Dependency Discovery via Multiscale Graph Correlation

Cencheng Shen

University of Delaware

Collaborators: Carey E. Priebe, Joshua T. Vogelstein, Shangsi Wang, Ronak Mehta, Eric Bridgeford, Sambit Panda, Junhao Xiong, Youjin Lee, Qing Wang, Alex Badea, Xu Ting, Mauro Maggioni.

Overview

1. Motivation

2. Methodology

3. Theoretical Properties

4. Simulations and Experiments

5. Summary

Motivation

Motivation

Given paired data $(\mathcal{X}_n, \mathcal{Y}_n) = \{(x_i, y_i) \in \mathbb{R}^p \times \mathbb{R}^q, \text{ for } i = 1, \ldots, n\}$,

Motivation

Given paired data $(\mathcal{X}_n, \mathcal{Y}_n) = \{(x_i, y_i) \in \mathbb{R}^p \times \mathbb{R}^q, \text{ for } i = 1, \ldots, n\}$,

- Are they related?

Motivation

Given paired data $(\mathcal{X}_n, \mathcal{Y}_n) = \{(x_i, y_i) \in \mathbb{R}^p \times \mathbb{R}^q, \text{ for } i = 1, \ldots, n\}$,

- Are they related?
- How are they related?

Motivation

Given paired data $(\mathcal{X}_n, \mathcal{Y}_n) = \{(x_i, y_i) \in \mathbb{R}^p \times \mathbb{R}^q, \text{ for } i = 1, \ldots, n\}$,

- Are they related?
- How are they related?

Motivation

Given paired data $(\mathcal{X}_n, \mathcal{Y}_n) = \{(x_i, y_i) \in \mathbb{R}^p \times \mathbb{R}^q, \text{ for } i = 1, \ldots, n\}$,

- Are they related?
- How are they related?

X	Y
brain connectivity	creativity / personality
brain shape	health
gene / protein	cancer
social networks	attributes
anything	anything else

Formal Definition of Independence Testing

$$(x_i, y_i) \stackrel{i.i.d.}{\sim} F_{XY}, \quad i = 1, \ldots, n$$

$$H_0 : F_{XY} = F_X F_Y,$$
$$H_A : F_{XY} \neq F_X F_Y.$$

Formal Definition of Independence Testing

$$(x_i, y_i) \overset{i.i.d.}{\sim} F_{XY}, \quad i = 1, \ldots, n$$

$$H_0 : F_{XY} = F_X F_Y,$$
$$H_A : F_{XY} \neq F_X F_Y.$$

A dependence / correlation measure is any statistic that can be used to test the above hypothesis.

Formal Definition of Independence Testing

$$(x_i, y_i) \stackrel{i.i.d.}{\sim} F_{XY}, \quad i = 1, \ldots, n$$

$$H_0 : F_{XY} = F_X F_Y,$$
$$H_A : F_{XY} \neq F_X F_Y.$$

A dependence / correlation measure is any statistic that can be used to test the above hypothesis.

We desire a universally consistent test via some dependence measure, i.e., the testing power converges to 1 as $n \to \infty$ against any dependent F_{XY}.

Formal Definition of Independence Testing

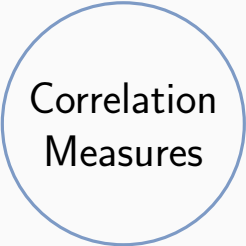$$(x_i, y_i) \overset{i.i.d.}{\sim} F_{XY}, \quad i = 1, \ldots, n$$

$$H_0 : F_{XY} = F_X F_Y,$$
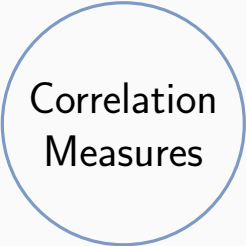$$H_A : F_{XY} \neq F_X F_Y.$$

A dependence / correlation measure is any statistic that can be used to test the above hypothesis.

We desire a universally consistent test via some dependence measure, i.e., the testing power converges to 1 as $n \to \infty$ against any dependent F_{XY}.

Without loss of generality, we shall assume F_{XY} has finite second moments.

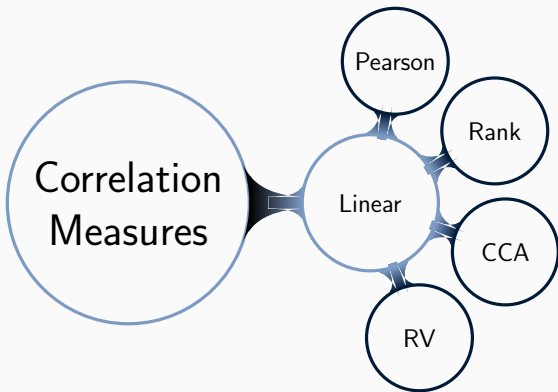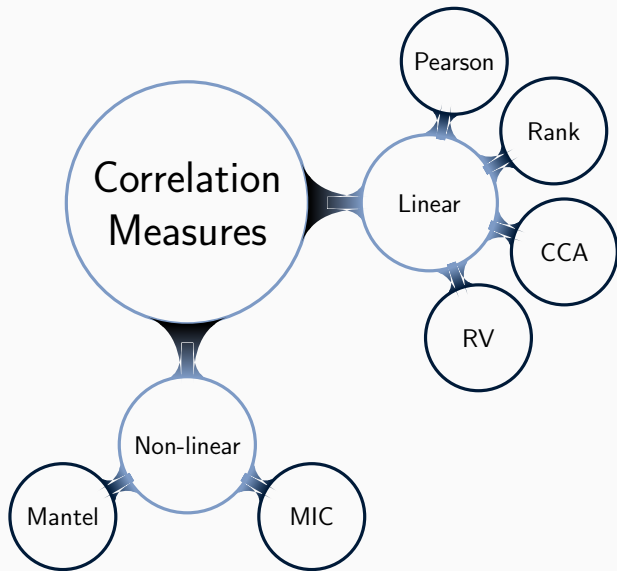# Benchmarks

Correlation
Measures
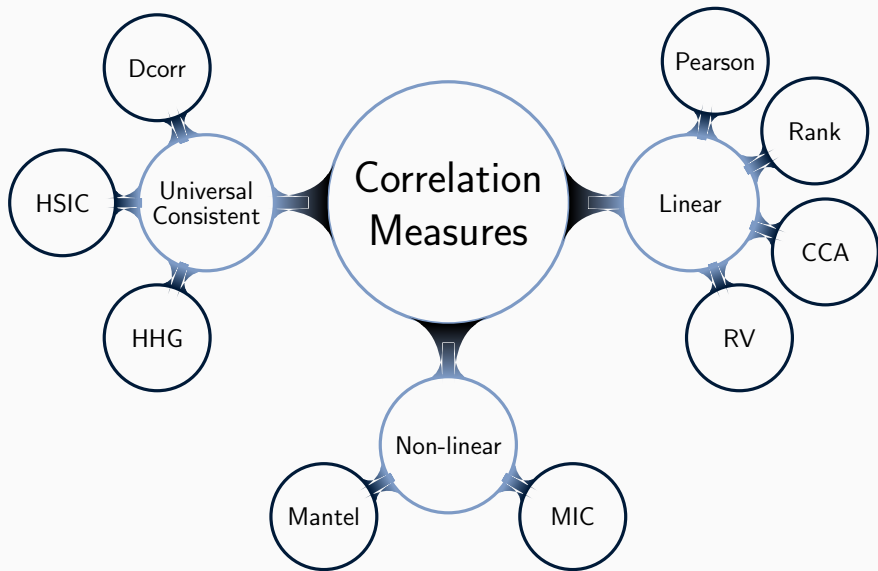
Benchmarks

Correlation
Measures

Benchmarks

Benchmarks

Benchmarks

Motivations

Modern data sets may be **high-dimensional, nonlinear, noisy, of limited sample size, structured, from disparate spaces**. Thus we desire a test that

Motivations

Modern data sets may be **high-dimensional, nonlinear, noisy, of limited sample size, structured, from disparate spaces**. Thus we desire a test that

- is consistent against all dependencies;

Motivations

Modern data sets may be **high-dimensional, nonlinear, noisy, of limited sample size, structured, from disparate spaces**. Thus we desire a test that

- is consistent against all dependencies;
- has good finite-sample testing performance;

Motivations

Modern data sets may be **high-dimensional, nonlinear, noisy, of limited sample size, structured, from disparate spaces**. Thus we desire a test that

- is consistent against all dependencies;
- has good finite-sample testing performance;
- is easy to understand and efficient to implement;

Motivations

Modern data sets may be **high-dimensional, nonlinear, noisy, of limited sample size, structured, from disparate spaces**. Thus we desire a test that

- is consistent against all dependencies;
- has good finite-sample testing performance;
- is easy to understand and efficient to implement;
- *provides insights into the dependency structure.

