This introduction to the `twinSIR` modeling framework of the R package **surveillance** is based on a publication in the *Journal of Statistical Software* – Meyer, Held, and Höhle (2016, Section 4) – which is the suggested reference if you use the `twinSIR` implementation in your own work.

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

Sebastian Meyer[*]
University of Zurich

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`"):

$$\lambda_i(t) = \lambda_0(t)\,\nu_i(t) + \sum_{j \in I(t)} \left\{ f(d_{ij}) + \boldsymbol{w}_{ij}^\top \boldsymbol{\alpha}^{(w)} \right\} , \tag{1}$$

if $i \in S(t)$, i.e., if individual i is currently susceptible, and $\lambda_i(t) = 0$ otherwise. The rate decomposes into two components. The first, endemic component consists of a Cox propor-

[*]Author of correspondence: Sebastian.Meyer@ifspm.uzh.ch

tional hazards formulation containing a semi-parametric baseline hazard $\lambda_0(t)$ and a log-linear predictor $\nu_i(t) = \exp\left(\boldsymbol{z}_i(t)^\top \boldsymbol{\beta}\right)$ 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

$$f(u) = \sum_{m=1}^{M} \alpha_m^{(f)} B_m(u) \; \geq 0\,, \tag{2}$$

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 $\|\boldsymbol{s}_i - \boldsymbol{s}_j\|$ between the individuals' locations $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$, 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 $\boldsymbol{w}_{ij}$ 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 $\boldsymbol{x}_i(t)^\top \boldsymbol{\alpha}$ by interchanging the summation order to

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

such that $\boldsymbol{x}_i(t)$ comprises all epidemic terms summed over $j \in I(t)$. Note that the use of additive covariates $\boldsymbol{w}_{ij}$ 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 $\boldsymbol{w}_{ij}$ are non-negative. Additionally, the parameter vector $\boldsymbol{\alpha}$ is constrained to be non-negative. Subsequent parameter inference is then based on the resulting constrained penalized likelihood which gives directly interpretable estimates of $\boldsymbol{\alpha}$. 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)
```

```
  PN    NAME FN HN AGE    SEX        PRO        ERU        CL DEAD IFTO SI
1  1 Mueller 41 61   7 female 1861-11-21 1861-11-25 1st class <NA>   45 10
```

```
2  2 Mueller 41 61   6 female 1861-11-23 1861-11-27 1st class <NA>   45 12
3  3 Mueller 41 61   4 female 1861-11-28 1861-12-02 preschool <NA>  172  9
4  4 Seibold 61 62  13   male 1861-11-27 1861-11-28 2nd class <NA>  180 10
5  5  Motzer 42 63   8 female 1861-11-22 1861-11-27 1st class <NA>   45 11
                C PR CA NI GE TD   TM x.loc y.loc  tPRO  tERU tDEAD    tR    tI
1 no complicatons  4  4  3  1 NA   NA 142.5 100.0 22.71 26.23    NA 29.23 21.71
2 no complicatons  4  4  3  1  3 40.3 142.5 100.0 24.21 28.79    NA 31.79 23.21
3 no complicatons  4  4  3  2  1 40.5 142.5 100.0 29.59 33.69    NA 36.69 28.59
4 no complicatons  1  1  1  1  3 40.7 165.0 102.5 28.12 29.03    NA 32.03 27.12
5 no complicatons  5  3  2  1 NA   NA 145.0 120.0 23.06 28.42    NA 31.42 22.06
```

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 $\boldsymbol{x}_i(t)$ 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,
+             nothousehold = 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 `nothousehold` ($x_{i,\bar{H}}(t)$) 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 `keep.cols`, we choose to only keep the covariates `SEX`, `AGE`, and school `CL`ass from `hagelloch.df`.

The first few rows of the generated `epidata` object are shown below:

