How are you dealing with your COUNT data?

Azuh, Ikedichi Edward

Programme:
Part 1: Talking Statistics with R

Date: 16.12.2022

Outline

1. Introduction
   - Prelim
   - Brief background
2. Dispersion
   - Equi-, Over- and Under-dispersion
   - Counts with zeros
3. Methods
   - Flowchart
   - Models
     - Poisson
     - Negative Binomial
     - Quasi-Poisson
     - COM-Poisson
     - Generalized-Poisson
     - Hurdle/Truncated Poisson
     - Hurdle/Truncated Negative Binomial
     - Zero Inflated Poisson
     - Zero Inflated Negative Binomial

Data partition

```
Library(base)
set.seed(123)
index <- sample(2, nrow(fulldata), replace = TRUE, p=c(.7,.3))
Train <- fulldata[index==1,]
Test <- fulldata[index==2,]
```

- Always a good practice in data analysis to partition your dataset into at least 2 groups/sets (Train and Test set).
- Can be in the ratios 60 :40, 70 :30, 75 :25, 80 :20 or 85 :15 in favour of the Training set depending on how big your full data is.
- If goal is not prediction, no need for partitioning.

What is a count data ?

A count data is a statistical data type describing countable quantities, data which can take only the counting numbers, non-negative integer values 0, 1, 2, 3, ... [Wikipedia]

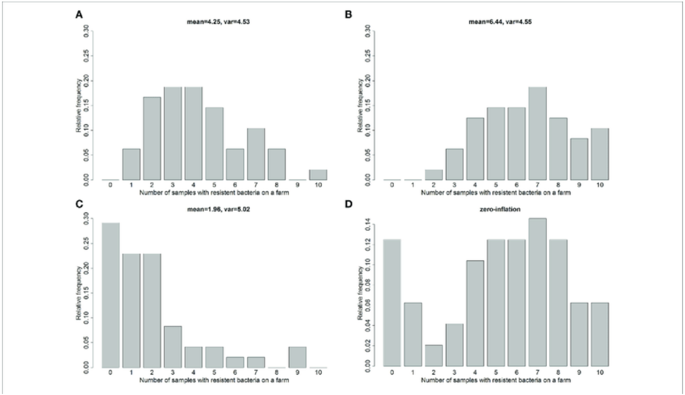


Figure – 1. Example of count data : sourced from google

Examples

Examples include : number of times a patient visits the hospital, number of items sold per day in a shop, number of graduates in a school every calender year, number of calls you receive daily, number of new members welcomed on a whatsapp group every month, etc.

Let us assume the country "Germany" want to study the effect of exchange RATE (Naira), AGE of applicants, TYPE of visa, GENDER of applicant and applicants level of EDUCation, on NUMBER of monthly visa applications received. The response of interest here is a COUNT (Number of monthly visa applications). And mathematically, the problem can be expressed as :

$$NUMBER = RATE + TYPE + AGE + GENDER + EDUC \tag{1}$$

$$NUMBER = RATE + TYPE + AGE + GENDER + (1|EDUC) \tag{2}$$

Dispersion

If our target variable really follows poisson distribution then its variance (V) should be approximately equal to its mean (μ), which is the null hypothesis of the following dispersiontest test against the alternative hypothesis that the variance of the form : V = $\mu + \alpha(\mu)$

$$\text{if } \alpha = 0 : \text{equidispersion}$$
$$\text{if } \alpha < 0 : \text{underdispersion}$$
$$\text{if } \alpha > 0 : \text{overdispersion}$$

The main discrete distribution for fitting a count data is Poisson. But the assumption of mean=variance is rarely met in the real world.

There are 3 possible scenarios :

- Equi-dispersion (Mean=Variance) : use the regular Poisson.
- Over-dispersion (Mean$<$Variance) : use the Negative Binomial or Quasi-Poisson.
- Under-dispersion (Mean$>$Variance) : use the Conway-Maxwell Poisson (COM-Poisson) or Generalized Poisson.

Does it really matter ?

