Actuariat IARD - ACT2040
Partie 3 - régression Poissonienne et biais minimal ($Y \in \mathbb{N}$)

Arthur Charpentier

charpentier.arthur@uqam.ca

http ://freakonometrics.hypotheses.org/

UQÀM
Université du Québec à Montréal

Automne 2013

Modélisation d'une variable de comptage

Références : Frees (2010), chapitre 12 (p 343-361) Greene (2012), section 18.3 (p 802-828) de Jong Heller (2008), chapitre 6, sur la régression de Poisson. Sur les méthodes de biais minimal, de Jong Heller (2008), section 1.3. Sinon

Cameron, A.C. & Trivedi, P.K. (1998). Regression analysis of count data, Cambridge University Press.

Denuit, M., Marechal, X., Pitrebois, S. & Walhin, J.-F. (2007) Actuarial Modelling of Claim Counts : Risk Classification, Credibility and Bonus-Malus Systems. Wiley.

Hilbe, J. M. (2007). Negative Binomial Regression, Cambridge University Press.

Remarque : la régression de Poisson est un cas particulier des modèles GLM, avec une loi de Poisson et une fonction de lien logarithmique.

```
regPoisson = function(formula, lien="log", data=NULL) {
glm(formula,family=poisson(link=log),data) }
```

Base pour les données de comptage

```
> sinistre=read.table("http://freakonometrics.free.fr/sinistreACT2040.txt",
+ header=TRUE,sep=";")
> sinistres=sinistre[sinistre$garantie=="1RC",]
> sinistres=sinistre[sinistres$cout>0,]
> contrat=read.table("http://freakonometrics.free.fr/contractACT2040.txt",
+ header=TRUE,sep=";")
> T=table(sinistres$nocontrat)
> T1=as.numeric(names(T))
> T2=as.numeric(T)
> nombre1 = data.frame(nocontrat=T1,nbre=T2)
> I = contrat$nocontrat%in%T1
> T1= contrat$nocontrat[I==FALSE]
> nombre2 = data.frame(nocontrat=T1,nbre=0)
> nombre=rbind(nombre1,nombre2)
> baseFREQ = merge(contrat,nombre)
```

Base pour les données de comptage

La variable d'intérêt Y est ici `nbre` ou `baseFREQ$nbre`

```
> head(baseFREQ,4)
  nocontrat exposition zone puissance agevehicule ageconducteur bonus marque
1        27        0.87    C         7           0             56    50     12
2       115        0.72    D         5           0             45    50     12
3       121        0.05    C         6           0             37    55     12
4       142        0.90    C        10          10             42    50     12
  carburant densite region nbre
1         D      93     13    0
2         E      54     13    0
3         D      11     13    0
4         D      93     13    0
```

Remarque : la base ici est une sous-base (particulière) de celle utilisée dans

http ://cran.r-project.org/doc/contrib/Charpentier_Dutang_actuariat_avec_R.pdf

Les conclusions risques d'être différentes aussi...

La loi de Poisson

La loi de Poisson est connue comme la *loi des petits nombres*,

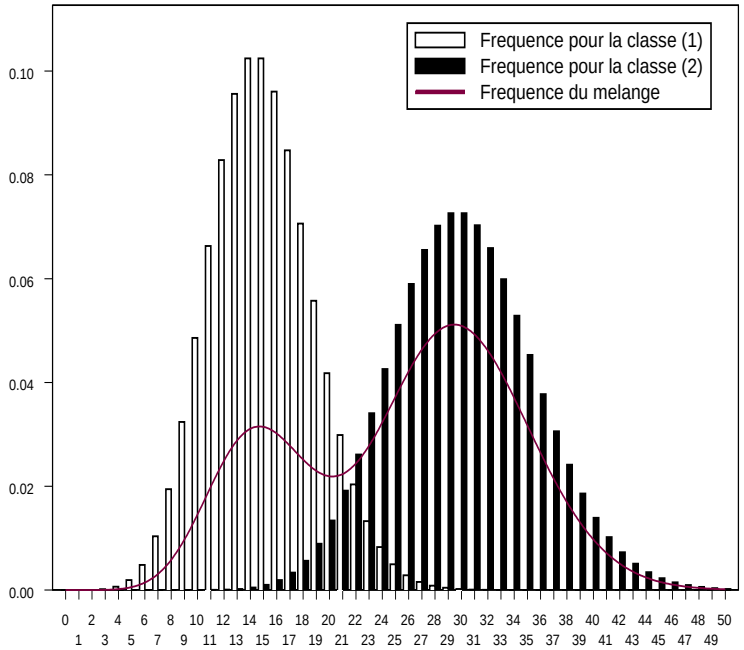
$$\mathbb{P}(N=k)=e^{-\lambda}\frac{\lambda^k}{k!},\forall k\in\mathbb{N}.$$

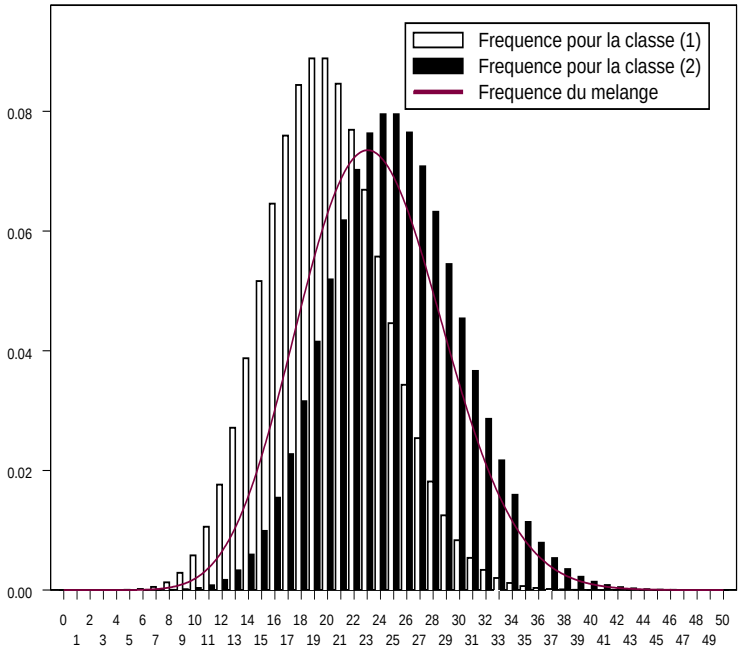
Pour rappel (cf premier cours), $\mathbb{E}(N)=\text{Var}(N)$.

La loi Poisson mélange

En présence de sur-dispersion $\mathbb{E}(N) < \text{Var}(N)$, on peut penser à une loi Poisson mélange, i.e. il existe Θ, variable aléatoire positive, avec $\mathbb{E}(\Theta) = 1$, telle que

$$\mathbb{P}(N = k|\Theta = \theta) = e^{-\lambda\theta}\frac{[\lambda\theta]^k}{k!}, \forall k \in \mathbb{N}.$$





La loi Binomiale Négative

La loi binomiale négative apparît dans le modèle Poisson mélange, lorsque $\Theta \sim \mathcal{G}(\alpha, \alpha)$. Dans ce cas,

$$\pi(\theta) = x^{\alpha-1} \frac{\alpha^\alpha \exp(-\alpha x)}{\Gamma(\alpha)}$$

$$\mathbb{E}(\Theta) = 1 \text{ et } \text{Var}(\Theta) = \frac{1}{\alpha}.$$

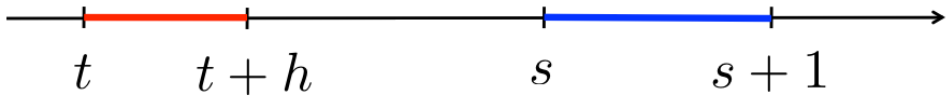
Dans ce cas, la loi (non conditionnelle) de N est

$$\mathbb{P}(N = k) = \int_0^\infty \mathbb{P}(N = k|\Theta = \theta)\pi(\theta)d\theta,$$

$$\mathbb{P}(N = k) = \frac{\Gamma(k+\alpha)}{\Gamma(k+1)\Gamma(\alpha)} \left(\frac{1}{1+\lambda/\alpha}\right)^\alpha \left(1 - \frac{1}{1+\lambda/\alpha}\right)^k, \forall k \in \mathbb{N}$$

Le processus de Poisson

Pour rappel, $(N_t)_{t\geq 0}$ est un processus de Poisson homogène (de paramètre λ) s'il est à accroissements indépendants, et le nombre de sauts observés pendant la période $[t, t+h]$ suit une loi $\mathcal{P}(\lambda \cdot h)$.



$N_{s+1} - N_s \sim \mathcal{P}(\lambda)$ est indépendant de $N_{t+h} - N_t \sim \mathcal{P}(\lambda \cdot h)$.

Exposition et durée d'observation

Soit N_i la frénquence annulisée de sinistre pour l'assuré i, et supposons $N_i \sim \mathcal{P}(\lambda)$.

Si l'assuré i a été observé pendant une période E_i, le nombre de sinistre observé est $Y_i \sim \mathcal{P}(\lambda \cdot E_i)$.

$$\mathcal{L}(\lambda, \boldsymbol{Y}, \boldsymbol{E}) = \prod_{i=1}^{n} \frac{e^{-\lambda E_i} [\lambda E_i]_i^Y}{Y_i!}$$

$$\log \mathcal{L}(\lambda, \boldsymbol{Y}, \boldsymbol{E}) = -\lambda \sum_{i=1}^{n} E_i + \sum_{i=1}^{n} Y_i \log[\lambda E_i] - \log \left(\prod_{i=1}^{n} Y_i! \right)$$

qui donne la condition du premier ordre

$$\frac{\partial}{\partial \lambda} \log \mathcal{L}(\lambda, \boldsymbol{Y}, \boldsymbol{E}) = -\sum_{i=1}^{n} E_i + \frac{1}{\lambda} \sum_{i=1}^{n} Y_i$$

qui s'annule pour

$$\widehat{\lambda} = \frac{\sum_{i=1}^n Y_i}{\sum_{i=1}^n E_i} = \sum_{i=1}^n \omega_i \frac{Y_i}{E_i} \text{ avec } \omega_i = \frac{E_i}{\sum_{i=1}^n E_i}$$

```
> N <- baseFREQ$nbre
> E <- baseFREQ$exposition
> (lambda=sum(N)/sum(E))
[1] 0.07279295
> weighted.mean(N/E,E)
[1] 0.07279295
> dpois(0:3,lambda)*100
[1] 92.979  6.768  0.246  0.006
```

Remarque : pour E_i on parlera d'exposition ou d'années police.

