Statistics 222, Spatial Statistics.

Outline for the day:

1. Simulating Hawkes processes. See simetasmay2017.r.
2. Estimating Hawkes models. fithawkes2017.r.
3. Superthinning, and fitting and simulating Strauss models. day11.r.
4. Nonparametric Hawkes estimation. day11.r.

1. The integral term in the loglikelihood for Hawkes processes.

loglikelihood = $\sum_i \log(\lambda(t_i,x_i,y_i))$ - $\iiint \lambda(t,x,y)\, dxdydt$.

The space-time region is B = [0,T] x S.

For a Hawkes process, $\lambda(t,x,y) = \mu\ \rho(x,y) + K\sum_{i:ti<t} g(t-t_i, x-x_i, y-y_i)$, where ρ and g are densities.

$\int_0^T \iint \lambda(t,x,y)\, dxdydt = \int_0^T \iint \mu\ \rho(x,y)\, dxdydt + \int_0^T \iint K\sum_{i:ti<t} g(t-t_i, x-x_i, y-y_i)\, dxdydt$

$= \mu\ T + \int_0^T \iint K\int_B 1_{\{t'<t\}}\, g(t-t', x-x', y-y')\, dN(t',x',y')\, dxdydt$

interchanging the integrals

$= \mu\ T + K\int_B \int_0^T \iint 1_{\{t'<t\}}\, g(t-t', x-x', y-y')\, dxdydt\, dN(t',x',y')$

changing coordinates, letting u = t-t', v = x-x', w = y-y',

$= \mu\ T + K\int_B \int_0^{T-t'} \iint_{S-(x',y')} g(u, v, w)\, dudvdw\, dN(t',x',y')$

$\sim \mu\ T + K \int_B (1)\, dN(t',x',y')$

$= \mu\ T + KN(B)$.

This is approximate because typically $\int_0^\infty \int_{-\infty}^\infty \int_{-\infty}^\infty g(u,v,w)\, dudvdw = 1$, but instead, we have $\int_0^{T-t'} \iint_{S-(x',y')} g(u,v,w)\, dudvdw$ which is often close to 1.

2. Nonparametric triggering function estimation.

Marsan and Lengliné (2008) assume g is a step function, and estimate steps β_k as parameters.

$$\ell(\theta) = \sum_i \log\left(\lambda(\tau_i, \mathbf{x}_i | \mathcal{H}_{\tau_i})\right) - \int_0^T \int_S \lambda(t, \mathbf{x} | \mathcal{H}_t)\, d\mathbf{x} dt$$

Setting the partial derivatives of this loglikelihood with respect to the steps β_k to zero yields

$$0 = \partial\ell(\theta)/\partial\beta_k = \sum_{(i,j):\tau_i - \tau_j \in U_k} K/\lambda(\tau_i) - Kn|U_k|,$$

where $|U_k|$ is the width of step k, for $k = 1,2,\ldots, p$. This is a system of p equations in p unknowns.

However, the equations are nonlinear. They depend on $1/\lambda(\tau_i)$.

Gradient descent methods: way too slow for large p.

Marsan and Lengliné (2008) find *approximate* maximum likelihood estimates using the E-M method for point processes. You pick initial values of the parameters, then given those, you know the probability event i triggered event j. Using these, you can weight each pair of points by its probability and re-estimate the parameters, and repeat until convergence.

This method works well but is iterative and time-consuming.

Analytic solution.

Set $p = n$. (p = *number of steps in the step function*, g, *and* n = *# of observed points*.)

Setting the derivatives of the loglikelihood to zero we have the p equations

$$0 = \partial\ell(\theta)/\partial\beta_k = \sum_{(i,j):\tau_i - \tau_j \in U_k} K/\lambda(\tau_i) - Kn|U_k|,$$

which are p linear equations in terms of $1/\lambda(\tau_i)$, for $i = 1, 2, ..., n$. (!)