The solution provided above are most appropriate to use if your count does not include zeros or excess zeros (that is, not zero inflated). But when it does, there is need to take care of the inflation or disturbances that could arise due to excess of zeros in the counts. They include (in addition to the list above) :

They include (in addition to the previous list) :

- Equi-dispersion (Mean=Variance) : use the Hurdle or Truncated Poisson, Zero-Inflated Poisson.
- Over-dispersion (Mean$<$Variance) : use the Hurdle or truncated Negative binomial, Zero-Inflated Negative binomial, COM-Poisson and Generalized Poisson.
- Under-dispersion (Mean$>$Variance) : use the Conway-Maxwell COM-Poisson and Generalized Poisson.

Flowchart

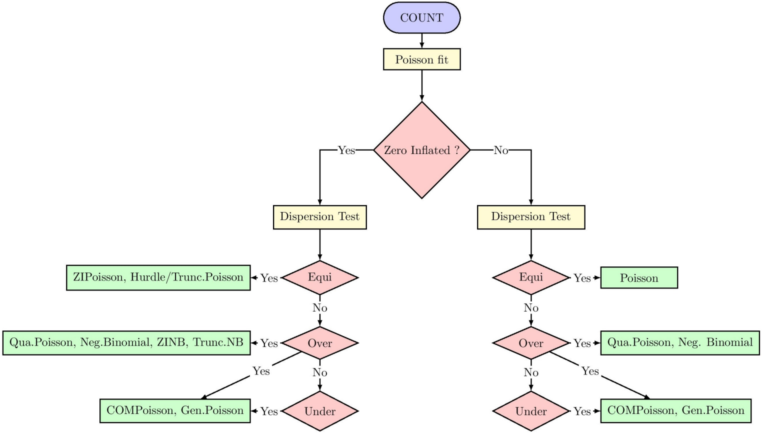


Figure – 2. Different ways to take care of a count data

An underdispersed data

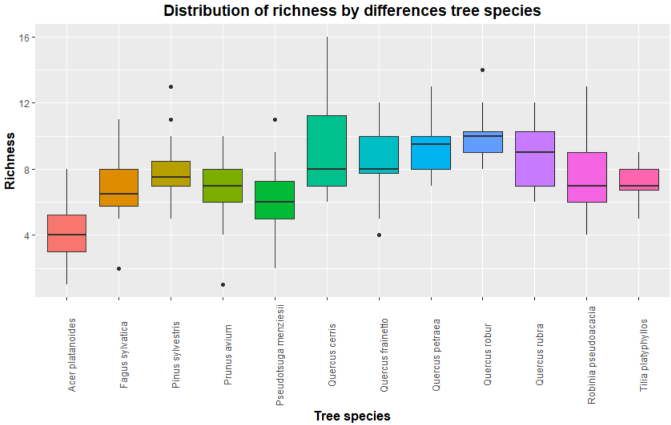


Figure – 3. Plot of Richness by Tree species

```
summary(data$Richness)
Min. 1st Qu. Median Mean 3rd Qu. Max.
1.000 6.000 8.000 7.717 9.000 16.000
```

Poisson

require(DHARMa, lme4, glmmTMB, glmmPQL)

$$P_1 < -glm(y \sim x_1 + x_2, family = "poisson", data = dd)$$
$$P_2 < -glmmTMB(y \sim x_1 + x_2 + (1|ID), family = "poisson", data = dd)$$
$$P_3 < -glmer(y \sim x_1 + x_2 + (1|ID), family = "poisson", data = dd)$$
$$P_4 < -glmmPQL(y \sim x_1 + x_2 + (1|ID), family = "poisson", data = dd)$$

Check for zero-inflation in count models :

library(performance)
check_zeroinflation(P_2)
testZeroInflation(P_2)

If the amount of observed zeros is larger than the amount of predicted zeros, the model is underfitting zeros, which indicates a zero-inflation in the data.

