Introduction à la régression non paramétrique

François Husson & Julie Josse

Laboratoire de mathématiques appliquées - AGROCAMPUS OUEST

4 janvier 2016

Le problème et les données

- Ozone phénomène complexe
- Enjeux de santé publique
- Mission Air Breizh : mesure, analyse, prévision → envoie tous les jours à 17 heures, l'indice de pollution du lendemain aux autorités
- Prev'Air :
  - modèle déterministe de simulation
  - national

⇒ **Modèle statistique de prévision local (Rennes) pour prévoir les concentrations maximales d'ozone du lendemain**

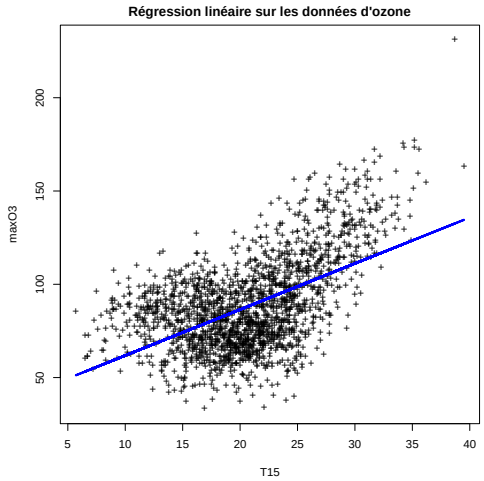
- Seuil de recommandation
- Seuil d'alerte (délai de mise en place des procédures)

Le problème et les données

Prévoir les pics d'ozone en fonction des prévisions météorologiques à Rennes (Air Breizh)

	maxO3	T6	T9	T12	T15	T18	Ne6	...	maxO3v
19940401	56	8.6	9.5	6.8	9.1	7.7	6		59.6
19940402	39.2	3.6	5.6	9.2	8.4	4.9	3		56
19940403	36	2.7	7.3	6.3	7	7.9	6		39.2
19940404	41.2	11.8	11.8	11	7	7.7	8		36
19940405	27.6	3.7	8.3	11.6	10.7	7.9	6		41.2
	...								...
	...								...
20050929	73	11.2	16	17.8	18.6	15.1	2		68
20050930	46	14.2	17.3	17.2	17.5	18	8		73

La régression linéaire simple



$$Y_i = f(x_i) + \varepsilon_i$$
$$Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$$

Approche paramétrique

- fonction de régression connue
- dépend d'un certain nombre de paramètres
- paramètres estimés à partir des données
- attractif car interprétation des paramètres et simplicité statistique

⇒ exemple le plus simple : la régression linéaire simple
⇒ ne reflète pas toujours la relation entre Y et x

Approche paramétrique : régression linéaire polynomiale

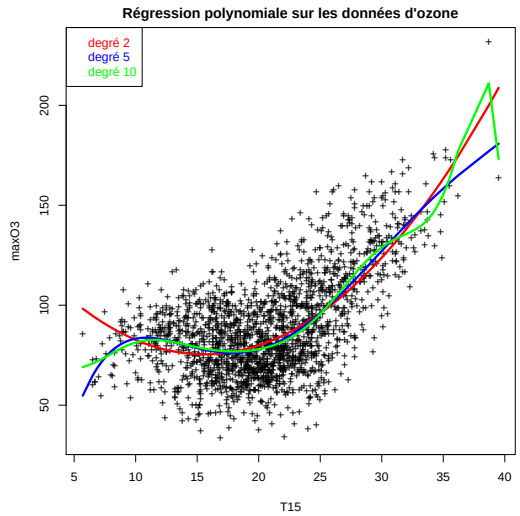
⇒ Visualisation graphique → régression linéaire n'est pas adaptée
⇒ Autre forme se dégage ?
⇒ Nouveau modèle → choix laborieux

