High-dimensional Time Series Clustering via Cross-Predictability

Dezhi Hong

University of Virginia

Quanquan Gu

University of Virginia

Kamin Whitehouse

University of Virginia

Abstract

be much larger than the length of each time series T.

Time series clustering often serves as an important ﬁrst

step for many applications and poses long-standing

challenges. In this paper, we explore the challenge of

time series clustering in the high-dimensional regime.

The key to time series clustering is how

to characterize the similarity between any

two time series. In this paper, we ex-

plore a new similarity measure called “cross-

predictability”: the degree to which a future

value in each time series is predicted by past

values of the others. However, it is challenging

to estimate such cross-predictability among

time series in the high-dimensional regime,

where the number of time series is much larger

than the length of each time series. We ad-

dress this challenge with a sparsity assump-

tion: only time series in the same cluster

have signiﬁcant cross-predictability with each

other. We demonstrate that this approach

is computationally attractive, and provide a

theoretical proof that the proposed algorithm

will identify the correct clustering structure

with high probability under certain conditions.

To the best of our knowledge, this is the ﬁrst

practical high-dimensional time series cluster-

ing algorithm with a provable guarantee. We

evaluate with experiments on both synthetic

data and real-world data, and results indicate

that our method can achieve more than 80%

clustering accuracy on real-world data, which

is 20% higher than the state-of-art baselines.

The key to time series clustering is how to character-

ize the similarity between any two time series. In the

past several decades, various metrics for measuring the

similarity/distance between time series have been in-

vestigated [8, 13, 6, 11, 18, 26, 10, 20, 34, 3], and so on.

Hidden Markov Models [29, 24] have also been utilized

to derive the distances between time series for cluster-

ing. Recently, a few new metrics [27, 19] to measure the

similarity between time series have been proposed and

applied to cluster brain-computer interface data and

motion capture data. However, all the aforementioned

work either did not provide theoretical guarantees for

their methods, or only considered scenarios where the

number of observations per time series T far exceeds

the number of time series d.

In this paper, we explore a new similarity measure

called “cross-predictability”: the degree to which a fu-

ture value in each time series is predicted by past values

of the others. This measure captures causal relation-

ships between time series, such as seasonal or diurnal

e↵ects on multiple environment sensors, market e↵ects

on multiple stock prices, and so on. However, it is

challenging to estimate such cross-predictability among

time series in the high-dimensional setting where d > T:

a conventional regression task, for example, would have

d variables and T equations, which is under-constrained.

Intuitively, only time series in the same cluster would

have signiﬁcant cross-predictability for each other, thus

yielding sparse relationships that are indicative of the

cluster structure. Consequently, we propose to esti-

mate cross-predictability by imposing a sparsity as-

sumption on the cross-predictability matrix, i.e., that

only time series in the same cluster have signiﬁcant

cross-predictability with each other. To do this, we

propose a new regularized Dantzig selector, which is a

1

INTRODUCTION

The proliferation of cheap, ubiquitous sensing infras-

tructure has enabled continuous monitoring of the

world, and many expect the Internet of Things to have

over 25 billion devices by 2020 [12]. In this paradigm,

time series data will often be high-dimensional: the

number of time series d (i.e., number of sensors) will

Proceedings of the 20th International Conference on Artiﬁ- variant of standard Dantzig selector [ ], to estimate the

7

cial Intelligence and Statistics (AISTATS) 2017, Fort Laud-

erdale, Florida, USA. JMLR: W&CP volume 54. Copyright

this approach is computationally attractive because it

2017 by the author(s).

similarity among the time series. We demonstrate that



High-dimensional Time Series Clustering via Cross-Predictability

involves solving d regularized Dantzig selectors that

can be optimized by alternating direction method of

multipliers (ADMM) [4] in parallel.

two related categories: clustering based on similarity

and subspace clustering.

Additionally, we provide a theoretical proof that the

proposed algorithm will identify the correct clustering

structure with high probability, if two conditions hold:

1) the individual time series themselves can be modeled

with an autoregressive model [14], and 2) the transition

matrix for the vector autoregressive model is block

diagonal, i.e., that it is actually possible to create

clusters such that time series in the same cluster are

cross-predictive while those in di↵erent clusters are not.

Similarity/Distance-based Time Series Cluster-

ing A wide range of classical similarity/distance met-

rics have been developed and studied [8], including Pear-

son’s correlation coecient [13], cosine similarity [6],

autocorrelation [11], dynamic time warping [18, 26],

Euclidean Distance [10], edit distance [20], distance

metric learning [34, 3], and so on. Studies have also

shown that time series can be modeled as generated

from Hidden Markov Models [29, 24], and the esti-

mated weight for each mixture can be used to cluster

the time series. Recently, Ryabko et al. [27] considered

brain-computer interface data for which independence

assumptions do not hold, and for clustering they pro-

posed a new distance metric to measure the similarity

between two time series distributions. Khaleghi et

al. [19] formulated a novel metric to quantify the dis-

tance between time series and proved the consistency

of k-means for clustering processes according only to

their distributions. However, in the aforementioned

studies, they either did not provide theoretical analysis

of the performance or only handled settings where the

number of time series d is smaller than the number

of observations T. Di↵erent from the above similarity

or distance metrics, we deﬁne the similarity between

time series from a new perspective - time series are

clustered based on how much they can be predicted by

each other.

To the best of our knowledge, this is the ﬁrst practi-

cal high-dimensional time series clustering algorithm

with a provable guarantee. It is worth noting that the

proposed algorithm can be generally applied to cluster

any high dimensional times series, regardless of the un-

derlying data distribution. We make the autoregressive

model assumption solely for the purpose of providing

the theoretical guarantees for our method.

To demonstrate the e↵ectiveness of our method, we

conduct experiments on a real-world data set of sensor

time series as well as simulations with synthetic data.

Our method can achieve more than 80% clustering

accuracy on the real-world data set, which is 20% higher

than the state-of-art baselines.

Notations We compile here some standard notations

used throughout the paper. In this paper, we use low-

ercase letters x, y, . . . to denote scalars, bold lowercase

letters x, y, . . . for vectors, and bold uppercase letters

X, Y, . . . for matrices. We denote random vectors by

X, Y . We denote the (i, j) entry of a matrix as Mij,

and use M to index the i-th row of a matrix (like-

Subspace Clustering (SC) Another relevant line

of research on high-dimensional data analysis is sub-

space clustering [9], where the assumption is that data

lie on the union of multiple lower-dimensional linear

spaces and data points can be clustered into the sub-

space they belong to. SC has been widely applied to

face images clustering [1], social graphs [17] and so on.

