Spike Train Measurement and Sorting

Christophe Pouzat

CNRS UMR 8118 and Paris-Descartes University
Paris, France
christophe.pouzat@gmail.com

May 7 2010

A brief introduction to a biological problem

Neurophysiologists are trying to record many neurons at once because:

- They can collect more data per experiment.
- They have reasons to think that neuronal information processing might involve synchronization among neurons, an hypothesis dubbed binding by synchronization in the field.

What is binding?

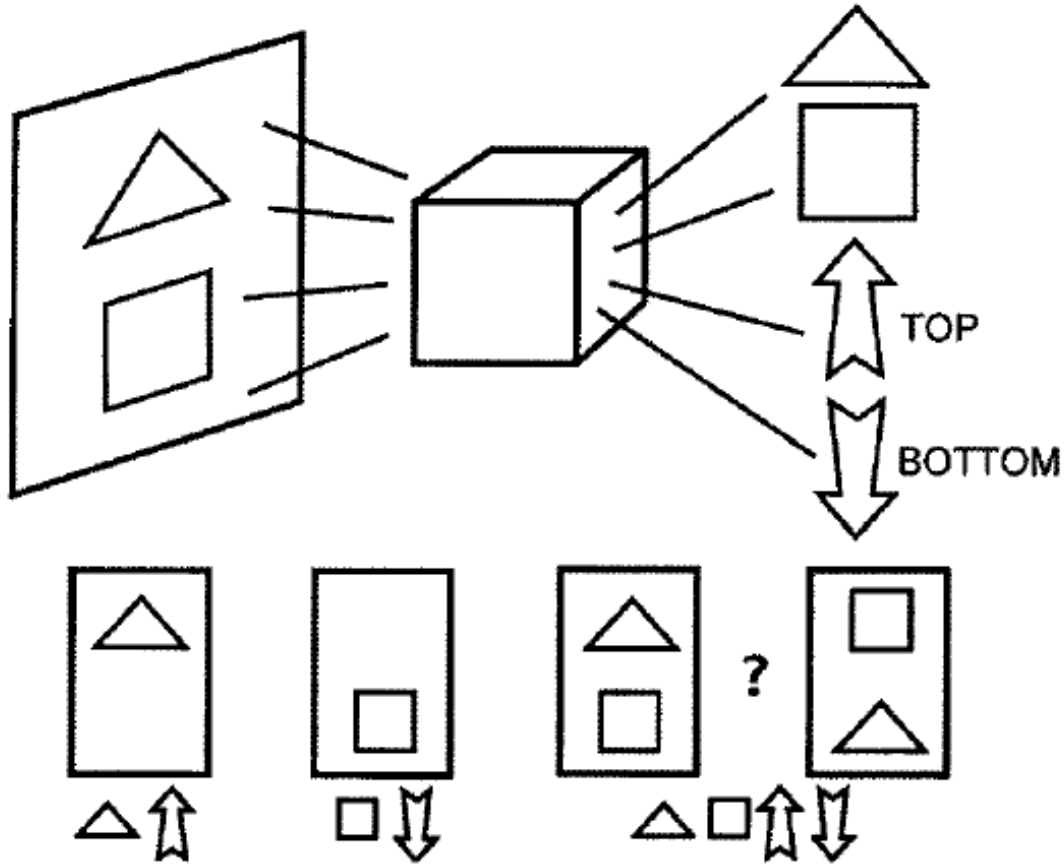


Figure 1. Rosenblatt's Example

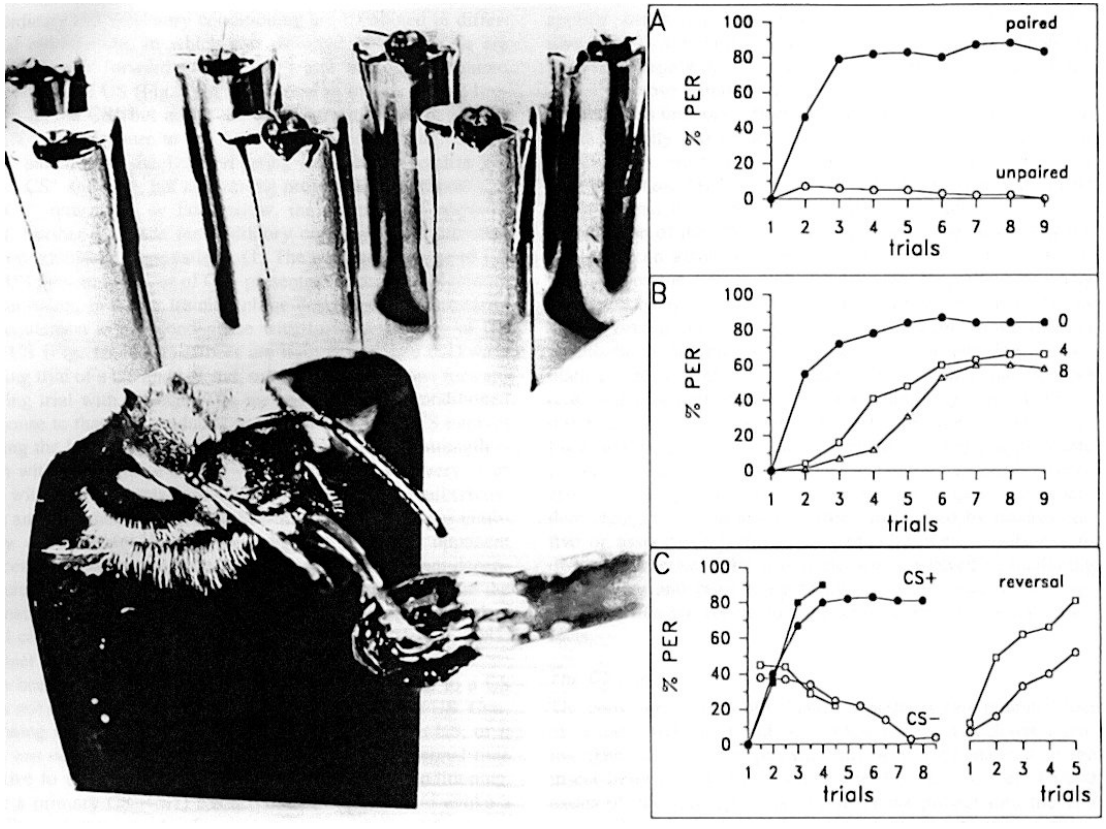
A toy example of a 4 neurons system. One neuron detects triangles, one detects squares, an other one responds to objects in the upper visual field, while the last one detects objects in the lower visual field.

The classical example shown in binding talks

Experimental problems of binding studies

- We must be sure that the animal recognizes the complex stimulus. The animal must therefore be conditioned.
- Working with vertebrates implies then the use of cats or monkeys.
- We then end up looking for synchronized neurons in networks made of 10^7 cells after spending months conditioning the animal... It is a bit like looking for a needle in a hay stack.
- *In vivo* recordings in vertebrates are moreover unstable: the heart must beat which expands the arteries. The tissue is therefore necessarily moving around the recording electrodes.

An alternative approach: proboscis extension and olfactory conditioning in insects



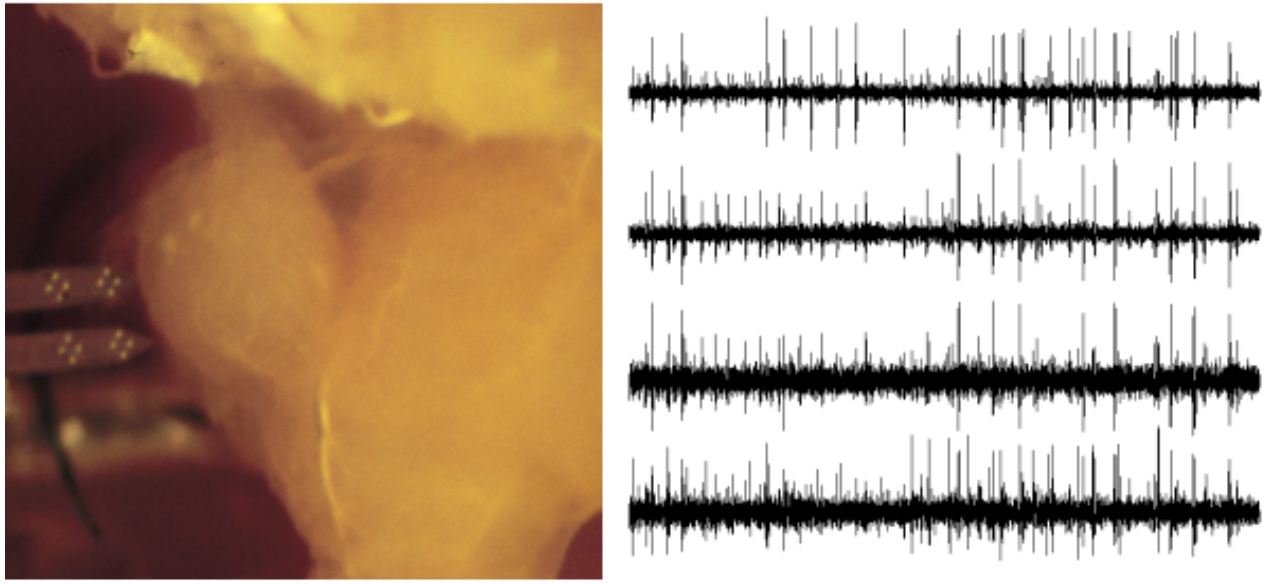
Learning curves obtained from honey bees, *Apis mellifera*, by Hammer and Menzel [1995].

What are we trying to do?

- An elegant series of experiments by Hammer and Menzel [1998] suggests that part of the conditioning induced neuronal modifications occur in the first olfactory relay of the insect: the antennal lobe.
- The (simple) idea is then to record neuronal responses in the antennal lobe to mixtures of pure compounds like citral and octanol in two groups of insects: one conditioned to recognize the mixture, the other one not.
- To demonstrate synchronization in one group and not in the other we must record several neurons at once for a long time.

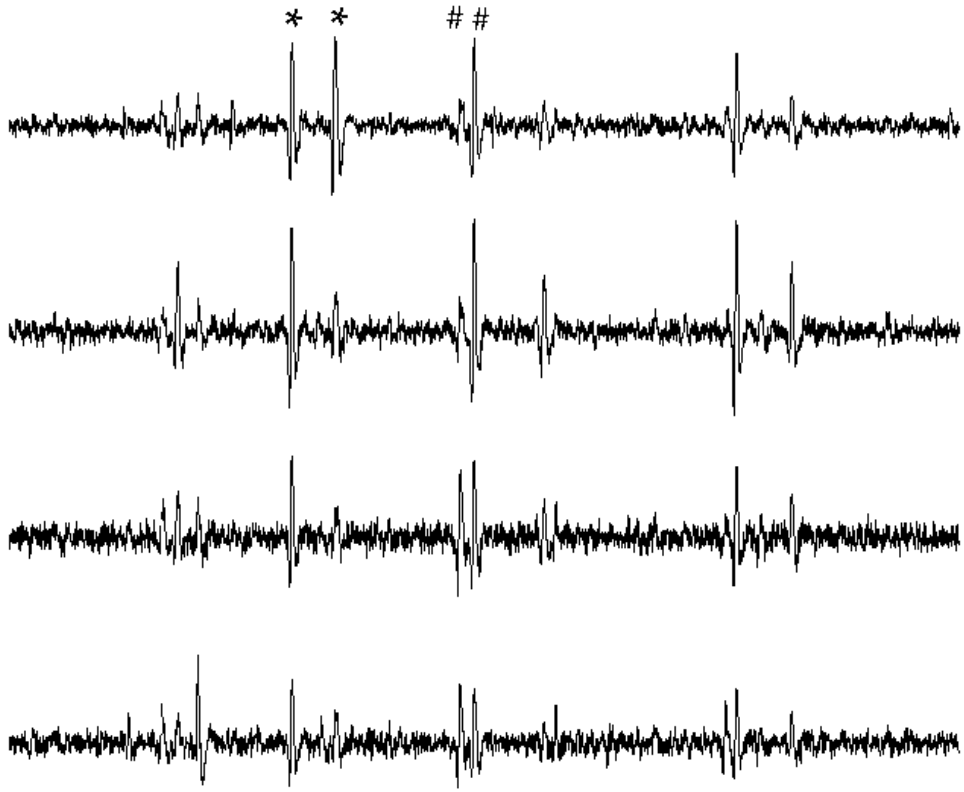
Multi-electrodes *in vivo* recordings in insects

"From the outside" the neuronal activity appears as brief electrical impulses: the action potentials or spikes.



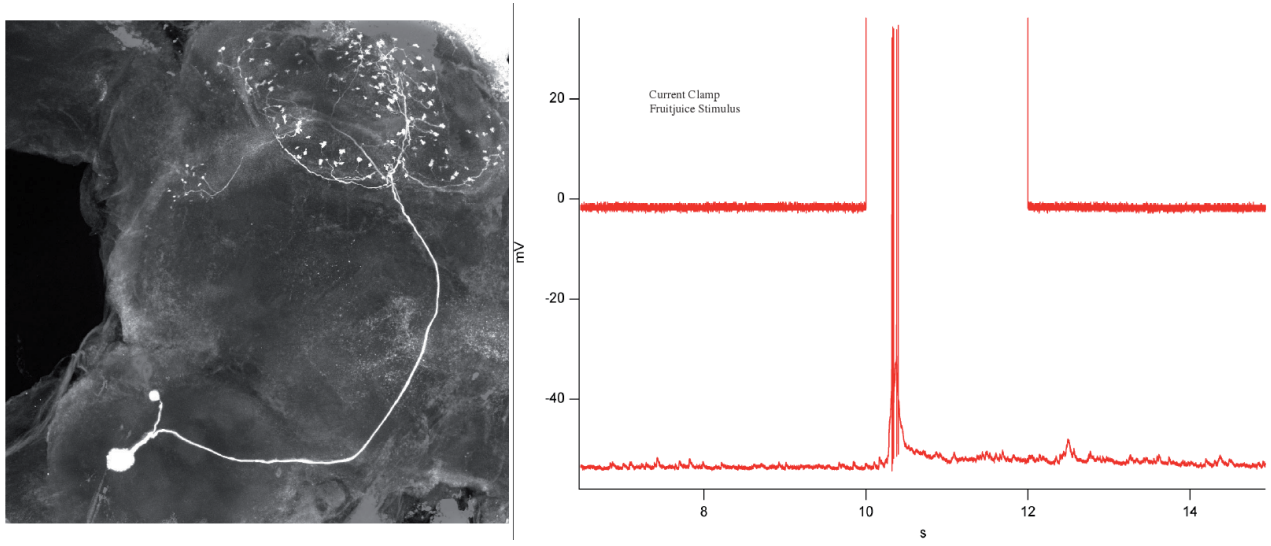
Left, the brain and the recording probe with 16 electrodes (bright spots). Width of one probe shank: 80 μm. Right, 1 sec of raw data from 4 electrodes. The local extrema are the action potentials.

Why are tetrodes used?



The last 200 ms of the previous figure. With the upper recording site only it would be difficult to properly classify the two first large spikes (**). With the lower site only it would be difficult to properly classify the two spikes labeled by ##.

Other experimental techniques can also be used



A single neuron patch-clamp recording coupled to calcium imaging. Data from Moritz Paehler and Peter Kloppenburg (Cologne University).

Data preprocessing: Spike sorting

To exploit our recordings we must first:

- Find out how many neurons are recorded.
- For each neuron estimate some features like the spike waveform, the discharge statistics, etc.
- For each detected event find the probability with which each neuron could have generated it.
- Find an automatic method to answer these questions.

Software issues

Spike sorting like any data analysis problem can be made tremendously easier by a "proper" software choice. We have chosen to work with `R` because:

- `R` is an open-source software running on basically any computer / OS combination available.
- It is actively maintained.
- It is an elegant programming language derived from `Lisp`.
- It makes trivial parallelization really trivial.
- It is easy to interface with `fortran`, `C` or `C++` libraries.

A similar problem

- Think of a room with many seating people who are talking to each other using a language we do not know.
- Assume that microphones were placed in the room and that their recordings are given to us.
- Our task is to isolate the discourse of each person.

We have therefore a situation like...

To fulfill our task we could make use of the following features:

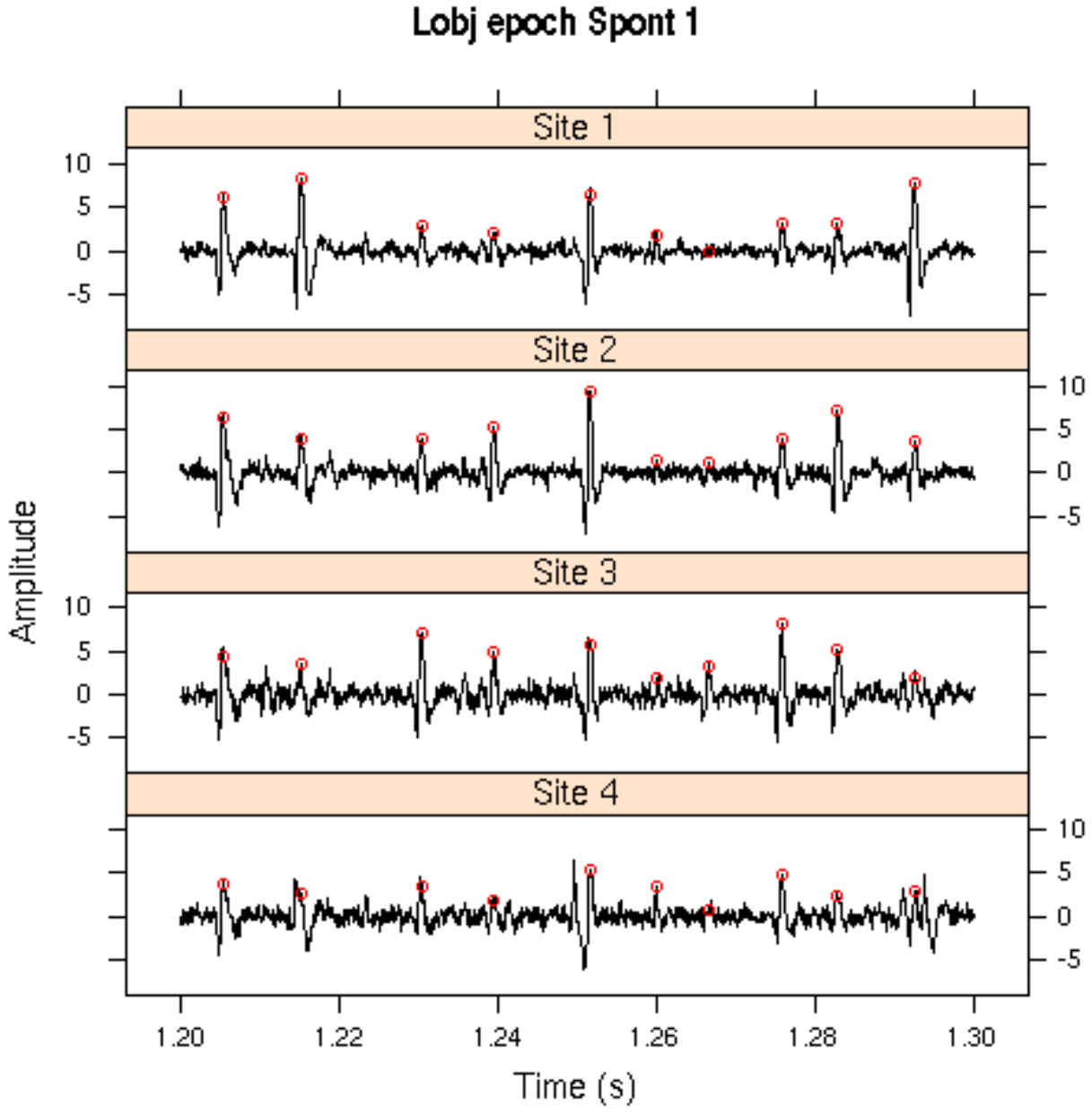
- Some people have a low pitch voice while other have a high pitch one.
- Some people speak loudly while other do not.
- One person can be close to one microphone and far from another such that its talk is simultaneously recorded by the two with different amplitudes.
- Some people speak all the time while other just utter a comment here and there, that is, the discourse statistics changes from person to person.

