Geolocation of mobile devices

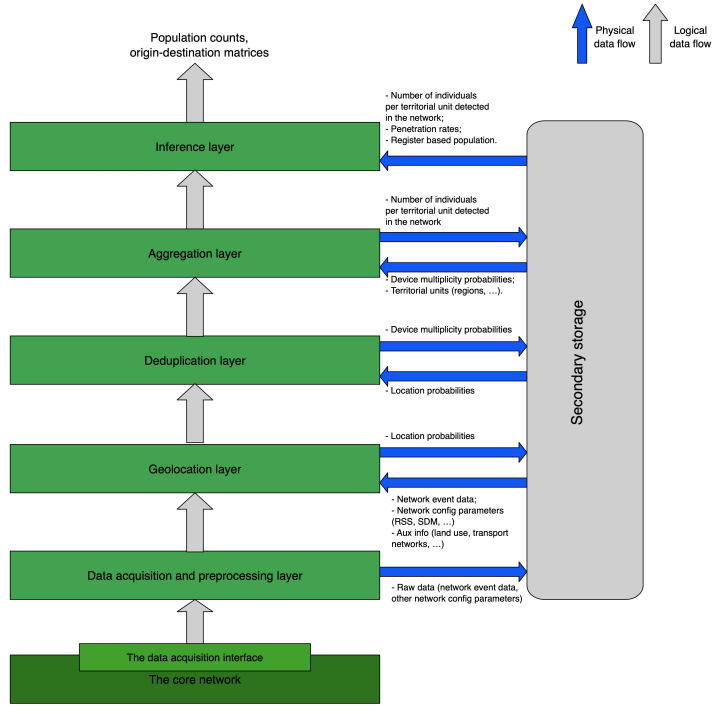
Luis Sanguiao

4/12/2020

Context

The European project ESSnet on Big Data II

- A modular statistical process was designed
- It goes from the raw telco data to the final target population estimates
- A set of R packages were developed to carry out the computations that this process requires
- *destim* package was developed for the geolocation layer



Why?

destim is just a Hidden Markov Model estimation package.
While some existing HMM estimation could have been used . . .

Why?

destim is just a Hidden Markov Model estimation package.
While some existing HMM estimation could have been used . . .

- The transition matrix is expected to be highly sparse. However, many HMM estimation packages are designed for dense transition matrices.

Why?

destim is just a Hidden Markov Model estimation package.
While some existing HMM estimation could have been used . . .

- The transition matrix is expected to be highly sparse. However, many HMM estimation packages are designed for dense transition matrices.

- Linear constraints are expected, otherwise the number of parameters would grow linearly with the area of the map. This suggests that Baum-Welch is not a good choice for maximize likelihood, but it is the main parameter estimator in many packages.

Why?

destim is just a Hidden Markov Model estimation package.
While some existing HMM estimation could have been used . . .

- The transition matrix is expected to be highly sparse. However, many HMM estimation packages are designed for dense transition matrices.

- Linear constraints are expected, otherwise the number of parameters would grow linearly with the area of the map. This suggests that Baum-Welch is not a good choice for maximize likelihood, but it is the main parameter estimator in many packages.

- *destim* is optimized for one specific kind of constraints: the equalities between two transition probabilities (*isotropy*). This is too specific and we don't know any other package with this characteristic.

Why?

destim is just a Hidden Markov Model estimation package.
While some existing HMM estimation could have been used . . .

- The transition matrix is expected to be highly sparse. However, many HMM estimation packages are designed for dense transition matrices.