So, if $p=n$, then we can use these equations to solve for $1/\lambda(\tau_i)$,

and if we know $1/\lambda(\tau_i)$, then we know $\lambda(\tau_i)$,

and if we know $\lambda(\tau_i)$, then we can solve for β_i because the def. of a Hawkes process is

$$\lambda(\tau_j) = \mu + K \sum_{i<j} g(\tau_j - \tau_i),$$

which results in n linear equations in the p unknowns β_1, β_2, ..., β_p, when g is a step function.

Analytic solution.

We can write the resulting estimator in very condensed form.

Let $\boldsymbol{\lambda} = \{\lambda(\tau_1),\lambda(\tau_2),\ldots, \lambda(\tau_n)\}$.

Suppose the steps of g have equal widths, $|U_1| = |U_2|$, etc. Call this width U.

Let A_{ij} = the number points τ_k such that $\tau_j - \tau_k$ is in U_i, for i,j in $\{1,2,\ldots,p\}$.

Then the loglikelihood derivatives equalling zero can be rewritten

$$0 = KA(1/\boldsymbol{\lambda}) - Kb,$$

where $\mathbf{b} = nU\mathbf{1}$, with $\mathbf{1} = \{1,1,\ldots,1\}$.

This has solution $\underline{1/\boldsymbol{\lambda} = A^{-1} b}$, if A is invertible.

Similarly, the Hawkes equation can be rewritten $\boldsymbol{\lambda} = \mu + KA^T\boldsymbol{\beta}$, whose solution is

$$\underline{\hat{b} = (KA^T)^{-1}(\lambda\text{-}\mu)}.$$

Combining these two underlined formulas yields the estimates

$$\hat{\beta} = (KA^T)^{-1}[1/(A^{-1}b) - \mu]$$

This is very simple, trivial to program, and rapid to compute.

Analytic solution.

There are problems, however.

1. Estimating $n=p$ steps. High variance.

   However, if we can assume g is smooth, then we can smooth our estimates for stability.

2. Need to estimate K and μ too.

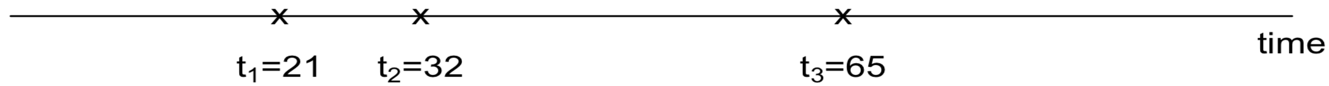
   We can use Marsan and Lengliné's method or take derivatives for these as well.

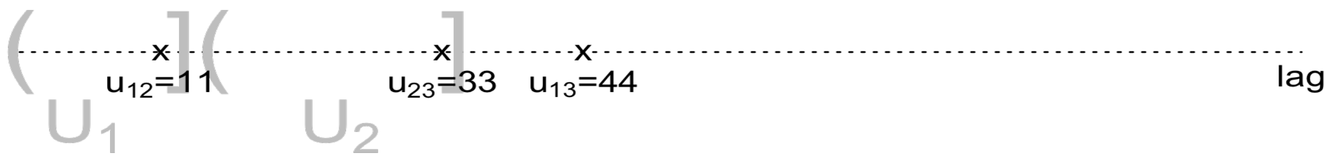
3. What about spatially-varying steps and unequally sized steps for g?

   No problem. The estimation generalizes in a completely obvious way.

4. A can be singular.
   We may need better solutions for this.
   I let $u_j = \tau_j - \tau_{j-1}$, sorted the u_j values, and then used $[u_{(1)}, u_{(2)})$, etc. as my binwidths, so each row and column of A would have at least one non-zero entry. If it still isn't singular, adding in a few random 1's into A often helps.

$$\hat{\beta} = (KA^T)^{-1}[1/(A^{-1}b) - \mu]$$





4. *A* can be singular.

