MUSÉUM NATIONAL D'HISTOIRE NATURELLE

SEMIN-R

Lissage et estimation fonctionnelle avec R

Christophe POUZAT
Laboratoire de Physiologie Cérébrale
CNRS ULR 8118, Université Paris-Descartes

Lissage et estimation fonctionnelle avec R

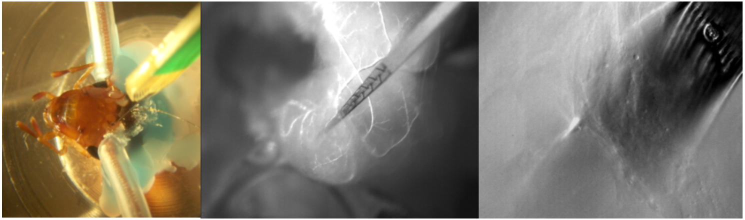
Christophe Pouzat

Laboratoire de Physiologie Cérébrale, CNRS UMR 8118, Université Paris-Descartes
e-mail: christophe.pouzat@gmail.com
web: http://www.biomedicale.univ-paris5.fr/physcerv/C_Pouzat.html

12 Février 2009

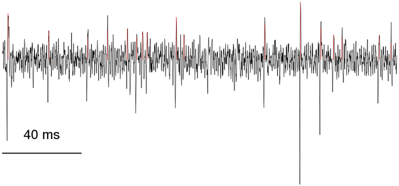
De quoi va-t-on parler ?

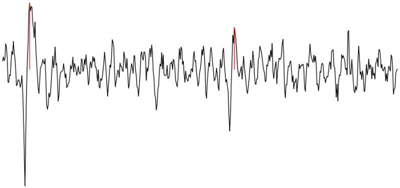
Enregistrement *in vivo* dans le premier relais olfactif d'un insecte



Une blatte (*Periplaneta americana*) "en position" avec la sonde d'enregistrement. Photos par Laurent Moreaux et Antoine Chaffiol.

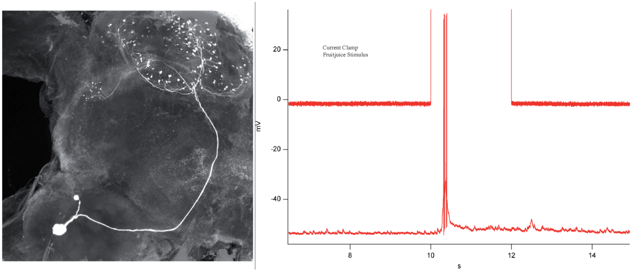
Données brutes et détection





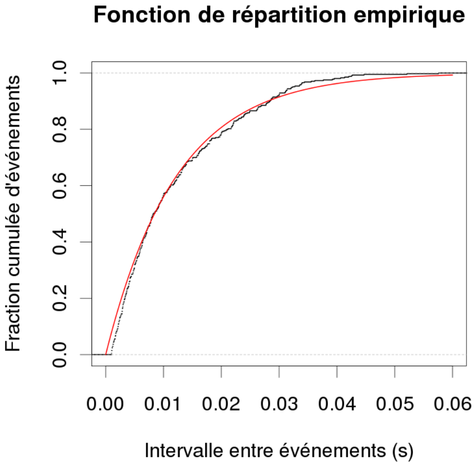
Vue de l'extérieur, l'activité des neurones se manifeste par l'émission d'impulsions électriques très brèves : les potentiels d'actions. Enregistrements par Nicole Lindemann et Antoine Chaffiol.