Motivations

Modern data sets may be **high-dimensional, nonlinear, noisy, of limited sample size, structured, from disparate spaces**. Thus we desire a test that

- is consistent against all dependencies;
- has good finite-sample testing performance;
- is easy to understand and efficient to implement;
- *provides insights into the dependency structure.

Motivations

Modern data sets may be **high-dimensional, nonlinear, noisy, of limited sample size, structured, from disparate spaces**. Thus we desire a test that

- is consistent against all dependencies;
- has good finite-sample testing performance;
- is easy to understand and efficient to implement;
- *provides insights into the dependency structure.
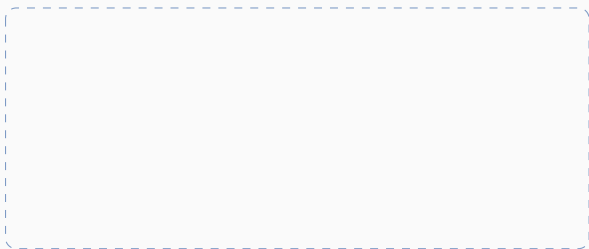
To that end, we propose the **multiscale graph correlation** in [*Shen et al.(2018)*].

Methodology

Flowchart of MGC

$$d(\cdot, \cdot) : \mathbb{R}^{p+q} \times \mathbb{R}^{p+q} \to \mathbb{R}$$

$(\mathcal{X}_n, \mathcal{Y}_n)$

Flowchart of MGC

$$d(\cdot, \cdot) : \mathbb{R}^{p+q} \times \mathbb{R}^{p+q} \to \mathbb{R}$$

$(\mathcal{X}_n, \mathcal{Y}_n) \longrightarrow$ Computing Distances & Centering: $A, B \in \mathbb{R}^{n \times n}$

Flowchart of MGC

$d(\cdot, \cdot) : \mathbb{R}^{p+q} \times \mathbb{R}^{p+q} \to \mathbb{R}$

$(\mathcal{X}_n, \mathcal{Y}_n) \longrightarrow$ Computing Distances & Centering: $A, B \in \mathbb{R}^{n \times n}$

$Dcov = \sum_{i,j=1}^{n} A_{ij} B_{ji}$

Flowchart of MGC

Flowchart of MGC

$d(\cdot, \cdot) : \mathbb{R}^{p+q} \times \mathbb{R}^{p+q} \to \mathbb{R}$

$(\mathcal{X}_n, \mathcal{Y}_n) \longrightarrow$
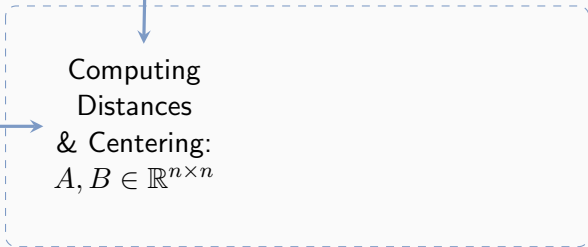
Computing Distances & Centering: $A, B \in \mathbb{R}^{n \times n}$

Incorporating Locality: $\{A^k, B^l \in \mathbb{R}^{n \times n}, \text{ for } k, l \in [n]\}$

$$Dcov = \sum_{i,j=1}^{n} A_{ij} B_{ji}$$

$$Dcov^{k,l} = \sum_{i,j=1}^{n} A_{ij}^k B_{ji}^l - \sum_{i,j=1}^{n} A_{ij}^k \sum_{i,j=1}^{n} B_{ij}^l$$

Flowchart of MGC

$d(\cdot, \cdot) : \mathbb{R}^{p+q} \times \mathbb{R}^{p+q} \to \mathbb{R}$

All Local
Correlations
$\{Dcorr^{k,l}\}$
$\in [-1, 1]^{n \times n}$

Computing
Distances
& Centering:
$A, B \in \mathbb{R}^{n \times n}$

Incorporating
Locality:
$\{A^k, B^l \in \mathbb{R}^{n \times n},$
for $k, l \in [n]\}$
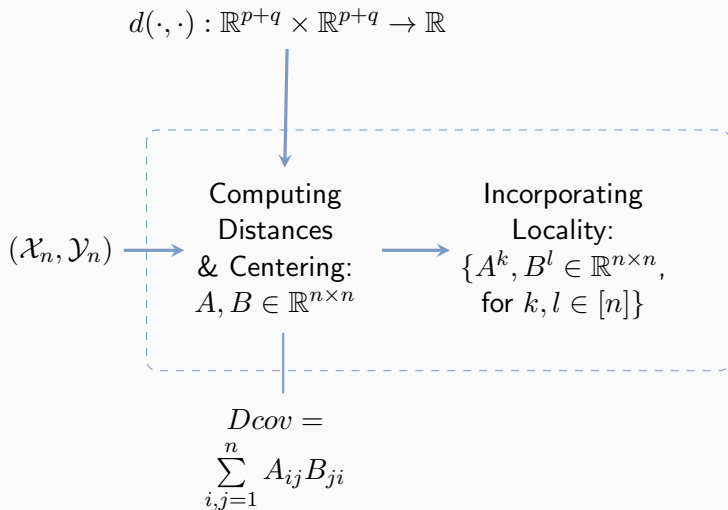
$(\mathcal{X}_n, \mathcal{Y}_n)$

$Dcov =$
$\sum_{i,j=1}^{n} A_{ij} B_{ji}$

$Dcov^{k,l} =$
$\sum_{i,j=1}^{n} A_{ij}^k B_{ji}^l - \sum_{i,j=1}^{n} A_{ij}^k \sum_{i,j=1}^{n} B_{ij}^l$

Flowchart of MGC

$d(\cdot, \cdot) : \mathbb{R}^{p+q} \times \mathbb{R}^{p+q} \to \mathbb{R}$

All Local Correlations $\{Dcorr^{k,l}\} \in [-1, 1]^{n \times n}$

Smoothed Maximum: $c^* \in [-1, 1]$, and Optimal Scale (k^*, l^*)

$(\mathcal{X}_n, \mathcal{Y}_n)$

Computing Distances & Centering: $A, B \in \mathbb{R}^{n \times n}$

Incorporating Locality: $\{A^k, B^l \in \mathbb{R}^{n \times n}, \text{ for } k, l \in [n]\}$

$$Dcov = \sum_{i,j=1}^{n} A_{ij} B_{ji}$$

$$Dcov^{k,l} = \sum_{i,j=1}^{n} A_{ij}^k B_{ji}^l - \sum_{i,j=1}^{n} A_{ij}^k \sum_{i,j=1}^{n} B_{ij}^l$$

Flowchart of MGC

$$d(\cdot,\cdot) : \mathbb{R}^{p+q} \times \mathbb{R}^{p+q} \to \mathbb{R}$$

$(\mathcal{X}_n, \mathcal{Y}_n)$

Computing Distances & Centering: $A, B \in \mathbb{R}^{n \times n}$

Incorporating Locality: $\{A^k, B^l \in \mathbb{R}^{n \times n},$ for $k, l \in [n]\}$

All Local Correlations $\{Dcorr^{k,l}\} \in [-1, 1]^{n \times n}$

Smoothed Maximum: $c^* \in [-1, 1]$, and Optimal Scale (k^*, l^*)

P-value by Permutation Test

$$Dcov = \sum_{i,j=1}^{n} A_{ij} B_{ji}$$

$$Dcov^{k,l} = \sum_{i,j=1}^{n} A_{ij}^k B_{ji}^l - \sum_{i,j=1}^{n} A_{ij}^k \sum_{i,j=1}^{n} B_{ij}^l$$

Computing Distance and Centering

Input: $\mathcal{X}_n = [x_1, \ldots, x_n]$ as the data matrix with each column representing one sample observation, and similarly $\mathcal{Y}_n$. A distance or kernel function $d(\cdot, \cdot) : \mathbb{R}^{p+q} \times \mathbb{R}^{p+q} \to \mathbb{R}$, by default the Euclidean distance.

Computing Distance and Centering

Input: $\mathcal{X}_n = [x_1, \ldots, x_n]$ as the data matrix with each column representing one sample observation, and similarly $\mathcal{Y}_n$. A distance or kernel function $d(\cdot, \cdot) : \mathbb{R}^{p+q} \times \mathbb{R}^{p+q} \to \mathbb{R}$, by default the Euclidean distance.

Distance Computation: Let $\tilde{A}$ be the $n \times n$ Euclidean distance matrices of $\mathcal{X}_n$:

$$\tilde{A}_{ij} = d(x_i, x_j) = \|x_i - x_j\|_2,$$

and similarly $\tilde{B}$ from $\mathcal{Y}_n$.

