Fast Computing via Recursive Dyadic Partitioning for Statistical Dependency

Workshop on Frame Theory and Sparse Representation for Complex Data
June 1, 2017

Xiaoming Huo
Georgia Institute of Technology
School of industrial and systems engineering

Agenda

I. Statistical Dependence
II. Distance Correlation
III. Fast Algorithm
IV. Simulations
V. An Application

I. Statistical Dependence

- Correlation
- Shortcoming of Pearson's linear correlation
- Related works

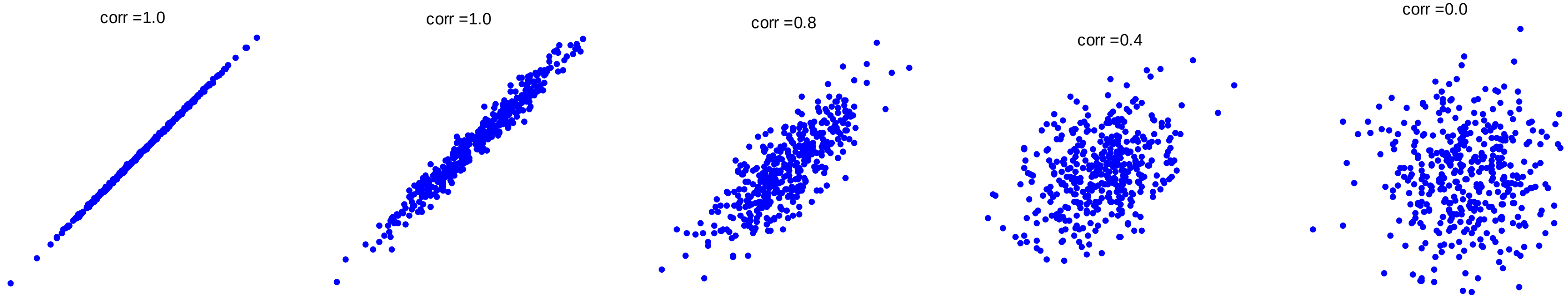
Linear Dependency (Pearson's)

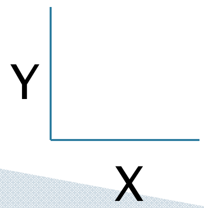
- Pearson's linear correlation coefficient:

$$\mathrm{Corr}(X,Y) = \frac{Cov(X,Y)}{\sqrt{Var(X)Var(Y)}}$$

- Karl Pearson (1895)

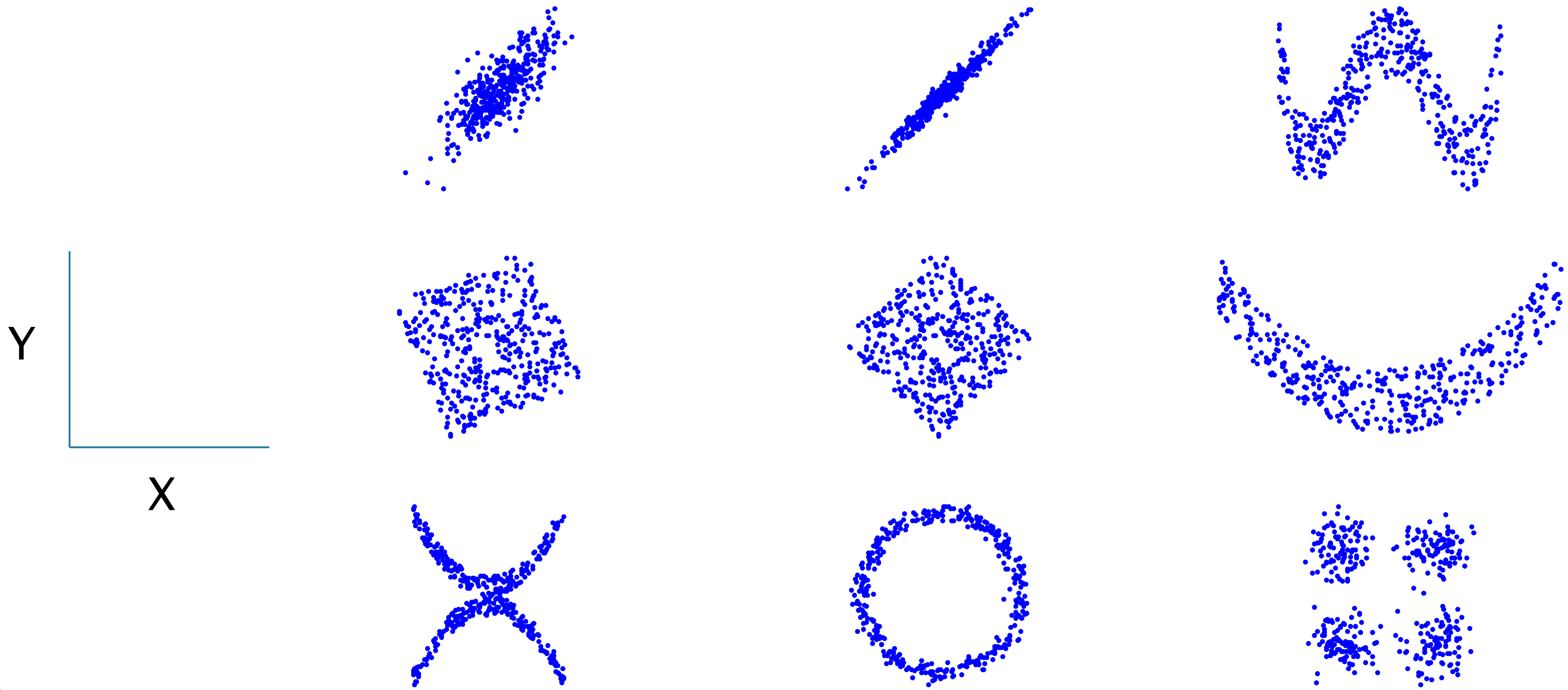
Corr=1.0 Corr=1.0 Corr=0.8 Corr=0.4 Corr=0.0



