Logiciels pour la statistique : Modèles linéaires généralisés avec SAS®

Modèles linéaires généralisés.

On les utilise quand un modèle linéaire classique ne peut être utilisé :

- normalité des résidus du modèles linéaires loin de suivre une loi normale,
- pour des variables discrètes.

Différences avec les modèles linéaires :

- loi sur la variable à expliquer,
- on n'explique pas directement la valeur de la variable à expliquer mais une fonction de celle-ci,
- utilisation d'une fonction lien,
- pas forcément de terme d'erreur.

Historique des modèles linéaires généralisés.

- Nelder J., Wedderburn R.(1972) : Generalized Linear Models, *Journal of the Royal Statistical Society*,bf 135(3) : 370-384.

- Mc Cullagh P., Nelder J. (1989) : *Generalized Linear Models, Second Edition.* Bocas Raton : Chapman and Hall/CRC.

Composantes des GLM.

On dispose d'un échantillon de n variables aléatoires $(Y_1, \cdots, Y_n)$ indépendantes.

Les GLM sont caractérisés par 3 composantes :

- Distribution : loi de probabilité de la variable Y à expliquer.
- Prédicteur linéaire η_i : une combinaison linéaire des variables explicatives.
- Fonction lien canonique : fonction g telle que le modèle s'écrive

$$g\left(\mathbb{E}\left[Y_i\right]\right) = \eta_i$$

Distribution.

La distribution choisie doit faire parti de la famille des lois exponentielles.

Lois continues :

- loi normale,
- loi Gamma,
- inverse gaussienne
- ...

Lois discrètes :

- loi Binomiale,
- loi de Poisson,
- ...

Prédicteur.

- Combinaison linéaire des variables explicatives.
- Peut contenir des interactions entre les variables.
- On va appliquer un protocole de sélection du modèle pour déterminer le meilleur prédicteur linéaire.

Fonction lien.

- Théorie : beaucoup de fonctions candidates.
- Pratique : choix quasi-imposé lorsqu'on a déterminé la distribution de la variable à expliquer.

Remarque

Un GLM avec une distribution normale et une fonction de lien identité est en fait un modèle linéaire classique.

GLM pour des données de comptage.