Computing Distance and Centering

Input: $\mathcal{X}_n = [x_1, \ldots, x_n]$ as the data matrix with each column representing one sample observation, and similarly $\mathcal{Y}_n$. A distance or kernel function $d(\cdot, \cdot) : \mathbb{R}^{p+q} \times \mathbb{R}^{p+q} \to \mathbb{R}$, by default the Euclidean distance.

Distance Computation: Let $\tilde{A}$ be the $n \times n$ Euclidean distance matrices of $\mathcal{X}_n$:

$$\tilde{A}_{ij} = d(x_i, x_j) = \|x_i - x_j\|_2,$$

and similarly $\tilde{B}$ from $\mathcal{Y}_n$.

Centering: Then we center $\tilde{A}$ and $\tilde{B}$ by columns, with the diagonals excluded:

$$A_{ij} = \begin{cases} \tilde{A}_{ij} - \frac{1}{n-1} \sum_{s=1}^n \tilde{A}_{sj}, & \text{if } i \neq j, \\ 0, & \text{if } i = j; \end{cases} \tag{1}$$

similarly for B.

Incorporating the Locality Principle

Incorporating the Locality Principle

Ranking: Define $\{R_{ij}^A\}$ as the "rank" of x_i relative to x_j, that is, $R_{ij}^A = k$ if x_i is the kth closest point (or "neighbor") to x_j, as determined by ranking the set $\{\tilde{A}_{1j}, \tilde{A}_{2j}, \ldots, \tilde{A}_{nj}\}$ by ascending order. Similarly define R_{ij}^B for the y's.

Incorporating the Locality Principle

Ranking: Define $\{R_{ij}^A\}$ as the "rank" of x_i relative to x_j, that is, $R_{ij}^A = k$ if x_i is the kth closest point (or "neighbor") to x_j, as determined by ranking the set $\{\tilde{A}_{1j}, \tilde{A}_{2j}, \ldots, \tilde{A}_{nj}\}$ by ascending order. Similarly define R_{ij}^B for the y's.

For any $(k, l) \in [n]^2$, define the rank truncated matrices A^k, B^l, and the joint distance matrix C^{kl} as

$$A_{ij}^k = A_{ij} \boldsymbol{I}(R_{ij}^A \leq k),$$
$$B_{ij}^l = B_{ij} \boldsymbol{I}(R_{ij}^B \leq l).$$

Incorporating the Locality Principle

Ranking: Define $\{R_{ij}^A\}$ as the "rank" of x_i relative to x_j, that is, $R_{ij}^A = k$ if x_i is the kth closest point (or "neighbor") to x_j, as determined by ranking the set $\{\tilde{A}_{1j}, \tilde{A}_{2j}, \ldots, \tilde{A}_{nj}\}$ by ascending order. Similarly define R_{ij}^B for the y's.

For any $(k, l) \in [n]^2$, define the rank truncated matrices A^k, B^l, and the joint distance matrix C^{kl} as

$$A_{ij}^k = A_{ij} \boldsymbol{I}(R_{ij}^A \leq k),$$
$$B_{ij}^l = B_{ij} \boldsymbol{I}(R_{ij}^B \leq l).$$

When ties occur, minimal rank is recommended, e.g., if Y only takes two value, R_{ij}^B takes value in $\{1, 2\}$ only. We assume no ties for each of presentation.

C. Shen

Local Distance Correlations

Local Distance Correlations

A Family of Local Correlations: Let $\circ$ denote the entry-wise product, $\hat{E}(\cdot) = \frac{1}{n(n-1)} \sum_{i \neq j}^{n}(\cdot)$ denote the diagonal-excluded sample mean of a square matrix, then the sample local covariance, variance, and correlation are defined as:

Local Distance Correlations

A Family of Local Correlations: Let $\circ$ denote the entry-wise product, $\hat{E}(\cdot) = \frac{1}{n(n-1)} \sum_{i \neq j}^{n} (\cdot)$ denote the diagonal-excluded sample mean of a square matrix, then the sample local covariance, variance, and correlation are defined as:

$$dCov^{k,l}(\mathcal{X}_n, \mathcal{Y}_n) = \hat{E}(A^k \circ B^{l'}) - \hat{E}(A^k)\hat{E}(B^l),$$
$$dVar^k(\mathcal{X}_n) = \hat{E}(A^k \circ A^{k'}) - \hat{E}^2(A^k),$$
$$dVar^l(\mathcal{Y}_n) = \hat{E}(B^l \circ B^{l'}) - \hat{E}^2(B^l),$$
$$dCorr^{k,l}(\mathcal{X}_n, \mathcal{Y}_n) = dCov^{k,l}(\mathcal{X}_n, \mathcal{Y}_n)/\sqrt{dVar^k(\mathcal{X}_n) \cdot dVar^l(\mathcal{Y}_n)}.$$

Local Distance Correlations

A Family of Local Correlations: Let $\circ$ denote the entry-wise product, $\hat{E}(\cdot) = \frac{1}{n(n-1)} \sum_{i \neq j}^{n} (\cdot)$ denote the diagonal-excluded sample mean of a square matrix, then the sample local covariance, variance, and correlation are defined as:

$$dCov^{k,l}(\mathcal{X}_n, \mathcal{Y}_n) = \hat{E}(A^k \circ B^{l'}) - \hat{E}(A^k)\hat{E}(B^l),$$

$$dVar^k(\mathcal{X}_n) = \hat{E}(A^k \circ A^{k'}) - \hat{E}^2(A^k),$$

$$dVar^l(\mathcal{Y}_n) = \hat{E}(B^l \circ B^{l'}) - \hat{E}^2(B^l),$$

$$dCorr^{k,l}(\mathcal{X}_n, \mathcal{Y}_n) = dCov^{k,l}(\mathcal{X}_n, \mathcal{Y}_n) / \sqrt{dVar^k(\mathcal{X}_n) \cdot dVar^l(\mathcal{Y}_n)}.$$

for $k, l = 1, \ldots, n$. If $dVar^k(\mathcal{X}_n) \cdot dVar^l(\mathcal{X}_n) \leq 0$, we set $dCorr^{kl}(\mathcal{X}_n, \mathcal{Y}_n) = 0$ instead.

Local Distance Correlations

A Family of Local Correlations: Let $\circ$ denote the entry-wise product, $\hat{E}(\cdot) = \frac{1}{n(n-1)} \sum_{i \neq j}^{n} (\cdot)$ denote the diagonal-excluded sample mean of a square matrix, then the sample local covariance, variance, and correlation are defined as:

$$dCov^{k,l}(\mathcal{X}_n, \mathcal{Y}_n) = \hat{E}(A^k \circ B^{l'}) - \hat{E}(A^k)\hat{E}(B^l),$$
$$dVar^k(\mathcal{X}_n) = \hat{E}(A^k \circ A^{k'}) - \hat{E}^2(A^k),$$
$$dVar^l(\mathcal{Y}_n) = \hat{E}(B^l \circ B^{l'}) - \hat{E}^2(B^l),$$
$$dCorr^{k,l}(\mathcal{X}_n, \mathcal{Y}_n) = dCov^{k,l}(\mathcal{X}_n, \mathcal{Y}_n)/\sqrt{dVar^k(\mathcal{X}_n) \cdot dVar^l(\mathcal{Y}_n)}.$$

for $k, l = 1, \ldots, n$. If $dVar^k(\mathcal{X}_n) \cdot dVar^l(\mathcal{X}_n) \leq 0$, we set $dCorr^{kl}(\mathcal{X}_n, \mathcal{Y}_n) = 0$ instead.

There are a maximum of n^2 different local correlations. At $k = l = n$, $dCorr^{kl}(\mathcal{X}_n, \mathcal{Y}_n)$ equals the "global" distance correlation $dCorr(\mathcal{X}_n, \mathcal{Y}_n)$ by *Szekely et al.(2007)*.

Smoothed Maximum $c^*(\mathcal{X}_n, \mathcal{Y}_n)$

One would like to use the optimal local correlation for testing.

Smoothed Maximum $c^*(\mathcal{X}_n, \mathcal{Y}_n)$

One would like to use the optimal local correlation for testing.

But directly taking the maximum local correlation

$$\max_{(k,l) \in [n]^2} \{Dcorr^{k,l}(\mathcal{X}_n, \mathcal{Y}_n)\}$$

will yield a biased statistic under independence, i.e., the maximum is always larger than 0 in expectation even under independent relationship!

Smoothed Maximum $c^*(\mathcal{X}_n, \mathcal{Y}_n)$

One would like to use the optimal local correlation for testing.

But directly taking the maximum local correlation

$$\max_{(k,l)\in[n]^2}\{Dcorr^{k,l}(\mathcal{X}_n, \mathcal{Y}_n)\}$$

will yield a biased statistic under independence, i.e., the maximum is always larger than 0 in expectation even under independent relationship!