D'autres techniques sont aussi applicables à ce système



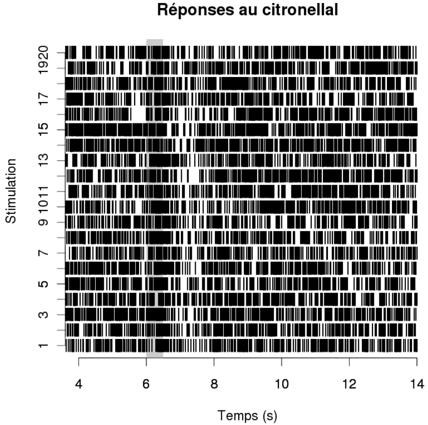
Un exemple d'enregistrement de patch-clamp couplé à l'imagerie calcique. Ici une cellule unique est enregistrée. Données obtenues par Moritz Paehler et Peter Kloppenburg (Université de Cologne).

Propriétés des décharges "brutes"



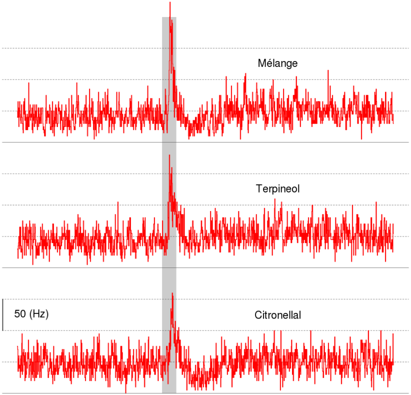
Sur des temps courts (ici 5 s) les décharges sont bien approximées par des processus de Poisson.

Un exemple de réponse à une stimulation olfactive



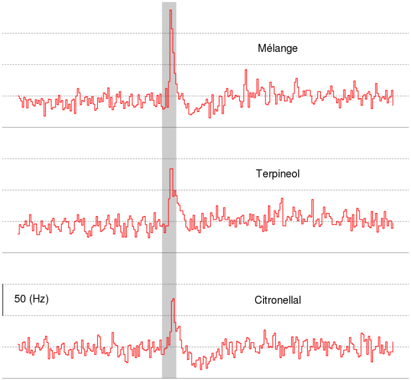
Il est difficile de distinguer quoi que ce soit avec ce type de représentation (la valve controllant l'arrivée de l'odeur est ouverte dans la partie grisée).

Estimation de la fréquence de décharge instantanée par moyennage des réponses



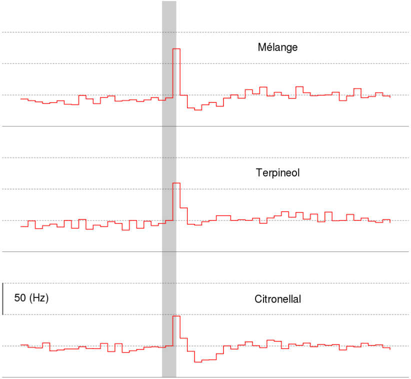
En “moyennant” les réponses des 20 stimulations et en discrétisant le temps on obtient quelque chose de plus clair (taille de la fenêtre de discrétisation : 10 ms).

Estimation de la fréquence de décharge instantanée par moyennage des réponses



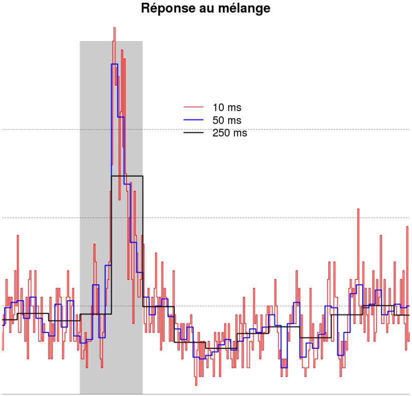
En “moyennant” les réponses des 20 stimulations et en discrétisant le temps on obtient quelque chose de plus clair (taille de la fenêtre de discrétisation : 50 ms).

Estimation de la fréquence de décharge instantanée par moyennage des réponses



En "moyennant" les réponses des 20 stimulations et en discrétisant le temps on obtient quelque chose de plus clair (taille de la fenêtre de discrétisation : 250 ms).

Le problème : quelle taille de bin choisir ?



Décharge du réseau entre 5.5 et 8.5 s en réponse au mélange.

