Space-Time Clustering with Stability Probe while Riding Downhill

Xin Huang
University of Texas at Dallas, USA
xxh130130@utdallas.edu

Iliyan R. Iliev
University of Southern Mississippi, USA
iliyan.iliev@usm.edu

Alexander Brenning
Friedrich Schiller University, Germany
alexander.brenning@uni-jena.de

Yulia R. Gel
University of Texas at Dallas, USA
ygl@utdallas.edu

ABSTRACT

We propose a new data-driven procedure for optimal selection of tuning parameters in dynamic clustering algorithms, using the notion of stability probe. Due to the shape of the stability probe dynamics, we refer to the new clustering stability procedure as Downhill Riding (DR). We study final sample performance of DR in conjunction with DBSCAN and TRUST in application to clustering synthetic times series and yearly temperature records in Central Germany.

Keywords

Automatic parameter selection, dynamic clustering, clustering stability

1. INTRODUCTION

Clustering of time series has received considerable attention in the last two decades both in data mining and statistical literature [32, 28, 39, 52, 19], with applications ranging from finance and communication sciences to neuroscience and geology. Most recently, the rampant growth of various remote sensing technologies has resulted in a spike of interest in space-time data mining and particularly clustering of environmental time series and spatio-temporal processes [37, 31, 47, 38]. However, many currently existing clustering procedures for space-time data are either based solely on geographical proximity, which does not account for drifts in space-time data distribution, or are restricted to a relatively small domain to avoid high spatial heterogeneity [45, 6]. Furthermore, the number of possible clusters is often fixed a-priori, which substantially limits the utility of such clustering procedures in environmental applications that are typically characterized by spatio-temporal non-stationarity and non-separability [23, 42].

Our initial interest in the topic was motivated by studies of the impact of climate change on insurance claim dynamics and early recognition of areas with the highest vulnerability to adverse weather, particularly, the so-called *"normal"* weather, with a low individual but high cumulative impact [for overview see, e.g., 26, 41, 43, 44, and references therein]. Remarkably, attribution analysis of such "normal" weather on the insurance industry is largely unexplored [14, 12, 44]. Since there exist multiple factors contributing to elevated insurance risks, e.g., city infrastructure, building codes, sociodemographics, landscape, as well as numerous latent variables, areas that are similar in their sensitivity to adverse weather are not necessarily close geographically. At the same time, the number of clusters, or areas with similar levels of vulnerability, is unknown and can vary over different time periods driven, for example, by the El Niño-Southern Oscillation (ENSO) cycles and other forcings. Moreover, the choice of optimal number of clusters is a longstanding problem in climate sciences [see, for instance, discussion by 46, 34]. How can we approach this problem then?

Remarkably, the number of dynamic data-driven clustering procedures for space-time data that allow the number, shape and distributional properties of clusters to vary, still remains quite limited. However, this research direction has received a flare of interest in recent years [22, 11, 4]. Two such dynamic clustering procedures are an efficient space-time data mining procedure (TRUST) of [13] that is based on interleaving spatial clustering and temporal trend detection; and a hierarchical spectral merger algorithm to cluster brain connectivity [19]. Alternatively, we can adjust various density-based clustering procedures such as DBSCAN [29, 18], OPTICS [2], DENCLUE [27] etc, to a space-time context.

Despite the potential of these dynamic clustering procedures, the price for their flexibility is usually a set of parameters that control clustering performance and are to be user-specified – for instance, the maximum radius of the neighborhood Eps in DBSCAN [18]; the steepness parameter ξ in OPTICS [2]; the value similarity threshold δ in TRUST [13]; and the kernel smoothing parameter h in DENCLUE [27]. The choice of these parameters can noticeably impact the number and shape of detected clusters, and ideally should be approached in an objective manner.

SIGKDD'16 Workshop on Mining and Learning from Time Series (MiLeTS) August 13–17, 2016, San Francisco, California, USA

ACM ISBN 123-4567-24-567/08/06...$15.00

DOI: 10.1145/1235

In this paper we propose a new data-driven and computationally efficient procedure for optimal selection of clustering tuning parameters using a *clustering stability probe*. Our approach is rooted in the so-called clustering (in)stability criteria [8, 7], based on the intuitive idea that if we randomly split our data into two non-overlapping subsets, then a good clustering algorithm should deliver similar clustering results. Hence, the idea is to perform multiple splits, using crossvalidation, and search for the case with the most similar (on average) partitions. Clustering (in)stability has gained an increased interest in machine learning and statistical sciences for identification of the optimal number of clusters, typically in conjunction with K-means [17, 8, 7, 51, 20].

Our approach, however, has two main advantages over conventional clustering (in)stability. First, instead of measuring the distance between each two partitions, which is a very computationally demanding if not prohibitive step, we select a *clustering probe* and define stability only on the basis of the distance between univariate probes. In this paper we are primarily interested in the utility of a number of clusters as a probe. Second, we advance the idea of a clustering (in)stability criterion to choose the optimal parameters in TRUST, DBSCAN and other dynamic clustering algorithms. Due to the shape of the stability probe dynamics, we refer to the new clustering stability procedure as Downhill Riding (DR). We outline the theoretical properties of the new DR procedure and evaluate its finite sample performance for dynamic clustering using synthetic time series. We also illustrate the DR procedure in application to dynamic cluster detection in yearly temperature records among 167 stations in Central Germany over 1951–2010.

The paper is organized as follows. The new stability probe approach, Downhill Riding (DR) algorithm, is presented in Section 2. In Section 3 we discuss TRUST and DBSCAN, i.e. the two primary clustering methods we focus on. The proposed Downhill Riding (DR) algorithm is then evaluated by extensive numerical studies in Section 4. In Section 5 we illustrate applications of new Downhill Riding (DR) procedure to analysis of the yearly temperature records in Central Germany. The paper is concluded by discussion in Section 6.

2. DOWNHILL RIDING PROCEDURE WITH CLUSTERING STABILITY PROBE

Preliminaries Let Ω_N be the observed data set that contains N multivariate items, i.e. $\Omega_N = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_N\}$ and $\mathbf{a}_i = (a_{i1}, a_{k2}, \ldots, a_{it})^T$, $1 \le i \le N$. (For instance, $\mathbf{a}_i$ may represent an i-th observed time series up to time point t.) Our goal is to partition Ω into subsets $C_1, C_2, \ldots, C_K$ such that $\bigcup_{k=1}^K C_k = \Omega$ and $C_i \bigcap C_j = \emptyset$ for $i \neq j$. A number of clusters K is unknown a-priori. To achieve such partition C, we use a clustering algorithm $M(\Omega, \nu)$, where ν is a generic notation for a set of tuning parameters that controls partitioning of Ω, and is usually pre-specified by user. (For simplicity, we start from a case of a single parameter ν but the idea can be extended to a more general case.) The resulting clustering performance is typically evaluated using standard information criteria such as the Normalized Mutual Information (NMI), the Jaccard Index, Rand Index etc [50, 35].

