2}}{Y_2!}e^{-\widehat{\mu_2}} \cdots \frac{\widehat{\mu_k}^{Y_k}}{Y_k!}e^{-\widehat{\mu_k}} \\
\text{saturated model: } L(\widetilde{\mu}) &= \frac{Y_1^{Y_1}}{Y_1!}e^{-Y_1} \cdot \frac{Y_2^{Y_2}}{Y_2!}e^{-Y_2} \cdots \frac{{Y_k}^{Y_k}}{Y_k!}e^{-Y_k}
\end{aligned}
$$

What is the deviance?

$$\begin{aligned}
\text{our model: } L(\widehat{\mu}) &= \frac{\widehat{\mu_1}^{Y_1}}{Y_1!} e^{-\widehat{\mu_1}} \cdot \frac{\widehat{\mu_2}^{Y_2}}{Y_2!} e^{-\widehat{\mu_2}} \cdots \frac{\widehat{\mu_k}^{Y_k}}{Y_k!} e^{-\widehat{\mu_k}} \\
\text{saturated model: } L(\widetilde{\mu}) &= \frac{Y_1^{Y_1}}{Y_1!} e^{-Y_1} \cdot \frac{Y_2^{Y_2}}{Y_2!} e^{-Y_2} \cdots \frac{Y_k^{Y_k}}{Y_k!} e^{-Y_k}
\end{aligned}$$

The *residual deviance* of our model is defined as

$$2 \cdot \left[\log\left(L(\widehat{\mu})\right) - \log\left(L(\widetilde{\mu})\right)\right].$$

What is the deviance?

$$\begin{aligned}\text{our model: } L(\widehat{\mu}) &= \frac{\widehat{\mu_1}^{Y_1}}{Y_1!}e^{-\widehat{\mu_1}} \cdot \frac{\widehat{\mu_2}^{Y_2}}{Y_2!}e^{-\widehat{\mu_2}} \cdots \frac{\widehat{\mu_k}^{Y_k}}{Y_k!}e^{-\widehat{\mu_k}} \\ \text{saturated model: } L(\widetilde{\mu}) &= \frac{Y_1^{Y_1}}{Y_1!}e^{-Y_1} \cdot \frac{Y_2^{Y_2}}{Y_2!}e^{-Y_2} \cdots \frac{Y_k^{Y_k}}{Y_k!}e^{-Y_k}\end{aligned}$$

The *residual deviance* of our model is defined as

$$2 \cdot \left[\log\left(L(\widehat{\mu})\right) - \log\left(L(\widetilde{\mu})\right)\right].$$

It measures how far our model is away from the theoretical optimum.

- The deviance is approximately χ^2_{df} distributed, where df is the degrees of freedom of our model.

- The deviance is approximately χ^2_{df} distributed, where df is the degrees of freedom of our model.
- Thus, the deviance should be of the same order of magnitude as df.

- The deviance is approximately χ^2_{df} distributed, where df is the degrees of freedom of our model.
- Thus, the deviance should be of the same order of magnitude as df.
- Check this to assess the fit of the model!

- The deviance is approximately χ^2_{df} distributed, where df is the degrees of freedom of our model.
- Thus, the deviance should be of the same order of magnitude as df.
- Check this to assess the fit of the model!

Analysis of deviance:

If D_1 and D_2 are the deviances of models M_1 with p_1 parameters and M_2 with p_2 parameters, and M_1 is nested in M_2 (i.e. the parameters of M_1 are a subset of the parameters of M_2), then $D_1 - D_2$ is approximately $\chi^2_{p_2-p_1}$-distributed.

- The deviance is approximately χ^2_{df} distributed, where df is the degrees of freedom of our model.
- Thus, the deviance should be of the same order of magnitude as df.
- Check this to assess the fit of the model!

Analysis of deviance:
If D_1 and D_2 are the deviances of models M_1 with p_1 parameters and M_2 with p_2 parameters, and M_1 is nested in M_2 (i.e. the parameters of M_1 are a subset of the parameters of M_2), then $D_1 - D_2$ is approximately $\chi^2_{p_2-p_1}$-distributed.
This Test is the classical likelihood-ratio test. (Note that $D_1 - D_2$ is 2x the log of the likelihood-ratio of the two models.)

```
> pmod1 <- glm(counts~foodlevel+species,
                              data=daph,family=poisson)
> pmod2 <- glm(counts~foodlevel*species,
                              data=daph,family=poisson)
> anova(pmod1,pmod2,test="Chisq")

Analysis of Deviance Table

Model 1: counts ~ foodlevel + species
Model 2: counts ~ foodlevel * species
  Resid. Df Resid. Dev Df Deviance    P(>|Chi|)
1         9     6.1162
2         8     6.0741  1 0.042071    0.8375
```

Why not the *F*-test?

Why not the *F*-test?
Remember that we did not estimate a variance σ^2 for the Poisson distribution.

Why not the F-test?
Remember that we did not estimate a variance σ^2 for the Poisson distribution.
There is an F-distribution approximation of a rescaled $D_1 - D_2$ for GLMs in which an extra variance parameter is estimated.

Why not the *F*-test?
Remember that we did not estimate a variance σ^2 for the Poisson distribution.
There is an *F*-distribution approximation of a rescaled $D_1 - D_2$ for GLMs in which an extra variance parameter is estimated.

Example: *overdispersed Poisson*, also called *quasipoisson* GLM. Here, $\mathbb{E} Y_i = \mu_i$ but $\mathrm{Var}\, Y_i = \phi \cdot \mu_i$ with the dispersion parameter $\phi > 1$.

Why not the F-test?
Remember that we did not estimate a variance σ^2 for the Poisson distribution.
There is an F-distribution approximation of a rescaled $D_1 - D_2$ for GLMs in which an extra variance parameter is estimated.

Example: *overdispersed Poisson*, also called *quasipoisson* GLM. Here, $\mathbb{E}Y_i = \mu_i$ but $\mathrm{Var}Y_i = \phi \cdot \mu_i$ with the dispersion parameter $\phi > 1$.
This is often used to model the influence of unknown external factors.

Why not the *F*-test?
Remember that we did not estimate a variance σ^2 for the Poisson distribution.
There is an *F*-distribution approximation of a rescaled $D_1 - D_2$ for GLMs in which an extra variance parameter is estimated.

Example: *overdispersed Poisson*, also called *quasipoisson* GLM. Here, $\mathbb{E}Y_i = \mu_i$ but $\mathrm{Var}Y_i = \phi \cdot \mu_i$ with the dispersion parameter $\phi > 1$.
This is often used to model the influence of unknown external factors.
Since the dispersion parameter is estimated, one can apply an *F* approximation in the analysis of deviance.

Why not the F-test?
Remember that we did not estimate a variance σ^2 for the Poisson distribution.
There is an F-distribution approximation of a rescaled $D_1 - D_2$ for GLMs in which an extra variance parameter is estimated.