Un peu de formalisme

- Toutes les fonctions de `R` que nous allons voir considèrent (au moins dans leur version de base) que les observations, Y_i, et le prédicteur (variable indépenante), x_i ($i = 1, \ldots, n$) sont reliés par le modèle de régression :

$$Y_i = \eta(x_i) + \epsilon_i, \quad i = 1, \ldots, n$$

- Les ϵ_i sont supposés être des variables aléatoires indépendantes et identiquement distribuées (IID) de moyenne 0 et de variance σ^2.

Homogénéité de la variance

- Avant d'aller plus loin il peut-être utile de vérifier que l'hypothèse IID pour les ϵ_i est raisonnable.
- On peut le faire avec un Bootstrap en ré-échantillonnant les 20 stimulations.
- Commençons par calculer le nombre total d'événements par bin :

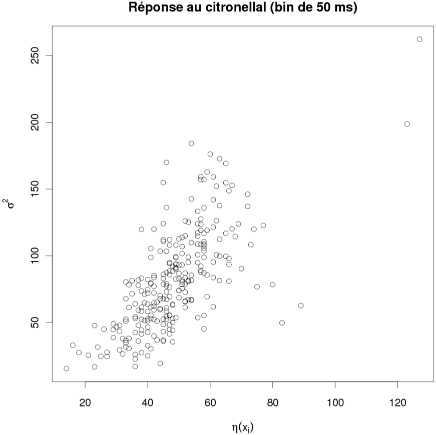
```
> citronC <- sapply(allCitron,
+ function(l) {
+ hist(l,seq(1,14,0.05),plot=FALSE)$counts
+ } )
> citronC.hat <- apply(citronC,1,sum)
```

Homogénéité de la variance

- L'estimateur de bootstrap du nombre total d'événements par bin est alors :

```
> citronC.boot <- sapply(1:1000,
+ function(idx) {
+ myCol <- sample(1:20,20,replace=TRUE)
+ bootD <- citronC[,myCol]
+ apply(bootD,1,sum)
+ }
+ )
> citronC.hat.var <- apply(citronC.boot,1,var)
```

Homogénéité de la variance



Ici les ϵ_i dépendent clairement des $\eta(x_i)$.

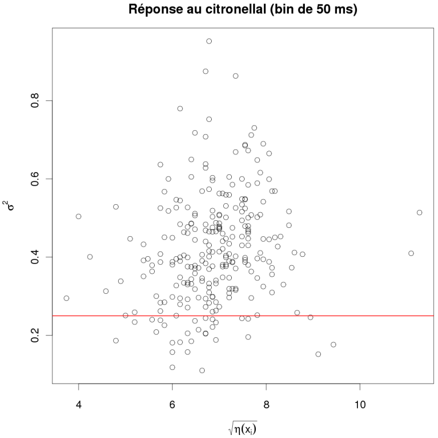
Stabilisation de la variance

- On se souvient que les décharges spontanées correspondent approximativement à des processus de Poisson (homogènes).
- On peut alors penser que les réponses aux odeurs correspondent approximativement à des processus de Poisson inhomogène.
- On se souvient aussi qu'une distribution de Poisson de paramètre ≥ 15 est approximativement gaussienne avec une variance égale à la moyenne.
- On se souvient peut-être que la racine carrée d'une variable aléatoire de Poisson (de paramètre ≥ 15) est approximativement gaussienne de variance 0.25 quelque soit la valeur du paramètre.

Stabilisation de la variance

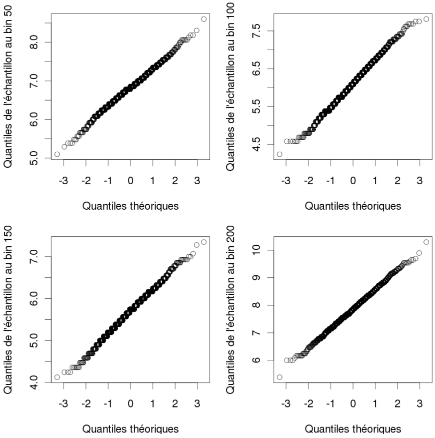
```
> citronSC.hat <- sqrt(apply(citronC,1,sum))
> citronSC.hat.var <- apply(sqrt(citronC.boot),
+ 1,var)
```

Stabilisation de la variance



Ici les ϵ_i dépendent nettement moins des $\sqrt{\eta(x_i)}$.