The Downhill Riding (DR) Algorithm As discussed by [29], choice of a tuning parameter ν, such as Eps in DBSCAN, steepness parameter ξ in OPTICS etc may substantially impact the resulting partitioning C. How can we choose ν in an objective manner while achieving the optimal clustering performance? In a nutshell, our intuitive idea is to look at the stability of a number of detected clusters as indicator for the underlying "ground truth".

In particular, let us select a number of clusters $\hat{K}$ as a clustering probe; obviously, $\hat{K}$ is a function of ν and Ω_N (i.e., $\hat{K}(\nu, \Omega_N)$). Suppose that Ω_N is a sufficiently large data set such that each true cluster is well represented in Ω_N. We now randomly split Ω_N into two subsets $\Omega^1_{N/2}$ and $\Omega^2_{N/2}$ of equal cardinality. If we partition $\Omega^1_{N/2}$ and $\Omega^2_{N/2}$ using the same clustering algorithm $M(\nu, \cdot)$, we intuitively expect that, if the tuning parameter ν is selected correctly, such partitions should be relatively similar, homogeneous or, at least,

$$|\hat{K}(\nu, \Omega^1_{N/2}) - \hat{K}(\nu, \Omega^2_{N/2})| \approx 0. \tag{1}$$

Hence, by viewing (1) as a function of ν, we can look at its minimum as indicator of correctly selected parameter ν. We define the function in (1) as the Cluster Deviation:

$$CD(\nu) = |\hat{K}(\nu, \Omega^1_{N/2}) - \hat{K}(\nu, \Omega^2_{N/2})|. \tag{2}$$

However, there exist two additional extreme scenarios when $CD(\nu) \approx 0$: first, when all $N/2$ items in $\Omega^1_{N/2}$ and $\Omega^2_{N/2}$ are partitioned into $N/2$ individual clusters; and second, when all data are grouped into a single cluster. Hence, we search for the local minimum in $CD(\nu)$ as the indicator of "truth". Since estimation uncertainty due to a single split of Ω_N into $\Omega^1_{N/2}$ and $\Omega^2_{N/2}$ might be high, we use the V-fold crossvalidation procedure with multiple splits (see Algorithm 1).

Note that the idea is intrinsically linked to the notion of clustering (in)stability [8, 7, 10, 49, 51]. However, in contrast to the earlier stability approaches of [17, 51, 20], we do not aim to evaluate closeness of cluster assignments of each observation but focus on a distance between probes.

```
Algorithm 1: Downhill Riding (DR)
  Input  : $\Delta = \{\nu_n, n = 1, 2, \ldots, M\}$, $\Omega$, $B$
  Output: optimal empirical estimate $\hat{\nu}^e$
for each $\nu_n \in \Delta$ do
    Compute $ACD(\nu_n)$;
    $\mathbb{A} \leftarrow \{\mathbb{A} \cup ACD(\nu_n)\}$;
end
Find $\nu^*_n$ such that $ACD(\nu^*_n)$ is a local minimum in $\mathbb{A}$;
if the number of $\nu^*_n$ equals 1 then
    return $\nu^*_n$;
else
    return $\arg\min \nu^*_n$;
end
```

We now define a new measure for the stability of a clustering algorithm, Average Cluster Deviation (ACD), as a function of the tuning parameter ν:

$$ACD(\nu) = \frac{1}{B} \sum_{b=1}^{B} \left| \hat{K}(\nu, \Omega^1_{N/2}, b) - \hat{K}(\nu, \Omega^2_{N/2,b}) \right|, \tag{3}$$

where B is the number of splits in cross-validation, and $\hat{K}(\nu, \Omega^1_{N/2}, b)$ and $\hat{K}(\nu, \Omega^2_{N/2,b})$ are the number of clusters delivered by M in application to the b-th splits of $\Omega^1_{N/2}, b)$ and $\Omega^2_{N/2}, b)$. The *optimal empirical estimate* $\hat{\nu}^e$ is the argument of the local minimum of ACD.

To get an initial validation insight to this idea, we now consider a relationship between ACD and NMI, in application to the TRUST clustering algorithm. Fig. 1 shows the aggregated dynamics, while Fig. 2 depicts a realization for single synthetic data set. Both Fig. 1 and 2 suggest that the local minimum for ACD indeed is well aligned with the global maximum of NMI. Note that as expected, ACD is close to 0 at lower or higher values of ν. Lower values of ν tend to correspond to a higher number of clusters, up to an extreme case of each sample forming a single cluster, which leads to lower (or even zero) values of ACD but also lower NMI. At the same time, higher values of ν lead to a lower number of clusters, up to an extreme case of all samples being in one group, which again leads to lower (or even zero) values of ACD but low NMI. Based on the $\bigwedge$-shape of ACD and our search for its right-hand side minimum, we call our algorithm a *Downhill Riding (DR)* procedure.

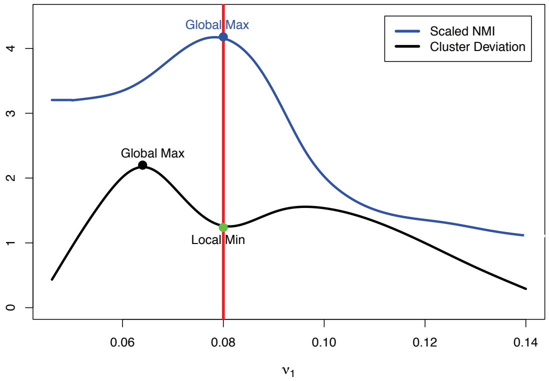


Figure 1: Aggregated dynamics of the Downhill Riding (DR) procedure.

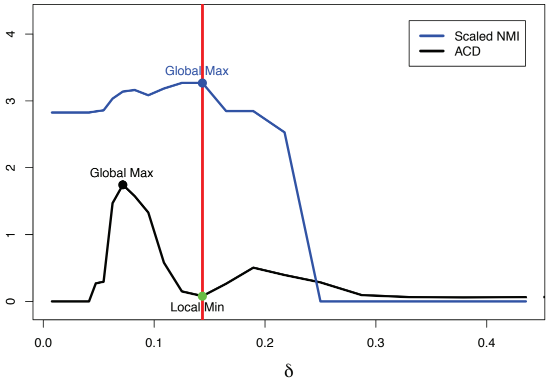


Figure 2: Dynamics of the Downhill Riding (DR) procedure for a single data realization with clustering by TRUST.

