

Generalized linear model validation

Bert van der Veen

Department of Mathematical Sciences, NTNU

Outline

- ▶ Models so far
- ▶ Other GLMs
- ▶ Other useful models
- ▶ Some other approaches (**tomorrow**)

Generalised linear models so far

- ▶ Linear regression
- ▶ Binomial regression
- ▶ Poisson regression
- ▶ Negative-binomial regression (for fixed dispersion)

Other GLMS not covered so far

- ▶ Gamma regression
- ▶ Log-normal regression
- ▶ Inverse Gaussian regression
- ▶ Tweedie regression
- ▶ Multinomial regression
- ▶ Ordinal regression
- ▶ Conway-Maxwell-Poisson regression (not included here)

Other useful models

- ▶ beta regression
- ▶ beta-binomial regression (overdispersed binomial, not covered here)
- ▶ Zero-inflated models
- ▶ Hurdle models

Other useful methods

more tomorrow

GLMs for positive continuous data

Data bounded at zero $y_i > 0$, and might or might not include zeros.

- ▶ Usually (right) skewed
- ▶ Variance increases with mean

E.g.,:

- ▶ Biomass
- ▶ Size measurements
- ▶ Nitrogen in soil
- ▶ Amount of rainfall
- ▶ Insurance claims

Options:

- ▶ Log-normal regression
- ▶ Inverse Gaussian GLM
- ▶ Gamma GLM
- ▶ **Tweedie GLMs**

Our example data: soil nitrogen

Originally by Lane et al. (2002)

- ▶ Soil nitrogen in kilograms per hectare
- ▶ Nitrogen fertilizer dose in kilograms per hectare

Fert	Source	SoilN
0	0	4.53
0	0	5.46
0	0	4.77
48	0	6.17
48	0	9.30
48	0	8.29
96	0	11.30
96	0	16.58

Log-normal regression

Connected to the normal distribution, in the sense:

$$\begin{aligned}
 \log(y)_i &\sim \mathcal{N}(\mu_i, \sigma^2) \\
 y_i &\sim \text{Lognormal}(\mu_i, \sigma^2)
 \end{aligned}
 \tag{1}$$

- ▶ So, easy to fit!
- ▶ I don't generally recommend response transformations
- ▶ Two parameters: μ_i and σ^2
- ▶ $\mathbb{E}(y_i|x_i) = \exp(\mu_i + \frac{\sigma^2}{2})$
- ▶ $\text{var}(y_i|x_i) = \{\exp(\sigma^2) - 1\} \exp(2\mu_i + \sigma^2)$

We are not modeling the mean but the location parameter μ_i

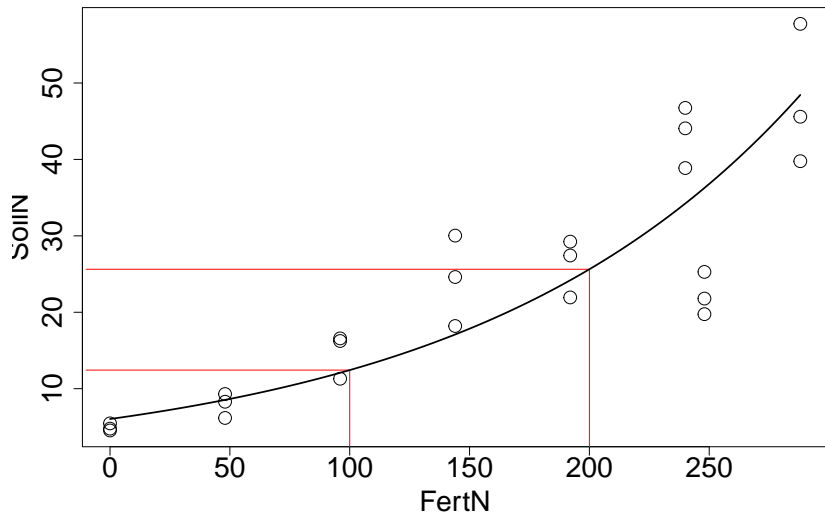
Log-normal regression in R

```
model1 <- lm(log(SoilN)~Fert, data = nitrogen)
fitted <- exp(predict(model1)) #We need to backtransform ourselves
cbind(est=coef(model1), confint(model1))
```