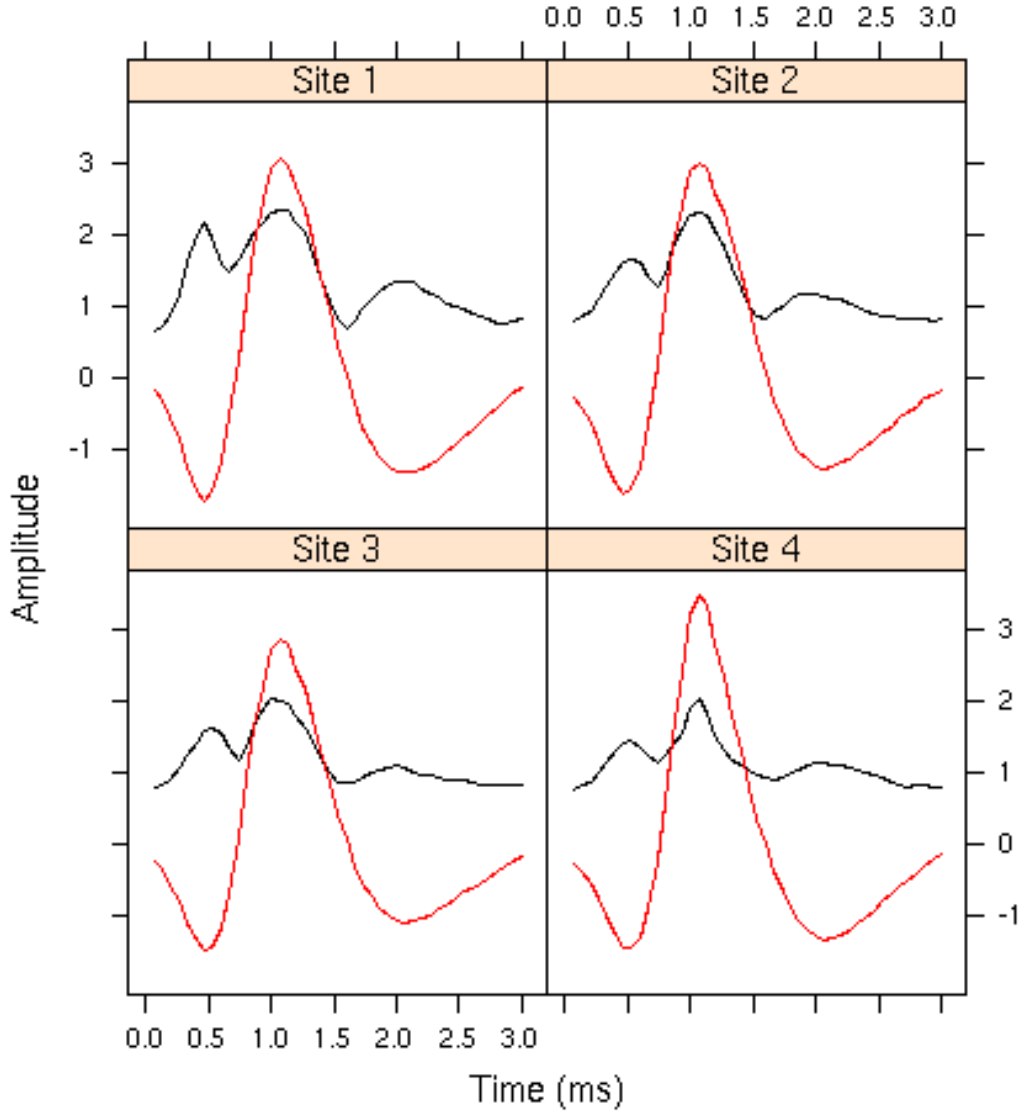
Spike sorting as a set of standard statistical problems

Efficient spike sorting requires:

1. Events detection followed by events space dimension reduction.
2. A clustering stage. This can be partially or fully automatized depending on the data.
3. Events classification.

Detection illustration



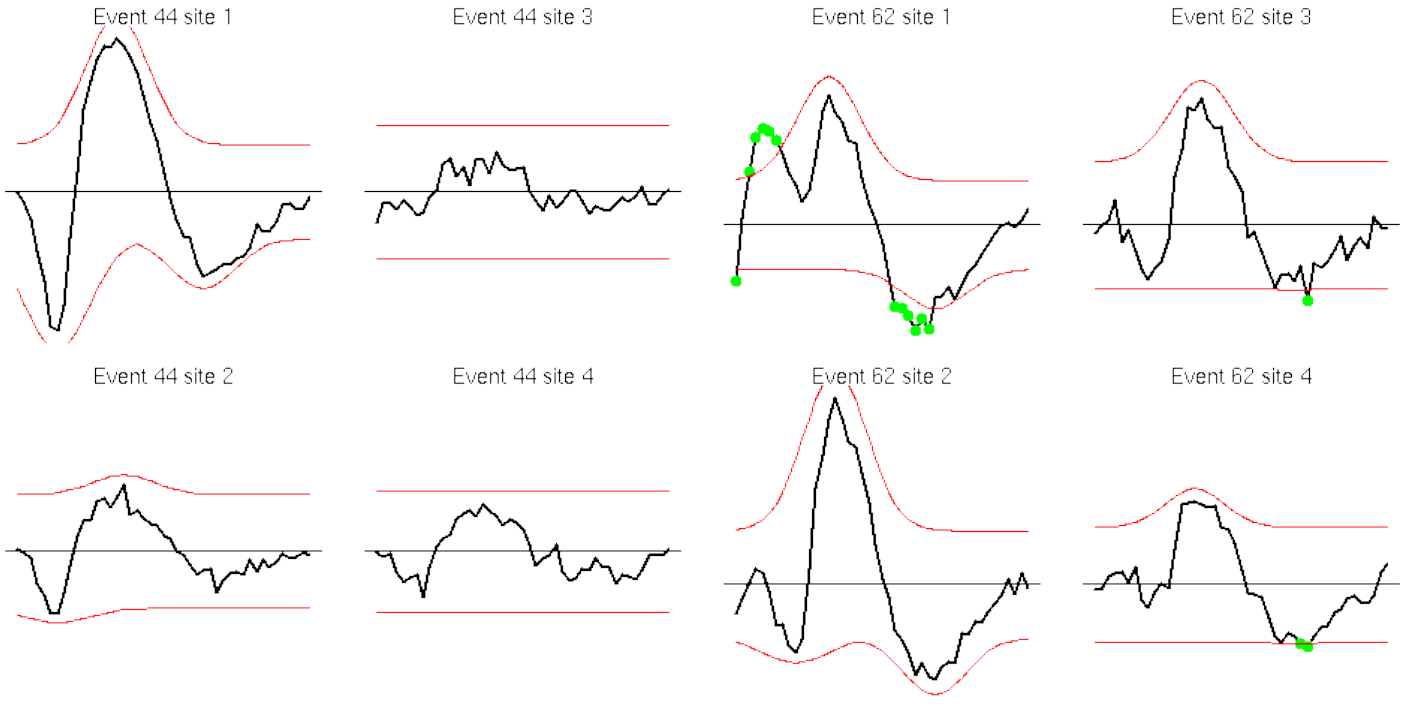


The mean event (red) and its standard deviation (black). Sample size: 1421 events detected during 30 s.

“Clean” events

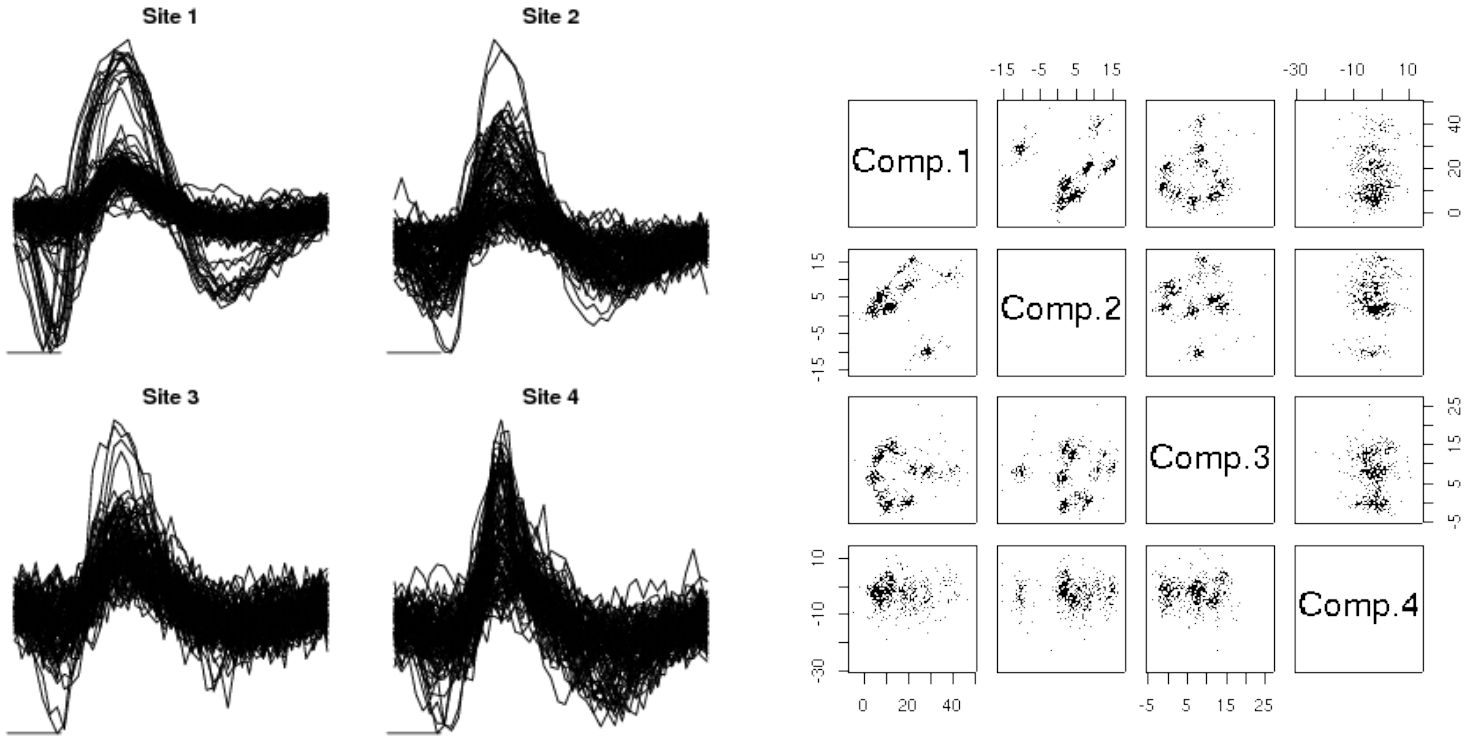
- When many neurons are active in the data set superposed events are likely to occur.
- Such events are due to the firing of 2 different neurons within one of our event defining window.
- Ideally we would like to identify and classify superposed events as such.
- We proceed in 3 steps:
  - A “clean” sample made of non-superposed events is first define.
  - A model of clean events is estimated on this sample.
  - The initial sample is classified and superpositions are identified.

Clean events selection illustration



Dimension Reduction

- The events making the sample you have seen are defined on 3 ms long windows with data sampled at 15 kHz.
- This implies that $4 \times 15 \times 10^3 \times 3 \times 10^{-3} = 180$ voltage measurements are used to describe our events.
- In other words our sample space is $\mathbb{R}^{180}$.
- Since it is hard to visualize objects and dangerous to estimate probability densities in such a space, we usually reduce the dimension of our sample space.
- We usually use a principal component analysis to this end. We keep components until the projection of the data on the plane defined by the last two appears featureless.



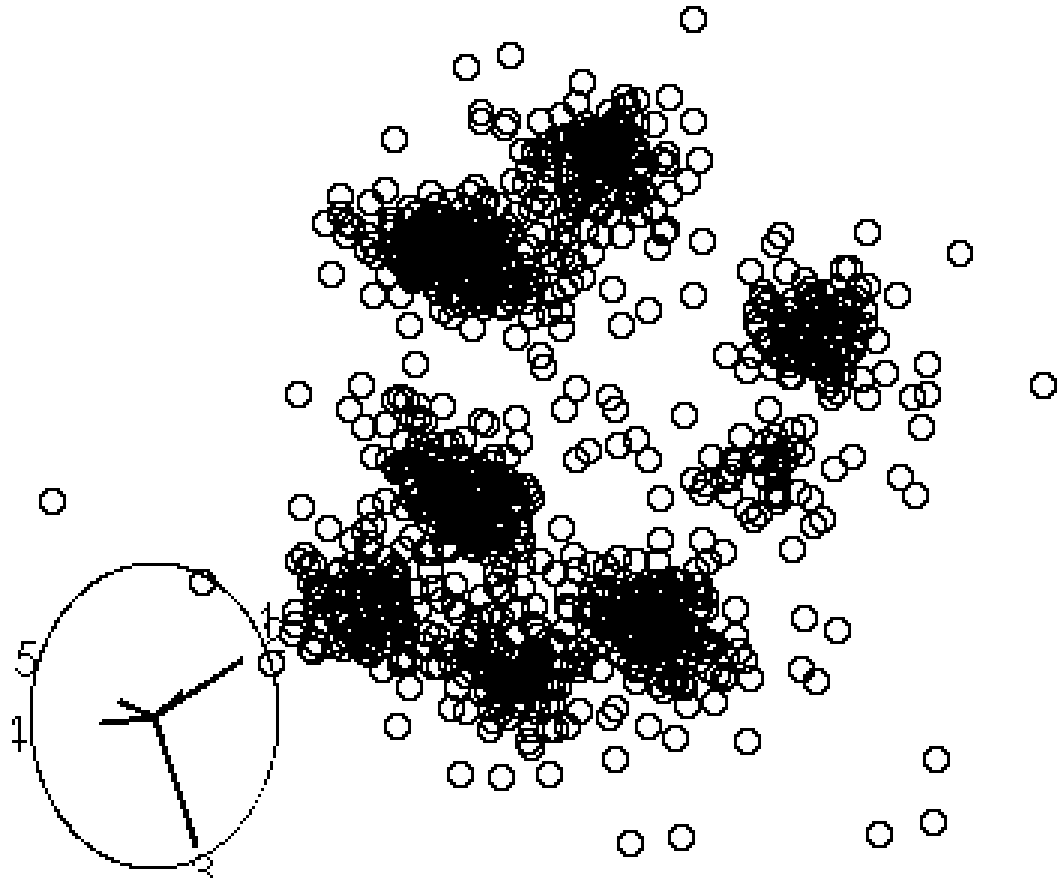
Left, 100 spikes (scale bar: 0.5 ms). Right, 1000 spikes projected on the subspace defined by the first 4 principal components.

High-dimensional data visualization

Before using clustering software on our data, looking at them with a dynamic visualization software can be enlightening.

- `GGobi` is an open-source software also running on Linux, Windows, Mac OS.
- It is actively maintained by Debby Swaine, Di Cook, Duncan Temple Lang and Andreas Buja.

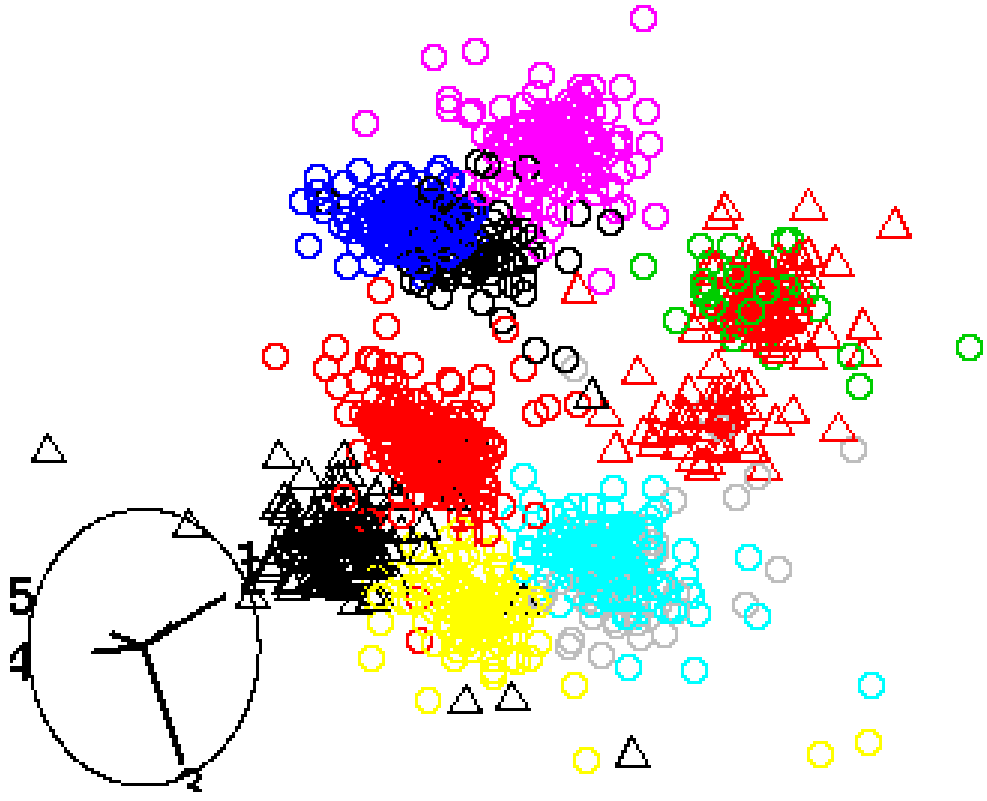
The minimal number of clusters present in the data is usually best estimated with the dynamic display supplemented by “projection pursuit”.



