Modèles Linéaires Appliqués / Régression
GLM : Lien & Variance

Arthur Charpentier

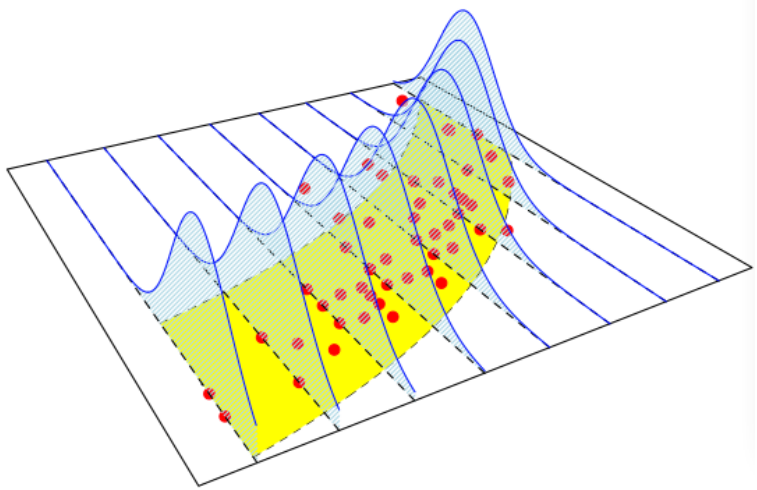
UQAM

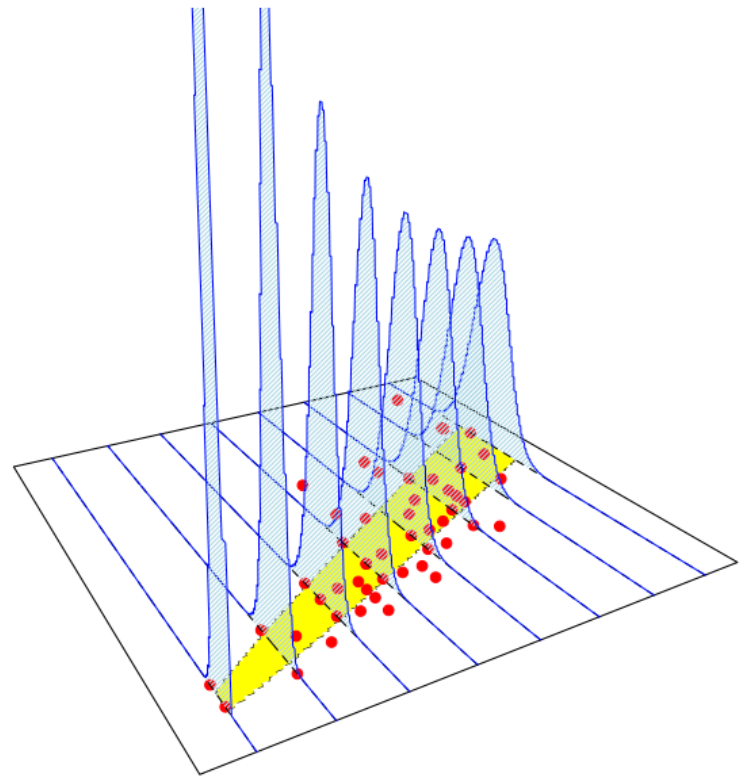
Hiver 2020 - COVID-19 # 14

GLM, Lien & Variance

Les GLM sont associés à deux composantes importantes

- lien : $\mathbb{E}(Y) = \mu = g^{-1}(\eta) = g^{-1}(\boldsymbol{x}^\top \beta)$
- variance : $\text{Var}(Y) = \varphi V(\mu)$





Quasi-Vraisemblance

La plupart des caractéristiques des GLM ne dépendent *que* des deux premiers moments de la distribution

- lien : $\mathbb{E}(Y) = \mu = g^{-1}(\eta) = g^{-1}(\boldsymbol{x}^\top \beta)$
- variance : $\text{Var}(Y) = \varphi V(\mu)$

Quasi log-densité :

$$Q(y_i; \mu_i) = \int_{y_i}^{\mu_i} \frac{y_i - t}{\varphi V(t)} \, dt$$

Quasi log-vraisemblance :

$$Q(\boldsymbol{\mu}, \varphi, \boldsymbol{y}) = \sum_{i=1}^{n} Q(\mu_i, y_i) = \sum_{i=1}^{n} \int_{y_i}^{\mu_i} \frac{y - t}{\varphi V(t)} \, dt$$