Recently, extensions to handling noisy data [21, 31, 33]

and data with irrelevant features [25] have been stud-

ied as well. SC achieves the state-of-art performance

while enjoying rigorous theoretical guarantees. The

key di↵erence between SC and our method is two-fold:

ﬁrst, SC assumes data lie on di↵erent subspaces and

even data in the same subspace are independent and

identically distributed (i.i.d.), while we assume the time

series follow a VAR model and are dependent for the

ones in the same cluster; second, SC mathematically

solves for each data point a linear regression problem

with all other data points being the candidate, while

in contrast, our study solves the regression problem to

estimate the prediction weights between observations

from di↵erent time stamps using all the time series.

i⇤

wise, M for the j-th column). We also use M

to represent a submatrix of M with its rows indexed

⇤j

S,T

by the indices in set S and columns indexed by T. In

addition, we write Sc to denote the complement of a set

S. For any matrix M, P(M) represents the symmetric

convex hull of its columns, i.e., P(M) = conv(±X).

For any matrices M , M , . . . M , we denote a block

1

2

k

diagonal matrix by diag(M , M , . . . , M ) such that

1

2

k

the k-th diagonal block is M . Throughout the pa-

k

per, we will use vector norm ` for 0 < q < 1 and

q

⇣

⌘

P

1/q

, kvk1,1

q

`1 of v deﬁned as kvk =

|v |

=

q

i

i

max |v |, and matrix norm ` , element-wise ` and

kvk =1

i

i

q

1

`

of M as kMk = max

kMvk , kMk

=

F

q

q

1,1

⇣

RELATED WORK

q

⌘

P

1/2

.

2

ij

max |M | , kMk

=

|M |

ij

ij

F

i,j

2

There has been a substantial body of work on time

series clustering, and in this section we brieﬂy overview



Dezhi Hong, Quanquan Gu, Kamin Whitehouse

3

METHODOLOGY

following Dantzig selector type estimator [15],

ˆ

ˆ

>

ˆ

1

A = arg min kAk subject to k⌃A ⌃k  µ,

1,1

1

3.1 The VAR Model

A

(3.3)

where µ > 0 is a tuning parameter. Since each row of

A is independent, the above optimization problem can

be decomposed into d independent sub-problems and

solved individually as follows:

Our algorithm is motivated by the autoregressive

model, and the later-on theoretical guarantee for our

algorithm also relies on the autoregressive model as-

sumption, so we brieﬂy review the stationary ﬁrst-

order vector autoregressive model with Gaussian noise

ˆ

ˆ

i

= arg min k k subject to k⌃ ˆ k  µ,

i 1,1

i

i

1

here. Let random vectors X , . . . , XT be from a

1

, and we further deﬁne

> 2 RT ⇥d

, where Xt =

is a d-dimensional vector and each

i

stationary process (X )1

(3.4)

t t=1

ˆ

>

S

i-th column of ⌃, and A = , . . . ,

X = [X , . . . , X , . . . X ]

where ˆ = (⌃) = X (X ) /(T 1), i.e., ˆ is the

1

(x , . . . , x )

t

T

i

1 ⇤i

ˆ

T

⇤i

i

h

i

> 2 Rd

>

ˆ

ˆ

ˆ

2 Rd⇥d

1

d

1

1

d

column of X is a one-dimensional time series with T

ˆ

Rd

ˆ

i

with each 2

. Therefore, the in (3.4) is an

samples. In particular, we assume each X can be

i

t

modeled by a ﬁrst-order vector autoregressive model:

estimation of the i-th row of the transition matrix A.

Furthermore, for each µ > 0, there always exists a

> 0 such that (3.4) is equivalent to the following

regularized Dantzig selector type estimator:

Xt+1 = AX + Z , for t = 1, 2, . . . , T 1. (3.1)

t

t

To secure the above process to be stationary, the tran-

sition matrix A must have bounded spectral norm,

i.e., kAk < 1. We also assume Z ⇠ N(0, ) is i.i.d.

ˆ

ˆ

i

= arg min k⌃ ˆ k

+ k k ,

(3.5)

i

i

1,1

i 1

i

where is a regularization parameter to determine

the sparsity of the estimation. (3.4) can be solved by

alternating direction method of multipliers (ADMM)

[4]. All the d optimization problems can be solved in

parallel, thus computationally ecient.

2

additive noise independent of X , and X has zero

t

t

t

mean and a covariance matrix ⌃, i.e., X ⇠ N(0, ⌃),

t

where ⌃= E[X X ] is the autocovariance matrix. In

>

t

t

addition, we have the lag-1 autocovariance matrix as

⌃ = E[X X ]. Since (X )1

is stationary, it is

t t=1

>

t+1

1

t

After solving the problem in (3.5), we construct an

easy to observe that the covariance matrix ⌃depends

h

i

>

>

on A and , i.e., ⌃= A ⌃A + , and we further

have:

ˆ

ˆ

ˆ

anity matrix W based on A = , . . . ,

by

1

d

symmetrization, and compute the corresponding Lapla-

cian to perform standard spectral clustering [23, 28]

to recover the clusters in the input time series. The

procedure is summarized in Algorithm 1.

⌃A> = ⌃1.

(3.2)

Essentially, the zero and nonzero entries in the tran-

sition matrix A directly reﬂect the Granger non-

causalities and causalities with regard to the stochastic

time series. In other words, a nonzero entry Aij implies

that the j-th time series is predictive for the i-th time

3.3 Discussion

At ﬁrst glance, the regularized Dantzig selector in (3.5)

and Lasso appear similar. However, di↵erent from

Lasso, the “input” of the regression problem in Eq (3.5)

is the lag-0 covariance matrix, and the “response” is

the lag-1 covariance matrix. Here the lag-one covari-

ance matrix encodes and includes into consideration

the ﬁrst-order temporal information, which is missing

in conventional similarity metrics such as correlation.

Additionally, di↵erent from the Lasso-based estimation

procedure [2], which penalizes the square loss, the reg-

series, with the magnitude |A | indicating how much

ij

the predictive power is. The new similarity measure

in our clustering algorithm is built upon such cross-

predictive relationship between time series. Now we

set to introduce the clustering algorithm.