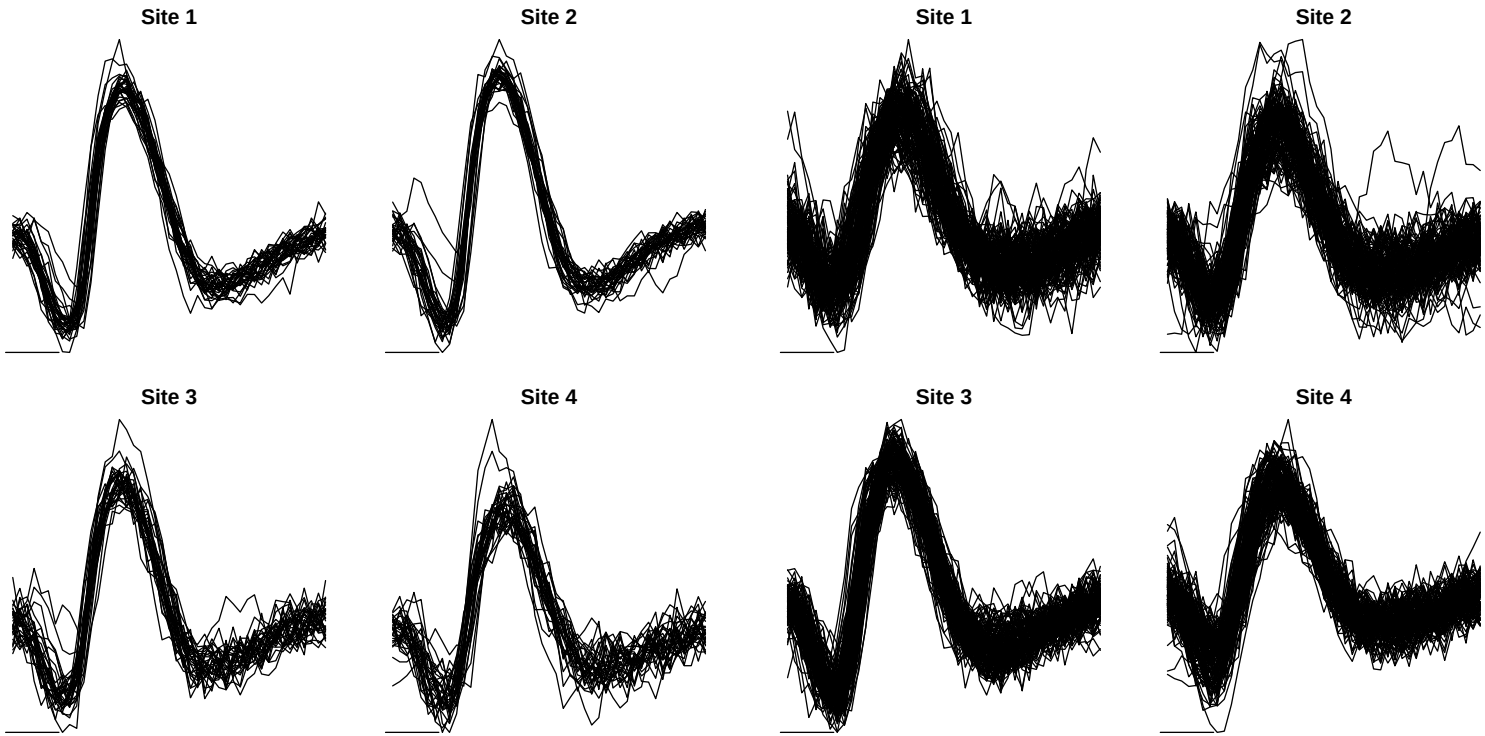
Semi-automatic and automatic clustering

- We perform semi-automatic clustering with `k-means` or `bagged clustering`.
- With these methods the user has to decide what is the "correct" number of clusters.
- Automatic clustering is performed by fitting a Gaussian mixture model to the data using `mclust` or `MixMod`.
- These two software provide criteria like the `BIC` (Bayesian Information Criterion) or the `AIC` (An Information Criterion, introduced by Akaike) to select the number of clusters.
- In practice the `BIC` works best but gives only an indication.

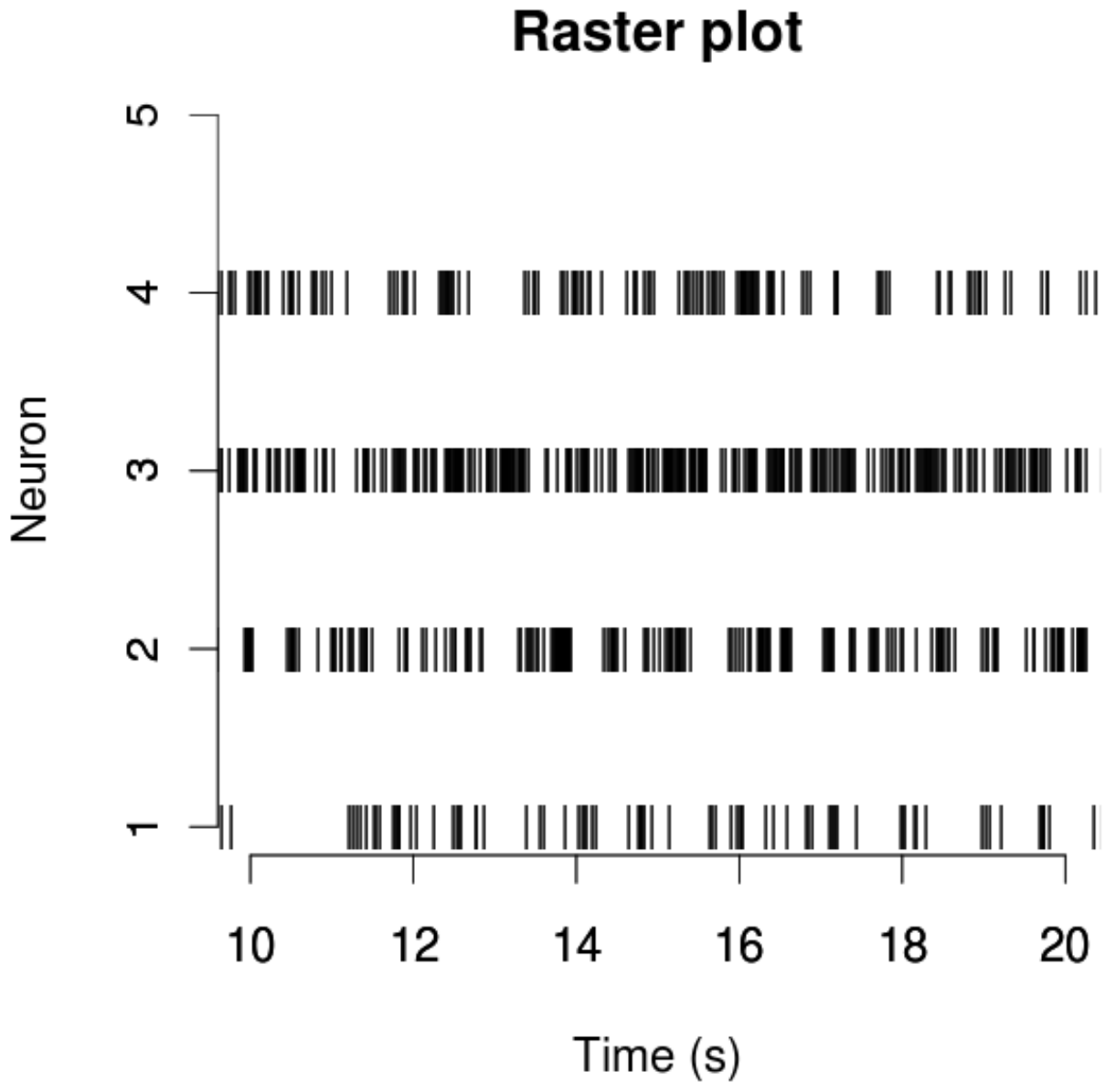
Clustering results on the previous projection



The action potentials of neuron 3 (left) and 10 (right)



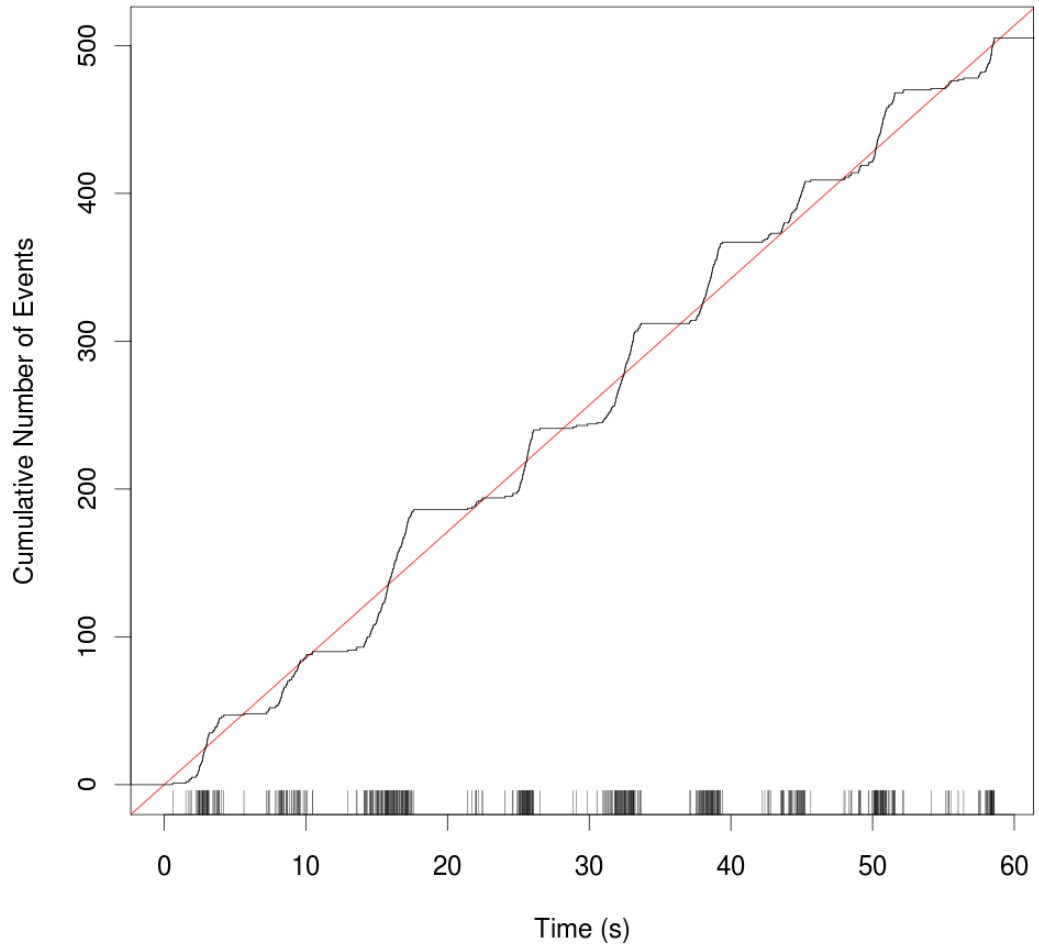
Spike trains



Studying spike trains *per se*

- A central working hypothesis of systems neuroscience is that action potential or spike occurrence times, as opposed to spike waveforms, are the sole information carrier between brain regions [Adrian and Zotterman, 1926a,b].
- This hypothesis legitimates and leads to the study of spike trains *per se*.
- It also encourages the development of models whose goal is to predict the probability of occurrence of a spike at a given time, without necessarily considering the biophysical spike generation mechanisms.

Spike trains are not Poisson processes



The “raw data” of one bursty neuron of the cockroach antennal lobe.
1 minute of spontaneous activity.

Homogenous Poisson Process

A homogenous Poisson process (HPP) has the following properties:

1. The process is homogenous (or stationary), that is, the probability of observing n events in $(t, t+\Delta t)$ depends only on Δt and not on t. If N is the random variable describing the number of events observed during Δt, we have:

$$\text{Prob}\{N = n\} = p_n(\Delta t)$$

2. The process is orderly, that is:

$$\lim_{\Delta t \to 0} \frac{\text{Prob}\{N > 1\}}{\text{Prob}\{N = 1\}} = 0$$

   There is at most one event at a time.

3. The process is without memory, that is, if T_i is the random variable corresponding to the interval between events i and $i+1$ then:

$$\text{Prob}\{T_i > t + s \mid T_i > s\} = \text{Prob}\{T_i > t\}, \quad \forall i.$$

HPP properties

We can show Pelat [1996] that a HPP has the following properties:

- There exists a $\nu > 0$ such that:

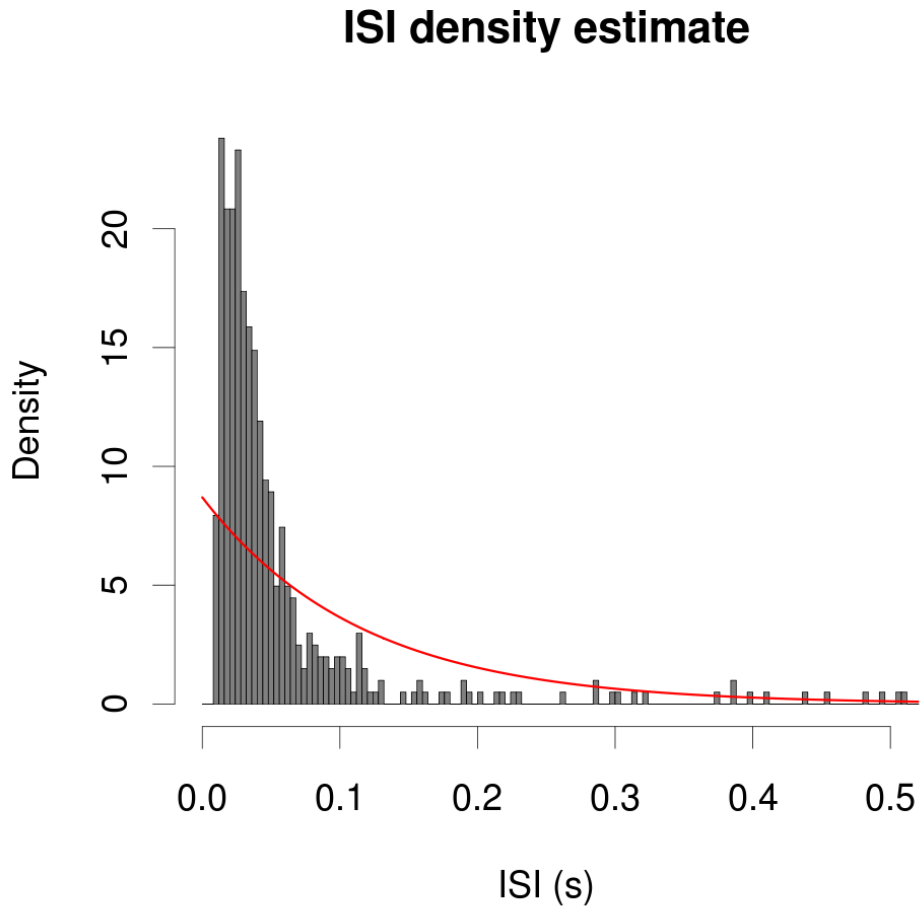
$$p(T_i = t) = \nu \exp(-\nu t), \quad t \geq 0,$$

  where $p(T_i = t)$ stands for the probability density function (pdf) of T_i.

- The number n of events observed in an interval $(t, t + \Delta t)$ is the realization of a Poisson distribution of parameter $\nu \Delta t$:

$$\text{Prob}\{N = n \text{ in } (t, t + \Delta t)\} = \frac{(\nu \Delta t)^n}{n!} \exp(-\nu \Delta t)$$

Spike trains are not Poisson processes (again)



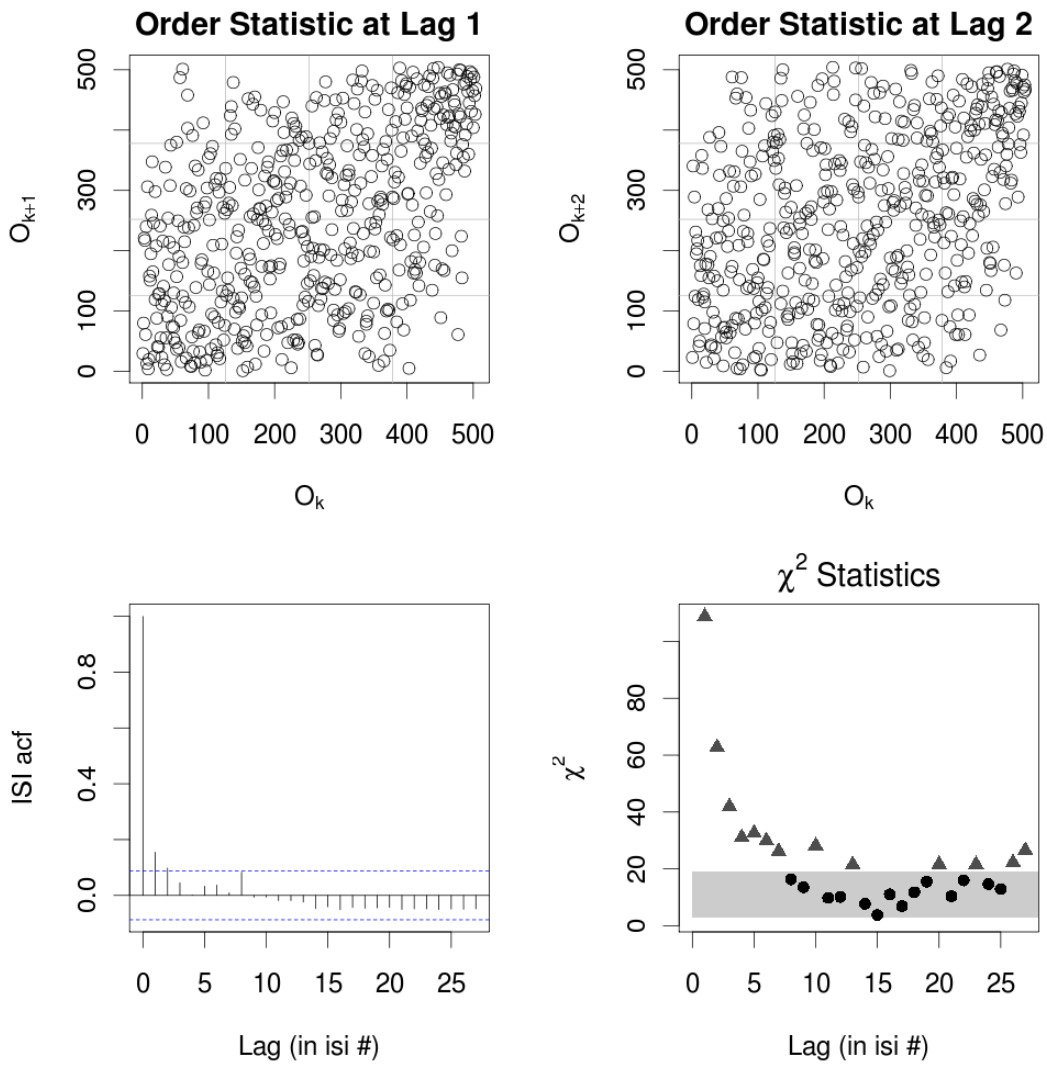
Density estimate (gray) and Poisson process fit (red) for the inter spike intervals (ISIs) of the previous train. The largest ISI was 3.8 s.

Renewal Processes

When a Poisson process does not apply, the next "simplest" process we can consider is the renewal process [Perkel et al., 1967] which can be defined as:

- The ISIs of a renewal process are identically and independently distributed (IID).
- This type of process is used to describe occurrence times of failures in "machines" like light bulbs, hard drives, etc.

Spike trains are rarely renewal processes



Some "renewal tests" applied to the previous data. See Pouzat and Chaffiol [2009a] for details.

A counting process formalism

Probabilists and Statisticians working on series of events whose only (or most prominent) feature is there occurrence time (car accidents, earthquakes) use a formalism based on the following three quantities [Brillinger, 1988].

- Counting Process: For points $\{t_j\}$ randomly scattered along a line, the counting process $N(t)$ gives the number of points observed in the interval $(0, t]$:

$$N(t) = \sharp\{t_j \text{ with } 0 < t_j \leq t\}$$

where $\sharp$ stands for the cardinality (number of elements) of a set.