Quasi score :

$$S_j(\beta_j) = \frac{\partial}{\partial \beta_j} Q(\boldsymbol{\beta}, \varphi, \boldsymbol{y}) = \sum_{i=1}^{n} \frac{\partial \mu_i}{\partial \beta_j} \frac{y_i - \mu_i(\beta_j)}{\varphi V(\mu_i)}$$

Quasi-Vraisemblance

Cas assez général, transformations puissances

- lien : $\eta = \mu^a$ ou $\mathbb{E}(Y) = \mu = \eta^{1/a} = (\boldsymbol{x}^\top\beta)^{1/a}$
- variance : $\text{Var}(Y) = \varphi\mu^b$

```
> glm(y~x,family=tweedie(var.power=b,link.power=a))
```

On peut définir une loi quasi-Poisson

$$\text{Poisson} \rightarrow \mathbb{E}(Y|\boldsymbol{X}=\boldsymbol{x}) = \log(\boldsymbol{x}^\top\beta) = \mu \text{ et } \text{Var}(Y|\boldsymbol{X}=\boldsymbol{x}) = \mu$$

$$\mathbb{E}(Y|\boldsymbol{X}=\boldsymbol{x}) = \log(\boldsymbol{x}^\top\beta) \text{ et } \text{Var}(Y|\boldsymbol{X}=\boldsymbol{x}) = \varphi\mu \rightarrow \text{quasi-Poisson}$$

On peut aussi définir une loi quasi-Binomiale,

$$\mu = \frac{e^{\boldsymbol{x}^\top\beta}}{1+e^{\boldsymbol{x}^\top\beta}} \text{ et } V(\mu) = \varphi\mu(1-\mu)$$

Quasi-Poisson

Loi binomiale négative, de moyenne $\mathbb{E}(Y) = \dfrac{n(1-p)}{p}$

$$f(y, n, p) = \frac{\Gamma(y+n)}{\Gamma(n)y!} p^n (1-p)^y, \ y \in \mathbb{N},$$

```
> Y = rnbinom(n=1e5, size=10, prob=.1)
```

ici $\mathbb{E}(Y) = 90$ (et $\log(90) = 4.49981$)

```
> summary(glm(Y~1,family=quasipoisson))

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 4.499805   0.001055    4266   <2e-16 ***

(Dispersion parameter quasipoisson family 10.01436)
```

Quasi-Poisson

```
> glm(formula = Y ~ YEARMARRIAGE + CHILDREN +
    RELIGIOUS + EDUCATION + SATISFACTION, family =
    poisson, data = base)

Coefficients:
             Estimate Std. Error z value Pr(>z)
(Intercept)  -0.83704    0.44538  -1.879  0.06019 .
YEARMARRIAGE  0.02287    0.01189   1.923  0.05452 .
CHILDREN      0.52029    0.16889   3.081  0.00207 **
RELIGIOUS    -0.24972    0.04802  -5.201 1.99e-07 ***
EDUCATION     0.10974    0.02288   4.797 1.61e-06 ***
SATISFACTION -0.34343    0.04555  -7.540 4.72e-14 ***

(Dispersion parameter poisson family taken to 1)
```