Fréquence annuelle, une variable tarifaire

```
> X1 <- baseFREQ$carburant
> sum(N[X1=="D"])
[1] 998
> sum(N[X1=="E"])
[1] 926
> tapply(N, X1, sum)
  D   E
998 926
```

On a ici le nombre d'accidents par type de carburant.

```
> tapply(N, X1, sum)/tapply(E, X1, sum)
         D          E
0.07971533 0.06656323
```

Fréquence annuelle, une variable tarifaire

Notons que l'on peut comparer ces moyenne (pondérées)

```
> library(weights)
> wtd.t.test(x=(N/E)[X1=="D"], y=(N/E)[X1=="E"],
+ weight=E[X1=="D"], weighty=E[X1=="E"],samedata=FALSE)
$test
[1] "Two Sample Weighted T-Test (Welch)"

$coefficients
     t.value           df      p.value
2.835459e+00 4.959206e+04 4.577838e-03

$additional
 Difference      Mean.x      Mean.y    Std. Err
0.013152097 0.079715332 0.066563235 0.004638437
```

Avec une p-value de l'ordre de 0.004 on accepte l'hypothèse que les deux moyennes sont distinctes. On retrouve un résultat semblable avec une régression linéraire

Fréquence annuelle, une variable tarifaire

```
> summary(lm(N/E~X1,weights=E))

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.079715   0.003386  23.540  < 2e-16 ***
X1E         -0.013152   0.004668  -2.818  0.00484 **
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

Residual standard error: 0.3789 on 49998 degrees of freedom
Multiple R-squared: 0.0001588,        Adjusted R-squared: 0.0001388
F-statistic: 7.939 on 1 and 49998 DF,  p-value: 0.00484
```

Fréquence annuelle et tableau de contingence

Supposons que l'on prenne en compte ici deux classes de risques.

```
> N <- baseFREQ$nbre
> E <- baseFREQ$exposition
> X1 <- baseFREQ$carburant
> X2 <- cut(baseFREQ$agevehicule,c(0,3,10,101),right=FALSE)
> names1 <- levels(X1)
> names2 <- levels(X2)
> (POPULATION=table(X1,X2))
   X2
X1  [0,3) [3,10) [10,101)
  D  7492  10203     6718
  E  6275   9417     9895
```

Fréquence annuelle et tableau de contingence

Mais la population (i.e. le nombre d'assurés) ne correspond pas forcément au nombre d'année-police.

```
> EXPOSITION=POPULATION
> for(k in 1:nrow(EXPOSITION)){
+ EXPOSITION[k,]=tapply(E[X1==names1[k]],
+ X2[X1==names1[k]],sum)}
> EXPOSITION
   X2
X1      [0,3)    [3,10) [10,101)
  D 3078.938 5653.109 3787.503
  E 2735.014 5398.950 5777.619
```

	[0,3)	[3,10)	[10,101)
D	3078.94	5653.11	3787.50
E	2735.01	5398.95	5777.62

Fréquence annuelle et tableau de contingence

On peut ensuite compter les sinistres

```
> SINISTRE=POPULATION
> for(k in 1:nrow(SINISTRE)){
+ SINISTRE[k,]=tapply(N[X1==names1[k]],
+ X2[X1==names1[k]],sum)}
> SINISTRE
   X2
X1  [0,3) [3,10) [10,101)
  D   242    479      277
  E   163    368      395
```

	[0,3)	[3,10)	[10,101)
D	242	479	277
E	163	368	395

Fréquence annuelle et tableau de contingence

et interpréter les valeurs précédantes en fréquence annuelle de sinistre,

```
> (FREQUENCE=SINISTRE/EXPOSITION)
   X2
X1        [0,3)     [3,10)   [10,101)
  D 0.07859854 0.08473214 0.07313526
  E 0.05959749 0.06816140 0.06836727
```

	[0,3)	[3,10)	[10,101)
D	0.0786	0.0847	0.0731
E	0.0596	0.0682	0.0684

Fréquence annuelle et tableau de contingence

Notons $Y_{i,j}$ le nombre de sinistres observé (non annualisé) lorsque la première variable (i.e. X_1) prend la valeur i (ici i prend deux valeurs, diesel ou essence orginaire) et la seconde variable (i.e. C) prend la valeur j (ici j prend trois valeurs, voiture neuve (0-3), récente (3-10) ou vieille (plus de 10 ans)). On notera $N_{i,j}$ la fréquence annualisée.

La matrice $\boldsymbol{Y} = [Y_{i,j}]$ est ici `FREQUENCE`. On suppose qu'il est possible de modéliser Y à l'aide d'un modèle multiplicatif à deux facteurs, associés à chaque des des variables. On suppose que

$$N_{i,j} = L_i \cdot C_j.$$

On notera $E_{i,j}$ l'exposition, i.e. `EXPOSITION`. L'estimation de $\boldsymbol{L} = (L_i)$ et de $\boldsymbol{C} = (C_j)$ se fait généralement de trois manières : par moindres carrés, par minimisation d'une distance (e.g. du chi-deux) ou par un principe de balancement (ou méthode des marges).

Méthode des marges, Bailey (1963)

Dans la méthode des marges (selon la terminologie de Bailey (1963), formellement, on veut

$$\sum_j Y_{i,j} = \sum_j E_{i,j} N_{i,j} = \sum_j E_{i,j} L_i \cdot C_j,$$

en somment sur la ligne i, pour tout i, ou sur la colonne j,

$$\sum_i Y_{i,j} = \sum_i E_{i,j} N_{i,j} = \sum_i E_{i,j} L_i \cdot C_j,$$

La première équation donne

$$L_i = \frac{\sum_j Y_{i,j}}{\sum_j E_{i,j} C_j}$$

et la seconde

$$C_j = \frac{\sum_i Y_{i,j}}{\sum_i E_{i,j} L_i}.$$

Méthode des marges, Bailey (1963)

On résoud alors ce petit systeme de maniere itérative (car il n'y a pas de solution analytique simple).

```
>  (m=sum(SINISTRE)/sum(EXPOSITION))
[1] 0.07279295
>  L<-matrix(NA,100,2);C<-matrix(NA,100,3)
>  L[1,]<-rep(m,2);colnames(L)=names1
>  C[1,]<-rep(m,3);colnames(C)=names2
>  for(j in 2:100){
+  L[j,1]<-sum(SINISTRE[1,])/sum(EXPOSITION[1,]*C[j-1,])
+  L[j,2]<-sum(SINISTRE[2,])/sum(EXPOSITION[2,]*C[j-1,])
+  C[j,1]<-sum(SINISTRE[,1])/sum(EXPOSITION[,1]*L[j,])
+  C[j,2]<-sum(SINISTRE[,2])/sum(EXPOSITION[,2]*L[j,])
+  C[j,3]<-sum(SINISTRE[,3])/sum(EXPOSITION[,3]*L[j,])
+  }
```

Méthode des marges, Bailey (1963)

On obtient ici les séquences suivantes

```
> L[1:5,]
              D          E
[1,] 0.07279295 0.07279295
[2,] 1.09509693 0.91441875
[3,] 1.09368399 0.91569373
[4,] 1.09366449 0.91571133
[5,] 1.09366422 0.91571157
> C[1:5,]
          [0,3)     [3,10)   [10,101)
[1,] 0.07279295 0.07279295 0.07279295
[2,] 0.06896336 0.07611702 0.07125554
[3,] 0.06897350 0.07612457 0.07124032
[4,] 0.06897364 0.07612468 0.07124011
[5,] 0.06897364 0.07612468 0.07124011
```

ce qui donne les prédictions pour la fréquence

```
> PREDICTION1=SINISTRE
> PREDICTION1[1,]<-L[100,1]*C[100,]
> PREDICTION1[2,]<-L[100,2]*C[100,]
> PREDICTION1
   X2
X1       [0,3)     [3,10)   [10,101)
  D 0.07543400 0.08325484 0.07791276
  E 0.06315996 0.06970825 0.06523539
```

	[0,3)	[3,10)	[10,101)
D	0.0754	0.0833	0.0779
E	0.0632	0.0697	0.0652

On notera que les marges sont identiques, par exemple pour la première ligne

```
> sum(PREDICTION1[1,]*EXPOSITION[1,])
[1] 998
> sum(SINISTRE[1,])
[1] 998
```

Remarque : Cette technique est équivalente à utiliser une régression log-Poisson sur les deux variables qualitatives,

```
> donnees <- data.frame(N,E,X1,X2)
> regpoislog <- glm(N~X1+X2,offset=log(E),data=donnees,
+ family=poisson(link="log"))
> newdonnees <- data.frame(X1=factor(rep(names1,3)),E=rep(1,6),
+ X2=factor(rep(names2,each=2)))
> matrix(predict(regpoislog,newdata=newdonnees,
+ type="response"),2,3)
           [,1]       [,2]       [,3]
[1,] 0.07543401 0.08325484 0.07791276
[2,] 0.06315996 0.06970825 0.06523539
```

	[0,3)	[3,10)	[10,101)
D	0.0754	0.0833	0.0779
E	0.0632	0.0697	0.0652

Méthode des moindres carrés

Parmi les méthodes proches de celles évoquées auparavant sur la méthode des marges, il est aussi possible d'utiliser une méthode par moindres carrés (pondérée). On va chercher à minimiser la somme des carrés des erreurs, i.e.

$$D = \sum_{i,j} E_{i,j}(N_{ij} - L_i \cdot C_j)^2$$

La condition du premier ordre donne ici $\frac{\partial D}{\partial L_i} = -2\sum_j C_j E_{i,j}(N_{i.j} - L_i \cdot C_j) = 0$ soit

$$L_i = \frac{\sum_j C_j E_{i,j} N_{i.j}}{\sum_j E_{i,j} C_j^2} = \frac{\sum_j C_j Y_{i.j}}{\sum_j E_{i,j} C_j^2}$$

L'autre condition du premier ordre donne

$$C_j = \frac{\sum_i L_i E_{i,j} N_{i.j}}{\sum_i E_{i,j} L_i^2} = \frac{\sum_i L_i Y_{i.j}}{\sum_i E_{i,j} L_i^2}$$

On résoud alors ce petit systeme de maniere itérative (car il n'y a pas de solution analytique simple).

```
> L=matrix(NA,100,2);C=matrix(NA,100,3)
> L[1,]=c(1,1);colnames(L)=rownames(EXPOSITION)
> C[1,]=c(1,1,1);colnames(C)=colnames(EXPOSITION)
> for(j in 2:100){
+ L[j,1]=sum(SINISTRE[1,]*C[j-1,])/sum(EXPOSITION[1,]*C[j-1,]^2)
+ L[j,2]=sum(SINISTRE[2,]*C[j-1,])/sum(EXPOSITION[2,]*C[j-1,]^2)
+ C[j,1]=sum(SINISTRE[,1]*L[j,])/sum(EXPOSITION[,1]*L[j,]^2)
+ C[j,2]=sum(SINISTRE[,2]*L[j,])/sum(EXPOSITION[,2]*L[j,]^2)
+ C[j,3]=sum(SINISTRE[,3]*L[j,])/sum(EXPOSITION[,3]*L[j,]^2)
+ }

> PREDICTION2[1,]=L[100,1]*C[100,]
> PREDICTION2[2,]=L[100,2]*C[100,]
> PREDICTION2
   X2
X1       [0,3)     [3,10)   [10,101)
  D 0.07575130 0.08339880 0.07753709
  E 0.06342566 0.06982882 0.06492089
```