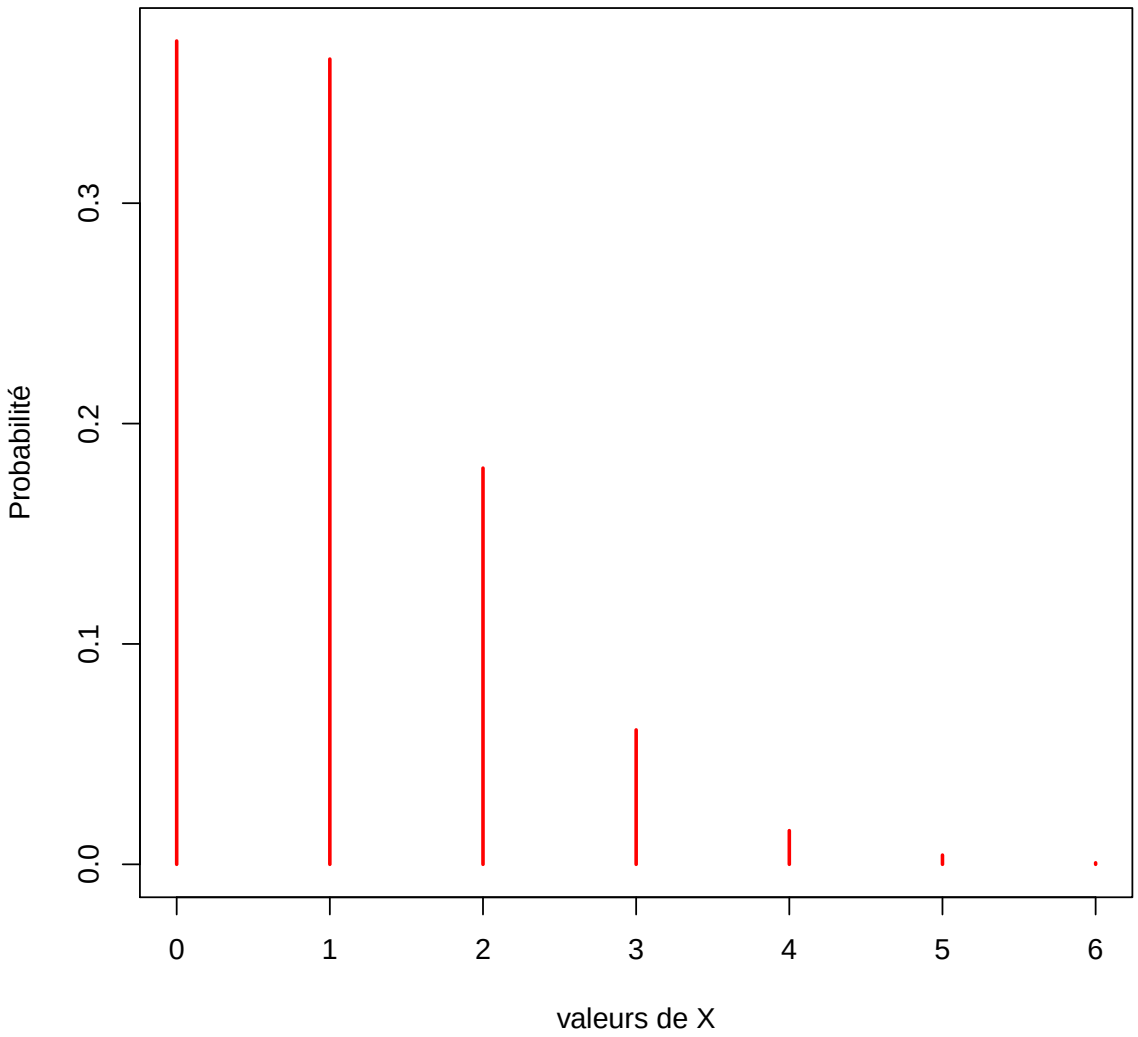
Données de comptage :

- variable discrète,
- valeurs possibles : $0, 1, 2, 3, 4, \cdots$: $\mathbb{N}$.