Distribution des $\sqrt{\eta(x_i)}$



Graphes quantiles-quantiles des $\sqrt{\eta(x_i)}$ obtenus par bootstrap (réponse au citronellal, bin de 50 ms).

Smoothing Splines 1

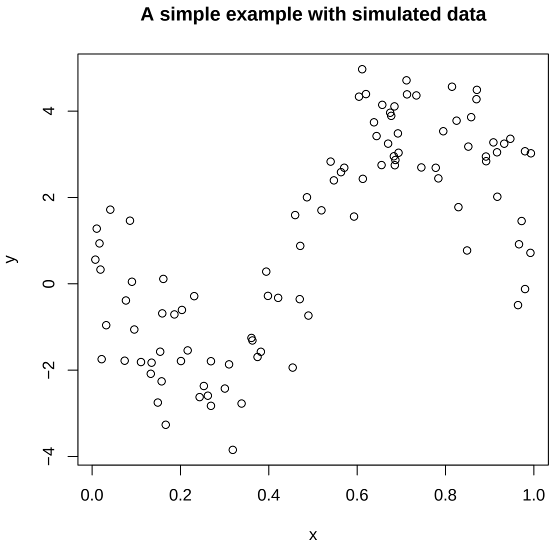
Given data :

$$Y_i = \eta(x_i) + \epsilon_i, \quad i = 1, \ldots, n$$

where $x_i \in [0,1]$ and $\epsilon_i \sim \mathrm{N}(0, \sigma^2)$, we want to find η_λ minimizing :

$$\frac{1}{n} \sum_{i=1}^{n} (Y_i - \eta_\lambda(x_i))^2 + \lambda \int_0^1 \big(\frac{d^2 \eta_\lambda}{dx^2}\big)^2 dx$$

Smoothing Splines 2



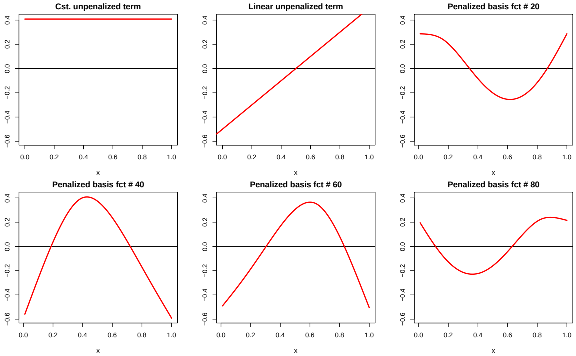
Smoothing Splines 3

It can be shown (but it's not exactly trivial) that, for a given λ, the solution of the functional minimization problem can be expressed on a finite basis :

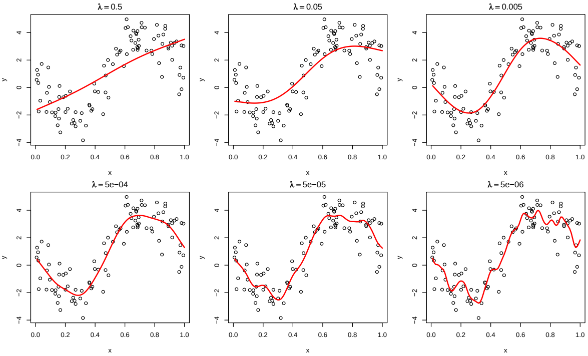
$$\eta_\lambda(x) = \sum_{\nu=0}^{m-1} d_\nu \, \phi_\nu(x) + \sum_{i=1}^{n} c_i \, R_1(x_i, x)$$

where the functions, $\phi_\nu()$, and $R_1(x_i,)$, are known.