Méthode des moindres carrés

La prédiction pour chacune des classes est alors ici

	[0,3)	[3,10)	[10,101)
D	0.0758	0.0834	0.0775
E	0.0634	0.0698	0.0649

Méthode du χ^2

Parmi les méthodes proches de celles évoquées dans la section **??** sur la méthode des marges, il est aussi possible d'utiliser une méthode basée sur la distance du chi-deux. On va chercher à minimiser

$$Q = \sum_{i,j} \frac{E_{i,j}(N_{i,j} - L_i \cdot C_j)^2}{L_i \cdot C_j}$$

Là encore on utilise les conditions du premier ordre, et on obtient

$$L_i = \left(\frac{\sum_j \left(\frac{E_{i,j} Y_{i,j}^2}{C_j} \right)}{\sum_j E_{i,j} C_j} \right)^{\frac{1}{2}}$$

et une expression du même genre pour C_j.

```
> L=matrix(NA,100,2);C=matrix(NA,100,3)
> L[1,]=c(1,1);colnames(L)=rownames(EXPOSITION)
```

```
> C[1,]=c(1,1,1);colnames(C)=colnames(EXPOSITION)
> for(j in 2:100){
+ L[j,1]=sqrt(sum(EXPOSITION[1,]*FREQUENCE[1,]^2/C[j-1,])/sum(EXPOSITION[1,]*C[j-1,]))
+ L[j,2]=sqrt(sum(EXPOSITION[2,]*FREQUENCE[2,]^2/C[j-1,])/sum(POPULATION[2,]*C[j-1,]))
+ C[j,1]=sqrt(sum(EXPOSITION[,1]*FREQUENCE[,1]^2/L[j,])/sum(EXPOSITION[,1]*L[j,]))
+ C[j,2]=sqrt(sum(EXPOSITION[,2]*FREQUENCE[,2]^2/L[j,])/sum(EXPOSITION[,2]*L[j,]))
+ C[j,3]=sqrt(sum(EXPOSITION[,3]*FREQUENCE[,3]^2/L[j,])/sum(EXPOSITION[,3]*L[j,]))
+ }
> PREDICTION3=SINISTRE
> PREDICTION3[1,]=L[100,1]*C[100,]
> PREDICTION3[2,]=L[100,2]*C[100,]
> PREDICTION3
   X2
X1       [0,3)     [3,10)   [10,101)
  D 0.08556959 0.09529172 0.09358149
  E 0.05254307 0.05851284 0.05746269
```

	[0,3)	[3,10)	[10,101)
D	0.0856	0.0953	0.0936
E	0.0525	0.0585	0.0575

Remarque : on le verra par la suite, cette méthode donne les mêmes résultats qu'une régression Gaussienne, avec un lien logarithmique

Une régression - standard - Gaussienne

$$\frac{Y_i}{E_i} = \beta_0 + \beta_1 X_{1,i} + \beta_2 X_{2,i} + \varepsilon_i$$

(avec des poids liés à l'exposition) donnerait ici

```
>  reggaus <- lm(N/E~X1+X2,weight=E,data=donnees)
>  matrix(predict(reggaus,newdata=newdonnees,
+  type="response"),2,3)
          [,1]       [,2]       [,3]
```

```
[1,] 0.07575523 0.08296676 0.07808159
[2,] 0.06279835 0.07000988 0.06512471
```

	1	2	3
1	0.0758	0.0830	0.0781
2	0.0628	0.0700	0.0651

On va ici chercher un modèle Gaussien avec un lien log,

$$Y_i = E_i \cdot \exp\left[\beta_0 + \beta_1 X_{1,i} + \beta_2 X_{2,i}\right] + \varepsilon_i$$

(malheureusement ici, l'algorithme ne converge pas...)

```
>  reggauslog <- glm(N~X1+X2,offset=log(E),data=donnees,
+  family=gaussian(link="log"),start=coefficients(reggaus))
Erreur dans eval(expr, envir, enclos) :
  impossible de trouver des valeurs initiales correctes : priÃ¨re d'en fournir
```

La régression de Poisson

L'idée est la même que pour la régression logistique : on cherche un modèle linéaire pour la moyenne. En l'occurence,

$$Y_i \sim \mathcal{P}(\lambda_i) \text{ avec } \lambda_i = \exp[\boldsymbol{X}_i'\boldsymbol{\beta}].$$

Dans ce modèle, $\mathbb{E}(Y_i|\boldsymbol{X}_i) = \text{Var}(Y_i|\boldsymbol{X}_i) = \lambda_i = \exp[\boldsymbol{X}_i'\boldsymbol{\beta}]$.

Remarque : on posera parfois $\theta_i = \eta_i = \boldsymbol{X}_i'\boldsymbol{\beta}$

La log-vraisemblance est ici

$$\log \mathcal{L}(\boldsymbol{\beta}; \boldsymbol{Y}) = \sum_{i=1}^{n} [Y_i \log(\lambda_i) - \lambda_i - \log(Y_i!)]$$

ou encore

$$\log \mathcal{L}(\boldsymbol{\beta}; \boldsymbol{Y}) = \sum_{i=1}^{n} Y_i \cdot [\boldsymbol{X}_i'\boldsymbol{\beta}] - \exp[\boldsymbol{X}_i'\boldsymbol{\beta}] - \log(Y_i!)$$

Le gradient est ici

$$\nabla \log \mathcal{L}(\boldsymbol{\beta}; \boldsymbol{Y}) = \frac{\partial \log \mathcal{L}(\boldsymbol{\beta}; \boldsymbol{Y})}{\partial \boldsymbol{\beta}} = \sum_{i=1}^{n} (Y_i - \exp[\boldsymbol{X}_i' \boldsymbol{\beta}]) \boldsymbol{X}_i'$$

alors que la matrice Hessienne s'écrit

$$H(\boldsymbol{\beta}) = \frac{\partial^2 \log \mathcal{L}(\boldsymbol{\beta}; \boldsymbol{Y})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}'} = -\sum_{i=1}^{n} (Y_i - \exp[\boldsymbol{X}_i' \boldsymbol{\beta}]) \boldsymbol{X}_i \boldsymbol{X}_i'$$

La recherche du maximum de $\log \mathcal{L}(\boldsymbol{\beta}; \boldsymbol{Y})$ est obtenu (numériquement) par l'algorithme de Newton-Raphson,

1. partir d'une valeur initiale $\boldsymbol{\beta}_0$
2. poser $\boldsymbol{\beta}_k = \boldsymbol{\beta}_{k-1} - H(\boldsymbol{\beta}_{k-1})^{-1} \nabla \log \mathcal{L}(\boldsymbol{\beta}_{k-1})$

où $\nabla \log \mathcal{L}(\boldsymbol{\beta})$ est le gradient, et $H(\boldsymbol{\beta})$ la matrice Hessienne (on parle parfois de Score de Fisher).

Par exemple, si on régresse sur l'âge du véhicule

```
Y <- baseFREQ$nbre
X1 <- baseFREQ$agevehicule
X=cbind(rep(1,length(X1)),X1)
```

on part d'une valeur initiale (e.g. une estimation classique de modèle linéaire)

```
beta=lm(Y~0+X)$coefficients
```

On fait ensuite une boucle (avec 50,000 lignes, l'algorithme du cours 2. ne fonctionne pas)

```
> for(s in 1:20){
+ gradient=t(X)%*%(Y-exp(X%*%beta))
+ hessienne=matrix(0,ncol(X),ncol(X))
+ for(i in 1:nrow(X)){
+ hessienne=hessienne + as.numeric(exp(X[i,]%*%beta))* (X[i,]%*%t(X[i,]))}
+ beta=beta+solve(hessienne)%*%gradient
}
```

Ici, on obtient les valeurs suivantes

```
             [,1]        [,2]
[1,]  0.034651 0.0005412
[2,]  -0.9318582 0.001040094
[3,]  -1.843778 0.002294
[4,]  -2.624283 0.005060941
[5,]  -3.144156 0.009455123
[6,]  -3.334736 0.01274805
[7,]  -3.354665 0.01331968
[8,]  -3.354849 0.01332686
[19,]  -3.354849 0.01332686
```

i.e. on converge (et vite là encore).

La régression de Poisson

On peut montrer que $\widehat{\boldsymbol{\beta}} \xrightarrow{\mathbb{P}} \boldsymbol{\beta}$ et

$$\sqrt{n}(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}) \xrightarrow{\mathcal{L}} \mathcal{N}(\mathbf{0}, I(\boldsymbol{\beta})^{-1}).$$

Numériquement, la encore, on peut approcher $I(\boldsymbol{\beta})^{-1}$ qui est la variance (asymptotique) de notre estimateur. Or $I(\boldsymbol{\beta}) = H(\boldsymbol{\beta})$, donc les écart-types de $\widehat{\boldsymbol{\beta}}$ sont

```
> sqrt(diag(solve(hessienne)))
[1] 0.036478386 0.003782998
```

On retrouve toutes ces valeurs en utilisant

```
> summary(regPoisson(N~X1))

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.354849   0.036478 -91.968  < 2e-16 ***
X1           0.013327   0.003783   3.523 0.000427 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 12931  on 49999  degrees of freedom
Residual deviance: 12919  on 49998  degrees of freedom
AIC: 16594

Number of Fisher Scoring iterations: 6
```

Prise en compte de l'exposition (offset)

oups, on a oublié de prendre l'exposition dans notre modèle. On a ajusté un modèle de la forme

$$Y_i \sim \mathcal{P}(\lambda_i) \text{ avec } \lambda_i = \exp[\beta_0 + \beta_1 X_{1,i}]$$

mais on voudrait

$$Y_i \sim \mathcal{P}(\lambda_i \cdot E_i) \text{ avec } \lambda_i = \exp[\beta_0 + \beta_1 X_{1,i}]$$

ou encore

$$Y_i \sim \mathcal{P}(\tilde{\lambda}_i) \text{ avec } \tilde{\lambda}_i = E_i \cdot \exp[\beta_0 + \beta_1 X_{1,i}] = \exp[\beta_0 + \beta_1 X_{1,i} + \log(E_i)]$$

Aussi, l'exposition intervient comme une variable de la régression, mais en prenant le logarithme de l'exposition, et en forçant le paramètre à être unitaire, i.e.

$$Y_i \sim \mathcal{P}(\tilde{\lambda}_i) \text{ avec } \tilde{\lambda}_i = E_i \cdot \exp[\beta_0 + \beta_1 X_{1,i}] = \exp[\beta_0 + \beta_1 X_{1,i} + 1 \log(E_i)]$$

```
> summary(regPoisson(N~X1+offset(log(E))))

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.586986   0.038389 -67.388   <2e-16 ***
X1          -0.004364   0.004103  -1.064    0.287
---
Signif. codes:  0 â***â 0.001 â**â 0.01 â*â 0.05 â.â 0.1 â â 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 12680  on 49999  degrees of freedom
Residual deviance: 12679  on 49998  degrees of freedom
AIC: 16354

Number of Fisher Scoring iterations: 6
```

