Individual based movement estimation (Residuals and Petersen)

Anders Nielsen & Tobias Mildenberger

DTU-Aqua

an@aqua.dtu.dk

Plan

- Reminder
- Residuals: conventional, archival, position and time
- Petersen: conditional on movement

Setup movement matrix

- Construct a giant matrix $M^\star_{N\times N}$, where N is the number of cells in the study area plus one cell for each of the ways a fish can die (natual or caught by fishing).
- Assuming we have one fishing fleet (one spatial field for its F), an advection field α, a diffusion field D, and one natural mortality field M, then we can setup the matrix as:

$$M^\star_{i\to j} = \begin{cases} D_i/\Delta^2 + 0.5\alpha_i^{(x)}/\Delta, & \text{if cell } j \text{ right of cell } i \\ D_i/\Delta^2 - 0.5\alpha_i^{(x)}/\Delta, & \text{if cell } j \text{ left of cell } i \\ D_i/\Delta^2 + 0.5\alpha_i^{(y)}/\Delta, & \text{if cell } j \text{ above cell } i \\ D_i/\Delta^2 - 0.5\alpha_i^{(y)}/\Delta, & \text{if cell } j \text{ below cell } i \\ F_i, & \text{if cell } j \text{ is 'caught'} \\ M_i, & \text{if cell } j \text{ is 'dead'} \\ -\sum_{j\neq i} M^\star_{i,j}, & \text{if } i = j \text{ (obviously calculated last in row)} \\ 0, & \text{otherwise} \end{cases}$$

  Δ is size of grid cell. Notice that e.g. F can depend on where the fish is.

- Solved with matrix exponential to find probability of each tag history.

Fields

- Fields are constructed from environmental inputs I_1, I_2,...,I_m or effort E supplied on a grid, as:

$$
\begin{aligned}
h(i) &= S_1^{(h)}(I_1(i)) + \cdots + S_m^{(h)}(I_m(i)) \text{ used as } \alpha(i) = \nabla h(i) \\
\log D(i) &= S_1^{(D)}(I_1(i)) + \cdots + S_m^{(D)}(I_m(i)) \\
F(i) &= \lambda E(i)
\end{aligned}
$$

- The S-functions are splines.
- Notice the discrete nature of these fields.

Archival tags

- Data:

$$o_0 = (x_0, y_0),\ o_1 = (x_1, y_1),\ \ldots,\ o_n = (x_n, y_n)$$

  First and last known, all others subject to observation uncertainty

- Model:

$$\begin{aligned} o_t &\sim N(\psi_t, \Sigma_o) \\ \psi_{t+\Delta t} &\sim N(\psi_t + \alpha(\psi_t, t)\Delta t, 2D(\psi_t, t)\Delta t I_{2\times 2}) \end{aligned}$$

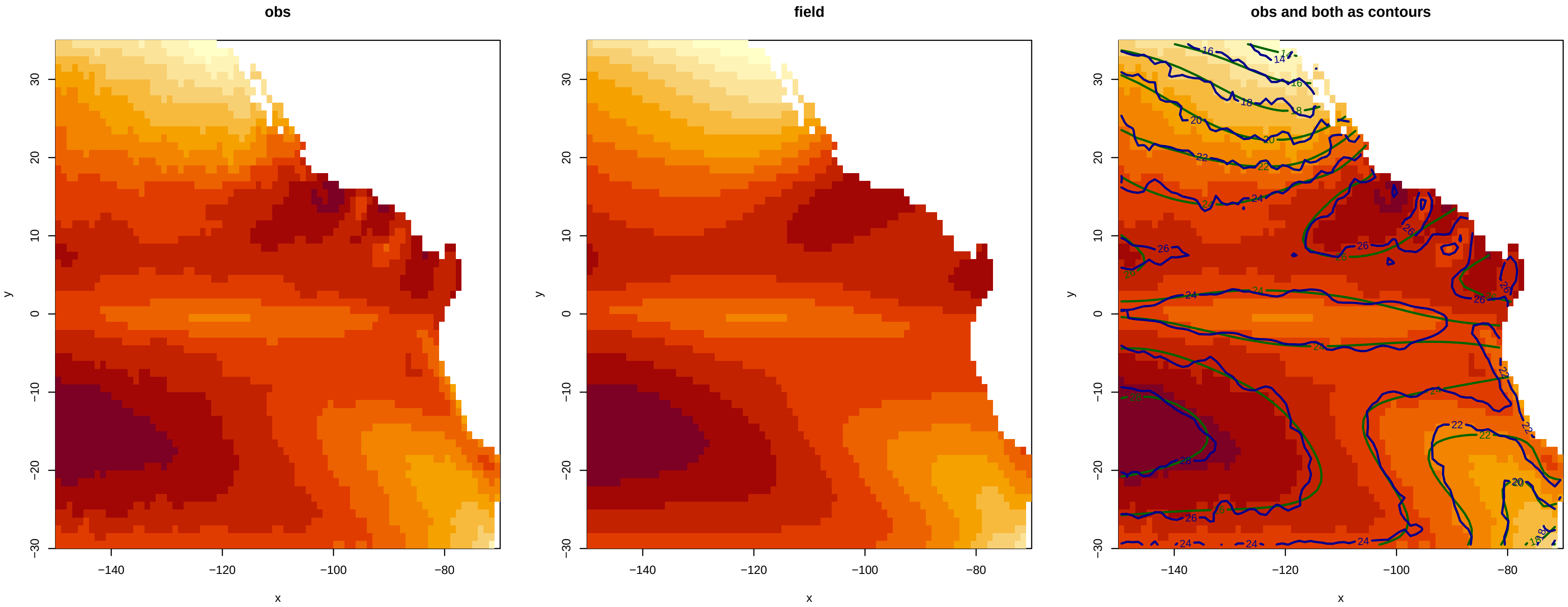
- Notice that next ψ depends on parameter values (via fields), so can switch to a new cell in grid ...
- Need a differentiable representation of each field

