mmpp: A Package for Calculating Similarity and Distance Metrics for Simple and Marked Temporal Point Process

by Hideitsu Hino, Ken Takano, and Noboru Murata

Abstract A simple temporal point process (SPP) is an important class of time series, where the sample realization of the process is solely composed of the times at which events occur. Particular examples of point process data are neuronal spike patterns or spike trains, and a large number of distance and similarity metrics for those data have been proposed. A marked point process (MPP) is an extension of a simple temporal point process, in which a certain vector valued mark is associated with each of the temporal points in the SPP. Analyses of MPPs are of practical importance because instances of MPPs include recordings of natural disasters such as earthquakes and tornadoes. In this paper, we introduce an R package **mmpp**, which implements a number of distance and similarity metrics for SPP, and also extends those metrics for dealing with MPP.

Introduction

A random point process is a mathematical model for describing a series of discrete events (Snyder and Miller, 1991). Let $\mathcal{X} = \{t; t_0 \leq t \leq t_0 + T\}$ be the base space, on which an *event* occurs. The base space can be quite abstract, but here we will take $\mathcal{X}$ to be a semi-infinite real line representing time. A set of ordered points on $\mathcal{X}$ is denoted as $x = \{x_1, x_2, \ldots, x_n\},\ x_i \leq x_{i+1}$, and called a *sample realization*, or simply *realization* of a point process.

Reflecting the importance of the analysis of point process in a broad range of science and engineering problems, there are already some R packages for modeling and simulating point process such as **splancs** (Rowlingson and Diggle, 1993), **spatstat** (Baddeley and Turner, 2005), **PtProcess** (Harte, 2010), and **stpp** (Gabriel et al., 2013). These packages support various approaches for the analysis of both simple and marked spatial or spatio-temporal point processes, namely, estimating an intensity function for sample points, visualizing the observed sample process, and running simulations based on the specified models.

To complement the above mentioned packages, in **mmpp** (Hino et al., 2015), we focus on the similarity or distance metrics between realizations of point process. Similarity and distance metrics are fundamental notion for multivariate analysis, machine learning and pattern recognition. For example, with an appropriate distance metric, a simple k nearest neighbor classifier and regressor (Cover and Hart, 1967) work satisfactory. Also, kernel methods (Shawe-Taylor and Cristianini, 2004) are a well known and widely used framework in machine learning, in which inferences are done solely based on the values of kernel function, which is considered as a similarity metric between two objects.

As for the distance and similarity metric for point processes, vast amount of methods are developed in the field of neuroscience (Kandel et al., 2000). In this field, neural activities are recorded as sequences of spikes (called *spike trains*), which is nothing but a realization of a simple point process (SPP). By its nature, the responses of neurons to the same stimulus can be different. To claim the repeatability and the reliability of experimental results, a number of different distance and similarity metrics between sequence of spikes are developed (Victor, 2005; Schreiber et al., 2003; Kreuz et al., 2007; Quian Quiroga et al., 2002; van Rossum, 2001).

The package **mmpp** categorizes commonly used metrics for spike trains and offers implementations for them. Since a spike train is a realization of simple point process, the original metrics developed in the field of neuroscience do not consider marked point process (MPP) realizations. **mmpp** extends conventional metrics for SPP to MPP. We have two main aims in the development of **mmpp**:

1. to have a systematic and unified platform for calculating the similarities and distances between SPP, and
2. to support marked point process to offer a platform for performing metric-based analysis of earthquakes, tornados, epidemics, or stock exchange data.

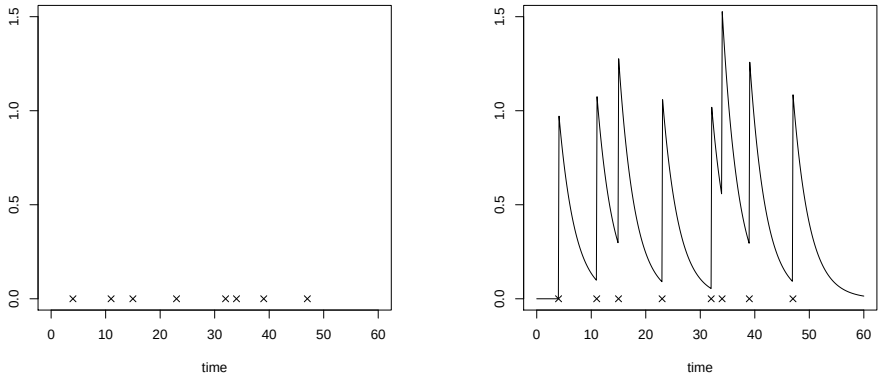

Figure 1: An example of continuation by smoothing. Left: event timings are marked with $\times$. Right: the corresponding continuous function $v_x(t)$.

Distances and similarities for point process

Since realizations of temporal SPP are ordered sets of the events, commonly used the Euclidean distance and inner product cannot be directly defined between them. Most of metrics for SPP realizations first transform the realizations $x = \{x_1, \ldots, x_n\},\ x_i \leq x_{i+1}$ and $y = \{y_1, \ldots, y_m\}, ; y_j \leq y_{j+1}$ to continuous functions $v_x(t)$ and $v_y(t)$, then define the distance or similarity metric between them. Based on the transformations, we categorize conventional methods for defining metrics on SPP realizations, and explain one by one in the following subsections.

We note that there are some attempts to directly define distances between SPP realizations. One of the most principled and widely used methods is based on the edit distance (Victor, 2005), and this method is extended to deal with MPP realizations by Suzuki et al. (2010). However, this approach is computationally expensive and prohibitive for computing the distance between spike trains with even few dozen spikes. We exclude this class of metric from the current version of **mmpp**.