Example: *overdispersed Poisson*, also called *quasipoisson* GLM. Here, $\mathbb{E}Y_i = \mu_i$ but $\operatorname{Var}Y_i = \phi \cdot \mu_i$ with the dispersion parameter $\phi > 1$.
This is often used to model the influence of unknown external factors.
Since the dispersion parameter is estimated, one can apply an F approximation in the analysis of deviance. But also χ^2 is still an option.

```
> qpmod1 <- glm(counts~foodlevel+species,data=daph,
                                  family=quasipoisson)
> qpmod2 <- glm(counts~foodlevel*species,data=daph,
                                  family=quasipoisson)
> anova(qpmod1,qpmod2,test="F")
Analysis of Deviance Table

Model 1: counts ~ foodlevel + species
Model 2: counts ~ foodlevel * species
  Resid. Df Resid. Dev Df Deviance      F Pr(>F)
1         9     6.1162
2         8     6.0741  1 0.042071 0.0572  0.817
```

```
> anova(qpmod1,qpmod2,test="Chisq")
Analysis of Deviance Table

Model 1: counts ~ foodlevel + species
Model 2: counts ~ foodlevel * species
  Resid. Df Resid. Dev Df Deviance P(>|Chi|)
1         9     6.1162
2         8     6.0741  1 0.042071     0.811
```

```
> expect <- predict(pmod1,type="response")
> sim <- rpois(12,expect)
> smod1 <- lm(sim~foodlevel+species,data=daph)
> smod2 <- lm(sim~foodlevel*species,data=daph)
> anova(smod1,smod2)
```

```
> expect <- predict(pmod1,type="response")
> sim <- rpois(12,expect)
> smod1 <- lm(sim~foodlevel+species,data=daph)
> smod2 <- lm(sim~foodlevel*species,data=daph)
> anova(smod1,smod2)

Analysis of Variance Table

Model 1: sim ~ foodlevel + species
Model 2: sim ~ foodlevel * species
  Res.Df     RSS Df Sum of Sq      F    Pr(>F)
1      9 1289.42
2      8  109.33  1    1180.1 86.348 1.464e-05 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

What is the problem? Normal distribution assumption or additivity?

What is the problem? Normal distribution assumption or additivity?

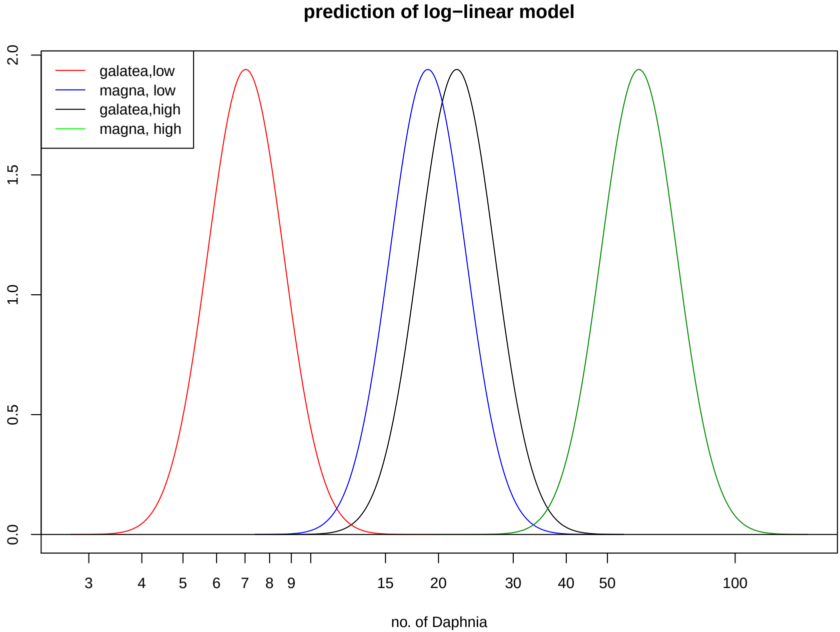
How about a multiplicative linear model?

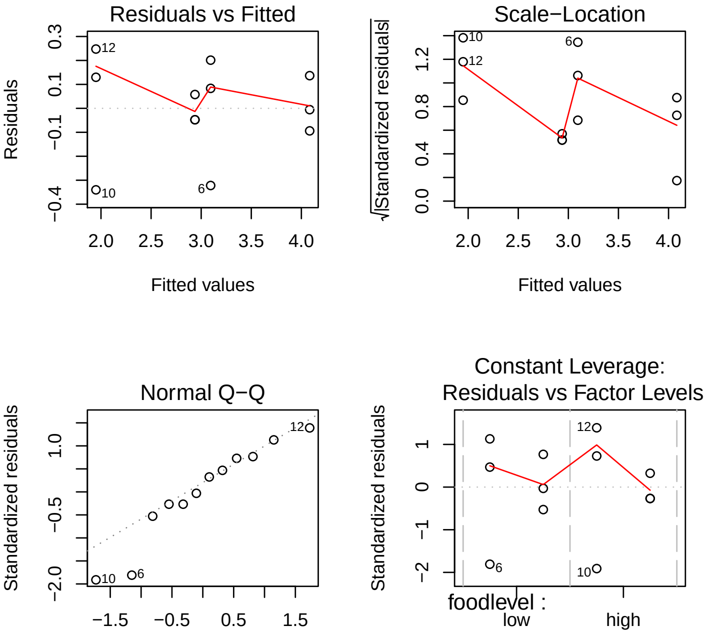
```
> expect <- predict(pmod1,type="response")
> sim <- rpois(12,expect)
> smod1 <- lm(log(sim)~foodlevel+species,data=daph)
> smod2 <- lm(log(sim)~foodlevel*species,data=daph)
> anova(smod1,smod2)
```

```
> expect <- predict(pmod1,type="response")
> sim <- rpois(12,expect)
> smod1 <- lm(log(sim)~foodlevel+species,data=daph)
> smod2 <- lm(log(sim)~foodlevel*species,data=daph)
> anova(smod1,smod2)

Analysis of Variance Table

Model 1: log(sim) ~ foodlevel + species
Model 2: log(sim) ~ foodlevel * species
  Res.Df     RSS Df Sum of Sq      F Pr(>F)
1      9 0.19216
2      8 0.19115  1 0.0010162 0.0425 0.8418
```

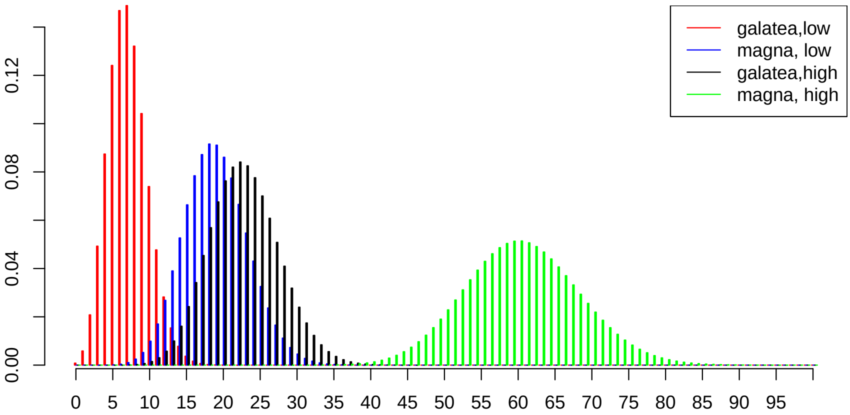
This solves the biggest problem, but what does the model say?

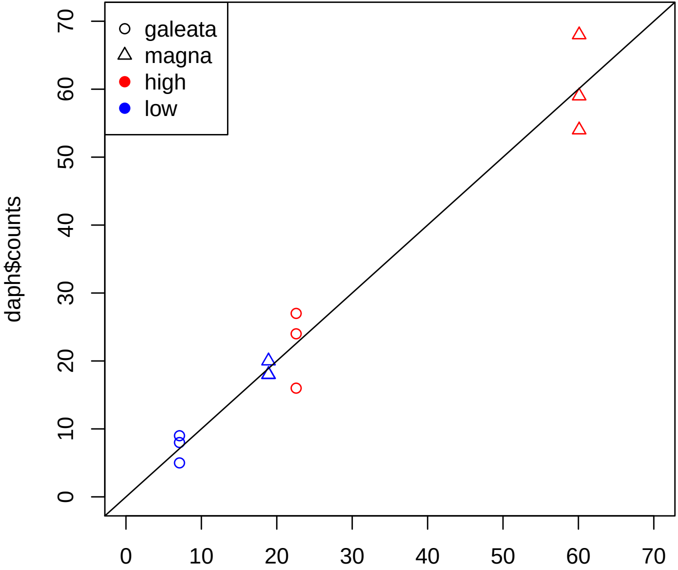
```
> lmod1 <- lm(log(counts)~foodlevel+species,data=daph)
[...]
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)    3.0946     0.1028  30.104 2.41e-10 ***
foodlevellow  -1.1450     0.1187  -9.646 4.83e-06 ***
speciesmagna   0.9883     0.1187   8.326 1.61e-05 ***
[...]
Residual standard error: 0.2056 on 9 degrees of freedom
[...]
```


prediction of log−linear model
galatea,low
magna, low
galatea,high
magna, high
0.0
0.5
1.0
1.5
2.0
3
4
5
6
7
8
9
15
20
30
40
50
100
no. of Daphnia


Residuals vs Fitted
Residuals
0.3
0.1
−0.1
−0.4
Fitted values
2.0
2.5
3.0
3.5
4.0
12
10
6
Scale−Location
√|Standardized residuals|
0.0
0.4
0.8
1.2
Fitted values
10
12
6
Normal Q−Q
Standardized residuals
−2.0
−0.5
1.0
−1.5
−0.5
0.5
1.5
12
10
6
Constant Leverage:
Residuals vs Factor Levels
Standardized residuals
−2
−1
0
1
foodlevel :
low
high
12
10
6