##	est	2.5 %	97.5 %
## (Intercept)	1.796792406	1.553746377	2.039838436
## Fert	0.007234334	0.005914211	0.008554456

- ▶ Soil nitrogen increases with supplied nitrogen
- ▶ $\log(2)/0.007234334 \approx 95$
- ▶ Soil nitrogen doubles for almost every 100 kg/hectare fertilizer

Log-normal regression: results



Inverse Gaussian GLM

- ▶ Sharper peak and heavier tails than log-normal
- ▶ Often used in presence of extreme events

$$\begin{aligned}
 \frac{1}{\mathbb{E}(y_i|\mathbf{x}_i)^2} &= \alpha + \mathbf{x}_i\beta \\
 \mathbb{E}(y_i|\mathbf{x}_i) &= \frac{1}{(\alpha + \mathbf{x}_i\beta)^2}
 \end{aligned}
 \tag{2}$$

- ▶ parameters: location μ_i and dispersion ϕ
- ▶ canonical link: squared inverse
- ▶ alternatively inverse, log-link

$$\text{var}(y_i|\mathbf{x}_i) = \frac{\mu_i^3}{\phi}$$

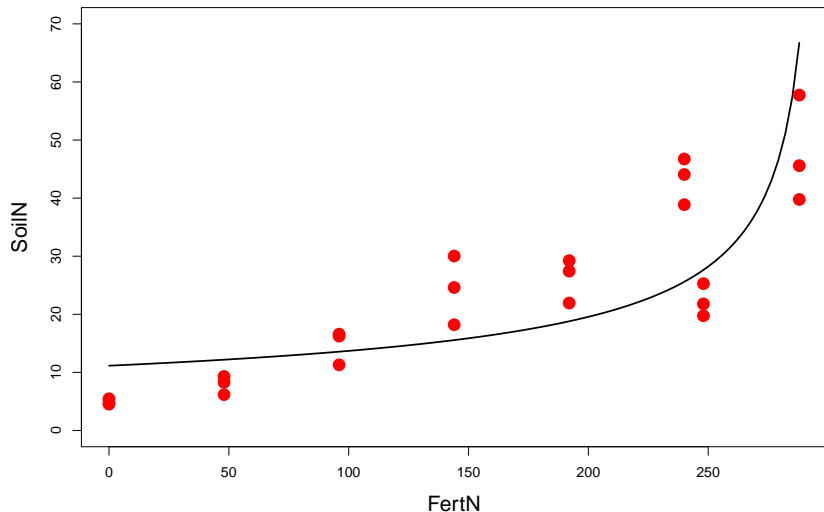
Inverse Gaussian GLM in R

```
model2 <- glm(SoilN~Fert, data = nitrogen, family = "inverse.gaussian")
summary(model2)
```

```
##
## Call:
## glm(formula = SoilN ~ Fert, family = "inverse.gaussian", data = nitrogen)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.039e-03  1.728e-03   4.652 0.000123 ***
## Fert        -2.714e-05  6.202e-06  -4.375 0.000241 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for inverse.gaussian family taken to be 0.01367622)
##
##      Null deviance: 0.72554  on 23  degrees of freedom
## Residual deviance: 0.37002  on 22  degrees of freedom
## AIC: 185.12
##
## Number of Fisher Scoring iterations: 6
```