In the following, we often use kernel smoothers and step functions for transforming SPP realizations. For notational convenience, we denote a kernel smoother with parameter τ by $h_\tau(t)$, and the Heaviside step function

$$u(t) = \begin{cases} 1, & t \geq 0, \\ 0, & t < 0. \end{cases} \tag{1}$$

Examples of smoother include Gaussian kernel smoother $h^g_\tau(t) = \exp(-t^2/(2\tau^2))/\sqrt{2\pi\tau^2}$ and Laplacian kernel smoother $h^l_\tau(t) = \exp(-|t|/\tau)/(2\tau)$.

Filtering to a continuous function

The most commonly used and intensively studied metrics for spike trains are based on the mapping of event sequence to a real valued continuous function as

$$x = \{x_1, \ldots, x_n\} \Rightarrow v_x(t) = \frac{1}{n} \sum_{i=1}^{n} h_\tau(t - x_i) \cdot u(t - x_i). \tag{2}$$

Figure 1 illustrates the transformation of an event sequence to a continuous function by the smoothing method.

Then, the inner product of x and y is defined by the usual ℓ_2 inner product in functional space by

$$k(x, y) = \int_0^\infty \mathrm{d}t v_x(t) v_y(t) \in [-\infty, \infty], \tag{3}$$

and similarly the distance is defined by

$$d(x, y) = \sqrt{\int_0^\infty \mathrm{d}t (v_x(t) - v_y(t))^2}. \tag{4}$$

When we use the Laplacian smoother $h^l_\tau(t) = \exp(-|t|/\tau)/(2\tau)$, the similarity and distance are

analytically given as

$$k(x,y) = \int_0^\infty \mathrm{d}t v_x(t) v_y(t) = \frac{1}{4\tau nm} \sum_{i=1}^{n} \sum_{j=1}^{m} \exp\left(-\frac{1}{\tau}|x_i - y_j|\right), \tag{5}$$

and

$$\begin{aligned} d^2(x,y) =& k(x,x) + k(y,y) - 2k(x,y) \\ =& \frac{1}{4\tau n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \exp\left(-\frac{|x_i - x_j|}{\tau}\right) + \frac{1}{4\tau m^2} \sum_{i=1}^{m} \sum_{j=1}^{m} \exp\left(-\frac{|y_i - y_j|}{\tau}\right) \\ & - \frac{1}{2\tau nm} \sum_{i=1}^{n} \sum_{j=1}^{m} \exp\left(-\frac{|x_i - y_j|}{\tau}\right), \end{aligned} \tag{6}$$

respectively. This distance eq. (6) is adopted by van Rossum (2001) for measuring the distance between spike trains. On the other hand, Schreiber et al. (2003) proposed to use the correlation defined by

$$\mathrm{cor}(x,y) = \frac{\int_0^\infty \mathrm{d}t v_x(t) v_y(t)}{\sqrt{\int_0^\infty \mathrm{d}t v_x(t) v_x(t)} \sqrt{\int_0^\infty \mathrm{d}t v_y(t) v_y(t)}} \tag{7}$$

to measure the similarity between spike trains. This class of measure is extended to take into account the effect of burst, i.e., short interval in which events occur in high frequency, and refractory period, i.e., short interval in which event tends to be suppressed immediately after the previous events (Houghton, 2009; Lyttle and Fellous, 2011). These two effects, namely burst and refractory period, are commonly observed in neural activities. They are also observed in earthquake catalogues. After large main shock, usually we observe high frequent aftershocks. On the other hand, suppression of events is sometimes happen, possibly because that after a big event, the coda is so large that one cannot detect smaller events under the large ongoing signal from the big event (Kagan, 2004; Iwata, 2008).

The filtering-based metric is computed by using the function `fmetric` in **mmpp**. First two arguments `S1` and `S2` are the (marked) point process realizations of the form of `matrix` object. The first column of `S1` and `S2` are the event timings and the rest are the marks. The argument `measure` can be either `"sim"` or `"dist"`, indicating the similarity or distance, respectively. By default, the function assumes the Laplacian smoother. When the argument `h` for `fmetric` is set to a function with scaling parameter τ as

```
> fmetric(S1, S2, measure = "sim", h = function(x, tau) exp(-x^2/tau), tau = 1)
```

the integrals in eq. (3) and eq. (4) are numerically done using the R function `integrate`. The function `h` should be square integrable and non-negative.

Intensity inner products

For analysis of point processes, the intensity function plays a central roll. Paiva et al. (2009) proposed to use the intensity function for defining the inner product between SPP realizations. Let $N(t)$ be the number of events observed in the interval $(0,t]$. The intensity function of a counting process $N(t)$ is defined by

$$\lambda_x(t) = \lim_{\epsilon \to 0} \frac{1}{\epsilon} \Pr[N(t+\epsilon) - N(t) = 1]. \tag{8}$$

We note that we can also consider the conditional intensity function reflecting the history up to the current time t, but we only explain the simplest case in this paper. Assuming that the SPP to be analysed is well approximated by a Poisson process, the intensity function is estimated by using a smoother $h_\tau(t)$ as

$$\hat{\lambda}_x(t) = \frac{1}{n} \sum_{i=1}^{n} h_\tau(t - x_i) \tag{9}$$

in non-parametric manner (Reiss, 1993). Using the estimates of intensity functions for processes behind x and y, Paiva et al. (2009) defined a similarity metric by

$$k(x,y) = \int_{-\infty}^{\infty} \mathrm{d}t \hat{\lambda}_x(t) \hat{\lambda}_y(t) = \frac{1}{4\tau^2 nm} \sum_{i=1}^{n} \sum_{j=1}^{m} \int_{-\infty}^{\infty} \mathrm{d}t h_\tau(t - x_i) h_\tau(t - y_j). \tag{10}$$

