Supplementary material for “An end-to-end statistical process with mobile network data for Official Statistics”

David Salgado¹,²*, Luis Sanguiao¹†, Bogdan Oancea³†, Sandra Barragán¹† and Marian Necula²†

¹Correspondence: david.salgado.fernandez@ine.es
²Dept. Methodology and Development of Statistical Production, Statistics Spain (INE), Paseo de la Castellana, 183, Madrid, Spain
Full list of author information is available at the end of the article
†The views expressed in this working paper are those of the authors and do not necessarily reflect the views of their affiliating institutions.

1 Introduction
A more complete list of published articles with statistical uses of mobile network data is provided in the references [1–73]. A great deal of unpublished work can also be found. We specifically recommend the conference series NetMob (http://www.netmob.org).

2 Data description
2.1 Scenario data
All input and output data for the simulator can be found at [URL]. Input data for the simulator are specified as xml files (persons.xml, simulation.xml, antennas.xml) and a wkt file for the irregular polygon (territory map). For the scenario used in the article, we have selected:

- simulation.xml contains general parameters for the simulation: see table 1.
- persons.xml contains general parameters for the displacement patterns: see table 2.
- antennas.xml contains parameters to configure each antenna: see table 3. We have configured 70 antennas with the marginal distributions included in table 4.

Output data from the simulator are obtained in csv format (we comment only the basic ones):

- persons.csv contains the real evolution of each individual, i.e. the ground truth. For each time instant \( t \) and each individual \( k \), the simulator records the position coordinates \( x \) and \( y \), the tile and the device(s) carried by the individual.
- SignalMeasure_MNO1.csv contains for each antenna the RSS in each tile of the reference grid.
- AntennaInfo_MNO_MNO1.csv contains the time sequence of connections. For each time instant \( t \) and each device, the simulator records the antenna to which it is connected, its true position coordinates \( (x, y) \) and tile, and a network event code for the type of connection.

For details about other parameters and files see [74].

3 Geolocation of mobile devices
3.1 Model construction
We include the mathematical details to compute the posterior location probabilities from the input data. This is conducted in steps:
1 Time discretization and padding.
2 Construction of the transition model.
3 Construction of the emission model.
4 Construction of the initial state (prior) distribution.
5 Computation of the likelihood function.
6 Parameter estimation (likelihood maximization).
7 Application of the forward-backward algorithm.

3.1.1 Time discretization and padding
We shall work in discrete times. To do this we need to relate three parameters, namely (i) the tile dimension $l$ (we assume a square grid for simplicity), (ii) the time increment $\Delta t$ between two consecutive time instants, and (iii) an upper bound $v_{\text{max}}$ for the velocity of the individuals in the population. As we argued in the main text, we impose that in the time interval $\Delta t$, the device $d$ at most can displace from one tile to an adjacent tile. Under this condition, we can trivially set $\Delta t \lesssim \frac{l}{v_{\text{max}}}$. For example, if $v_{\text{max}} = 150\text{km/h} \approx 42\text{ms}^{-1}$, then $\Delta t \lesssim \frac{100}{42} \approx 2\text{s}$. Conversely, if the time increment $\Delta t$ is fixed, then the maximum distance in terms of the number of tiles will be $\lceil \frac{v_{\text{max}} \cdot \Delta t}{l} \rceil$, which expresses the number of time instants to insert in the time sequence to guarantee the maximum one-tile distance restriction.

If in the dataset the device $d$ is detected at longer time periods, e.g. once in a minute, then we artificially introduce missing values at intervals $\Delta t$ between every two observed values. This artificial non-response allows us to work with parsimonious models easier to estimate instead of using more complex transition matrices.

Notice that we have used an a priori value for $v_{\text{max}}$, but we can also possibly make an estimation using the observed values $E_d(t)$ and geometrical considerations about the respective coverage areas and their mutual distance.

Additionally, each observed time instance $t$ is approximated to its closest multiple integer of $\Delta t$. Thus, we will have as input data a sequence of time instants at multiples $t_n = \Delta t \cdot n, (n \geq 0)$ and a randomly alternate sequence of missing values and of observed event variables $E_{t_n}$ (hereafter for ease of notation we drop out any reference to mobile device $d$ since we are only focusing on one device).