3.2 The Proposed Clustering Algorithm

Our algorithm ﬁrst estimates the cross-predictability

among the time series, and then identiﬁes the clustering

structure based on the estimated relationship. To in-

troduce our proposed algorithm, we need the following

ularized Dantzig selector estimator penalizes the `

loss.

1,1

It is also worth noting that Algorithm 1 shares a similar

high level idea as the subspace clustering (SC) algo-

rithm [30, 33, 31], but the key di↵erence is that SC

considers the relationship between each data point and

all the other points, while in contrast, our estimator

solves the regression problem to estimate the predic-

tive relationship between the observations from time

R(T 1)⇥d, X

notations: X = [X , . . . , X

1

]>

2

S

=

T

S

T 1

>

R(T

1)⇥d, ⌃= X X /(T 1), and

ˆ

>

[X , . . . , X ]

T

2

2

S

ˆ

>

⌃ = X X /(T 1). Inspired by the relationship

1

S

T

in Eq. (3.2), our main idea is to estimate A based on

the relationship between A and the autocovariance

and lag-1 autocovariance matrices. This motivates the



High-dimensional Time Series Clustering via Cross-Predictability

4.1 Preliminaries

Algorithm 1: Time Series Clustering Algorithm

Input: Time series X = [X , . . . X ] 2 RT ⇥d,

>

To deﬁne the clusters among time series X un-

der the context of VAR model, we assume A =

diag(A , . . . , A , . . . , A ) to be block diagonal, where

1

R(T 1)⇥d,

T

X = [X , . . . , X

]>

2

S

T

1

T 1

T

>

R(T 1)⇥d,

X

ˆ

= [X , . . . , X ]

2

2

1

l

k

A 2 Rd ⇥d and the number of time series d satis-

⌃= X X /(T 1), and ˆ = X (X ) /(T 1)

>

>

l

ﬁes d =

l

l

S

S

i

S

T

⇤i

P

k

l=1

d . Consequently, we can rewrite X

Output: Cluster membership of each time series Y

1. Solve for each i = 1, . . . , d:

⇥

l

⇤

as X = X1, . . . , Xl, . . . , Xk with each Xl 2 RT ⇥d

l

obeying:

ˆ

ˆ

i

= arg min k⌃ ˆ k

+ k k ;

i 1

i

i

1,1

l

l

= A X + Z , for t = 1, 2, . . . , T 1,

t+1 t

l

l

X

i

t

h

i

2. Set A = , . . . , >;

which essentially deﬁnes the clustering structure in the

time series, such that the data Xtl+1 2 Rd at time point

ˆ

ˆ

ˆ

d

1

l

3. Construct the anity graph G with nodes being

the d time series in X, and edge weights given by the

l

t + 1 depends only on the data X from the previous

t

time point t in the same block indexed by A . In other

l

words, as an e↵ect of A , data are more predictive for

ˆ ˆ >

of graph G, with M = diag(m , m , . . . , m ) and

matrix W = A + A

;

l

4. Compute the unnormalized Laplacian L = M W

each other in the same block, rather than for those

in the other blocks. The block diagonal transition

matrix A gives rise to the fact that the time series

1

2

d

P

d

j=1

m =

i

W ;

ij

in X 2

RT ⇥d formulate k clusters C , . . . , C , . . . , C of

1 l k

l l

5. Compute the ﬁrst k eigenvectors , . . . , of L

1

k

and let V 2 Rd be the matrix containing as

⇥k

RT ⇥dl , and each C contains d one-dimensional time

RT denoted as Xl. Without loss of generality,

columns the ﬁrst k eigenvectors;

0 2 Rk

series of

let X = X1, . . . , Xl, . . . , Xk be ordered. We further

⇥

⇤

6. Cluster time series xi

, as the i-th row of V,

with the k-means algorithm into clusters

write S to denote the set of indices corresponding to

l

C , . . . , C , . . . , C .

the columns of X that belong to cluster C .

1

l

k

l

Deﬁnition 4.1 (Cluster Recovery Property). The clus-

ters {C }k and the time series X from these clusters

l

l=1

obey the cluster recovery property (CRP) with a param-

A

XT

XS

eter , if and only if it holds that for all i, the optimal

solution ˆ to (3.5) satisﬁes: (1) ˆ is nonzero; (2) the

X2

T - 1

X1

T - 1

Xt+1,di

Xt,

i

indices of nonzero entries in correspond to only the

i

d

A,di

ˆ

i

XT

XT1

columns of X that are in the same cluster as X⇤i.

d

d

d

This property ensures that the output coecient ma-

trix A and anity matrix W will be exactly block

ˆ

Figure 1: Illustration of the Proposed Regularized

Dantzig Selector: it solves the regression problem to

estimate the predictive relationship between the obser-

vations from time t + 1 and time t considering all the

time series.

diagonal, with each cluster represented in a disjoint

block. Particularly, recall that we assume the transition

matrix A in the VAR model to be block diagonal, and

therefore the CRP is guaranteed to hold for data gener-

ated from such a model. For convenience, we will refer

to the second requirement as the “Self-Reconstruction

Property (SRP)” from now on.

t + 1 and the observations from time t considering all

the time series, as illustrated in Figure 1. Another

fundamental di↵erence here is, SC assumes data are

i.i.d. and lie on di↵erent subspaces, while here the time

series data are obviously dependent. This poses a big

challenge to the theoretical analysis of our algorithm.

Deﬁnition 4.2 (Inradius [30]). The inradius of a con-

vex body P, denoted by r(P), is deﬁned as the radius

of the largest Euclidean ball inscribed in P.

By the deﬁnition, the radius of a P(X) measures the

dispersion of the time series in X. Naturally, well-

dispersed data will yield a large inradius while data

with skewed distribution will have a small inradius.