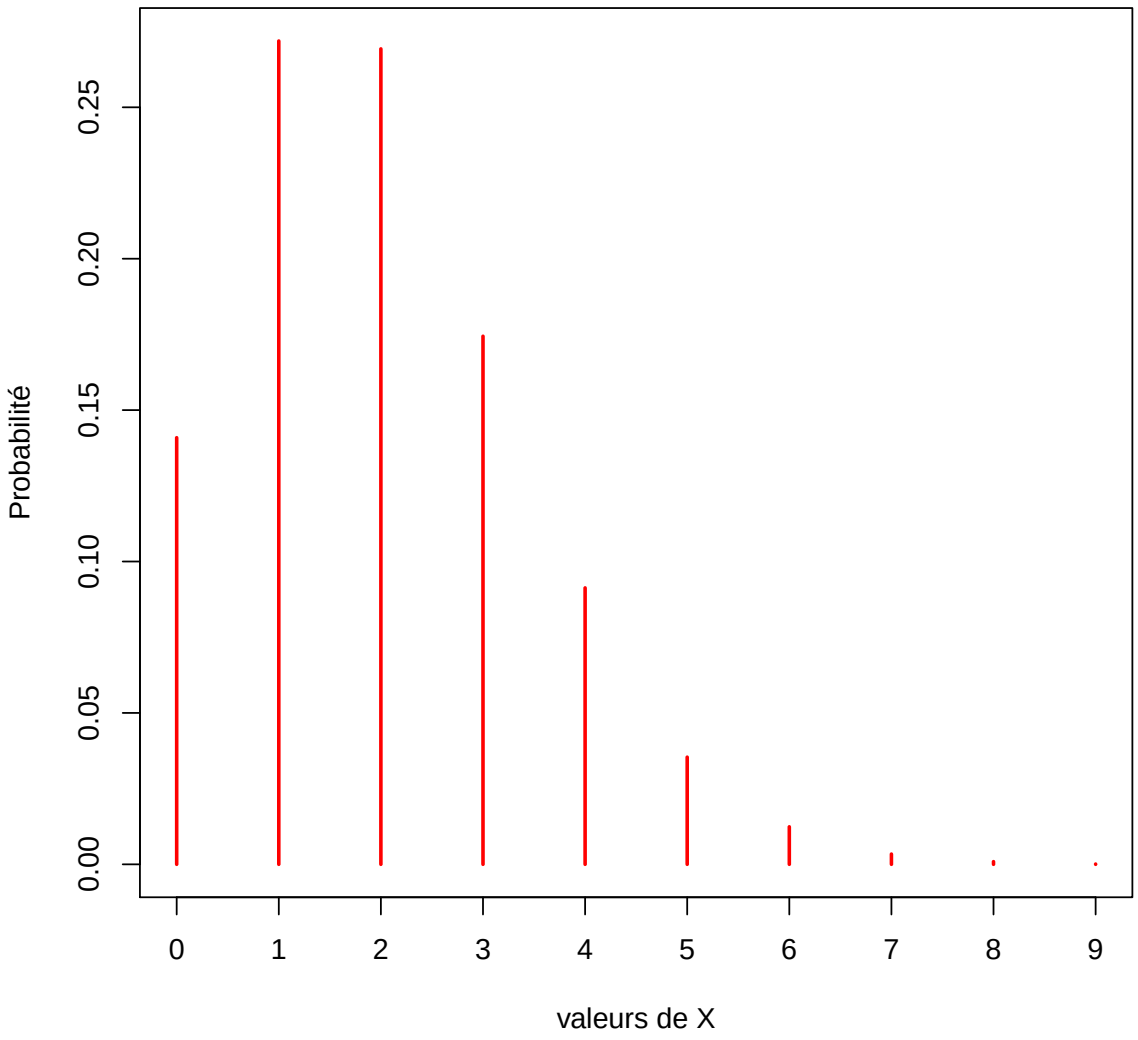
Soit Y une variable aléatoire suivant une loi de Poisson :

- valeurs possibles : $\mathbb{N}$
- paramètre : λ
- $\forall k \in \mathbb{N},\ \mathbb{P}(Y = k) = \dfrac{\lambda^k}{k!} e^{-\lambda}$
- $\mathbb{E}[Y] = \lambda$ et $Var(Y) = \lambda$