Differentiable representation of field

- Need to represent each field, so we evaluate anywhere $\alpha(x, y)$, differentiable in both x and y direction.
- Options: Splines, polynomials. neural network, (local regression), ...
- Splines and polynomials can be tricky to setup in 2D and can have erratic behavior near edges
- Tried first with a neural networks, but landed on local regression

Neural network representation

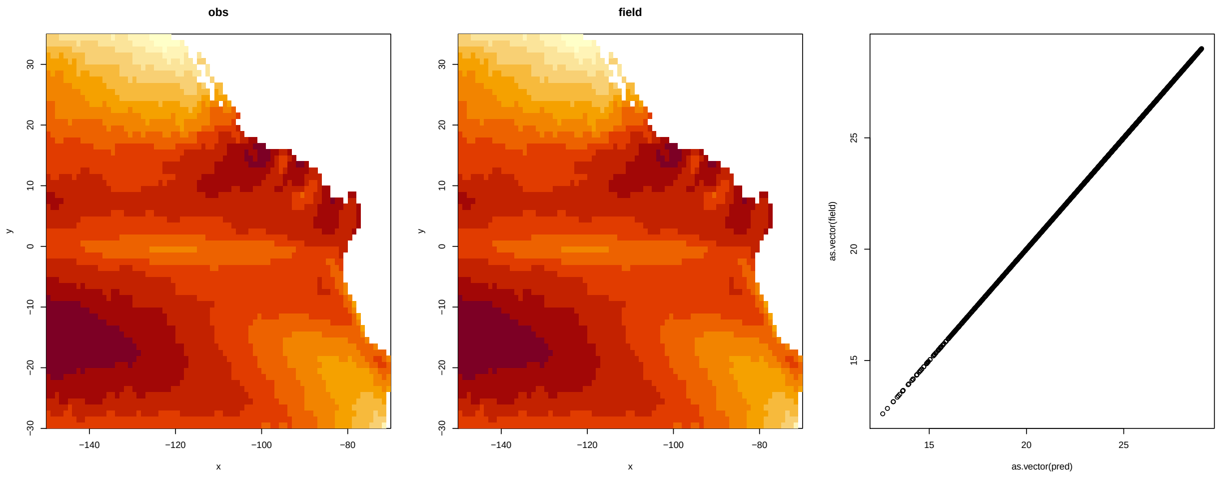
- A one layer[a] network was fitted ("trained") externally
- 3 inputs x, y, and 1. 15 nodes in the hidden layer gives. 60 model parameters
- When the network has been trained we can represent the field by these 60 parameters
- Differentible and can easily be calculated: $f(x,y) = v_1\phi(w_{1,1}x + w_{1,2}y + w_{1,3}) + \cdots + v_{15}\phi(w_{15,1}x + w_{15,2}y + w_{15,3})$



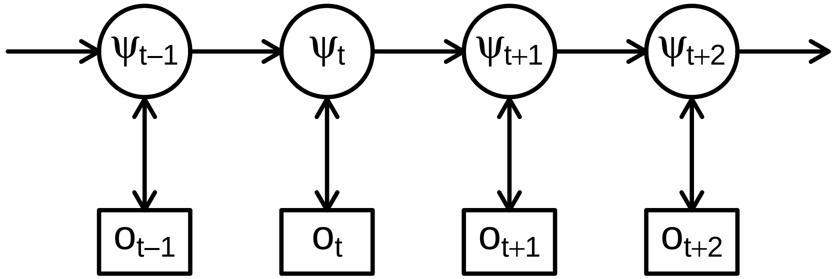
[a] two layer also tested

Local regression

- Use data as is
- Evaluate the field anywhere by weighted average of observations “nearby”
- This approach conflicted with a design principle in old TMB, but Kasper made it possible in new TMB!
- If we set the neaby-radius $R = 1$ (equal to distance between observations), then no smoothing is applied
- Nothing needs to be optimized
- The field is differentiable, but getting e.g. useful $\nabla h(x, y)$ needs some consideration
- Either apply smoothing, or define additional fields from Δx and Δy fields