4

MAIN RESULTS

In this section, we state our main theory - a provable

guarantee for successfully recovering the underlying

clustering structure of the input time series. We ﬁrst

introduce some necessary deﬁnitions for understanding

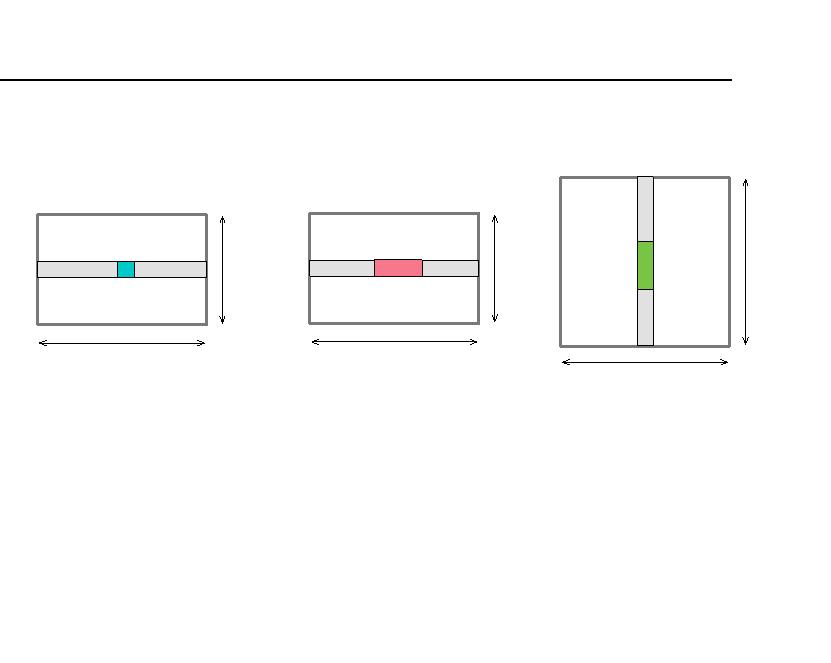
our main theorem.

4.2 Theoretical Guarantees

One of our major contributions in this paper is to pro-

vide theoretical guarantees for successfully recovering

the clustering structure in the data.



Dezhi Hong, Quanquan Gu, Kamin Whitehouse

Theorem 4.3. Under the assumption of VAR model

with a block diagonal transition matrix, we compactly

indicates that the sample complexity for CRP to hold is

O(log d). In other words, for the algorithm to succeed,

the number of time series d is allowed to grow expo-

nentially with the length of time series T; as long as

log d is smaller than the length of time series T, our

theory holds. Indeed, it is desired to see such a property

for high-dimensional data, as d can often possibly far

exceeds the number of samples T.

denote Pl = P(⌃

1

let

), Pl = P((⌃ )

), rl = r(Pl ),

0

Sl,Sl

1

1 Sl,Sl

0

0

rl = r(Pl ), and r r = min rl rl for l = 1, 2, ..., k, and

1

0

1

l

0 1

r

16k⌃k max ⌃

6 log d + 4

2

j

jj

⇢ =

.

(4.1)

(4.2)

min ⌃ (1 kAk )

T

j

jj

2

Remark 4.6 (A Uniform Parameter ). Another direct

observation from the main theorem is that we can ﬁnd

a uniform value for , within the range as speciﬁed

in (4.3), which can work for the regression task in (3.5)

for all i = 1, ..., d. In other words, problem in (3.5) is

solvable with a single and can be solved in parallel

for each i = 1, ..., d.

Furthermore, if

k⌃

k1,1 + 2⇢

1,1

Sc,Sl

r r >

0 1

l

,

k k

2⇢

S

l

where 2 Rdl is a column of ⌃, then with probability

S

1

l

at least 1 6d the cluster recovery property holds for

all the values of the regularization parameter in the

range:

1

5

EXPERIMENTS

1

1

In this section, we demonstrate the correctness of our

theoretical ﬁndings and the e↵ectiveness of our pro-

posed clustering algorithm on both synthetic and real-

world data. We ﬁrst experiment with di↵erent sets of

parameters, including the number of time series d, the

length of the time series T, and the number of clusters

k in the data. The experimental results conﬁrm that,

when the required condition in Theorem 4.3 is satisﬁed,

the clusters in the data can be recovered perfectly. We

further apply the algorithm to a real-world data set

where the task is to group sensor time series by their

type of measurement (e.g., a temperature sensor vs. a

humidity sensor). Our algorithm is able to outperform

the state-of-art baselines by more than 20% measured

by adjusted rand index.

,

r r (k k

2⇢) ⇢ < < ⇢ + k⌃

k

(4.3)

0

1

S

1,1

Sc,Sl 1,1

l

l

which is guaranteed to be non-empty.

We defer the full proof of the theorem in the supplemen-

tary material. The theorem provides an upper bound

and a lower bound for the regularization parameter ,

to successfully recover the underlying clustering struc-

ture in the time series: on the one hand, cannot be

too large, otherwise A will be too dense to perform

clustering on. On the other hand, as approximates 0

the connectivity among time series decreases because

the optimal solution to (3.5) becomes more sparse. To

guarantee the obtained solution is nontrivial (i.e., ˆi

is nonzero), must be larger than a certain value. In

addition, a lower bound on r r is established, which

0

1

5.1 Baselines

imposes a requirement on the dispersion of the covari-

ance between time series within the same cluster. We

further make the following remarks:

In our proposed clustering algorithm, we estimate the

similarity matrix with the regularized Dantzig selector

(referred to as CP). As baselines, instead of using our

estimator, we consider the following methods to obtain

the similarity matrix, and the rest of the clustering

procedure remains the same as ours:

Remark 4.4 (Tolerance of Noise across Clusters).

From (4.2) we see that, for the CRP to hold, the dis-

persion of the columns of ⌃(each column is taken as

a data point 2 Rd) needs to be suciently large. r is

0

the dispersion of covariance between time series in a

Correlation Coecient (CC): In this baseline, we

compute the Pearson correlation coecient between

all pairs of time series, and use these coecients to

construct the similarity matrix.

cluster S , and r is the dispersion of lag-1 covariance

l

1

between time series in a cluster S . The RHS of (4.2)

l

depends on the scale of ⌃

mum correlation between the time series in one cluster

and reﬂects the maxi-

l

Sc,S

l

Cosine Similarity (Cosine): In the second baseline,

we compute the pairwise cosine similarity for all the

time series, and preserve only the similarity scores for

the top-k nearest neighbors for each time series and

put them as the row of the similarity matrix. Our

experiment shows that the results are not sensitive to

k and we set k = 5.