- History: The history, $\mathcal{H}_t$, consists of the variates determined up to and including time t that are necessary to describe the evolution of the counting process.
- Conditional Intensity: For the process N and history $\mathcal{H}_t$, the conditional intensity at time t is defined as:

$$\lambda(t \mid \mathcal{H}_t) = \lim_{h \downarrow 0} \frac{\text{Prob}\{\text{event} \in (t, t+h] \mid \mathcal{H}_t\}}{h}$$

for small h one has the interpretation:

$$\text{Prob}\{\text{event} \in (t, t+h] \mid \mathcal{H}_t\} \approx \lambda(t \mid \mathcal{H}_t)\, h$$

Meaning of "spike train analysis" in this talk

In this talk “spike train analysis” can be narrowly identified with conditional intensity estimation:

$$\text{spike train analysis} \equiv \text{get } \hat{\lambda}(t \mid \mathcal{H}_t)$$

where $\hat{\lambda}$ stands for an estimate of λ.

Goodness of fit tests for counting processes

- All goodness of fit tests derive from a mapping or a "time transformation" of the observed process realization.
- Namely one introduces the integrated conditional intensity :

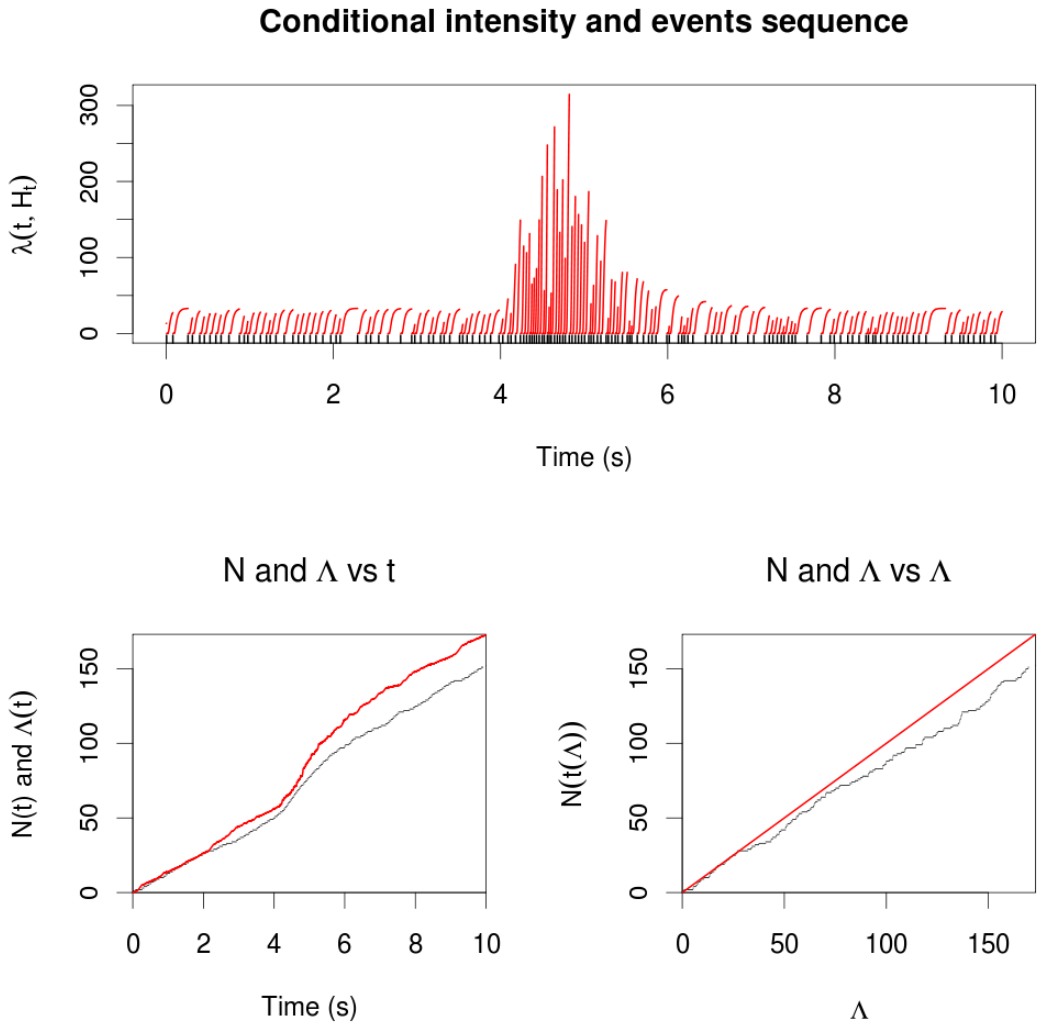
$$\Lambda(t) = \int_0^t \lambda(u \mid \mathcal{H}_u)\, du$$

- If Λ is correct it is not hard to show [Brown et al., 2002, Pouzat and Chaffiol, 2009b] that the process defined by :

$$\{t_1, \ldots, t_n\} \mapsto \{\Lambda(t_1), \ldots, \Lambda(t_n)\}$$

  is a Poisson process with rate 1.

Time transformation illustrated

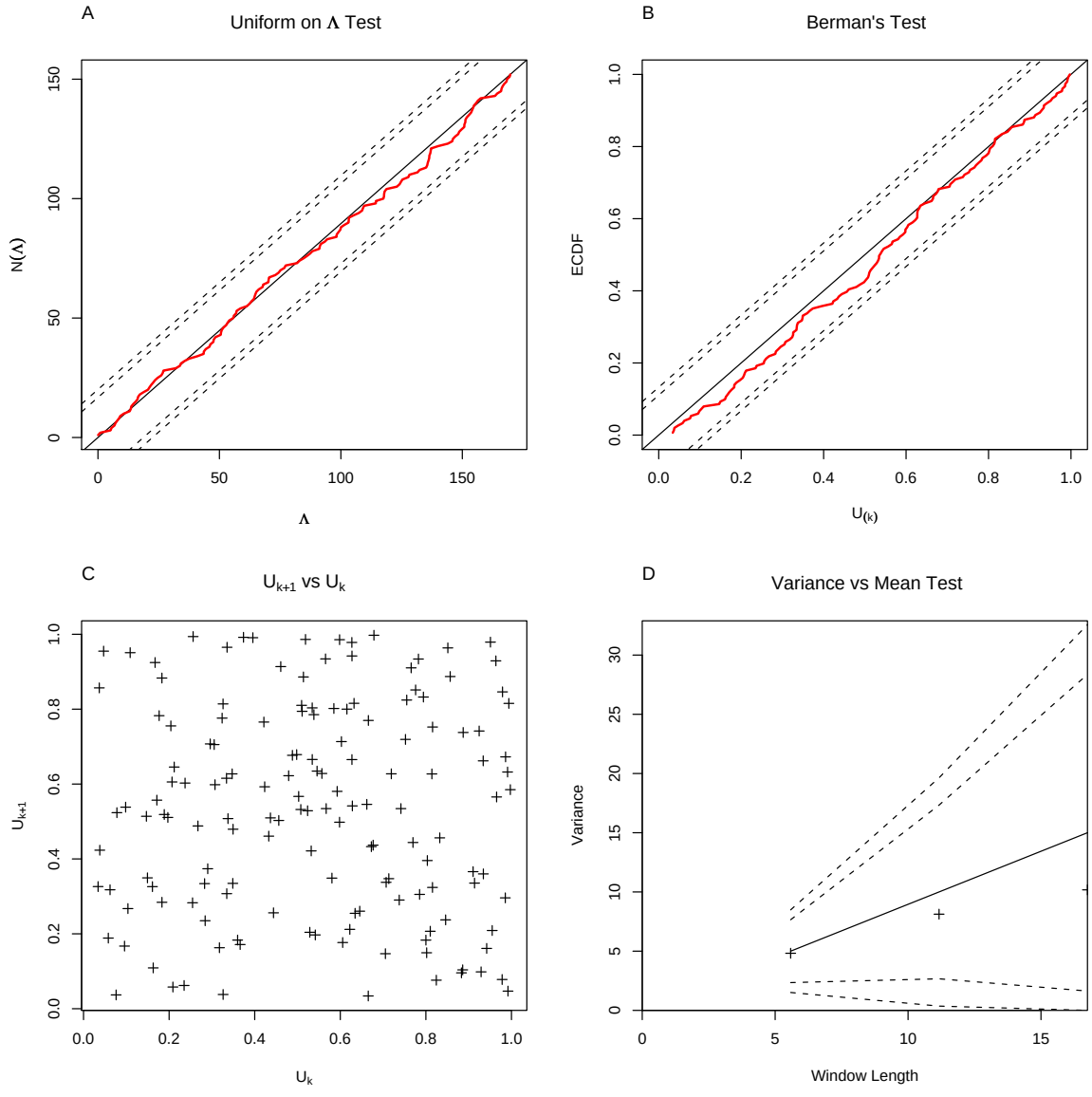


An illustration with simulated data. See Pouzat and Chaffiol [2009b] for details.

Ogata's tests

Ogata [1988] introduced several procedures testing the time transformed event sequence against the uniform Poisson hypothesis: If a homogeneous Poisson process with rate 1 is observed until its nth event, then the event times, $\{\Lambda(t_i)\}_{i=1}^{n}$, have a uniform distribution on $(0, \Lambda(t_n))$ [Cox and Lewis, 1966]. This uniformity can be tested with a Kolmogorov test.

First test displayed on the upper left



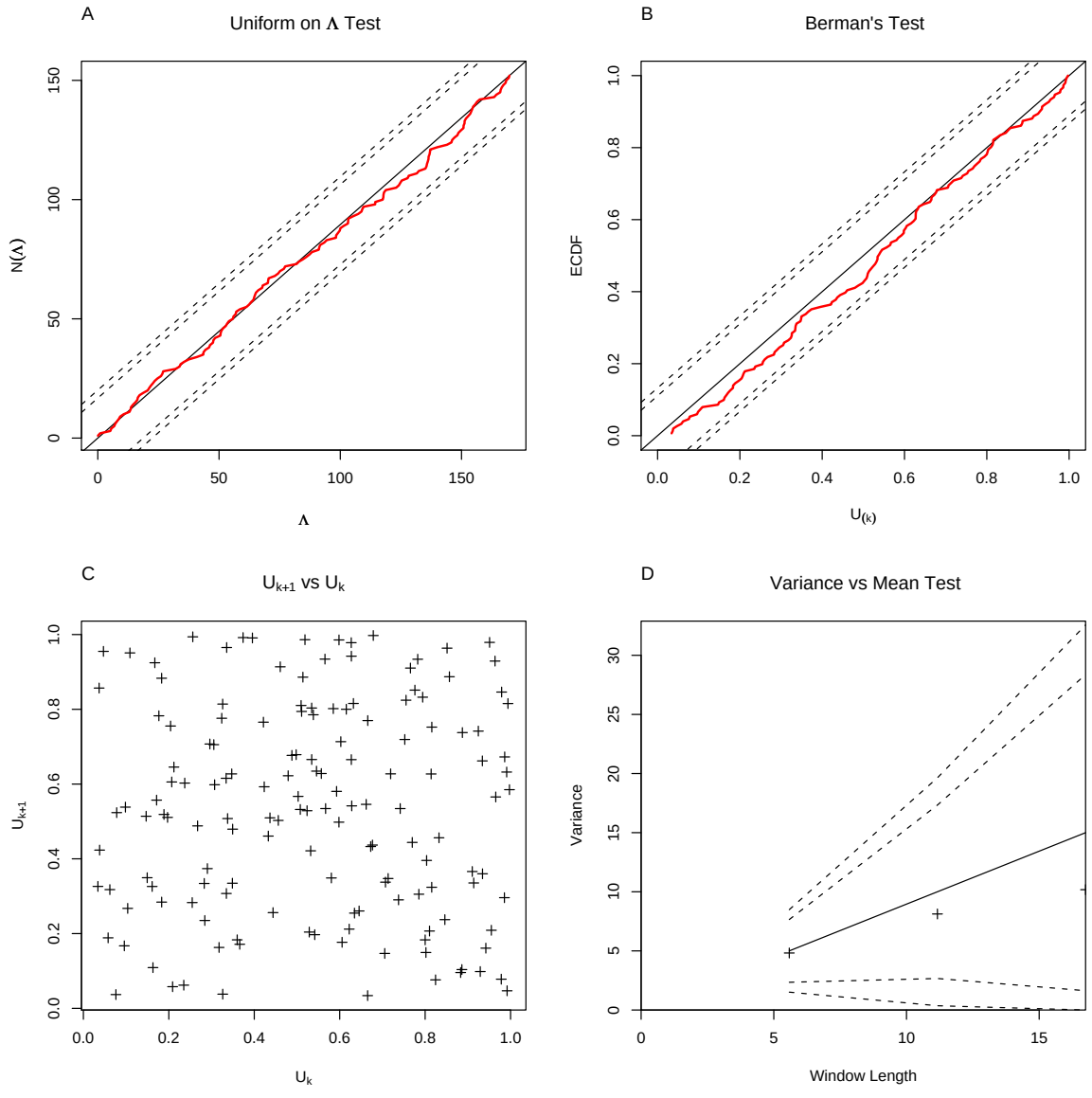
Ogata's tests on the simulated data.

The u_k defined, for $k > 1$, by:

$$u_k = 1 - \exp\left(-\left(\Lambda(t_k) - \Lambda(t_{k-1})\right)\right)$$

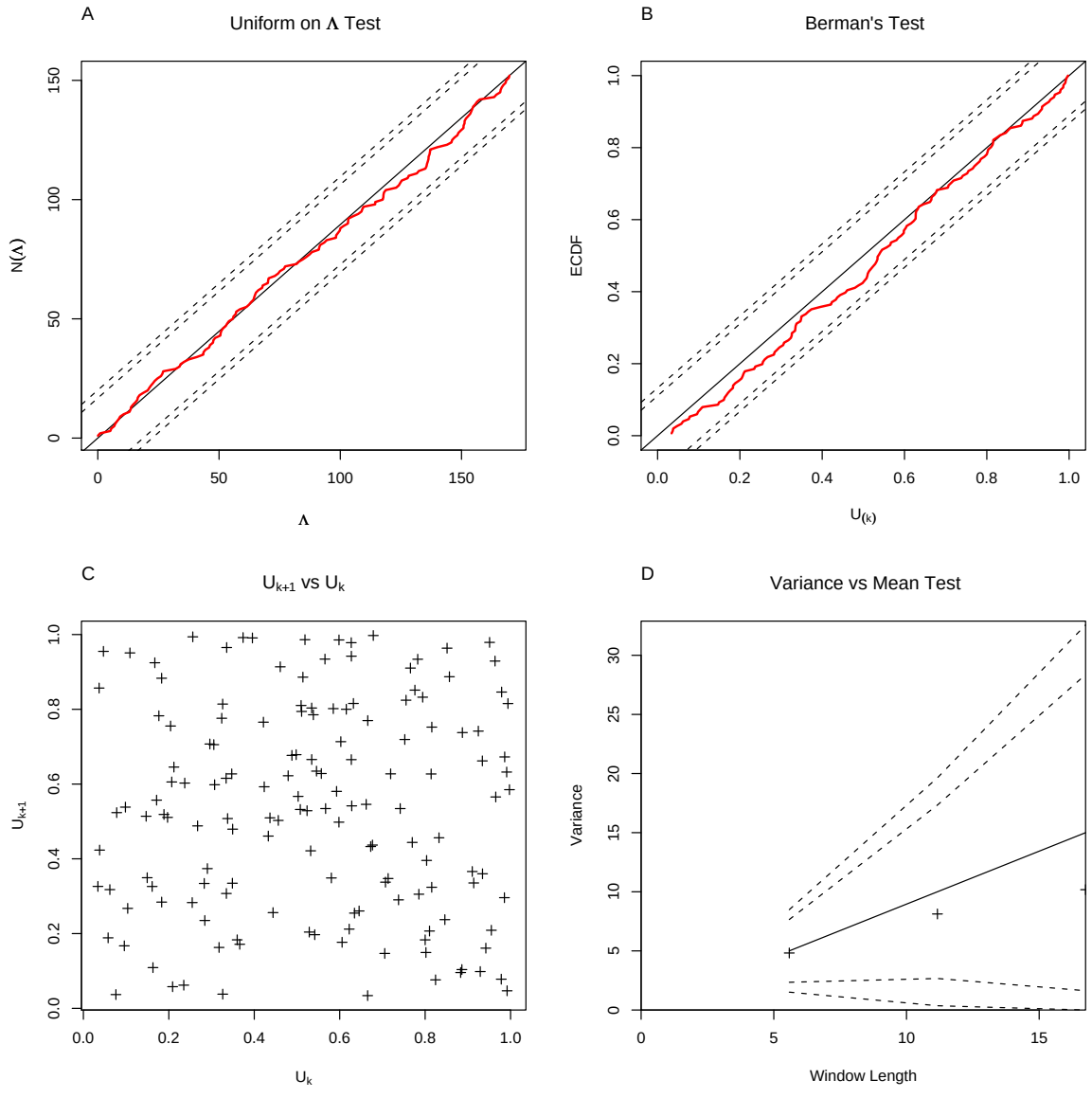
should be IID with a uniform distribution on $(0, 1)$. The empirical cumulative distribution function (ECDF) of the sorted $\{u_k\}$ can be compared to the ECDF of the null hypothesis with a Kolmogorov test. This test is attributed to Berman in [Ogata, 1988] and is the test proposed and used by [Brown et al., 2002].

Second test displayed on the upper right



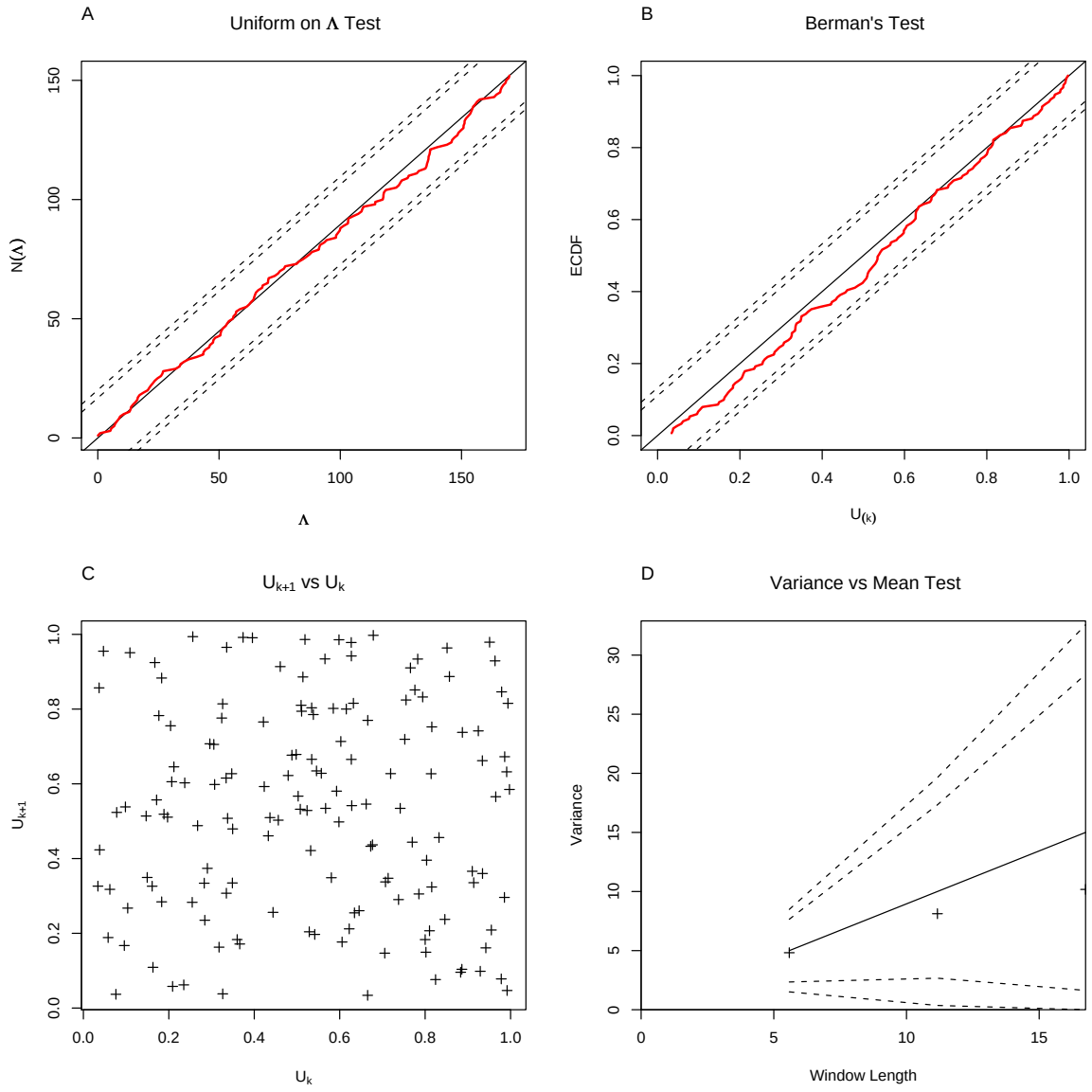
A plot of u_{k+1} vs u_k exhibiting a pattern would be inconsistent with the homogeneous Poisson process hypothesis. A shortcoming of this test is that it is only graphical and that it requires a fair number of events to be meaningful.