3.1.2 Construction of the transition model
Now we specify a model for the transition between tiles (states) $\{T_i\}_{i=1,\ldots,N_T}$. For ease of explanation and notation, let us change the notation of each tile $T_i$ to a two-dimensional index $T_{(i,j)}$. Accordingly, each tile will be specified in this section by a pair of integer coordinates. The correspondence between both enumerations is arbitrary, but fixed once it has been chosen. We again assume time homogeneity for simplicity. Thus, $P(T_{(r,s)}|T_{(i,j)})$ will denote $P(T_{(r,s)}(t_n + \Delta t)|T_{(i,j)}(t_n))$ for any $t_n = 0, 1, \ldots$. We assume a square regular grid for simplicity.

Now, we make use of our preceding imposition by which an individual can at most reach an adjacent tile in time $\Delta t$. Thus,
\[
\mathbb{P}(T(r,s)|T(i,j)) = 0 \quad \max\{|r-i|, |s-j|\} \geq 2, \quad r, s, i, j = 1, \ldots, \sqrt{N_T}. \tag{1a}
\]

Now, we assume that we have no further auxiliary information to model these transitions and impose rectangular isotropic conditions:

\[
\begin{align*}
\mathbb{P}(T(i\pm 1,j)|T(i,j)) &= \theta_1 \quad i, j = 1, \ldots, \sqrt{N_T}, \tag{1b} \\
\mathbb{P}(T(i,j\pm 1)|T(i,j)) &= \theta_2 \quad i, j = 1, \ldots, \sqrt{N_T}. \tag{1c}
\end{align*}
\]

The last set of conditions is row-stochasticity:

\[
\begin{align*}
\sum_{r,s=1}^{N_T} \mathbb{P}(T(r,s)|T(i,j)) &= 1, \quad i, j = 1, \ldots, \sqrt{N_T}, \tag{1d} \\
\mathbb{P}(T(r,s)|T(i,j)) &\geq 0, \quad i, j, r, s = 1, \ldots, \sqrt{N_T}.
\end{align*}
\]

Now back to the original notation for tiles and using the usual notation for the transition matrix \( A = [a_{ij}] \), with \( a_{ij} = \mathbb{P}(T_j|T_i) \), conditions (1) amounts to having a highly sparse transition matrix \( A \) with up to 4 terms equal to \( \theta_1 \) and \( \theta_2 \) (each) per row and diagonal entries guaranteeing row-stochasticity.

For the generic case of a square grid with size \( N_T \), we have

\[
A(\theta_1, \theta_2) = \\
\begin{bmatrix}
D_1(\theta_1, \theta_2) & M(\theta_1, \theta_2) & O & O & \cdots & O \\
M(\theta_1, \theta_2) & D_2(\theta_1, \theta_2) & M(\theta_1, \theta_2) & O & \cdots & O \\
O & M(\theta_1, \theta_2) & D_2(\theta_1, \theta_2) & M(\theta_1, \theta_2) & \cdots & O \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
O & O & O & M(\theta_1, \theta_2) & D_2(\theta_1, \theta_2) & M(\theta_1, \theta_2) \\
O & O & O & O & M(\theta_1, \theta_2) & D_1(\theta_1, \theta_2)
\end{bmatrix},
\]

where
\[
D_1(\theta_1, \theta_2) = \begin{pmatrix}
1 - 2\theta_1 - \theta_2 & \theta_1 & 0 & 0 & \cdots & 0 \\
\theta_1 & 1 - 3\theta_1 - 2\theta_2 & \theta_1 & 0 & \cdots & 0 \\
0 & \theta_1 & 1 - 3\theta_1 - 2\theta_2 & \theta_1 & \cdots & 0 \\
0 & 0 & \theta_1 & 1 - 3\theta_1 - 2\theta_2 & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & 0 & \cdots & 1 - 2\theta_1 - \theta_2
\end{pmatrix}_{N_T \times N_T},
\]