Archival tags - now all OK



- Now we can do exactly what we want for archival tags
- Model:

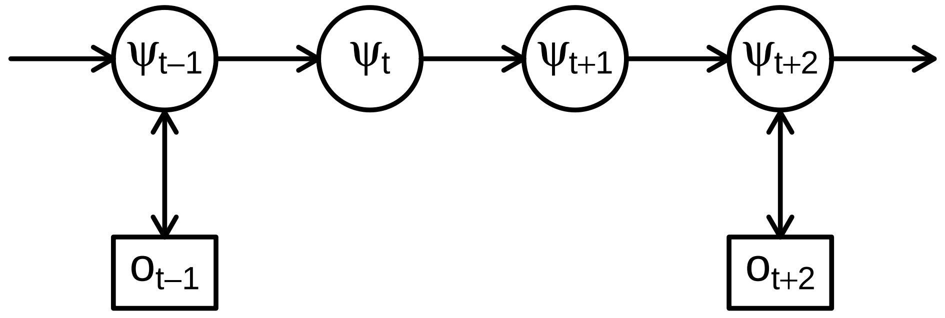
$$o_t \sim N(\psi_t, \Sigma_o)$$

$$\psi_{t+\Delta t} \sim N(\psi_t + \alpha(\psi_t, t)\Delta t, 2D(\psi_t, t)\Delta t I_{2\times 2})$$

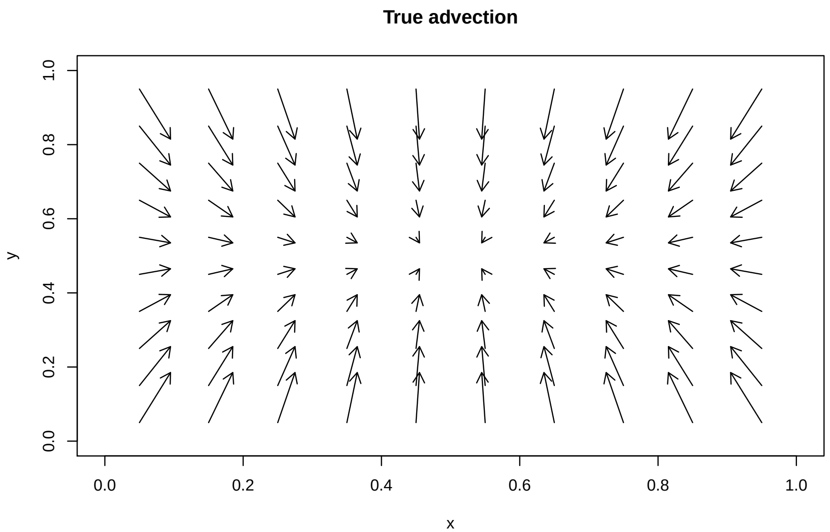
- Implemented via an old-style Kalman filter
- Fairly consistent with the matrix exponential approach for conventional tags, but notice that we did not need a grid for the archival tags...