Particularly, when we use a Gaussian smoother $h_\tau^g(t) = \exp\left(-t^2/(2\tau^2)\right)/\sqrt{2\pi\tau}$, the integral is

analytically computed and we obtain an explicit formula

$$k(x,y) = \int_{-\infty}^{\infty} \mathrm{d}t \hat{\lambda}_x(t)\hat{\lambda}_y(t) = \frac{1}{4\sqrt{\pi}\tau nm} \sum_{i=1}^{n}\sum_{j=1}^{m} \exp\left(-\frac{(x_i - y_j)^2}{4\tau^2}\right). \tag{11}$$

The distance metric is also defined as

$$d(x,y) = \int_{-\infty}^{\infty} \mathrm{d}t (\hat{\lambda}_x(t) - \hat{\lambda}_y(t))^2, \tag{12}$$

and it is also simplified when we use the Gaussian smoother.

This class of measure is most in alignment with the statistical model of point process. We estimated the intensity function in a versatile non-parametric approach, but it is reasonable to use other models such as Hawkes model (Hawkes, 1971) when we should include the self-exciting nature of the process.

The intensity inner product metric is computed by using the function `iipmetric` in **mmpp**. In the current version, the function assumes the Gaussian smoother, and its scaling parameter is specified by the argument `tau` as

```
> iipmetric(S1, S2, measure = "sim", tau = 1)
```

Co-occurrence metric

For comparing two SPP realizations, it is natural to *count* the number of events which can be considered to be co-occurring. There are two metrics for SPP realizations based on the notion of co-occurrence.

The first one proposed by Quian Quiroga et al. (2002) directly *counts near-by events*. The closeness of two events are defined by adaptively computed thresholds, making the method free from tuning parameter. Suppose we have two SPP realizations $x = \{x_1, \ldots, x_n\}$ and $y = \{y_1, \ldots, y_m\}$. For any events $x_i \in x$ and $y_j \in y$, a threshold under which the two events are considered to be synchronous with each other is defined as half of the minimum of the four inter event intervals around these two events:

$$\tau_{ij} = \frac{1}{2}\min\{x_{i+1} - x_i, x_i - x_{i-1}, y_{j+1} - y_j, y_j - y_{j-1}\}. \tag{13}$$

We note that τ_{ij} in the above definition depends on x and y, though, for the sake of notational simplicity, we simply denote by τ_{ij}. Then, the function that counts the number of events in y which is coincided with those in x is defined by

$$c(x|y) = \sum_{i=1}^{n}\sum_{j=1}^{m} P_{ij}, \tag{14}$$

$$P_{ij} = \begin{cases} 1, & 0 < x_i - y_j < \tau_{ij}, \\ 1/2, & x_i = y_j, \\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

Using this counting function, a similarity measure between x and y is defined as

$$k(x,y) = \frac{c(x|y) + c(y|x)}{\sqrt{nm}}, \tag{16}$$

and a distance measure is obtained by the transformation (Lyttle and Fellous, 2011):

$$d(x,y) = 1 - k(x,y). \tag{17}$$

The second metric based on the counting co-occurrence is proposed by Hunter and Milton (2003), which transforms x to a continuous function $v_x(t)$, and sums up the near-by events in proportion to their degree of closeness. Denoting the closest event time in y from an event $x_i \in x$ by $y_{(x_i)}$, we define a function which measures degree of closeness by

$$dc_\tau(x_i) = \exp\left(-\frac{|x_i - y_{(x_i)}|}{\tau}\right). \tag{18}$$

Then, a similarity metric between x and y is defined by

$$k(x,y) = \frac{\frac{1}{n}\sum_{i=1}^{n} dc_\tau(x_i) + \frac{1}{m}\sum_{j=1}^{m} dc_\tau(y_j)}{2}, \tag{19}$$

and the distance is naturally defined by

$$d(x, y) = 1 - k(x, y). \tag{20}$$

The co-occurrence based metrics are computed by using the function `coocmetric`. By default, the function assumes the counting similarity in eq. (16). The smoothed counting similarity is computed by specifying the argument `type = "smooth"` as

```
> coocmetric(S1, S2, measure = "sim", type = "smooth", tau = 1)
```

Inter event interval

Assume an SPP realization $x = \{x_1, \ldots, x_n\}, x_n < T$ such that for every event time $x_i, 0 < x_i < T$, where T is the horizon of the time interval. In inter event interval proposed by Kreuz et al. (2007), the SPP realization x is first modified to include artificial events corresponding to the beginning and end of the interval as

$$x = \{x_0 = 0, x_1, \ldots, x_n, x_{n+1} = T\}. \tag{21}$$

Then each event is mapped to a function $v_x(t)$ as

$$v_x(t) = \sum_{i=0}^{n+1} f_i(t), \qquad f_i(t) = \begin{cases} 0, & t \notin [x_i, x_{i+1}), \\ x_{i+1} - x_i, & t \in [x_i, x_{i+1}). \end{cases} \tag{22}$$

Two SPP realizations x and y are transformed to $v_x(t)$ and $v_y(t)$, then they are used to define an intermediate function

$$I_{xy}(t) = \frac{\min\{v_x(t), v_y(t)\}}{\max\{v_x(t), v_y(t)\}}. \tag{23}$$

This function takes value 1 when x is identical to y, and takes smaller value when x and y are highly dissimilar. By using this intermediate function, the similarity measure is defined by

$$k(x, y) = \frac{1}{T} \int_0^T \mathrm{d}t I_{xy}(t), \tag{24}$$

and the distance is defined by

$$d(x, y) = \frac{1}{T} \int_0^T \mathrm{d}t (1 - I_{xy}(t)), \tag{25}$$

which is originally defined in (Kreuz et al., 2007). A simple example of transformation $x \Rightarrow v_x(t), y \Rightarrow v_y(t)$ and $x, y \Rightarrow 1 - I_{xy}(t)$ is illustrated in Figure 2.