Quasi-Poisson

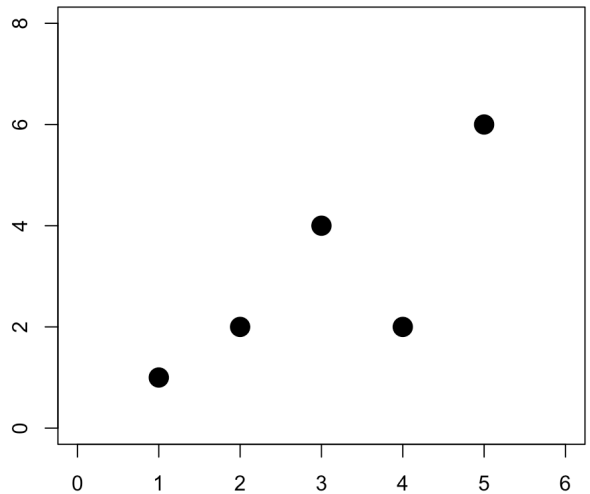
```
> glm(formula = Y ~ YEARMARRIAGE + CHILDREN +
    RELIGIOUS + EDUCATION + SATISFACTION, family =
    quasipoisson, data = base)

Coefficients:
             Estimate Std. Error t value Pr(>t)
(Intercept)  -0.83704    0.84741  -0.988  0.32370
YEARMARRIAGE  0.02287    0.02263   1.011  0.31267
CHILDREN      0.52029    0.32134   1.619  0.10599
RELIGIOUS    -0.24972    0.09136  -2.733  0.00647 **
EDUCATION     0.10974    0.04353   2.521  0.01198 *
SATISFACTION -0.34343    0.08667  -3.963 8.38e-05 ***

(Dispersion parameter for quasipoisson taken to 3.620)
```

Jeu de Données

```
> x = c(1,2,3,4,5)
> y = c(1,2,4,2,6)
> plot(x,y,pch=19)
> base = data.frame(x=x, y=y)
```



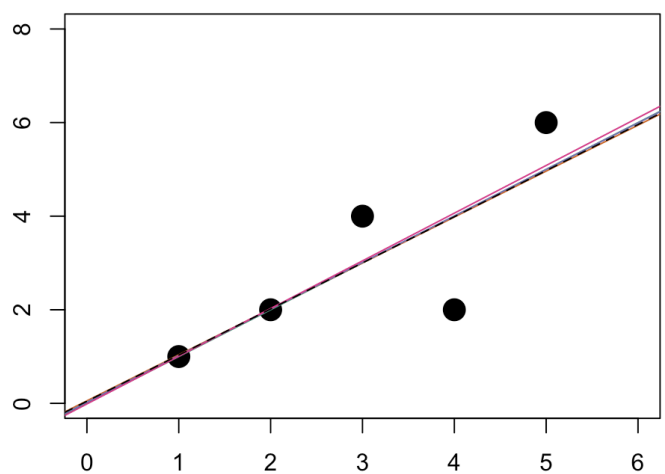
On considère plusieurs modèles de régression

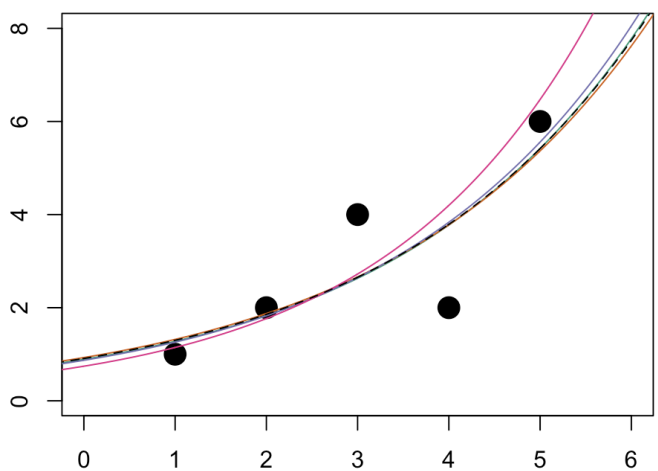
```
> regNId = glm(y~x,family=gaussian(link="identity"))
> regNlog = glm(y~x,family=gaussian(link="log"))
> regPId = glm(y~x,family=poisson(link="identity"))
> regPlog = glm(y~x,family=poisson(link="log"))
> regGId = glm(y~x,family=Gamma(link="identity"))
> regGlog = glm(y~x,family=Gamma(link="log"))
> regIGId = glm(y~x,family=inverse.gaussian(link= ...
> regIGlog = glm(y~x,family=inverse.gaussian(link= ..
```

Jeu de Données

```
> DF=data.frame(NId=predict(regNId,type="response"),
+               Nlog=predict(regNlog,type="response"),

> DF
  obs NId  Nlog  PInd  Plog   GId  Glog  TwId Twlog
1   1   1 1.277 1.037 1.319 1.021 1.261 1.031 1.299
2   2   2 1.833 2.018 1.874 2.016 1.827 2.016 1.857
3   4   3 2.631 3.000 2.661 3.010 2.647 3.002 2.653
4   2   4 3.776 3.982 3.779 4.004 3.835 3.987 3.791
5   6   5 5.419 4.963 5.367 4.999 5.557 4.973 5.418
```