Smoothing Splines 4



Smoothing Splines 5 : What about λ ?



Smoothing Splines 6 : Cross-Validation

Ideally we would like λ such that :

$$\frac{1}{n}\sum_{i=1}^{n}(\eta_\lambda(x_i) - \eta(x_i))^2$$

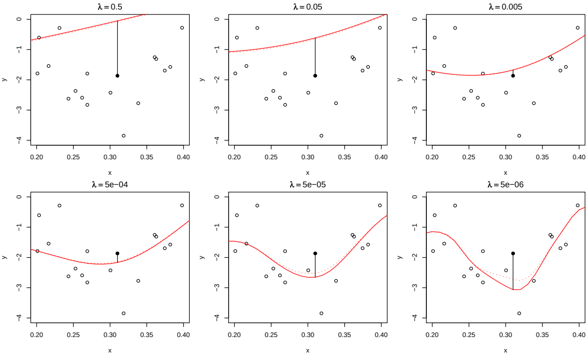
is minimized... but we don't know the true η. So we choose λ minimizing :

$$V_0(\lambda) = \frac{1}{n}\sum_{i=1}^{n}(\eta_\lambda^{[i]}(x_i) - Y_i)^2$$

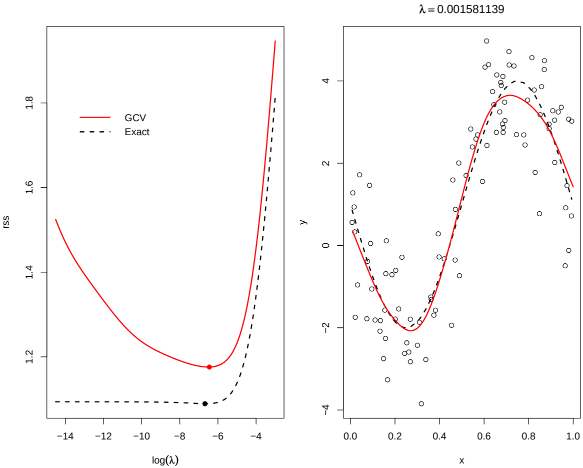
where $\eta_\lambda^{[k]}$ is the minimizer of the "delete-one" functional :

$$\frac{1}{n}\sum_{i \neq k}(Y_i - \eta_\lambda(x_i))^2 + \lambda \int_0^1 \big(\frac{d^2\eta_\lambda}{dx^2}\big)^2 dx$$

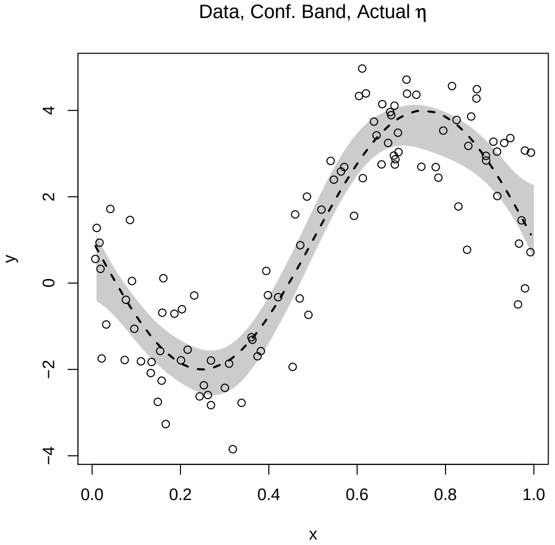
Smoothing Splines 7



Smoothing Splines 8



Smoothing Splines 9 : The Theory (worked out by Grace Wahba) also gives us confidence bands