```
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      0 142.5 100.0 female   7 1st class
2     1  2     0 1.136       1     0      0 142.5 100.0 female   6 1st class
3     1  3     0 1.136       1     0      0 142.5 100.0 female   4 preschool
4     1  4     0 1.136       1     0      0 165.0 102.5   male  13 2nd class
5     1  5     0 1.136       1     0      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 `epidata` structure inherits from counting processes as implemented by the `Surv` class of package **survival** and also used in **timereg**. Specifically, the observation period is split up into consecutive time intervals (`start`; `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 `hagelloch` event history consists of 375 time `BLOCK`s of 188 rows, where each row describes the state of individual `id` during the corresponding time interval. The susceptibility status and the I- and R-events are captured by the columns `atRiskY`, `event` and `Revent`, respectively. The `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 `stop` time. Note that at most one entry in the `event` and `Revent` columns is 1, all others are 0.

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

Display	Subset	Modify
`print`	`[`	`update`
`summary`		
`plot`		
`animate`		
`stateplot`		

Table 1: Generic and *non-generic* functions applicable to `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 `animate`d map can also be produced to view the households' states over time and a simple `stateplot` shows the changes for a selected unit.

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

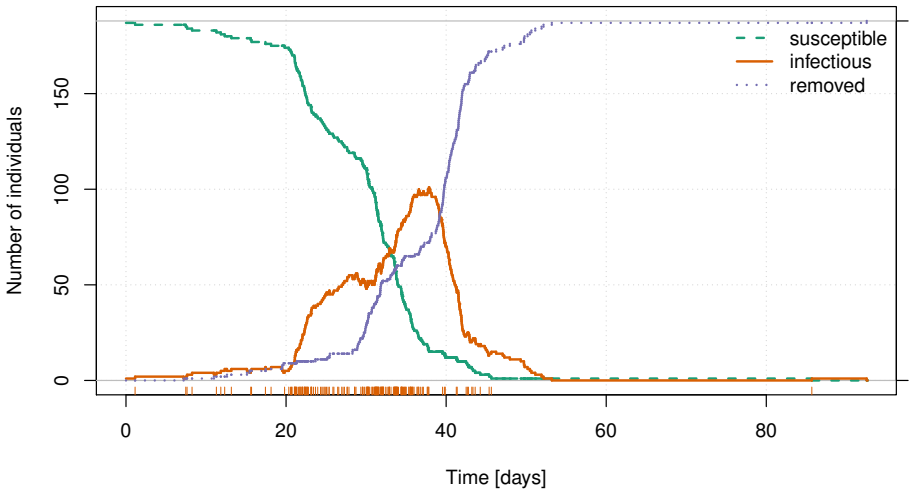


Figure 1: Evolution of the 1861 Hagelloch measles epidemic in terms of the numbers of susceptible, infectious, and recovered children. The bottom `rug` marks the infection times `tI`.

```
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")
```

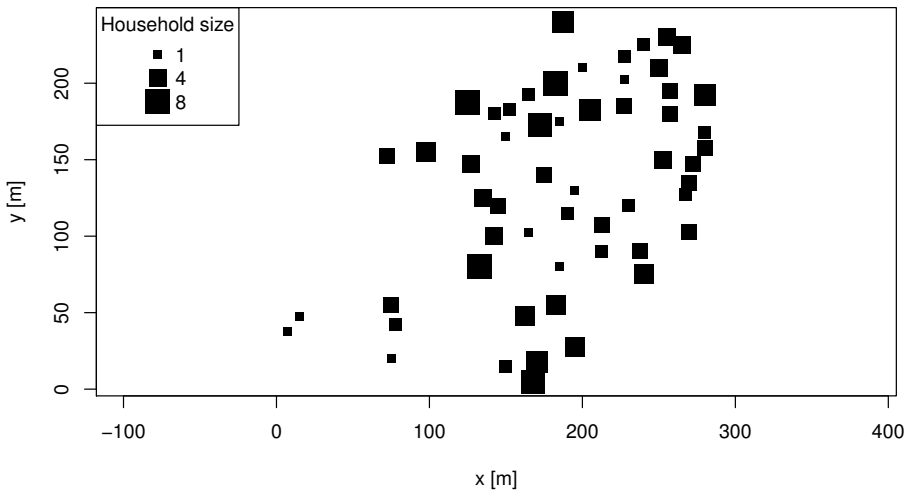


Figure 2: Spatial locations of the Hagelloch households. The size of each dot is proportional to the number of children in the household.