Asymptotics Properties To proceed with the theoretical properties of the DR procedure, we adopt notions of clustering consistency and stability discussed by [8, 10].

Definition Assume that the observed data Ω_N has been sampled from an underlying population Ω according to some probability measure P. Let Q be a clustering loss function on the set $\mathbb{S}$ of all partitions of the population Ω. Let $C^*(\Omega)$ be a unique minimizer of Q. A clustering algorithm $M(\nu)$ is called *asymptotically consistent* if it delivers a partition $C(\Omega_N)$ such that $Q(C(\Omega_N))$ converges to $Q(C^*(\Omega))$ as $N \to \infty$.

Proposition 1. *Let M be an asymptotically consistent clustering algorithm such that $Q(C(\Omega_N))$ converges to $Q(C^*(\Omega))$ at rate $r_{N,k}$ where $r_{N,k}$ is a nonincreasing sequence of positive numbers. Let ν^* be a a value of a tuning parameter ν that delivers $M(\Omega) = C^*$. Then, in probability*

$$\hat{\nu}^0 \underset{N\to\infty}{\to} \nu^*,$$

where $\hat{\nu}^o$ is the argument of the local minimum of the oracle loss function, or the Expected Cluster Deviation,

$$E\left|\hat{K}(\nu, \Omega^1_{N/2}, b) - \hat{K}(\nu, \Omega^2_{N/2}, b)\right|.$$

Proof of Proposition 1 is approached similarly to Theorem 1 of [51].

Now, let $\hat{\nu}^e$ be the empirical counterpart of $\hat{\nu}^o$, that is,

$$\hat{\nu}^e = \arg\min \frac{1}{B}\sum_{b=1}^{B}\left|\hat{K}(\nu, \Omega^1_{N/2}, b) - \hat{K}(\nu, \Omega^2_{N/2}, b)\right|.$$

The next proposition states that $\hat{\nu}^e$ and $\hat{\nu}^o$ are asymptotically negligible.

Proposition 2. *Let $\Omega_N = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_N\}$ be a sample from an underlying population Ω, where $\mathbf{a}_i$, $i = 1, \ldots, N$ are mutually independent random vectors. Let M be an asymptotically consistent clustering algorithm, and let $C^*(\Omega) = \{C^*_1, \ldots, C^*_K\}$ be the true clustering of Ω. Let N_i be the number of items in Ω_N corresponding to the true cluster C^*_i in Ω. Then, if $K \ll N_i$, $N_i \to \infty$, and $N \to \infty$*

$$|\hat{\nu}^e - \hat{\nu}^0| \to 0, \tag{4}$$

in probability.

The proof of Proposition 2 is based on the Chebyshev inequality and approached in a similar manner as Theorem 3 by [9].

3. CLUSTERING ALGORITHMS

We now discuss the two main clustering algorithms that illustrate the application of the Downhill Riding procedure – TRUST and DBSCAN.

3.1 The TRUST Algorithm

The TRUST algorithm is an unsupervised clustering algorithm designed for space-time data streams by Ciampi [13, 3]. Specifically, TRUST integrates spatial clustering and temporal trend detection with a goal to continuously group geo-referenced data according to a similar temporal trajectory in time. TRUST has the following advantages:

- as opposed to k-means, no number of clusters needs to be pre-specified a-priori;

- the algorithm is applicable to arbitrarily shaped clusters;
- the algorithm can dynamically detect the drift of space-time data distributions by using a sliding window moving from past to recent.

The core approach of TRUST is based on the extension of a sliding-window model to multiple spatially distributed data sources. In particular, let observations collected at multiple spatial devices at the same time point be referred as a *layer*, and several consecutive layers constitute a *slide* (see Figure 3). A sliding *window* consists of multiple consecutive layers and moves from past to recent. The main steps of TRUST is first to detect trend-clusters over the slide time (i.e., *slide-level* clustering), based on closeness (homogeneity) of sources within a layer where level of homogeneity is controlled by a threshold δ; and then to approximate trend-clusters by combining the slide-level trend cluster sets (i.e. *window-level* clustering). Window and slide sizes can be defined via expert knowledge, e.g. corresponding to climate cycles; however, there exists no objective way to select the key parameter δ, which serves as a particular motivation of our DR approach.

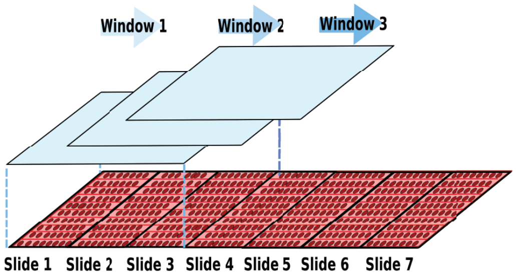


Figure 3: The framework of TRUST.

3.2 The DBSCAN Algorithm

Density-Based Spatial Clustering of Applications with Noise (DBSCAN) of [18, 29] is one of the most widely used clustering algorithms and is the 2014 SIGKDD Test of Time Award. DBSCAN also does not require a pre-defined number of clusters and can detect arbitrarily shaped clusters like TRUST. The core idea of DBSCAN is as follows: given a set of points in some space, it groups together points that are in a high-density region (i.e., neighbors of the points are close to each other), and marks points as outliers if they lie alone in low-density regions (whose nearest neighbors are far away). DBSCAN requires two parameters: the maximum radius of the neighborhood, Eps and the minimum number of points required to form a dense region, minPts. Selection of both Eps and minPts is typically performed in a subjective manner.

4. NUMERICAL EXPERIMENTS

4.1 Benchmark Iris Data

We start from evaluating the performance of the DR procedure in application to DBSCAN. We compare the DR performance against DBSCAN with a conventionally pre-selected Eps [18] as well as against OPTICS [2], using the benchmark Iris data [21, 30]. Despite the popularity of DBSCAN in spatial clustering, the selection of its key parameter Eps is mostly based on heuristic and subjective methods such as, for instance, as a *sorted k-dist graph* [18]. (The sorted k-dist graph algorithm is implemented in R by [25]). As a generalized version of DBSCAN, the OPTICS algorithm first constructs a reachability plot from the data and then automatically detects clusters, based on an extraction algorithm without a need to specify Eps. The algorithm used for automatic extraction of clusters in OPTICS in this paper is based on the approach of [40, 53].

The Iris data contain 150 samples with 4 variables for 3 clusters. The conventionally pre-selected Eps for DBSCAN, using the sorted k-dist graph, is set to 0.5. As Table 1 indicates, for each possible value of parameter MinPts, DBSCAN with Downhill Riding outperforms DBSCAN with the conventional selection and OPTICS, providing higher NMI values.