Third test displayed on the lower left



The last test is obtained by splitting the transformed time axis into K_w non-overlapping windows of the same size w, counting the number of events in each window and getting a mean count N_w and a variance V_w computed over the K_w windows. Using a set of increasing window sizes: $\{w_1, \ldots, w_L\}$ a graph of V_w as a function of N_w is build. If the Poisson process with rate 1 hypothesis is correct the result should fall on a straight line going through the origin with a unit slope. Pointwise confidence intervals can be obtained using the normal approximation of a Poisson distribution.

Fourth test displayed on the lower right



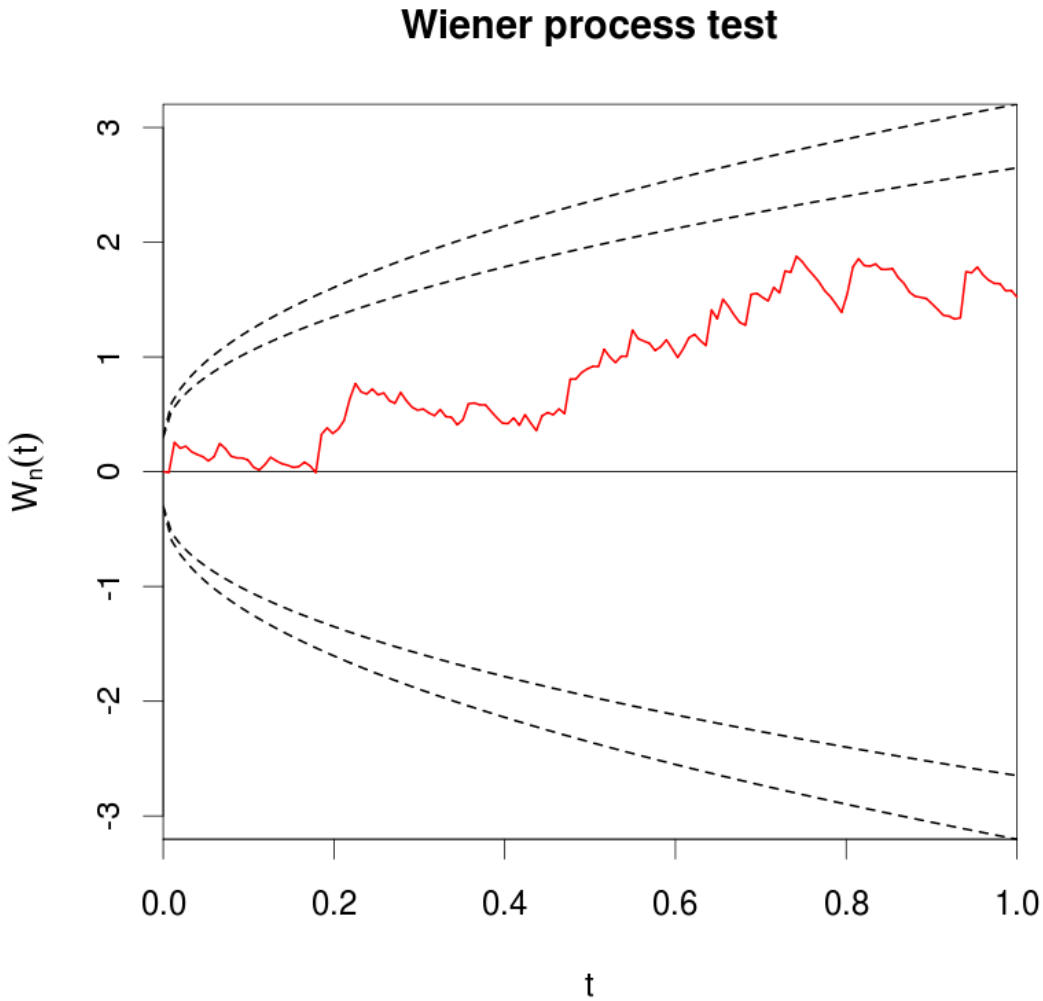
A new test based on Donsker's theorem

- We propose an additional test built as follows :

$$\begin{aligned} X_j &= \Lambda(t_{j+1}) - \Lambda(t_j) - 1 \\ S_m &= \sum_{j=1}^{m} X_j \\ W_n(t) &= S_{\lfloor nt \rfloor} / \sqrt{n} \end{aligned}$$

- Donsker's theorem [Billingsley, 1999, Durrett, 2009] implies that if Λ is correct then W_n converges weakly to a standard Wiener process.
- We therefore test if the observed W_n is within the tight confidence bands obtained by Kendall et al. [2007] for standard Wiener processes.

Illustration of the proposed test



The proposed test applied to the simulated data. The boundaries have the form: $f(x; a, b) = a + b\sqrt{x}$.

Where Are We?

- We are now in the fairly unusual situation (from the neuroscientist's viewpoint) of knowing how to show that the model we entertain is wrong without having an explicit expression for this model...
- We now need a way to find candidates for the CI: $\lambda(t \mid \mathcal{H}_t)$.

What Do We "Put" in $\mathcal{H}_t$?

- It is common to summarize the stationary discharge of a neuron by its inter-spike interval (ISI) histogram.
- If the latter histogram is not a pure decreasing mono-exponential, that implies that $\lambda(t \mid \mathcal{H}_t)$ will at least depend on the elapsed time since the last spike: $t - t_l$.
- For the real data we saw previously we also expect at least a dependence on the length of the previous inter spike interval, isi_1. We would then have:

$$\lambda(t \mid \mathcal{H}_t) = \lambda(t - t_l, isi_1)$$

What About The Functional Form?

- We haven't even started yet and we are already considering a function of at least 2 variables: $t - t_l, isi_1$. What about its functional form?
- Following Brillinger [1988] we discretize our time axis into bins of size h small enough to have at most 1 spike per bin.
- We are therefore lead to a binomial regression problem.
- For analytical and computational convenience we are going to use the logistic transform:

$$\log\left(\frac{\lambda(t - t_l, isi_1)\, h}{1 - \lambda(t - t_l, isi_1)\, h}\right) = \eta(t - t_l, isi_1)$$

The Discretized Data

	event	time	neuron	lN.1	i1
14604	0	58.412	1	0.012	0.016
14605	1	58.416	1	0.016	0.016
14606	0	58.420	1	0.004	0.016
14607	1	58.424	1	0.008	0.016
14608	0	58.428	1	0.004	0.008
14609	0	58.432	1	0.008	0.008
14610	1	58.436	1	0.012	0.008
14611	0	58.440	1	0.004	0.012

`event` is the discretized spike train, `time` is the bin center time, `neuron` is the neuron to whom the spikes in `event` belong, `lN.1` is $t - t_l$ and `i1` is isi_1.

Smoothing spline

- Since cellular biophysics does not provide much guidance on how to build $\eta(t - t_l, isi_1)$ we have chosen to use the nonparametric smoothing spline [Wahba, 1990, Green and Silverman, 1994, Eubank, 1999, Gu, 2002] approach implemented in the `gss` (general smoothing spline) package of Chong Gu for `R`.
- $\eta(t - t_l, isi_1)$ is then uniquely decomposed as :

$$\eta(t - t_l, isi_1) = \eta_\emptyset + \eta_l(t_t - l) + \eta_1(isi_1) + \eta_{l,1}(t - t_l, isi_1)$$

- Where for instance:

$$\int \eta_1(u)du = 0$$

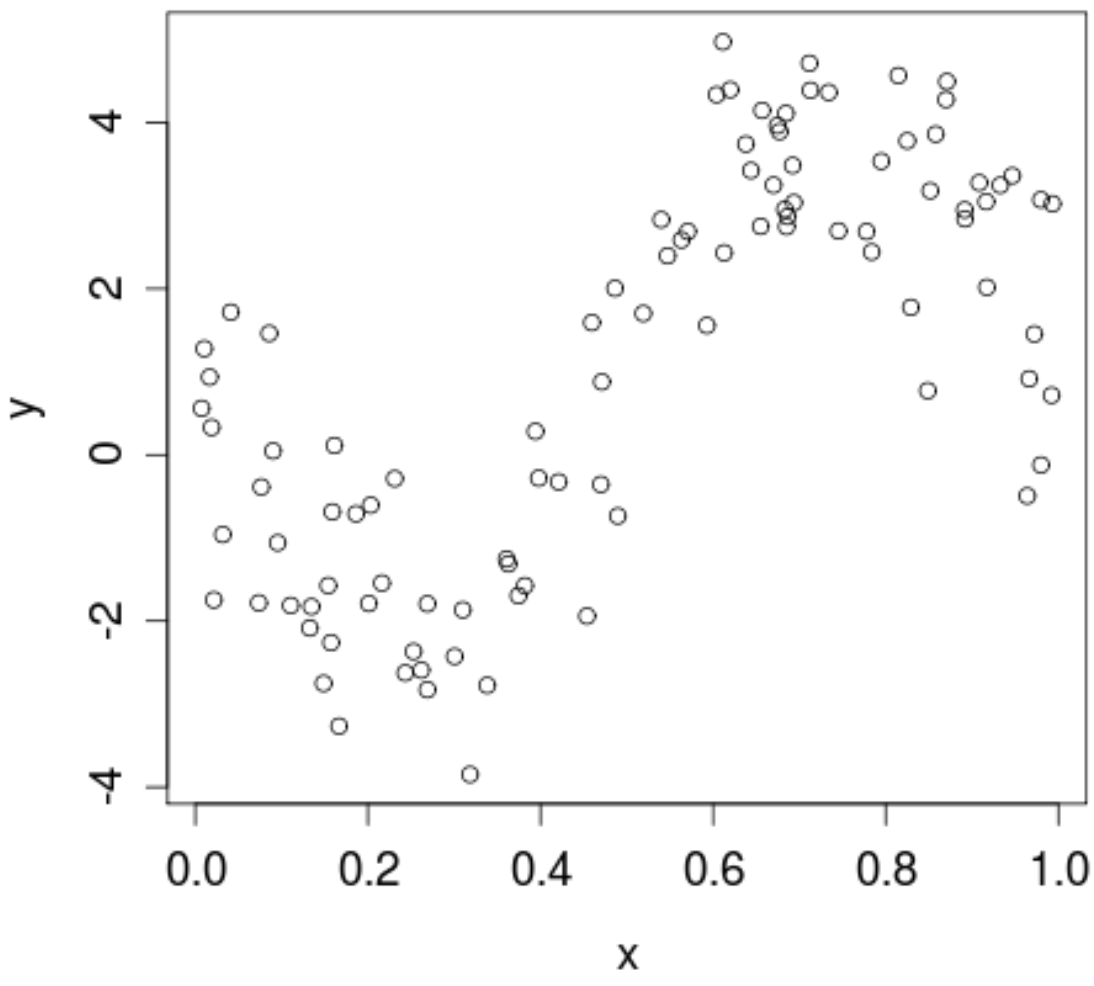
the integral being evaluated on the definition domain of the variable isi_1.

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_ρ minimizing:

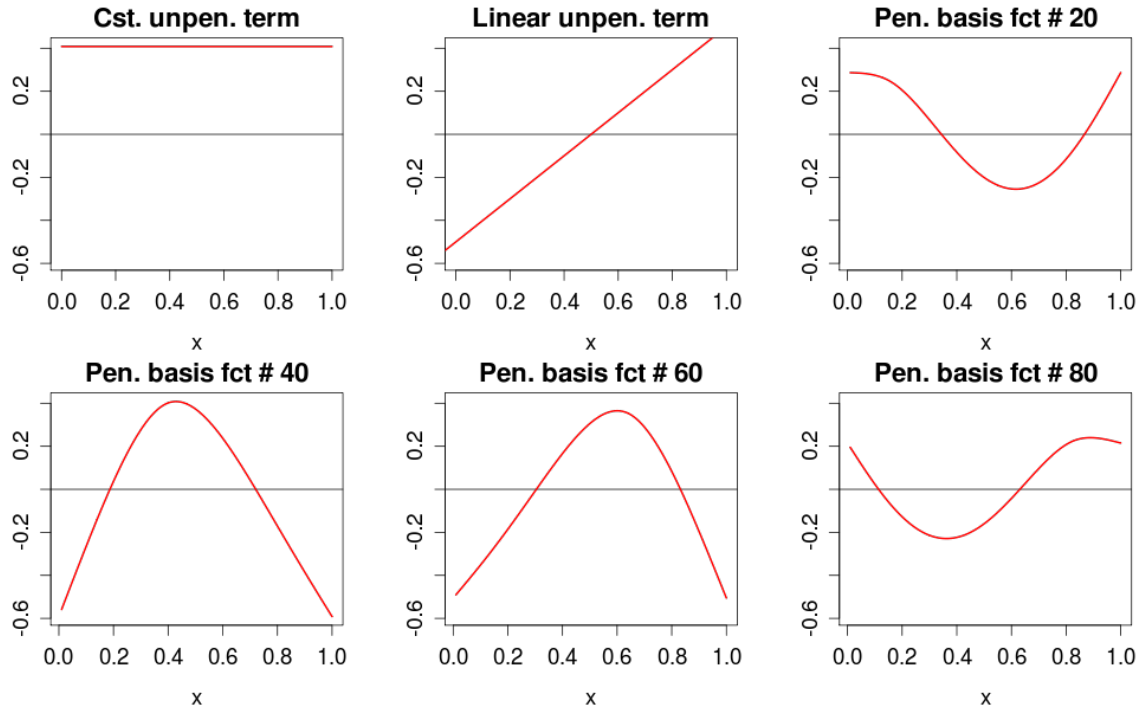
$$\frac{1}{n}\sum_{i=1}^{n}(Y_i - \eta_\rho(x_i))^2 + \rho \int_0^1 \left(\frac{d^2\eta_\rho}{dx^2}\right)^2 dx$$


A simple example with simulated data
y
x
4
2
0
-2
-4
0.0
0.2
0.4
0.6
0.8
1.0

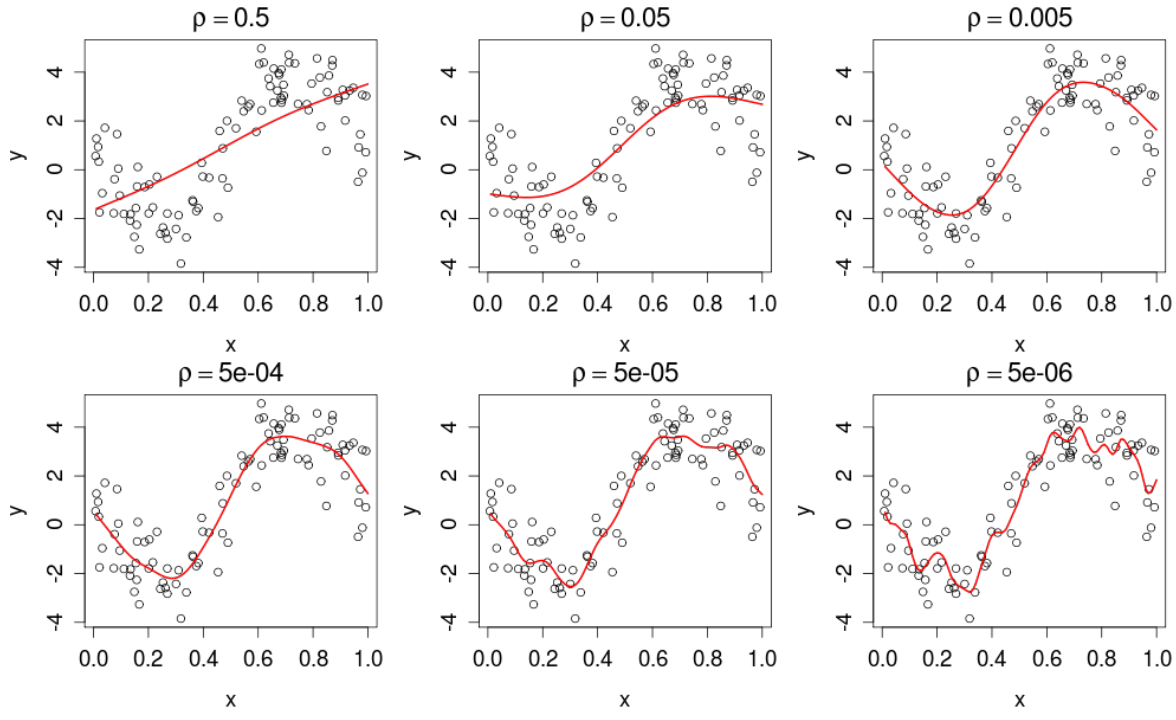
It can be shown [Wahba, 1990] that, for a given ρ, the solution of the functional minimization problem can be expressed on a finite basis:

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

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


Cst. unpen. term
Linear unpen. term
Pen. basis fct # 20
Pen. basis fct # 40
Pen. basis fct # 60
Pen. basis fct # 80
0.2
-0.2
-0.6
0.0
0.2
0.4
0.6
0.8
1.0
x

What about ρ?