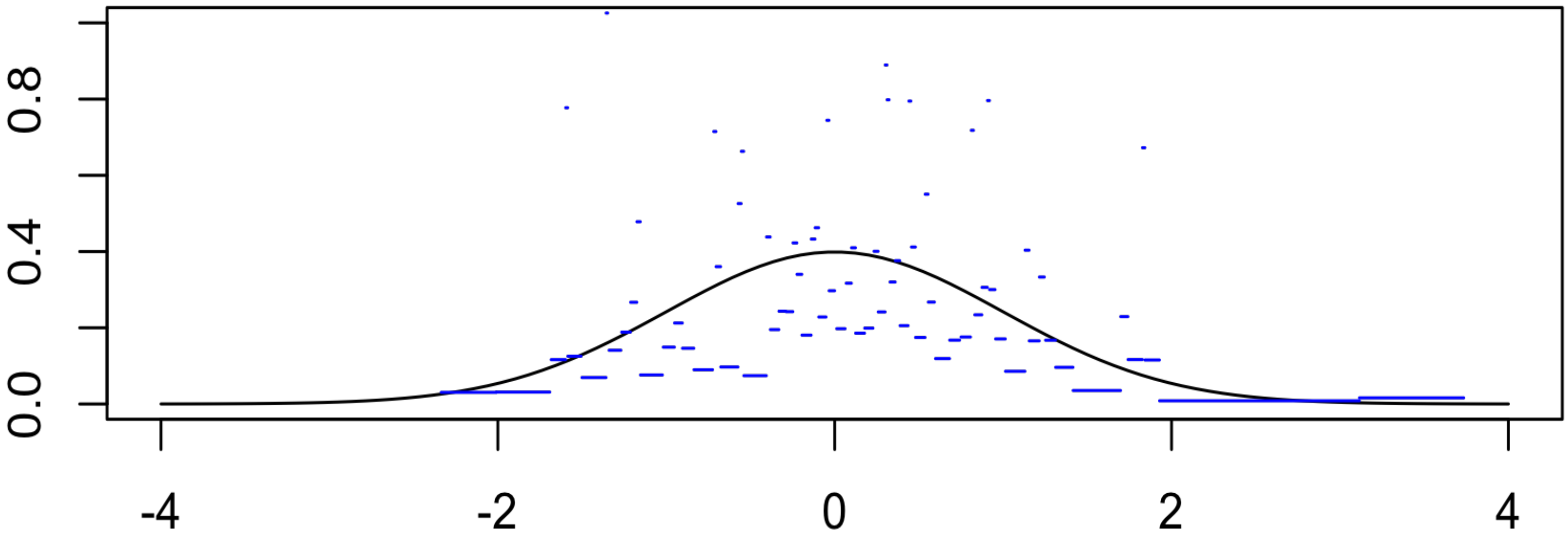
We may need better solutions for this.

I let $u_j = \tau_j - \tau_{j-1}$, sorted the u_j values, and then used $[u_{(1)}, u_{(2)})$, etc. as my binwidths, so each row and column of *A* would have at least one non-zero entry. If it still isn't singular, adding in a few random 1's into *A* often helps.

$$\hat{\beta} = (KA^T)^{-1}[1/(A^{-1}b) - \mu]$$

Note: take the simple case of a dataset where point i is only influenced by point i-1. This is basically a renewal process, and we are just estimating a renewal density.