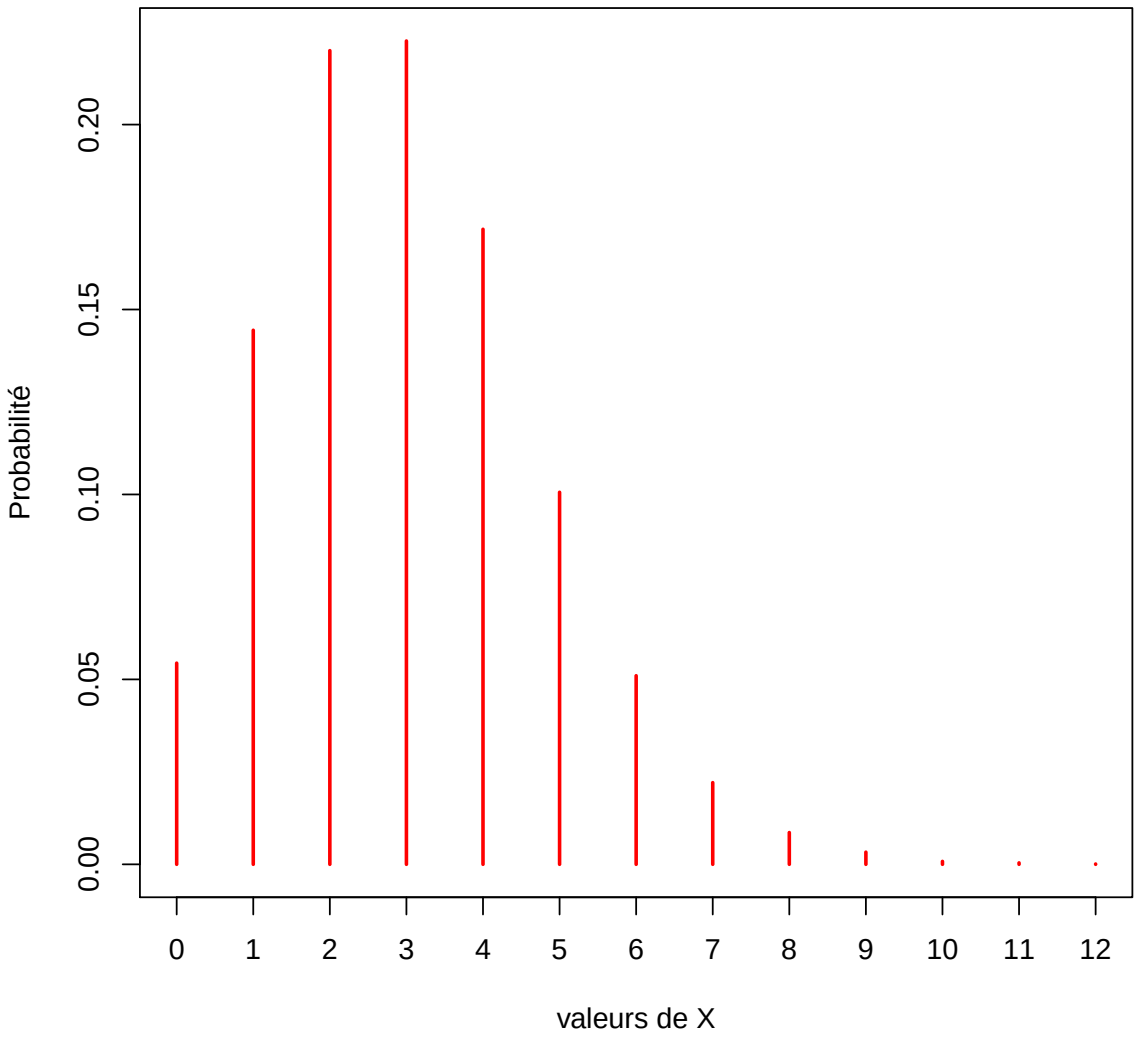
(Simulation d'une) Loi de Poisson pour λ=1



(Simulation d'une) Loi de Poisson pour λ=2



(Simulation d'une) Loi de Poisson pour λ=3



GLM avec distribution de Poisson.

- $Y_i \sim \mathscr{P}(\mu_i)$.
- $\mathbb{E}[Y_i] = \mu_i$ et $Var(Y) = \mu_i$.
- $\log(\mu_i) = \eta(X_{i1}, \cdots, X_{iq})$.

Remarque

La fonction lien pour un tel modèle est donc la fonction log.

Exemple avec SAS®

- Télécharger le fichier RoadKills.txt sur la page de Laurent Gardes.
- Import des données

```
PROC IMPORT DATAFILE=
"/folders/myfolders/Documents/UElogiciels/RoadKills.txt"
        OUT= work.rk ;
        GETNAMES=YES ;
RUN ;
```

- Affichage des attributs des variables :

```
PROC CONTENTS DATA= work.rk ; RUN ;
```

Exemple avec SAS®.

- On veut étudier le nombre d'individus écrasés en fonction de la distance au parc le plus proche.

- Représentation graphique

```
PROC SGPLOT DATA=rk ;
        scatter y=tot_N x=D_park ;
run ;
```

Exemple avec SAS®.