S and any time series from all the other clusters.

l

Remark 4.5 (Sample Complexity). We can observe

that the factor before the square root in (4.1) is bounded

by the largest and smallest eigenvalue of ⌃, and there-

p

fore we can rewrite ⇢ =  (6 log d + 4)/T where

 is a constant dependent on ⌃. We can further

derive from (4.2) that T > 42(6 log d + 4)(r r +

0

1

1)2/(r r k k

k⌃

k1,1)2, which essentially

Autocorrelation (ACF): This baseline ﬁrst com-

0

1

S

1,1

Sc,Sl

l

l



High-dimensional Time Series Clustering via Cross-Predictability

putes the autocorrelation vectors (with a lag up to

the metric Self-Reconstruction Property Violation Rate

(VioRate) of the estimated transition matrix A as fol-

lows:

50) for each time series, and then further calculates

the Euclidean distance between each pair of time series

based on the autocorrelation vectors. We use the im-

plementation in [22] to obtain the distance matrix ﬁrst,

and then convert the distance into similarity score with

Gaussian kernel function. (Smaller distances should

map to larger similarity scores.)

~~P~~

~~|~~A ~~|~~

ij

i,j~~2~~/~~C~~l

ij

V ioRate = ~~P~~

,

|

|

A

i,j2Cl

where (i, j)

2 C denotes that the i-th time series X

l

⇤i

and the j-th time series X are in the same cluster

⇤j

C for some l (likewise for (i, j) 2/ C ). By deﬁnition,

l

l

Dynamic Time Warping (DTW): DTW is a pop-

ular method to compute the similarity between time

series. Here we compute the pairwise DTW similar-

ity score for all the time series, and then normalize

similarity scores to between 0 and 1.

V ioRate measures relatively how signiﬁcant the predic-

tive weights are for pairs of time series across di↵erent

clusters, compared to the weights for pairs in the same

cluster. For a trivial solution, i.e., A = 0, the V ioRate

is deﬁned to be 1 while for a solution satisfying the

self-reconstruction property, the V ioRate should be

exactly 0. The violation rates for di↵erent T/ log(d)

and values when k = 25 are illustrated in Figure 2a;

the results conﬁrm our theoretical ﬁndings. We ob-

serve that when is small, the solution violates the

nonzero requirement, thus the V ioRate being 1 (refer

to the two rows at the bottom). When is suciently

large within a range, the violation rates are zero, in-

dicating all the entries in the o↵-diagonal blocks of

the estimated A are zero, which satisﬁes the SRP. In

Figure 2b, we show the quality of time series clustering

(measured by adjusted rand index and higher is better)

with the corresponding A obtained in Figure 2a. We

can notice that cases perfectly satisfying the nonzero

and SRP requirements can produce perfect clustering

results. Furthermore, it is also clear that exact self-

reconstruction condition is not necessary for perfect

clustering.

We also implement a baseline that does not rely on the

similarity between time series:

Principal Component Analysis (PCA): In this

method, PCA is ﬁrst applied to reduce the dimension-

ality of each original time series by preserving d(=4)

principle components, and then k-means is applied to

these PCA scores for clustering.

5.2 Synthetic Data

In this section, we show the e↵ectiveness of our pro-

posed clustering algorithm via numerical simulations.

Particularly, the data X at a time point t are generated

t

from a VAR model as deﬁned in (3.1), and we generate

the input time series X as follows: (1) We ﬁrst generate

the block diagonal transition matrix A with k clusters,

and the values within each block are generated with a

Bernoulli distribution; (2) Since we assume (X)1

be stationary, we then rescale A such that its spectral

to

t=1

We next investigate how the number of clusters k a↵ects

the clustering performance, where we vary the value of

k from 5 to 25. For the regularization parameter , we

scan through the same exponential space as the above

experiment with 5-fold cross-validation, and choose

the one with the minimal cross-validation error. We

norm kAk = ↵ < 1; (3) Given A, ⌃is generated such

2

that the elements on the diagonal equal to 1 and the

o↵-diagonal elements are set to a same small value, e.g.,

0.1. Then we rescale ⌃to have its spectral norm satisfy

k⌃k = 2kAk ; (4) Next, according to the stationary

2

2

property, the covariance matrix of the additive noise Zt

ﬁx kAk at 0.4 and report the average results of the

2

>

follows

= ⌃ A ⌃A, where

must be a positive

deﬁnite matrix; (5) We can then generate X from the

100 runs for each set of parameters as illustrated in

Figure 3. We clearly see that a larger k leads to better

clustering results, which makes sense since the more the

number of clusters is, the sparser A is, and therefore

the more accurate the estimation of A is.

1

multivariate normal distribution with the parameters

generated in previous steps, and obtain the following

X with the VAR model. We ﬁx the number of time

t

series d at 100, and choose the length of the time series

T from a grid of {1, 3, 5, 7, 9} ⇥ log(d) (rounded to the

closet integer), i.e, the ratio of T/ log(d) varies from 1

to 9. For each value of T, we repeat the data gener-

ation process for 100 times and report the average of

the experimental results.

We also examine the e↵ect of the transition matrix’s

spectral norm kAk on the clustering quality. To this

2

end, we set kAk = ↵ and vary ↵ from 0.1 to 0.9, and

2

the covariance matrix ⌃and

are generated in the

same way as described earlier. For the parameter , we

take the same cross-validation procedure as above. We

ﬁx the number of clusters k at 25 and report the average

results of the 100 runs for each set of parameters, as

shown in Figure 4. We observe that, for a certain

value of T/ log(d), the clustering quality increases as

the spectral norm of the transition matrix decreases.

We ﬁrst experiment with di↵erent values for the reg-

ularization parameter and examine if the two re-

quirements as stated in Deﬁnition 4.1 are satisﬁed.

We scan through an exponential space of from

1/(log(d)/T) ⇥ 101 to 1/(log(d)/T) ⇥ 10 and deﬁne

3



Dezhi Hong, Quanquan Gu, Kamin Whitehouse

1

1

3

2.56

2.11

1.67

1.22

0.78

0.33

-0.11

-0.56

-1

3

0.9

0.9

0.8

0.7

0.6

0.5

0.4

0.3

0.2

0.1

0

2.56

2.11

1.67

1.22

0.78

0.33

-0.11

-0.56

0.8

0.7

0.6

0.5

0.4

0.3

0.2

0.1

-1

1

3

5

7

9

1

3

5

7

9

Value of T/log(d)

Value of T/log(d)

(a) Self-Reconstruction Property (SRP) violation

rate for di↵erent T/ log(d) against di↵erent with

k = 25: too small a will produce trivial solutions

(A = 0, thus V ioRate = 1) while a suciently

large gives a solution satisfying both the nonzero

and SRP requirements (V ioRate = 0).