Cross-validation

Ideally we would like ρ such that:

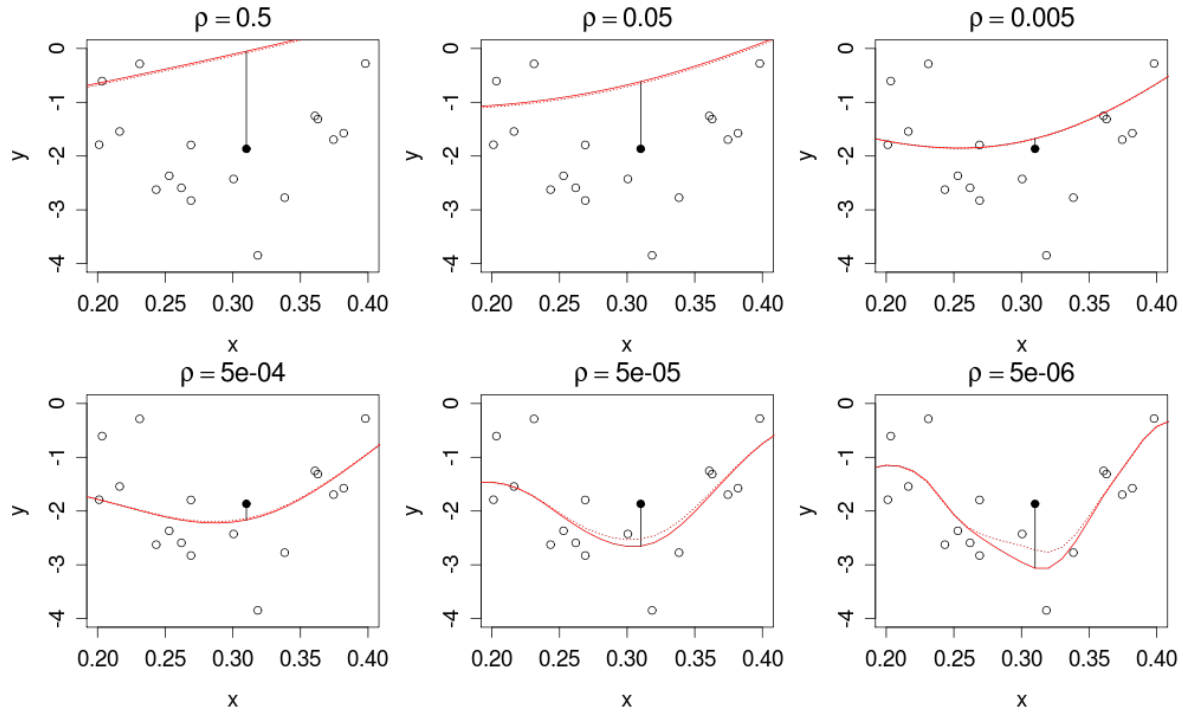
$$\frac{1}{n}\sum_{i=1}^{n}(\eta_\rho(x_i) - \eta(x_i))^2$$

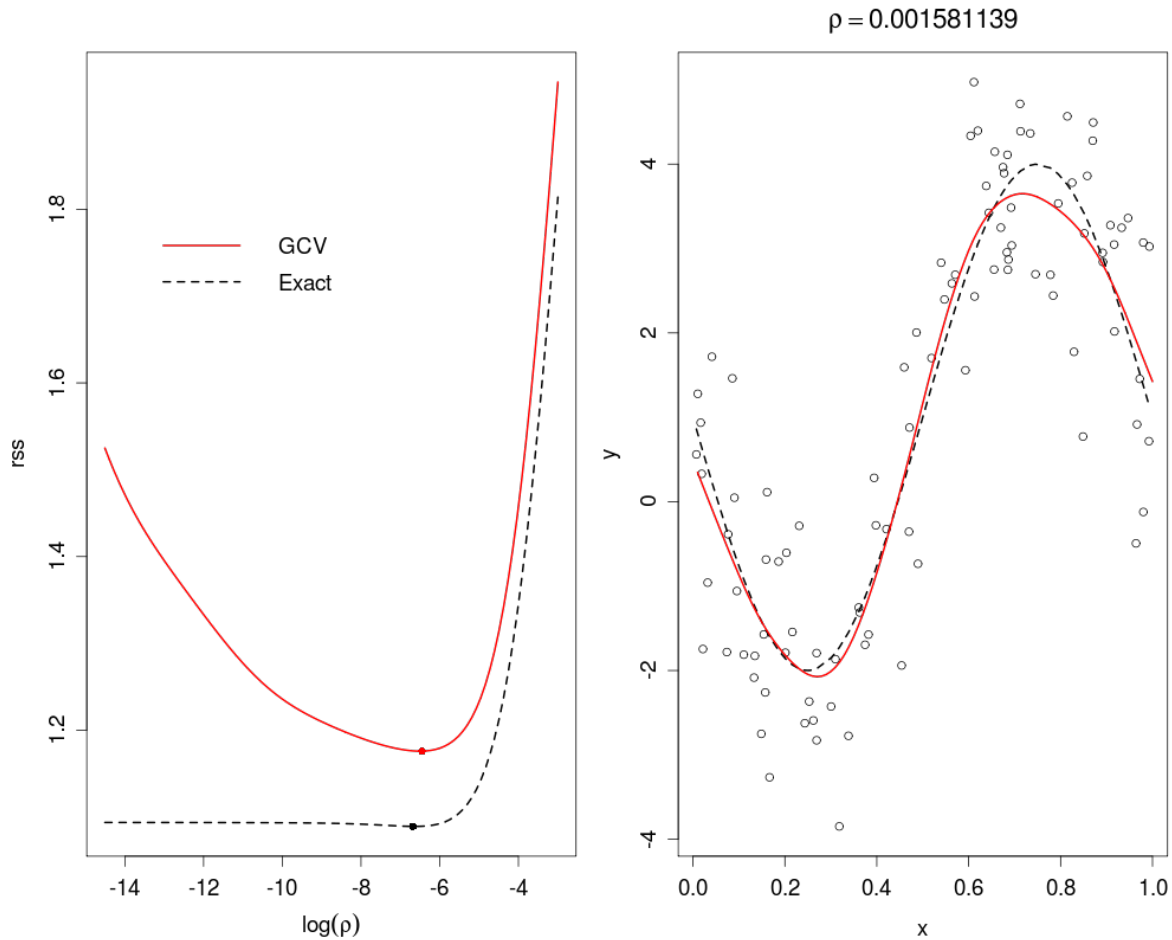
is minimized... but we don't know the true η. So we choose ρ minimizing:

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

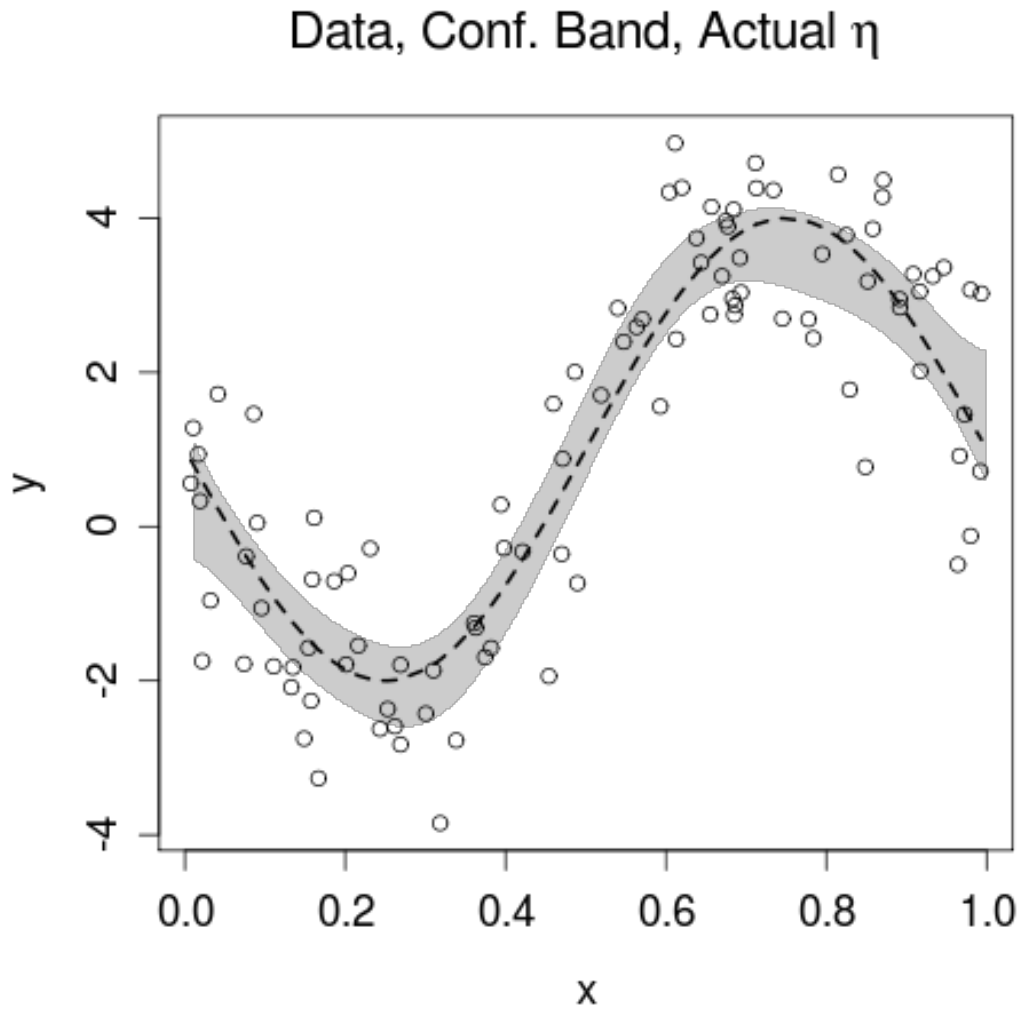
where $\eta_\rho^{[k]}$ is the minimizer of the "delete-one" functional:

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


ρ = 0.5
ρ = 0.05
ρ = 0.005
ρ = 5e-04
ρ = 5e-05
ρ = 5e-06
x
y
0.20
0.25
0.30
0.35
0.40
0
-1
-2
-3
-4


GCV
Exact
rss
log(ρ)
ρ = 0.001581139
y
x

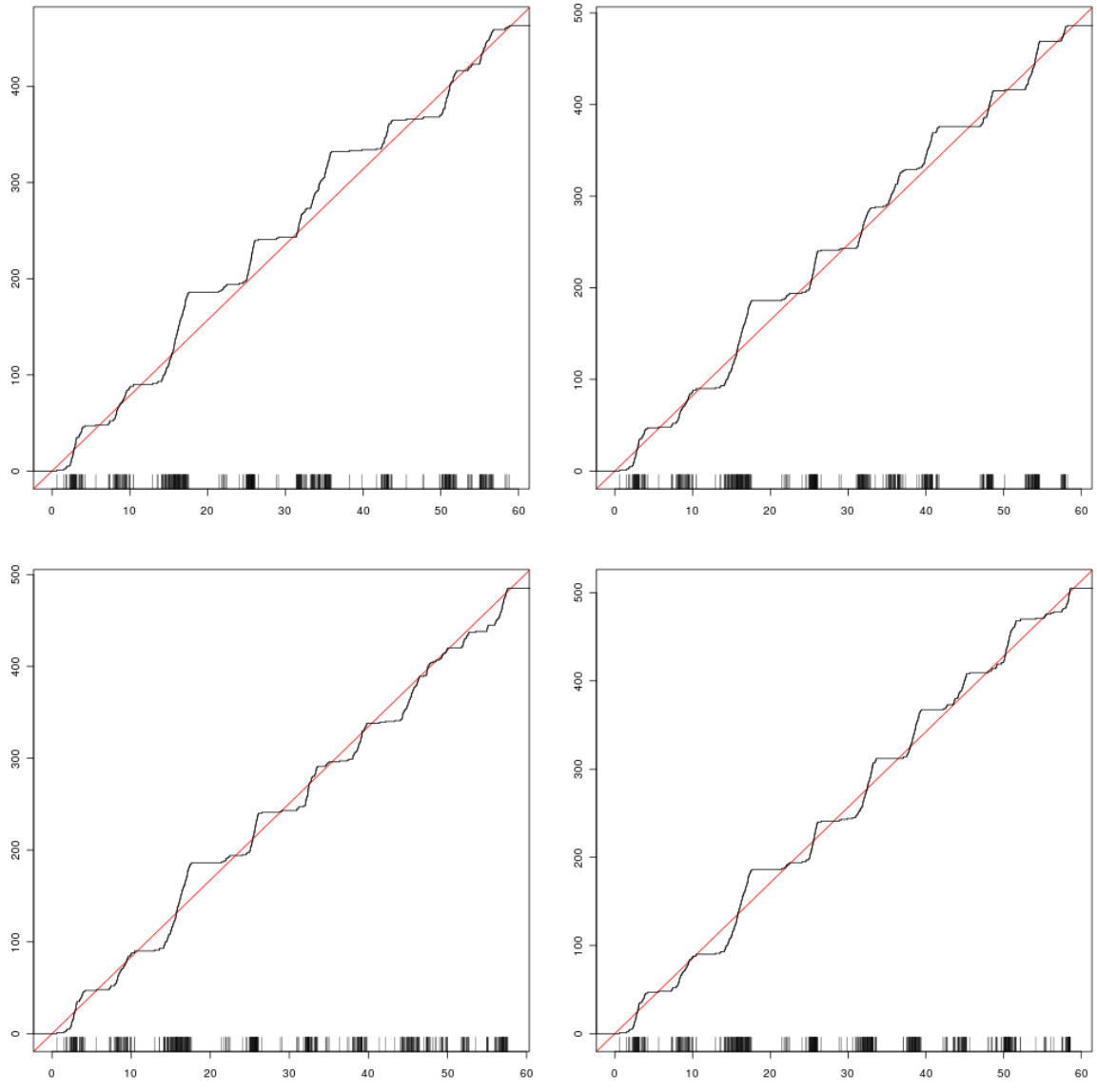
The theory (worked out by Grace Wahba) also gives us confidence bands



Going back to the real train

- On the next figure the actual spike train you saw previously will be shown again.
- Three other trains will be shown with it. The second half ($t \geq 29.5$) of each of these trains has been simulated.
- The simulation was performed using the same model obtained by fitting the first half of the data set.

Which one is the actual train?



The actual train can is in the lower right corner of the previous figure.

Towards the candidate model

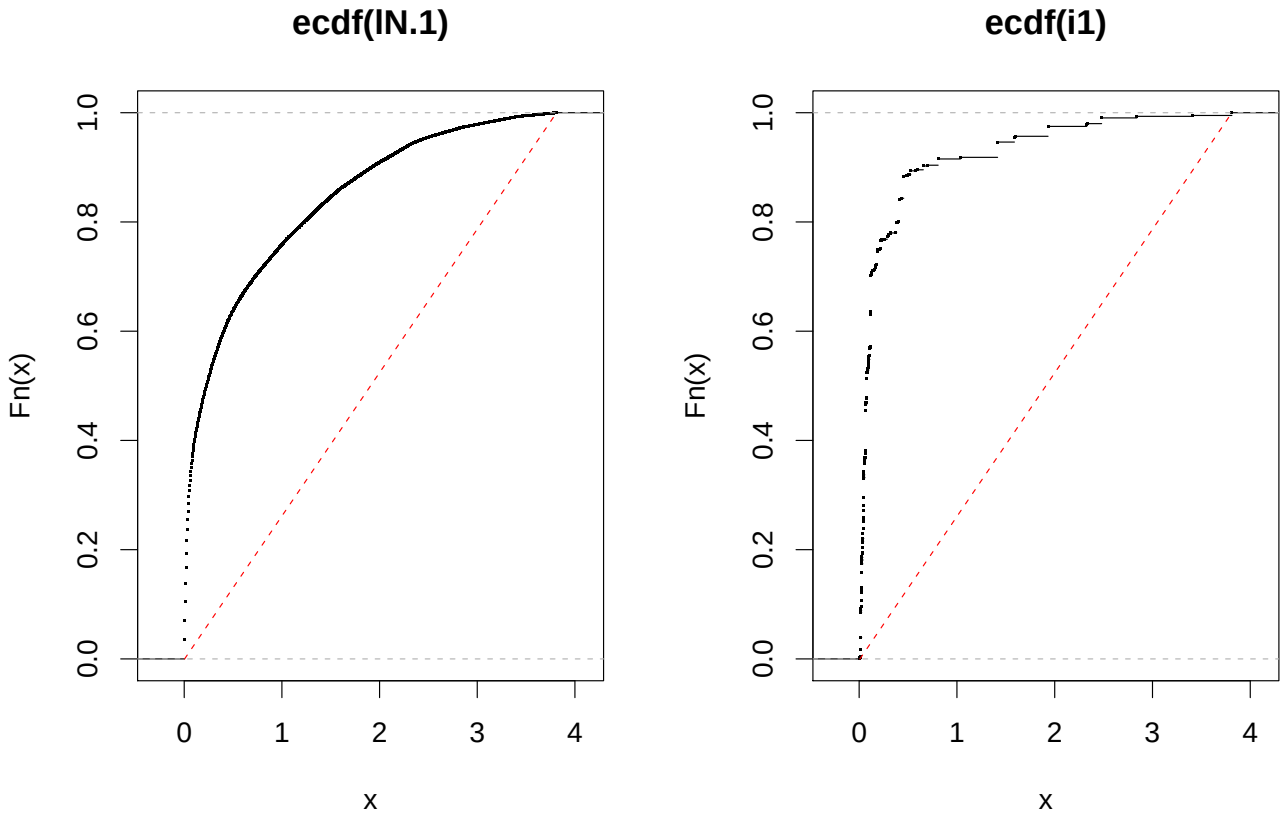
- We said previously that we would start with a 2 variables model:

$$\eta(t - t_l, isi_1) = \eta_\emptyset + \eta_l(t_t - l) + \eta_1(isi_1) + \eta_{l,1}(t - t_l, isi_1)$$

- Since we are using non-parametric method we should not apply our tests to the data used to fit the model. Otherwise our P-values will be wrong.
- We therefore systematically split the data set in two parts, fit the same (structural) model to each part and test it on the other part.

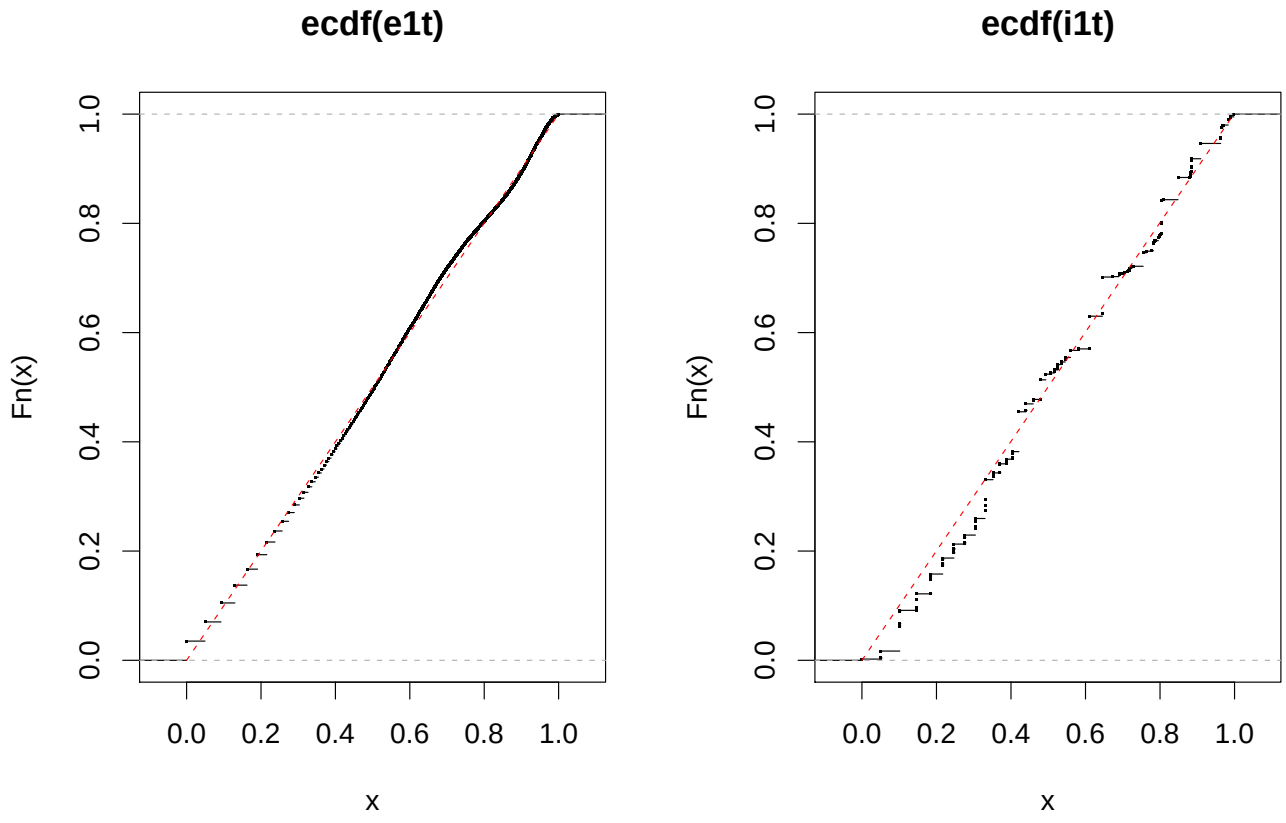
An important detail

The distributions of our variables, $t - t_l$ and isi_1 are very non-uniform:



For reasons we do not fully understand yet, fits are much better if we map our variables onto uniform ones.