Régression de Poisson multiple

On peut (bien entendu) régresser sur plusieurs variables explicatives

```
> model1=regPoisson(nbre~zone+as.factor(puissance)+agevehicule+
+        ageconducteur+carburant+offset(log(exposition)),data=baseFREQ)
> summary(model1)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)           -2.546970   0.122723 -20.754  < 2e-16 ***
zoneB                  0.011487   0.097559   0.118   0.9063
zoneC                  0.196208   0.077090   2.545   0.0109 *
zoneD                  0.403382   0.078788   5.120 3.06e-07 ***
zoneE                  0.594872   0.079207   7.510 5.90e-14 ***
zoneF                  0.684673   0.143612   4.768 1.87e-06 ***
as.factor(puissance)5  0.135072   0.081393   1.659   0.0970 .
as.factor(puissance)6  0.161305   0.079692   2.024   0.0430 *
as.factor(puissance)7  0.164168   0.079039   2.077   0.0378 *
as.factor(puissance)8  0.122254   0.110876   1.103   0.2702
as.factor(puissance)9  0.181978   0.123996   1.468   0.1422
```

```
as.factor(puissance)10  0.254358   0.119777   2.124   0.0337 *
as.factor(puissance)11  0.001156   0.170163   0.007   0.9946
as.factor(puissance)12  0.243677   0.223207   1.092   0.2750
as.factor(puissance)13  0.513950   0.284159   1.809   0.0705 .
as.factor(puissance)14  0.582564   0.295482   1.972   0.0487 *
as.factor(puissance)15  0.173748   0.383322   0.453   0.6504
agevehicule             0.001467   0.004191   0.350   0.7264
ageconducteur          -0.008844   0.001658  -5.335 9.58e-08 ***
carburantE             -0.201780   0.049265  -4.096 4.21e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 12680  on 49999  degrees of freedom
Residual deviance: 12524  on 49980  degrees of freedom
AIC: 16235

Number of Fisher Scoring iterations: 6
```

Effets marginaux, et élasticité

Les effets marginaux de la variable k pour l'individu i sont donnés par

$$\frac{\partial \mathbb{E}(Y_i|\boldsymbol{X}_i)}{\partial X_{i,k}} = \frac{\partial \exp[\boldsymbol{X}_i'\boldsymbol{\beta}]}{\partial X_k} = \exp[\boldsymbol{X}_i'\boldsymbol{\beta}] \cdot \beta_k$$

estimés pas $\exp[\boldsymbol{X}_i'\widehat{\boldsymbol{\beta}}] \cdot \widehat{\beta}_k$

Par exemple, pour avoir l'effet marginal de la variable `ageconducteur`,

```
> coef(model1)[19]
ageconducteur
 -0.008844096
> baseFREQ[1:4,]
  nocontrat exposition zone puissance agevehicule ageconducteur
1        27       0.87    C         7           0            56
2       115       0.72    D         5           0            45
3       121       0.05    C         6           0            37
4       142       0.90    C        10          10            42
```

...

```
> effet19=predict(model1,type="response")*coef(model1)[19]
> effet19[1:4]
            1             2             3             4
-5.265678e-04 -4.690555e-04 -3.569767e-05 -6.846838e-04
```

On peut aussi calculer les effets marginaux moyens de la variable k, $\overline{Y} \cdot \widehat{\beta}_k$

```
> mean(predict(model1,type="response"))*coef(model1)[19]
ageconducteur
-0.0003403208
```

Autrement dit, en vieillissant d'un an, il y aura (en moyenne) 0.00034 accident de moins, par an, par assuré.

Ici, on utilise des changements en unité ($\partial X_{i,k}$), mais il est possible détudier l'impact de changement en proportion. Au lieu de varier d'une utilité, on va considérer un changement de 1%.

Interprétation, suite

Ici, les zones prennent les valeurs A B C D E ou F, selon la densité en nombre d'habitants par km^2 de la commune de résidence (A ="1-50", B="50-100", C="100-500", D="500-2,000", E="2,000-10,000", F="10,000+").

Dans la sortie, nous avons obtenu

```
Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
carburantE           -0.201780   0.049265  -4.096 4.21e-05 ***
```

i.e.

```
> coefficients(model1)["carburantE"]
carburantE
-0.2017798
```

Autrement dit, à caractéristiques identiques, un assuré conduisant un véhicule essence a une fréquence de sinistres presque 20% plus faible qu'un assuré conduisant un véhicule diesel,

```
> exp(coefficients(model1)["carburantE"])
carburantE
 0.8172749
```

Si on regarde maintenant la zone, la zone A a disparu (c'est la zone de référence). On notera que la zone B n'est pas significativement différente de la zone A.

```
Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
zoneB                0.011487   0.097559   0.118   0.9063
zoneC                0.196208   0.077090   2.545   0.0109 *
zoneD                0.403382   0.078788   5.120 3.06e-07 ***
zoneE                0.594872   0.079207   7.510 5.90e-14 ***
zoneF                0.684673   0.143612   4.768 1.87e-06 ***
```

Notons que l'on pourrait choisir une autre zone de référence

```
> baseFREQ$zone=relevel(baseFREQ$zone,"C")
```

Si on refait la régression, on trouve que tous les zones sont disinctes de la zone C.

```
> model1=regPoisson(nbre~zone+as.factor(puissance)+agevehicule+
+        ageconducteur+carburant+offset(log(exposition)),data=baseFREQ)
> summary(model1)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)           -2.350761   0.112384 -20.917  < 2e-16 ***
zoneA                 -0.196208   0.077090  -2.545 0.010921 *
zoneB                 -0.184722   0.086739  -2.130 0.033202 *
zoneD                  0.207174   0.064415   3.216 0.001299 **
zoneE                  0.398664   0.064569   6.174 6.65e-10 ***
zoneF                  0.488465   0.135765   3.598 0.000321 ***
as.factor(puissance)5  0.135072   0.081393   1.659 0.097017 .
as.factor(puissance)6  0.161305   0.079692   2.024 0.042959 *
as.factor(puissance)7  0.164168   0.079039   2.077 0.037798 *
as.factor(puissance)8  0.122254   0.110876   1.103 0.270195
as.factor(puissance)9  0.181978   0.123996   1.468 0.142209
as.factor(puissance)10 0.254358   0.119777   2.124 0.033704 *
as.factor(puissance)11 0.001156   0.170163   0.007 0.994578
as.factor(puissance)12 0.243677   0.223207   1.092 0.274960
as.factor(puissance)13 0.513950   0.284159   1.809 0.070502 .
```

```
as.factor(puissance)14  0.582564   0.295482   1.972 0.048659 *
as.factor(puissance)15  0.173748   0.383322   0.453 0.650356
agevehicule             0.001467   0.004191   0.350 0.726375
ageconducteur          -0.008844   0.001658  -5.335 9.58e-08 ***
carburantE             -0.201780   0.049265  -4.096 4.21e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 12680  on 49999  degrees of freedom
Residual deviance: 12524  on 49980  degrees of freedom
AIC: 16235

Number of Fisher Scoring iterations: 6
```

Pareil pour la zone D. En revanche, si la modalité de référence devient la zone E, on note que la zone F ne se distingue pas,

```
> baseFREQ$zone=relevel(baseFREQ$zone,"E")
```

puisque l'on a

```
Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)          -1.952098   0.112128 -17.410  < 2e-16 ***
zoneD                -0.191490   0.066185  -2.893  0.00381 **
zoneC                -0.398664   0.064569  -6.174 6.65e-10 ***
zoneA                -0.594872   0.079207  -7.510 5.90e-14 ***
zoneB                -0.583385   0.088478  -6.594 4.29e-11 ***
zoneF                 0.089801   0.135986   0.660  0.50902
```

On pourrait être tenté de regroupe A et B, et E et F. En effet, les tests de Student suggèrent des regroupement (pour chacune des paires). Pour faire un regroupement des deux paires, on fait un test de Ficher,

```
> baseFREQ$zone=relevel(baseFREQ$zone,"C")
> model1=regPoisson(nbre~zone+as.factor(puissance)+agevehicule+
+        ageconducteur+carburant+offset(log(exposition)),data=baseFREQ)
>  library(car)
>  linearHypothesis(model1,c("zoneA=zoneB","zoneE=zoneF"))
```

```
Linear hypothesis test

Hypothesis:
zoneA - zoneB = 0
zoneE - zoneF = 0

Model 1: restricted model
Model 2: nbre ~ zone + as.factor(puissance) + agevehicule + ageconducteur +
    carburant + offset(log(exposition))

  Res.Df Df  Chisq Pr(>Chisq)
1  49982
2  49980  2 0.4498     0.7986
```

On peut accepter (avec une telle p-value) un regroupement. Construisons cette nouvelle variable

```
> baseFREQ$zonesimple=baseFREQ$zone
> baseFREQ$zonesimple[baseFREQ$zone%in%c("A","B")]="A"
> baseFREQ$zonesimple[baseFREQ$zone%in%c("E","F")]="E"
```

(pour des raisons techniques, il faut le nom existe déjà, i.e on ne peut pas créer la classe AB).

```
> model1=regPoisson(nbre~zonesimple+as.factor(puissance)+agevehicule+
+        ageconducteur+carburant+offset(log(exposition)),data=baseFREQ)
> summary(model1)

Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)          -1.943339   0.111282 -17.463  < 2e-16 ***
zonesimpleD          -0.201446   0.064289  -3.133  0.00173 **
zonesimpleC          -0.408524   0.062658  -6.520 7.03e-11 ***
zonesimpleA          -0.599785   0.066128  -9.070  < 2e-16 ***
```