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

$$\lambda_i(t) = Y_i(t) \cdot \left[\exp(\beta_0) + \alpha_H x_{i,H}(t) + \alpha_{c1} x_{i,c1}(t) + \alpha_{c2} x_{i,c2}(t) + \alpha_{\bar{H}} x_{i,\bar{H}}(t)\right] , \qquad (4)$$

where $Y_i(t) = \mathbb{1}(i \in S(t))$ 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 α_{c1} and α_{c2} on the force of infection. The model is estimated by maximum likelihood (Höhle 2009) using the call

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

and the fit is summarized below:

```
R> set.seed(1)
R> 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 β_0:

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

                     2.5 %   97.5 %
cox(logbaseline) 0.0001113 0.003617
```

Note that Wald confidence intervals for the epidemic parameters $\boldsymbol{\alpha}$ 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 α_{c1} and α_{c2} on an equidistant grid of 25 points within the corresponding 95% Wald CIs, we do:

```
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> prof$ci.hl

   idx   hl.low    hl.up wald.low  wald.up      mle
c1   2 0.015219 0.034969 0.014041 0.033744 0.023892
c2   3 0.001576 0.004535 0.001453 0.004411 0.002932
```

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

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

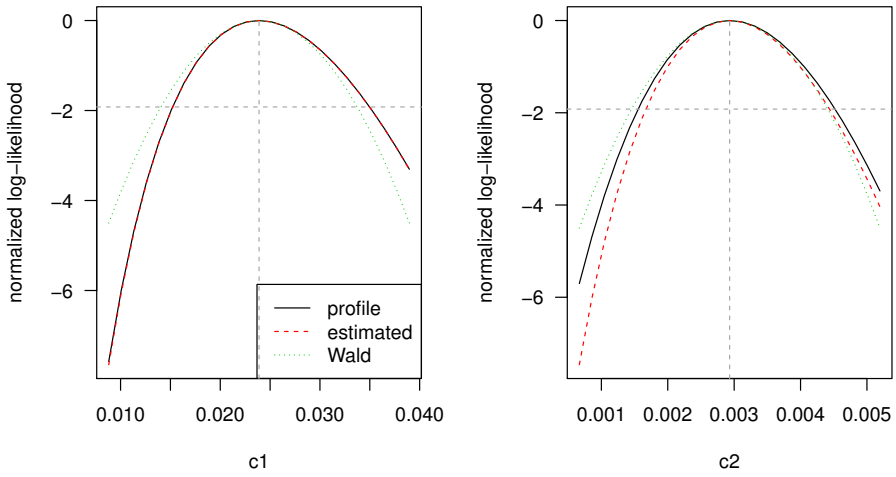


Figure 3: Normalized log-likelihood for α_{c1} and α_{c2} when fitting the `twinSIR` model formulated in Equation (4) to the Hagelloch data.

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 $\boldsymbol{\alpha}$ 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 2013). If there are less than three constrained parameters in a `twinSIR` model, the penalty is computed analytically.