Dispersion test :

DHARMa : : testDispersion(Diagnose$_1$)

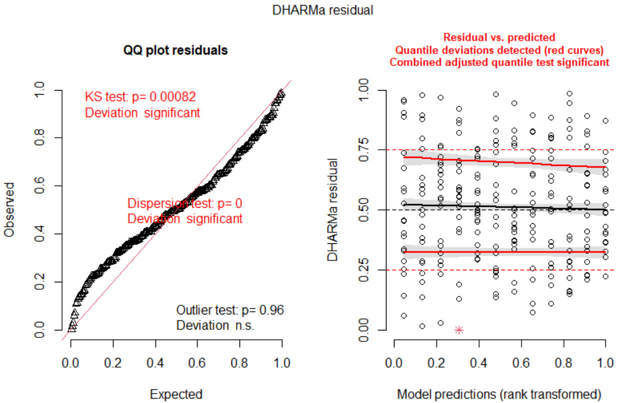
```
data: simulationOutput
dispersion = 0.48878, p-value < 2.2e-16
alternative hypothesis: two.sided
```

Diagnostic-Poisson

```
Call:
glm(formula = Richness ~ Specie, family = "poisson") \
(Dispersion parameter for poisson family taken to be 1)
```

Model diagnostics :

$\text{Diagnose}_1 < -\mathit{DHARMa} :: \mathit{simulateResiduals}(\mathit{fittedModel} = P, \mathit{plot} = T)$



Negative Binomial

require(DHARMa, MASS, glmmTMB, VGAM)

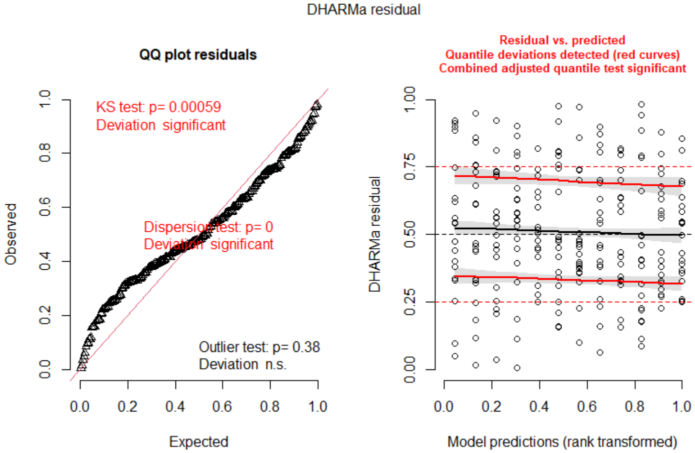
$$
\begin{aligned}
&\mathrm{NB}_1 < -\mathit{MASS} :: \mathit{glm.nb}(y \sim x_1 + x_2, \mathit{data} = \mathit{dd})\\
&\mathrm{NB}_2 < -\mathit{VGAM} :: \mathit{vglm}(y \sim x_1 + x_2, \mathit{family} = \mathit{posnegbinomial}(), \mathit{data} = \mathit{dd})\\
&\mathrm{NB}_3 < -\mathit{glmmTMB}(y \sim x_1 + x_2, \mathit{family} = \mathit{nbinom}2, \mathit{data} = \mathit{dd})\\
&\mathrm{NB}_4 < -\mathit{glmmTMB}(y \sim x_1 + x_2 + (1|\mathit{ID}), \mathit{family} = \mathit{nbinom}2, \mathit{data} = \mathit{dd})\\
&\mathrm{NB}_5 < -\mathit{MASS} :: \mathit{glmer.nb}(y \sim x_1 + x_2 + (1|\mathit{ID}), \mathit{data} = \mathit{dd})\\
&\mathrm{NB}_6 < -\mathit{vglm}(y \sim x_1 + x_2 + (1|\mathit{ID}), \mathit{family} = \mathit{posnegbinomial}(), \mathit{data} = \mathit{dd})
\end{aligned}
$$

```
Call:
glmmTMB(formula = Richness ~ Specie, family = "nbinom2")
(Dispersion parameter for nbinom2 family (): 1.91e+07)
```

Diagnostic-Negative Binomial

Model diagnostics :

$\text{Diagnose}_2 <-$ *DHARMa* :: $\textit{simulateResiduals}(\textit{fittedModel} = \textit{NB}, \textit{plot} = T)$



Quasi-Poisson

require(DHARMa, lme4, glmmTMB)