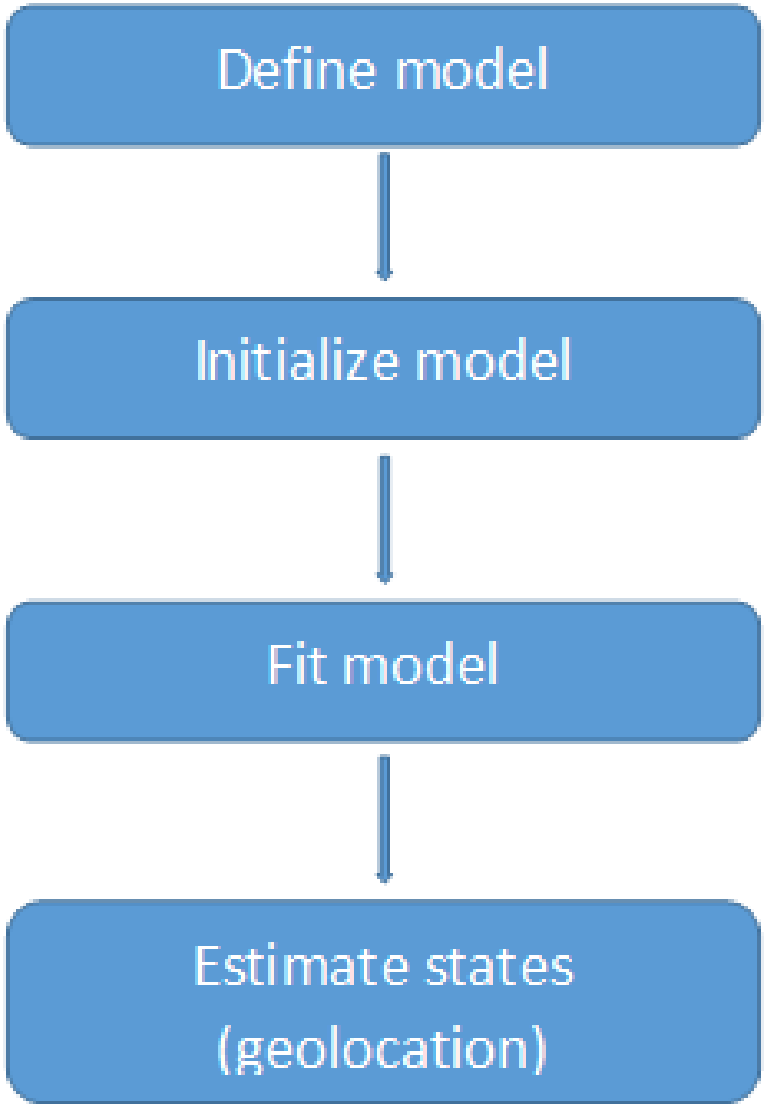
- Linear constraints are expected, otherwise the number of parameters would grow linearly with the area of the map. This suggests that Baum-Welch is not a good choice for maximize likelihood, but it is the main parameter estimator in many packages.

- *destim* is optimized for one specific kind of constraints: the equalities between two transition probabilities (*isotropy*). This is too specific and we don't know any other package with this characteristic.

- In general, it is an unusual HMM estimator, because it is designed for a high number of states but a small number of parameters.

How *destim* works

- The initialization finds transition probabilities that satisfy the constraints, and also computes a QR decomposition to find a minimal set of parameters.
- Rcpp and RcppEigen are used to improve performance, specially for sparse linear algebra computations.
- For likelihood optimization, constrained optimization packages are used: solnp and donlp2. It is also possible to use *constrOptim* from package stats.



Basic usage (I)

```
model <- HMM(5)
model <- addtransition(model,c(1,2))
model <- addtransition(model,c(2,3))
model <- addtransition(model,c(3,4))
model <- addtransition(model,c(4,5))
model <- addtransition(model,c(5,1))
model <- addconstraint(model,c(2,4))
emissions(model)<-matrix(c(0.3, 0.3, 0.7, 0.9, 0.9,
                           0.7, 0.7, 0.3, 0.1, 0.1),
                         nrow = 5, ncol = 2)
model <- initparams(model)
model <- minparams(model)
logLik(model,c(1L,2L))
## [1] 2.056399
model <- fit(model, c(1L, 2L), init = T, method = "constrOptim")
logLik(model,c(1L,2L))
## [1] 1.213004
```