We therefore map our variables using a smooth version of the ECDF estimated from the first half of the data set.



These mapped variables ECDFs are obtained from the whole data set.

Towards the candidate model

- We are going to actually fit 2 models to our data set:
  - Model 1:

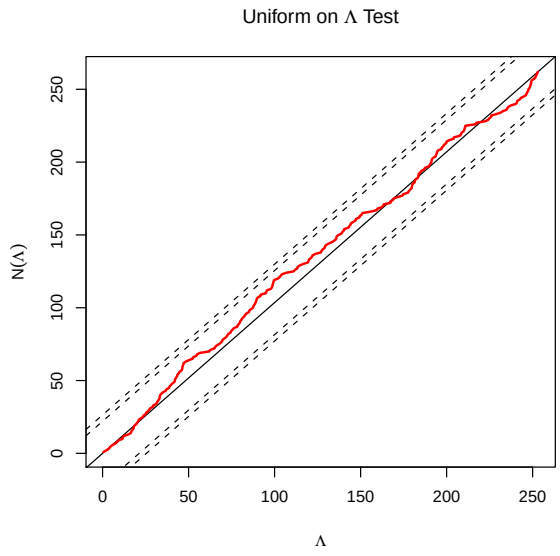
$$\eta(t - t_l, isi_1) = \eta_\emptyset + \eta_l(t_t - l) + \eta_1(isi_1) + \eta_{l,1}(t - t_l, isi_1)$$

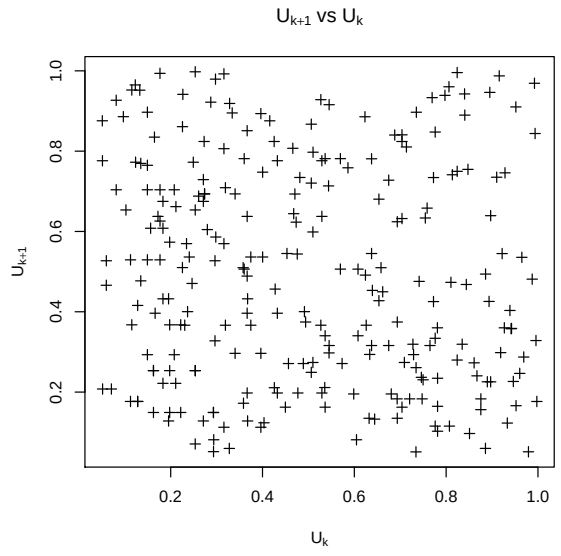
  - Model 2:

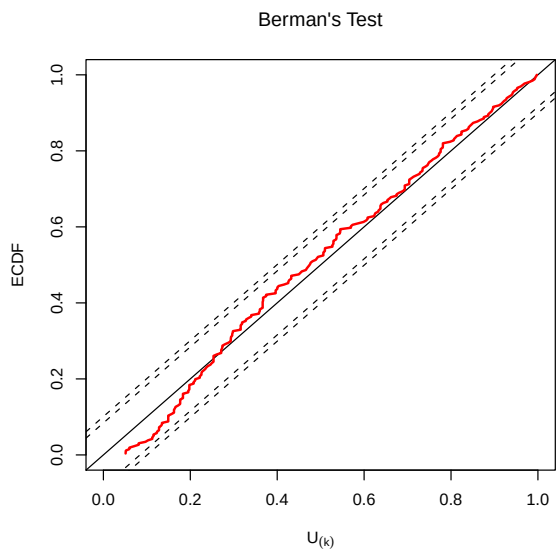
$$\eta(t - t_l, isi_1) = \eta_\emptyset + \eta_l(t_t - l) + \eta_1(isi_1)$$

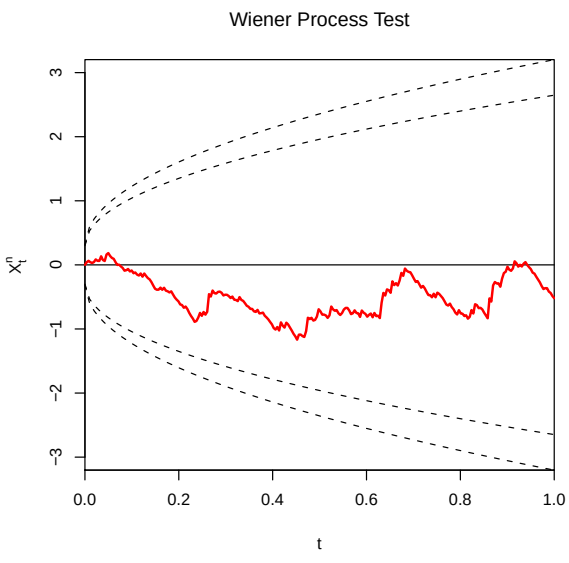
    Model 2 is called an additive model in the literature.
- Clearly Model 1 is more complex than Model 2

Model 1 Fit Early Test Late

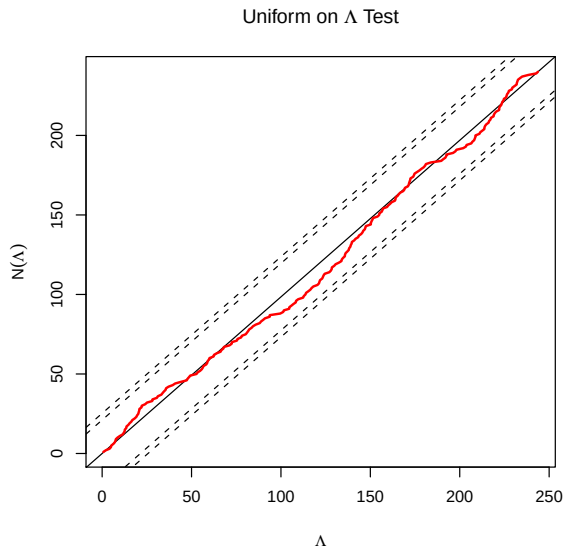


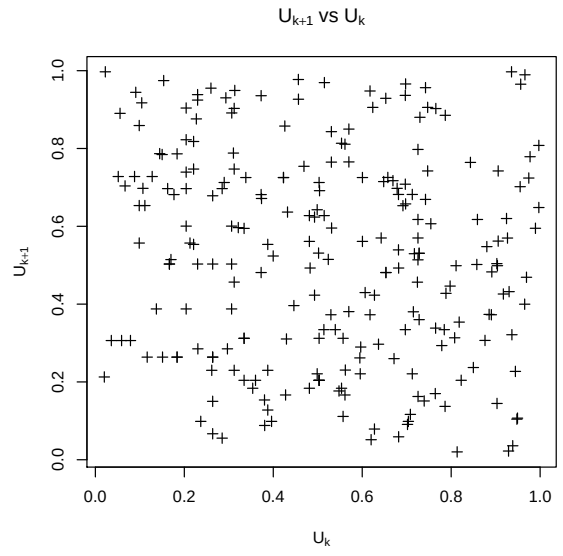


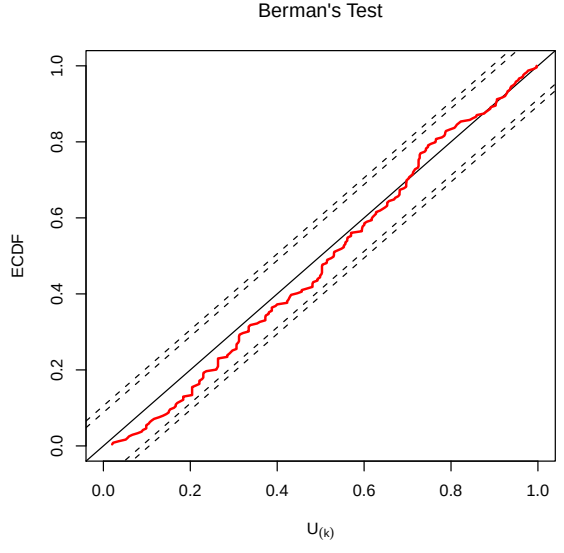


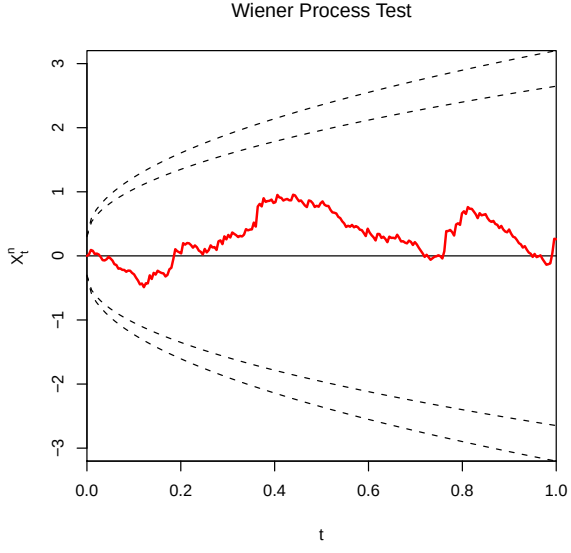


Model 1 Fit Late Test Early

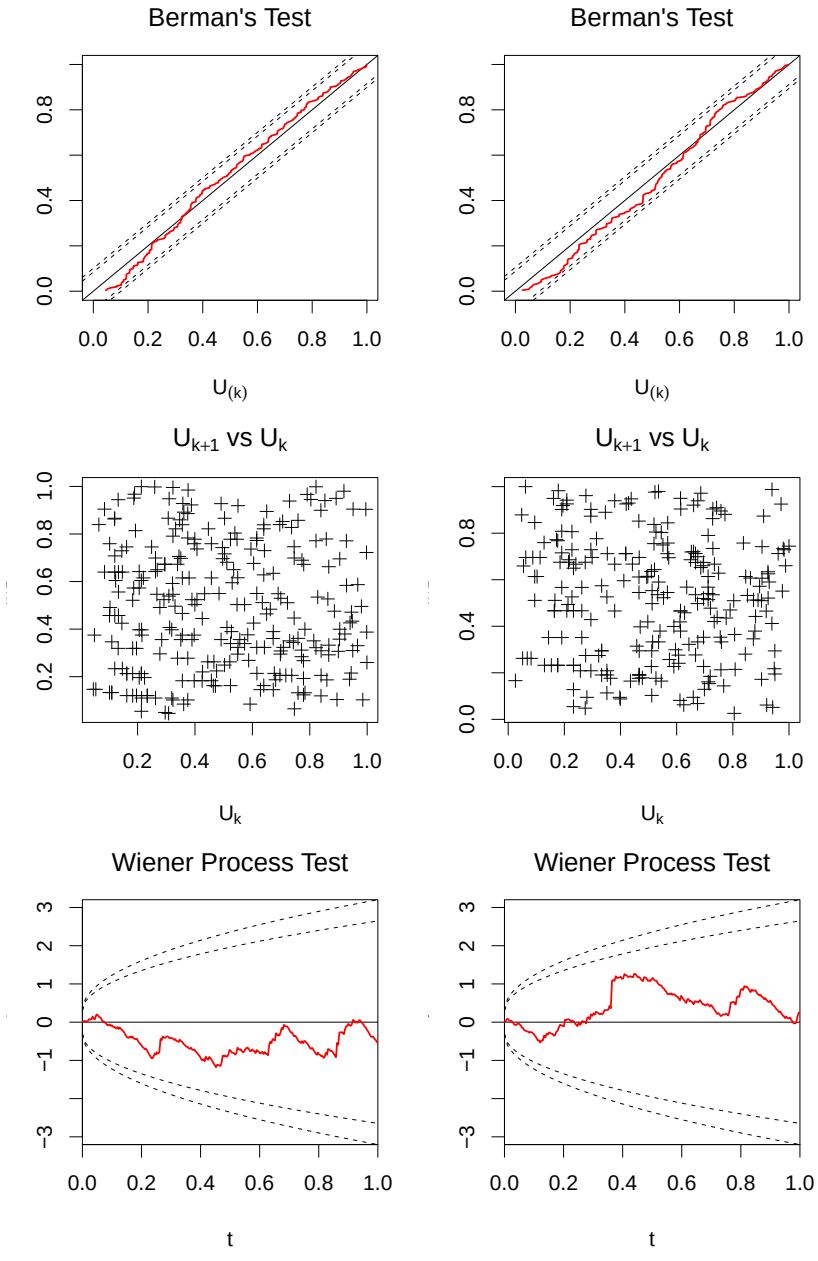








Model 2 Fit Early Test Late and Fit Late Test Early



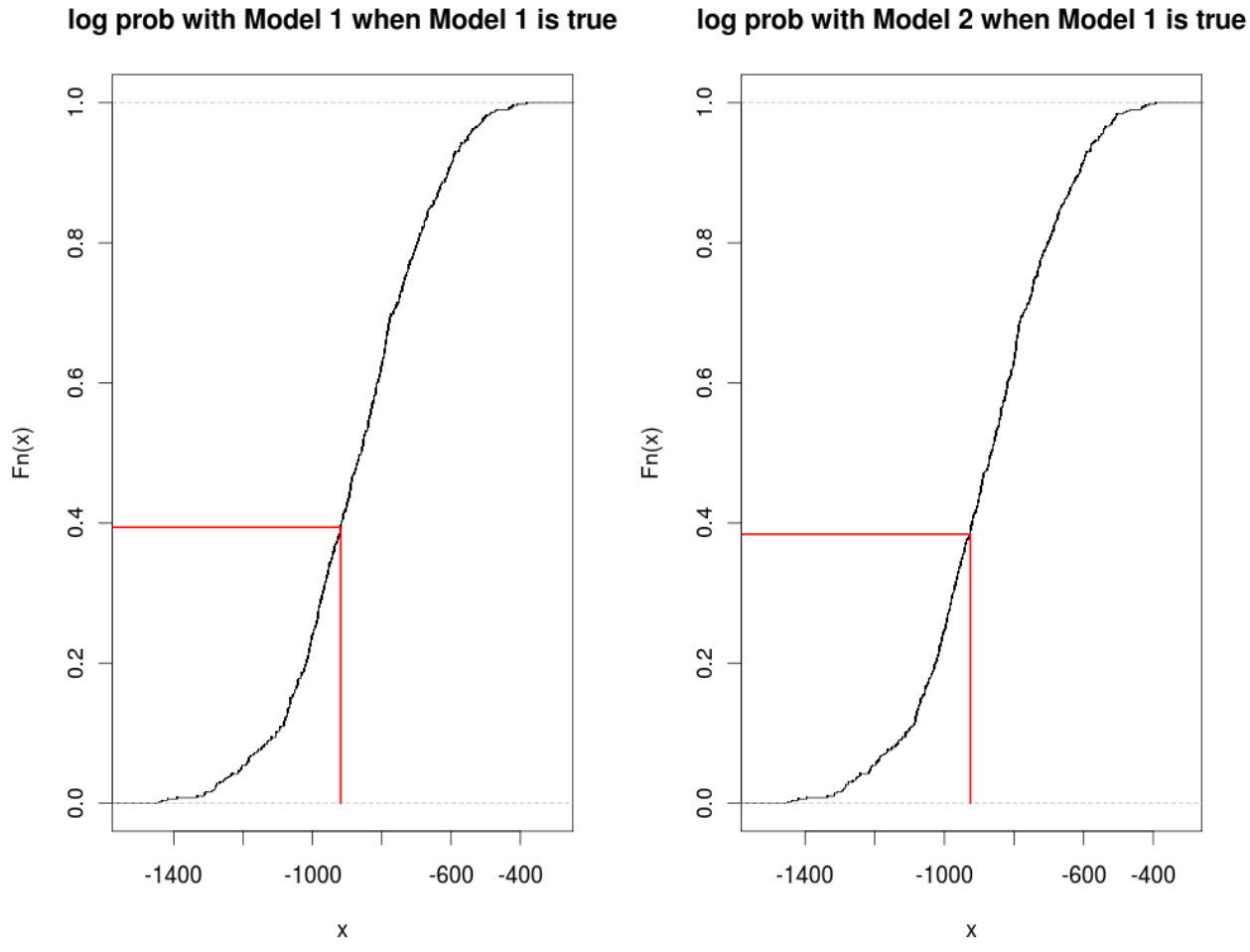
- We now have two candidate models passing our tests. Which one should we choose?
- We could argue that since Model 2 is the simplest, we should keep it.
- We could also use the probability (or its log) given by each model to the data. Let y_i be the indicator of the presence ($y_i = 1$) or absence ($y_i = 0$) of a spike in bin i. Let $p_{1,i}$ and $p_{2,i}$ the probabilities of having a spike in bin i given by model 1 and 2. Then,

$$\text{Prob}\{Y_i = y_i \mid \text{Model } k\} = p_{k,i}^{y_i}(1 - p_{k,i})^{1-y_i}$$

  We can therefore attach a number (a probability) to our binned spike train and we get for the log probability, -918.517 with Model 1 and -925.393 with Model 2.
- These last two numbers are obtained with data (y_i) of the second half and a model (p_i) fitted to the first half.

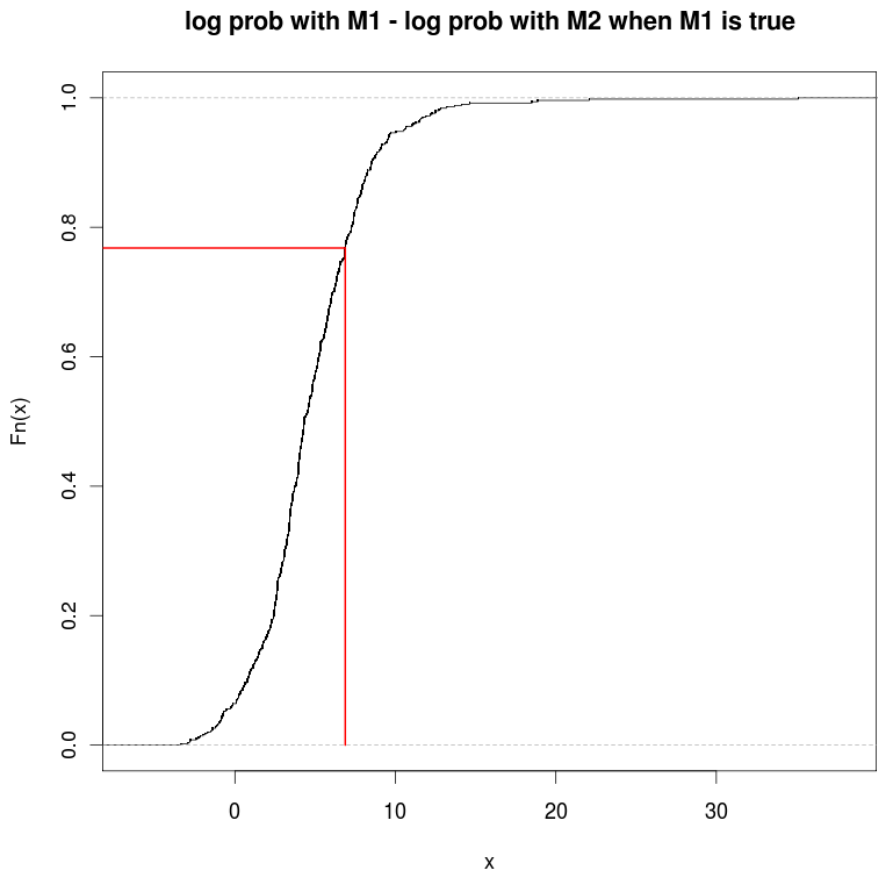
- The simplicity argument would lead us to select Model 2 while the probability argument would lead us to select Model 1.
- The question becomes: How much confidence can we have is the difference of 7 found between the two log probabilities?
- We address this question with a "parametric" bootstrap approach [Davison and Hinkley, 1997].
  - Assume Model k fitted to the first half is correct.
  - Simulate 500 spike trains corresponding to the second half using Ogata's thinning method [Ogata, 1981].
  - Compute the log probability with both models.
  - Get some summary stats out of these simulations.