```
> summary(pmod1)
[..]
glm(formula = counts ~ foodlevel + species,
          family = poisson, data = daph)
[..]
             Estimate Std. Error z value Pr(>|z|)
(Intercept)    3.1166     0.1105  28.215  < 2e-16 ***
foodlevellow  -1.1567     0.1298  -8.910  < 2e-16 ***
speciesmagna   0.9794     0.1243   7.878 3.32e-15 ***
[..]
(Dispersion parameter for poisson family taken to be 1)
[..]
Residual deviance:   6.1162  on  9  degrees of freedom
AIC: 70.497
```


galatea,low
magna, low
galatea,high
magna, high
0.00
0.04
0.08
0.12
0
5
10
15
20
25
30
35
40
45
50
55
60
65
70
75
80
85
90
95


galeata
magna
high
low
daph$counts
0
10
20
30
40
50
60
70
0
10
20
30
40
50
60
70

Since the variance is proportional to the expectation value in the Poisson model, usual residuals are not so informatative.

Since the variance is proportional to the expectation value in the Poisson model, usual residuals are not so informatative.

Instead use *deviance residuals*. Let d_i be the contribution of observation *i* (row *i* in the data table) to the Deviance, then the deviance residual of observation *i* is

$$\mathrm{sign}(Y_i - \widehat{\mu}_i) \cdot \sqrt{d_i}.$$

Since the variance is proportional to the expectation value in the Poisson model, usual residuals are not so informatative.

Instead use *deviance residuals*. Let d_i be the contribution of observation i (row i in the data table) to the Deviance, then the deviance residual of observation i is

$$\mathrm{sign}(Y_i - \widehat{\mu}_i) \cdot \sqrt{d_i}.$$

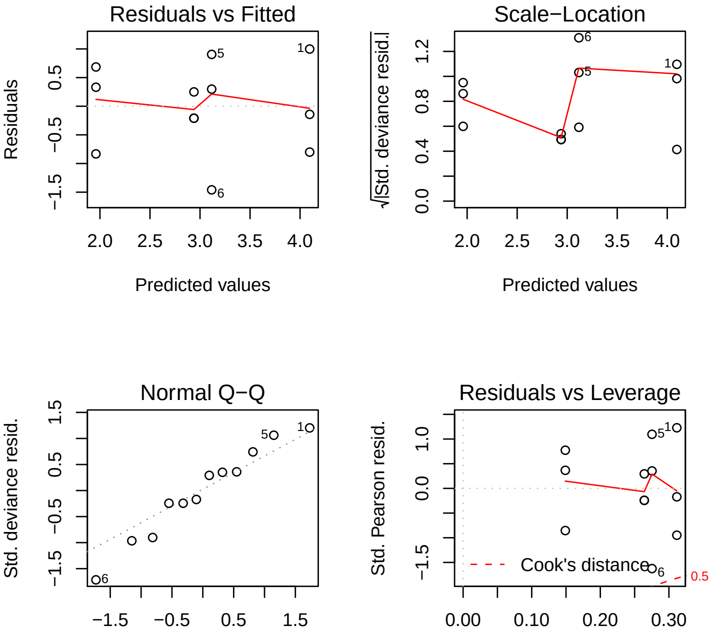
The deviance residuals are the default residuals given by R for GLMs. They have similar properties as the standard residuals in the normal linear model.

Since the variance is proportional to the expectation value in the Poisson model, usual residuals are not so informatative.

Instead use *deviance residuals*. Let d_i be the contribution of observation i (row i in the data table) to the Deviance, then the deviance residual of observation i is

$$\mathrm{sign}(Y_i - \widehat{\mu}_i) \cdot \sqrt{d_i}.$$

The deviance residuals are the default residuals given by R for GLMs. They have similar properties as the standard residuals in the normal linear model.
In the following plot obtained with plot(pmod1) the word “residual” always refers to deviance residuals.


Residuals vs Fitted
Residuals
−1.5
−0.5
0.5
Predicted values
2.0
2.5
3.0
3.5
4.0
5
1
6
Scale−Location
√|Std. deviance resid.|
0.0
0.4
0.8
1.2
Predicted values
2.0
2.5
3.0
3.5
4.0
6
1
5
Normal Q−Q
Std. deviance resid.
−1.5
−0.5
0.5
1.5
−1.5
−0.5
0.5
1.5
5
1
6
Residuals vs Leverage
Std. Pearson resid.
−1.5
0.0
1.0
0.00
0.10
0.20
0.30
5
1
6
Cook's distance
0.5

Contents

In the lecture about linear regression we analysed a data set to find out whether the county size (number of females living in a county) has an effect on the risk of dying by breast cancer. Since the response variable in this data set are deaths counts, it seems natural to fit a Poisson GLM.

```
> str(canc)
'data.frame': 301 obs. of  2 variables:
 $ deaths     : int  1 0 3 4 3 4 1 5 5 5 ...
 $ inhabitants: int  445 559 677 681 746 869 950 976 ...
```

First trial:

```
> mod0 <- glm(deaths~inhabitants,data=canc,family=poisson)
> summary(mod0)

Call:
glm(formula = deaths ~ inhabitants, family = poisson,
data = canc)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-13.8783   -2.6449   -0.8845    1.8160    6.9909

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) 2.961e+00  1.320e-02   224.2   <2e-16 ***
inhabitants 4.044e-05  3.374e-07   119.9   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

Before we complain about the large residual deviance... we ask ourselves whether this is a plausible model.

Before we complain about the large residual deviance... we ask ourselves whether this is a plausible model.

Let D_i be the *expected* number of deaths in county i and S_i its size. Then the model assumes

$$\log(D_i) = a + b \cdot S_i$$

Before we complain about the large residual deviance... we ask ourselves whether this is a plausible model.

Let D_i be the *expected* number of deaths in county i and S_i its size. Then the model assumes

$$\log(D_i) = a + b \cdot S_i$$

or, equivalently,

$$D_i = e^{a+b\cdot S_i} = e^a \cdot \left(e^{S_i}\right)^b$$

Before we complain about the large residual deviance... we ask ourselves whether this is a plausible model.

Let D_i be the *expected* number of deaths in county i and S_i its size. Then the model assumes

$$\log\left(D_i\right) = a + b \cdot S_i$$

or, equivalently,

$$D_i = e^{a+b\cdot S_i} = e^a \cdot \left(e^{S_i}\right)^b$$

this is not a plausible model.