Table 1: Performance of DBSCAN with Eps-opt selected using Downhill Riding and with Eps-kdist selected by conventional *k-dist graph* and OPTICS.

	MinPts	Eps-opt	Eps-kdist	OPTICS
NMI	5	0.76	0.61	0.75
NMI	6	0.76	0.60	0.75
NMI	7	0.76	0.58	0.75

4.2 Synthetic data

To further evaluate the performance of the Downhill Riding procedure, we proceed with a series of Monte Carlo simulations. The first set of simulations estimate the finite sample performance of Downhill Riding when used with DBSCAN and TRUST compared to the highest empirically achievable performance of the two algorithms without our procedure. We then proceed with simulations that compare our automatic procedure to algorithms that have the benefit of a-priori knowledge of the clustering parameters.

Automatic selection study For the first set of simulations, we produce a data stream of 20 time series, denoted by Y, and obtained by sequencing 2 consecutive periods (slides) of 80 time points (layers). The cluster configurations that were used for the generation of Y are shown in Table 2.

Table 2: Cluster configuration of 20 time series on 2 time periods.

Time series model	**Time period 1**	**Time period 2**
$AR(1), \phi_1 = 0.5,$ $\varepsilon_t \sim N(0,1)$	*Time series* 2, 11, 14, 16	*Time series* 7, 15, 17, 18, 5
$AR(2), \phi_1 = 0.6, \phi_2 = 0.2,$ $\varepsilon_t \sim N(1,1)$	*Time series* 6, 7, 13, 17	*Time series* 2, 8 12, 13, 14
$MA(1), \theta_1 = 0.7,$ $\varepsilon_t \sim N(2,1)$	*Time series* 1, 8, 9, 19	*Time series* 3, 11, 16, 19, 20
$MA(2), \theta_1 = 0.8, \theta_2 = -0.6,$ $\varepsilon_t \sim N(3,1)$	*Time series* 3, 10, 12, 18	-
$ARMA(1,1), \phi_1 = 0.8, \theta_1 = 0.2,$ $\varepsilon_t \sim N(4,1)$	*Time series* 4, 5, 15, 20	*Time series* 1, 4, 6, 9, 10

Tables 3 and 4 present the finite sample performance of Downhill Riding when used with TRUST and DBSCAN measured in terms of ACD and NMI. We calculate ACD and NMI with different values of δ (for TRUST) and Eps (for DBSCAN). The number of detected clusters for TRUST corresponds to slide-level clustering by setting slide size $p = 80$; and for DBSCAN, this corresponds to Euclidian metric.

Table 3: Performance of TRUST with δ_{opt} selected by Downhill Riding and TRUST with δ_{oracle}. Slide-level trend continuity threshold θ is 0.9, and slide size p is 80. Number of Monte Carlo experiments is 100. Number of cross-validation splits T is 100.

	δ_{opt}	δ_{oracle}
Average NMI	$0.82_{(0.01)}$	$0.83_{(0.02)}$
Average ACD	$0.11_{(0.10)}$	$0.82_{(0.66)}$

Table 4: Performance of DBSCAN with Eps$_{opt}$ selected by Downhill Riding and DBSCAN with Eps$_{oracle}$. MinPts is 3. Number of Monte Carlo experiments is 100. Number of cross-validation splits T is 100.

	Eps$_{opt}$	Eps$_{oracle}$
Average NMI	$0.80_{(0.06)}$	$0.88_{(0.04)}$
Average ACD	$0.64_{(0.09)}$	$0.73_{(0.09)}$

We find that for TRUST, δ_{opt}, selected using Downhill Riding is close to δ_{oracle} and yields NMI comparable to the highest empirically achievable NMI (Table 3). The results are similar for DBSCAN, where the Eps selected by Downhill Riding produces NMI close to the highest possible (Table 4). The findings show that our automatic parameter selection procedure tends to deliver close to the empirically achievable levels of TRUST and DBSCAN despite the lack of a-priori knowledge.

Comparative study The second set of simulations compare the finite sample performance of TRUST and DBSCAN with automatic parameter selection using Downhill Riding to a number of clustering algorithms that are provided with the true number of clusters. The competing algorithms include k-means and a number of conventional hierarchical clustering approaches using different feature-based distance measures [36]. We simulate a data stream of 20 time series, denoted by Y, obtained by sequencing 4 consecutive periods of 120 time points (layers), generated according to the cluster configurations reported in Table 5.

The clustering algorithms are applied on each period of Y. For TRUST, we set the layer size p as 40, the slide continuity threshold θ as 0.9, the window size ω as 3 (step size as 3), and the window continuity threshold ϵ as 0.6. For DBSCAN, the parameter MinPts in DBSCAN is set as 3. The true number of clusters is set as known for the competing algorithms in each period. The dissimilarity measures in k-means and DBSCAN are Euclidean distance; the linkage function in all feature-based hierarchical clustering is complete linkage.

The clustering performance is evaluated on each of the 4 periods – measuring the amount of agreement between the true cluster partition $G = \{G_1, \ldots, G_{20}\}$ (the "ground-truth"), which is known, and the experimental cluster solution $A = \{A_1, \ldots, A_{20}\}$ yielded by the clustering algorithms. Validation measures include standard clustering external criteria (NMI and Jaccard Index) and internal criteria (Mean Absolute Percentage Error (MAPE)).

The comparative results are shown in Figure 4. Using three evaluating measures produced consistent results: based on NMI (Figure 4a), TRUST ranked 4th, 2nd, 5th, 4th among 9 competitors in the four time periods, respectively. The two most competitive clustering algorithms are K-Means and CORT. However, both are allowed to know the true number of clusters and thus are advantaged over DBSCAN and TRUST. Nevertheless, TRUST still outperforms NP, LNP, ACF and COR, which also have the informational advantage. With the additional information, k-means outperforms TRUST in all four time periods. Temporal correlation-based (CORT) and integrated periodogram-based (IP) clustering outperform TRUST except in time period 2. Similar performance is observed with Jaccard Index (Figure 4b) and MAPE (Figure 4c) (For MAPE, smaller values indicate better performance). TRUST outperforms DBSCAN (both are using Downhill Riding), in all four time periods using all three measures. In general, DBSCAN delivers lower performance than TRUST, which can be explained by the fact that DBSCAN primarily focuses on spatial rather than spatio-temporal clustering.