\[
D_2(\theta_1, \theta_2) = \begin{pmatrix}
1 - 3\theta_1 - 2\theta_2 & \theta_1 & 0 & 0 & \cdots & 0 \\
\theta_1 & 1 - 4\theta_1 - 4\theta_2 & \theta_1 & 0 & \cdots & 0 \\
0 & \theta_1 & 1 - 4\theta_1 - 4\theta_2 & \theta_1 & \cdots & 0 \\
0 & 0 & \theta_1 & 1 - 4\theta_1 - 4\theta_2 & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & 0 & \cdots & 1 - 3\theta_1 - 2\theta_2
\end{pmatrix}_{N_T \times N_T},
\]

\[
M(\theta_1, \theta_2) = \begin{pmatrix}
\theta_1 & \theta_2 & 0 & 0 & \cdots & 0 \\
\theta_2 & \theta_1 & \theta_2 & 0 & \cdots & 0 \\
0 & \theta_2 & \theta_1 & \theta_2 & \cdots & 0 \\
0 & 0 & \theta_2 & \theta_1 & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & 0 & \cdots & \theta_1
\end{pmatrix}_{N_T \times N_T},
\]

\[
O = \begin{pmatrix}
0 & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & 0
\end{pmatrix}_{N_T \times N_T}.
\]

Notice that \(A(\theta_1, \theta_2)\) fulfills all restrictions (1). Indeed, in our proposed implementation, in order to seek future generalization, we will work with a generic block-tridiagonal matrix (2), where the restrictions (1a) leading to 0 have been included, and complemented with the rest of restrictions (1b)-(1d) in matrix form. Thus, we write \(C \cdot \text{vec}(\tilde{A}) = \mathbf{b}\), where \(\text{vec}(\tilde{A})\) stands for the non-null elements of \(A\) in vector form. The rows of \([C \ \mathbf{b}]\) encode each of the restrictions (1b), (1c), and (1d). For example, \(a_{12} = \theta_1\) and \(a_{21} = \theta_1\) produce a row like this

\[
C_i \cdot \text{vec}(\tilde{A}) = \begin{pmatrix}
\cdots & 0 & 1 & \cdots & \cdots & -1 & \cdots & \cdots \\
\cdots & a_{12} & \cdots & \cdots & a_{21} & \cdots & \cdots & \cdots \\
\end{pmatrix}^T = b_i = 0. \quad (3)
\]

Thus, in our software implementation to test this proposal, we have considered a block-tridiagonal matrix like (2) together with a set of linear restrictions of the form \(C \cdot \text{vec}(\tilde{A}) = \mathbf{b}\).

### 3.1.3 Construction of the emission model

The emission model is specified by the HMM emission probabilities \(b_{ik} = P(E_{i_n} = E_k | T_{i_n} = i)\), where \(E_k\) is a possible value for the observed event variables
$E_{tn}$ and $i$ denotes the tile index. We assume time homogeneity. This conditional probability is computed using the radio wave propagation model of our choice:

$$b_{ik}^{RSS} \propto \text{RSS}(d(E_k, T_i))$$  \hspace{1cm} (4) \\
$$b_{ik}^{SDM} \propto \text{SDM}(d(E_k, T_i))$$  \hspace{1cm} (5)$$

where $d(E_k, T_i)$ stands for the distance between the antenna generating the event $E_k$ and tile $T_i$. The proportional constant is fixed to normalize the probability functions.

Up to this point we have as input data the sequence of observed and missing values $a_{tn} \in \{0, 1, \ldots, N_A\}$ for $t_n = 0, 1, \ldots, T$. We already have the emission matrix $B$, too.

### 3.1.4 Construction of the initial state (prior) distribution

For illustrative purposes, we consider two choices: (i) uniform prior, i.e. $\pi_i^{\text{uniform}} = \frac{1}{N_T}$ and (ii) $\pi_i^{\text{network}} \propto \sum_k (\text{RSS}(d(E_k, T_i)))$ (where RSS is expressed in watts) or $\pi_i^{\text{network}} \propto \sum_k (\text{SDM}(d(E_k, T_i)))$, depending on the emission model.