Instead, we take a smoothed maximum, by finding a connected region in the local correlation map with significant local correlatons – if such a region exists, use the maximum within the region.

Smoothed Maximum

Pick a threshold $\tau \geq 0$ (we choose by an approximate null distribution of $Dcorr$, which is symmetric beta and converges to 0 as $n \to \infty$), compute the set

$$\{(k, l) \text{ such that } Dcorr^{k,l}(\mathcal{X}_n, \mathcal{Y}_n) > \max\{\tau, Dcorr(\mathcal{X}_n, \mathcal{Y}_n)\}\},$$

Smoothed Maximum

Pick a threshold $\tau \geq 0$ (we choose by an approximate null distribution of $Dcorr$, which is symmetric beta and converges to 0 as $n \to \infty$), compute the set

$$\{(k,l) \text{ such that } Dcorr^{k,l}(\mathcal{X}_n, \mathcal{Y}_n) > \max\{\tau, Dcorr(\mathcal{X}_n, \mathcal{Y}_n)\}\},$$

and calculate the largest connected component R of the set.

Smoothed Maximum

Pick a threshold $\tau \geq 0$ (we choose by an approximate null distribution of $Dcorr$, which is symmetric beta and converges to 0 as $n \to \infty$), compute the set

$$\{(k, l) \text{ such that } Dcorr^{k,l}(\mathcal{X}_n, \mathcal{Y}_n) > \max\{\tau, Dcorr(\mathcal{X}_n, \mathcal{Y}_n)\}\},$$

and calculate the largest connected component R of the set.

If there are sufficiently many elements in R $(> 2n)$, take the maximum correlation within R as MGC statistic $c^*(\mathcal{X}_n, \mathcal{Y}_n)$,

Smoothed Maximum

Pick a threshold $\tau \geq 0$ (we choose by an approximate null distribution of $Dcorr$, which is symmetric beta and converges to 0 as $n \to \infty$), compute the set

$$\{(k, l) \text{ such that } Dcorr^{k,l}(\mathcal{X}_n, \mathcal{Y}_n) > \max\{\tau, Dcorr(\mathcal{X}_n, \mathcal{Y}_n)\}\},$$

and calculate the largest connected component R of the set.

If there are sufficiently many elements in R $(> 2n)$, take the maximum correlation within R as MGC statistic $c^*(\mathcal{X}_n, \mathcal{Y}_n)$, and set the neighborhood pair as the optimal scale (k^*, l^*).

Permutation Test

To get a p-value by MGC for any given data, we utilize the permutation test: randomly permute index of the second data set for r times, compute the permuted MGC statistic $c^*(\mathcal{X}_n, \mathcal{Y}_n^\pi)$ for each permutation π, and estimate

$$Prob(c^*(\mathcal{X}_n, \mathcal{Y}_n) > c^*(\mathcal{X}_n, \mathcal{Y}_n^\pi))$$

as the p-value.

Permutation Test

To get a p-value by MGC for any given data, we utilize the permutation test: randomly permute index of the second data set for r times, compute the permuted MGC statistic $c^*(\mathcal{X}_n, \mathcal{Y}_n^\pi)$ for each permutation π, and estimate

$$Prob(c^*(\mathcal{X}_n, \mathcal{Y}_n) > c^*(\mathcal{X}_n, \mathcal{Y}_n^\pi))$$

as the p-value.

This is a standard nonparametric testing procedure employed by Mantel, Dcorr, HHG, HSIC, where the null distribution of the dependency measure cannot be exactly derived.

Computation Complexity

- Distance computation takes $\mathcal{O}(n^2 \max(p, q))$
- Centering takes $\mathcal{O}(n^2)$
- Ranking takes $\mathcal{O}(n^2 log(n))$
- **All local correlations can be iteratively computed in $\mathcal{O}(n^2)$**
- The smoothed maximum takes $\mathcal{O}(n^2)$
- Storage requirement is $\mathcal{O}(n^2)$

Computation Complexity

- Distance computation takes $\mathcal{O}(n^2 \max(p, q))$
- Centering takes $\mathcal{O}(n^2)$
- Ranking takes $\mathcal{O}(n^2 log(n))$
- **All local correlations can be iteratively computed in $\mathcal{O}(n^2)$**
- The smoothed maximum takes $\mathcal{O}(n^2)$
- Storage requirement is $\mathcal{O}(n^2)$

Overall, MGC can be computed in $\mathcal{O}(n^2 \max(p, q, \log n))$. Without the ranking process, the global correlation (Dcorr) waives the $\log n$ part and takes $\mathcal{O}(n^2 \max(p, q))$.

Computation Complexity

- Distance computation takes $\mathcal{O}(n^2 \max(p, q))$
- Centering takes $\mathcal{O}(n^2)$
- Ranking takes $\mathcal{O}(n^2 log(n))$
- **All local correlations can be iteratively computed in $\mathcal{O}(n^2)$**
- The smoothed maximum takes $\mathcal{O}(n^2)$
- Storage requirement is $\mathcal{O}(n^2)$

Overall, MGC can be computed in $\mathcal{O}(n^2 \max(p, q, \log n))$. Without the ranking process, the global correlation (Dcorr) waives the $\log n$ part and takes $\mathcal{O}(n^2 \max(p, q))$.

The permutation test takes $\mathcal{O}(n^2 \max(r, p, q, \log n))$ for r random permutations.

Computation Complexity

- Distance computation takes $\mathcal{O}(n^2 \max(p, q))$
- Centering takes $\mathcal{O}(n^2)$
- Ranking takes $\mathcal{O}(n^2 log(n))$
- **All local correlations can be iteratively computed in $\mathcal{O}(n^2)$**
- The smoothed maximum takes $\mathcal{O}(n^2)$
- Storage requirement is $\mathcal{O}(n^2)$

Overall, MGC can be computed in $\mathcal{O}(n^2 \max(p, q, \log n))$. Without the ranking process, the global correlation (Dcorr) waives the $\log n$ part and takes $\mathcal{O}(n^2 \max(p, q))$.

The permutation test takes $\mathcal{O}(n^2 \max(r, p, q, \log n))$ for r random permutations.

On a standard PC with Matlab, testing $n = 1000$ takes about 1 minutes.
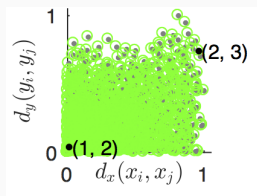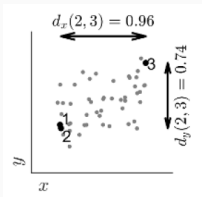
Examples

Examples
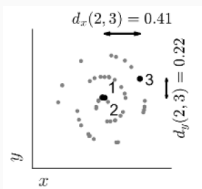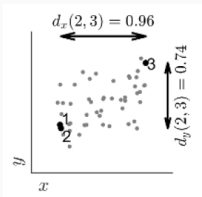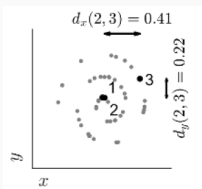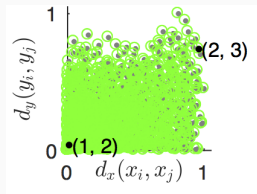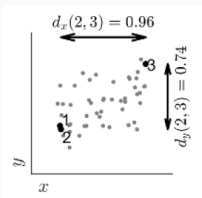
Examples

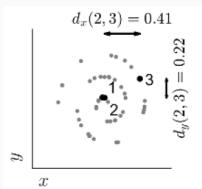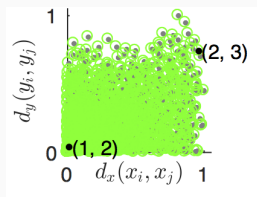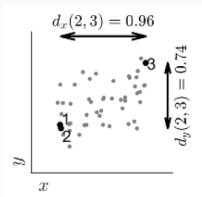# Examples

$Dcorr(\mathcal{X}_n, \mathcal{Y}_n) = 0.15$
$MGC(\mathcal{X}_n, \mathcal{Y}_n) = 0.15$
p-vals: < 0.001

Examples

$Dcorr(\mathcal{X}_n, \mathcal{Y}_n) = 0.15$
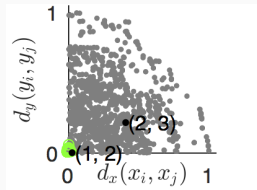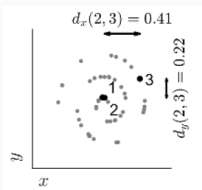$MGC(\mathcal{X}_n, \mathcal{Y}_n) = 0.15$
p-vals: < 0.001