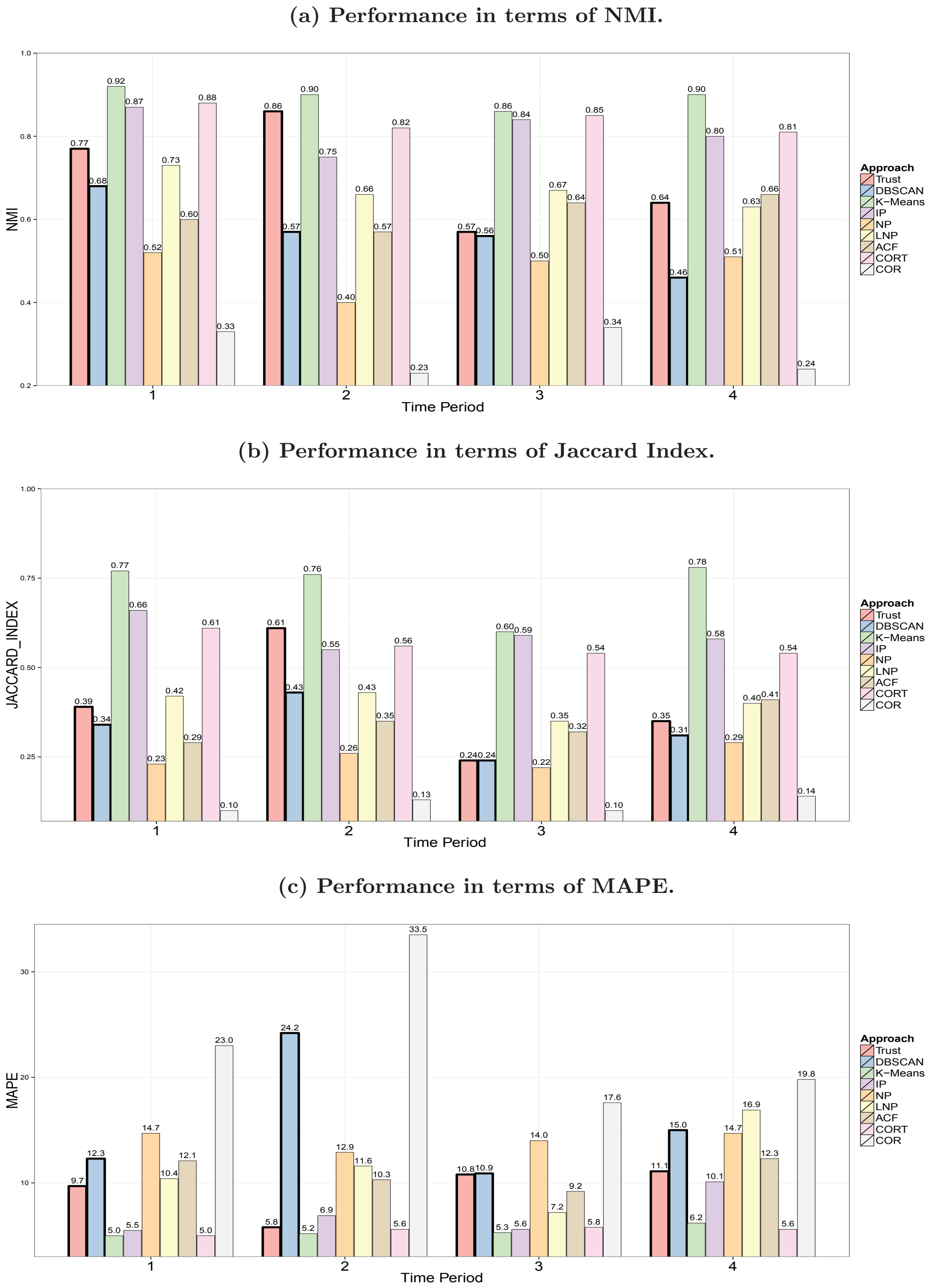
The performance of Downhill Riding with TRUST is competitive, especially given that the competing algorithms have an advantage of knowing the true number of clusters in each time period, thus operating with more information about the data. Despite this, using our automatic selection procedure, TRUST still outperforms some of its competitors in most of the time periods, such as normalized periodogram-based (LNP, NP), autocorrelation-based (ACF), and correlation-based clustering (COR). Combined with the results from the benchmark Iris data study above, the results show that our automatic procedure is not just on par with the competition that has the informational advantage, but at times better. These findings imply that the new DR algorithm can be particularly useful in studies where there is no knowledge of the parameters or number of clusters, such as when exploring environmental, insurance, or social science data, without imposing considerable performance trade-offs.

The study has been conducted using statistical software R [1] on a machine with 2.83 GHz Intel Xeon processor and 16 GB RAM. With synthetic data, the average elapsed time of one experiment with TRUST is 34.3 min with 556.5 MB average virtual memory usage; and the average elapsed time of one experiment with DBSCAN is 5 s with 566.3 MB average virtual memory usage. For the comparative study, the average elapsed time of one experiment is 3.3 h with 616.7 MB average virtual memory usage, and for the Iris real data DBSCAN comparison, the average elapsed time of one experiment is 4.5 s with 0.3 MB average virtual memory usage.

5. CASE STUDY

Observed temperature data We applied TRUST and DBSCAN to yearly temperature records from 167 weather stations in Central Germany in a 60-year period 1951–2010 [16]. Analyzing temperature data for such a long period can provide some important insights into climate change and the differences of these effects in the various geographic areas. The controlling parameters δ for TRUST and Eps for DBSCAN are set as 0.036 and 6.5 by "Downhill Riding". We select 15-year intervals as a time period to perform clustering since the climate of Europe exhibits cycles of 12-16 years [48]. Thus, the 60 year temperature data is segmented into 4 non-overlapped time periods, each of which is clustered by TRUST and DBSCAN, respectively. We set the layer size p as 40, the window size ω as 3 (with step size as 3), and set MinPts of DBSCAN as 3.

[1] The R code is available in a statistical software R package funtimes [33]


(a) Performance in terms of NMI.

(b) Performance in terms of Jaccard Index.

(c) Performance in terms of MAPE.

Figure 4: Comparative study over four time periods using three measures. Black boxes show algorithms use Downhill Riding.

Table 5: Cluster configuration of 20 time series on 4 time periods.