### 3.1.5 Computation of the likelihood

The likelihood is trivially computed using the numerical proviso of setting emission probabilities equal to 1 when there is a missing value in the observed variables (e.g. due to time padding). The general expression for the likelihood is

$$L(E) = \sum_{i_0=1}^{N_T} \cdots \sum_{i_T=1}^{N_T} \prod_{n=1}^{N} \mathbb{P}(T_{t_n} = i_n | T_{t_{n-1}} = i_{n-1}) \mathbb{P}(E_{tn} | T_{tn} = i_n)$$

Notice that the emission probabilities only contribute numerically providing no parameter whatsoever to be estimated.

### 3.1.6 Parameter estimation

The estimation of the unknown parameters $\theta$ is conducted maximizing the likelihood. The restrictions coming from the transition model (1) makes the optimization problem not trivial. Notice that the EM algorithm is not useful. Instead, we provide a taylor-made solution seeking for future generalizations with more realistic choices of transition probabilities incorporating land use information. Formally, the optimization problem is given by:

$$\max \quad h(a)$$

s.t. \hspace{0.5cm} $C \cdot a = b$

$$a_k \in [0, 1],$$  \hspace{1cm} (7)
where \( \mathbf{a} \) stands for the nonnull entries of the transition probability matrix \( \mathbf{A} \), the objective function \( h(\mathbf{a}) \) is derived from the likelihood \( L \) expressed in terms of the nonnull entries of the transition matrix \( \mathbf{A} \), and the system \( \mathbf{C} \cdot \mathbf{a} = \mathbf{b} \) expresses the sets of restrictions from the transition model (1) not involving null rhs terms (restrictions (1b), (1c), and (1d)).

Let us quantify the number of variables and restrictions in order to propose an abstract procedure possibly generalized to other situations. We illustrate this procedure for a square regular grid of size \( N_T \). The number of zeroes in the transition matrix \( \mathbf{A} \) can be computed as follows:

- There exist 4 rows in \( \mathbf{A} \) corresponding to the 4 vertices in the grid. Each of these rows contains \( N_T - 4 \) zeroes.
- There exist 4 sets of \( \sqrt{N_T} - 2 \) rows in \( \mathbf{A} \) corresponding to boundary tiles not being vertices. Each of these rows contains \( N_T - 6 \) zeroes.
- There exist \( (\sqrt{N_T} - 2)^2 \) rows in \( \mathbf{A} \) corresponding to this same number of inner tiles. Each of these rows contains \( N_T - 9 \) zeroes.

Thus, the total number of zeroes in \( \mathbf{A} \) is given by \( 4 \times (N_T - 4) + 4 \times (\sqrt{N_T} - 2) \times (N_T - 6) + (\sqrt{N_T} - 2)^2 \times (N_T - 9) = N_T^2 - 9 \cdot N_T + 12\sqrt{N_T} - 4 \). The number of non-null components of \( \mathbf{a} \) in problem (7) is \( d = 9 \cdot N_T - 12\sqrt{N_T} + 4 \).

The number of restrictions \( n_r \) not involving zeroes depends very sensitively on the particular transition model chosen for the displacements. In the rectangular isotropic model considered above, we need to identify the number of entries (i) equal to \( \theta_1 \), (ii) equal to \( \theta_2 \), and (iii) in the diagonal (thus guaranteeing the row-stochasticity restriction). Using the same counting procedure as above, the number of entries equal to \( \theta_1 \) will be given by \( 4 \times 2 + 4 \times (\sqrt{N_T} - 2) \times 3 + (\sqrt{N_T} - 2)^2 \times 4 = 4 \cdot N_T - 4\sqrt{N_T} - 1 \) rows. For \( \theta_2 \), we get \( 4 \times (\sqrt{N_T} - 1)^2 - 1 \) rows. From the row-stochasticity restriction we get \( N_T \) rows. Thus, the matrix \( \mathbf{C} \) will have dimensions \( n_r \times d \), with \( n_r = 4 \cdot N_T - 4\sqrt{N_T} - 1 + 4 \times (\sqrt{N_T} - 1)^2 - 1 + N_T = 9 \cdot N_T - 12\sqrt{N_T} + 2 \). Notice that \( d - n_r = 2 \), as expected, since we have two free parameters \( \theta_1 \) and \( \theta_2 \).