Lissage et modèles non-paramétriques

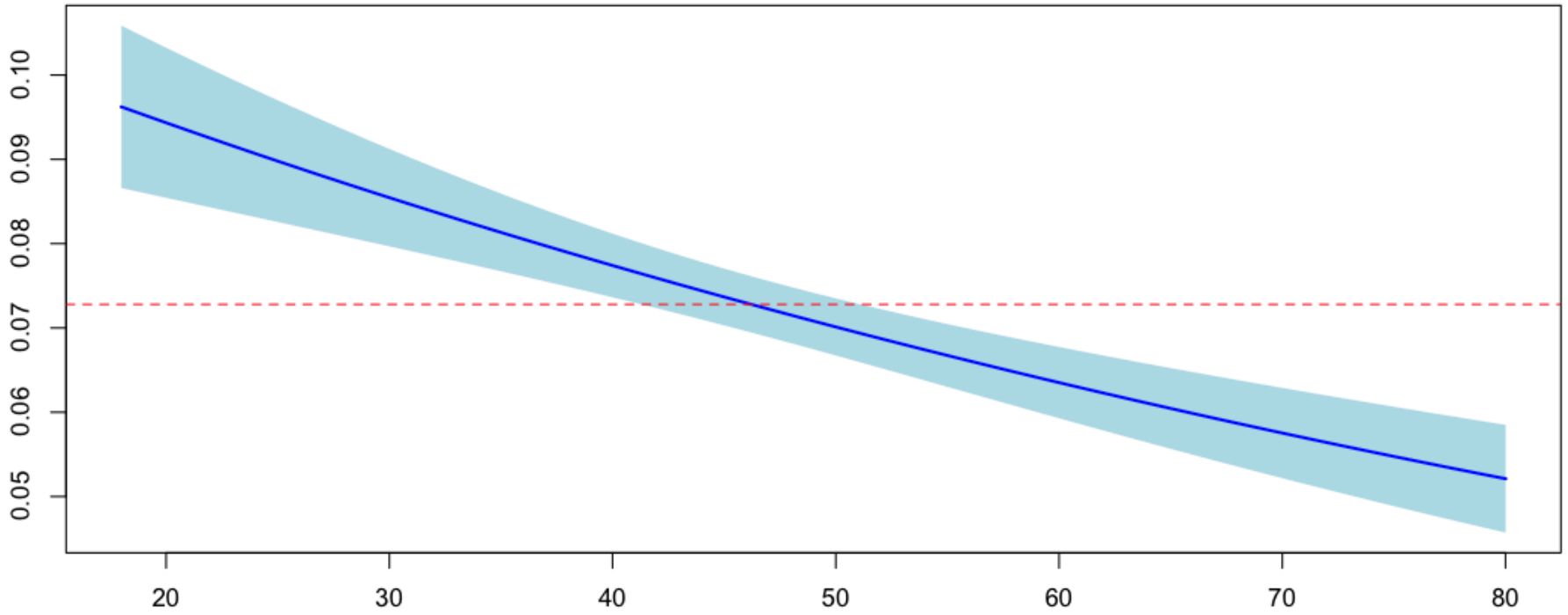
Dans le modèle où seul l'âge du conducteur intervient, on a

```
> model2=regPoisson(nbre~ageconducteur+offset(log(exposition)),data=baseFREQ)
> summary(model2)

Call:
glm(formula = formula, family = poisson(link = log), data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.5063  -0.3504  -0.2601  -0.1412  13.3387

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -2.163061   0.077380  -27.95  < 2e-16 ***
ageconducteur -0.009891   0.001635   -6.05 1.45e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

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

    Null deviance: 12680  on 49999  degrees of freedom
Residual deviance: 12642  on 49998  degrees of freedom
AIC: 16317

Number of Fisher Scoring iterations: 6
```

i.e. la fréquence annuelle de sinistre suit une loi de Poisson de paramètre $\widehat{\lambda} = \exp[\widehat{\beta}_0 + \widehat{\beta}_1 x]$ pour un assuré d'âge x.

```
age=18:80
lambda=predict(model2,newdata=data.frame(ageconducteur=age,exposition=1),type="response")
plot(age,lambda,type="l",lwd=2,col="blue")
abline(h=sum(N)/sum(E))
```



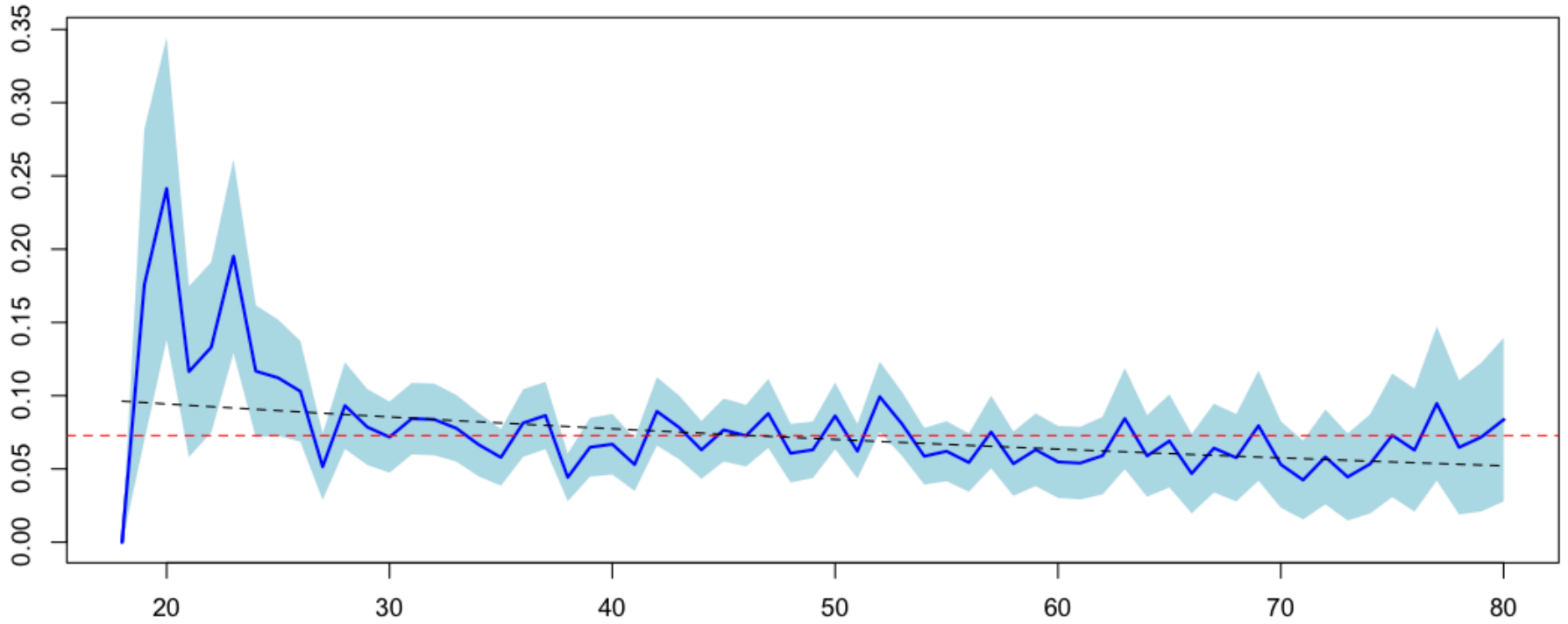
Mais est-ce raisonnable de supposer une décroissance exponentielle en fonction de l'âge ?

On peut tenter une régression paramétrique en prenant l'âge comme variable catégorielle

```
> model2b=regPoisson(nbre~as.factor(ageconducteur)+offset(log(exposition)),data=baseFREQ)
> summary(model2b)
```

```
Call:
glm(formula = formula, family = poisson(link = log), data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.6949  -0.3425  -0.2521  -0.1381  12.8912

Coefficients:
                              Estimate Std. Error z value Pr(>|z|)
(Intercept)                  -14.2522   233.9921  -0.061    0.951
as.factor(ageconducteur)19    12.5148   233.9923   0.053    0.957
as.factor(ageconducteur)20    12.8309   233.9922   0.055    0.956
as.factor(ageconducteur)21    12.1014   233.9923   0.052    0.959
as.factor(ageconducteur)22    12.2352   233.9922   0.052    0.958
as.factor(ageconducteur)23    12.6185   233.9922   0.054    0.957
as.factor(ageconducteur)24    12.1043   233.9922   0.052    0.959
as.factor(ageconducteur)25    12.0647   233.9922   0.052    0.959

...

as.factor(ageconducteur)97    -1.9218  1227.3574  -0.002    0.999
```

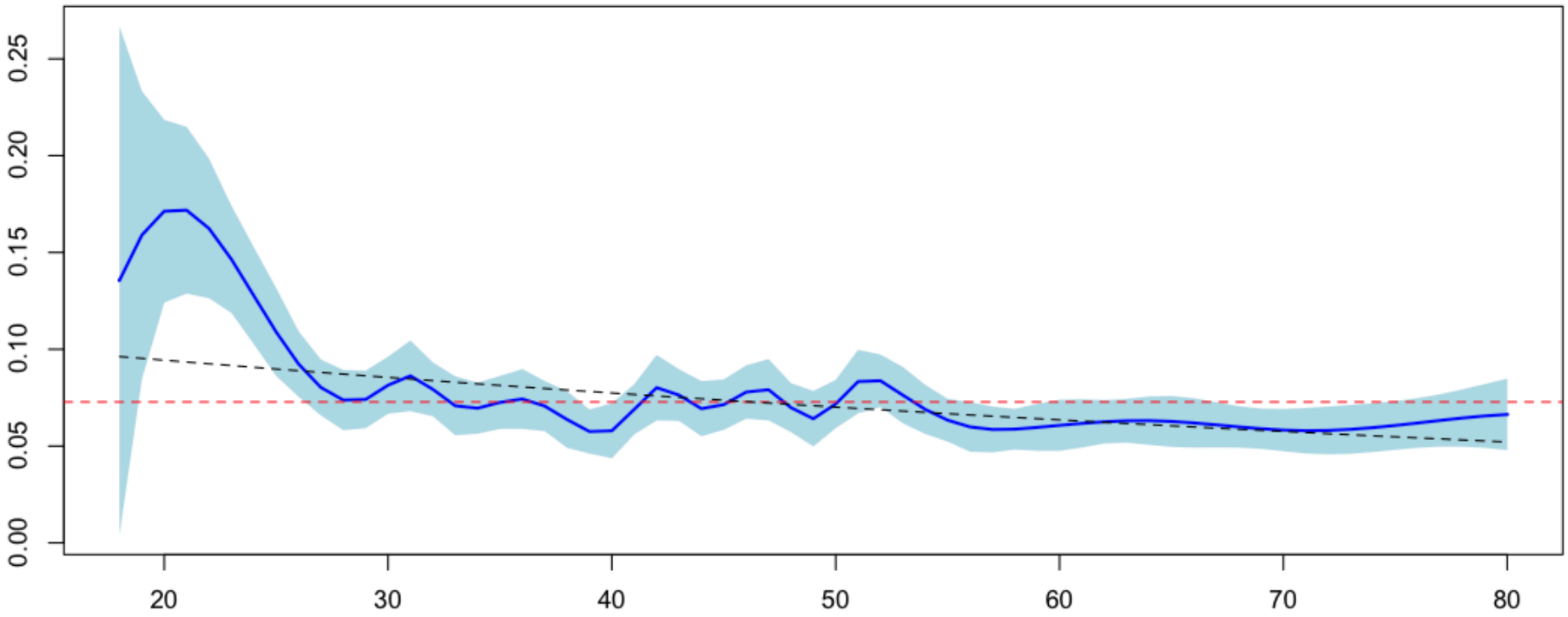
```
as.factor(ageconducteur)98    -2.0504  2116.3379  -0.001    0.999
as.factor(ageconducteur)99    -1.7031   926.5006  -0.002    0.999
as.factor(ageconducteur)100   -1.8760  2116.3379  -0.001    0.999

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 12680  on 49999  degrees of freedom
Residual deviance: 12517  on 49918  degrees of freedom
AIC: 16352

Number of Fisher Scoring iterations: 14
```