Examples
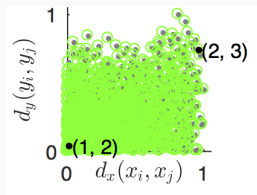

$Dcorr(\mathcal{X}_n, \mathcal{Y}_n) = 0.15$
$MGC(\mathcal{X}_n, \mathcal{Y}_n) = 0.15$
p-vals: < 0.001
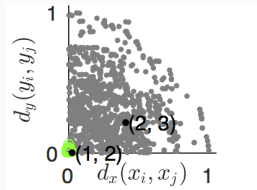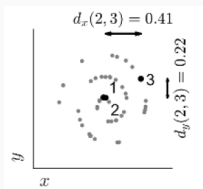
$Dcorr(\mathcal{X}_n, \mathcal{Y}_n) = 0.01$
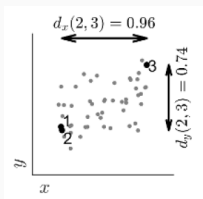$MGC(\mathcal{X}_n, \mathcal{Y}_n) = 0.13$
p-vals: 0.3 vs < 0.001

MGC is applicable to similarity / kernel matrix

Theorem 1 (Transforming kernel to distance)

Given any kernel function $k(\cdot, \cdot)$, define an induced semi-metric as

$$d(i, j) = 1 - k(i, j) / \max_{i, j = 1, \ldots, n} \{k(i, j)\}.$$

MGC is applicable to similarity / kernel matrix

Theorem 1 (Transforming kernel to distance)

Given any kernel function $k(\cdot, \cdot)$, define an induced semi-metric as

$$d(i,j) = 1 - k(i,j)/\max_{i,j=1,\ldots,n}\{k(i,j)\}.$$

Namely, given a sample kernel matrices $K_{n \times n}$, one can compute the induced distance matrix by

$$D = J - K/\max_{i,j \in [1,\ldots,n]^2}\{K(i,j)\},$$

and apply MGC or other distance-based correlation to the induced distance matrices.

MGC is applicable to similarity / kernel matrix

Theorem 1 (Transforming kernel to distance)

Given any kernel function $k(\cdot, \cdot)$, define an induced semi-metric as

$$d(i,j) = 1 - k(i,j)/\max_{i,j=1,\ldots,n}\{k(i,j)\}.$$

Namely, given a sample kernel matrices $K_{n \times n}$, one can compute the induced distance matrix by

$$D = J - K/\max_{i,j \in [1,\ldots,n]^2}\{K(i,j)\},$$

and apply MGC or other distance-based correlation to the induced distance matrices.

The kernel correlation HSIC is equivalent to distance correlation.

Theoretical Properties

Basic Properties of Sample MGC

Theorem 2 (Well-behaved Correlation Measure)

Basic Properties of Sample MGC

Theorem 2 (Well-behaved Correlation Measure)

1. *Boundedness:* $c^*(\mathcal{X}_n, \mathcal{Y}_n) \in [-1, 1]$.

Basic Properties of Sample MGC

Theorem 2 (Well-behaved Correlation Measure)

1. *Boundedness:* $c^*(\mathcal{X}_n, \mathcal{Y}_n) \in [-1, 1]$.

2. *Symmetric:* $c^*(\mathcal{X}_n, \mathcal{Y}_n) = c^*(\mathcal{Y}_n, \mathcal{X}_n)$.

Basic Properties of Sample MGC

Theorem 2 (Well-behaved Correlation Measure)

1. *Boundedness:* $c^*(\mathcal{X}_n, \mathcal{Y}_n) \in [-1, 1]$.

2. *Symmetric:* $c^*(\mathcal{X}_n, \mathcal{Y}_n) = c^*(\mathcal{Y}_n, \mathcal{X}_n)$.

3. *Invariant:* $c^*(\mathcal{X}_n, \mathcal{Y}_n)$ is invariant to any distance-preserving transformations ϕ, δ applied to $\mathcal{X}_n$ and $\mathcal{Y}_n$ each (i.e., rotation, scaling, translation, reflection).

Basic Properties of Sample MGC

Theorem 2 (Well-behaved Correlation Measure)

1. *Boundedness:* $c^*(\mathcal{X}_n, \mathcal{Y}_n) \in [-1, 1]$.

2. *Symmetric:* $c^*(\mathcal{X}_n, \mathcal{Y}_n) = c^*(\mathcal{Y}_n, \mathcal{X}_n)$.

3. *Invariant:* $c^*(\mathcal{X}_n, \mathcal{Y}_n)$ *is invariant to any distance-preserving transformations* ϕ, δ *applied to* $\mathcal{X}_n$ *and* $\mathcal{Y}_n$ *each (i.e., rotation, scaling, translation, reflection).*

4. *1-Linear:* $c^*(\mathcal{X}_n, \mathcal{Y}_n) = 1$ *if and only if* F_X *is non-degenerate and* (X, uY) *are dependent via an isometry for some non-zero constant* u.

Consistency of Sample MGC

Theorem 3 (Consistency)

Consistency of Sample MGC

Theorem 3 (Consistency)

1. *0-Indep:* $c^*(\mathcal{X}_n, \mathcal{Y}_n) \overset{n \to \infty}{\to} 0$ *if and only if independence.*

Consistency of Sample MGC

Theorem 3 (Consistency)

1. *0-Indep*: $c^*(\mathcal{X}_n, \mathcal{Y}_n) \overset{n \to \infty}{\to} 0$ *if and only if independence.*

2. *Valid Test: Under the permutation test, Sample MGC is a valid test, i.e., it controls the type 1 error level α.*

Consistency of Sample MGC

Theorem 3 (Consistency)

1. *0-Indep:* $c^*(\mathcal{X}_n, \mathcal{Y}_n) \overset{n \to \infty}{\to} 0$ *if and only if independence.*

2. *Valid Test: Under the permutation test, Sample MGC is a valid test, i.e., it controls the type 1 error level* α.

3. *Consistency: At any type 1 error level* α, *testing power* $\beta(c^*(\mathcal{X}_n, \mathcal{Y}_n)) \overset{n \to \infty}{\to} 1$ *against any dependent* F_{XY}.

Consistency of Sample MGC

Theorem 3 (Consistency)

1. 0-Indep: $c^*(\mathcal{X}_n, \mathcal{Y}_n) \overset{n \to \infty}{\to} 0$ if and only if independence.

2. Valid Test: Under the permutation test, Sample MGC is a valid test, i.e., it controls the type 1 error level α.

3. Consistency: At any type 1 error level α, testing power $\beta(c^*(\mathcal{X}_n, \mathcal{Y}_n)) \overset{n \to \infty}{\to} 1$ against any dependent F_{XY}.

The distance correlation also shares the same properties.

Defining Population MGC

Suppose $(X, Y), (X', Y'), (X'', Y''), (X''', Y''')$ are *iid* as F_{XY}.

Defining Population MGC

Suppose $(X, Y), (X', Y'), (X'', Y''), (X''', Y''')$ are *iid* as F_{XY}. Let $\boldsymbol{I}(\cdot)$ be the indicator function, define two random variables

$$\boldsymbol{I}_{X,X'}^{\rho_k} = \boldsymbol{I}(\int_{B(X,d(X',X))} dF_X(u) \leq \rho_k)$$

$$\boldsymbol{I}_{Y',Y}^{\rho_l} = \boldsymbol{I}(\int_{B(Y',d(Y'-Y))} dF_Y(u) \leq \rho_l)$$

for $\rho_k, \rho_l \in [0, 1]$.