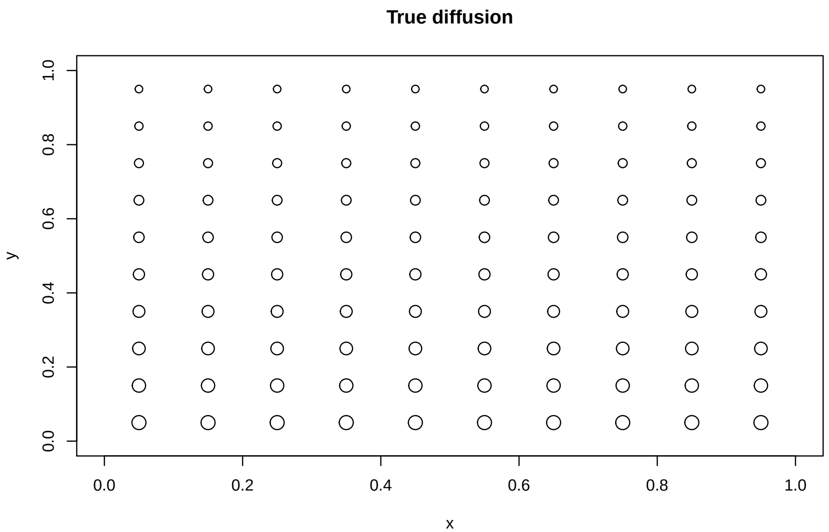
Conventional as archival

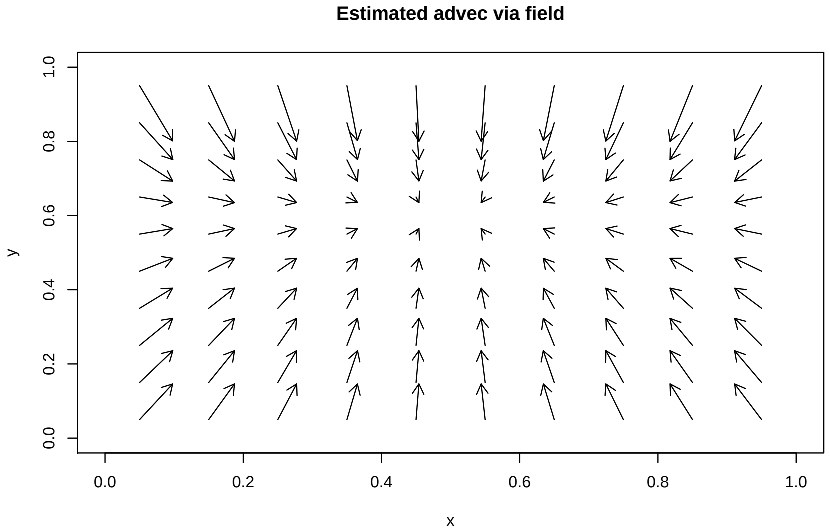
- Can we consider a conventional tag as an archival where all the observations between start and end are missing
- Yes! In a state-space model we can have time-steps without corresponding observations
- However, we need to decide how many steps we we want to resolve its path into
- After that a Kalman filter without data update is used
- Advantage: No grid needed
- Advantage: Only local calculations needed for each tag
- Advantage: Fast! This was actually a bit of a surprise
- Advantage: Calculations can be done in parallel
- Advantage: The movement matrix can be extracted if we need it (and at different resolutions)

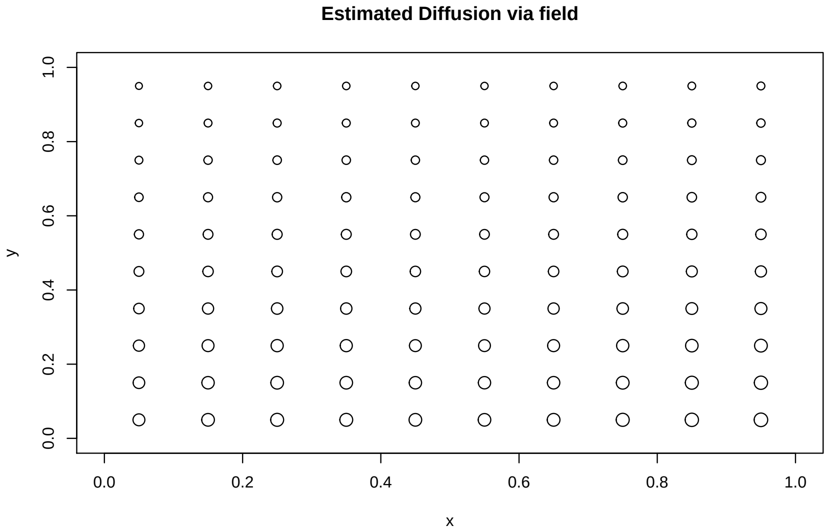


Simulation test (1000 conventional, 100 Archival)









Survival and catch

- A fish that is recaptured is surviving until that, so a history of:

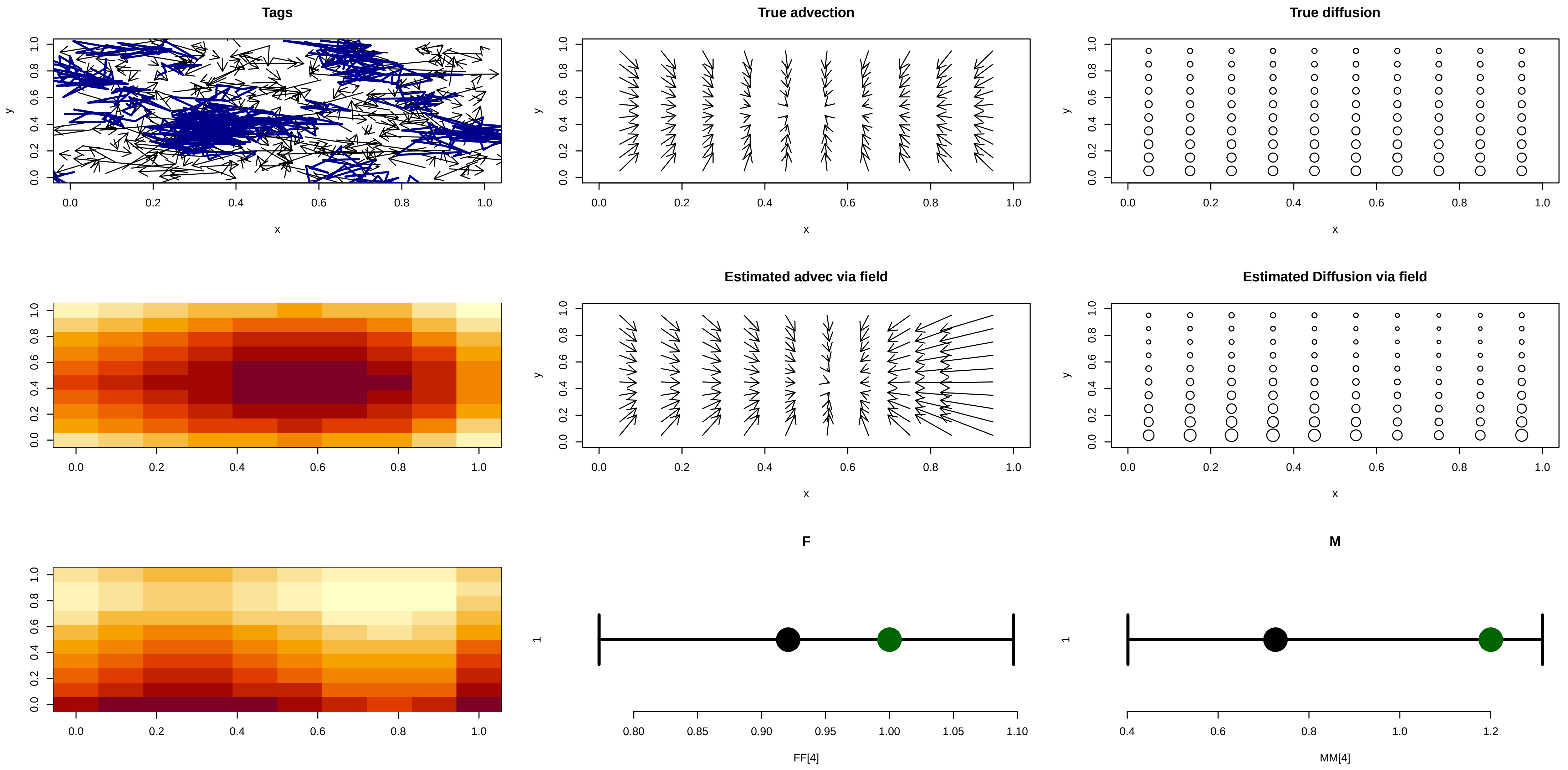
$$e^{-(M+F(x_1,y_1))\Delta t_1} e^{-(M+F(x_2,y_2))\Delta t_2} \cdots e^{-(M+F(x_{m-1},y_{m-1}))\Delta t_{m-1}} \frac{F(x_m,y_m)}{F(x_m,y_m)+M}(1-e^{-(M+F(x_m,y_m))\Delta t_m})$$