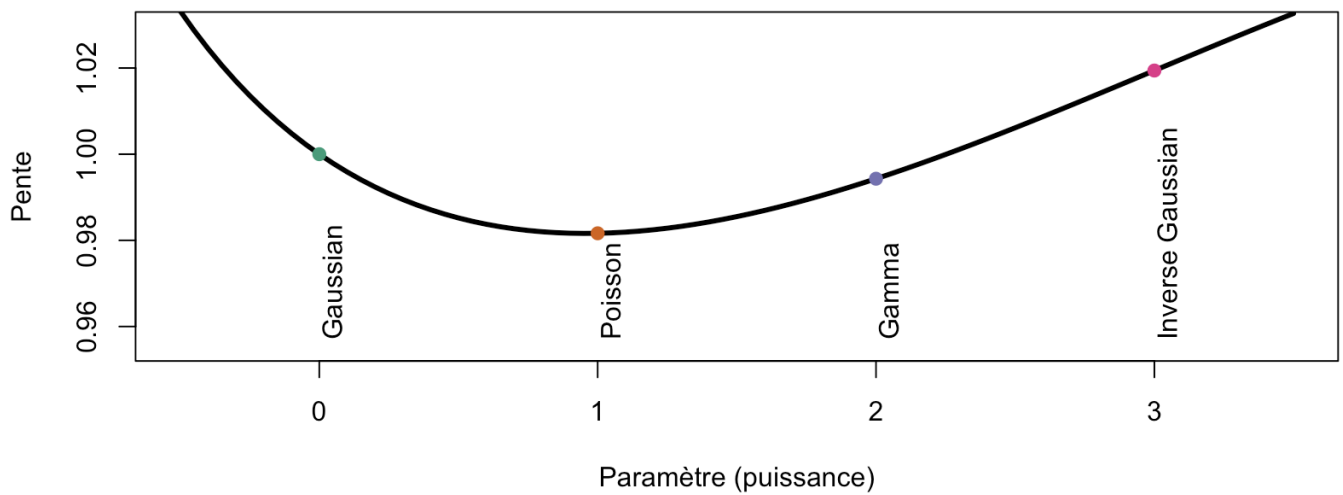




Jeu de Données

Pentes des modèles avec différentes lois, avec un lien 'identité'

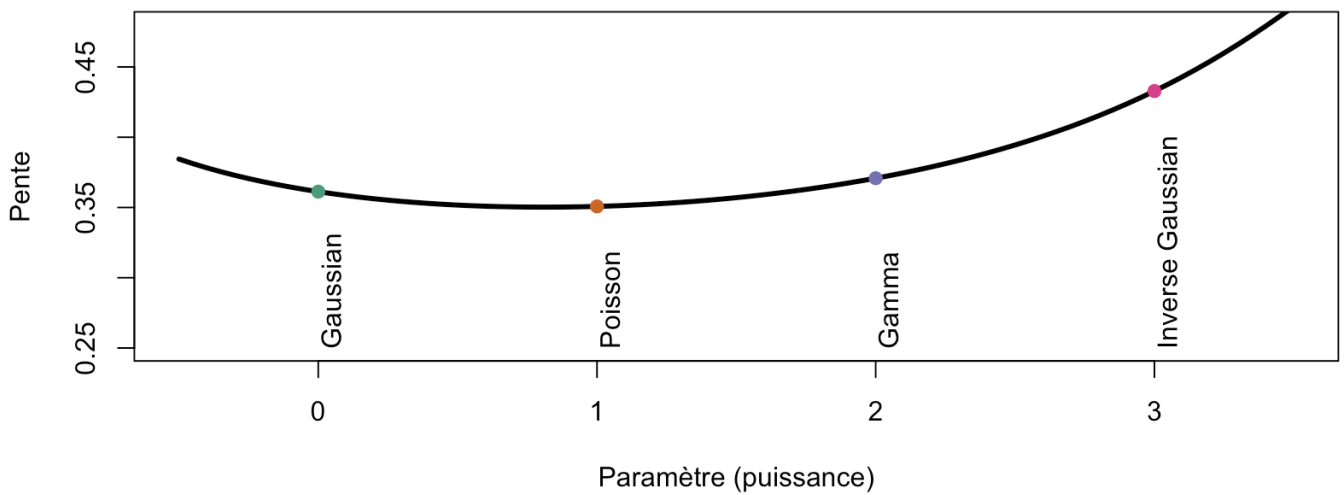
```
> pente = function(gamma) summary(glm(y~x,family=
    tweedie(var.power=gamma,link.power=1),data=base))
    $coefficients[2,1:2]
```



