twinSIR: Individual-level epidemic modeling for a fixed population with known distances

Sebastian Meyer*  
Friedrich-Alexander-Universität Erlangen-Nürnberg

Leonhard Held  
University of Zurich

Michael Höhle  
Stockholm University

Abstract

The availability of geocoded health data and the inherent temporal structure of communicable diseases have led to an increased interest in statistical models and software for spatio-temporal data with epidemic features. The R package surveillance can handle various levels of aggregation at which infective events have been recorded. This vignette illustrates the analysis of individual-level surveillance data for a fixed population, of which the complete SIR event history is assumed to be known. Typical applications for the multivariate, temporal point process model “twinSIR” of Höhle (2009) include the spread of infectious livestock diseases across farms, household models for childhood diseases, and epidemics across networks. We first describe the general modeling approach and then exemplify data handling, model fitting, and visualization for a particularly well-documented measles outbreak among children of the isolated German village Hagelloch in 1861.

Keywords: individual-level surveillance data, endemic-epidemic modeling, infectious disease epidemiology, self-exciting point process, branching process with immigration.

1. Model class: twinSIR

The spatio-temporal point process regression model “twinstim” (Meyer, Elias, and Höhle 2012, illustrated in vignette("twinstim")) is indexed in a continuous spatial domain, i.e., the set of possible event locations consists of the whole observation region and is thus infinite. In contrast, if infections can only occur at a known discrete set of sites, such as for livestock diseases among farms, the conditional intensity function (CIF) of the underlying point process formally becomes \( \lambda_i(t) \). It characterizes the instantaneous rate of infection of individual \( i \) at time \( t \), given the sets \( S(t) \) and \( I(t) \) of susceptible and infectious individuals, respectively (just before time \( t \)). Höhle (2009) proposed the following endemic-epidemic multivariate temporal point process model (“twinSIR”):

\[
\lambda_i(t) = \lambda_0(t) \nu_i(t) + \sum_{j \in I(t)} \left\{ f(d_{ij}) + w_{ij}^T \alpha^{(w)} \right\},
\]

if \( i \in S(t) \), i.e., if individual \( i \) is currently susceptible, and \( \lambda_i(t) = 0 \) otherwise. The rate

*Author of correspondence: seb.meyer@fau.de
decomposes into two components. The first, endemic component consists of a Cox proportional hazards formulation containing a semi-parametric baseline hazard $\lambda_0(t)$ and a log-linear predictor $\nu_i(t) = \exp\left(z_i(t)^\top \beta\right)$ of covariates modeling infection from external sources. Furthermore, an additive epidemic component captures transmission from the set $I(t)$ of currently infectious individuals. The force of infection of individual $i$ depends on the distance $d_{ij}$ to each infective source $j \in I(t)$ through a distance kernel $f(u) = \sum_{m=1}^M \alpha_m^{(f)} B_m(u) \geq 0$, (2)

which is represented by a linear combination of non-negative basis functions $B_m$ with the $\alpha_m^{(f)}$'s being the respective coefficients. For instance, $f$ could be modeled by a B-spline (Fahrmeir, Kneib, Lang, and Marx 2013, Section 8.1), and $d_{ij}$ could refer to the Euclidean distance $\|s_i - s_j\|$ between the individuals’ locations $s_i$ and $s_j$, or to the geodesic distance between the nodes $i$ and $j$ in a network. The distance-based force of infection is modified additively by a linear predictor of covariates $w_{ij}$ describing the interaction of individuals $i$ and $j$ further. Hence, the whole epidemic component of Equation 1 can be written as a single linear predictor $x_i(t)^\top \alpha$ by interchanging the summation order to

$$\sum_{m=1}^M \alpha_m^{(f)} \sum_{j \in I(t)} B_m(d_{ij}) + \sum_{k=1}^K \alpha_k^{(w)} \sum_{j \in I(t)} w_{ijk} = x_i(t)^\top \alpha,$$ (3)