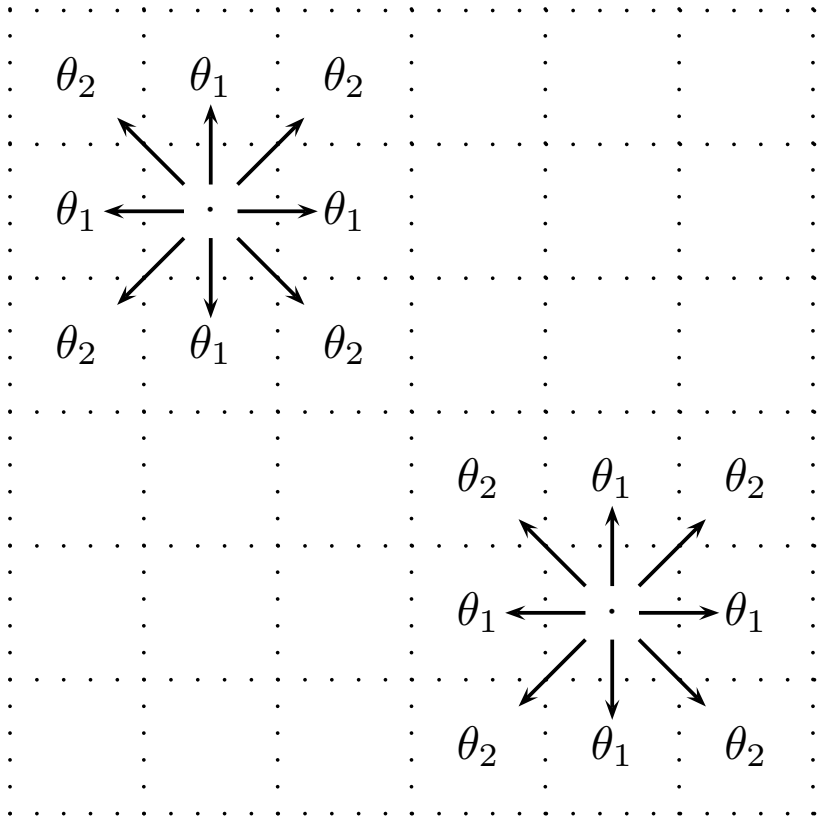
Basic usage (II)

```
sstates(model, c(1L, 2L))
## 5 x 2 sparse Matrix of class "dgCMatrix"
##
## [1,] 0.07884862 0.46667993
## [2,] 0.09784651 0.07884862
## [3,] 0.22716072 0.26821705
## [4,] 0.06279566 0.08818847
## [5,] 0.53334849 0.09806593
scpstates(model, c(1L,2L))
## 5 x 5 sparse Matrix of class "dgCMatrix"
##
## [1,] 9.685594e-10 7.884862e-02 .           .          .
## [2,] .            2.804493e-09 0.09784651 .          .
## [3,] .            .            0.17037054 0.05679018 .
## [4,] .            .            .          0.03139828 0.03139737
## [5,] 4.666799e-01 .            .          .          0.06666856
```

The *rectangle* model

- An isotropic model for a square grid.
- No matter how big the map is, it only has two parameters.
- It is an easy model, good for testing.

```
model <- HMMrectangle(100,100)
nstates(model)
## [1] 10000
ntransitions(model)
## [1] 88804
nconstraints(model)
## [1] 88802
```