The abstract optimization problem is thus

\[
\begin{align*}
\text{max} \quad & h(\mathbf{a}) \\
\text{s.t.} \quad & \mathbf{C} \cdot \mathbf{a} = \mathbf{b} \\
& \mathbf{a} \in [0,1]^d,
\end{align*}
\]

where \( \mathbf{C} \in \mathbb{R}^{n_r \times d} \) and \( \mathbf{b} \in \mathbb{R}^d \). The objective function \( h(\mathbf{a}) \) is indeed a polynomial in the non-null entries \( \mathbf{a} \). This problem can be further simplified using the matrix QR decomposition. Write \( \mathbf{C} = \mathbf{Q} \cdot \mathbf{R} \), where \( \mathbf{Q} \) is an orthogonal matrix of dimensions \( n_r \times n_r \) and \( \mathbf{R} \) is an upper triangular matrix of dimensions \( n_r \times d \). Then we can rewrite the linear system as \( \mathbf{R} \cdot \mathbf{a} = \mathbf{Q}^T \cdot \mathbf{b} \) and we can linearly solve variables \( a_1, \ldots, a_n_r \) in terms of variables \( a_{n_r+1}, \ldots, a_d \):

\[
\begin{pmatrix} a_1 & \cdots & a_{n_r} \end{pmatrix}^T = \tilde{\mathbf{C}}_{n_r \times (d-n_r)} \begin{pmatrix} a_{n_r+1} & \cdots & a_d \end{pmatrix}^T.
\]
The system (8) then reduces to

$$\begin{align*}
\max & \quad \tilde{h}(a_{n_r+1}, \ldots, a_d) \\
\text{s.t.} & \quad 0 \leq \tilde{C} \cdot \begin{pmatrix} a_{n_r+1} & \cdots & a_d \end{pmatrix}^T \leq 1.
\end{align*}$$

(9)

In our current software implementation we resort to general-purpose optimizers. It remains for future work to find an optimised algorithm to solve (9). The solution $\mathbf{a}^*$ to problem (8) will be introduced in the transition probability matrix, which will thus be denoted by $\tilde{\mathbf{A}}$.

3.1.7 Application of the forward-backward algorithm

Once the HMM has been fitted, we can readily apply the well-known forward-backward algorithm [see e.g. 75] to compute the target location probabilities $\gamma_{it}$ and $\gamma_{tij}$. No novel methodological content is introduced at this point. For our implementation, we have used the scaled version of the algorithm (see [75]).

3.1.8 Software implementation

To carry out the computation described above upon the synthetic scenario generated by the network event data simulator we have used the prototyping R package called `destim` developed for these purposes by the authors [76]. This package contains a specific implementation of the rectangular geolocation model described in the preceding sections.

3.2 Model evaluation

The center of location probabilities and the root mean squared dispersion can be obtained naturally from a bias-variance decomposition of a mean squared distance. Let us denote by $\mathbf{R}_{dt} \in \{\mathbf{r}_i^{(c)}\}_{i=1,\ldots,N_T}$ the random vector for the position of a device according to the distribution of posterior location probabilities $\gamma_{dti}^{(c)}$. Let us shortly denote $\mathbf{R}_{dt} = E \mathbf{R}_{dt} = \sum_{i=1}^{N_T} \gamma_{dti} \mathbf{r}_i^{(c)}$. Let us also denote the true position of device $d$ at time $t$ by $\mathbf{r}_{dt}^*$. Then, we can decompose

$$\begin{align*}
\text{msd}_{dt} & = E \|\mathbf{R}_{dt} - \mathbf{r}_{dt}^*\|^2 \\
& = E \| (\mathbf{R}_{dt} - \bar{\mathbf{R}}_{dt}) + (\bar{\mathbf{R}}_{dt} - \mathbf{r}_{dt}^*) \|^2 \\
& = E \left[ (\mathbf{R}_{dt} - \bar{\mathbf{R}}_{dt}, \mathbf{R}_{dt} - \bar{\mathbf{R}}_{dt}) \right] + \\
& \quad 2 \cdot E \left[ (\mathbf{R}_{dt} - \bar{\mathbf{R}}_{dt}, \bar{\mathbf{R}}_{dt} - \mathbf{r}_{dt}^*) \right] + \\
& \quad E \left[ (\bar{\mathbf{R}}_{dt} - \mathbf{r}_{dt}^*, \bar{\mathbf{R}}_{dt} - \mathbf{r}_{dt}^*) \right] \\
& = \text{rmsd}_{dt}^2 + \text{msd}_{dt}^2.
\end{align*}$$