Jeu de Données

Pentes des modèles avec différentes lois, avec un lien 'log'

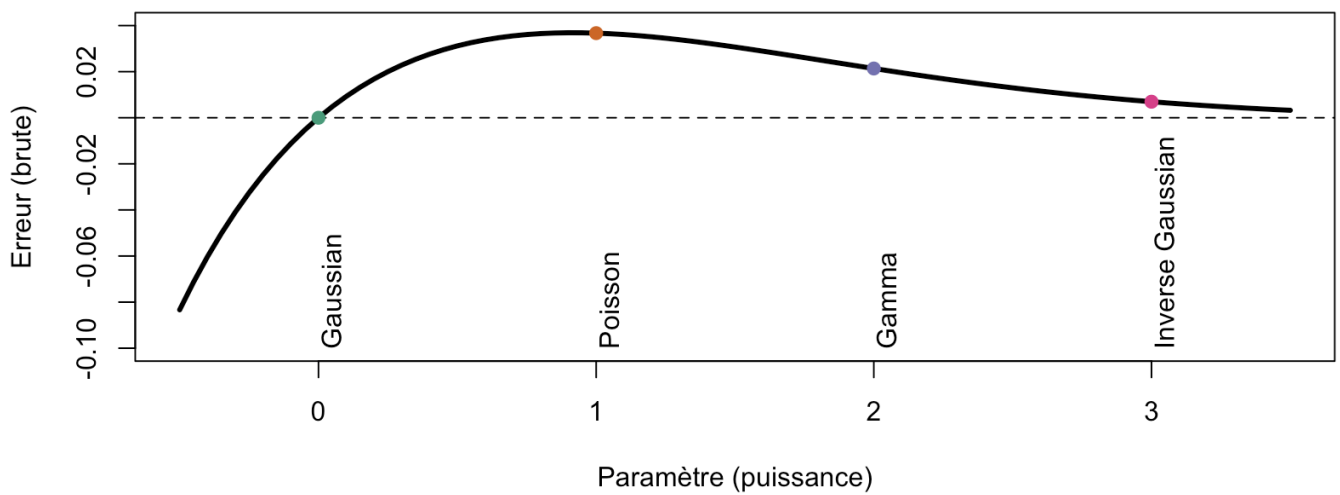
```
> pente = function(gamma) summary(glm(y~x,family=
    tweedie(var.power=gamma,link.power=0),data=base))
    $coefficients[2,1:2]
```



Jeu de Données

Erreur de prévision $y_1 - \widehat{\mu}_1$ avec différentes lois, avec un lien 'identité'

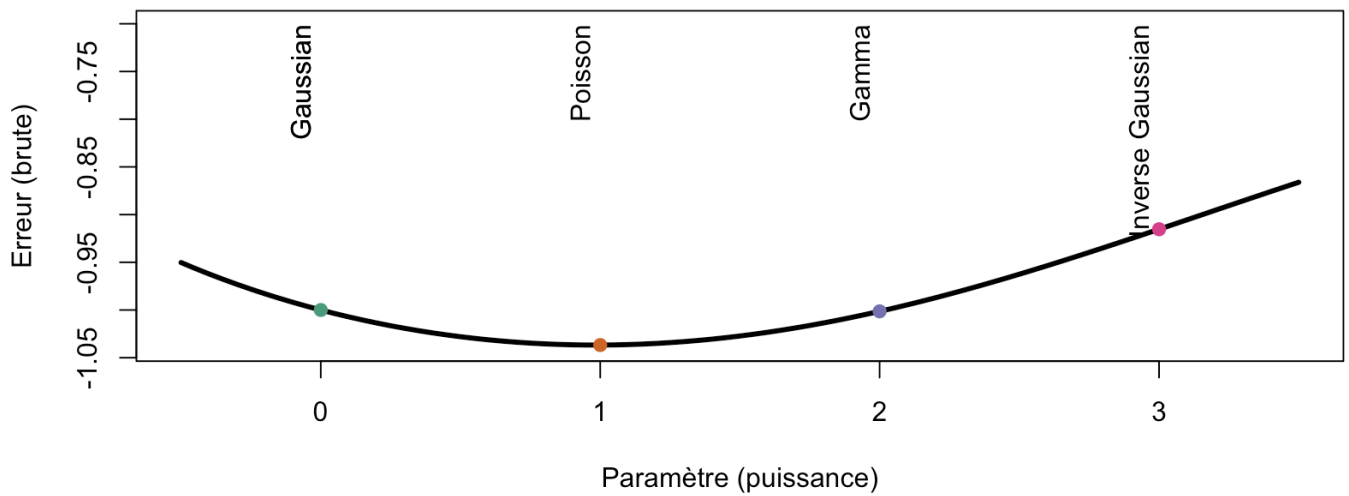
```
> pente = function(gamma) summary(glm(y~x,family=
    tweedie(var.power=gamma,link.power=0),data=base))
    $coefficients[2,1:2]
```