- Ajustement du modèle

```
PROC GENMOD DATA=rk ;
        MODEL tot_N=D_park / dist=poisson link=log ;
run ;
```

Sélection du modèle :

Dans le jeu de données il y a beaucoup de variables explicatives (cf *Mixed Effects Models and Extensions in Ecology with R* de Zuur et al., p385).

- 17 variables explicatives.
- 52 observations.
- Corrélation entre les variables explicatives ?

La matrice des corrélations serait de taille 17×17 : pas évident de voir quelque chose.
On va utiliser le VIF : Variance Inflation Factors : on obtient une valeur par variable.
VIF inférieur à 3 : aucun soucis de colinéarité.

Sélection du modèle

Il faut d'abord faire une régression linéaire multiple pour pouvoir calculer les VIF associés aux variables.

```
PROC REG DATA=rk ;
      model TOT_N= OPEN_L OLIVE MONT MONT_S
  POLIC SHRUB URBAN WAT_RES L_WAT_C
  L_D_ROAD L_P_ROAD D_WAT_RES D_WAT_COUR
  D_PARK N_PATCH P_EDGE L_SDI / VIF ;
run ;
```

On doit retirer en premier la variable avec le plus grand VIF (si il est supérieur à 3).

Sélection du modèle

```
PROC REG DATA=rk ;
      TOT_N= OPEN_L OLIVE MONT_S
   POLIC SHRUB URBAN WAT_RES L_WAT_C
   L_D_ROAD L_P_ROAD D_WAT_RES D_WAT_COUR
   D_PARK N_PATCH P_EDGE L_SDI / VIF ;
run ;
```

Sélection du modèle

Après quelques étapes, on obtient :

```
PROC REG DATA=rk ;
      model TOT_N =OPEN_L OLIVE MONT_S
    POLIC SHRUB URBAN WAT_RES L_WAT_C
    L_D_ROAD L_P_ROAD D_WAT_RES
    D_WAT_COUR D_PARK, / VIF ;
run ;
```

Sélection du modèle

- Si tous les VIF ≤ 3 on commence la sélection du modèle sur la base de l'AIC ou par des LRT.
- Il faut faire attention de revenir au modèle GLM.
- AIC :
  - Il faut ajuster tous les sous-modèles...
  - et garder celui avec le plus petit AIC ou le modèle complet.
- LRT :
  - Il faut ajouter l'option type3 dans la ligne du paramètre model.
  - On obtient un tableau avec une ligne par variable.
  - Si toutes les p-values des tests $LRT < 5\%$ on arrête
  - Sinon on retire la variable avec la p-value la plus grande.

Sélection du modèle

```
PROC GENMOD DATA=rk;
      model TOT_N =OPEN_L OLIVE MONT_S
    POLIC SHRUB URBAN WAT_RES L_WAT_C
    L_D_ROAD L_P_ROAD D_WAT_RES
    D_WAT_COUR D_PARK,
    / link=log dist=poisson type3;
run;
PROC GENMOD DATA=rk;
      model TOT_N =OPEN_L OLIVE MONT_S
    POLIC SHRUB WAT_RES L_WAT_C
    L_D_ROAD L_P_ROAD D_WAT_RES
    D_WAT_COUR D_PARK,
    / link=log dist=poisson type3;
run;
```

Sélection du modèle

```
PROC GENMOD DATA=rk;
      model TOT_N =OPEN_L OLIVE MONT_S
    SHRUB L_WAT_C
    L_P_ROAD D_WAT_RES
    D_PARK,
    / link=log dist=poisson type3;
run;
```

Sur-dispersion pour les modèles de Poisson

- Loi de Poisson :
  - espérance = variance.
  - Généralement ce n'est pas le cas : espérance $<$ variance.
  - On dit alors que le modèle est sur-dispersé.
- Pour savoir si un modèle est sur-dispersé on peut calculer le ratio :

$$\Phi = \frac{\text{Deviance}}{n-p}$$

où

  - n : nombre d'observation
  - p : nombre de paramètres.