(b) Clustering quality for di↵erent T/ log(d)

against di↵erent with k = 25: cases satisfy-

ing the nonzero and SRP conditions yield perfect

clustering results. It is also clear that the exact

self-reconstruction condition (V ioRate = 0) is not

necessary for perfect clustering.

Figure 2: Self-Reconstruction Property Violation Rate and the Corresponding Clustering Quality (measured by

Adjusted Rand Index) with Di↵erent T/ log(d) Against Di↵erent .

1

0.8

0.6

0.4

0.2

0

This indicates that the spectral norm of the transition

matrix is a critical factor and veriﬁes the theoretical

ﬁndings in (4.1).

To compare our method with the baselines described

in §5.1, we further conduct two sets of experiments on

synthetic data with di↵erent parameters. To generate

the synthetic data in the ﬁrst experiment (referred to

as Synthetic Data-1 in Table 1), we set the number of

time series d = 50, the length of time series T = 50, the

number of clusters k = 5, and the transition matrix’s

k=5

k=10

k=15

k=20

k=25

1

3

5

7

Value of T/log(d)

9

spectral norm kAk = 0.5. For the second experiment

2

Figure 3: Clustering Quality for Di↵erent

T/ log(d) Against Di↵erent Number of Clusters

k: larger k is better.

(Synthetic Data-2 in Table 1), we change T to 100, and

the rest of parameters remain the same. We see that,

when d is comparable to T in the ﬁrst experiment, our

method (CP) performs signiﬁcantly better than the

baselines. When the number of samples T is increased

to 100, all the baselines see performance boost, while

our method produces perfect clustering results.

1

α=0.1

0.8

α=0.3

α=0.5

0.6

0.4

0.2

0

=0.7

α

α=0.9

One shall note the better performance of PCA, our

understanding is that PCA extracts better explana-

tory components out of the sample covariance matrix,

which still captures the underlying causal relationship

between variables, though it does not consider the ﬁrst-

order temporal information as our proposed method

does. For the other baselines, they simply compute

similarity directly between variables, which is not su-

ciently e↵ective in characterizing the relationship be-

tween time series in the high-dimensional setting.

1

3

5

7

9

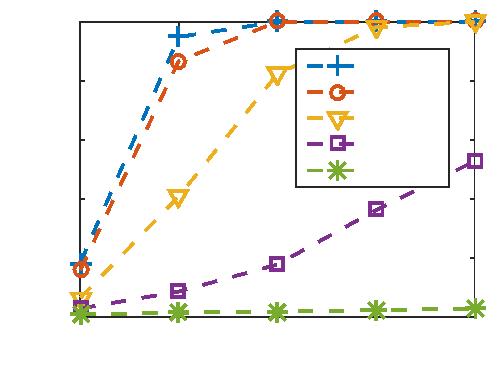
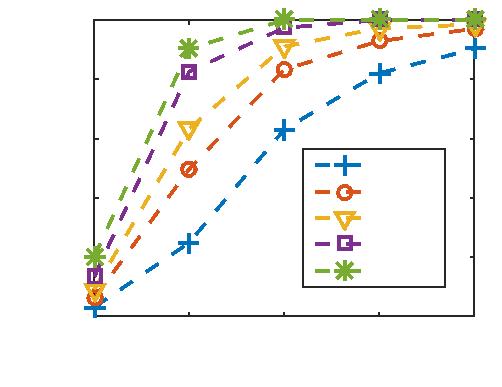
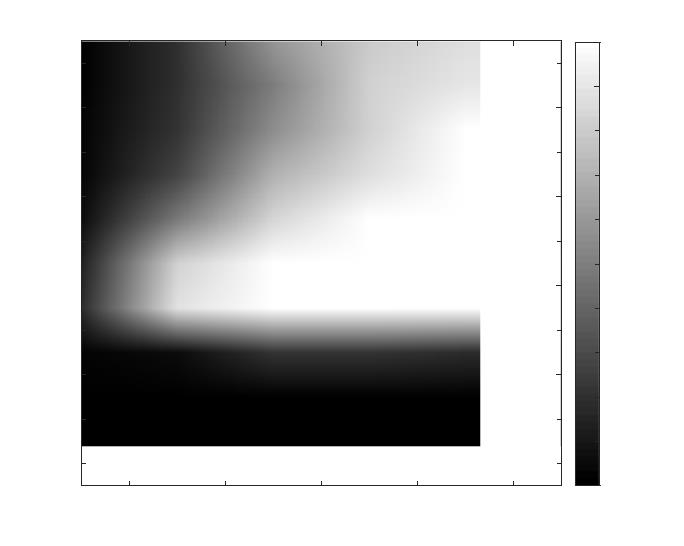
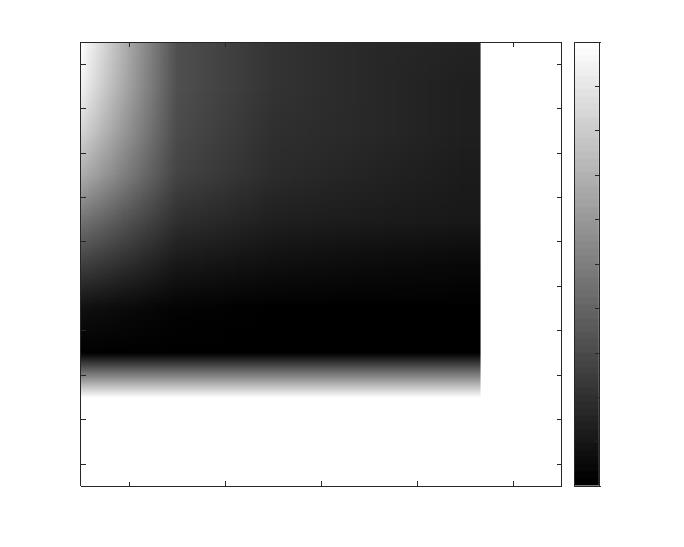
Value of T/log(d)

Figure 4: Clustering Quality for Di↵erent

T/ log(d) Against Di↵erent Values ↵ for Spectral

Norm kAk : smaller ↵ is better.