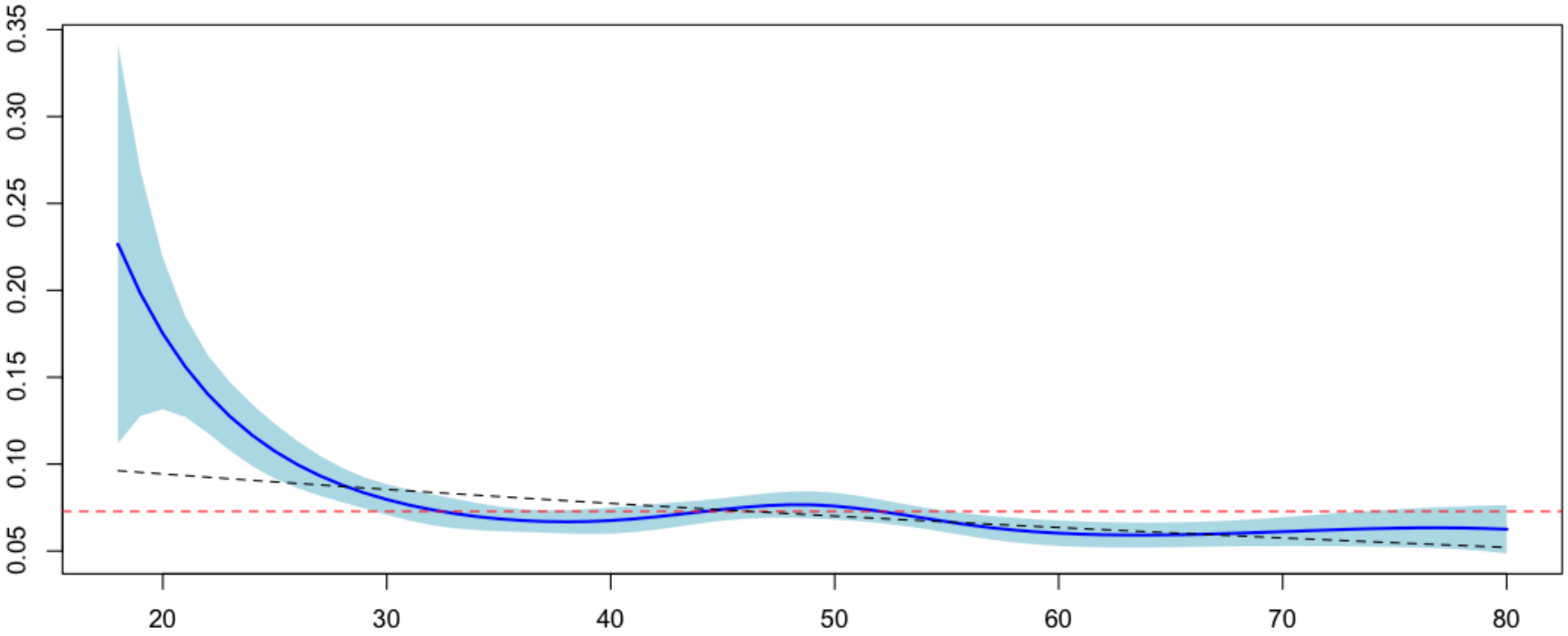
On peut utiliser des méthodes de lissage pour estimer $s(\cdot)$ telle que

$$\widehat{\lambda} = \exp[\widehat{\beta}_0 + \widehat{s}(x)]$$

Lissage et modèles non-paramétriques



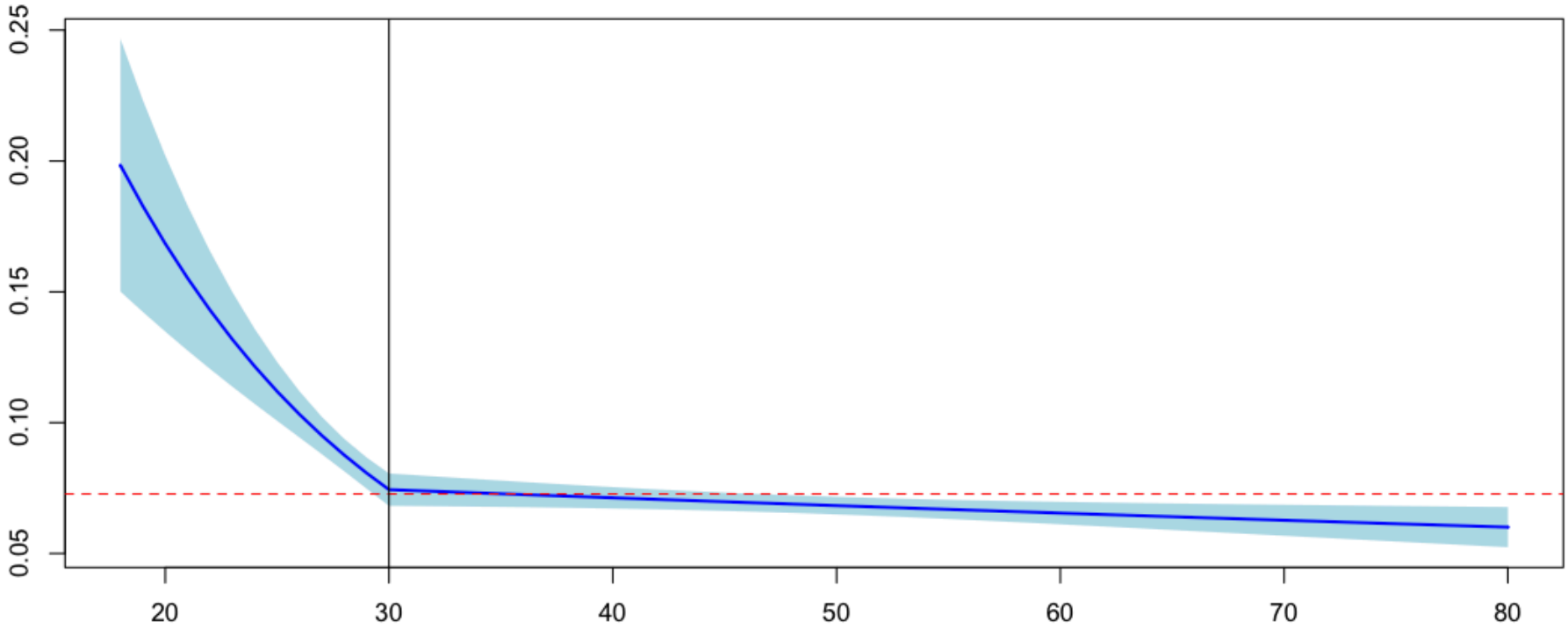
Lissage et modèles non-paramétriques



Le lissage par splines ?

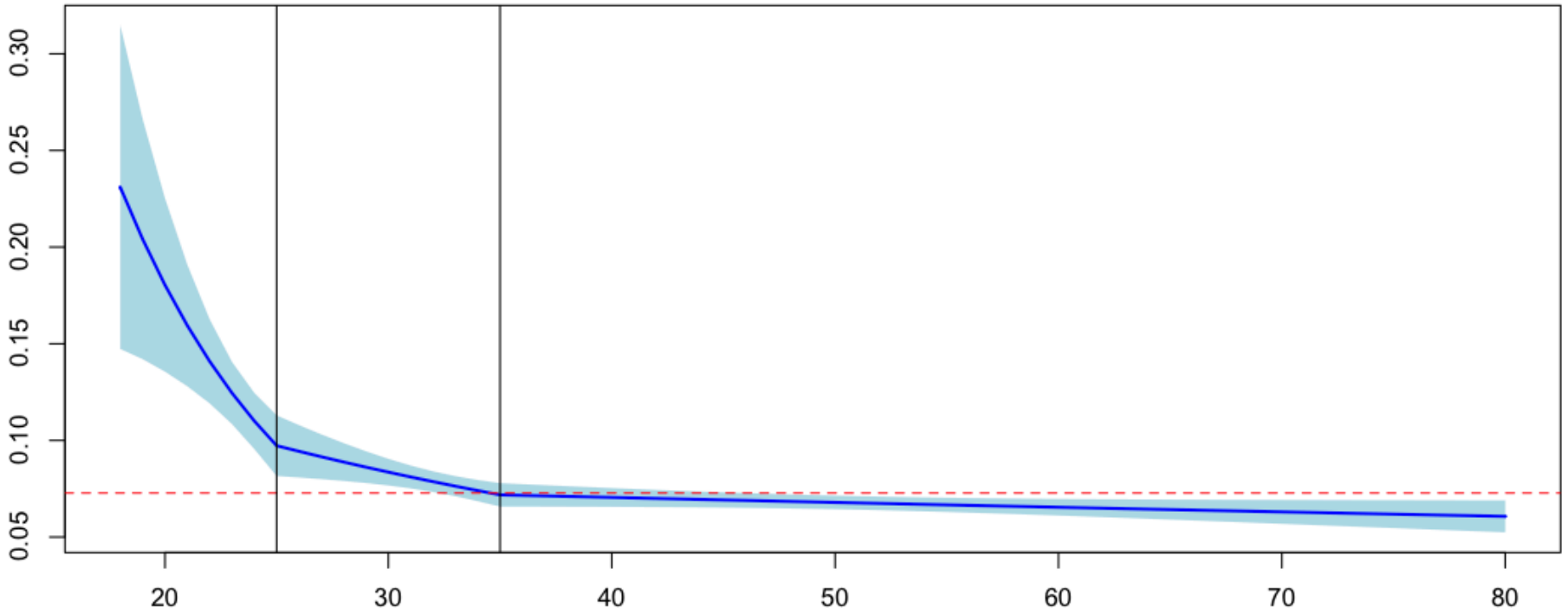
Une première idée pourrait être de faire des modèles linéaires par morceaux et continus. Par exemple, on peut introduire une rupture en $x = 30$

```
K=30
model2spline=regPoisson(nbre~bs(ageconducteur,knots=c(K),
degree=1)+offset(log(exposition)),data=baseFREQ)
lambda=predict(model2spline,newdata=data.frame(ageconducteur=age,exposition=1),type="respo
plot(age,lambda,type="l",lwd=2,col="blue")
abline(h=sum(N)/sum(E),col="red",lty=2)
abline(v=K)
```



ou encore en $x = 25$ et $x = 35$,

```
K=c(25,35)
```



Mais le lissage par des fonctions linéaires par morceaux garantie la continué de la fonction, pas de sa dérivée.