The inter event interval metrics are computed by using the function `ieimetric` as

```
> ieimetric(S1, S2, measure = "sim")
```

Extension to marked point process

Sometimes events considered in point process entail certain vector valued *marks*. For example, seismic events are characterized by the time point the earthquake happens, and a set of attributes such as magnitude, depth, longitude, and latitude of the hypo-center. To deal with marked point process, we extend the base space $\mathcal{X}$ as $\mathcal{X} = \{t; t_0 \leq t \leq t_0 + T\} \times \mathbb{R}^p$, the Cartesian product of the time interval $[t_0, t_0 + T]$ and a region of the p dimensional Euclidean space corresponding to marks. Realizations of MPP are denoted by, for example, $x = \{(x_1, \boldsymbol{r}_1), \ldots, (x_n, \boldsymbol{r}_n)\}$ and $y = \{(y_1, \boldsymbol{s}_1), \ldots, (y_m, \boldsymbol{s}_m)\}$.

There might be many possibilities for the way of extension. The packages **mmpp** takes the simplest way to deal with marks in a unified and computationally efficient manner, namely, the density or weight of marks are included in the metrics for SPP by Gaussian windowing as shown in eq. (27).

Filtering to continuous function

In the same manner as eq. (2), the marked point process realization $x = \{(x_1, \boldsymbol{r}_1), \ldots, (x_n, \boldsymbol{r}_n)\}$ is transformed to continuous function as

$$x = \{(x_1, \boldsymbol{r}_i), \ldots, (x_n, \boldsymbol{r}_n)\} \Rightarrow v_x(t, \boldsymbol{z}) = \frac{1}{n} \sum_{i=1}^{n} h_M(\boldsymbol{z} - \boldsymbol{r}_i) h_\tau(t - x_i) \cdot u(t - x_i), \tag{26}$$

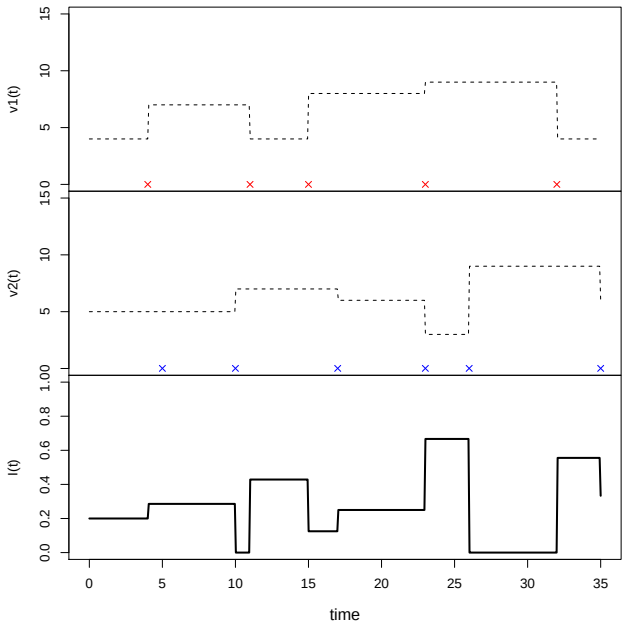

Figure 2: Example of transformation of two point process realizations x and y into the intermediate function I_{xy}. Top panel shows x and corresponding continuous function $v_x(t)$. Middle panel shows y and $v_y(t)$. Bottom panel shows the intermediate function $1 - I_{xy}(t)$.

where

$$h_M(z) = (2\pi)^{-p/2}|M|^{1/2}\exp\left(-\frac{1}{2}z^\top Mz\right) = (2\pi)^{-p/2}|M|^{1/2}\exp\left(-\frac{1}{2}\|z\|_M^2\right), M \in \mathbb{R}^{p\times p}, \quad (27)$$

where $|M|$ is the determinant of a matrix M, and $\|z\|_M^2 = z^\top Mz$. Integrating with respect to both time t and mark z, we define the inner product by

$$k(x,y) = \int_{\mathbb{R}^p} \mathrm{d}z \int_0^\infty \mathrm{d}t v_x(t,z)v_y(t,z), \quad (28)$$

and the distance by

$$d^2(x,y) = \int_{\mathbb{R}^p} \mathrm{d}z \int_0^\infty \mathrm{d}t (v_x(t,z) - v_y(t,z))^2. \quad (29)$$

By virtue of Gaussian windowing, the integral with respect to mark is explicitly written as

$$\int_{\mathbb{R}^p} \mathrm{d}z \exp\left(-\frac{1}{2}\|z-r_i\|_M^2 - \frac{1}{2}\|z-s_j\|_M^2\right) = (2\pi)^{\frac{p}{2}}\sqrt{|2M|}\exp\left(-\frac{1}{4}\|r_i - s_j\|_M^2\right). \quad (30)$$

Furthermore, when we use Laplacian smoother for transforming temporal SPP, we obtain

$$k(x,y) = \frac{|M|^{1/2}}{2^{p+2}\pi^{p/2}\tau nm}\sum_{i=1}^n\sum_{j=1}^m \exp\left(-\frac{\|r_i - s_j\|_M^2}{4}\right)\exp\left(-\frac{|x_i - y_j|}{\tau}\right). \quad (31)$$

The distance metric is also calculated in the same manner.