- Au lieu de la déviance, on peut prendre le "Pearson Chi-Squared" qui est une estimation de la déviance.
- Si Φ n'est pas proche de 1, on peut considérer qu'il y a sur-dispersion.

Sur-dispersion pour les modèles de Poisson

- Dans un tel cas la loi de Poisson n'est plus réellement adaptée...
- mais reste pourtant la loi à utiliser !
- On va forcer SAS à permettre de prendre en compte un facteur multiplicatif entre la variance et l'espérance.
- Si on veut que $\Phi = 1$ pour la déviance : Dscale.
- Si on veut que $\Phi = 1$ pour le χ^2 de Pearson : Pscale.

Remarque

Si le ratio Φ est supérieur à 15, il faut envisager l'utilisation d'autres types de modèles :

- *GLM Negative Binomial*
- *Zero Inflated Models*

Sélection du modèle pour quasi-Poisson

- Un modèle sur-dispersé va entrainer des erreurs au niveau des p-valeurs
- car les écart-types sont sous-estimés.
- Il faut reprendre la sélection du modèle à partir de la fin de l'étude des VIF.

```
PROC GENMOD DATA=rk;
      model TOT_N =OPEN_L OLIVE MONT_S
    POLIC SHRUB URBAN WAT_RES L_WAT_C
    L_D_ROAD L_P_ROAD D_WAT_RES
    D_WAT_COUR D_PARK,
    / link=log dist=poisson pscale type3;
run;
```

Sélection du modèle

```
PROC GENMOD DATA=rk;
      model TOT_N =OLIVE MONT_S
    SHRUB L_WAT_C
    L_P_ROAD D_WAT_RES
    D_PARK,
    / link=log dist=poisson pscale type3;
run;
```

GLM pour des données d'absence/présence.

Variable binaire :

- 2 valeurs possibles : 0 et 1,
- présence/absence,
- succès/échec.

Soit Y une variable aléatoire suivant une loi de Bernoulli :

- valeurs possibles : 0 et 1,
- paramètre : $0 < p < 1$,
- $\mathbb{P}(Y = 1) = p$,
- $\mathbb{P}(Y = 0) = 1 - p$
- $\mathbb{E}[Y] = p$ et $Var(Y) = p(1-p)$.

GLM pour des données d'absence/présence.

Remarque

- *La somme de k loi de Bernouilli indépendantes de même paramètre p est une loi loi Binomiale de paramètres (k, p).*
- *La loi Binomiale correspond au nombre de succès sur k essais indépendants ayant chacun la même probabilité de succès.*
- *La loi de Bernoulli de paramètre p correspond à une loi Binomiale de paramètres $(1, p)$.*
- *Un GLM avec une loi de Bernouilli est aussi appellée régression logistique.*

Régression logistique.

- $Y_i \sim \mathscr{B}(1, \pi_i)$
- $\mathbb{E}[Y_i] = \pi_i$ et $Var(Y) = \pi_i \times (1 - \pi_i)$
- $\text{logit}(\pi_i) = \eta(X_{i1}, \cdots, X_{iq})$

Remarque

- *La fonction lien pour un tel modèle est donc la fonction logit.*
- $\forall 0 < p < 1,\ logit(p) = \dfrac{p}{1-p}$.
- *Il y a d'autres fonctions lien envisageables mais logit est la plus fréquente.*

Exemple avec SAS®.

- Télécharger le fichier Boar.txt sur la page de Laurent Gardes.
- Import des données

```
PROC IMPORT DATAFILE=
"/folders/myfolders/Documents/UElogiciels/boar.txt"
        OUT= work.boar ;
        GETNAMES=YES ;
RUN ;
```

- Affichage des attributs des variables :

```
PROC CONTENTS DATA= work.boar ; RUN ;
```

Exemple avec SAS®.

- On veut étudier la probabilité qu'un sanglier soit porteur de la tuberculose en fonction de la longueur de son corps.

- Représentation graphique

```
PROC SGPLOT DATA=boar ;
        scatter y=Tb x=LengthCT ;
run ;
```

Exemple avec SAS®.