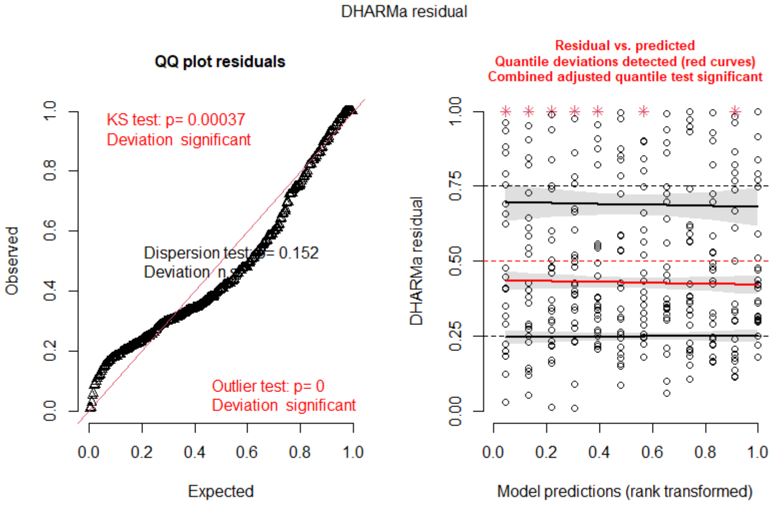
$$QP_1 < -glm(y\text{~}x_1 + x_2, family = ''quasipoisson'', data = dd)$$
$$QP_2 < -glmmTMB(y\text{~}x_1 + x_2, family = nbinom1, data = dd)$$
$$QP_3 < -glmer(y\text{~}x_1 + x_2 + (1|ID), family = ''quasipoisson'', data = dd)$$
$$QP_4 < -glmmTMB(y\text{~}x_1 + x_2 + (1|ID), family = nbinom1, data = dd)$$

```
Call:
glm(formula = Richness ~ Specie, family = "quasipoisson")
glmmTMB(formula = Richness ~ Specie, family = "nbinom1")
(Dispersion parameter for nbinom1 family (): 6.47e-09)
(Dispersion parameter for quasipoisson family : 0.5273202)
```

Diagnostic-Quasi-Poisson

Model diagnostics :

$\text{Diagnose}_3 < -\mathit{DHARMa} :: \mathit{simulateResiduals}(\mathit{fittedModel} = \mathit{QP}, \mathit{plot} = T)$



COM-Poisson

$\text{devtools : :install\_}github(''\ thomas - fung/mpcmp''\ )$
require(DHARMa, mpcmp, glmmTMB, COMPoissonReg)

$\text{COMP}_1 < -glm.cmp(y\text{~}x_1 + x_2, data = dd)$
$\text{COMP}_2 < -glmmTMB(y\text{~}x_1 + x_2, family = compois(link = ''\ log''\ ), data = dd)$
$\text{COMP}_3 < -glmmTMB(y\text{~}x_1 + x_2 + (1|ID), family = compois(), data = dd)$

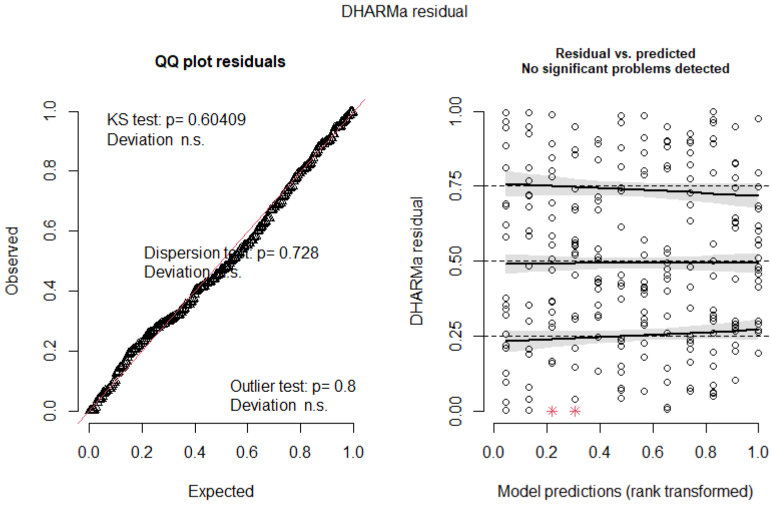
The mpcmp package also provides a range of diagnostic plots with :
autoplot(COMP_1)

```
Call:
glmmTMB(formula = Richness ~ Specie, family = compois())
(Dispersion parameter for compois family (): 0.486)
```