⇒ Régression polynomiale

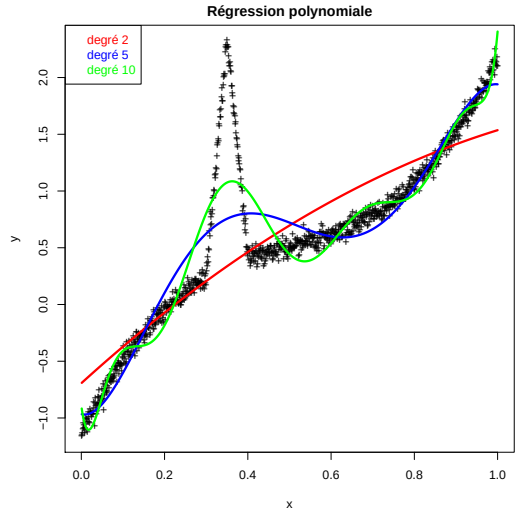
$$Y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + ... + \beta_d x_i^d + \varepsilon_i$$

- Quel ordre de polynôme choisir ?

Approche paramétrique : régression linéaire polynomiale



Approche paramétrique : régression linéaire polynomiale



Fortes variations locales ⇒ Impossibilité de les modéliser avec un polynôme, même d'ordre élevé

Approche paramétrique versus approche non paramétrique

Approche paramétrique :

$$Y = \beta_0 + \beta_1 x_i + \varepsilon_i$$

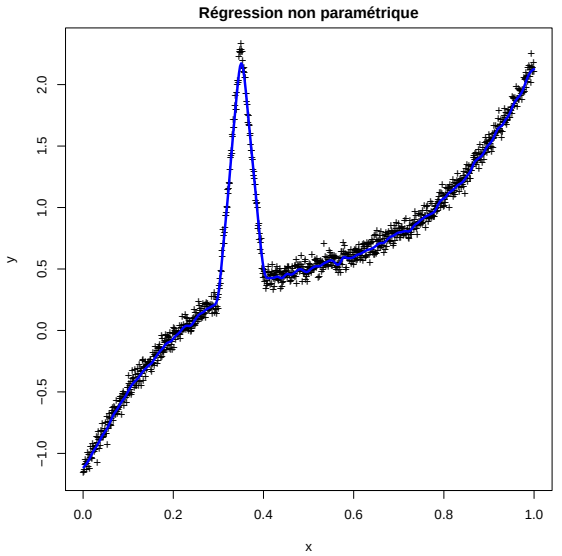
$$Y = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + ... + \beta_d x_i^d + \varepsilon_i$$

Approche non paramétrique :

$$Y = \beta_0 + f(x_i) + \varepsilon_i$$

- Pas de structure de la fonction de régression
- La relation entre Y et x est ajustée à partir des données
  "Let the data show the appropriate functional form" (Hastie)
- Avantage : flexibilité, capte des variations inattendues

Approche non paramétrique



⇒ Passe « au plus près » des données : lisseur

Approche non paramétrique

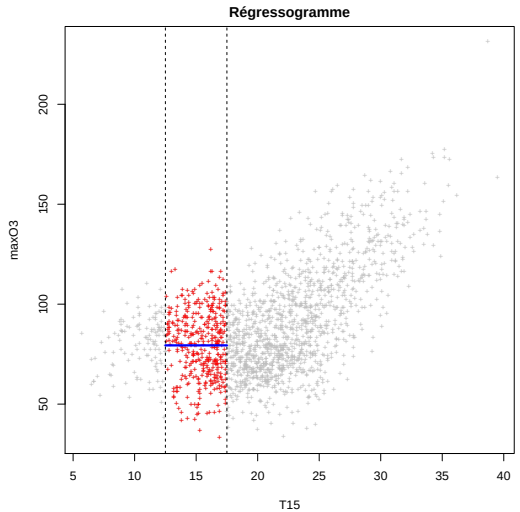
Définition d'un lisseur (Hastie) :
A smoother is a tool for summarizing the trend of a response measurement Y of one predictor X_1. It produces an estimate of the trend that is less variable than Y itself; hence the name of smoother.