We note that the effect of marks depend on the units used for the various marks. It is reasonable to estimate the variance of each mark, and set the diagonal elements of M be reciprocal of the variances, as adopted as the default setting for M in **mmpp**.

Intensity inner product

Extending kernel estimation eq. (9) to multivariate kernel estimation as

$$\hat{\lambda}_x(t,z) = \frac{1}{n}\sum_{i=1}^n h_M(z - r_i)h_\tau(t - x_i), \quad (32)$$

we obtain the estimate of intensity function of the marked point process. We note that kernel density estimation for multidimensional variable is inaccurate in general, and we can instead estimate ground intensity function $\lambda(t)$ and density function for mark $\lambda(z)$ separately. However, in many applications, the dimension of marks is not so high, and currently we adopt the kernel based estimator in eq. (32). The intensity inner product for MPP realizations is then defined by

$$k(x,y) = \int_{\mathbb{R}^p} \mathrm{d}\boldsymbol{z} \int_{-\infty}^{\infty} \mathrm{d}t \hat{\lambda}_x(t,\boldsymbol{z}) \hat{\lambda}_y(t,\boldsymbol{z}). \tag{33}$$

When we use the Gaussian smoother $h_\tau^g(t) = \exp\left(-t^2/\tau\right)/\sqrt{2\pi\tau^2}$, the integral is explicitly computed and we obtain

$$k(x,y) = \frac{1}{\pi^{(p+1)/2}\tau^{1/2}|M|^{1/2}nm} \sum_{i=1}^{n}\sum_{j=1}^{m} \exp\left(-\frac{(x_i-y_j)^2}{2\tau}\right) \exp\left(-\frac{\|\boldsymbol{r}_i - \boldsymbol{s}_j\|_M^2}{4}\right). \tag{34}$$

The distance metric is also defined by

$$d(x,y) = \int_{\mathbb{R}^p} \mathrm{d}\boldsymbol{z} \int_{-\infty}^{\infty} \mathrm{d}t (\hat{\lambda}_x(t,\boldsymbol{z}) - \hat{\lambda}_y(t,\boldsymbol{z}))^2. \tag{35}$$

For estimating the intensity function, a simple Poisson process is assumed. This assumption is relaxed with more flexible models such as ETAS model (Ogata, 1988, 1998), where the intensity function is estimated by using R packages **SAPP** and **etasFLP**, for example. The extension of the intensity-based metric to support other form of intensity estimation such as Hawkes and ETAS models remains our important future work.

Co-occurrence metric

For extending co-occurrence metric based on counting the synchronous events, eq. (15) is replaced with a weighted counter

$$P_{ij} = \exp(-\|\boldsymbol{r}_i - \boldsymbol{s}_j\|_M^2) \times \begin{cases} 1, & 0 < x_i - y_j < \tau_{ij}, \\ 1/2, & x_i = y_j, \\ 0, & \text{otherwise.} \end{cases} \tag{36}$$

For extending the co-occurrence metric based on smoothed count of the synchronous events, eq. (18) is replaced with a weighted smoothed counter

$$dc_{\tau,M}(x_i) = \exp\left(-\|\boldsymbol{r}_i - \boldsymbol{s}_{(x_i)}\|_M^2\right) \times \exp\left(-\frac{|x_i - y_{(x_i)}|}{\tau}\right). \tag{37}$$

Inter event interval

For weighting the inter event interval by using marks associated with two MPP realizations x and y, we define index extraction operators as follows. We modify an MPP realization $x = \{(x_1, \boldsymbol{r}_1), \ldots, (x_n, \boldsymbol{r}_n)\}$ to include artificial events and marks corresponding to the beginning and end of the interval as

$$x = \{(x_0 = 0, \boldsymbol{r}_0 = \boldsymbol{0}), (x_1, \boldsymbol{r}_1), \ldots, (x_n, \boldsymbol{r}_n), (x_{n+1} = T, \boldsymbol{r}_{n+1} = \boldsymbol{0})\}. \tag{38}$$

Then we define operators

$$\begin{aligned} \underline{q} &: [0,T] \times \mathcal{X} \to \mathbb{R} \\ &\quad (t,x) \mapsto i, \quad \text{s.t.} \quad t \in [x_i, x_{i+1}], \end{aligned} \tag{39}$$

$$\begin{aligned} \overline{q} &: [0,T] \times \mathcal{X} \to \mathbb{R} \\ &\quad (t,x) \mapsto i+1, \quad \text{s.t.} \quad t \in [x_i, x_{i+1}]. \end{aligned} \tag{40}$$

The intermediate function eq.(23) is modified to take into account the dissimilarity of marks:

$$I_{xy}(t) = \frac{\min(v_x(t), v_y(t))}{\max(v_x(t), v_y(t))} \frac{\exp(-\|\boldsymbol{r}_{\underline{q}(t,x)} - \boldsymbol{s}_{\underline{q}(t,y)}\|_M^2) + \exp(-\|\boldsymbol{r}_{\overline{q}(t,x)} - \boldsymbol{s}_{\overline{q}(t,y)}\|_M^2)}{2}. \tag{41}$$

The distance and similarity are then calculated using eq. (25) and eq. (24).