Interpretation is -very- difficult with inverse-type link functions

Inverse Gaussian GLM: results



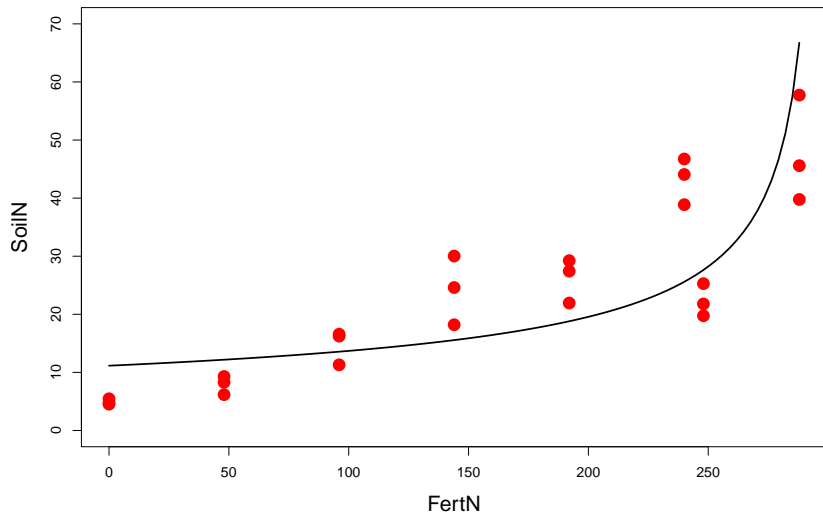
Gamma GLM

- ▶ (negative) Inverse link is canonical
 - ▶ provide starting positive values mustart
- ▶ Log-link is more stable

```
model3 <- glm(SoilN~Fert, data = nitrogen, family = "inverse.gaussian")
summary(model3)
```

```
##
## Call:
## glm(formula = SoilN ~ Fert, family = "inverse.gaussian", data = nitrogen)
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.039e-03  1.728e-03   4.652 0.000123 ***
## Fert        -2.714e-05  6.202e-06  -4.375 0.000241 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for inverse.gaussian family taken to be 0.01367622)
##
##      Null deviance: 0.72554  on 23  degrees of freedom
## Residual deviance: 0.37002  on 22  degrees of freedom
## AIC: 185.12
##
```

Gamma GLM: results



Tweedie distributions

Many of the distributions so far are special forms of the Tweedie family of distributions:

$$\text{var}(y_i | \mathbf{x}_i) = \mu_i^\xi \phi$$

- ▶ Normal distribution $\xi = 0$
- ▶ Poisson distribution $\xi = 1$

- ▶ Gamma distribution $\xi = 2$
- ▶ Inverse Gaussian distribution $\xi = 3$

With Tweedie we can:

- ▶ Analyse positive continuous data (with zeros!)
- ▶ Counts

Tweedie GLM in R: finding ξ

```
out.est <- tweedie::tweedie.profile(SoilN ~ Fert, xi.vec = se
```

```
## 2.01 2.06 2.11 2.16 2.21 2.26 2.31 2.36 2.41 2.46 2.51 2.5
## .....Done.
```

```
out.est2 <- tweedie::tweedie.profile(SoilN ~ Fert, xi.vec =
```

```
## 3.022245 3.032245 3.042245 3.052245 3.062245 3.072245 3.08
## .....
```

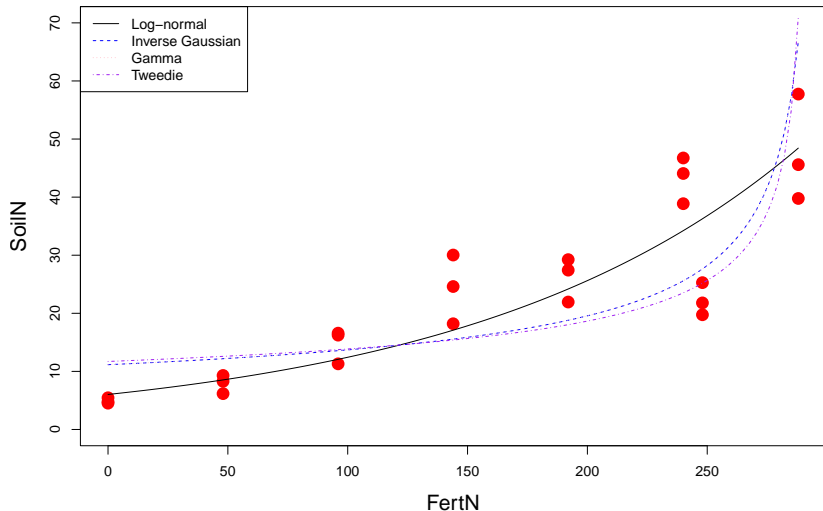
```
plot(out.est2, xlab = expression(xi), ylab="Log-likelihood",
abline(v=out.est2$xi.max, col = "red")
```