such that $x_i(t)$ comprises all epidemic terms summed over $j \in I(t)$. Note that the use of additive covariates $w_{ij}$ on top of the distance kernel in (1) is different from twinstim’s multiplicative approach. One advantage of the additive approach is that the subsequent linear decomposition of the distance kernel allows one to gather all parts of the epidemic component in a single linear predictor. Hence, the above model represents a CIF extension of what in the context of survival analysis is known as an additive-multiplicative hazard model (Martinussen and Scheike 2006). As a consequence, the twinSIR model could in principle be fitted with the timereg package, which yields estimates for the cumulative hazards. However, Höhle (2009) chooses a more direct inferential approach: To ensure that the CIF $\lambda_i(t)$ is non-negative, all covariates are encoded such that the components of $w_{ij}$ are non-negative. Additionally, the parameter vector $\alpha$ is constrained to be non-negative. Subsequent parameter inference is then based on the resulting constrained penalized likelihood which gives directly interpretable estimates of $\alpha$. Future work could investigate the potential of a multiplicative approach for the epidemic component in twinSIR.

2. Data structure: epidata

New SIR-type event data typically arrive in the form of a simple data frame with one row per individual and sequential event time points as columns. For the 1861 Hagelloch measles epidemic, which has previously been analyzed by, e.g., Neal and Roberts (2004), such a data set of the 188 affected children is contained in the surveillance package:

R> data("hagelloch")
R> head(hagelloch.df, n = 5)
The help("hagelloch") contains a description of all columns. Here we concentrate on the event columns PRO (appearance of prodromes), ERU (eruption), and DEAD (day of death if during the outbreak). We take the day on which the index case developed first symptoms, 30 October 1861 (min(hagelloch.df$PRO)), as the start of the epidemic, i.e., we condition on this case being initially infectious. As for twinstim, the property of point processes that concurrent events have zero probability requires special treatment. Ties are due to the interval censoring of the data to a daily basis – we broke these ties by adding random jitter to the event times within the given days. The resulting columns tPRO, tERU, and tDEAD are relative to the defined start time. Following Neal and Roberts (2004), we assume that each child becomes infectious (S $\rightarrow$ I event at time tI) one day before the appearance of prodromes, and is removed from the epidemic (I $\rightarrow$ R event at time tR) three days after the appearance of rash or at the time of death, whichever comes first.

For further processing of the data, we convert hagelloch.df to the standardized epidata structure for twinSIR. This is done by the converter function as.epidata, which also checks consistency and optionally pre-calculates the epidemic terms $x_i(t)$ of Equation 3 to be incorporated in a twinSIR model. The following call generates the epidata object hagelloch:

```R
hagelloch <- as.epidata(hagelloch.df,
 + t0 = 0, tI.col = "tI", tR.col = "tR",
 + id.col = "PN", coords.cols = c("x.loc", "y.loc"),
 + f = list(household = function(u) u == 0,
 + nohousehold = function(u) u > 0),
 + w = list(c1 = function (CL.i, CL.j) CL.i == "1st class" & CL.j == CL.i,
 + c2 = function (CL.i, CL.j) CL.i == "2nd class" & CL.j == CL.i),
 + keep.cols = c("SEX", "AGE", "CL"))
```

The coordinates (x.loc, y.loc) correspond to the location of the household the child lives in and are measured in meters. Note that twinSIR allows for tied locations of individuals, but assumes the relevant spatial location to be fixed during the entire observation period. By default, the Euclidean distance between the given coordinates will be used. Alternatively, as.epidata also accepts a pre-computed distance matrix via its argument D without requiring spatial coordinates. The argument f lists distance-dependent basis functions $B_m$ for which the epidemic terms $\sum_{j \in I(t)} B_m(d_{ij})$ shall be generated. Here, household ($x_{i,H}(t)$) and nohousehold ($x_{i,g}(t)$) count for each child the number of currently infective children in its household and outside its household, respectively. Similar to Neal and Roberts (2004), we also calculate the covariate-based epidemic terms $c1$ ($x_{i,c1}(t)$) and $c2$ ($x_{i,c2}(t)$) counting
the number of currently infective classmates. Note from the corresponding definitions of \( w_{ij1} \) and \( w_{ij2} \) in \( w \) that \( c1 \) is always zero for children of the second class and \( c2 \) is always zero for children of the first class. For pre-school children, both variables equal zero over the whole period. By the last argument \texttt{keep.cols}, we choose to only keep the covariates \texttt{SEX}, \texttt{AGE}, and school \texttt{Class} from \texttt{hagelloch.df}.