- Ajustement du modèle

```
PROC GENMOD DATA=boar ;
      MODEL Tb=LengthCT / dist=binomial link=logit ;
run ;
```

On obtient :

$$\text{logit}(\pi_i) = -3.89 + 0.03 \times \text{LengthCT}_i$$

et donc :

$$\pi_i = \frac{e^{-3.89+0.03\times\text{LengthCT}_i}}{1+e^{-3.89+0.03\times\text{LengthCT}_i}}$$

GLM pour des proportions.

Proportion :

- pourcentage donc compris entre 0 et 100%
- il s'agit généralement d'un ratio :
  - $\frac{\text{Nombre de succès}}{\text{Nombre total d'individus}}$
  - $\frac{\text{Nombre de présence}}{\text{Nombre total d'individus}}$
- Ainsi expliquer un ratio revient à expliquer un nombre d'occurences de l'événement d'intérêt sachant le nombre de tentatives.

GLM pour des proportions.

Soit Y une variable aléatoire suivant une loi Binomiale :

- paramètres :
  - $0 < p < 1$: probabilité de succès sur un essai
  - n : nombre de tentatives
- valeurs possibles : $0, 1, 2, \cdots, n$
- $\mathbb{E}[Y] = np$ et $Var(Y) = np(1-p)$

Remarque

- *Le n est fixé donc la seule chose aléatoire est le p.*

GLM pour des proportions.

- $Y_i \sim \mathscr{B}(n_i, \pi_i)$.
- $\mathbb{E}[Y_i] = n_i \times \pi_i$ et $Var(Y) = n_i \times \pi_i \times (1 - \pi_i)$.
- $\text{logit}(\pi_i) = \eta(X_{i1}, \cdots, X_{iq})$.

Remarque

- *La fonction lien pour un tel modèle est donc la fonction logit.*
- *Il y a d'autres fonctions lien envisageables mais logit est la plus fréquente.*

Exemple avec SAS®.

- Télécharger le fichier TbDeer.txt sur la page de Laurent Gardes.
- Import des données

```
PROC IMPORT DATAFILE=
"/folders/myfolders/Documents/UElogiciels/TbDeer.txt"
        OUT= work.TbDeer ;
        GETNAMES=YES ;
RUN ;
```

- Affichage des attributs des variables :

```
PROC CONTENTS DATA= work.TbDeer ; RUN ;
```

Exemple avec SAS®.

- Ajustement du modèle

```
PROC GENMOD DATA=TbDeer ;
        MODEL DeerPosCervi/DeerSampledCervi=OpenLand
        ScrubLand QuercusPlants QuercusTrees ReedDeerIndex
        EstateSize Fenced PinePlantation
/ dist=binomial link=logit ;
run ;
```

Etude des VIF.

```
PROC REG DATA=TbDeer;
      MODEL DeerPosCervi=OpenLand
      ScrubLand QuercusPlants QuercusTrees ReedDeerIndex
      EstateSize Fenced PinePlantation
/ VIF;
run;
```

Sélection du modèle.

```
PROC GENMOD DATA=TbDeer ;
      MODEL DeerPosCervi/DeerSampledCervi=OpenLand
      ScrubLand QuercusPlants QuercusTrees ReedDeerIndex
      EstateSize Fenced
/ dist=binomial link=logit type3 ;
run ;
```

Sur-dispersion pour les modèles de proportions :

- Pour savoir si un modèle est sur-dispersé on peut calculer le ratio :

$$\Phi = \frac{\text{Deviance}}{n-p}$$

où

  - n : nombre d'observation
  - p : nombre de paramètres.

- $n-p$ correspond au Df de la Deviance.

- Au lieu de la déviance, on peut prendre le "Pearson Chi-Squared" qui est une estimation de la déviance.

- Si Φ n'est pas proche de 1, on peut considérer qu'il y a sur-dispersion.