Defining Population MGC

Suppose $(X, Y), (X', Y'), (X'', Y''), (X''', Y''')$ are *iid* as F_{XY}. Let $\boldsymbol{I}(\cdot)$ be the indicator function, define two random variables

$$\boldsymbol{I}_{X,X'}^{\rho_k} = \boldsymbol{I}(\int_{B(X,d(X',X))} dF_X(u) \leq \rho_k)$$

$$\boldsymbol{I}_{Y',Y}^{\rho_l} = \boldsymbol{I}(\int_{B(Y',d(Y'-Y))} dF_Y(u) \leq \rho_l)$$

for $\rho_k, \rho_l \in [0, 1]$. Further define

$$g_X^{\rho_k} = (d(X, X') - d(X, X''))\boldsymbol{I}_{X,X'}^{\rho_k}$$

$$g_{Y'}^{\rho_l} = (d(Y', Y) - d(Y', Y'''))\boldsymbol{I}_{Y',Y}^{\rho_l}$$

Defining Population MGC

Suppose $(X, Y), (X', Y'), (X'', Y''), (X''', Y''')$ are *iid* as F_{XY}. Let $\boldsymbol{I}(\cdot)$ be the indicator function, define two random variables

$$\boldsymbol{I}_{X,X'}^{\rho_k} = \boldsymbol{I}(\int_{B(X,d(X',X))} dF_X(u) \leq \rho_k)$$

$$\boldsymbol{I}_{Y',Y}^{\rho_l} = \boldsymbol{I}(\int_{B(Y',d(Y'-Y))} dF_Y(u) \leq \rho_l)$$

for $\rho_k, \rho_l \in [0, 1]$. Further define

$$g_X^{\rho_k} = (d(X, X') - d(X, X''))\boldsymbol{I}_{X,X'}^{\rho_k}$$

$$g_{Y'}^{\rho_l} = (d(Y', Y) - d(Y', Y'''))\boldsymbol{I}_{Y',Y}^{\rho_l}$$

The population local covariance can be defined as

$$Dcov^{\rho_k, \rho_l}(X, Y) = E(g_X^{\rho_k} g_{Y'}^{\rho_l}) - E(g_X^{\rho_k})E(g_{Y'}^{\rho_l}).$$

Defining Population MGC

Suppose $(X, Y), (X', Y'), (X'', Y''), (X''', Y''')$ are *iid* as F_{XY}. Let $\boldsymbol{I}(\cdot)$ be the indicator function, define two random variables

$$\boldsymbol{I}_{X,X'}^{\rho_k} = \boldsymbol{I}(\int_{B(X,d(X',X))} dF_X(u) \leq \rho_k)$$

$$\boldsymbol{I}_{Y',Y}^{\rho_l} = \boldsymbol{I}(\int_{B(Y',d(Y'-Y))} dF_Y(u) \leq \rho_l)$$

for $\rho_k, \rho_l \in [0, 1]$. Further define

$$g_X^{\rho_k} = (d(X, X') - d(X, X''))\boldsymbol{I}_{X,X'}^{\rho_k}$$

$$g_{Y'}^{\rho_l} = (d(Y', Y) - d(Y', Y'''))\boldsymbol{I}_{Y',Y}^{\rho_l}$$

The population local covariance can be defined as

$$Dcov^{\rho_k,\rho_l}(X, Y) = E(g_X^{\rho_k} g_{Y'}^{\rho_l}) - E(g_X^{\rho_k})E(g_{Y'}^{\rho_l}).$$

Normalizing and taking a smoothed maximum yield population MGC.

Sample to Population

Sample to Population

Under the Euclidean distance, the population version can be equivalently defined via an integral of characteristic functions of $F_{XY} - F_X F_Y$ with respect to a non-negative weight function $w(t, s)$.

Sample to Population

Under the Euclidean distance, the population version can be equivalently defined via an integral of characteristic functions of $F_{XY} - F_X F_Y$ with respect to a non-negative weight function $w(t, s)$.

For general metric or kernel function $d(\cdot, \cdot)$, the population version can also be defined as the integral of $d(X, X')d(Y, Y')$ with respect to $(F_{XY} - F_X F_Y)(F_{X'Y'} - F_{X'} F_{Y'})$.

Sample to Population

Under the Euclidean distance, the population version can be equivalently defined via an integral of characteristic functions of $F_{XY} - F_X F_Y$ with respect to a non-negative weight function $w(t, s)$.

For general metric or kernel function $d(\cdot, \cdot)$, the population version can also be defined as the integral of $d(X, X')d(Y, Y')$ with respect to $(F_{XY} - F_X F_Y)(F_{X'Y'} - F_{X'} F_{Y'})$.

When the metric is of strong negative type or the kernel is characteristic, $c^*(X, Y) = 0$ if and only if independence. For arbitrary metric or kernel, the if direction is still true but not the only if direction.

Theoretical Advantages of MGC

Theorem 4 (Convergence, Mean and Variance)

Theoretical Advantages of MGC

Theorem 4 (Convergence, Mean and Variance)

1. 0-Indep: When the metric is of strong negative type or the kernel is characteristic, $c^*(X, Y) = 0$ if and only if independence.

Theoretical Advantages of MGC

Theorem 4 (Convergence, Mean and Variance)

1. 0-Indep: When the metric is of strong negative type or the kernel is characteristic, $c^*(X, Y) = 0$ if and only if independence.

2. Convergence: $c^*(\mathcal{X}_n, \mathcal{Y}_n) \overset{n \to \infty}{\to} c^*(X, Y)$.

Theoretical Advantages of MGC

Theorem 4 (Convergence, Mean and Variance)

1. 0-Indep: When the metric is of strong negative type or the kernel is characteristic, $c^*(X, Y) = 0$ if and only if independence.

2. Convergence: $c^*(\mathcal{X}_n, \mathcal{Y}_n) \overset{n \to \infty}{\to} c^*(X, Y)$.

3. Almost Unbiased: $E(c^*(\mathcal{X}_n, \mathcal{Y}_n)) = c^*(X, Y) + \mathcal{O}(1/n)$.

Theoretical Advantages of MGC

Theorem 4 (Convergence, Mean and Variance)

1. 0-Indep: When the metric is of strong negative type or the kernel is characteristic, $c^*(X, Y) = 0$ if and only if independence.

2. Convergence: $c^*(\mathcal{X}_n, \mathcal{Y}_n) \overset{n \to \infty}{\to} c^*(X, Y)$.

3. Almost Unbiased: $E(c^*(\mathcal{X}_n, \mathcal{Y}_n)) = c^*(X, Y) + \mathcal{O}(1/n)$.

4. Diminishing Variance: $Var(c^*(\mathcal{X}_n, \mathcal{Y}_n)) = \mathcal{O}(1/n)$.

Theoretical Advantages of MGC

Theorem 4 (Convergence, Mean and Variance)

1. 0-Indep: When the metric is of strong negative type or the kernel is characteristic, $c^*(X, Y) = 0$ if and only if independence.

2. Convergence: $c^*(\mathcal{X}_n, \mathcal{Y}_n) \overset{n \to \infty}{\to} c^*(X, Y)$.

3. Almost Unbiased: $E(c^*(\mathcal{X}_n, \mathcal{Y}_n)) = c^*(X, Y) + \mathcal{O}(1/n)$.

4. Diminishing Variance: $Var(c^*(\mathcal{X}_n, \mathcal{Y}_n)) = \mathcal{O}(1/n)$.

The last three properties also hold for any local correlation by $(\rho_k, \rho_l) = (\frac{k-1}{n-1}, \frac{l-1}{n-1})$, as well as the distance correlation, i.e., $k = l = n$.

Theoretical Advantages of MGC

Theorem 5 (Advantages of Population MGC vs Dcorr)

Theoretical Advantages of MGC

Theorem 5 (Advantages of Population MGC vs Dcorr)

1. For any dependent F_{XY}, $c^*(X, Y) \geq Dcorr(X, Y)$.

Theoretical Advantages of MGC

Theorem 5 (Advantages of Population MGC vs Dcorr)

1. For any dependent F_{XY}, $c^*(X, Y) \geq Dcorr(X, Y)$.

2. There exists dependent F_{XY} such that $c^*(X, Y) > Dcorr(X, Y)$.

Theoretical Advantages of MGC

Theorem 5 (Advantages of Population MGC vs Dcorr)

1. For any dependent F_{XY}, $c^*(X, Y) \geq Dcorr(X, Y)$.

2. There exists dependent F_{XY} such that $c^*(X, Y) > Dcorr(X, Y)$.

As MGC and Dcorr share similar variance and same mean under the null, the first moment advantage in the alternative is translated to the testing power.

Theoretical Advantages of MGC

Theorem 5 (Advantages of Population MGC vs Dcorr)

1. For any dependent F_{XY}, $c^*(X, Y) \geq Dcorr(X, Y)$.

2. There exists dependent F_{XY} such that $c^*(X, Y) > Dcorr(X, Y)$.

As MGC and Dcorr share similar variance and same mean under the null, the first moment advantage in the alternative is translated to the testing power.

Theorem 6 (Optimal Scale of MGC Implies Geometry Structure)

Theoretical Advantages of MGC

Theorem 5 (Advantages of Population MGC vs Dcorr)

1. For any dependent F_{XY}, $c^*(X, Y) \geq Dcorr(X, Y)$.

2. There exists dependent F_{XY} such that $c^*(X, Y) > Dcorr(X, Y)$.

As MGC and Dcorr share similar variance and same mean under the null, the first moment advantage in the alternative is translated to the testing power.

Theorem 6 (Optimal Scale of MGC Implies Geometry Structure)

If the relationship is linear (or with independent noise), the global scale is always optimal and $c^*(X, Y) = Dcorr(X, Y)$.

Theoretical Advantages of MGC

Theorem 5 (Advantages of Population MGC vs Dcorr)

1. For any dependent F_{XY}, $c^*(X, Y) \geq Dcorr(X, Y)$.

2. There exists dependent F_{XY} such that $c^*(X, Y) > Dcorr(X, Y)$.

As MGC and Dcorr share similar variance and same mean under the null, the first moment advantage in the alternative is translated to the testing power.