Solution: take the log of S_i.

$$\log(D_i) = a + b \cdot \log(S_i)$$

Solution: take the log of S_i.

$$\log(D_i) = a + b \cdot \log(S_i)$$

or, equivalently,

$$D_i \;=\; e^{a+b\cdot\log(S_i)} \;=\; e^a \cdot \left(e^{\log(S_i)}\right)^b$$

Solution: take the log of S_i.

$$\log(D_i) = a + b \cdot \log(S_i)$$

or, equivalently,

$$D_i \;=\; e^{a+b\cdot\log(S_i)} \;=\; e^a \cdot \left(e^{\log(S_i)}\right)^b \;= e^{\,a} \cdot S_i^b$$

Solution: take the log of S_i.

$$\log(D_i) = a + b \cdot \log(S_i)$$

or, equivalently,

$$D_i \;=\; e^{a+b\cdot\log(S_i)} \;=\; e^a \cdot \left(e^{\log(S_i)}\right)^b \;= e^{\,a} \cdot S_i^b$$

If $b = 1$, then e^a is just the individual risk to die by breast cancer (during the time span of the survey).

```
> mod1 <- glm(deaths~log(inhabitants),data=canc,family=poisson)
> summary(mod1)
[..]
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)      -5.531496   0.093003  -59.48   <2e-16 ***
log(inhabitants)  0.988350   0.009406  105.08   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 12994.06  on 300  degrees of freedom
Residual deviance:   785.85  on 299  degrees of freedom
AIC: 2282.9
```

```
> mod1 <- glm(deaths~log(inhabitants),data=canc,family=poisson)
> summary(mod1)
[..]
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)      -5.531496   0.093003  -59.48   <2e-16 ***
log(inhabitants)  0.988350   0.009406  105.08   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 12994.06  on 300  degrees of freedom
Residual deviance:   785.85  on 299  degrees of freedom
AIC: 2282.9
```

Too much residual deviance for df=299 ⇒ Let's allow for overdispersion!

```
> mod2 <- glm(deaths˜log(inhabitants),data=canc,family=quasipoisson
> summary(mod2)
[...]
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)      -5.53150    0.14865  -37.21   <2e-16 ***
log(inhabitants)  0.98835    0.01503   65.75   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for quasipoisson family taken to be 2.554585)

    Null deviance: 12994.06  on 300  degrees of freedom
Residual deviance:   785.85  on 299  degrees of freedom
```

```
> mod2 <- glm(deaths~log(inhabitants),data=canc,family=quasipoisson
> summary(mod2)
[...]
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)      -5.53150    0.14865  -37.21   <2e-16 ***
log(inhabitants)  0.98835    0.01503   65.75   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for quasipoisson family taken to be 2.554585)

    Null deviance: 12994.06  on 300  degrees of freedom
Residual deviance:   785.85  on 299  degrees of freedom
```

What does the highly significant *p*-value for log(inhabitants) say?

```
> mod2 <- glm(deaths~log(inhabitants),data=canc,family=quasipoisson
> summary(mod2)
[...]
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)      -5.53150    0.14865  -37.21   <2e-16 ***
log(inhabitants)  0.98835    0.01503   65.75   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for quasipoisson family taken to be 2.554585)

    Null deviance: 12994.06  on 300  degrees of freedom
Residual deviance:   785.85  on 299  degrees of freedom
```

What does the highly significant *p*-value for log(inhabitants) say?
It says that the coefficient *b* is significantly different from 0.

```
> mod2 <- glm(deaths~log(inhabitants),data=canc,family=quasipoisson
> summary(mod2)
[...]
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       -5.53150    0.14865  -37.21   <2e-16 ***
log(inhabitants)   0.98835    0.01503   65.75   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for quasipoisson family taken to be 2.554585)

    Null deviance: 12994.06  on 300  degrees of freedom
Residual deviance:   785.85  on 299  degrees of freedom
```

What does the highly significant p-value for log(inhabitants) say?
It says that the coefficient b is significantly different from 0.

But our question is rather whether b is significantly different from 1!

Trick: Fit a model

$$\log\left(D_i\right) = a + \log\left(S_i\right) + b \cdot \log\left(S_i\right)$$

Trick: Fit a model

$$\log(D_i) = a + \log(S_i) + b \cdot \log(S_i)$$

which is equivalent to

$$D_i = e^a \cdot S_i \cdot S_i^b.$$

Trick: Fit a model

$$\log(D_i) = a + \log(S_i) + b \cdot \log(S_i)$$

which is equivalent to

$$D_i = e^a \cdot S_i \cdot S_i^b.$$

Then the question is whether b is significantly different from 0.

Trick: Fit a model

$$\log(D_i) = a + \log(S_i) + b \cdot \log(S_i)$$

which is equivalent to

$$D_i = e^a \cdot S_i \cdot S_i^b.$$

Then the question is whether b is significantly different from 0.

in R: use the command `offset` to tell R not to estimate a coefficient for the first $\log(S_i)$

```
> mod3 <- glm(deaths~offset(log(inhabitants))+log(inhabitants),
                                      data=canc,family=quasipoisson)
> summary(mod3)
[...]
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       -5.53150    0.14865 -37.212   <2e-16 ***
log(inhabitants) -0.01165    0.01503  -0.775    0.439
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for quasipoisson family taken to be 2.554585)
```

```
> mod3 <- glm(deaths~offset(log(inhabitants))+log(inhabitants),
                                  data=canc,family=quasipoisson)
> summary(mod3)
[...]
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)      -5.53150    0.14865 -37.212   <2e-16 ***
log(inhabitants) -0.01165    0.01503  -0.775    0.439
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for quasipoisson family taken to be 2.554585)
```

Thus, the expected number of deaths seems to be just proportional to the number of inhabitants. No significant dependence of the death rate on the county size was found.

Another way of testing this:

```
> mod4 <- glm(deaths~offset(log(inhabitants)),
                   data=canc,family=quasipoisson)
> anova(mod4,mod3,test="F")
Analysis of Deviance Table

Model 1:
deaths ~ offset(log(inhabitants))
Model 2:
deaths ~ offset(log(inhabitants)) + log(inhabitants)

  Resid. Df Resid. Dev Df Deviance      F Pr(>F)
1       300     787.38
2       299     785.85  1   1.5315 0.5995 0.4394
```

Contents

Contents

```
> fly <- read.csv("Flies_AnaCatalan.csv",h=T,sep=";")
> fly
    odorant resp air PI     sex day species
1       CO2    1  29 NA   males   1     mel
2       CO2    2  28 NA   males   1     mel
3       CO2    1  25 NA   males   1     mel
.        .     .   . .      .     .      .
.        .     .   . .      .     .      .
.        .     .   . .      .     .      .
753   30CO2    4   7 NA females   2     vir
754   30CO2    6  12 NA females   2     vir
755   30CO2    6  11 NA females   2     vir
756   30CO2    6  15 NA females   2     vir
```

```
> str(fly)
'data.frame': 756 obs. of  7 variables:
 $ odorant: Factor w/ 3 levels "30CO2","CO2",..: 2 2 2 2
 $ resp   : int  1 2 1 2 5 4 9 5 5 11 ...
 $ air    : int  29 28 25 17 36 42 38 13 19 25 ...
 $ PI     : logi  NA NA NA NA NA NA ...
 $ sex    : Factor w/ 2 levels "females","males": 2 2 2 2
 $ day    : int  1 1 1 1 1 1 2 2 2 2 ...
 $ species: Factor w/ 11 levels "ana","atr","ere",..: 5 5
```

Model

In experiment i (row i of the data table) there are n_i flies. Each of these flies decided independently of all other to go to the odorant with probability p_i and, thus, to go to the fresh air with probability $(1 - p_i)$.

Model

In experiment i (row i of the data table) there are n_i flies. Each of these flies decided independently of all other to go to the odorant with probability p_i and, thus, to go to the fresh air with probability $(1 - p_i)$.

Thus, the number Y_i of flies which went to the odorant is binomially distributed:

$$Y_i \sim \mathrm{bin}(n_i, p_i)$$

Model

In experiment i (row i of the data table) there are n_i flies. Each of these flies decided independently of all other to go to the odorant with probability p_i and, thus, to go to the fresh air with probability $(1 - p_i)$.

Thus, the number Y_i of flies which went to the odorant is binomially distributed:

$$\begin{aligned} Y_i &\sim \mathrm{bin}(n_i, p_i) \\ \Pr(Y_i = k) &= \binom{n_i}{k} \cdot p_i^k \cdot (1 - p_i)^{n_i - k} \end{aligned}$$

Model

In experiment i (row i of the data table) there are n_i flies. Each of these flies decided independently of all other to go to the odorant with probability p_i and, thus, to go to the fresh air with probability $(1 - p_i)$.