Diagnostic-COM-Poisson

Model diagnostics :

Diagnose$_4 < -$*simulateResiduals*(*fittedModel* = *COMP*, *plot* = *T*)

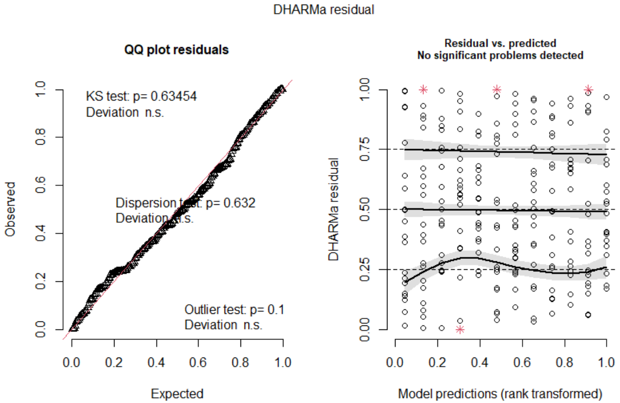


Generalized-Poisson

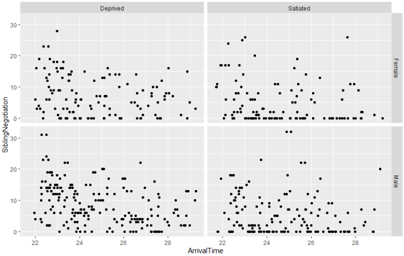
GENP <- glmmTMB($(y \sim x_1 + x_2 + (1|ID), family = genpois(), data = dd)$

```
Call:
glmmTMB(formula = Richness ~ Specie, family = genpois())
(Dispersion parameter for compois family (): 0.512)
```

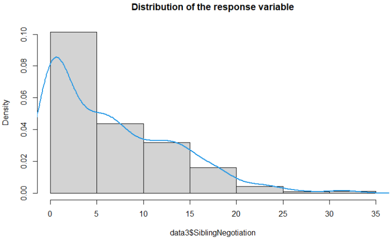
$\text{Diagnose}_5 < -simulateResiduals(fittedModel = GENP, plot = T)$



An overdispersed data with zero inflation



(a) Owls data from glmmTMB package



(b) Distribution of response

```
Poisson <- glmmTMB(SiblingNegotiation ~ FoodTreatment + SexParent +
ArrivalTime + (1|Nest), data=Owls, family = poisson)
```

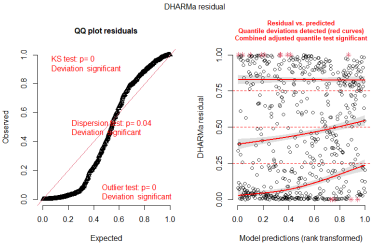
```
check_zeroinflation(Poisson)
```

```
Observed zeros: 156
Predicted zeros: 11
Ratio: 0.07
Model is underfitting zeros (probable zero-inflation).
```

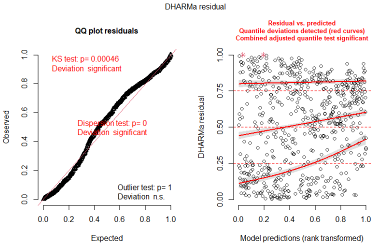
With wrong approaches

Poisson <- glmmTMB(SiblingNegotiation ~ FoodTreatment + SexParent + ArrivalTime + (1|Nest) + (1|OBlevel), family = poisson, data=Owls)

NB <- glmmTMB(SiblingNegotiation ~ FoodTreatment + SexParent + ArrivalTime + (1|Nest), family = nbinom2, data=Owls)



(c) Poisson



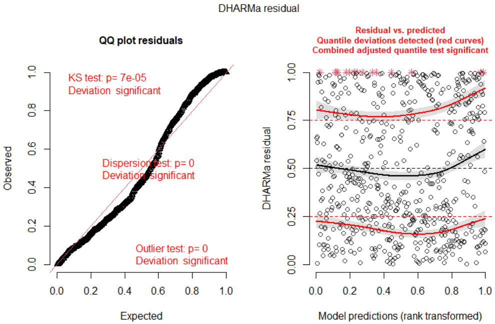
(d) Negative Binomial

Hurdle/Truncated Poisson

require(DHARMa, pscl, glmmTMB)