- Objectif descriptif
- Estimation de la fonction de régression

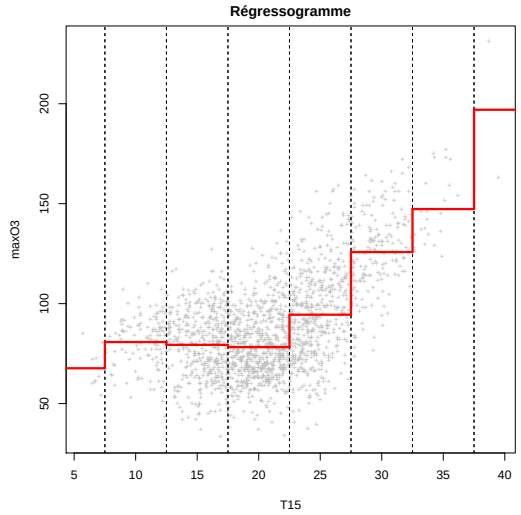
⇒ Moyenne : concept de base du lissage

Régressogramme (Bin smoother)

- Découper les x en intervalles réguliers
- Calculer la moyenne des Y dans chaque intervalle

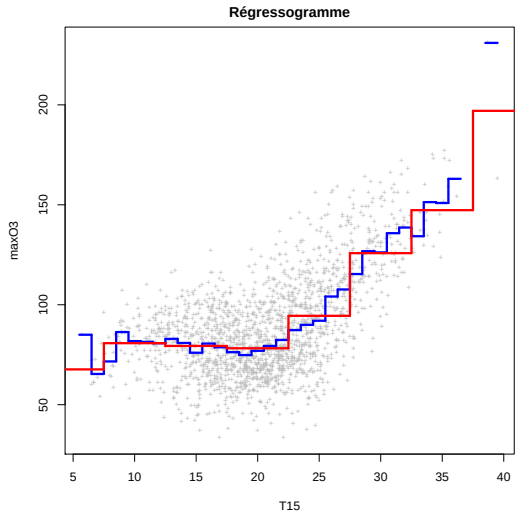


Régressogramme (Bin smoother)



- Choix de la fenêtre (dualité biais - variance)
- Problème de discontinuité ⇒ Prendre des régions qui se chevauchent

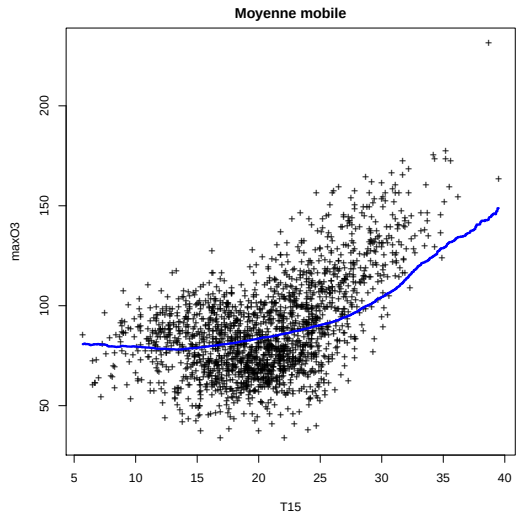
Régressogramme (Bin smoother)



Importance du choix de la fenêtre : dualité biais - variance

Moyenne mobile

- Principe : définir, en chaque point, un voisinage pour calculer la moyenne de Y (moyenne sur des intervalles glissants)
- Avantage : simple et intuitif



Moyenne mobile

Problèmes :

- Taille du voisinage
  ⇒ paramètre de lissage (dualité biais - variance)
  ⇒ non paramétrique ne signifie pas absence de paramètres !

- Poids identiques à tous les points du voisinage et nuls aux autres ⇒ Prendre des poids "continus"

Moyenne mobile pondérée : Nadaraya-Watson

Moyenne mobile calculée par :

$$\frac{\sum_i p(x_i) Y_i}{\sum_i p(x_i)}$$

avec les poids

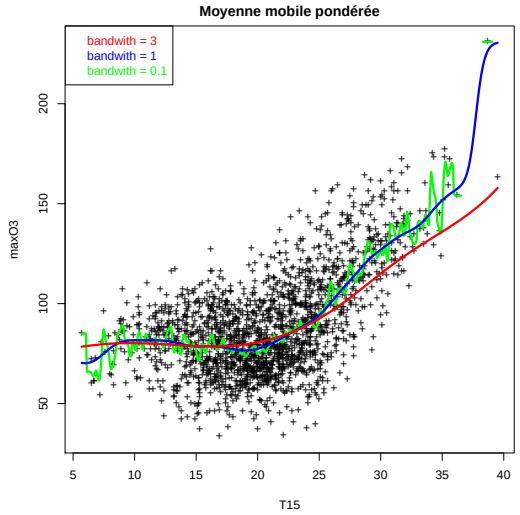
$$p(x_i) = K\left(\frac{x_i - x_0}{\lambda}\right)$$