2



High-dimensional Time Series Clustering via Cross-Predictability

Table 1: Experimental Comparisons with Baselines: results on synthetic and real data demonstrate the advantage

of our proposed algorithm (CP), and each cell includes the average clustering performance (adjusted rand index)

of 10 runs with standard deviation.

CC

Cosine

ACF

DTW

PCA

CP

Synthetic Data-1

(d = 50, T = 50)

Synthetic Data-2

(d = 50, T = 100)

Real Data

0.383 ± 0.171 0.521 ± 0.123 0.240 ± 0.147 0.282 ± 0.184 0.786 ± 0.139 0.943 ± 0.165

0.603 ± 0.181 0.551 ± 0.143 0.253 ± 0.109 0.410 ± 0.105 0.912 ± 0.141 1.000 ± 0.000

0.617 ± 0.031 0.542 ± 0.014 0.362 ± 0.113 0.523 ± 0.119 0.456 ± 0.144 0.824 ± 0.025

5.3 Real-world Data

in Figure 5 (the data points beyond 0.15 all drop to

zero, thus omitted in the ﬁgure). It again conﬁrms

our theoretical ﬁndings in the sense that the proposed

clustering algorithm can work when is suciently

large, even not perfectly. We also examine how well the

baselines (detailed in §5.1) perform on the real data set,

and the results are summarized in Table 1. Our method

can achieve more than 80% accuracy and outperforms

the best baseline by more than 20%, indicating that

our method can still be e↵ective when the assumption

of VAR with a block diagonal transition matrix might

not be satisﬁed.

To further examine how e↵ective our proposed algo-

rithm is in practice, we also apply it to a real-world

data set, where the assumption of VAR model with

block diagonal transition matrix might not be perfectly

satisﬁed. The data set [16] contains data collected

from 204 sensor time series from 51 rooms on 4 dif-

ferent ﬂoors of a large oce building on a university

campus. Each room is instrumented with 4 di↵erent

types of sensors: a CO sensor, a temperature sensor,

2

a humidity sensor and a light sensor. The data from

each sensor is recorded every 15 minutes and the data

set contains one-week worth of data. There are missing

values in the one-week period, so the total number of

observations T is smaller than the number of sensor

time series d. Our goal is to assign each sensor time

series into the correct type cluster, e.g., a tempera-

6

CONCLUSIONS

In this paper, we study the time series clustering

problem with a new similarity measure in the high-

dimensional regime, where the number of time series

is much larger than the length of time series. Dif-

ferent from existing metrics, our similarity measure

quantiﬁes the “cross-predictability” between time se-

ries, i.e., the degree to which a future value in each

time series is predicted by past values of the others.

We impose a sparsity assumption and propose a reg-

ularized Dantzig selector estimator to learn the cross-

predictability among time series for clustering. We

further provide a theoretical proof that the proposed

algorithm will successfully recover the clustering struc-

ture in the data with high probability under certain

conditions. Experiments on both synthetic and real-

world data verify the correctness of our ﬁndings, and

demonstrate the e↵ectiveness of the algorithm. For the

real-world task of sensor type clustering, our method is

able to outperform the state-of-art baselines by more

than 20% with regard to clustering quality.

ture cluster or a CO cluster. Recognizing the type

2

of sensors is often an important step for many useful

applications. For instance, when applying analytics

stacks comprised of a bundle of analytics jobs to a

building for energy savings, every particular analytics

job requires as input some speciﬁc types of sensors.

0.8

0.6

0.4

0.2

0

0.05

0.1

0.15

Regularization Parameter 1/λ

7

Acknowledgments

Figure 5: Clustering Quality of Our Regularized

Dantzig Selector-based Spectral Clustering Algorithm:

the algorithm works with a wide range of .

We would like to thank the anonymous reviewers for

their invaluable comments. This work was funded

by NSF awards CNS-1305362, IIS-1618948, and IIS-

165253. The views and conclusions contained in this

paper are those of the authors and should not be inter-

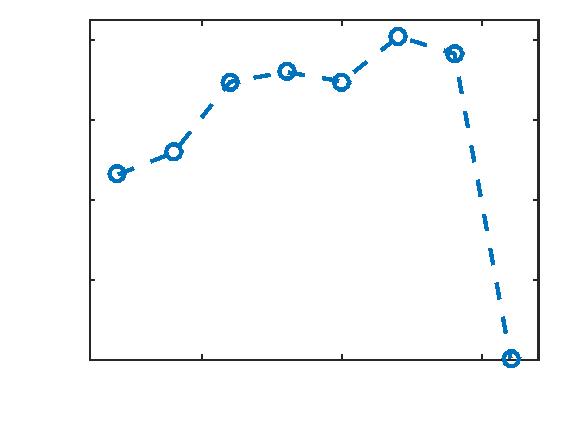
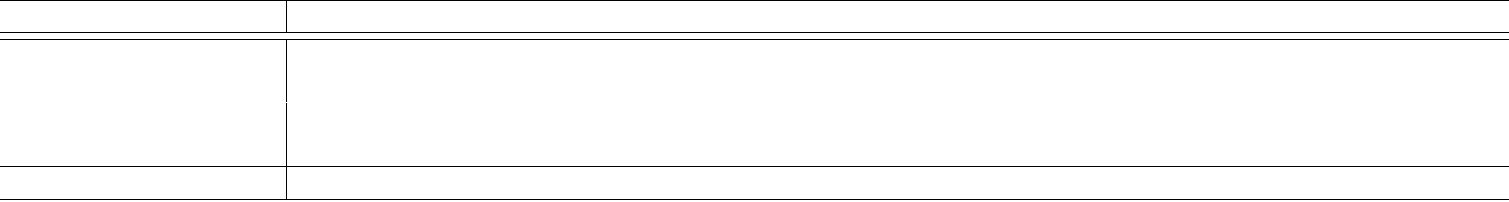
preted as representing any funding agencies.

In this case, we do not know the values of the parame-

ters in the sucient condition in Theorem 4.3, so we

cannot ﬁne-tune . We roughly scan through the en-

tire range of [0,1] for 1/ and the results are shown



Dezhi Hong, Quanquan Gu, Kamin Whitehouse

References

[14] J. D. Hamilton. Time series analysis, volume 2.

Princeton university press Princeton, 1994.