Tweedie GLM in R: fitting

```
model4 <- glm(SoilN~Fert, data = nitrogen, family = statmod::tweedie(var.power = out.est2$xi.max))
summary(model4)
```

```
##
## Call:
## glm(formula = SoilN ~ Fert, family = statmod::tweedie(var.power = out.est2$xi.max),
##      data = nitrogen)
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.175e-03  6.002e-04   3.624  0.00150 **
## Fert        -7.467e-06  2.113e-06  -3.535  0.00186 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Tweedie family taken to be 0.004092795)
##
##      Null deviance: 0.20212  on 23  degrees of freedom
## Residual deviance: 0.12582  on 22  degrees of freedom
## AIC: NA
##
## Number of Fisher Scoring iterations: 6
```

All together



Conclusion

- ▶ They can be relatively similar, but link function matters.
- ▶ **Mind the tail behavior**

Tweedie for flexible modeling of positive data (potentially with zeros)

GLMs for categorical data

Binomial models assume **dichotomy**. Alternative: **polychotomy**, multiple groups.

Multinomial

Gives the number of successes for a set of outcomes (instead of either success or fail)



Figure 1: We roll a dice many times, and count the number of times it hits each side. If we only throw the dice once, the throw follows a **categorical distribution**.

Multinomial logistic regression

The model we assume is:

$$\begin{aligned} \text{pr}(y_{ij}) = \pi_{ij} &= \frac{\exp(\eta_{ij})}{1 + \exp(\eta_{ij})} \\ \log\left(\frac{\pi_{ij}}{1 - \pi_{ij}}\right) &= \eta_{ij} = \mathbf{x}_{ij}\boldsymbol{\beta}_j \end{aligned} \tag{3}$$

- ▶ E.g., probability of observing species at a site.
- ▶ We could fit multiple binomial models
 - ▶ Gets a bit messy: $y_{i1}|N_i \sim \text{Bin}(\pi_{i1}; N_i)$
 - ▶ $y_{i2}|N_i - y_{i1} \sim \text{Bin}(\pi_{ij}; N_i - y_{i1})$
- ▶ Equivalent to using a Poisson regression with fixed total

Cumulative link models

- ▶ Add order to the probabilities

Ordinal: proportional odds

More commonly, the categories are ordered.

$$\text{pr}(y_i \leq j) = \sum_{k=1}^j \text{pr}(y_k) = \frac{\exp(\theta_j + \beta^\top \mathbf{x}_i)}{1 + \exp(\theta_j + \beta^\top \mathbf{x}_i)} \quad (4)$$

▶ E.g., cover classes

Ordinal: proportional odds

Referred to as “proportional odds” because:

$$\frac{\pi_{ij}/(1 - \pi_{ij})}{\pi_{i+1j}/(1 - \pi_{i+1j})} = \exp\{\beta(\mathbf{x}_{ij} - \mathbf{x}_{i+1j})\} \quad (5)$$

the log-odds is proportional to the difference in the covariates.

Ordinal example: vegetation data

```
data(dune,dune.env,package="vegan")
data <- cbind(y=as.factor(dune$Bracruta), dune.env)
(model5 <- MASS::polr(y~A1, data = data))
```