Jeu de Données

Erreur de prévision $y_5 - \widehat{\mu}_5$ avec différentes lois, avec un lien 'identité'

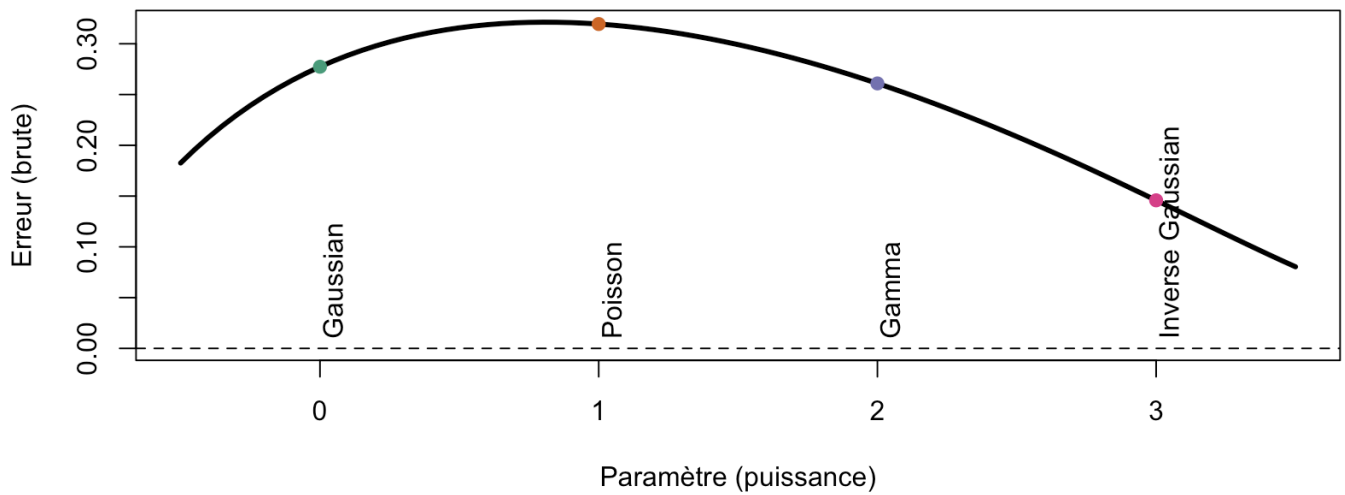
```
> pente = function(gamma) summary(glm(y~x,family=
    tweedie(var.power=gamma,link.power=0),data=base))
    $coefficients[2,1:2]
```



Jeu de Données

Erreur de prévision $y_1 - \widehat{\mu}_1$ avec différentes lois, avec un lien 'log'

```
> pente = function(gamma) summary(glm(y~x,family=
    tweedie(var.power=gamma,link.power=0),data=base))
    $coefficients[2,1:2]
```



Jeu de Données

Erreur de prévision $y_5 - \widehat{\mu}_5$ avec différentes lois, avec un lien 'log'

```
> pente = function(gamma) summary(glm(y~x,family=
    tweedie(var.power=gamma,link.power=0),data=base))
    $coefficients[2,1:2]
```