HURD.P$_1 < -hurdle(y{\sim}x_1 + x_2, link = ''logit'', dist = ''poisson'', data = dd)$
HURD.P$_2 < -glmmTMB(y{\sim}x_1 + x_2, family = truncated\_poisson, data = dd)$

T.P <- glmmTMB(SiblingNegotiation ~ FoodTreatment + SexParent + ArrivalTime +(1|Nest), zi = ~ FoodTreatment + SexParent + ArrivalTime, family=truncated_$poisson, data = Owls$)

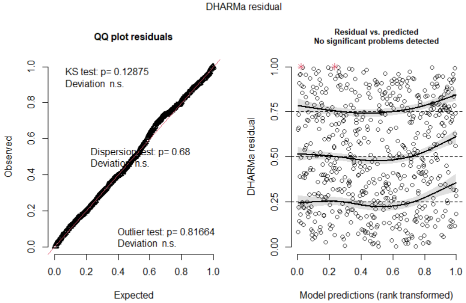


Hurdle/Truncated Negative Binomial

require(DHARMa, pscl, glmmTMB)

HURD.NB$_1 < -hurdle(y \sim x_1 + x_2|group, link = "\,logit", dist = "\,negbin", data)$
HURD.NB$_2 < -glmmTMB(y \sim x_1 + x_2 + (1|group), zi = \sim x_1 + x_2, family =$ $truncated\_nbinom2, data = dd)$

T.NB <- glmmTMB(SiblingNegotiation ~ FoodTreatment + SexParent + ArrivalTime +(1|Nest), zi = ~ FoodTreatment + SexParent + ArrivalTime, family=truncated_$nbinom2, data = Owls$)

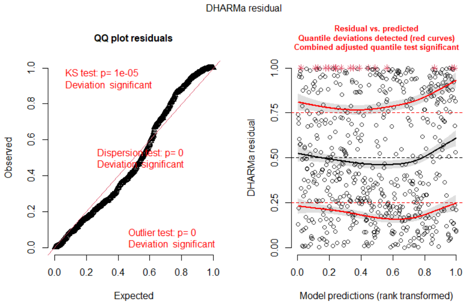


Zero Inflated Poisson

require(DHARMa, pscl, glmmTMB)

$\mathrm{ZIP}_1 < -\mathit{zeroinfl}(y\text{~}x_1 + x_2|group, link = ''\,logit'', dist = ''\,poisson'', data)$
$\mathrm{ZIP}_2 < -\mathit{glmmTMB}(y\text{~}x_1 + x_2 + (1|group), zi = \text{~}x1 + x2, family = poisson, data = dd)$

ZI.P <- glmmTMB(SiblingNegotiation ~ FoodTreatment + SexParent + ArrivalTime + (1|Nest), zi = ~ FoodTreatment + SexParent + ArrivalTime, family = poisson, data=Owls)

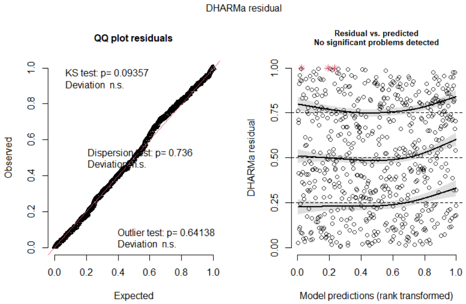


Zero Inflated Negative Binomial

require(DHARMa, pscl, glmmTMB)

$\text{ZINB}_1 < -\mathit{zeroinfl}(y\text{~}x_1 + x_2|group, link = \text{"}logit\text{"}, dist = \text{"}negbin\text{"}, data)$
$\text{ZINB}_2 < -\mathit{glmmTMB}(y\text{~}x_1 + x_2 + (1|group), zi = \text{~}x1 + x2, family = nbinom2, data = dd)$

ZI.NB <- glmmTMB(SiblingNegotiation ~ FoodTreatment + SexParent + ArrivalTime + (1|Nest), zi = ~ FoodTreatment + SexParent + ArrivalTime, family = nbinom2, data=Owls)



thank you