Thus, the number Y_i of flies which went to the odorant is binomially distributed:

$$\begin{aligned} Y_i &\sim \mathrm{bin}(n_i, p_i) \\ \Pr(Y_i = k) &= \binom{n_i}{k} \cdot p_i^k \cdot (1 - p_i)^{n_i - k} \\ \mathbb{E} Y_i &= n_i \cdot p_i \end{aligned}$$

Model

In experiment i (row i of the data table) there are n_i flies. Each of these flies decided independently of all other to go to the odorant with probability p_i and, thus, to go to the fresh air with probability $(1 - p_i)$.

Thus, the number Y_i of flies which went to the odorant is binomially distributed:

$$\begin{aligned} Y_i &\sim \mathrm{bin}(n_i, p_i) \\ \Pr(Y_i = k) &= \binom{n_i}{k} \cdot p_i^k \cdot (1 - p_i)^{n_i - k} \\ \mathbb{E}\, Y_i &= n_i \cdot p_i \\ \mathrm{Var}\, Y_i &= n_i \cdot p_i \cdot (1 - p_i) \end{aligned}$$

Model

In experiment i (row i of the data table) there are n_i flies. Each of these flies decided independently of all other to go to the odorant with probability p_i and, thus, to go to the fresh air with probability $(1 - p_i)$.

Thus, the number Y_i of flies which went to the odorant is binomially distributed:

$$\begin{aligned} Y_i &\sim \mathrm{bin}(n_i, p_i) \\ \Pr(Y_i = k) &= \binom{n_i}{k} \cdot p_i^k \cdot (1 - p_i)^{n_i - k} \\ \mathbb{E}\, Y_i &= n_i \cdot p_i \\ \mathrm{Var}\, Y_i &= n_i \cdot p_i \cdot (1 - p_i) \end{aligned}$$

How does p_i depend on the odorant and on the species?

Binomial GLM with logit link

Similar as in Poisson GLMs we assume:

$$\ell(p_i) \;=\; \eta_i \;=\; b_0 + b_1 \cdot X_{1,i} + \cdots + b_k \cdot X_{k,i}$$

Binomial GLM with logit link

Similar as in Poisson GLMs we assume:

$$\ell(p_i) \;=\; \eta_i \;=\; b_0 + b_1 \cdot X_{1,i} + \cdots + b_k \cdot X_{k,i}$$

The default link of the Binomial GLM is the logit link:

$$\eta \;=\; \mathrm{logit}(p) \;=\; \log(p/(1-p))$$

Binomial GLM with logit link

Similar as in Poisson GLMs we assume:

$$\ell(p_i) \;=\; \eta_i \;=\; b_0 + b_1 \cdot X_{1,i} + \cdots + b_k \cdot X_{k,i}$$

The default link of the Binomial GLM is the logit link:

$$\eta \;=\; \mathrm{logit}(p) \;=\; \log(p/(1-p))$$

Its inverse is the logistic function

$$p \;=\; \frac{1}{1+e^{-\eta}}$$

Binomial GLM with logit link

Similar as in Poisson GLMs we assume:

$$\ell(p_i) \;=\; \eta_i \;=\; b_0 + b_1 \cdot X_{1,i} + \cdots + b_k \cdot X_{k,i}$$

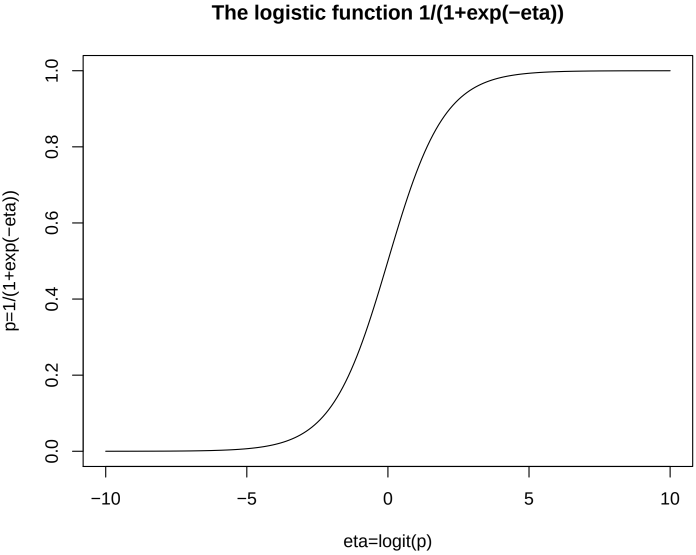
The default link of the Binomial GLM is the logit link:

$$\eta \;=\; \mathrm{logit}(p) \;=\; \log(p/(1-p))$$

Its inverse is the logistic function

$$p \;=\; \frac{1}{1+e^{-\eta}}$$

Binomial GLM with the logit link is also called *logistic regression*.


The logistic function 1/(1+exp(−eta))
p=1/(1+exp(−eta))
0.0
0.2
0.4
0.6
0.8
1.0
−10
−5
0
5
10
eta=logit(p)

Likelihood and Deviance

If $\widehat{p}_1, \ldots, \widehat{p}_k$ are the estimated p_i in our model, then the likelihood of the fitted parameters is

$$\begin{aligned} L(\widehat{p}) \;=\; & \binom{n_1}{Y_1} {\widehat{p}_1}^{Y_1}(1-\widehat{p}_1)^{n_1-Y_1} \cdot \binom{n_2}{Y_2} {\widehat{p}_2}^{Y_2}(1-\widehat{p}_2)^{n_2-Y_2} \cdots \\ & \cdots \binom{n_k}{Y_k} {\widehat{p}_k}^{Y_k}(1-\widehat{p}_k)^{n_k-Y_k} \end{aligned}$$

Using this likelihood, the *deviance* and the deviance residuals are defined like in the Poisson GLM.

Analysis of deviance and overdispersion

Note that, like in the Poisson model, $\text{Var}\, Y_i = n_i \cdot p_i \cdot (1 - p_i)$ is fixed for given $\mathbb{E}\, Y_i = n_i p_i$. Thus, the χ^2 approximation should be used in the anaysis of deviance.

Analysis of deviance and overdispersion

Note that, like in the Poisson model, $\mathrm{Var}\, Y_i = n_i \cdot p_i \cdot (1 - p_i)$ is fixed for given $\mathbb{E} Y_i = n_i p_i$. Thus, the χ^2 approximation should be used in the anaysis of deviance.

There is an overdispersed binomial GLM (available in R with the option family=quasibinomial) with an additional dispersion parameter. For these models one can use both χ^2 approximation and F approximations in analyses of deviance.