On peut introduire les blue*b*-splines de degré n, définies, sur une grille de m points t_i (appelés noeuds) dans $[0,1]$, avec $0 \leq t_0 \leq t_1 \leq \ldots \leq t_m \leq 1$, par

récurrence sur le degré inférieur par une relation de la forme

$$b_{j,0}(t) := \begin{cases} 1 & \text{si} \quad t_j \leqslant t < t_{j+1} \\ 0 & \text{sinon} \end{cases}$$

et

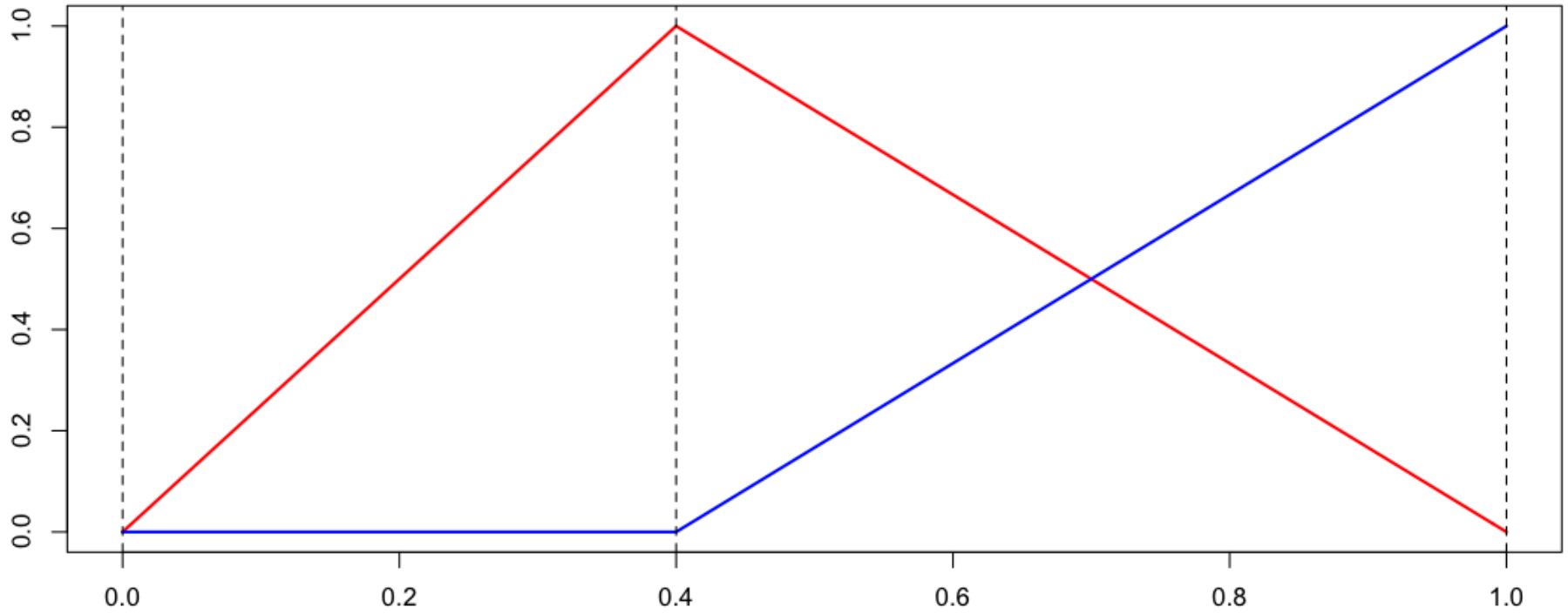
$$b_{j,n}(t) := \frac{t - t_j}{t_{j+n} - t_j} b_{j,n-1}(t) + \frac{t_{j+n+1} - t}{t_{j+n+1} - t_{j+1}} b_{j+1,n-1}(t).$$

Numériquement

```
 B=function(x,j,n,K){
 b=0
 a1=a2=0
 if(((K[j+n+1]>K[j+1])&(j+n<=length(K))&(n>0))==TRUE){a2=(K[j+n+1]-x)/
(K[j+n+1]-K[j+1])*B(x,j+1,n-1,K) }
 if(((K[j+n]>K[j])&(n>0))==TRUE){a1=(x-K[j])/
(K[j+n]-K[j])*B(x,j,n-1,K)}
 if(n==0){ b=((x>K[j])&(x<=K[j+1]))*1 }
 if(n>0){b=a1+a2}
 return(b)}
```

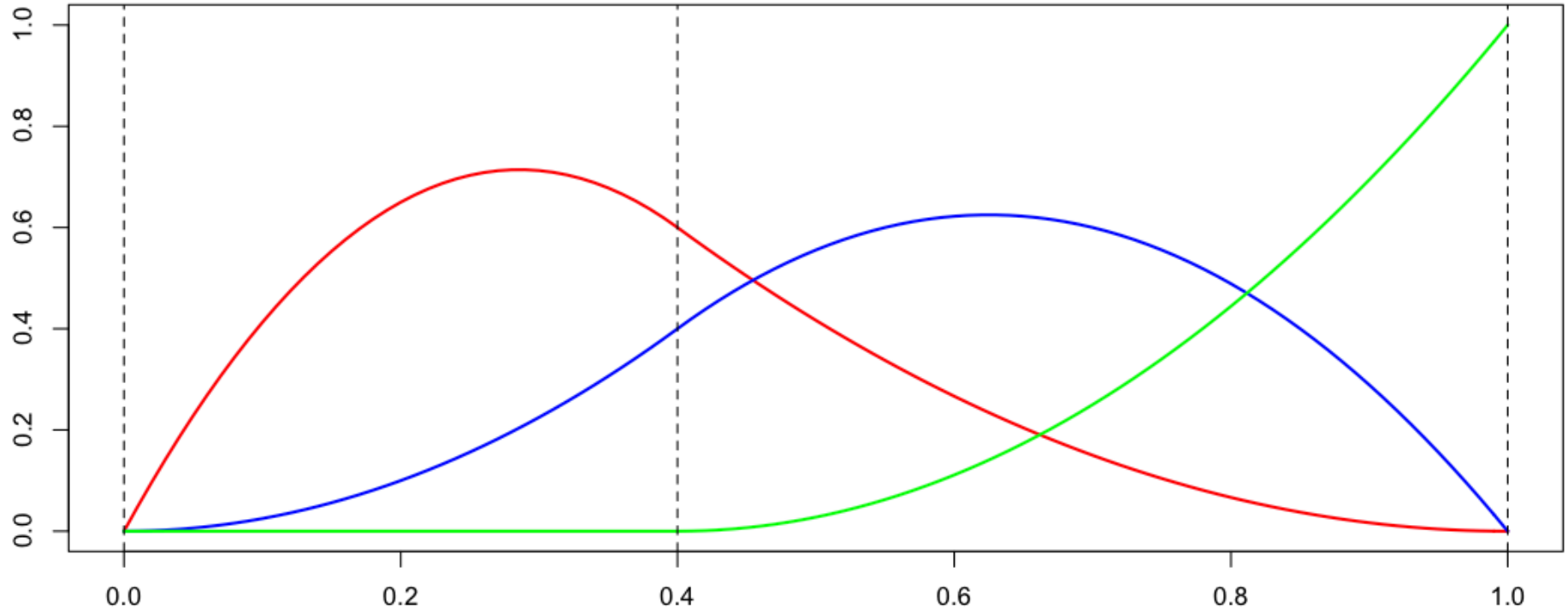
Pour les splines de degré 1, avec un seul noeud on a comme base de fonctions

```
u=seq(0,1,by=.01)
plot(u,B(u,1,1,c(0,.4,1,1)),lwd=2,col="red",type="l",ylim=c(0,1))
lines(u,B(u,2,1,c(0,.4,1,1)),lwd=2,col="blue")
abline(v=c(0,.4,1),lty=2)
```



et les splines de degré 2, avec un seul noeud on a comme base de fonctions

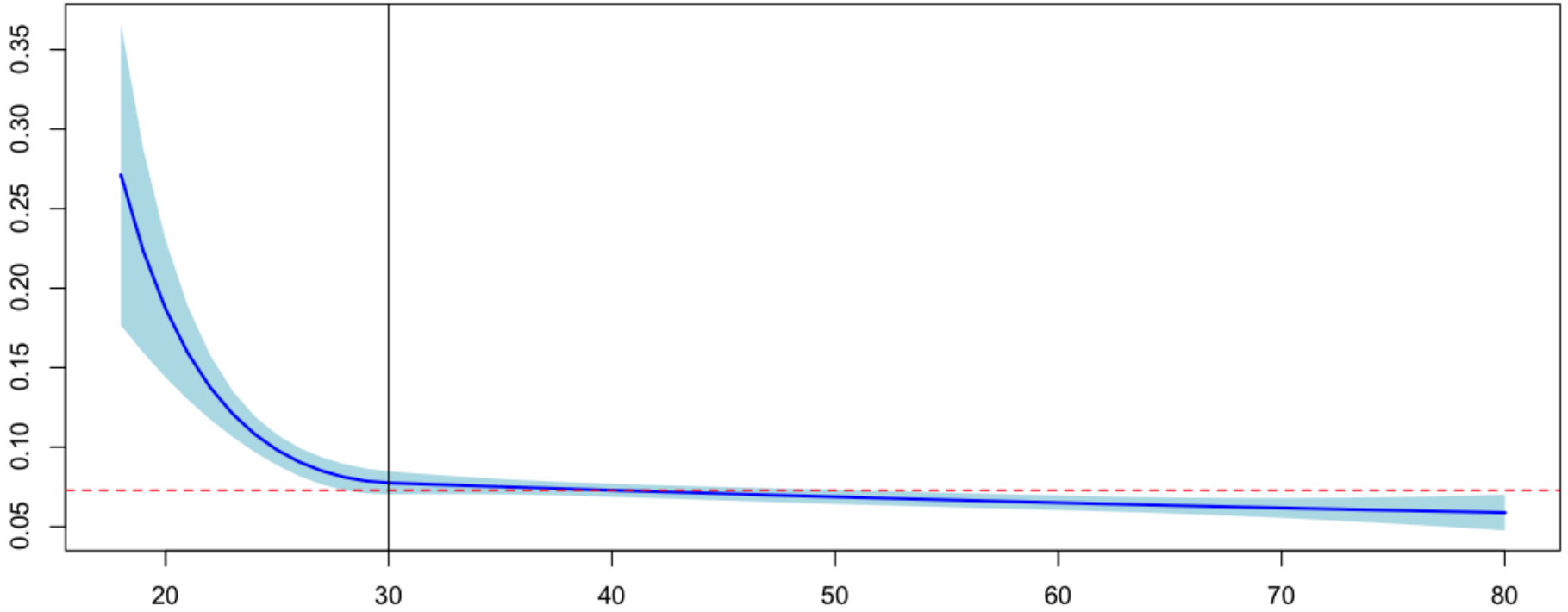
```
u=seq(0,1,by=.01)
plot(u,B(u,1,2,c(0,0,.4,1,1,1)),lwd=2,col="red",type="l",ylim=c(0,1))
lines(u,B(u,2,2,c(0,0,.4,1,1,1)),lwd=2,col="blue")
lines(u,B(u,3,2,c(0,0,.4,1,1,1)),lwd=2,col="green")
abline(v=c(0,.4,1),lty=2)
```