The usage of the functions `fmetric`, `iipmetric`, `coocmetric`, and `ieimetric` do not change for

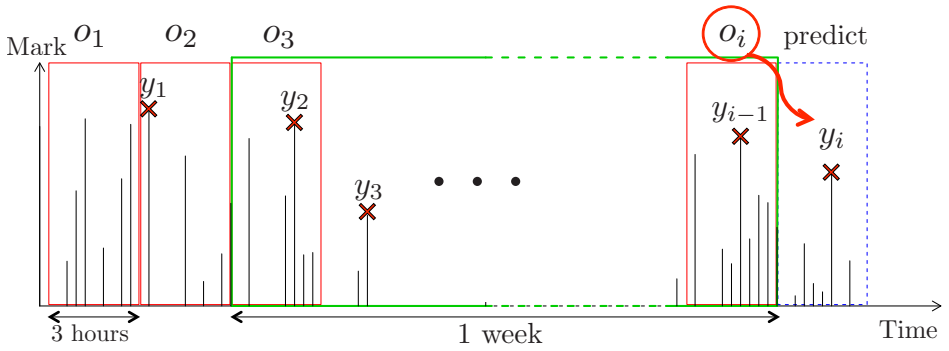

Figure 3: An illustrative diagram of the problem setting. Horizontal axis corresponds to time, and vertical axis shows marks. Though the dimension of the mark is four, it is shown as one-dimensional axis. The process is splited by a three hour window, and each window is assigned an output variable, which is the maximum magnitude in the next time window. Using the past one week data, the output variable of the next time window is predicted by the nearest neighbor regression.

marked point process data, except one additional argument `M`, which is the precision matrix M in eq. (27). By default, it is automatically set to the diagonal matrix with the diagonal elements equal to the reciprocal of the variance of corresponding marks of `S1` and `S2`. We can also specify the matrix M manually as

```
> fmetric(S1, S2, measure = "sim", M = diag(3))
```

where the number of marks is assumed to be three.

An example with the Miyagi20030626 data set

This section illustrates the use of the package with a simple experiment. We use the `Miyagi20030626` dataset contained within the package.

```
> library(mmpp)
> data(Miyagi20030626)
```

The dataset is composed of 2305 aftershocks of 26th July 2003 earthquake of M6.2 at the northern Miyagi-Ken Japan, which is a reparameterization of the `main2006JUL26` dataset from the **SAPP** package. Each record has 5 dimensions, `time`, `longitude`, `latitude`, `depth`, and `magnitude` of its hypo-center. The `time` is recorded in seconds from the main shock.

To illustrate the use of the package, we consider a simple prediction task. We first split the original dataset by a time-window of length $60 \times 60 \times 3$, which means that time interval of each partial point process splited by this window is three hours.

```
> sMiyagi <- splitMPP(Miyagi20030626, h = 60*60*3, scaleMarks = TRUE)$S
```

Then, the maximum magnitude in each partial point process realization is computed.

```
> ## target of prediction is the maximum magnitude in the window
> m <- NULL
> for (i in 1:length(sMiyagi)) {
+   m <- c(m, max(sMiyagi[[i]]$magnitude))
+ }
```

The task we consider is the prediction of the maximum magnitude in the *next* three hours using the past one week of data. We formulate this problem as a regression problem. Let the partial point process realization in the i-th window be o_i, and let the maximum magnitude in the $i+1$-th window be m_i. Then the problem is predicting m_{i+1} given o_{i+1} and the past $24 \times 7/3 = 56$ hours of data $\{(o_{i-\ell}, m_{i-\ell})\}_{\ell=0}^{55}$. See Figure 3 for an illustrative diagram of the problem setting.

```
> m <- m[-1]
> sMiyagi[[length(sMiyagi)]] <- NULL
> ## number of whole partial MPPs splited by a 3-hour time window
> N <- length(sMiyagi)
```

```
> ## training samples are past one week data
> Ntr <- 24*7/3
> ## number of different prediction methods
> Nd <- 10
```

For the purpose of illustrating the use of package and show the effect of different similarity metrics, we adopt the nearest neighbor regression. That is, given the current realization o_i, we find the most similar realization $o_j \in \{o_{i-l}\}_{l=0}^{55}$, and use the corresponding maximum magnitude m_j as the predictor for m_{i+1}. We use ten different similarity metrics supported in the package, and evaluate the mean absolute errors. The metrics used for this experiments are filter based metric in eq. (3), intensity inner product metric in eq. (10), co-occurrence with counting in eq. (16), co-occurrence with smoothed counting in eq. (19), and inter event interval metric in eq. (24), and their MPP extensions.

```
> err <- matrix(0, N - Ntr, Nd)
> colnames(err) <- c("f SPP","iip SPP","cooc (s) SPP","cooc (c) SPP","iei SPP",
+                    "f MPP","iip MPP","cooc (s) MPP","cooc (c) MPP","iei MPP")
```