Contents

```
> fly <- read.csv("Flies_AnaCatalan.csv",h=T,sep=";")
> fly
    odorant resp air PI     sex day species
1       CO2    1  29 NA   males   1     mel
2       CO2    2  28 NA   males   1     mel
3       CO2    1  25 NA   males   1     mel
.         .    .   . .      .     .       .
.         .    .   . .      .     .       .
.         .    .   . .      .     .       .
753   30CO2    4   7 NA females   2     vir
754   30CO2    6  12 NA females   2     vir
755   30CO2    6  11 NA females   2     vir
756   30CO2    6  15 NA females   2     vir
```

```
> modelbin <- glm(cbind(resp,air)~(sex+species)*odorant+day,
+               subset=odorant!="oct",
+               data=fly,family=binomial)
> summary(modelbin)

Call:
glm(formula = cbind(resp, air) ~ (sex + species) * odorant +
    day, family = binomial, data = fly,
                          subset = odorant != "oct")

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.3735  -0.9693  -0.1187   0.7240   4.4994

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)           -1.376503   0.123901 -11.110  < 2e-16 ***
sexmales               0.131066   0.053810   2.436 0.014863 *
speciesatr             0.227528   0.145096   1.568 0.116854
speciesere             0.057917   0.150061   0.386 0.699528
speciesmau             0.141718   0.163017   0.869 0.384658
```

```
speciesmel          -1.128202   0.164920  -6.841 7.87e-12 ***
speciespse           1.318299   0.143279   9.201  < 2e-16 ***
speciessec          -0.518238   0.143658  -3.607 0.000309 ***
speciessim           0.427407   0.136345   3.135 0.001720 **
speciestei          -0.266130   0.144181  -1.846 0.064921 .
speciesvir           0.424609   0.173881   2.442 0.014608 *
speciesyak          -0.454361   0.170760  -2.661 0.007795 **
odorantCO2          -0.922118   0.171020  -5.392 6.97e-08 ***
day                 -0.008059   0.014922  -0.540 0.589129
sexmales:odorantCO2 -0.023450   0.067791  -0.346 0.729408
speciesatr:odorantCO2 1.180104  0.194524   6.067 1.31e-09 ***
speciesere:odorantCO2 1.473309  0.200023   7.366 1.76e-13 ***
speciesmau:odorantCO2 1.214336  0.222429   5.459 4.78e-08 ***
speciesmel:odorantCO2 1.530291  0.219269   6.979 2.97e-12 ***
speciespse:odorantCO2 0.384300  0.195086   1.970 0.048849 *
speciessec:odorantCO2 2.046612  0.194380  10.529  < 2e-16 ***
speciessim:odorantCO2 1.369519  0.189228   7.237 4.57e-13 ***
```

```
speciestei:odorantCO2  1.033078   0.199579   5.176 2.26e-07 ***
speciesvir:odorantCO2  1.262574   0.225086   5.609 2.03e-08 ***
speciesyak:odorantCO2  1.919994   0.215587   8.906  < 2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2429.1  on 663  degrees of freedom
Residual deviance: 1187.1  on 639  degrees of freedom
AIC: 3430.7

Number of Fisher Scoring iterations: 4
```

A residual deviance of 1187.1 on 639 degrees of freedom is very high and indicates that the model parameters cannot fully explain the data.

A residual deviance of 1187.1 on 639 degrees of freedom is very high and indicates that the model parameters cannot fully explain the data.

$\Rightarrow$ Fit an overdispersed model!

A residual deviance of 1187.1 on 639 degrees of freedom is very high and indicates that the model parameters cannot fully explain the data.

$\Rightarrow$ Fit an overdispersed model!

There is a price we have to pay for overdispersion: Since it is not a clearly defined distribution, AIC is not available for model selection.

A residual deviance of 1187.1 on 639 degrees of freedom is very high and indicates that the model parameters cannot fully explain the data.

⇒ Fit an overdispersed model!

There is a price we have to pay for overdispersion: Since it is not a clearly defined distribution, AIC is not available for model selection.

Select parameters

1. that seem important to you from the biological context
2. or have low *p*-values.

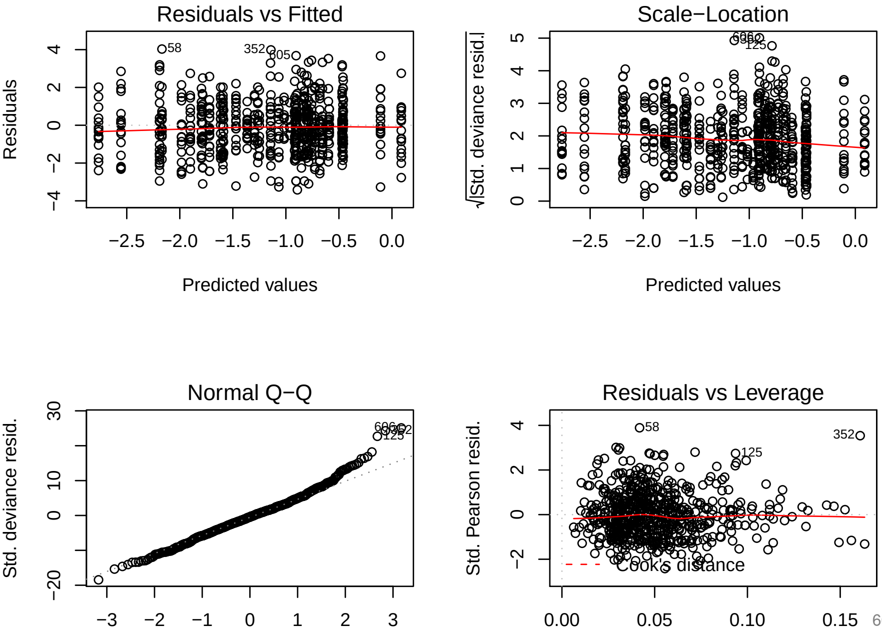
```
> model <- glm(cbind(resp,air)~(sex+species)*odorant+day,
+              subset=odorant!="oct",
+              data=fly,family=quasibinomial)
> drop1(model,test="F")
Single term deletions

Model:
cbind(resp, air) ~ (sex + species) * odorant + day
                Df Deviance F value  Pr(F)
<none>               1187.1
day              1   1187.3  0.1571 0.6920
sex:odorant      1   1187.2  0.0644 0.7997
species:odorant 10   1431.1 13.1365 <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

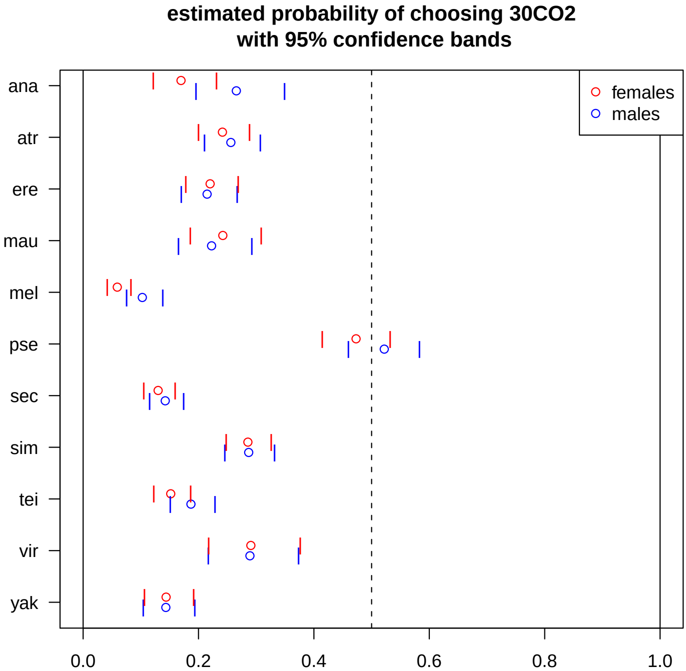
```
> model2 <- update(model,~.-day)
> drop1(model2,test="F")
Single term deletions

Model:
cbind(resp, air) ~ sex + species + odorant + sex:odorant
                Df Deviance F value  Pr(F)
<none>                1187.3
sex:odorant      1    1187.5  0.0673 0.7953
species:odorant 10    1432.6 13.2215 <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