The first few rows of the generated \texttt{epidata} object are shown below:

```r
R> head(hagelloch, n = 5)

    BLOCK id start stop atRiskY event Revent x.loc y.loc SEX AGE CL
 1    1   1   0  1.136   1   0 142.5 100.0  female  7  1st class
 2    1   2   0  1.136   1   0 142.5 100.0  female  6  1st class
 3    1   3   0  1.136   1   0 142.5 100.0  female  4  preschool
 4    1   4   0  1.136   1   0 165.0 102.5   male  13  2nd class
 5    1   5   0  1.136   1   0 145.0 120.0  female  8  1st class

household nothousehold c1 c2
 1       0 1 0 0
 2       0 1 0 0
 3       0 1 0 0
 4       0 1 0 1
 5       0 1 0 0
```

The \texttt{epidata} structure inherits from counting processes as implemented by the \texttt{Surv} class of package \texttt{survival} and also used in \texttt{timereg}. Specifically, the observation period is split up into consecutive time intervals \([\texttt{start}; \texttt{stop}]\) of constant conditional intensities. As the CIF \( \lambda_i(t) \) of Equation (1) only changes at time points, where the set of infectious individuals \( I(t) \) or some endemic covariate in \( \nu_i(t) \) change, those occurrences define the break points of the time intervals. Altogether, the \texttt{hagelloch} event history consists of 375 time \texttt{BLOCKs} of 188 rows, where each row describes the state of individual \texttt{id} during the corresponding time interval. The susceptibility status and the I- and R-events are captured by the columns \texttt{atRiskY}, \texttt{event} and \texttt{Revent}, respectively. The \texttt{atRiskY} column indicates if the individual is at risk of becoming infected in the current interval. The event columns indicate, which individual was infected or removed at the \texttt{stop} time. Note that at most one entry in the \texttt{event} and \texttt{Revent} columns is 1, all others are 0.

Apart from being the input format for \texttt{twinSIR} models, the \texttt{epidata} class has several associated methods (Table 1), which are similar in spirit to the methods described for \texttt{epidataCS}.

<table>
<thead>
<tr>
<th>Display</th>
<th>Subset</th>
<th>Modify</th>
</tr>
</thead>
<tbody>
<tr>
<td>print</td>
<td>[</td>
<td>update</td>
</tr>
<tr>
<td>summary</td>
<td></td>
<td></td>
</tr>
<tr>
<td>plot</td>
<td></td>
<td></td>
</tr>
<tr>
<td>animate</td>
<td></td>
<td></td>
</tr>
<tr>
<td>stateplot</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Generic and non-generic functions applicable to \texttt{epidata} objects.

For example, Figure 1 illustrates the course of the Hagelloch measles epidemic by counting processes for the number of susceptible, infectious and removed children, respectively. Figure 2
shows the locations of the households. An animated map can also be produced to view the households’ states over time and a simple stateplot shows the changes for a selected unit.

```r
R> plot(hagelloch, xlab = "Time [days]")
```

![Figure 1: Evolution of the 1861 Hagelloch measles epidemic in terms of the numbers of susceptible, infectious, and recovered children. The bottom rug marks the infection times \( t_I \).](image)

```r
R> hagelloch.coords <- summary(hagelloch)$coordinates
R> plot(hagelloch.coords, xlab = "x [m]", ylab = "y [m]",
+     pch = 15, asp = 1, cex = sqrt(multiplicity(hagelloch.coords)))
R> legend(x = "topleft", pch = 15, legend = c(1, 4, 8), pt.cex = sqrt(c(1, 4, 8)),
+     title = "Household size")
```