Fonction des poids décroissante en $|x - x_0|$ et symétrique

- λ : largeur de la fenêtre
- λ élevé $\Rightarrow$ les x_i ont le même poids $\Rightarrow$ approximation est lisse
- Exemple du noyau gaussien

$$p(x_i) = \frac{1}{\sqrt{2\pi}} exp\left(-\frac{(x_i - x_0)^2}{2}\right)$$

Moyenne mobile pondérée : Nadaraya-Watson



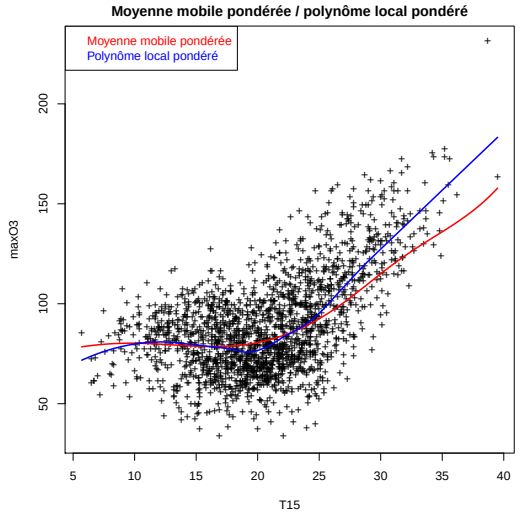

```
> library(KernSmooth)
> plot(maxO3~T15,data=ozone,main="Moyenne mobile pondérée",pch="+")
> fx=locpoly(ozone$T15,ozone$maxO3,degree=0,bandwidth=0.1)
> lines(fx$x,fx$y, col="green",lwd=2)
```


Régression polynomiale locale pondérée (loess)

Pourquoi se contenter de la moyenne ?
On en veut toujours plus : régression polynomiale locale pondérée

- méthode `loess`
- souvent on se contente de polynôme de degré 2
- choix d'un voisinage autour de x_0 ou plus proches voisins
- `span` : proportion de points constituant le voisinage

Régression polynomiale locale pondérée (loess)




```
> plot(maxO3~T15,data=ozone,main="Moyenne mobile pondérée vs polynôme local pondéré",pch="+")
> fx=locpoly(ozone\$T15,ozone\$maxO3,degree=0,bandwidth=3)
> lines(fx\$x,fx\$y, col="red",lwd=2)
> fx=loess(ozone\$maxO3[order(ozone\$T15)] ~ ozone\$T15[order(ozone\$T15)], span = 0.5, degree = 2)
> lines(ozone\$T15[order(ozone\$T15)],predict(fx), col="blue",lwd=2)
```

Régression polynomiale locale pondérée (loess)

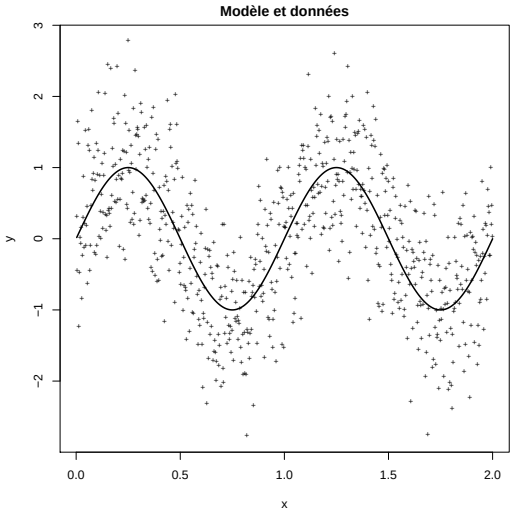
Paramètre de la méthode :