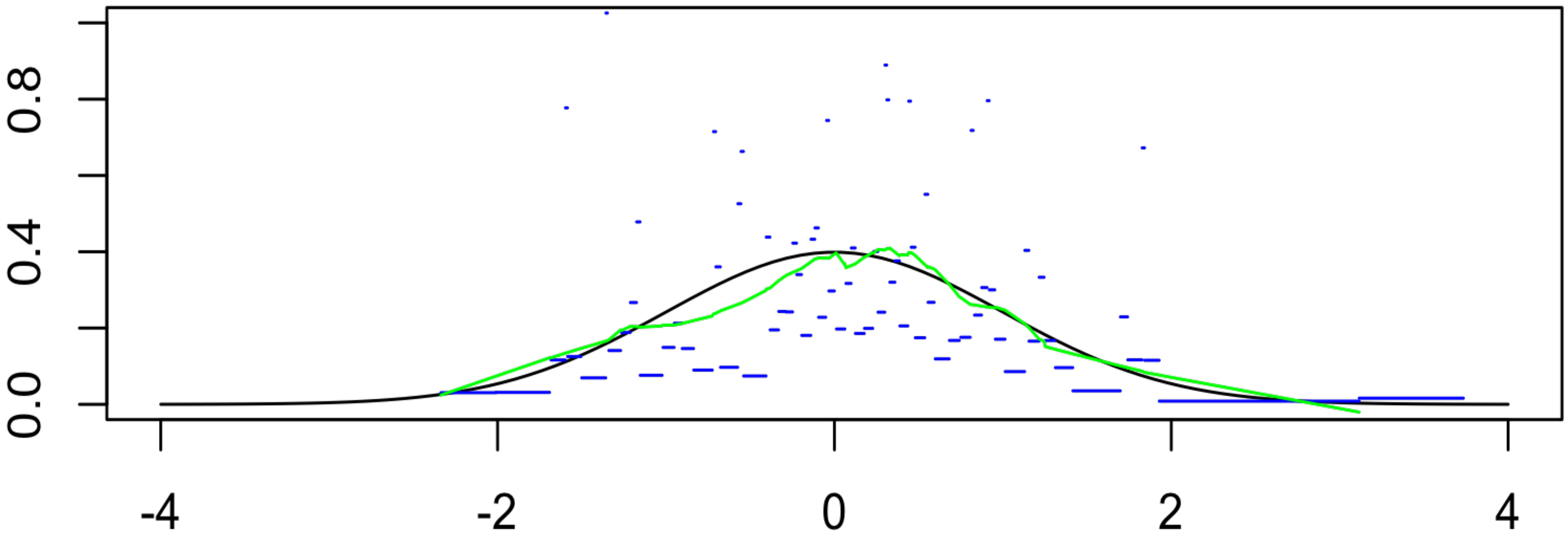
Here A = I, K = 1, and we get the density estimator $1/\{n(x_i - x_{i-1})\}$.



$$\hat{\beta} = (KA^T)^{-1}[1/(A^{-1}b) - \mu]$$

Note: take the simple case of a dataset where point i is only influenced by point i-1. This is basically a renewal process, and we are just estimating a renewal density.

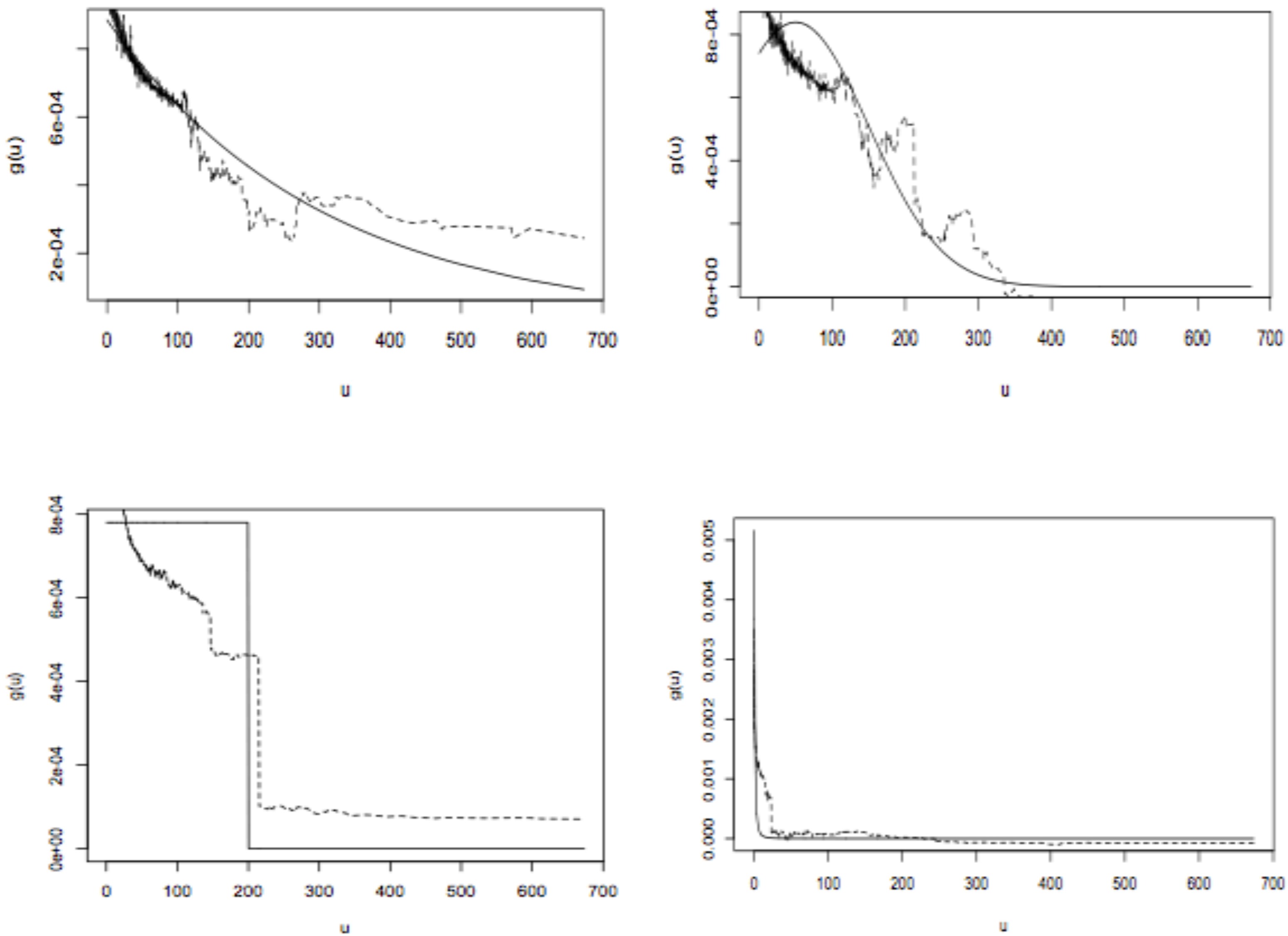
Here A = I, K = 1, and we get the density estimator $1/\{n(x_i - x_{i-1})\}$.