Time series model	Time period 1	Time period 2	Time period 3	Time period 4
$AR(1), \phi_1 = 0.5,$ $\varepsilon_t \sim N(0,1)$	*Time series* 2, 11, 14, 16	*Time series* 5, 7, 15, 17, 18	*Time series* 3,4, 10,16	*Time series* 5, 7, 10 ,14, 16
$AR(2), \phi_1 = 0.3, \phi_2 = -0.3,$ $\varepsilon_t \sim N(1,1)$	-	-	*Time series* 9,12 13, 18	*Time series* 1, 4, 11, 17, 18
$AR(2), \phi_1 = 0.6, \phi_2 = 0.2,$ $\varepsilon_t \sim N(2,1)$	*Time series* 6, 7, 13, 17	*Time series* 2, 8, 12, 13, 14	-	*Time series* 6, 8, 13, 15, 19
$MA(1), \theta_1 = 0.7,$ $\varepsilon_t \sim N(3,1)$	*Time series* 1, 8, 9, 19	*Time series* 3, 11, 16, 19, 20	*Time series* 2, 11, 14,15	-
$MA(2), \theta_1 = 0.8, \theta_2 = -0.6,$ $\varepsilon_t \sim N(4,1)$	*Time series* 3, 10, 12, 18	-	*Time series* 5, 7 8, 17	*Time series* 2, 3, 9, 12, 20
$ARMA(1,1), \phi_1 = 0.8, \theta_1 = 0.2,$ $\varepsilon_t \sim N(5,1)$	*Time series* 4, 5, 15, 20	*Time series* 1, 4, 6, 9, 10	*Time series* 1, 6 19, 20	-

The clustering results based on TRUST and DBSCAN show similar patterns where elevation is a dominant factor: elevation of weather stations – one of the key factors in temperature differences – is found to be relatively homogeneous within each cluster. Figure 5 shows the results of TRUST clustering in time period 4 in a topographic map. The contour lines show places of equal elevation. Different clusters are labeled with different colors. The weather stations in the yellow cluster are mostly located in areas below 300 m; while the weather stations in the red cluster are mostly located in areas around 500-600 m. The fact that elevation strongly affects temperature is well known in climate sciences. Hence, we are interested to investigate potential less explicit latent factors affecting temperature dynamics and segmentation.

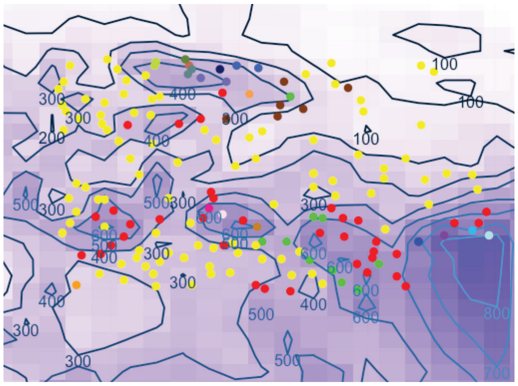


Figure 5: Clustering of weather stations in time period 4 (year 46-60) by TRUST.

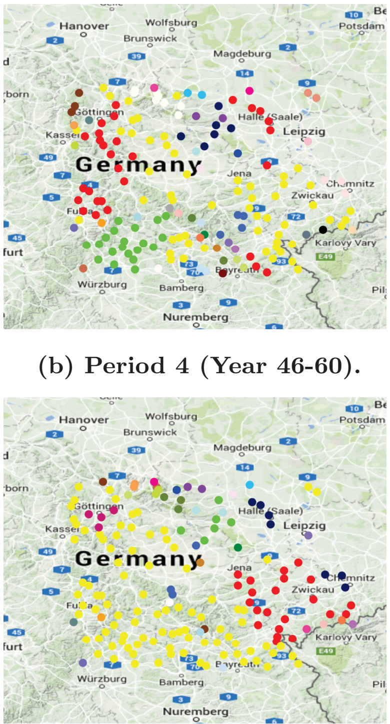
Elevation Scaled temperature data We now consider elevation scaled temperature where the impact of elevation has been removed according to [5, 15]. In particular, let X be elevation and Y be temperature, then:

$$Y_n^t = \beta_{t0} + \beta_{t1} X_n + \epsilon_n^t, n = 1, 2, \ldots, 167, t = 1, 2, \ldots, 60 \quad (5)$$

The residuals ϵ_n^t $(n = 1, 2, \ldots, 167, t = 1, 2, \ldots, 60)$ from linear regression are combined into a new data set where TRUST and DBSCAN are applied with the same framework setting as in previous temperature data clusterings. Optimal Downhill Riding parameters are δ and Eps, set as 0.1 and 0.65.

The resulting patterns are different from the ones observed in the temperature clustering. Figure 6a and Figure 6b show the clustering results for periods 1 and 4 by TRUST in terrain maps, respectively. Climate stations in Halle (Saale) area are grouped together in both time periods 1 and 4 (red dots in Figure 6a and navy dots in Figure 6b). Average residuals of the two clusters are 0.2 and 0.6 respectively, which makes sense because Halle (Saale) area corresponds to the dry region of Central Germany. In addition, a handful of stations north/northwest of Karlovy Vary show unique patterns: individual weather stations form their own mini-clusters. For example, black and grey dots in Figure 6a, and light pink, rosy dots in Figure 6b. These weather stations are all in a part of a mountain range called the Ore Mountains where they are probably located in fairly unique topographic situations, e.g. mountain top, or valley. In mountain areas, the orientation of a valley can have a large influence on the movement of air masses, so valleys of different orientations may be distinct enough to be placed in different clusters. Similar patterns are observed in the DBSCAN clustering results depicted in Figure 7.

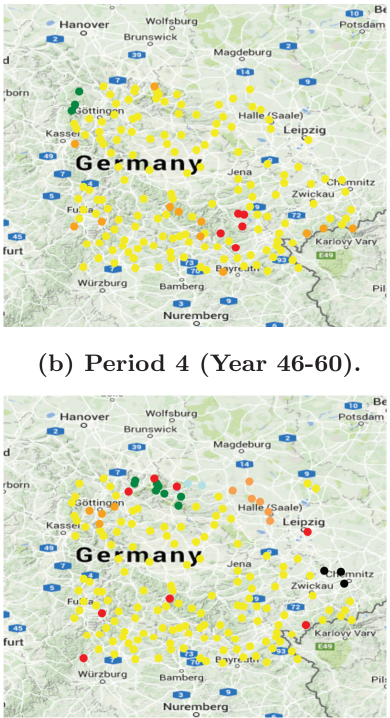
Between periods 1 and 4, we observe how the cluster patterns change dynamically. Weather stations in the southwest area are grouping into one big yellow cluster, changing from a mean residual -0.23 for the red cluster and -0.01 for the green cluster in period 1 to -0.02 for the yellow cluster in period 4. And weather stations in the west are grouping into another cluster (red) with a mean residual 0.12. Remarkably, the TRUST algorithm identifies partly changing clusters of temperature residuals in the early and late periods. Spatially varying climatic changes have been observed elsewhere before [1]; such patterns that would explain these observed changes in clustering may potentially be related to the complexity of topography in the studied region (orographic effects), changes in cloud cover and atmospheric dust content due to reduced industrial emissions first in West and later in East Germany, or confounding with spatially varying changes in precipitation, for example. While such explanations are not immediately evident from the clusters produced by TRUST, this knowledge discovery technique provides a starting point for further climatological analyses of local patterns of climate change. Knowledge of the existence and location of regions with homogeneous patterns may furthermore be instrumental in the geostatistical interpolation of instationary random fields of climatic parameters [24].