- rayon du voisinage ou proportion (span) des points pris en compte dans le lissage
  - span proche de 0 ⇒ interpolation : biais faible, variance forte
  - span proche de 1 ⇒ régression constante : biais fort, variance faible

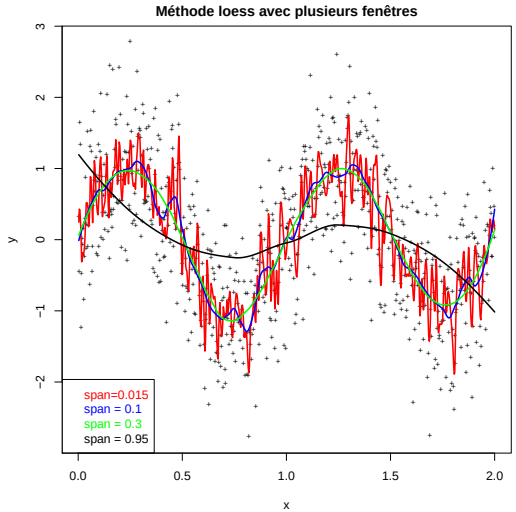
⇒ Arbitrage entre biais et variance

Choix de fenêtre

Le modèle pour générer les données et les données



Choix de fenêtre




```
> plot(y~x,cex=.7,pch="+",main="Méthode loess avec plusieurs fenêtres")
> pred <- loess(y ~ x, span = 0.015, degree = 2)
> points(pred$fitted[order(x)]~x[order(x)],col="red",lwd=2,type="l")
```

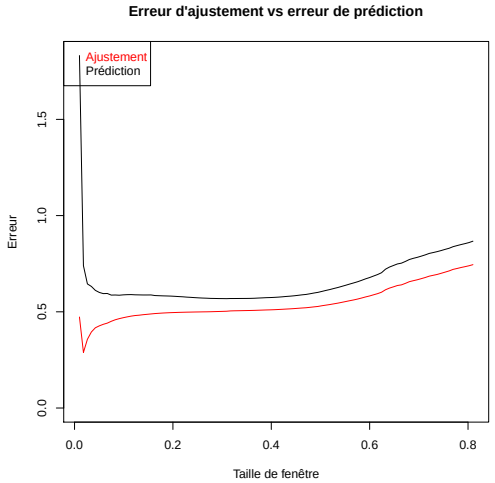

Choix de fenêtre

Estimation de la fenêtre optimale par apprentissage - validation :

- Séparer le jeu de données en proportion 2/3 pour apprentissage et 1/3 pour validation
- Faire varier la taille de la fenêtre
  - Estimer le modèle sur les données d'apprentissage
  - Calculer l'erreur sur les données de validation
- Choisir la fenêtre qui minimise les erreurs de prédiction

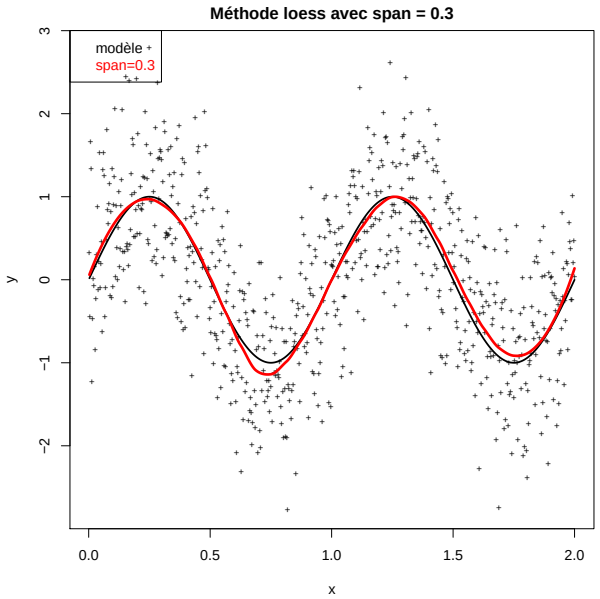
Si peu de données ⇒ validation croisée

Erreur d'ajustement - erreur de prévision



Choix de fenêtre

Avec la meilleure fenêtre (`span = 0.3`)