Log Probs When Model 1 is True



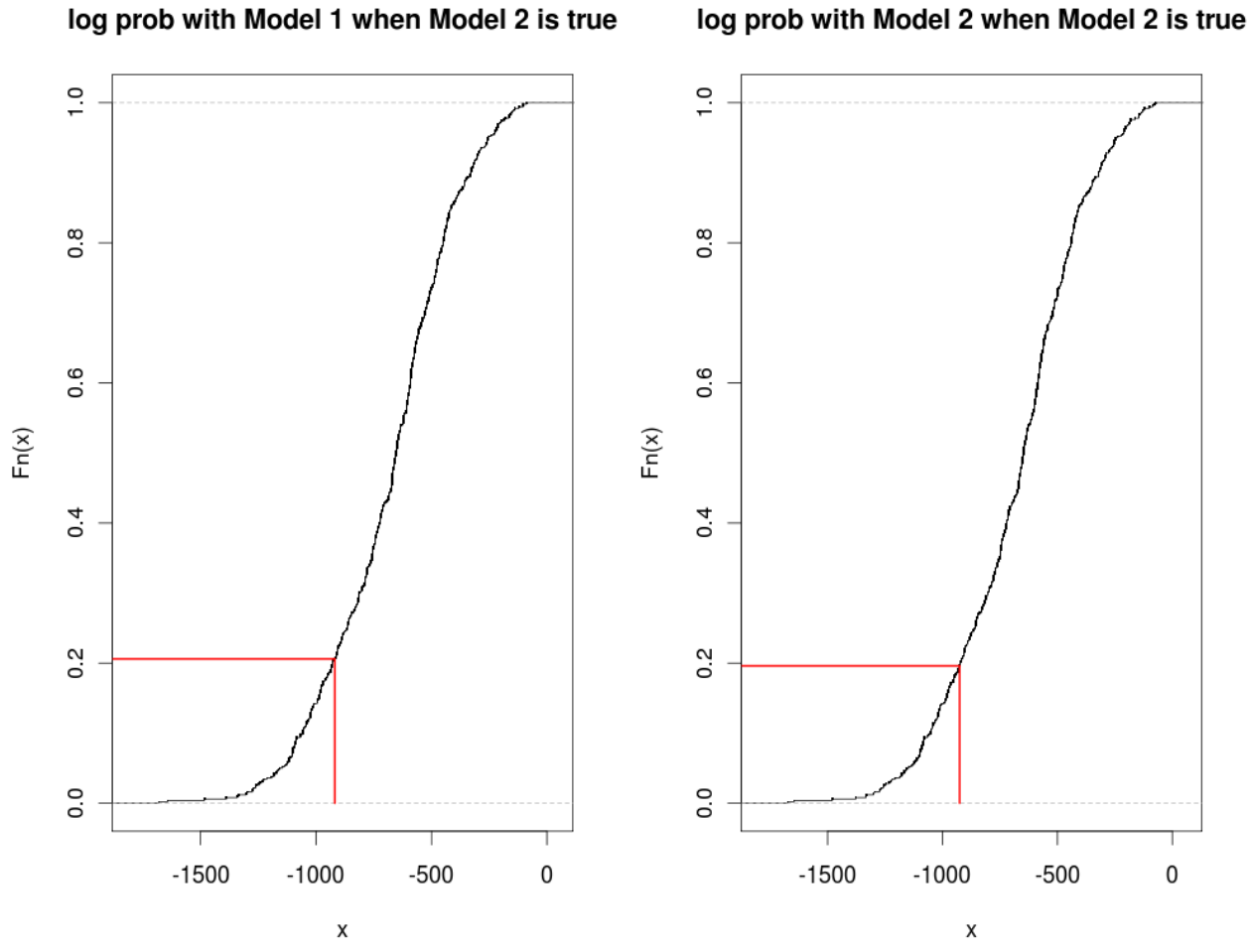
Red lines correspond to observed values.

Log Prob Difference When Model 1 is True



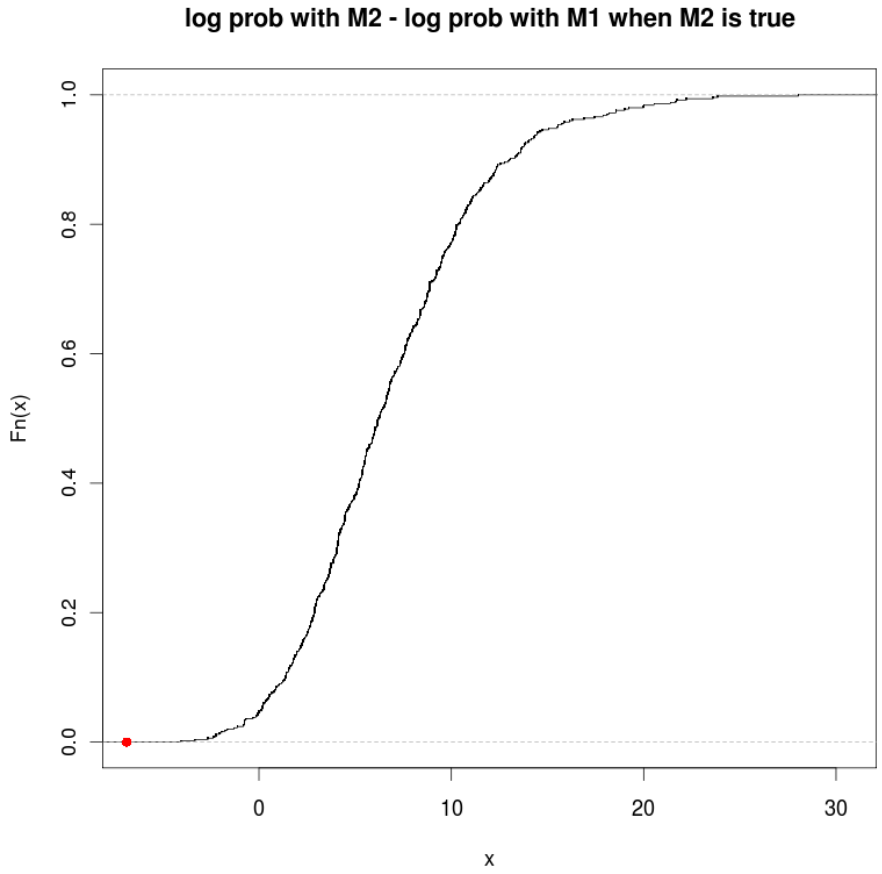
Red lines correspond to observed value. The mean value of this difference, 4.78 ± 0.16, is an estimator of the Kullback-Leibler divergence between Models 1 and 2.

Log Probs When Model 2 is True



Red lines correspond to observed values.

Log Prob Difference When Model 2 is True



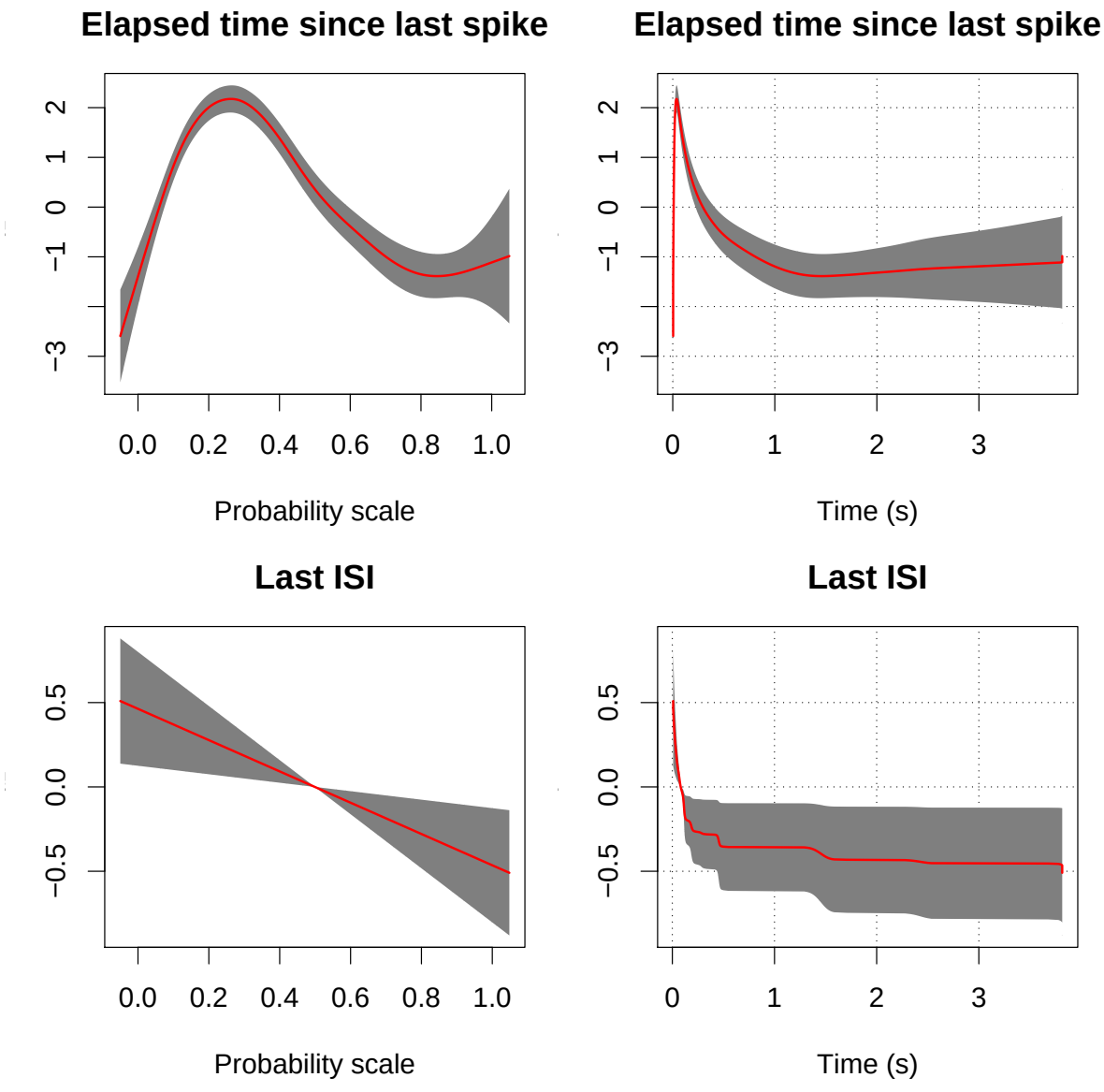
Red lines correspond to observed value. The mean value of this difference, 6.85 ± 0.22, is an estimator of the Kullback-Leibler divergence between Models 2 and 1.

- Our "parametric bootstrap" approach clearly rules out Model 2 as a candidate model.
- We are therefore left with the model including interactions between its two variables, Model 1:

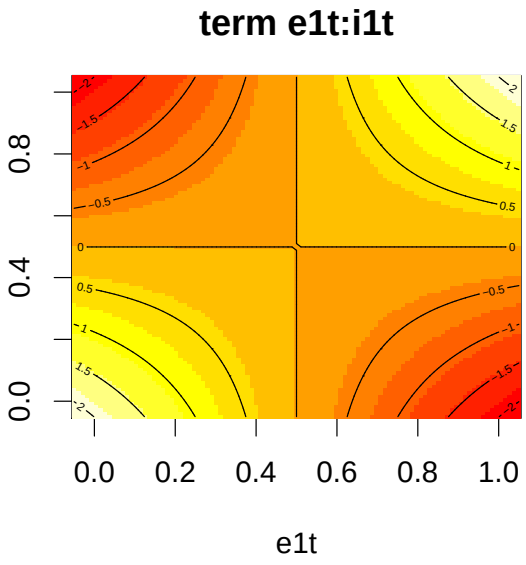
$$\eta(t - t_l, isi_1) = \eta_\emptyset + \eta_l(t_t - l) + \eta_1(isi_1) + \eta_{l,1}(t - t_l, isi_1)$$

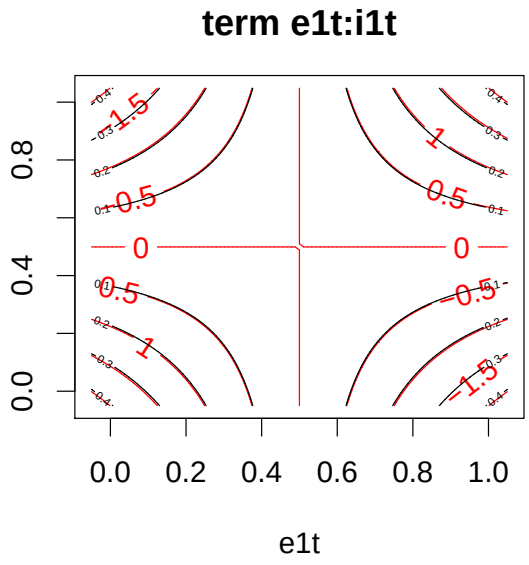
- The plots of the model terms, $\eta_l(t_t - l)$, $\eta_1(isi_1)$ and $\eta_{l,1}(t - t_l, isi_1)$ were obtained after refitting Model 1 to the full data set.

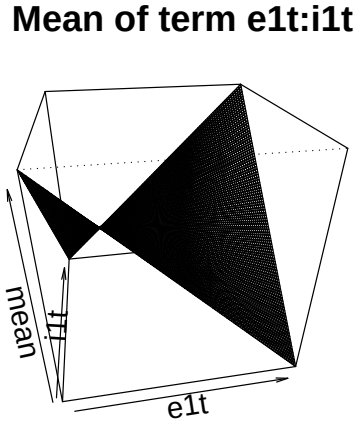
The functional forms: Uni-variate terms

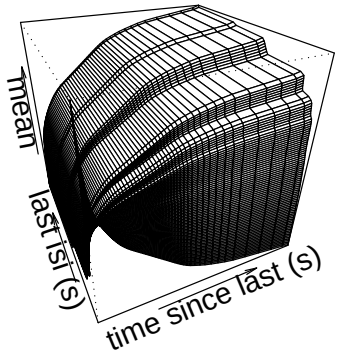


The functional forms: Interaction term

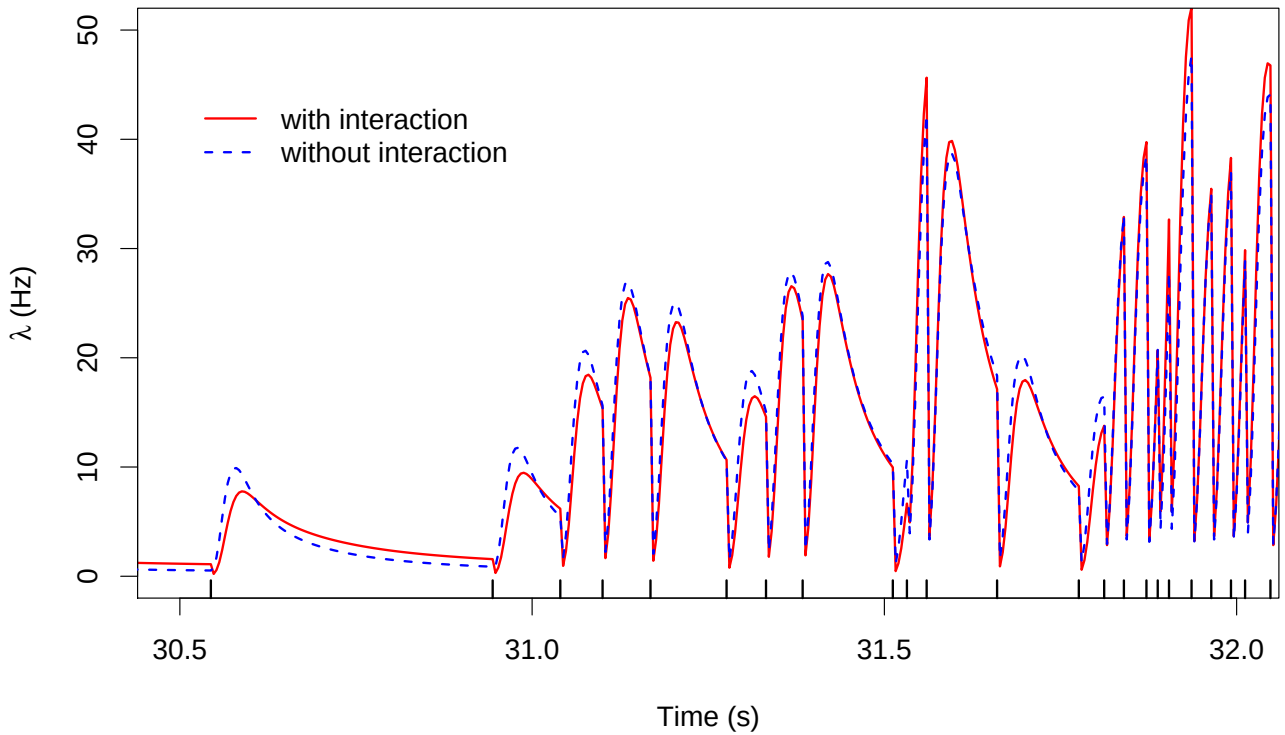








Intensities of Models 1 and 2



Conclusions

- We have now a "general" estimation method for the conditional intensity of real spike trains.
- The method is implemented in the `STAR` (Spike Train Analysis with R) package available on `CRAN` (the Comprehensive R Archive Network).
- An ongoing systematic study (see the STAR web site) shows:
  - Most of our discharges can be explained by models involving $t - t_l$ and isi_1.
  - "Irregular bursty" discharges require an additive model like Model 2 here while "Regular bursty" ones require an interaction term like in Model 1 here.
  - Some neurons require functional coupling with other neurons.
  - Analysis of odour responses will follow soon.

Acknowledgments

I would like to warmly thank:

- Joachim Rudolph and Mamadou Mboup for their kind invitation.
- Ofer Mazor, Matthieu Delescluse, Gilles Laurent, Jean Diebolt and Pascal Viot for working with me on the spike sorting problem.
- Antoine Chaffiol and the whole Kloppenburg lab (Univ. of Cologne) for providing high quality data and for being patient enough with a slow developer like myself.
- Chong Gu for developing the `gss` package and for collaborating on this conditional intensity estimation problem.
- The `R` guys for making such a wonderful data analysis tool.
- Vilmos Prokaj, Olivier Faugeras and Jonhatan Touboul for pointing Donsker's theorem to me.
- Carl van Vreeswijk for feed-back on this work.
- You guys for listening to / reading me up to that point.