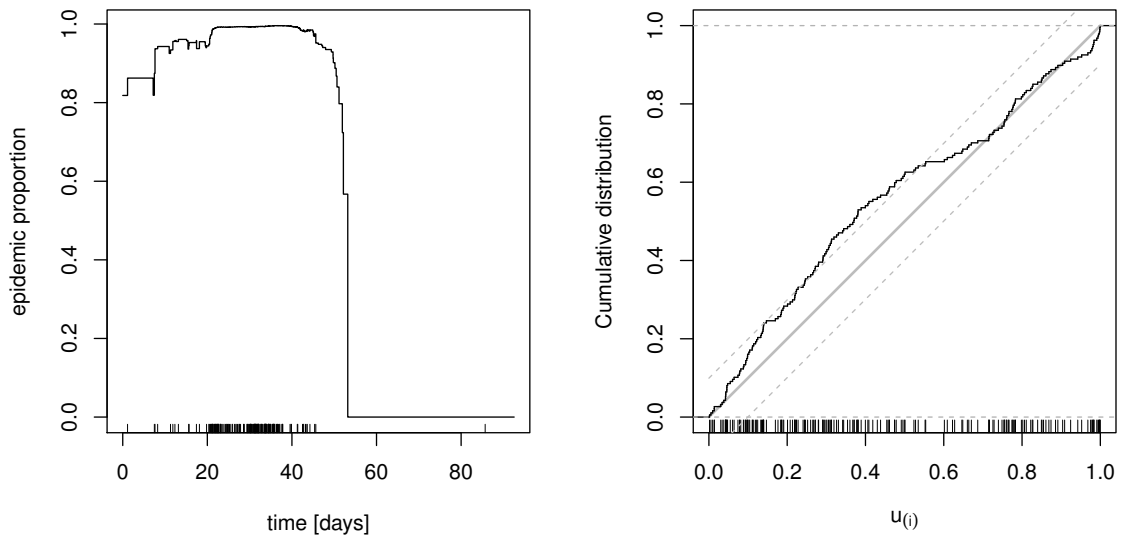
3.2. Model diagnostics

Display	Extract	Other
`print`	`vcov`	`simulate`
`summary`	`logLik`	
`plot`	`AIC`	
`intensityplot`	`extractAIC`	
`checkResidualProcess`	`profile`	
	`residuals`	

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.

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> plot(hagellochFit, which = "epidemic proportion", xlab = "time [days]")
```



(a) Epidemic proportion.

(b) Transformed residuals.

Figure 4: Diagnostic plots for the `twinSIR` model formulated in Equation 4.

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)
R> 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])
R> hagellochFit_fstep <- twinSIR(
+    ~household + c1 + c2 + B1 + B2 + B3,
+    data = update(hagelloch, f = fstep))

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

                   df  AIC
hagellochFit        5 1245
hagellochFit_fstep  7 1246
```

Hence the simpler model with just a `nothousehold` 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

Fahrmeir L, Kneib T, Lang S, Marx B (2013). *Regression: Models, Methods and Applications*. Springer-Verlag. ISBN 978-3-642-34332-2. doi:10.1007/978-3-642-34333-9.

Höhle M (2009). "Additive-Multiplicative Regression Models for Spatio-Temporal Epidemics." *Biometrical Journal*, **51**(6), 961–978. doi:10.1002/bimj.200900050.

Hughes AW, King ML (2003). "Model Selection Using AIC in the Presence of One-Sided Information." *Journal of Statistical Planning and Inference*, **115**(2), 397–411. doi:10.1016/S0378-3758(02)00159-3.

Martinussen T, Scheike TH (2006). *Dynamic Regression Models for Survival Data*. Statistics for Biology and Health. Springer-Verlag.

Meyer S, Elias J, Höhle M (2012). "A Space-Time Conditional Intensity Model for Invasive Meningococcal Disease Occurrence." *Biometrics*, **68**(2), 607–616. doi:10.1111/j.1541-0420.2011.01684.x. http://arxiv.org/abs/1508.05740.

Meyer S, Held L, Höhle M (2016). "Spatio-Temporal Analysis of Epidemic Phenomena Using the R Package `surveillance`." *Journal of Statistical Software*. In press. Preprint available at http://arxiv.org/abs/1411.0416.

Neal PJ, Roberts GO (2004). "Statistical Inference and Model Selection for the 1861 Hagelloch Measles Epidemic." *Biostatistics*, **5**(2), 249–261. doi:10.1093/biostatistics/5.2.249.

Silvapulle MJ, Sen PK (2005). *Constrained Statistical Inference: Order, Inequality, and Shape Constraints*. Wiley Series in Probability and Statistics. Wiley. ISBN 0-471-20827-2. doi:10.1002/9781118165614.

Turlach BA (2013). *quadprog: Functions to solve Quadratic Programming Problems*. R package version 1.5-5, ported to R by Andreas Weingessel, URL https://CRAN.R-project.org/package=quadprog.