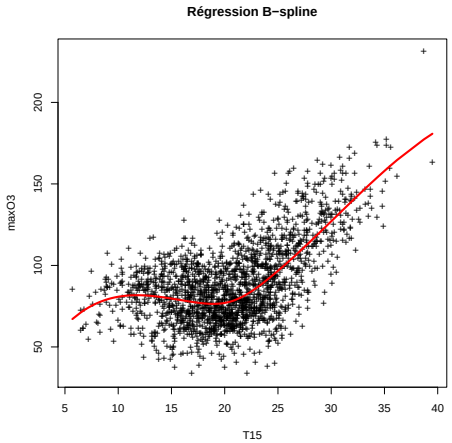


Splines

Autre type de lisseur : les splines
Régression polynomiale par morceaux :

- nécessité de déterminer les nœuds (les points de jonction) : nombre et positions
- degré du polynôme (souvent polynôme cubique)

Splines




```
library(splines)
base=bs(ozone[order(ozone$T15),"T15"],knots=quantile(ozone[,"T15"],c(.25,.5,.75)),int=FALSE,degree=3)
reg=lm(ozone[order(ozone$T15),"maxO3"]~base)
plot(maxO3~T15,data=ozone,main="Régression B-spline",pch="+")
lines(ozone[order(ozone$T15),"T15"],reg$fit, col="red",lwd=3)
```

Cas multidimensionnel

Modèle paramétrique :

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i1}^2 + \beta_4 x_{i1} x_{i2} + ... + \beta_j x_{ip} + \varepsilon_i$$

Extension naturelle au modèle non paramétrique :

$$Y_i = f(x_{i1}, x_{i2}, ..., x_{ip}) + \varepsilon_i$$

Fléau de la dimension (peu de données dans un voisinage multidimensionnel) $\Rightarrow$ Estimation trop difficile de f

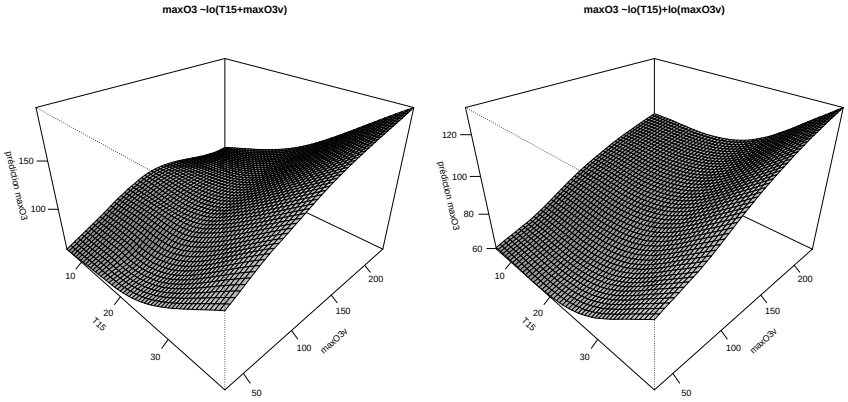
Simplification avec modèle additif :

$$Y_i = f_1(x_{i1}) + f_2(x_{i2}) + ... + f_p(x_{ip}) + \varepsilon_i$$

Cas multidimensionnel

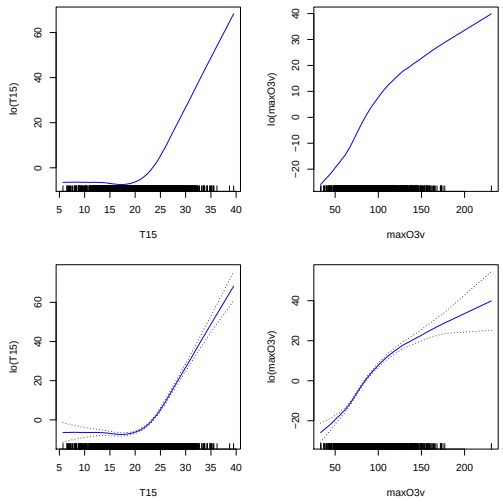
Différence entre :

- modèle non paramétrique général : $Y = f(x_1, x_2, ..., x_p) + \varepsilon$
- et modèle additif $Y = f_1(x_1) + f_2(x_2) + ... + f_p(x_p) + \varepsilon$



Modèle additif

Quel est l'effet d'un facteur sur Y, les autres étant constants ?