Theorem 6 (Optimal Scale of MGC Implies Geometry Structure)

If the relationship is linear (or with independent noise), the global scale is always optimal and $c^*(X, Y) = Dcorr(X, Y)$.
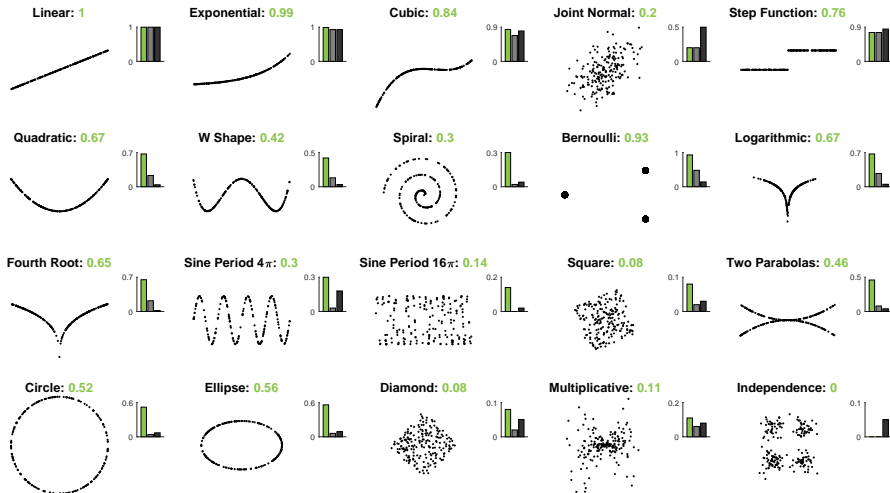
Conversely, the optimal scale being local, i.e., $c^*(X, Y) > Dcorr(X, Y)$, implies a non-linear relationship.

Simulations and Experiments

Visualizations of 20 Simulation Settings

Visualizations of 20 Simulation Settings

MGC, Distance Correlation, and Pearson's Correlation for 20 Dependencies

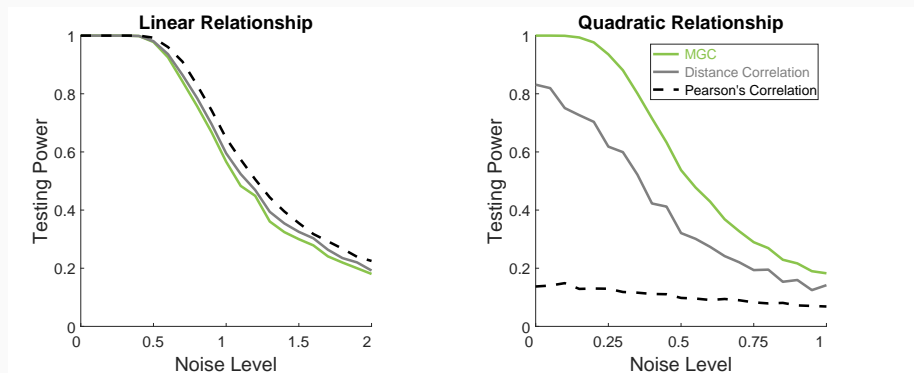# Testing Power: Linear vs Nonlinear

Power is the probability of rejecting the null when the alternative is true.

Testing Power: Linear vs Nonlinear

Power is the probability of rejecting the null when the alternative is true.

$$n = 30, p = q = 1,$$
$$X \sim Uniform(-1, 1),$$
$$\epsilon \sim Normal(0, noise),$$
$$Y = X + \epsilon \text{ and } Y = X^2 + \epsilon.$$

Required Sample Size

Required Sample Size

Required sample size $N_{\alpha,\beta}(c)$ to achieve a power of β at type 1 error level α using a statistic c. We compute the required sample size $N_{\alpha=0.05, \beta=0.85}(c)$:

Required Sample Size

Required sample size $N_{\alpha,\beta}(c)$ to achieve a power of β at type 1 error level α using a statistic c. We compute the required sample size $N_{\alpha=0.05,\beta=0.85}(c)$:

in linear relationship, 40 for all three methods;
in quadratic relationship, 80 for MGC, 180 for Dcorr, and > 1000 for Pearson.

Required Sample Size

Required sample size $N_{\alpha,\beta}(c)$ to achieve a power of β at type 1 error level α using a statistic c. We compute the required sample size $N_{\alpha=0.05,\beta=0.85}(c)$:

in linear relationship, 40 for all three methods;
in quadratic relationship, 80 for MGC, 180 for Dcorr, and > 1000 for Pearson.

Next we compute the size for each simulation, and summarize by the median over close-to-linear (type 1-5) and strongly non-linear relationships (type 6-19).

Required Sample Size

Required sample size $N_{\alpha,\beta}(c)$ to achieve a power of β at type 1 error level α using a statistic c. We compute the required sample size $N_{\alpha=0.05,\beta=0.85}(c)$:

in linear relationship, 40 for all three methods;
in quadratic relationship, 80 for MGC, 180 for Dcorr, and > 1000 for Pearson.

Next we compute the size for each simulation, and summarize by the median over close-to-linear (type 1-5) and strongly non-linear relationships (type 6-19).

We consider univariate (1D) and multivariate (10D) cases.

Median Size Table

Testing Methods	1D Lin	1D Non-Lin	10D Lin	10D Non-Lin
MGC	**50**	**90**	60	**165**
Dcorr	**50**	250	60	515
Pearson / RV / CCA	**50**	>1000	**50**	>1000
HHG	70	**90**	100	315
HSIC	70	95	100	400
MIC	120	180	n/a	n/a

Extracting Signal Brain Region from fMRI images

Extracting Signal Brain Region from fMRI images

We consider predicting the site and sex based on functional magnetic resonance image (fMRI) graphs. Two datasets used are SWU4 and HNU1, which have 467 and 300 samples respectively.

Each sample is an fMRI scan registered to the MNI152 template using the Desikan altas, which has 70 regions. They are transformed to graph structure using the NeuroDatas MRI Graphs pipeline [1].

We compute the dependency measure between each brain region and sex. Rank the brain region via magnitude of the measure, and include all significant ($p - val < 0.05$) brain regions. Then run leave-one-out cross validation with K-Nearest Neighbor classifier to verify the results. Repeat it for the site property.

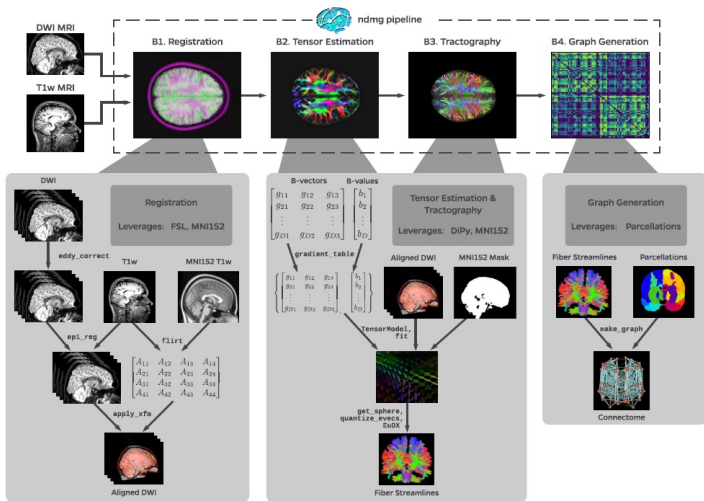[1] https://github.com/neurodata/ndmg

Figure 7: **ndmg detailed pipeline**. The NDMG pipeline consists of 4 main steps: Registration (**B1**), Tensor Estimation (**B2**), Tractography (**B3**), and Graph Generation (**B4**). Each of these sections leverages pubicly available tools and data to robustly produce the desired derivative of each step. Alongside derivative production, NDMG produces QA figures at each stage, as can be seen in **B1-4**, that enable qualitative evaluation of the pipeline's performance.