The following code performs the above explained experiment.

```
> for( t in 1:(N - Ntr)) {
+   qid <- Ntr + t
+   q <- sMiyagi[[qid]]
+   ## simple PP
+   ## fmetric with tau = 1
+   sim2query <- NULL
+   for (i in 1:Ntr) {
+     sim2query <- c(sim2query, fmetric(q$time, sMiyagi[[qid - i]]$time))
+   }
+   err[t, 1] <- abs(m[qid] - m[t:(Ntr + t - 1)][which.max(sim2query)])
+   ## iipmetric with tau = 1
+   sim2query <- NULL
+   for (i in 1:Ntr) {
+     sim2query <- c(sim2query, iipmetric(q$time, sMiyagi[[qid - i]]$time))
+   }
+   err[t, 2] <- abs(m[qid] - m[t:(Ntr + t - 1)][which.max(sim2query)])
+   ## coocmetric (smooth) with tau = 1
+   sim2query <- NULL
+   for (i in 1:Ntr) {
+     sim2query <- c(sim2query, coocmetric(q$time, sMiyagi[[qid - i]]$time,
+                    type = "smooth"))
+   }
+   err[t, 3] <- abs(m[qid] - m[t:(Ntr + t - 1)][which.max(sim2query)])
+   ## coocmetric (count)
+   sim2query <- NULL
+   for (i in 1:Ntr) {
+     sim2query <- c(sim2query, coocmetric(q$time, sMiyagi[[qid - i]]$time,
+                    type = "count"))
+   }
+   err[t, 4] <- abs(m[qid] - m[t:(Ntr + t - 1)][which.max(sim2query)])
+   ## iei metric
+   sim2query <- NULL
+   for (i in 1:Ntr) {
+     sim2query <- c(sim2query, ieimetric(q$time, sMiyagi[[qid - i]]$time))
+   }
+   err[t, 5] <- abs(m[qid] - m[t:(Ntr + t - 1)][which.max(sim2query)])
+   ## marked PP with latitude, longitude, depth, and magnitude
+   ## fmetric with tau = 1
+   sim2query <- NULL
+   for (i in 1:Ntr) {
+     sim2query <- c(sim2query, fmetric(q, sMiyagi[[qid - i]]))
+   }
+   err[t, 6] <- abs(m[qid] - m[t:(Ntr + t - 1)][which.max(sim2query)])
+   ## iipmetric with tau = 1
+   sim2query <- NULL
+   for (i in 1:Ntr) {
+     sim2query <- c(sim2query, iipmetric(q, sMiyagi[[qid - i]]))
```

```
+   }
+   err[t, 7] <- abs(m[qid] - m[t:(Ntr + t - 1)][which.max(sim2query)])
+   ## coocmetric (smooth) with tau=1
+   sim2query <- NULL
+   for (i in 1:Ntr) {
+     sim2query <- c(sim2query, coocmetric(q, sMiyagi[[qid - i]], type = "smooth"))
+   }
+   err[t, 8] <- abs(m[qid] - m[t:(Ntr + t - 1)][which.max(sim2query)])
+   ## coocmetric (count)
+   sim2query <- NULL
+   for (i in 1:Ntr) {
+     sim2query <- c(sim2query, coocmetric(q, sMiyagi[[qid - i]], type = "count"))
+   }
+   err[t, 9] <- abs(m[qid] - m[t:(Ntr + t - 1)][which.max(sim2query)])
+   ## iei metric
+   sim2query <- NULL
+   for (i in 1:Ntr) {
+     sim2query <- c(sim2query,ieimetric(q, sMiyagi[[qid - i]]))
+   }
+   err[t, 10] <- abs(m[qid] - m[t:(Ntr + t - 1)][which.max(sim2query)])
+ }
> colMeans(err)
  f SPP      iip SPP     cooc (s) SPP      cooc (c) SPP     iei SPP
  0.7002634  0.6839529   0.7263602         0.6632930        0.7905148
  f MPP      iip MPP     cooc (s) MPP      cooc (c) MPP     iei MPP
  0.6839529  0.6317594   0.6643804         0.6622056        0.7698548
```

From this simple example, we can see that the prediction accuracy is improved by taking the marks into account.

Summary and future directions

mmpp is the first R package dedicated to calculation of the similarity and distance metrics for marked point process realizations. It provides implementation of several similarity metrics for simple point processes, originally proposed in the literature of neuroscience, and also provides extensions of these metrics to those for marked point processes.

A simple example of a real dataset presented in this paper illustrates the importance of taking the marks into account in addition to the event timing, and it also illustrates the possibilities of the package **mmpp** with a user guide for practitioners.

The development of **mmpp** package has only just begun. Currently, we are considering supporting burst sensitive and refractory period sensitive versions of `fmetric`, since these properties are commonly observed in both neural activities and seismic event recordings. In the current version of **mmpp**, for treating MPP, event timings and marks are assumed to be separable, and all the marks are simultaneously estimated by a kernel density estimator. This is a strong assumption and other possibilities for modeling MPP should be considered. For example, it is popular to group spatio-temporal events, i.e., event timings and locations, and treat marks such as magnitude in seismic events as purely *marks*. Then, the separability between marks and spatio-temporal events can be tested by using test statistics proposed in (Schoenberg, 2004; Chang and Schoenberg, 2011). Separability assumption offers computational advantages, though, it would miss the intrinsic structure and relationship between event timings and marks. In principle, the separability hypothesis should be tested before calculating the metrics. Frameworks for flexible modeling of marked sample sequences with statistical validation such as nonparametric test would be implemented in future version of **mmpp**. We are also considering to extend the intensity inner product metric to support other form of intensity estimation such as Hawkes and ETAS models.

Different similarity metrics capture different aspects of the point process realizations. Our final goal of the development of the package **mmpp** is in providing a systematic way to select or combine appropriate metrics for analysing a given point process realizations and certain task such as prediction of magnitude or clustering similar seismic events.

Acknowledgement

The authors are grateful to T. Iwata for helpful discussions and suggestions. The authors would like to express their special thanks to the editor, the associate editor and three anonymous reviewers whose comments led to valuable improvements of the manuscript. Part of this work is supported by JSPS KAKENHI Grant Number 25870811, 26120504, and 25120009.