$$\hat{\beta} = (KA^T)^{-1}[1/(A^{-1}b) - \mu]$$

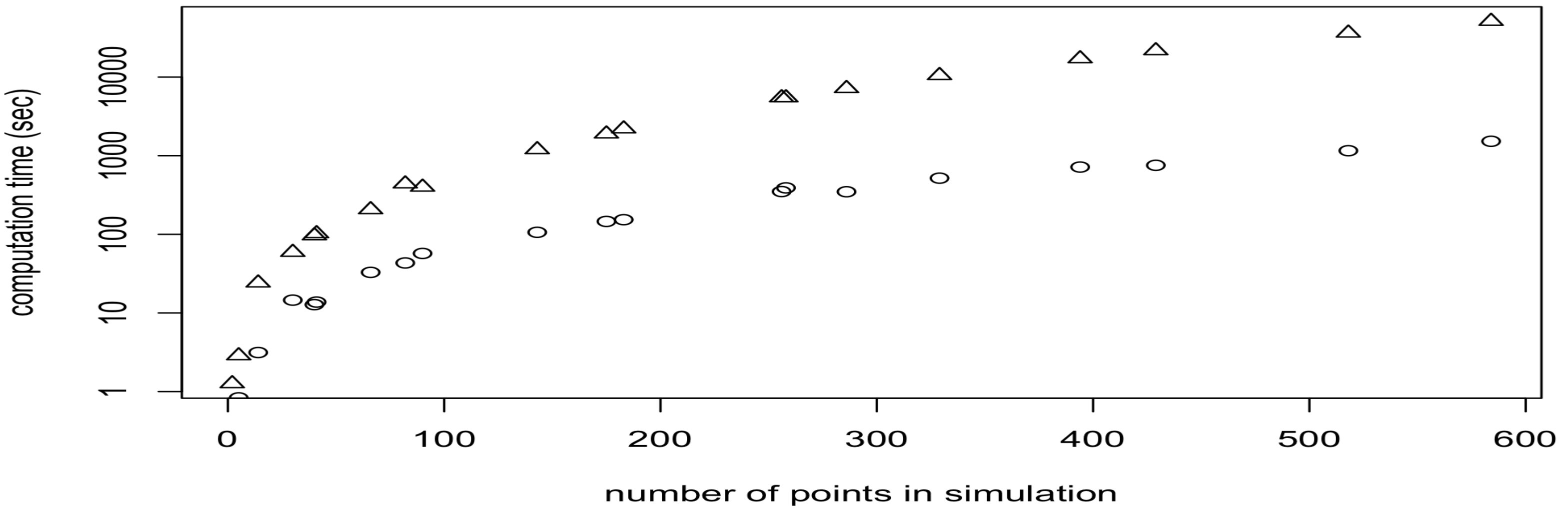
Computation time and performance comparison.