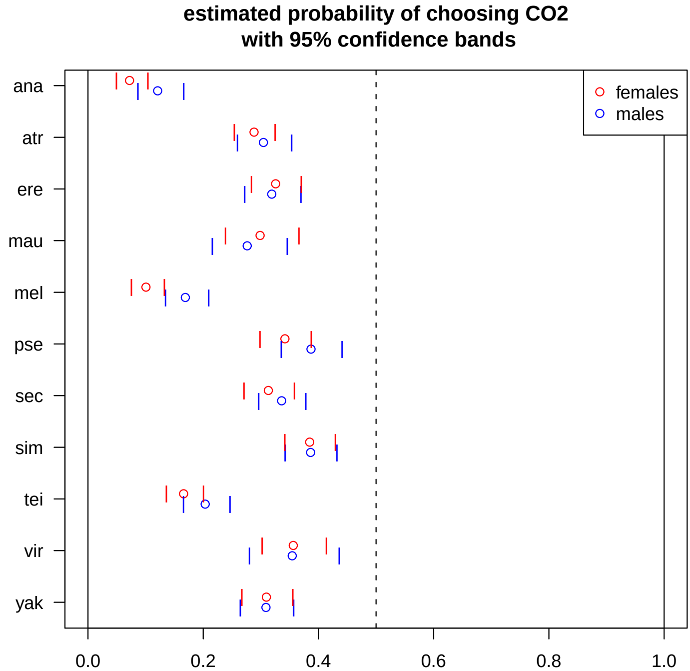
```
> model3 <- update(model2,~.-sex:odorant)
> drop1(model3,test="F")
Single term deletions

Model:
cbind(resp, air) ~ sex + species + odorant + species:odor
                Df Deviance F value   Pr(F)
<none>               1187.5
sex              1   1200.0  6.7785 0.00944 **
species:odorant 10   1432.7 13.2366 < 2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
> model4 <- glm(cbind(resp,air)~sex+species+odorant
+                          +species:odorant+species:sex,
+               subset=odorant!="oct",
+               data=fly,family=quasibinomial)
> anova(model3,model4,test="F")
Analysis of Deviance Table

Model 1: cbind(resp, air) ~ sex + species + odorant + spe
Model 2: cbind(resp, air) ~ sex + species + odorant + spe
    species:sex
  Resid. Df Resid. Dev Df Deviance      F Pr(>F)
1       641     1187.5
2       631     1157.1 10   30.395 1.7232  0.072 .
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
> drop1(model4,test="F")
Single term deletions

Model:
cbind(resp, air) ~ sex + species + odorant + species:odor
    species:sex
                Df Deviance F value   Pr(F)
<none>                1157.1
species:odorant 10   1402.9 13.4043 < 2e-16 ***
sex:species     10   1187.5  1.6575 0.08708 .
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```


Residuals vs Fitted
Residuals
Predicted values
58
352
605
−4
−2
0
2
4
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
Scale−Location
√|Std. deviance resid.|
Predicted values
606
352
125
0
1
2
3
4
5
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
Normal Q−Q
Std. deviance resid.
606
352
125
−20
0
10
30
−3
−2
−1
0
1
2
3
Residuals vs Leverage
Std. Pearson resid.
58
352
125
Cook's distance
−2
0
2
4
0.00
0.05
0.10
0.15


estimated probability of choosing 30CO2
with 95% confidence bands
ana
atr
ere
mau
mel
pse
sec
sim
tei
vir
yak
females
males
0.0
0.2
0.4
0.6
0.8
1.0


estimated probability of choosing CO2
with 95% confidence bands
ana
atr
ere
mau
mel
pse
sec
sim
tei
vir
yak
○ females
○ males
0.0
0.2
0.4
0.6
0.8
1.0

```
> newdata <- data.frame(species=rep(levels(fly$species),4),
+          odorant=rep(levels(fly$odorant)[1:2],rep(22,2)),
+          sex=rep(rep(levels(fly$sex),2),rep(11,4)))
> newdata
   species odorant     sex
1      ana   30CO2 females
2      atr   30CO2 females
3      ere   30CO2 females
4      mau   30CO2 females
5      mel   30CO2 females
6      pse   30CO2 females
7      sec   30CO2 females
8      sim   30CO2 females
9      tei   30CO2 females
10     vir   30CO2 females
11     yak   30CO2 females
12     ana   30CO2   males
13     atr   30CO2   males
14     ere   30CO2   males
15     mau   30CO2   males
16     mel   30CO2   males
```

```
23      ana     CO2 females
24      atr     CO2 females
25      ere     CO2 females
26      mau     CO2 females
27      mel     CO2 females
28      pse     CO2 females
29      sec     CO2 females
30      sim     CO2 females
31      tei     CO2 females
32      vir     CO2 females
33      yak     CO2 females
34      ana     CO2   males
35      atr     CO2   males
36      ere     CO2   males
37      mau     CO2   males
38      mel     CO2   males
39      pse     CO2   males
40      sec     CO2   males
41      sim     CO2   males
42      tei     CO2   males
43      vir     CO2   males
```

```
> predict(model4,newdata,type="link")
          1           2           3           4           5
-1.58789551 -1.14469372 -1.26487696 -1.14101650 -2.76586374 -0.1077
          7           8           9          10          11
-1.90097360 -0.91699408 -1.72012424 -0.89185179 -1.78389658 -1.0172
         13          14          15          16          17
-1.06650110 -1.29566564 -1.25030454 -2.16842944  0.08781449 -1.7959
         19          20          21          22          23
-0.91001993 -1.47044203 -0.89969326 -1.78744176 -2.55428808 -0.9039
         25          26          27          28          29
-0.72774118 -0.85332683 -2.19052045 -0.65510800 -0.78579246 -0.4694
         31          32          33          34          35
-1.61457993 -0.59147161 -0.80167681 -1.98367468 -0.82573216 -0.7585
         37          38          39          40          41
-0.96261487 -1.59308615 -0.45953795 -0.68077358 -0.46245135 -1.3648
         43          44
-0.59931308 -0.80522198
```

```
> predict(model4,newdata,type="response")
         1          2          3          4          5          6
0.16968019 0.24145963 0.22013549 0.24213378 0.05919695 0.47308714 0
         8          9         10         11         12         13
0.28557077 0.15185516 0.29072783 0.14382265 0.26555715 0.25606905 0
        15         16         17         18         19         20
0.22264743 0.10262158 0.52193952 0.14234421 0.28699576 0.18687544 0
        22         23         24         25         26         27
0.14338666 0.07213894 0.28824462 0.32569061 0.29873544 0.10060499 0
        29         30         31         32         33         34
0.31307282 0.38475223 0.16595372 0.35629727 0.30966695 0.12092766 0
        36         37         38         39         40         41
0.31896554 0.27635496 0.16895014 0.38709544 0.33608867 0.38640446 0
        43         44
0.35450087 0.30890960
```

Compute an approx. 95% confidence range

```
> case <- data.frame(species="mel",odorant="CO2",sex="males")
> (pred <- predict(model4,case,type="link",se.fit=TRUE) )
$fit
-1.593086
$se.fit
[1] 0.1327248
$residual.scale
[1] 1.328106
> invlink <- function(x) {    ## inverse link function
+   1/(1+exp(-x))
+ }
> invlink(pred$fit)     ## prediction
0.1689501
> invlink(pred$fit-2*pred$se.fit)   ## lower bound
0.1348738
> invlink(pred$fit+2*pred$se.fit)   ## upper bound
0.2095506
```

This can be done simultaneously for a whole data frame (e.g. newdata) instead just for one on case (in our example mel/CO2/males)

This can be done simultaneously for a whole data frame (e.g. newdata) instead just for one on case (in our example mel/CO2/males)

Should be done on the linear predictor (“link”) scale and not on the response scale because it is based on a normal distribution approximation, which is only (more or less) valid on the linear predictor scale. (Remember: for a normal distribution, $>$ 95% are within the 2σ-bounds around the mean.)

Contents

S. Foitzik, I.M. Kureck, M.H. Rüger, D. Metzler (2010) Alternative reproductive tactics and the influence of local competition on sex allocation in the ant *Hypoponera opacior*. *Behavioral Ecology and Sociobiology*, to appear.

How does the ratio of queens and males produced by an ant nest depend on the nest size?

- Winged sexuals were observed in June, unwinged sexuals in August.
- New queens and workers have more genetic material in common than new males and workers.
- Queens are larger than males and thus more costly to produce.
- Other factors: local resource competition, local mate competition...

Variables in the ants data set.

Nest.size	number of workers in the nest
Puppen	pupae produced by the nest
New.Males	new males produced by the nest
New.Queens	new queens produced by the nest
month	6=June, 8=August

(Many more variables in full dataset)

```
> str(ants)
'data.frame': 229 obs. of  5 variables:
 $ Puppen     : int  71 16 7 6 12 13 330 12 180 0 ...
 $ Nest.size  : int  39 6 5 2 5 4 18 9 47 10 ...
 $ New.Males  : int  0 1 3 0 0 0 2 2 0 0 ...
 $ New.Queens: int  1 3 9 0 2 0 2 1 0 0 ...
 $ month      : int  6 6 6 6 6 6 6 6 6 6 ...
> attach(ants)
> productivity <-  ( Puppen +  New.Males +
                                   New.Queens )/ (Nest.size)
```