(10)

This decomposition motivates the definition of the figures of merit proposed in the main text. We can also compare directly the mean squared distance (see figure 1). The overall performance is similar for the four models.
4 Device duplicity 

4.1 The double-device emission model 

To apply formulas for the computation of the device duplicity probabilities we need to compute the likelihood for the HMM model described above for each device separately and for each pair of devices according to figure 7 in the main text. To do this, we just need to have a new emission model producing a double event data sequence. The emission probabilities in this augmented model are computed using the original emission probabilities:

\[ P(E_{dt}, E_{d't} | T_{dt}, \Gamma^{aux}) = P(E_{dt} | T_{dt}, \Gamma^{aux}) \cdot P(E_{d't} | T_{d't}, \Gamma^{aux}) \]  

(11)

Once these emission probabilities are computed, the computation of the likelihoods \( \ell_{dd'} \) runs similar to the single-device case.

The computation depends on prior choice of the parameters \( \lambda_d \), i.e. the ratio between the prior probability of no duplicity to the prior probability of duplicity. For the computation in the main text, this was chosen according to the parameters in the network event data simulator. In practice, this is not the case, but the MNO can provide a prior estimation of the number of subscribers with more than one device. In any case, we run the computation of \( P_d^{(2)} \) for all \( d \) and checked the number of true/false positive/negative cases obtained. This is represented in figure 2, where we observed that around the chosen value, the classification is robust.

4.2 Software implementation 

To carry out the computation described above upon the synthetic scenario generated by the network event data simulator we have used the prototyping R package called deduplication developed for these purposes by the authors [77]. This package implements the computation of the device duplicity probabilities described in the main text, including the computation of the double-device emission model for the underlying HMM. This package contains another two deduplication procedures based on pairwise comparisons and trajectory comparisons. In this work we have included only the alternative producing the best disambiguating method on our scenario.

5 Statistical filtering 

To apply the proposed trajectory indicators to the synthetic scenario generated by the network event data simulator we have profusely used the R package trajr [78], with slight modifications on some functions to adequate to our trajectories.

6 Aggregation of individuals detected by a network 

The core of the aggregation module is the generation of random multidimensional variates according to the Poisson-multinomial distribution as a sum of categorical (multinoulli) variables. This is directly implemented in the prototyping R package called aggregation developed for these purposes by the authors [79]. This package takes as input both the posterior location probabilities \( \gamma_{dts} \), the device duplicity
probabilities $p_d^{(2)}$ for all devices $d$, and the spatial aggregation of tiles $i$ into larger territorial units and produce $n$ random multidimensional variates according to the Poisson-multinomial distribution defined by equation (17) in the main text. The package also implements the similar computation for the origin-destination matrix according to equation (21) in the main text.

7 Inference

The different models proposed for the inference module have been directly implemented using standard distributions in base R and package extraDistr [80], except for the continuous mixtures integrating the full hierarchy of levels for the observation and/or the state processes. The credible interval computations, both for the inference and the aggregation module, have been carried out using the R package bayestestR [81]. All credible intervals included in this work are high-density intervals [see e.g. 82].

Declaration

Availability of data and materials
Data, scripts, and source code are freely available at [URL].

Competing interests
The authors declare that they have no competing interests.

Funding
This work is part of ongoing projects at Statistics Spain (INE) and Statistics Romania (INS) in joint collaboration with the European Statistical System under Grant Agreement Number 847375-2018-NL-BIGDATA (ESSnet on Big Data II).

Authors’ contributions
All authors have contributed equally.