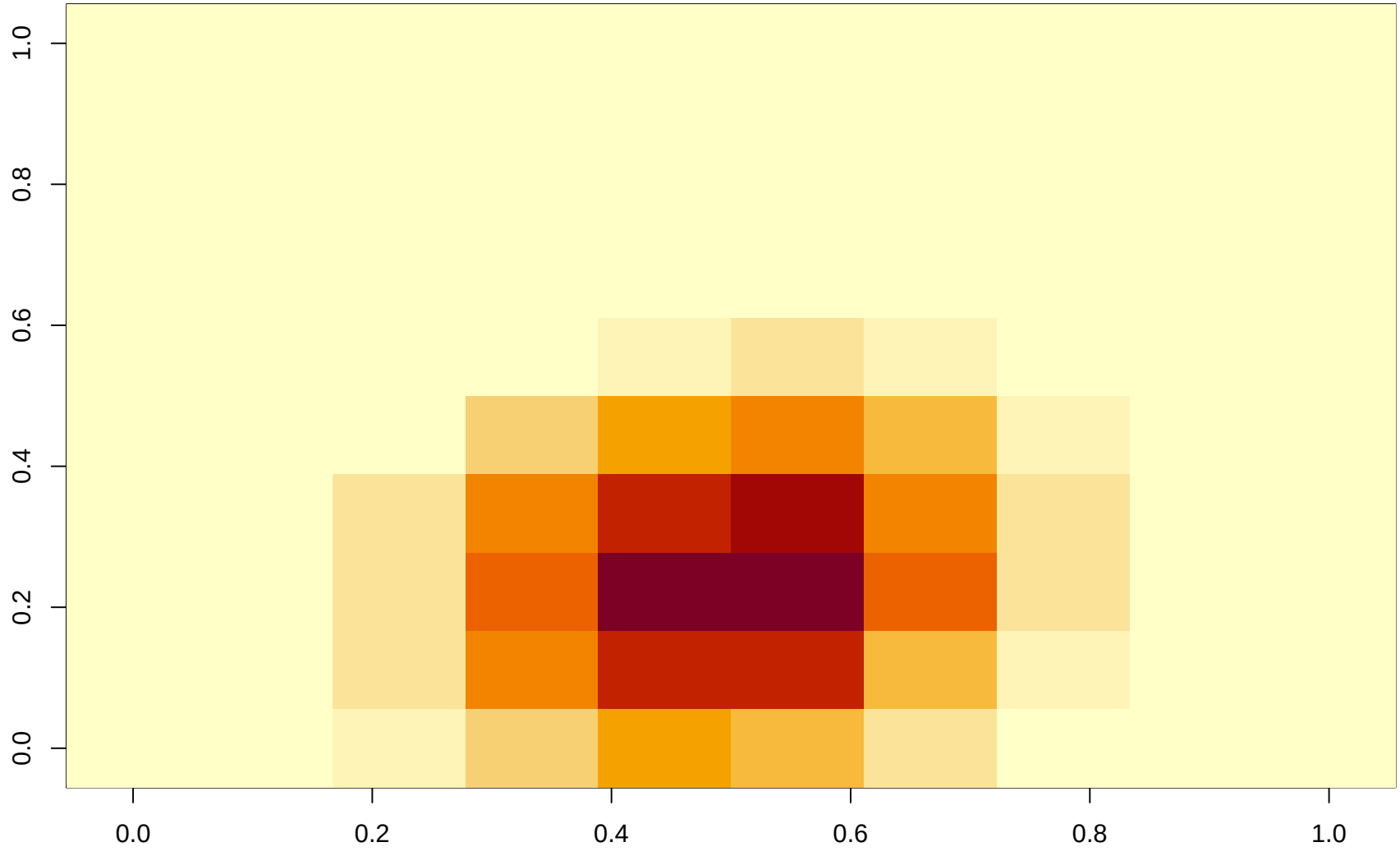


Example with the *rectangle* model (I)

```
model <- HMMrectangle(10,10)
tws <- matrix(c(3.2, 6.1, 2.2, 5.7, 5.9, 9.3, 5.4,
                4.0, 2.9, 8.6, 6.9, 6.2, 9.7, 1.3),
              nrow = 2, ncol = 7)
S <- function(x) if (x > 5) return(0) else return(20*log(5/x))
emissions(model)<-createEM(c(10,10), tws, S)
model <- initparams(model)
model <- minparams(model)
obs <- c(1,2,NA,NA,NA,NA,7,7)
model <- fit(model, obs)
```

Example with the *rectangle* model (II)

```
image(matrix(sstates(model, obs)[,4], ncol = 10))
```



Some remarks about computacional efficiency

- The package is expected to scale well for map size, because sparsity. Very long gaps between detection events might be problematic.
- A desktop computer can carry out the computations for a few devices in a small country. RAM size would still be a limitation though.
- As the number of devices to estimate is expected to be huge, it is an embarrassingly parallel problem.
- Keeping a small number of parameters makes also easier the task of the optimizer.

Thank you!