```
> M0 <- glm(cbind(New.Queens,New.Males)~(as.factor(month)
+           +Nest.size+productivity)^2,family=binomial)
> summary(M0)
[...]
                           Estimate Std. Error z value Pr(>|z|)
(Intercept)                    -0.428  0.3175  -1.3 0.1776
as.factor(month)8              -0.205  0.3664  -0.5 0.5757
Nest.size                       0.066  0.0177   3.7 0.0001 ***
productivity                    0.002  0.0178   0.1 0.8670
as.factor(month)8:Nest.size    -0.030  0.0171  -1.8 0.0710 .
as.factor(month)8:productivity -0.016  0.0165  -0.9 0.3225
Nest.size:productivity         -0.000  0.0007  -0.5 0.5988
[..]
    Null deviance: 494.61  on 138  degrees of freedom
Residual deviance: 354.96  on 132  degrees of freedom
  (10 observations deleted due to missingness)
AIC: 529.5
```

We already have lots of parameters and interactions in the model, but the residual deviance of 354.96 is still to high for 132 degrees of freedom.

We already have lots of parameters and interactions in the model, but the residual deviance of 354.96 is still to high for 132 degrees of freedom.

⇒ Use *overdispersed* binomial (quasibinomial).

```
> M1 <- glm(cbind(New.Queens,New.Males)~(as.factor(month)
+           +Nest.size+productivity)^2,family=quasibinomial)
> summary(M1)
[..]
                         Estimate Std. Error t value Pr(>|t|)
(Intercept)                    -0.4281  0.470  -0.9   0.36
as.factor(month)8              -0.2050  0.542  -0.3   0.70
Nest.size                       0.0667  0.026   2.5   0.01 *
productivity                    0.0029  0.026   0.1   0.91
as.factor(month)8:Nest.size    -0.0309  0.025  -1.2   0.22
as.factor(month)8:productivity -0.0164  0.024  -0.6   0.50
Nest.size:productivity         -0.0003  0.001  -0.3   0.72
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for quasibinomial family 2.190267)

    Null deviance: 494.61  on 138  degrees of freedom
Residual deviance: 354.96  on 132  degrees of freedom
  (10 observations deleted due to missingness)
AIC: NA
```

- Less significance now.
- Residual deviance still the same, but no reason to worry for overdispersed models
- AIC not available anymore; that's a real pity!

```
> drop1(M1,test="F")
Single term deletions

Model:
cbind(New.Queens, New.Males) ~ (as.factor(month)
       + Nest.size + productivity)^2
                              Df Deviance F value  Pr(F)
<none>                             354.96
as.factor(month):Nest.size     1   358.39  1.2754 0.2608
as.factor(month):productivity  1   355.94  0.3642 0.5472
Nest.size:productivity         1   355.24  0.1035 0.7482
```

Model selection when AIC is not available.

- Apply backward model selection strategy: apply drop1 and remove the variable with the highest p-value. Apply drop1 on the reduced model and repeat this again and again until you only variables are left which are significant or almost significant.
- Variables will not be removed if they are involved in interactions, because drop1 won't show those variables.
- Do not remove a variable if there is a good biological reason why it should be in the model.

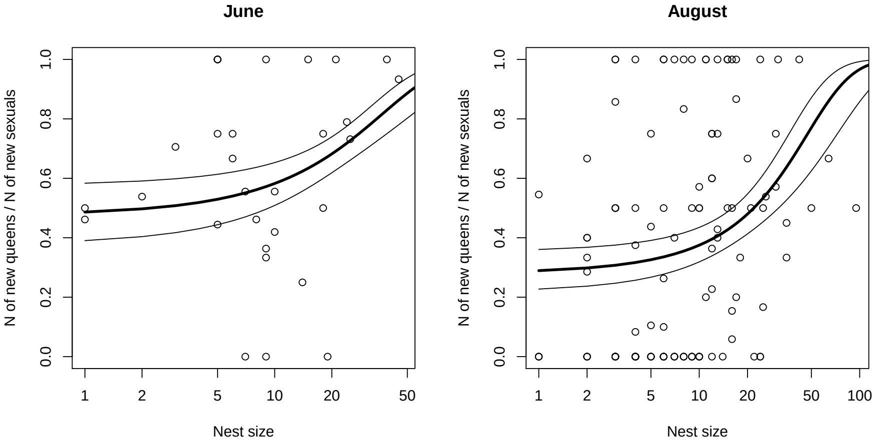
```
> M2 <- update(M1,~.-as.factor(month):productivity)
> drop1(M2,test="F")
Single term deletions

Model:
cbind(New.Queens, New.Males) ~ as.factor(month)
    + Nest.size + productivity + as.factor(month):Nest.si
    + Nest.size:productivity
                           Df Deviance F value  Pr(F)
<none>                         355.94
as.factor(month):Nest.size  1   358.86  1.0911 0.2981
Nest.size:productivity      1   355.96  0.0067 0.9349
```

```
> M3 <- update(M2,~.-Nest.size:productivity)
> drop1(M3,test="F")
Single term deletions

Model:
cbind(New.Queens, New.Males) ~ as.factor(month) +
    Nest.size + productivity +
    as.factor(month):Nest.size
                          Df Deviance F value  Pr(F)
<none>                         355.96
productivity               1   358.57  0.9832 0.3232
as.factor(month):Nest.size  1   359.40  1.2952 0.2571
```

```
> M4 <- update(M3,~.-productivity )
> drop1(M4,test="F")
Single term deletions

Model:
cbind(New.Queens, New.Males) ~ as.factor(month) +
    Nest.size + as.factor(month):Nest.size
                            Df Deviance F value  Pr(F)
<none>                           358.57
as.factor(month):Nest.size  1   360.07  0.5626 0.4545
```

```
> M5 <- update(M4,~.-as.factor(month):Nest.size)
> drop1(M5,test="F")
Single term deletions

Model:
cbind(New.Queens, New.Males) ~ as.factor(month) + Nest.size
                 Df Deviance F value     Pr(F)
<none>                360.07
as.factor(month)  1   399.32  14.828 0.0001806 ***
Nest.size         1   417.47  21.684 7.559e-06 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
> summary(M5)

Call:
glm(formula = cbind(New.Queens, New.Males) ~ as.factor(month) +
    Nest.size, family = quasibinomial)

Deviance Residuals:
    Min        1Q    Median        3Q       Max
-3.5049   -0.8569   0.0000    0.3521    4.2843

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)       -0.156142   0.236048  -0.661    0.509
as.factor(month)8 -0.839253   0.202793  -4.138 6.10e-05 ***
Nest.size          0.045656   0.009749   4.683 6.76e-06 ***
```


June
August
N of new queens / N of new sexuals
Nest size
0.0
0.2
0.4
0.6
0.8
1.0
1
2
5
10
20
50
100

```
plot(Nest.size[month==6],
  New.Queens[month==6]/(New.Males[month==6]+New.Queens[month==6]),
  main="June", log="x", xlab="Nest size",
  ylab="N of new queens / N of new sexuals")

hypotheticaljune <- data.frame(month=6,Nest.size=0:200)

pred <- predict(M5,hypotheticaljune,type="link",se.fit=TRUE)

lines(0:200,1/(1+exp(-pred$fit)),lwd=3)

lines(0:200,1/(1+exp(-(pred$fit+2*pred$se.fit))))

lines(0:200,1/(1+exp(-(pred$fit-2*pred$se.fit))))
```

Contents

GLMs and their links (canonical links first)

Poisson	log(μ), μ, $\sqrt{\mu}$
binomial	logit, probit, cloglog
gaussian	μ
Gamma	$-1/\mu$, μ, log(μ)
inverse gaussian	$-2/\mu^2$

Also interesting: **negative binomial** as alternative to overdispersed Poisson.