[1] R. Basri and D. W. Jacobs. Lambertian re-

ﬂectance and linear subspaces. Pattern Analysis

and Machine Intelligence, IEEE Transactions on,

25(2):218–233, 2003.

[15] F. Han, H. Lu, and H. Liu. A direct estimation of

high dimensional stationary vector autoregressions.

Journal of Machine Learning Research, 16:3115–

3150, 2015.

[2] S. Basu, G. Michailidis, et al. Regularized estima-

tion in sparse high-dimensional time series models.

The Annals of Statistics, 43(4):1535–1567, 2015.

[16] D. Hong, J. Ortiz, K. Whitehouse, and D. Culler.

Towards automatic spatial veriﬁcation of sensor

placement in buildings. In Proceedings of the 5th

ACM Workshop on Embedded Systems For Energy-

Ecient Buildings, pages 1–8. ACM, 2013.

[3] M. Bilenko, S. Basu, and R. J. Mooney. Integrating

constraints and metric learning in semi-supervised

clustering. In Proceedings of the 21st international

conference on Machine learning, page 11. ACM,

2004.

[17] A. Jalali, Y. Chen, S. Sanghavi, and H. Xu. Clus-

tering partially observed graphs via convex opti-

mization. In Proceedings of the 28th International

Conference on Machine Learning, pages 1001–1008,

2011.

[4] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eck-

stein. Distributed optimization and statistical

learning via the alternating direction method of

multipliers. Foundations and Trends in Machine

R

[18] E. Keogh. Exact indexing of dynamic time warp-

ing. In Proceedings of the 28th international con-

ference on Very Large Data Bases, pages 406–417.

VLDB Endowment, 2002.

Learning, 3(1):1–122, 2011.

[5] R. Brandenberg, A. Dattasharma, P. Gritzmann,

and D. Larman. Isoradial bodies. Discrete &

Computational Geometry, 32(4):447–457, 2004.

[19] A. Khaleghi, D. Ryabko, J. Mary, and P. Preux.

Consistent algorithms for clustering time series.

Journal of Machine Learning Research, 17(3):1–32,

2016.

[6] D. Cai, X. He, and J. Han. Document cluster-

ing using locality preserving indexing. Knowl-

edge and Data Engineering, IEEE Transactions

on, 17(12):1624–1637, 2005.

[20] J.-G. Lee, J. Han, and K.-Y. Whang. Trajectory

clustering: a partition-and-group framework. In

Proceedings of the 2007 ACM SIGMOD interna-

tional conference on Management of data, pages

593–604. ACM, 2007.

[7] E. Candes and T. Tao. The dantzig selector: sta-

tistical estimation when p is much larger than n.

The Annals of Statistics, pages 2313–2351, 2007.

[8] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang,

and E. Keogh. Querying and mining of time series

data: experimental comparison of representations

and distance measures. Proceedings of the VLDB

Endowment, 1(2):1542–1552, 2008.

[21] G. Liu, Z. Lin, and Y. Yu. Robust subspace seg-

mentation by low-rank representation. In Pro-

ceedings of the 27th international conference on

machine learning, pages 663–670, 2010.

[9] E. Elhamifar and R. Vidal. Sparse subspace clus-

tering. In Computer Vision and Pattern Recog-

nition, 2009. CVPR 2009. IEEE Conference on,

pages 2790–2797. IEEE, 2009.

[22] P. Montero and J. A. Vilar. Tsclust: An r package

for time series clustering.

[23] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral

clustering: Analysis and an algorithm. In Ad-

vances in Neural Information Processing Systems,

volume 14, pages 849–856, 2001.

[10] C. Faloutsos, M. Ranganathan, and Y. Manolopou-

los. Fast subsequence matching in time-series

databases, volume 23. ACM, 1994.

[24] A. Panuccio, M. Bicego, and V. Murino. A hidden

markov model-based approach to sequential data

clustering. In Structural, Syntactic, and Statisti-

cal Pattern Recognition, pages 734–743. Springer,

2002.

[11] P. Galeano and D. Pen˜a. Multivariate analysis in

vector time series. 2001.

[12] Gartner Inc. http://www.gartner.com/

newsroom/id/2636073.

[13] X. Golay, S. Kollias, G. Stoll, D. Meier, A. Vala-

vanis, and P. Boesiger. A new correlation-based

fuzzy logic clustering algorithm for fmri. Magnetic

Resonance in Medicine, 40(2):249–260, 1998.

[25] C. Qu and H. Xu. Subspace clustering with ir-

relevant features via robust dantzig selector. In

Advances in Neural Information Processing Sys-

tems, pages 757–765, 2015.



High-dimensional Time Series Clustering via Cross-Predictability

[26] T. Rakthanmanon, B. Campana, A. Mueen,

G. Batista, B. Westover, Q. Zhu, J. Zakaria, and

E. Keogh. Searching and mining trillions of time

series subsequences under dynamic time warping.

In Proceedings of the 18th ACM SIGKDD inter-

national conference on Knowledge discovery and

data mining, pages 262–270. ACM, 2012.

[27] D. Ryabko and J. Mary. Reducing statistical time-

series problems to binary classiﬁcation. In Ad-

vances in Neural Information Processing Systems,

pages 2060–2068, 2012.

[28] J. Shi and J. Malik. Normalized cuts and image

segmentation. Pattern Analysis and Machine In-

telligence, IEEE Transactions on, 22(8):888–905,

2000.

[29] P. Smyth. Clustering sequences with hidden

markov models. Advances in neural information

processing systems, pages 648–654, 1997.

[30] M. Soltanolkotabi and E. J. Candes. A geometric

analysis of subspace clustering with outliers. The

Annals of Statistics, pages 2195–2238, 2012.

[31] M. Soltanolkotabi, E. Elhamifar, E. J. Candes,

et al. Robust subspace clustering. The Annals of

Statistics, 42(2):669–699, 2014.

[32] M. J. Wainwright. Sharp thresholds for high-

dimensional and noisy sparsity recovery using-

constrained quadratic programming (lasso). Infor-

mation Theory, IEEE Transactions on, 55(5):2183–

2202, 2009.

[33] Y.-X. Wang and H. Xu. Noisy sparse subspace

clustering. Journal of Machine Learning Research,

17(12):1–41, 2016.

[34] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell.

Distance metric learning with application to clus-

tering with side-information. Advances in neural

information processing systems, 15:505–512, 2003.