On peut uiliser des b-splines de degré 2 pour faire du lissage,

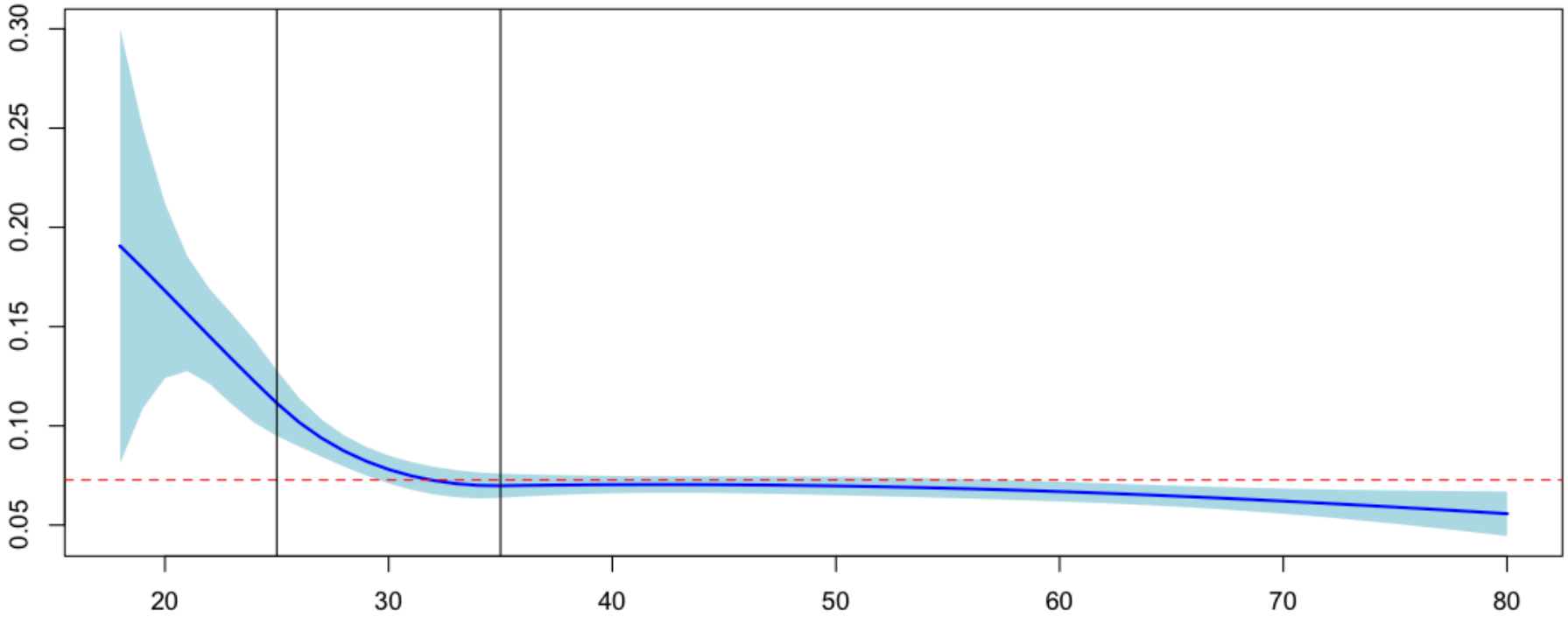
```
K=30
```

```
model2spline=regPoisson(nbre~bs(ageconducteur,knots=c(K),
degree=2)+offset(log(exposition)),data=baseFREQ)
lambda=predict(model2spline,newdata=data.frame(ageconducteur=age,exposition=1),type="respo
plot(age,lambda,type="l",lwd=2,col="blue")
abline(h=sum(N)/sum(E),col="red",lty=2)
abline(v=K)
```



avec un noeud, ou deux,

```
K=c(25,35)
model2spline=regPoisson(nbre~bs(ageconducteur,knots=c(K),
degree=2)+offset(log(exposition)),data=baseFREQ)
```

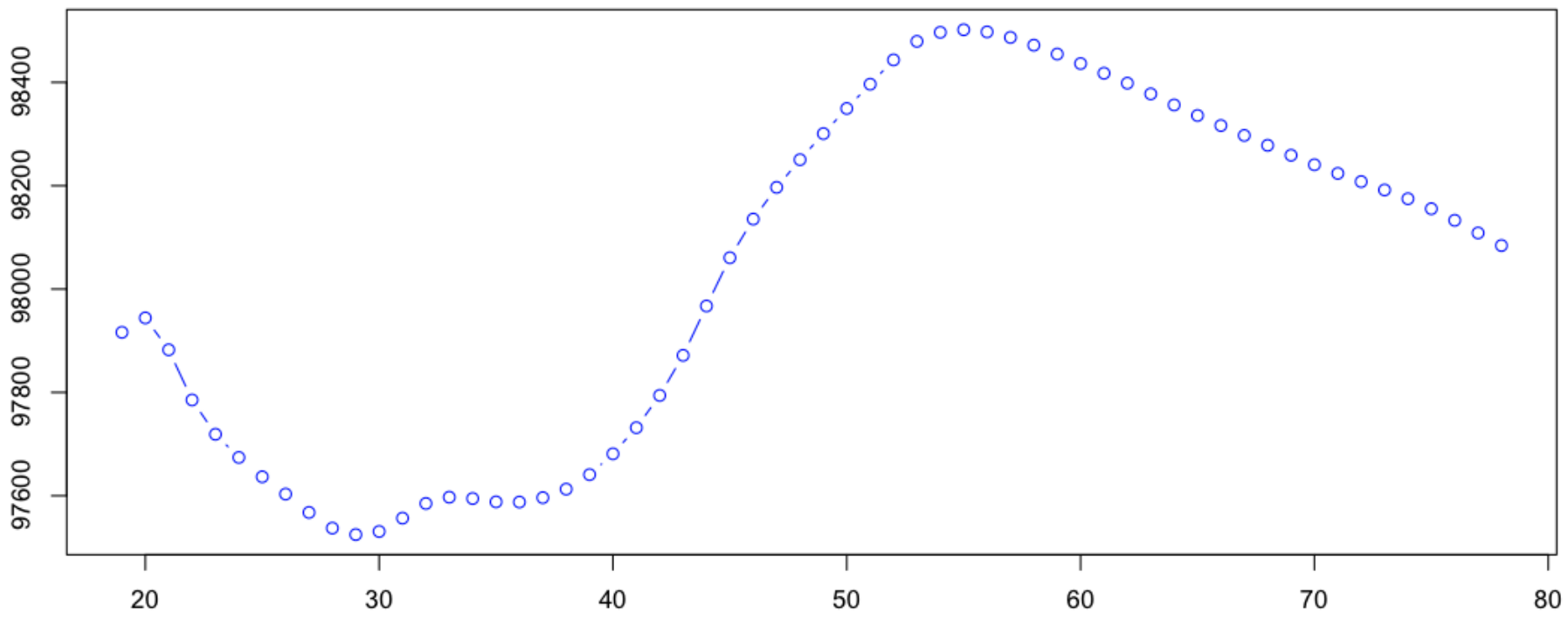


Pour trouver où mettre le noeud de manière optimale, on peut fixer un critère (classique) comme minimiser la somme des carrés des erreurs (de Pearson, e.g.) ou le critère d'Akaike

```
 SSR=AKAIKE=matrix(NA,60)
for(i in 1:60){
model2spline=regPoisson(nbre~bs(ageconducteur,knots=18+i,degree=2)+offset(log(exposition))
SSR[i]=sum(residuals(model2spline,type="pearson")^2)
AKAIKE[i]=AIC(model2spline)}
```

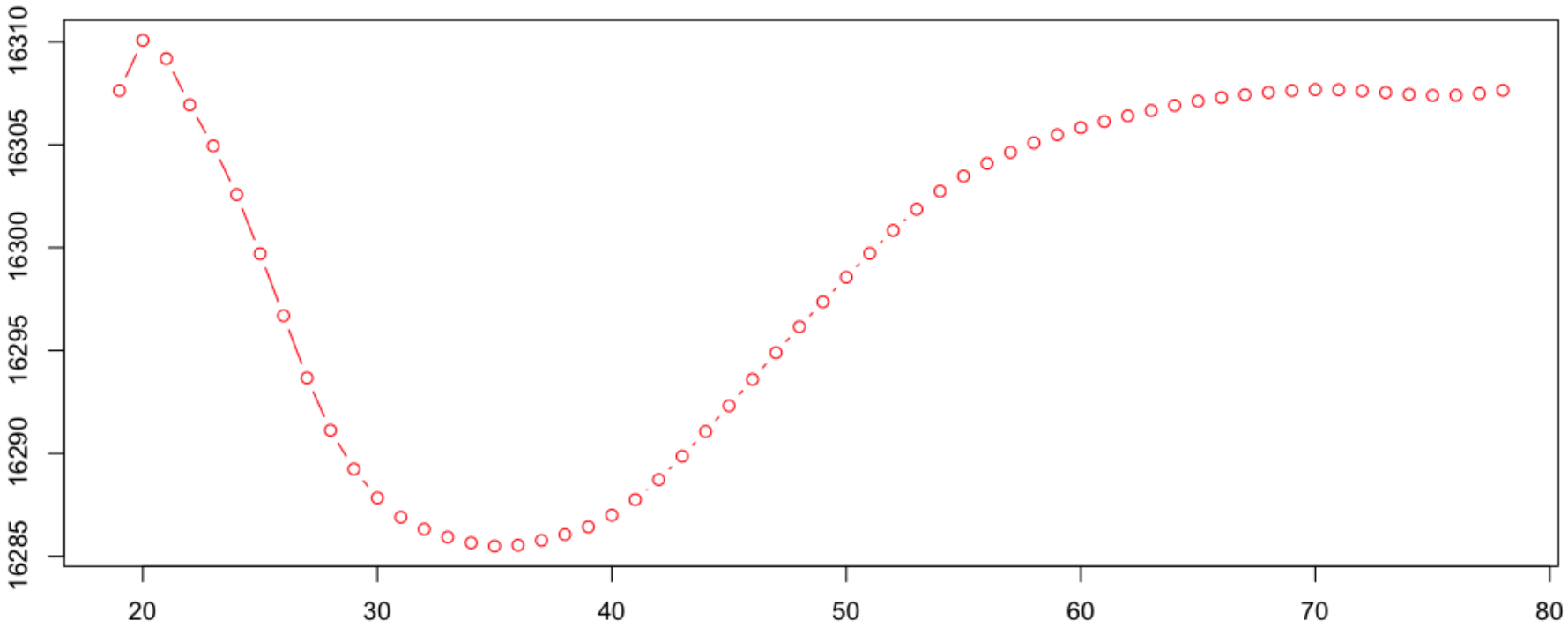
Pour la somme des carrés des résidus, on a une rupture vers 29 ans,

```
 plot(18+1:60,SSR,type="b",col="blue")
```



et pour le critère d'Akaike, on a une rupture vers 35 ans,

```
plot(18+1:60,AKAIKE,type="b",col="red")
```



Remarque : Par défaut, R utilise des splines cubiques, de degré 3.