(a) Period 1 (Year 1-15).

(b) Period 4 (Year 46-60).

Figure 6: Clustering of weather stations in time period 1 and 4 by TRUST.



(a) Period 1 (Year 1-15).

(b) Period 4 (Year 46-60).

Figure 7: Clustering of weather stations in time period 1 and 4 by DBSCAN.

Table 6: MAPE of 4 time periods by TRUST and DBSCAN on observed temperature data.

TRUST MAPE (Year 1-15)	TRUST MAPE (Year 16-30)	TRUST MAPE (Year 31-45)	TRUST MAPE (Year 46-60)
0.07	0.07	0.06	0.03
DBSCAN MAPE (Year 1-15)	DBSCAN MAPE (Year 16-30)	DBSCAN MAPE (Year 31-45)	DBSCAN MAPE (Year 46-60)
0.11	0.11	0.10	0.09

Table 7: MAPE of 4 time periods by TRUST and DBSCAN on scaled temperature data.

TRUST MAPE (Year 1-15)	TRUST MAPE (Year 16-30)	TRUST MAPE (Year 31-45)	TRUST MAPE (Year 46-60)
0.63	1.12	1.10	0.94
DBSCAN MAPE (Year 1-15)	DBSCAN MAPE (Year 16-30)	DBSCAN MAPE (Year 31-45)	DBSCAN MAPE (Year 46-60)
1.69	1.28	1.74	1.75

MAPE values for 4 periods of observed temperature data and scaled temperature data are shown in Table 6 and Table 7. TRUST outperforms DBSCAN in each of the 4 periods (with smaller MAPE) on both observed temperature data and scaled temperature data.

6. CONCLUSION

In this paper, we advance the idea of clustering (in)stability from a case of selecting a "true" number of clusters to a choice of optimal tuning parameters in a broad range of dynamic clustering algorithms. We propose a new data-driven and computationally efficient procedure called Downhill Riding (DR) for optimal selection of clustering tuning parameters in dynamic clustering algorithms like TRUST and DBSCAN using a *clustering stability probe*. Using simulations, as well as real data, we show the effectiveness of the new procedure for selection of optimal parameters. The finite sample performance of Downhill Riding for dynamic clustering of synthetic time series is close to the optimal for these algorithms. Furthermore, the performance of clustering algorithms using Downhill Riding against competing algorithms that have a-priori knowledge of the parameters, shows that our procedure is a viable alternative, and often performs better. We also illustrate the Downhill Riding procedure in dynamic cluster detection in yearly temperature records among 167 stations in Central Germany over 1951-2010. Based on the clustering results of TRUST and DBSCAN, not only do we discover a well known pattern but also a dynamic pattern, which is useful when studying spatially varying climatic changes. In the future, we plan to extend the use of the new Downhill Riding procedure to other other dissimilarity measures and stability probes, and investigate the utility of Downhill Riding in other clustering algorithms.

Acknowledgements

Authors would like to thank Vyacheslav Lyubchich for helpful discussions throughout the project. This work was made possible by the facilities of the Shared Hierarchical Academic Research Computing Network.

References

[1] O. Anisimov, V. Kokorev, and Y. Zhil'tsova. Temporal and spatial patterns of modern climatic warming: case study of northern eurasia. *Climatic Change*, 118(3-4):871–883, 2013.

[2] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander. Optics: ordering points to identify the clustering structure. In *Proc. SIGMOD'99 Int. Conf. on Management of Data*, volume 28, pages 49–60. ACM, 1999.

[3] A. Appice, A. Ciampi, and D. Malerba. Summarizing numeric spatial data streams by trend cluster discovery. *Data Mining and Knowledge Discovery*, 29(1):84–136, 2015.

[4] S. Banerjee, B. P. Carlin, and A. E. Gelfand. *Hierarchical modeling and analysis for spatial data*. CRC Press, 2014.

[5] R. G. Barry. *Mountain weather and climate*. Psychology Press, 1992.

[6] R. Beelen, G. Hoek, D. Vienneau, M. Eeftens, K. Dimakopoulou, X. Pedeli, M.-Y. Tsai, N. Künzli, T. Schikowski, A. Marcon, et al. Development of NO_2 and NO_x land use regression models for estimating air pollution exposure in 36 study areas in europe–the escape project. *Atmospheric Environment*, 72:10–23, 2013.

[7] S. Ben-David and U. Von Luxburg. Relating clustering stability to properties of cluster boundaries. In *Proc. COLT*, volume 2008, pages 379–390, 2008.

[8] S. Ben-David, U. Von Luxburg, and D. Pál. A sober look at clustering stability. In *Learning Theory*, pages 5–19. Springer, 2006.

[9] P. Bickel and Y. Gel. Banded regularization of covariance matrices in application to parameter estimation and forecasting of time series. *Journal of the Royal Statistical Society, Ser. B*, 73(5):711–728, 2011.

[10] S. Bubeck and U. v. Luxburg. Nearest neighbor clustering: A baseline method for consistent clustering with arbitrary objective functions. *The Journal of Machine Learning Research*, 10:657–698, 2009.

[11] F. Cao, M. Ester, W. Qian, and A. Zhou. Density-based clustering over an evolving data stream with noise. In *Proc. SDM*, volume 6, pages 328–339. SIAM, 2006.

[12] CIA. *Water damage risk and Canadian property insurance pricing*. Canadian Institute of Actuaries, 2014.

[13] A. Ciampi, A. Appice, and D. Malerba. Discovering trend-based clusters in spatially distributed data streams. In *International Workshop of Mining Ubiquitous and Social Environments*, pages 107–122, 2010.

[14] L. Curry, A. Weaver, and E. Wiebe. *Determining the impact of climate change on insurance risk and the global community. Phase I: Key climate indicators*. Sponsored by the American Academy of Actuaries' Property/Casualty Extreme Events Committee, CAS, CIA, and SOA, and with input from CIWG, 2012.

[15] C. Daly, M. Halbleib, J. I. Smith, W. P. Gibson, M. K. Doggett, G. H. Taylor, J. Curtis, and P. P. Pasteris. Physiographically sensitive mapping of climatological temperature and precipitation across the conterminous united states. *Int. J. of Climatology*, 28(15):2031–2064, 2008.