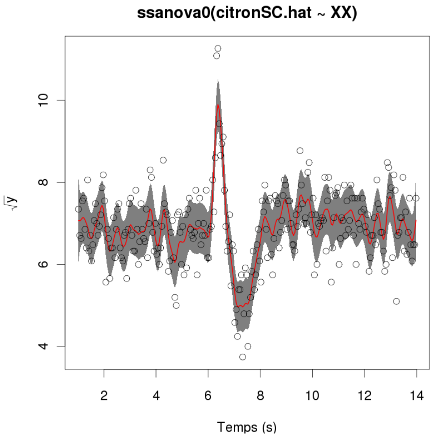


General Smoothing Spline

- La méthodologie des splines de lissage avec paramètre de lissage estimé par validation croisée est mise en oeuvre dans le paquet gss (General Smoothing Spline) développé par Chong Gu (Université de Purdue).
- Nous utilisons d'abord la fonction `ssanova0` :

```
> XX <- seq(1.025,13.975,0.05)
> GF1 <- ssanova0(citronSC.hat ~ XX)
```

Ajustement avec ssanova0



Les intervalles de confiance sont obtenus avec :

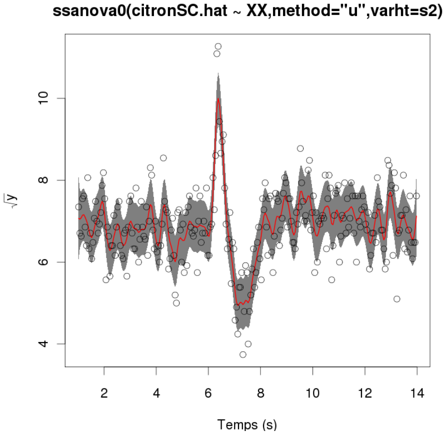
```
> estGF1 <- predict(GF1,
+ newdata=data.frame(XX=XX),se=TRUE)
```

Variation autour de ssanova0

- Avec `ssanova0` il est possible de fixer la variance des ϵ_i au moment de la validation croisée :

```
> s2 <- mean(citronSC.hat.var)
> GF2 <- ssanova0(citronSC.hat ~ XX,
+ method="u",varht=s2)
```

Variation autour de ssanova0



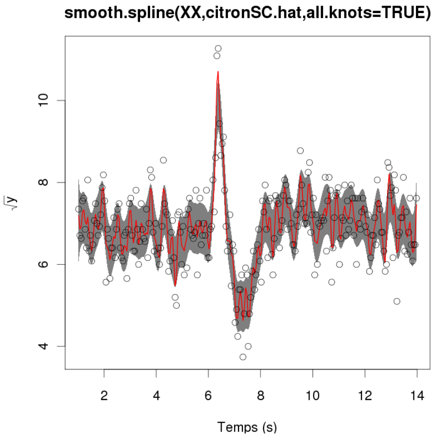
Il n'y a pas grande différence.

Fonction smooth.spline

- Le paquet `stats` contient la fonction `smooth.spline` similaire à `ssanova0` mais “moins puissante”.
- `smooth.spline` ne génère pas d'intervalles de confiance.

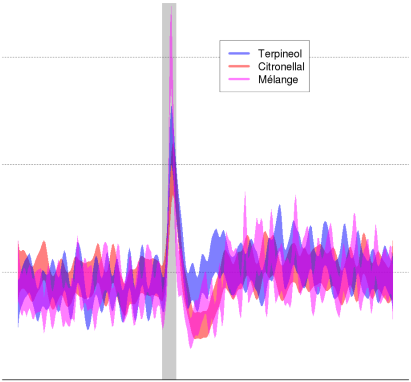
```
> SF <- smooth.spline(XX,citronSC.hat,
+ all.knots=TRUE)
```

Fonction smooth.spline



Les intervalles de confiance sont ceux de `ssanov0`.

Retour aux données



La suite

- On peut “jouer” avec d’autres arguments de `smooth.spline`, mais globalement, l’essentiel a été couvert.
- `ssanova0` peut-être utilisée avec plusieurs prédicateurs.
- Le paquet `gss` met en oeuvre des tests pour évaluer le rôle de différents prédicateurs.
- Avec `gss`, les ϵ_i peuvent avoir une distribution binomiale, Poisson, inverse gaussienne, etc. Il faut alors utiliser la fonction `gssanova0`.