- A fish that is never recaptured is not caught first step, second step, ...

$$\begin{aligned}1-\Bigg(&\frac{F(x_1,y_1)}{F(x_1,y_1)+M}(1-e^{-(M+F(x_1,y_1))\Delta t_1})\\ &+e^{-(M+F(x_1,y_1))\Delta t_1}\frac{F(x_2,y_2)}{F(x_2,y_2)+M}(1-e^{-(M+F(x_2,y_2))\Delta t_2})\\ &+e^{-(M+F(x_1,y_1))\Delta t_1}e^{-(M+F(x_2,y_2))\Delta t_2}\frac{F(x_3,y_3)}{F(x_3,y_3)+M}(1-e^{-(M+F(x_3,y_3))\Delta t_3})\\ &+\cdots\ \text{(or truncated)}\Bigg)\end{aligned}$$

- Calculating these probabilities from the tracks is an approximation
- If focus was track's reconstructions, then these probabilities should be included in filtering

Simulation test (1000(283) Conventional, 100(21) Archival)



Residual for position (conventional tag)

- Can only calculate these if the tag is recaptured
- According to the model the position follows a 2D independent normal, so

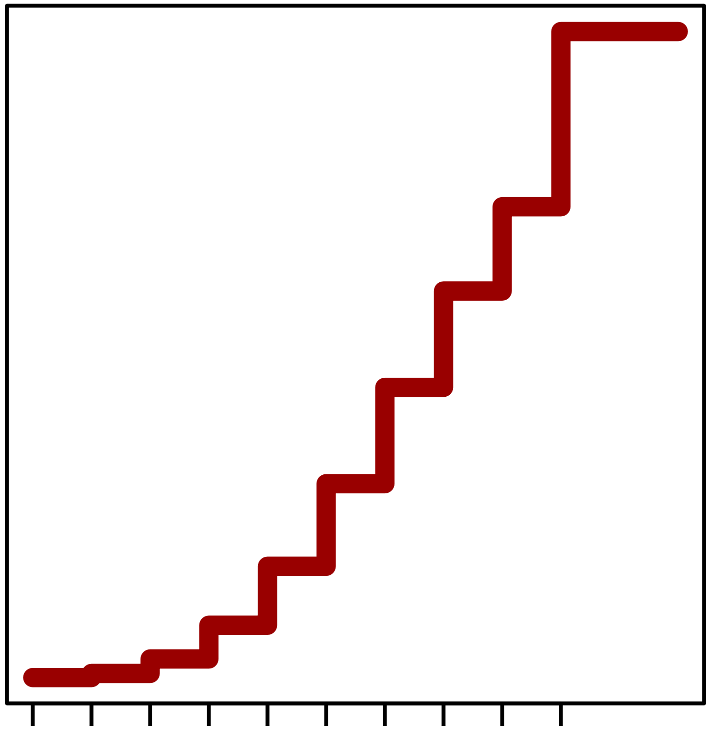
$$r_x = (x - \mu_x)/\sigma_x \quad \& \quad r_y = (y - \mu_y)/\sigma_y$$

  no problem.