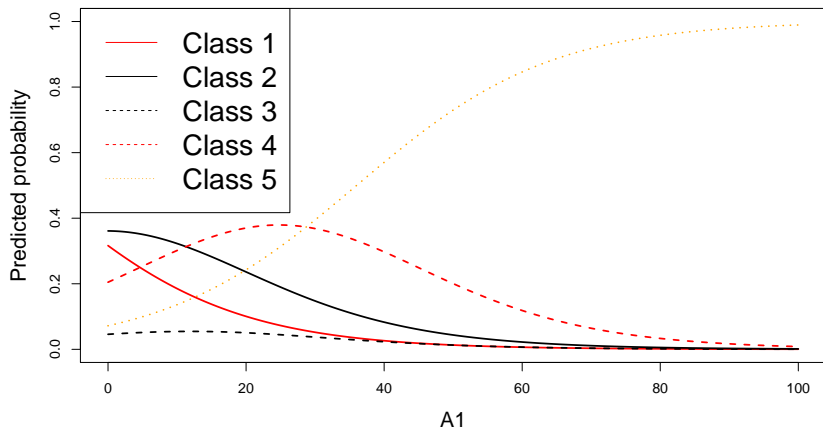
```
## Call:
## MASS::polr(formula = y ~ A1, data = data)
##
## Coefficients:
##          A1
## 0.07104338
##
## Intercepts:
##          0|2          2|3          3|4          4|6
## -0.7704179  0.7426109  0.9615655  2.5585698
##
## Residual Deviance: 57.48771
## AIC: 67.48771
```

Example: vegetation data

```
predict(model5,type="probs")
```

##		0	2	3	4	6
## 1		0.2750122	0.3576581	0.04925726	0.2317706	0.08630192
## 2		0.2652086	0.3558300	0.05000602	0.2386505	0.09030488
## 3		0.2542818	0.3532918	0.05080543	0.2465371	0.09508384
## 4		0.2556313	0.3536348	0.05070893	0.2455506	0.09447432
## 5		0.2282902	0.3449388	0.05251291	0.2662255	0.10803261
## 6		0.2542818	0.3532918	0.05080543	0.2465371	0.09508384
## 7		0.2750122	0.3576581	0.04925726	0.2317706	0.08630192
## 8		0.2556313	0.3536348	0.05070893	0.2455506	0.09447432
## 9		0.2624490	0.3552399	0.05021166	0.2406205	0.09147894
## 10		0.2679868	0.3563901	0.04979663	0.2366821	0.08914442
## 11		0.2652086	0.3558300	0.05000602	0.2386505	0.09030488
## 12		0.2346085	0.3472870	0.05212735	0.2613202	0.10465704

Ordinal example: vegetation data



Note that ordering has introduced some shapes.

Proportions: brief mentioning

Sometimes, mistakenly, logistic regression is referred to as “modeling proportions”

- ▶ Binomial responses are dichotomous (either 0 or 1)
- ▶ Proportions usually include 0 and/or 1, but also everything between

$$y_i \sim \text{Beta}(\mu_i, \phi) \quad (6)$$

- ▶ “beta regression”, which is not a GLM
- ▶ various packages implement it
- ▶ Require data between 0 and 1 (i.e., without 0 or 1)

When there are zeros and/or ones..zero-inflated models or hurdle models (ZIB or ordered beta regression).

Excess zeros

When there are more zeros than the selected distribution can generate.

- ▶ Can be accounted for with additional model terms (covariate)
- ▶ Alternatively:
 - ▶ Zero-inflated models
 - ▶ Hurdle models
 - ▶ Negative-binomial models: can handle excess zeros very well
- 1) Zero-inflated models: another process generates additional zeros (e.g., colonisation vs. abundance)
- ▶ e.g., Species abundance
- 2) Hurdle models: only one of the two processes generates values below the hurdle

Zero-inflation (Poisson)

$$y_i \sim ZIP(\lambda_{ij}, \pi_i) \quad (7)$$

$$y_i | \nu_j \sim \begin{cases} p \text{Pois}(\lambda_i) & \text{if } y_i > 0 \\ \begin{cases} p \exp(-\lambda_i) & \text{if } \nu_i = 1 \\ 1 - p & \text{if } \nu_i = 0 \end{cases} & \text{if } y_i = 0 \end{cases} \quad (8)$$

- ▶ can choose to separately model p as a logistic regression
- ▶ quadratic mean-variance as in NB