Sur-dispersion pour les modèles de Poisson

- Dans un tel cas la loi Binomiale n'est plus réellement adaptée...
- mais reste pourtant la loi à utiliser !
- On va forcer SAS à permettre de prendre en compte un facteur facteur d'échelle qui va pouvoir gérer la surdispersion.
- Si on veut que $\Phi = 1$ pour la déviance : Dscale.
- Si on veut que $\Phi = 1$ pour le χ^2 de Pearson : Pscale.

Sélection du modèle pour quasi-Binomiale

- Un modèle sur-dispersé va entrainer des erreurs au niveau des p-valeurs
- car les écart-types sont sous-estimés.
- Il faut reprendre la sélection du modèle à partir de la fin de l'étude des VIF.

```
PROC GENMOD DATA=TbDeer ;
      MODEL DeerPosCervi/DeerSampledCervi=OpenLand
      ScrubLand QuercusPlants QuercusTrees ReedDeerIndex
      EstateSize Fenced
/ dist=binomial link=logit dscale type3 ;
run ;
```

Utilisation d'un offset.

- Exemple : Roulin et Bersier (2007)
- utilisé dans Zuur et *al.* (2009)

But :

- espèce : chouette effraie,
- analyser les comportements de mendicité des oisillons
- avec différentes variables explicatives :
  - sexe du parent
  - type de nourriture
  - temps d'arrivée du parent

Loi de Poisson ?

Variable à expliquer :

- nombre d'appels des oisillons de la nichée juste avant l'arrivée d'un parent,
- donnée de comptage $\Longrightarrow$ loi de Poisson,
- dépend du nombre d'oisillons présents dans le nid.

Pour que les données des différentes nichées soient comparables, il faudrait diviser le nombre d'appels par le nombre d'oisillons. .. mais alors nous n'aurons certainement plus des vraies données de comptage (nombres entiers) et donc ne pourrons plus utiliser la loi de Poisson.

- Utilisation du nombre d'appels non corrigé $\Longrightarrow$ loi de Poisson.
- Il faut rajouter la taille du nid (notée B_i) dans le modèle...
- mais pas comme une variable explicative : il ne faut pas de paramètres estimé.

GLM de Poisson :

$$\log(\mu_i) = \eta(X_{i,1}, \cdots, X_{i,q}) + \log(B_i)$$

ainsi

$$\log(\mu_i) - \log(B_i) = \eta(X_{i,1}, \cdots, X_{i,q})$$

or

$$\log(\mu_i) - \log(B_i) = \log\left(\frac{\mu_i}{B_i}\right)$$

Solution : offset

- Considération du nombre d'appels non corrigé mais sans paramètre estimé.
- C'est ce qu'on appelle un offset.
- Dans , on peut utiliser l'option offset dans la procédure GENMOD .

Exemple avec SAS®.

- Télécharger le fichier Owls.txt sur la page de Laurent Gardes.
- Import des données

```
PROC IMPORT DATAFILE=
"/folders/myfolders/Documents/UElogiciels/Owls.txt"
        OUT= work.Owls;
        GETNAMES=YES;
RUN;
```

- Affichage des attributs des variables :

```
PROC CONTENTS DATA= work.Owls; RUN;
```

Dans un premier temps, il nous faut créer une nouvelle variable qui correspond au log de Broodsize.

```
DATA Owls1 ;
      SET Owls ;
      logBZ = log(Broodsize) ;
RUN ;
```

On va travailler avec la variable ArrivalTime en version centrée réduite :

```
PROC STANDARD DATA=Owls1 MEAN=0 STD=1
OUT=Owls2 ;
      VAR ArrivalTime ; RUN ;
```

Ajustement du modèle :

```
PROC GENMOD DATA=Owls2;
      CLASS FoodTreatment SexParent
      MODEL SiblingNegotiation =ArrivalTime
   FoodTreatment SexParent SexParent*ArrivalTime
   SexParent*FoodTreatment FoodTreatment*ArrivalTime
   / link=log dist=poisson type3 offset=logBZ;
run;
```