Bibliography

A. Baddeley and R. Turner. spatstat: An R package for analyzing spatial point patterns. *Journal of Statistical Software*, 12(6):1–42, 1 2005. URL http://www.jstatsoft.org/v12/i06. [p1]

C.-H. Chang and F. Schoenberg. Testing separability in marked multidimensional point processes with covariates. *Annals of the Institute of Statistical Mathematics*, 63(6):1103–1122, 2011. doi: 10.1007/s10463-010-0284-7. [p10]

T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967. doi: 10.1109/TIT.1967.1053964. [p1]

E. Gabriel, B. S. Rowlingson, and P. J. Diggle. stpp: An R package for plotting, simulating and analyzing spatio-temporal point patterns. *Journal of Statistical Software*, 53(2):1–29, 4 2013. URL http://www.jstatsoft.org/v53/i02. [p1]

D. Harte. PtProcess: An R package for modelling marked point processes indexed by time. *Journal of Statistical Software*, 35(8):1–32, 2010. URL http://www.jstatsoft.org/v35/i08/. [p1]

A. G. Hawkes. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, 58(1): 83–90, 1971. doi: 10.2307/2334319. [p4]

H. Hino, K. Takano, Y. Yoshikawa, and N. Murata. mmpp: Various similarity and distance metrics for marked point processes, 2015. URL http://CRAN.R-project.org/package=mmpp. [p1]

C. Houghton. Studying spike trains using a van Rossum metric with a synapse-like filter. *Journal of Computational Neuroscience*, 26(1):149–155, 2009. doi: 10.1007/s10827-008-0106-6. [p3]

J. D. Hunter and J. G. Milton. Amplitude and frequency dependence of spike timing: Implications for dynamic regulation. *Journal of Neurophysiology*, 90(1):387–394, July 2003. doi: 10.1152/jn.00074.2003. [p4]

T. Iwata. Low detection capability of global earthquakes after the occurrence of large earthquakes: Investigation of the Harvard CMT catalogue. *Geophysical Journal International*, 174(3):849–856, 2008. doi: 10.1111/j.1365-246X.2008.03864.x. [p3]

Y. Y. Kagan. Short-term properties of earthquake catalogs and models of earthquake source. *Bulletin of the Seismological Society of America*, 94(4):1207–1228, 2004. doi: 10.1785/012003098. [p3]

E. R. Kandel, J. H. Schwartz, and T. M. Jessell. *Principles of Neural Science*. McGraw-Hill Medical, 4th edition, July 2000. [p1]

T. Kreuz, J. S. Haas, A. Morelli, H. D. Abarbanel, and A. Politi. Measuring spike train synchrony. *Journal of Neuroscience Methods*, 165(1):151–161, 2007. doi: 10.1016/j.jneumeth.2007.05.031. [p1, 5]

D. Lyttle and J.-M. Fellous. A new similarity measure for spike trains: Sensitivity to bursts and periods of inhibition. *Journal of Neuroscience Methods*, 199(2):296–309, 2011. doi: 10.1016/j.jneumeth.2011.05.005. [p3, 4]

Y. Ogata. Statistical models for earthquake occurrences and residual analysis for point processes. *Journal of the American Statistical Association*, 83(401):9–27, Mar. 1988. doi: 10.2307/2288914. [p7]

Y. Ogata. Space-time point-process models for earthquake occurrences. *Annals of the Institute of Statistical Mathematics*, 50(2):379–402, June 1998. doi: 10.1023/a:1003403601725. [p7]

A. R. C. Paiva, I. Park, and J. C. Príncipe. A reproducing kernel hilbert space framework for spike train signal processing. *Neural Comput.*, 21(2):424–449, Feb. 2009. doi: 10.1162/neco.2008.09-07-614. [p3]

R. Quian Quiroga, T. Kreuz, and P. Grassberger. Event synchronization: A simple and fast method to measure synchronicity and time delay patterns. *Phys. Rev. E*, 66:041904, Oct 2002. doi: 10.1103/PhysRevE.66.041904. [p1, 4]

R.-D. Reiss. *A Course on Point Processes*. Springer Series in Statistics. Springer, 1993. [p3]

B. Rowlingson and P. Diggle. Splancs: Spatial point pattern analysis code in S-plus. *Computers & Geosciences*, 19(5):627–655, 1993. doi: 10.1016/0098-3004(93)90099-Q. [p1]

F. P. Schoenberg. Testing separability in spatial-temporal marked point processes. *Biometrics*, 60(2): 471–481, 2004. doi: 10.1111/j.0006-341X.2004.00192.x. [p10]

S. Schreiber, J. Fellous, D. Whitmer, P. Tiesinga, and T. Sejnowski. A new correlation-based measure of spike timing reliability. *Neurocomputing*, 52–54(0):925 – 931, 2003. doi: 10.1016/S0925-2312(02)00838-X. Computational Neuroscience: Trends in Research 2003. [p1, 3]

J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, New York, NY, USA, 2004. [p1]

D. L. Snyder and M. I. Miller. *Random Point Processes in Time and Space*. Springer-Verlag, New York, NY, USA, 2nd edition, 1991. [p1]

S. Suzuki, Y. Hirata, and K. Aihara. Definition of distance for marked point process data and its application to recurrence plot-based analysis of exhange tick data of foreign currencies. *International Journal of Bifurcation and Chaos*, 20(11):3699–3708, 2010. doi: 10.1142/S0218127410027970. [p2]

M. C. W. van Rossum. A novel spike distance. *Neural Computation*, 13(4):751–763, 2001. doi: 10.1162/089976601300014321. [p1, 3]

J. D. Victor. Spike train metrics. *Current Opinion in Neurobiology*, 15:585–592, October 2005. doi: 10.1016/j.conb.2005.08.002. [p1, 2]

Hideitsu Hino
Graduate School of Systems and Information Engineering, University of Tsukuba
1–1–1 Tennoudai, Tsukuba, Ibaraki, 305–8573
Japan
hinohide@cs.tsukuba.ac.jp

Ken Takano
School of Science and Engineering, Waseda University
3–4–1 Ohkubo, Shinjuku-ku, Tokyo 169–8555
Japan
ken.takano@toki.waseda.jp

Noboru Murata
School of Science and Engineering, Waseda University
3–4–1 Ohkubo, Shinjuku-ku, Tokyo, 169–8555
Japan
noboru.murata@eb.waseda.ac.jp