Residual for recapture time (conventional tag)

- Model discrete in time, so recapture time is $t_1, t_2, \ldots, t_n$, or ‘never’
- According to the model we can calculate probability of each
- Use randomized quantile residuals:

$$r_t = \Phi^{-1}(U), \quad U \sim \text{unif}(P(T < t_{\text{obs}}), P(T \leq t_{\text{obs}}))$$



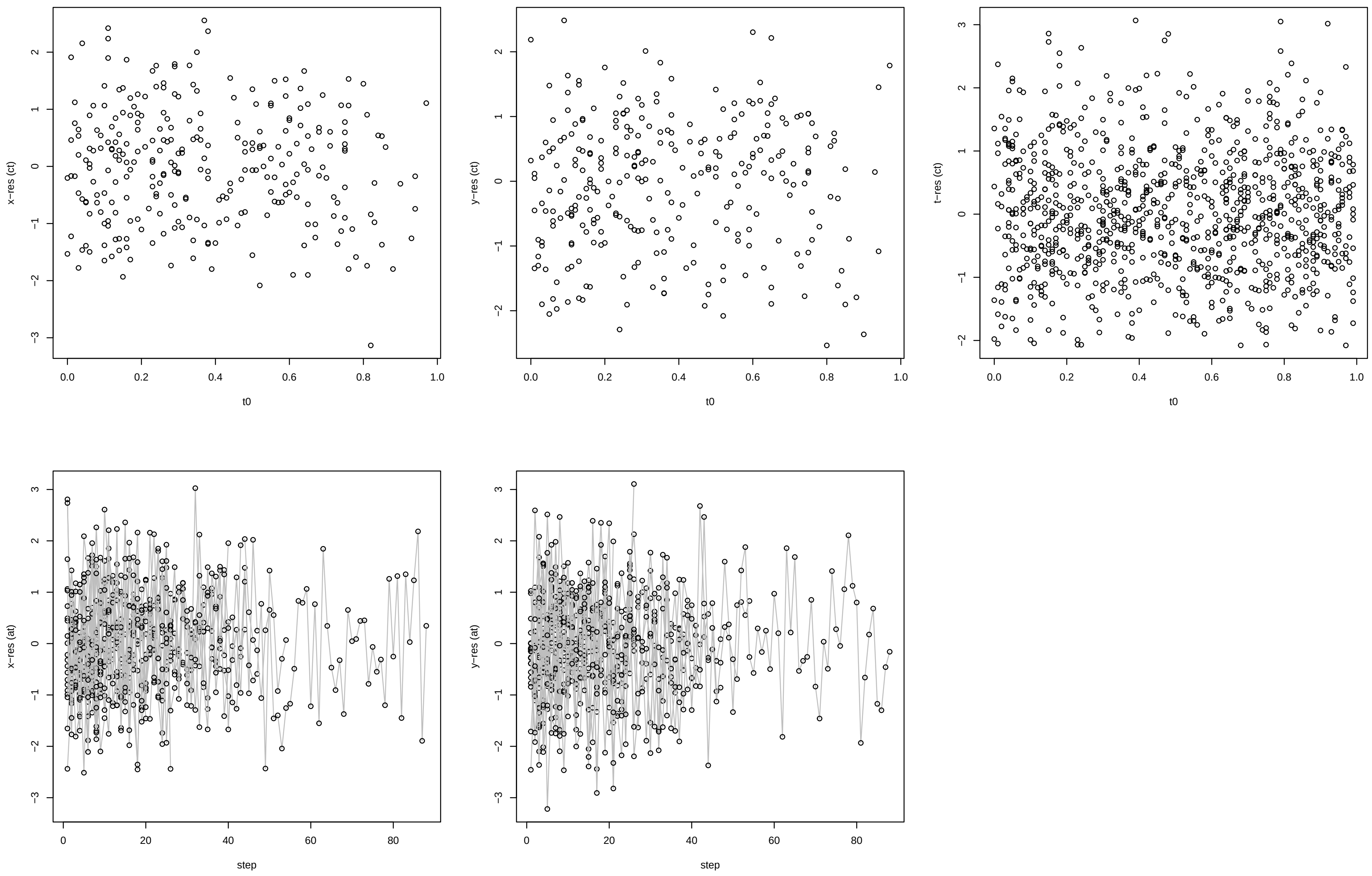
Residual for positions of archival tags

- Multiple residuals per tag
- Independent if we focus on prediction residuals

$$r_{x,i} = (x_i - \mu_{x_i|x_{i-}})/\sigma_{x_i|x_{i-}} \quad \& \quad r_{y,i} = (y_i - \mu_{y_i|y_{i-}})/\sigma_{y_i|y_{i-}}$$