Example: Cod parasite data

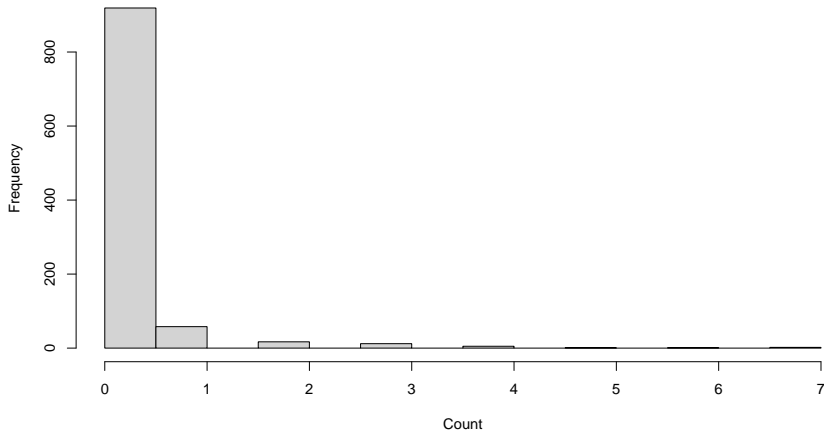
Mistletoe data covered in Zuur (2021)

- ▶ Count of mistletoe infections in a tree stand
- ▶ DBH, tree infected or not, size of largest mistletoe

CI_stem	N12	No.of.mistletoes	Stand	ID	DBH	Distance.from
5.343255	36	1	äobes	921	26	
4.771435	32	2	äobes	923	27	
5.856576	32	0	äobes	920	25	
5.093324	39	0	äobes	916	25	
6.452976	32	0	äobes	918	24	
5.598422	32	0	äobes	917	26	
5.301989	32	0	äobes	919	28	
6.384254	29	0	äobes	913	20	

Example: Mistletoe infections

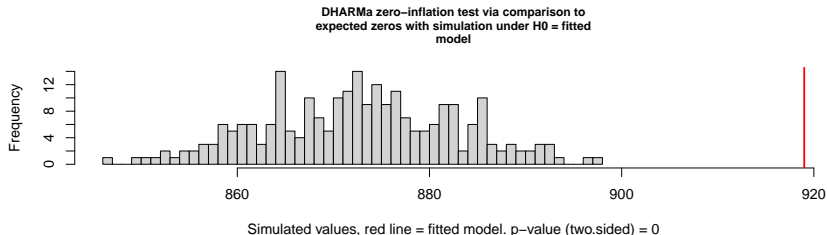
Mistletoe infections



Example: Mistletoe infections

```
model6 <- glm(No.of.mistletoes~DBH, family="poisson", data = data)
```

```
DHARMA::testZeroInflation(model6)
```

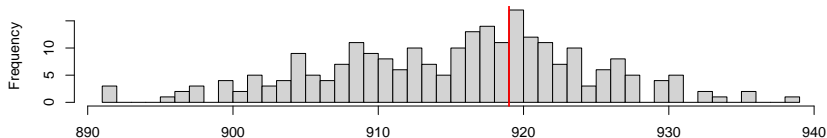


```
##
## DHARMA zero-inflation test via comparison to expected zeros with
## simulation under H0 = fitted model
##
## data: simulationOutput
## ratioObsSim = 1.0525, p-value < 2.2e-16
```

Example: Mistletoe infections

```
model7 <- glmmTMB::glmmTMB(No.of.mistletoes~DBH,ziformula=~1, family="poisson", data = data)
```

DHARma zero-inflation test via comparison to expected zeros with simulation under H0 = fitted model



Simulated values, red line = fitted model. p-value (two.sided) = 0.84

```
##
## DHARma zero-inflation test via comparison to expected zeros with
## simulation under H0 = fitted model
##
## data: simulationOutput
## ratioObsSim = 1.0035, p-value = 0.84
## alternative hypothesis: two.sided
```

Summary

- ▶ At least not mentioned: Generalized and Conway-Maxwell Poisson
- ▶ Ordered beta
- ▶ Truncation