### 3. Modeling and inference

#### 3.1. Basic example

To illustrate the flexibility of `twinSIR` we will analyze the Hagelloch data using class room and household indicators similar to Neal and Roberts (2004). We include an additional endemic background rate \( \exp(\beta_0) \), which allows for multiple outbreaks triggered by external sources. Consequently, we do not need to ignore the child that got infected about one month after the end of the main epidemic (see the last event mark in Figure 1). Altogether, the CIF for a child \( i \) is modeled as

\[
\lambda_i(t) = Y_i(t) \cdot \left[ \exp(\beta_0) + \alpha_{H}x_{i,H}(t) + \alpha_{c1}x_{i,c1}(t) + \alpha_{c2}x_{i,c2}(t) + \alpha_{\bar{H}}x_{i,\bar{H}}(t) \right],
\]

where \( Y_i(t) = \mathbb{I}(i \in S(t)) \) is the at-risk indicator. By counting the number of infectious classmates separately for both school classes as described in the previous section, we allow for class-specific effects \( \alpha_{c1} \) and \( \alpha_{c2} \) on the force of infection. The model is estimated by maximum likelihood (Höhle 2009) using the call
Modeling epidemics in a fixed population with known distances

Figure 2: Spatial locations of the Hagelloch households. The size of each dot is proportional to the number of children in the household.

```R
hagellochFit <- twinSIR(~household + c1 + c2 + nothousehold, data = hagelloch)
```

and the fit is summarized below:

```R
set.seed(1)
summary(hagellochFit)
```

Call:
twinSIR(formula = ~household + c1 + c2 + nothousehold, data = hagelloch)

Coefficients:

| Estimate | Std. Error | z value | Pr(>|z|) |
|----------|------------|---------|----------|
| household | 0.026868 | 0.006113 | 4.39 | 1.1e-05 *** |
| c1       | 0.023892 | 0.005026 | 4.75 | 2.0e-06 *** |
| c2       | 0.002932 | 0.000755 | 3.88 | 0.0001 *** |
| nothousehold | 0.000831 | 0.000142 | 5.87 | 4.3e-09 *** |
| cox(logbaseline) | -7.362644 | 0.887989 | -8.29 | < 2e-16 *** |

---

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Total number of infections: 187

One-sided AIC: 1245 (simulated penalty weights)

Log-likelihood: -619

Number of log-likelihood evaluations: 119

The results show, e.g., a 0.0239 / 0.0029 = 8.149 times higher transmission between individuals in the 1st class than in the 2nd class. Furthermore, an infectious housemate adds 0.0269 / 0.0008 = 32.32 times as much infection pressure as infectious children outside the household. The endemic background rate of infection in a population with no current measles cases is
estimated to be $\exp(\hat{\beta}_0) = \exp(-7.363) = 0.0006345$. An associated Wald confidence interval (CI) based on the asymptotic normality of the maximum likelihood estimator (MLE) can be obtained by $\exp$-transforming the `confint` for $\beta_0$:

```r
R> exp(confint(hagellochFit, parm = "cox(logbaseline)"))
```

<table>
<thead>
<tr>
<th></th>
<th>2.5 %</th>
<th>97.5 %</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>cox(logbaseline)</code></td>
<td>0.0001113</td>
<td>0.003617</td>
</tr>
</tbody>
</table>

Note that Wald confidence intervals for the epidemic parameters $\alpha$ are to be treated carefully, because their construction does not take the restricted parameter space into account. For more adequate statistical inference, the behavior of the log-likelihood near the MLE can be investigated using the `profile`-method for `twinSIR` objects. For instance, to evaluate the normalized profile log-likelihood of $\alpha_{c1}$ and $\alpha_{c2}$ on an equidistant grid of 25 points within the corresponding 95% Wald CIs, we do:

```r
R> prof <- profile(hagellochFit, + list(c(match("c1", names(coef(hagellochFit)))), NA, NA, 25), + c(match("c2", names(coef(hagellochFit)))), NA, NA, 25))
```

The profiling result contains 95% highest likelihood based CIs for the parameters, as well as the Wald CIs for comparison:

```r
R> prof$ci.hl
```