- A bit challenging how we plot them

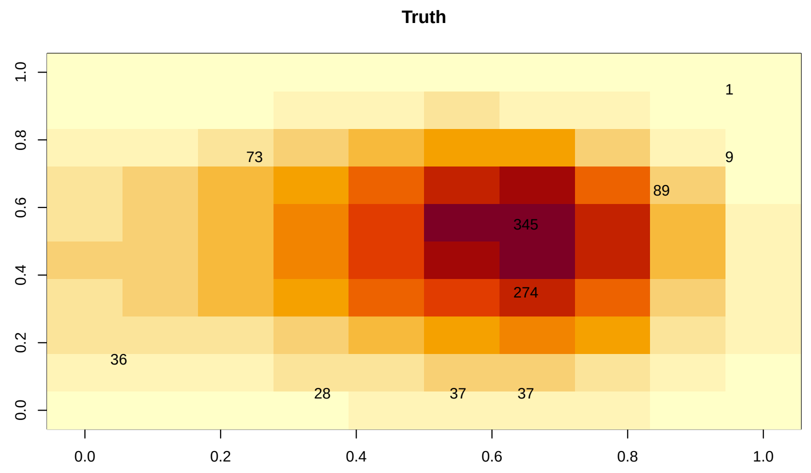
Residual example for the simulated case

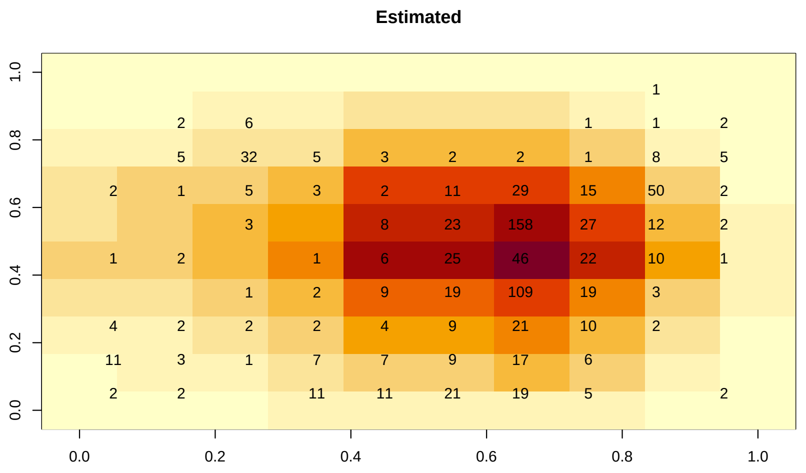


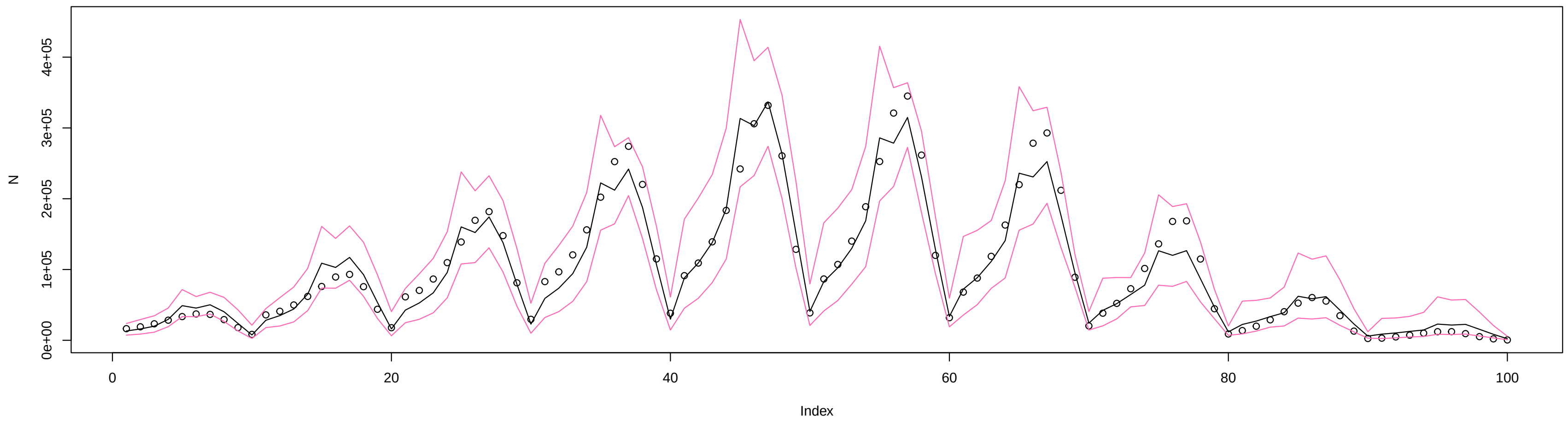
Exploration of Petersen type estimator

- Goal is to minimize reliance on the "effort" data
- We noticed that the movement pattern fairly constant with different effort patterns
- So can we treat movement as known, and use recovered tag-fraction to estimate abundance
- Condition on movement we can predict how many tags T_i will be in a given cell
- We know the catch taken C_i
- The expected nummber of tags observed will be $\frac{T_i}{N_i}C_i$
- So the observed number of tags may be $t_i \sim \text{Pois}\left(\frac{T_i}{N_i}C_i\right)$
- Coupled with a simple spatial model $N \sim \text{Gaussian}(\mu, \Sigma)$ (e.g. AR(1)×AR(1))
- Still very early, not all issues settled, but looks promising.