Desikan Atlas With Site Difference Highlighted
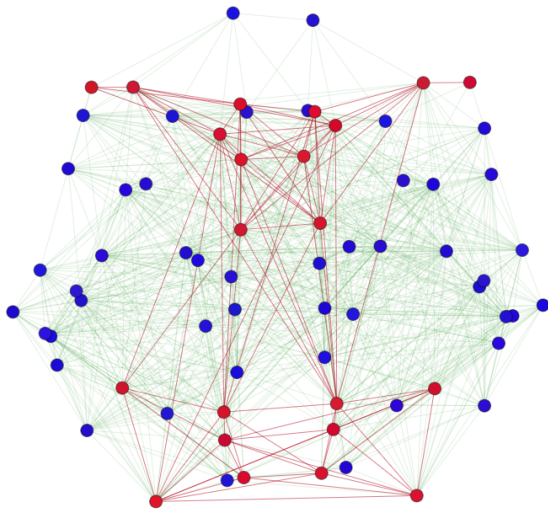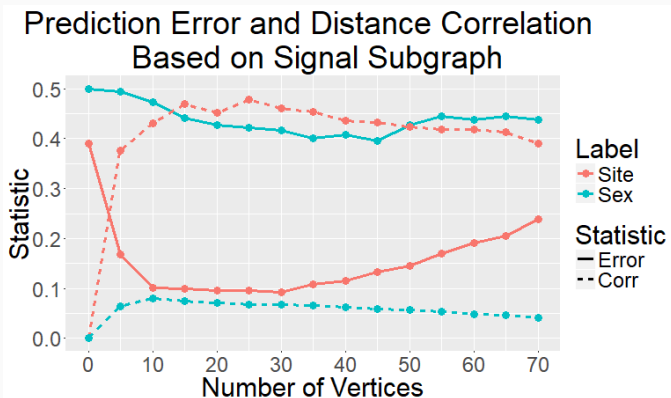
Figure: A total of 22 regions are recognized for site difference, which maximizes the MGC statistic and almost minimizes the leave-one-out cross validation error. It is no longer the case for sex, for which neither the MGC nor the error are too significant for any size of subgraph.

Summary

Summary

Summary

Distance-based correlation is valid and universally consistent for testing independence. MGC utilizes the locality principle to achieve better testing power and sheds insight into the dependency structure.

Summary

Distance-based correlation is valid and universally consistent for testing independence. MGC utilizes the locality principle to achieve better testing power and sheds insight into the dependency structure.

- Using a proper distance or kernel ensures the universal consistency.

Summary

Distance-based correlation is valid and universally consistent for testing independence. MGC utilizes the locality principle to achieve better testing power and sheds insight into the dependency structure.

- Using a proper distance or kernel ensures the universal consistency.
- Compute all local correlations iteratively and finds the optimal one can help the testing power when sample size is limited.

Summary

Distance-based correlation is valid and universally consistent for testing independence. MGC utilizes the locality principle to achieve better testing power and sheds insight into the dependency structure.

- Using a proper distance or kernel ensures the universal consistency.
- Compute all local correlations iteratively and finds the optimal one can help the testing power when sample size is limited.
- The optimal scale gives information on linear vs nonlinear dependency.

Summary

Distance-based correlation is valid and universally consistent for testing independence. MGC utilizes the locality principle to achieve better testing power and sheds insight into the dependency structure.

- Using a proper distance or kernel ensures the universal consistency.
- Compute all local correlations iteratively and finds the optimal one can help the testing power when sample size is limited.
- The optimal scale gives information on linear vs nonlinear dependency.
- It can be used in a variety of applications to replace the Pearson's correlation.

Advantages of MGC

Advantages of MGC

1. Performant under any joint distribution of finite second moments:

Advantages of MGC

1. Performant under any joint distribution of finite second moments:
   - Equals 0 asymptotically if and only if independence.

Advantages of MGC

1. Performant under any joint distribution of finite second moments:
   - Equals 0 asymptotically if and only if independence.
   - Amplify the dependency signal while mostly avoiding the sample bias.

Advantages of MGC

1. Performant under any joint distribution of finite second moments:
   - Equals 0 asymptotically if and only if independence.
   - Amplify the dependency signal while mostly avoiding the sample bias.
   - Superior finite-sample performance over all benchmarks, against linear / nonlinear / noisy / high-dimensional relationships.

Advantages of MGC

1. Performant under any joint distribution of finite second moments:
   - Equals 0 asymptotically if and only if independence.
   - Amplify the dependency signal while mostly avoiding the sample bias.
   - Superior finite-sample performance over all benchmarks, against linear / nonlinear / noisy / high-dimensional relationships.

Advantages of MGC

1. Performant under any joint distribution of finite second moments:
   - Equals 0 asymptotically if and only if independence.
   - Amplify the dependency signal while mostly avoiding the sample bias.
   - Superior finite-sample performance over all benchmarks, against linear / nonlinear / noisy / high-dimensional relationships.

2. It works for:

Advantages of MGC

1. Performant under any joint distribution of finite second moments:
   - Equals 0 asymptotically if and only if independence.
   - Amplify the dependency signal while mostly avoiding the sample bias.
   - Superior finite-sample performance over all benchmarks, against linear / nonlinear / noisy / high-dimensional relationships.

2. It works for:
   - Low- and high-dimensional data.

Advantages of MGC

1. Performant under any joint distribution of finite second moments:
   - Equals 0 asymptotically if and only if independence.
   - Amplify the dependency signal while mostly avoiding the sample bias.
   - Superior finite-sample performance over all benchmarks, against linear / nonlinear / noisy / high-dimensional relationships.

2. It works for:
   - Low- and high-dimensional data.
   - Euclidean and structured data (e.g., images, networks, shapes).

Advantages of MGC

1. Performant under any joint distribution of finite second moments:
   - Equals 0 asymptotically if and only if independence.
   - Amplify the dependency signal while mostly avoiding the sample bias.
   - Superior finite-sample performance over all benchmarks, against linear / nonlinear / noisy / high-dimensional relationships.

2. It works for:
   - Low- and high-dimensional data.
   - Euclidean and structured data (e.g., images, networks, shapes).
   - Any dissimilarity / similarity / kernel matrix.

Advantages of MGC

1. Performant under any joint distribution of finite second moments:
   - Equals 0 asymptotically if and only if independence.
   - Amplify the dependency signal while mostly avoiding the sample bias.
   - Superior finite-sample performance over all benchmarks, against linear / nonlinear / noisy / high-dimensional relationships.

2. It works for:
   - Low- and high-dimensional data.
   - Euclidean and structured data (e.g., images, networks, shapes).
   - Any dissimilarity / similarity / kernel matrix.

Advantages of MGC

1. Performant under any joint distribution of finite second moments:
   - Equals 0 asymptotically if and only if independence.
   - Amplify the dependency signal while mostly avoiding the sample bias.
   - Superior finite-sample performance over all benchmarks, against linear / nonlinear / noisy / high-dimensional relationships.

2. It works for:
   - Low- and high-dimensional data.
   - Euclidean and structured data (e.g., images, networks, shapes).
   - Any dissimilarity / similarity / kernel matrix.

3. Intuitive to understand and efficient to implement in $\mathcal{O}(n^2 log n)$.

Some Recent Advances in Computation

In practice:

Some Recent Advances in Computation

In practice:

1. Distance correlation and MGC can now be tested without resorting to permutation (similar to Pearson's t-test).

Some Recent Advances in Computation

In practice:

1. Distance correlation and MGC can now be tested without resorting to permutation (similar to Pearson's t-test).

2. When $p = q = 1$ and using Euclidean distance, there is a special fast implementation of distance correlation. The running time becomes $O(n \log n)$ and storage requirement becomes $O(n)$, making it ideal and scalable to millions and billions of observations. (less than 10 seconds for 1 million observations on a standard PC using Matlab)

Some Recent Advances in Computation

In practice:

1. Distance correlation and MGC can now be tested without resorting to permutation (similar to Pearson's t-test).

2. When $p = q = 1$ and using Euclidean distance, there is a special fast implementation of distance correlation. The running time becomes $O(n \log n)$ and storage requirement becomes $O(n)$, making it ideal and scalable to millions and billions of observations. (less than 10 seconds for 1 million observations on a standard PC using Matlab)

Thus when n is small (say less than a few thousands), MGC is the better choice; whereas distance correlation can better handle extremely large data.

Open Source Packages

Python package in https://github.com/neurodata/mgcpy/ and
forthcoming in scikit-learn

R package in https://github.com/neurodata/MGC/ and CRAN

Matlab code https://github.com/neurodata/mgc-matlab

References

1. C. Shen, C. E. Priebe, and J. T. Vogelstein, "From distance correlation to the multiscale graph correlation," Journal of the American Statistical Association, 2019.

2. J. T. Vogelstein, E. Bridgeford, Q. Wang, C. E. Priebe, M. Maggioni, and C. Shen, "Discovering and Deciphering Relationships Across Disparate Data Modalities," eLife, 2019.

3. Y. Lee, C. Shen, and J. T. Vogelstein, "Network dependence testing via diffusion maps and distance-based correlations," Biometrika, 2019.

4. S. Wang, C. Shen, A. Badea, C. E. Priebe, and J. T. Vogelstein, "Signal subgraph estimation via iterative vertex screening," under review.

5. C. Shen and J. T. Vogelstein, "The Exact Equivalence of Distance and Kernel Methods for Hypothesis Testing," under review.