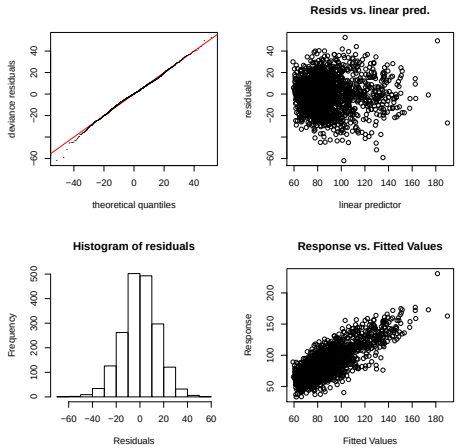

```
> library(gam)
> res.gam = gam(maxO3~lo(T15)+lo(maxO3v),data=ozone)
> plot(res.gam,ask = TRUE)
```


Package mgcv

Package très complet
Utilise principalement les splines

Propose une solution pour le choix délicat des paramètres de lissage par validation croisée généralisée

Package mgcv




```
library(mgcv)
res.mgcv = gam(maxO3~s(T15)+s(maxO3v),data=ozone)
plot(res.mgcv,col="red")
gam.check(res.mgcv)
```


Choix de modèle

Besoin de sélectionner des variables

- Test de modèles emboîtés
- Critère AIC ou BIC
- Par validation croisée : trouver le modèle à une variable qui prédit le mieux, puis à 2 variables, ...

Choix de modèle

```
> library(gam)
> res.gam = gam(maxO3~lo(T15)+lo(maxO3v),data=ozone)
> res.gam1 = gam(maxO3~lo(maxO3v),data=ozone)
> anova(res.gam1,res.gam)
Analysis of Deviance Table

Model 1: maxO3 ~ lo(maxO3v)
Model 2: maxO3 ~ lo(T15) + lo(maxO3v)
  Resid. Df Resid. Dev     Df Deviance P(>|Chi|)
1    1881.1     605476
2    1877.7     415221 3.4195   190255 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> res.gam2 = gam(maxO3~lo(T15),data=ozone)
> anova(res.gam2,res.gam)
Analysis of Deviance Table

Model 1: maxO3 ~ lo(T15)
Model 2: maxO3 ~ lo(T15) + lo(maxO3v)
  Resid. Df Resid. Dev     Df Deviance P(>|Chi|)
1    1881.6     613498
2    1877.7     415221 3.8569   198277 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```


Méthode IBR : iterative Bias Reduction

Utiliser un lisseur très lisse (donc très biaisé)

- estimer le biais
- corriger le lisseur en soustrayant le biais

et itérer.

```
library(ibr)
res.ibr <- ibr(maxO3~.,data=ozone[ind.app,],control.par=list(really.big=TRUE),df=1.1)
pred.ibr <- predict(res.ibr,newdata=ozone[ind.pred,])
```

Prendre `df = 1.1` dans la pratique

Conclusion

Peu de données $\Rightarrow$ faire des hypothèses sur les liaisons (modèles paramétriques)

Beaucoup de données $\Rightarrow$ possibilité d'utiliser des modèles additifs

Problèmes : éviter le surajustement (sélection de variables), choix des paramètres de lissage

Extension aux modèles additifs généralisés (GAM) : l'erreur peut ne pas être normale, Y peut être qualitative

Références

Packages R :

- package `KernSmooth`
- package `splines`
- fonction `gam` du package `gam`
- fonction `gam` du package `mgcv`
- fonction `ibr` du package `ibr`

Références bibliographiques :

- Trevor Hastie & Rob Tibshirani (1995). Generalized Additive Models. *Chapman & Hall*
- Simon Wood (2006). Generalized additive models :an introduction with R. *Chapman & Hall*