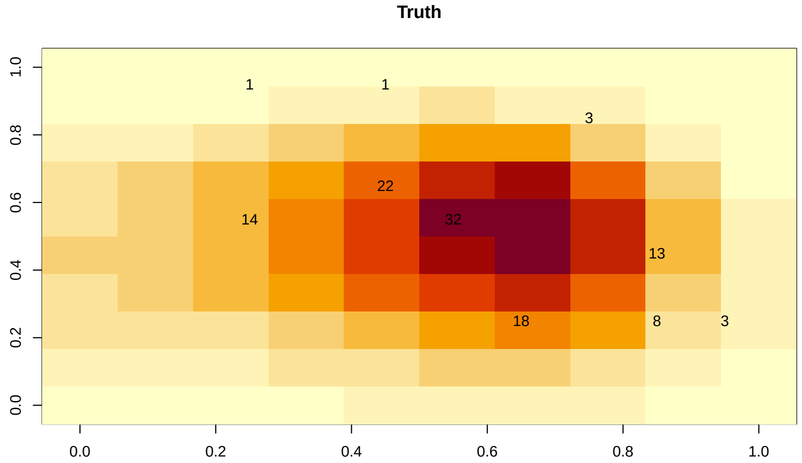
1000 tags released in 10 positions

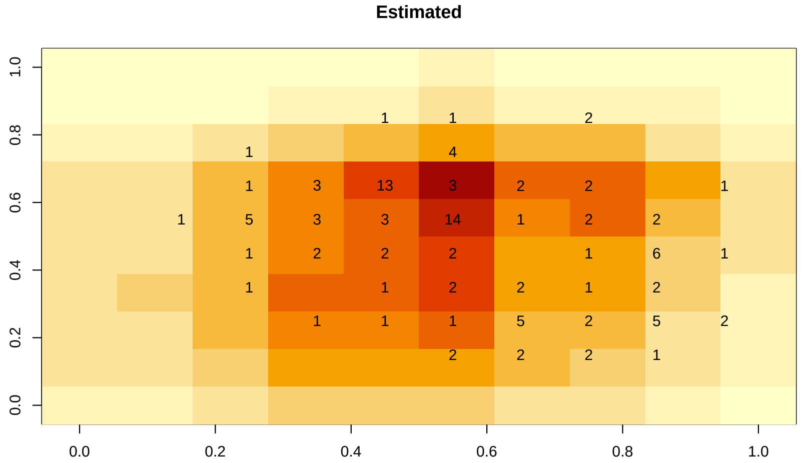


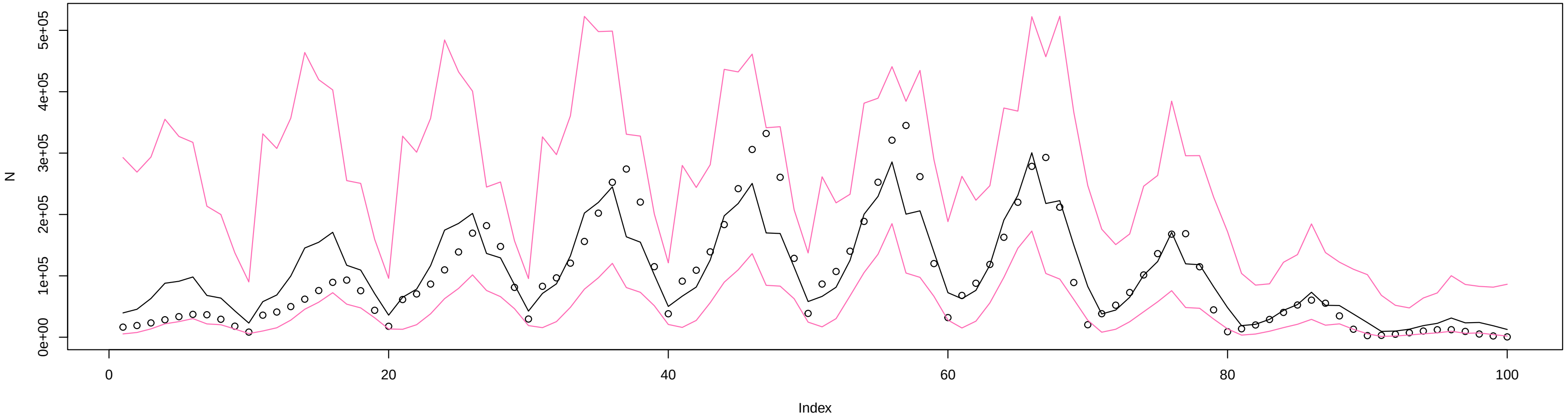




100 tags released in 10 positions







100 tags released in 5 positions