[16] Deutscher Wetterdienst Data Archive, 2015. http://www.dwd.de, Accessed: 2016-05-26.

[17] S. Dudoit and J. Fridlyand. A prediction-based resampling method for estimating the number of clusters in a dataset. *Genome Biology*, 3(7):research0036.1–0036.21.

[18] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. KDD'96*, volume 96, pages 226–231. ACM, 1996.

[19] C. Euan, H. Ombao, and J. Ortega. Spectral synchronicity in brain signals. *arXiv preprint arXiv:1507.05018*, 2015.

[20] Y. Fang and J. Wang. Selection of the number of clusters via the bootstrap method. *Computational Statistics & Data Analysis*, 56(3):468–477, 2012.

[21] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2):179–188, 1936.

[22] M. M. Gaber, A. Zaslavsky, and S. Krishnaswamy. Mining data streams: a review. *ACM SIGMOD Record*, 34(2):18–26, 2005.

[23] T. Gneiting. Nonseparable, stationary covariance functions for space–time data. *Journal of the American Statistical Association*, 97(458):590–600, 2002.

[24] J. Guinness, M. L. Stein, et al. Interpolation of nonstationary high frequency spatial–temporal temperature data. *Annals of Applied Statistics*, 7(3):1684–1708, 2013.

[25] M. Hahsler. *dbscan: Density Based Clustering of Applications with Noise (DBSCAN) and Related Algorithms*, 2015. R package version 0.9-6.

[26] O. Haug, X. K. Dimakos, J. F. Vardal, M. Aldrin, and E. Meze-Hausken. Future building water loss projections posed by climate change. *Scandinavian Actuarial Journal*, 2011(1):1–20, 2011.

[27] A. Hinneburg and D. A. Keim. An efficient approach to clustering in large multimedia databases with noise. In *KDD*, volume 98, pages 58–65, 1998.

[28] E. Keogh and J. Lin. Clustering of time-series subsequences is meaningless: implications for previous and future research. *Knowledge and Information Systems*, 8(2):154–177, 2005.

[29] J. Kogan, C. Nicholas, and M. Teboulle. *Grouping multidimensional data*. Springer, 2006.

[30] M. Lichman. UCI machine learning repository, 2013.

[31] A. C. Lozano, H. Li, A. Niculescu-Mizil, Y. Liu, C. Perlich, J. Hosking, and N. Abe. Spatial-temporal causal modeling for climate change attribution. In *Proc. KDD'09*, pages 587–596. ACM, 2009.

[32] T. Lux and M. Marchesi. Volatility clustering in financial markets: a microsimulation of interacting agents. *International Journal of Theoretical and Applied Finance*, 3(04):675–702, 2000.

[33] V. Lyubchich and et al. *funtimes: Functions for Time Series Analysis*, 2016. R package version 2.3.

[34] I. Mahlstein and R. Knutti. Regional climate change patterns identified by cluster analysis. *Climate Dynamics*, 35(4):587–600, 2010.

[35] M. Meilă. Comparing clusterings – an information based distance. *Journal of Multivariate Analysis*, 98(5):873–895, 2007.

[36] P. Montero and J. A. Vilar. Tsclust: An R package for time series clustering. *J. of Stat. Software*, 2014.

[37] S. Phoha, N. Jacobson, D. Friedlander, and R. Brooks. Sensor network based localization and target tracking through hybridization in the operational domains of beamforming and dynamic space-time clustering. In *Proc. GLOBECOM'03*, volume 5, pages 2952–2956. IEEE, 2003.

[38] P. Rashidi and D. J. Cook. Mining sensor streams for discovering human activity patterns over time. *Proc. 2010 IEEE ICDM*, pages 431–440, 2010.

[39] C. Ratanamahatana, E. Keogh, A. J. Bagnall, and S. Lonardi. A novel bit level time series representation with implication of similarity search and clustering. In *Advances in Knowledge Discovery and Data Mining*, pages 771–777. Springer, 2005.

[40] J. Sander, X. Qin, Z. Lu, N. Niu, and A. Kovarsky. Automatic extraction of clusters from hierarchical clustering representations. In *Proc. PAKDD*, pages 75–87. Springer, 2003.

[41] I. Scheel, E. Ferkingstad, A. Frigessi, O. Haug, M. Hinnerichsen, and E. Meze-Hausken. A Bayesian hierarchical model with spatial variable selection: the effect of weather on insurance claims. *Journal of the Royal Statistical Society, Ser. C*, 62(1):85–100, 2013.

[42] M. Sherman. *Spatial statistics and spatio-temporal data: covariance functions and directional properties*. John Wiley & Sons, 2011.

[43] A. B. Smith and R. W. Katz. US billion-dollar weather and climate disasters: data sources, trends, accuracy and biases. *Natural Hazards*, 67(2):387–410, 2013.

[44] M. Soliman, V. Lyubchich, Y. R. Gel, D. Naser, and S. Esterby. *Machine Learning and Data Mining Approaches to Climate Science*, chapter Evaluating the impact of climate change on dynamics of house insurance claims, pages 175–183. Springer, Switzerland, 2015.

[45] M. L. Stein. Space–time covariance functions. *Journal of the American Statistical Association*, 100(469):310–321, 2005.

[46] Y. Unal, T. Kindap, and M. Karaca. Redefining the climate zones of turkey using cluster analysis. *International Journal of Climatology*, 23(9):1045–1055, 2003.

[47] T. Urbancic, C. Trifu, J. Long, and R. Young. Space-time correlations ofb values with stress release. *Pure and Applied Geophysics*, 139(3-4):449–462, 1992.

[48] R. Vines. European rainfall patterns. *Journal of Climatology*, 5(6):607–616, 1985.

[49] U. Von Luxburg. *Clustering Stability*. Now Publishers Inc, 2010.

[50] S. Wagner and D. Wagner. *Comparing clusterings: an overview*. Universität Karlsruhe, Fakultät für Informatik Karlsruhe, 2007.

[51] J. Wang. Consistent selection of the number of clusters via crossvalidation. *Biometrika*, 97(4):893–904, 2010.

[52] L. Wei and E. Keogh. Semi-supervised time series classification. In *Proc. KDD'06*, pages 748–753. ACM, 2006.

[53] A. X. Zhang, A. Noulas, S. Scellato, and C. Mascolo. Hoodsquare: Modeling and recommending neighborhoods in location-based social networks. In *Proc. 2013 Int. Conf. on Social Computing (SocialCom)*, pages 69–74. IEEE, 2013.