<table>
<thead>
<tr>
<th>idx</th>
<th>hl.low</th>
<th>hl.up</th>
<th>wald.low</th>
<th>wald.up</th>
<th>mle</th>
</tr>
</thead>
<tbody>
<tr>
<td>c1</td>
<td>2</td>
<td>0.015219</td>
<td>0.034969</td>
<td>0.014041</td>
<td>0.033744</td>
</tr>
<tr>
<td>c2</td>
<td>3</td>
<td>0.001576</td>
<td>0.004535</td>
<td>0.001453</td>
<td>0.004411</td>
</tr>
</tbody>
</table>

The entire functional form of the normalized profile log-likelihood on the requested grid as stored in `prof$lp` can be visualized by:

```r
R> plot(prof)
```

The above model summary also reports the one-sided AIC (Hughes and King 2003), which can be used for model selection under positivity constraints on $\alpha$ as described in Höhle (2009). The involved parameter penalty is determined by Monte Carlo simulation, which is why we did `set.seed` before the `summary` call. The algorithm is described in Silvapulle and Sen (2005, p. 79, Simulation 3) and involves quadratic programming using package `quadprog` (Turlach and Weingessel 2019). If there are less than three constrained parameters in a `twinSIR` model, the penalty is computed analytically.

### 3.2. Model diagnostics

Table 2 lists all methods for the `twinSIR` class. For example, to investigate how the conditional intensity function decomposes into endemic and epidemic components over time, we produce Figure 4a by:

```r
R> plot(hagellochFit, which = "epidemic proportion", xlab = "time [days]")
```
Modeling epidemics in a fixed population with known distances

Figure 3: Normalized log-likelihood for $\alpha_{c1}$ and $\alpha_{c2}$ when fitting the twinSIR model formulated in Equation (4) to the Hagelloch data.

<table>
<thead>
<tr>
<th>Display</th>
<th>Extract</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>print</td>
<td>vcov</td>
<td>simulate</td>
</tr>
<tr>
<td>summary</td>
<td>logLik</td>
<td></td>
</tr>
<tr>
<td>plot</td>
<td>AIC</td>
<td></td>
</tr>
<tr>
<td>intensityplot</td>
<td>extractAIC</td>
<td></td>
</tr>
<tr>
<td>checkResidualProcess</td>
<td>profile</td>
<td></td>
</tr>
<tr>
<td></td>
<td>residuals</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Generic and non-generic functions for twinSIR. There are no specific coef or confint methods, since the respective default methods from package stats apply outright.

Note that the last infection was necessarily caused by the endemic component since there were no more infectious children in the observed population which could have triggered the new case. We can also inspect temporal Cox-Snell-like residuals of the fitted point process using the function checkResidualProcess as for the spatio-temporal point process models in vignette("twinstim"). The resulting Figure 4b reveals some deficiencies of the model in describing the waiting times between events, which might be related to the assumption of fixed infection periods.

To illustrate AIC-based model selection, we may consider a more flexible model for local spread using a step function for the distance kernel $f(u)$ in Equation 2. An updated model with $B_1 = I_{(0;100)}(u)$, $B_2 = I_{(100;200)}(u)$, $B_3 = I_{(200;\infty)}(u)$ can be fitted as follows:

```R
knots <- c(100, 200)
fstep <- list(
  + B1 = function(D) D > 0 & D < knots[1],
  + B2 = function(D) D >= knots[1] & D < knots[2],
  + B3 = function(D) D >= knots[2])
hagellochFit_fstep <- twinSIR(~household + c1 + c2 + B1 + B2 + B3,
  + data = update(hagelloch, f = fstep))
```
Figure 4: Diagnostic plots for the twinSIR model formulated in Equation 4.

R> set.seed(1)
R> AIC(hagellochFit, hagellochFit_fstep)

<table>
<thead>
<tr>
<th></th>
<th>df</th>
<th>AIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>hagellochFit</td>
<td>5</td>
<td>1245</td>
</tr>
<tr>
<td>hagellochFit_fstep</td>
<td>7</td>
<td>1246</td>
</tr>
</tbody>
</table>

Hence the simpler model with just a nohousehold component instead of the more flexible distance-based step function is preferred.

4. Simulation

Simulation from fitted twinSIR models is described in detail in Höhle (2009, Section 4). The implementation is made available by an appropriate simulate-method for class twinSIR. We skip the illustration here and refer to help("simulate.twinSIR").

References