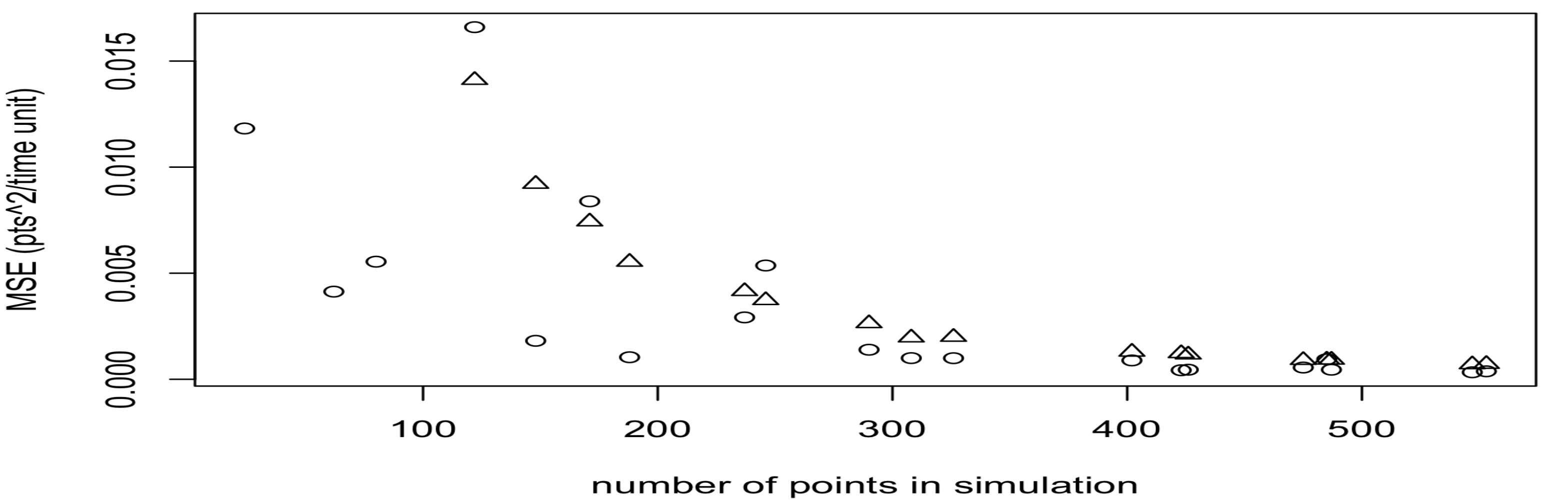
Test of concept. Examples of exponential, truncated normal, uniform, and Pareto g.



Computation time and performance comparison.

Triangles = Marsan and Lengliné (2008) method. Circles = analytic method.





The idea is to

let p=n,

let the derivatives of the log-likelihood be zero,

solve for $1/\lambda_i$ and therefore get λ_i,

and solve for β.

a. One can have major computation time savings from this method.

For datasets of only 100-300 points the savings are negligible.

However, for 5,000 points, the Marsan and Lengliné (2008) algorithm with 100 iterations takes about 7 hours, whereas the analytic method takes 1.3 min.

This speed facilitates computations like simulation based confidence intervals.

b. How far can this go?

It extends very readily to space-time-magnitude and estimation of μ .

Would this work for other types of models too? What are the limits on this method?

c. What about when A is singular? More work is needed.