Acknowledgements
The authors acknowledge M.Á. Martínez-Vidal, S. Lorenzo, M. Suárez-Castillo, R. Radini, T. Tuoto, M. Offermans, M. Tennekes, S. Hadam, and F. Ricciato for invaluable insights and debates.

Author details

References


**Figures**

Figure 1 *Mean squared distance.* (Top) Distribution of mean squared distances $\text{msd}_{dt}$ for models RSS and SDM with uniform and network priors. (Middle) Time evolution of distributions of mean squared distances $\text{msd}_{dt}$ for the same models. (Bottom) Distribution per device of mean square distance $\text{msd}_{dt}$ for all times $t$ for the same models.
Figure 2 Device duplicity classification for different $\lambda$. Number of true/false positive/negative cases in the classification of device duplicity for different values of the prior ratio $\lambda_d = \lambda$ for all devices $d$. The red vertical lines represent the value used in the main text.

Tables

Table 1 Simulation parameters. Generic parameters included in simulation.xml.

<table>
<thead>
<tr>
<th>Time (s)</th>
<th>MNO</th>
<th>Others</th>
</tr>
</thead>
<tbody>
<tr>
<td>start_time</td>
<td>0</td>
<td>displacement random walk w/ drift</td>
</tr>
<tr>
<td>end_time</td>
<td>900</td>
<td>connection_type strength</td>
</tr>
<tr>
<td>time_increment</td>
<td>10</td>
<td>connection_threshold -85dBm</td>
</tr>
<tr>
<td>time_stay</td>
<td>20</td>
<td>grid_tile_dimensions 250 m × 250 m</td>
</tr>
<tr>
<td>interval_btw_stays</td>
<td>120</td>
<td></td>
</tr>
</tbody>
</table>

Table 2 Persons parameters. Parameters included in persons.xml (not exhaustive).

<table>
<thead>
<tr>
<th>Persons</th>
</tr>
</thead>
<tbody>
<tr>
<td>num_persons</td>
</tr>
<tr>
<td>speed_walk</td>
</tr>
<tr>
<td>speed_car</td>
</tr>
</tbody>
</table>

Table 3 Antennas parameters. Parameters per antenna included in antennas.xml.

<table>
<thead>
<tr>
<th>Antenna</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNO_name</td>
</tr>
<tr>
<td>max_connections</td>
</tr>
<tr>
<td>power</td>
</tr>
<tr>
<td>attenuationfactor</td>
</tr>
<tr>
<td>type</td>
</tr>
<tr>
<td>$S_{\text{min}}$ (thrsh_RSS)</td>
</tr>
<tr>
<td>$Q_{\text{min}}$ (thrsh_SDM)</td>
</tr>
<tr>
<td>$S_{\text{mid}}$</td>
</tr>
<tr>
<td>$S_{\text{steep}}$</td>
</tr>
<tr>
<td>coords</td>
</tr>
</tbody>
</table>
Table 4 Antenna configuration parameters. Marginal distributions of network configuration parameters included in antenna.xml.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>min</th>
<th>q1</th>
<th>q2</th>
<th>mean</th>
<th>q3</th>
<th>max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power (W)</td>
<td>5.000</td>
<td>10.000</td>
<td>10.000</td>
<td>9.574</td>
<td>10.000</td>
<td>10.000</td>
</tr>
<tr>
<td>Path Loss</td>
<td>3.800</td>
<td>3.900</td>
<td>3.900</td>
<td>3.939</td>
<td>4.000</td>
<td>4.000</td>
</tr>
<tr>
<td>Radius CoverArea (m)</td>
<td>1121.353</td>
<td>1333.521</td>
<td>1530.999</td>
<td>1483.766</td>
<td>1603.719</td>
<td>1947.483</td>
</tr>
<tr>
<td>S_{steep}</td>
<td>0.500</td>
<td>0.900</td>
<td>0.900</td>
<td>0.959</td>
<td>0.900</td>
<td>3.000</td>
</tr>
<tr>
<td>S_{mid} (dBm)</td>
<td>-94.000</td>
<td>-80.000</td>
<td>-80.000</td>
<td>-80.871</td>
<td>-79.000</td>
<td>-76.000</td>
</tr>
</tbody>
</table>