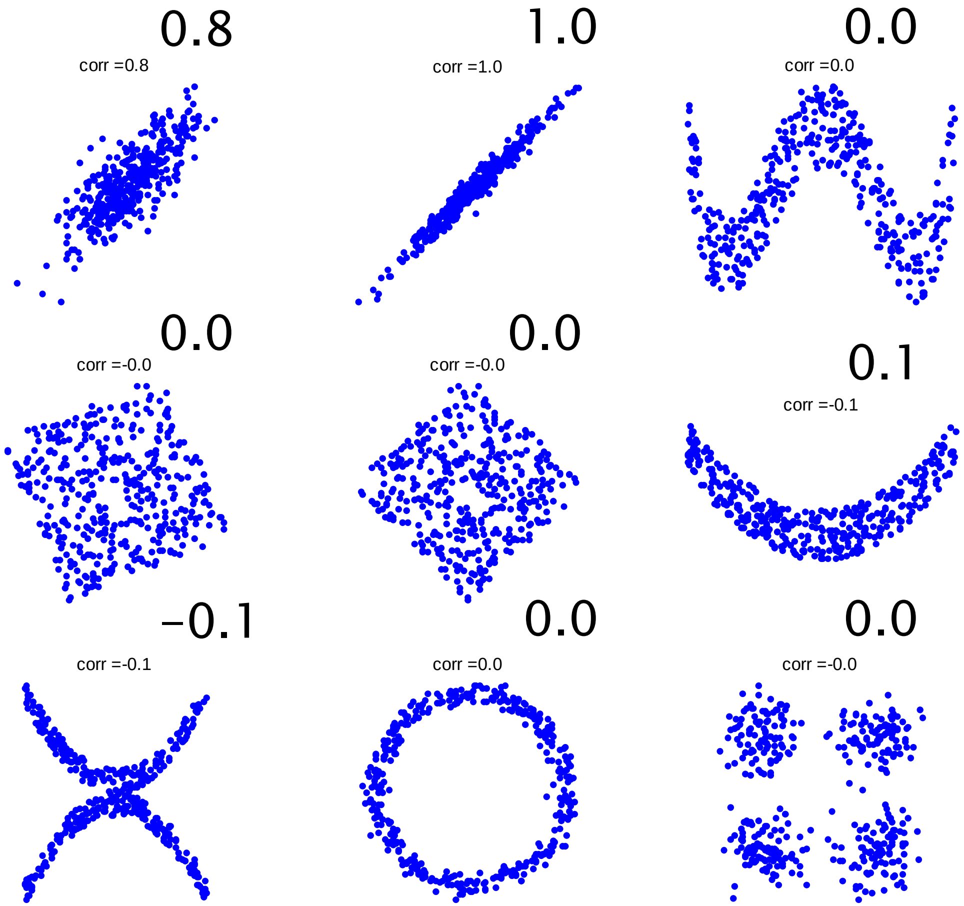


Nonlinearity

- Dependency could be complicated



Pearson's corr. not effective

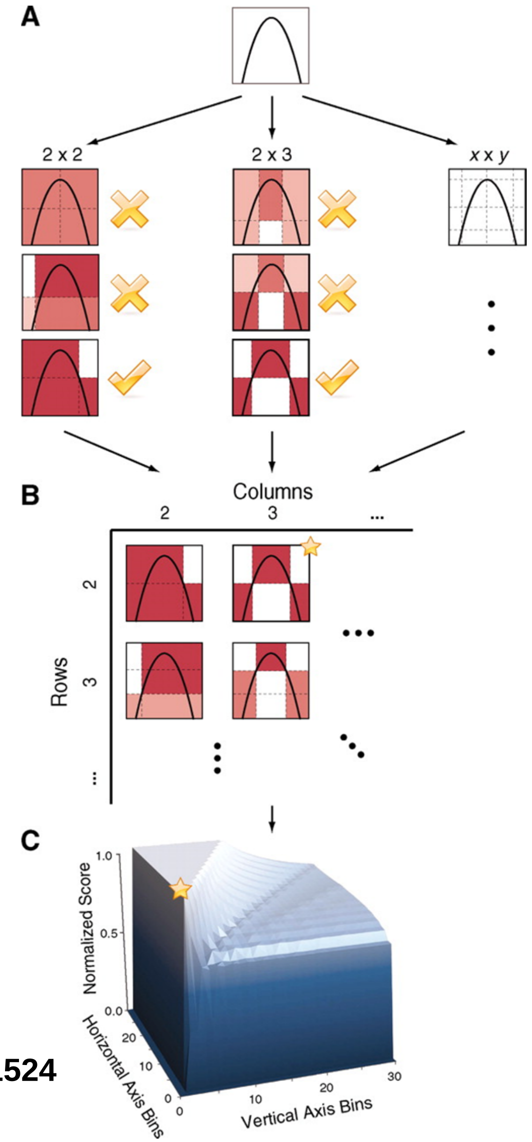


Related work

- Alternating Conditional Expectations or backfitting algorithm (ACE). Breiman and Friedman. JASA 1985.
- Kernel Canonical Correlation Analysis (KCCA). Bach and Jordan. JMLR 2002.
- (Copula) Maximum Mean Discrepancy (MMD, CMMD). Gretton, Borgwardt, Rasch, Scholkopf, and Smola. JMLR 2012. Poczos, Ghahramani, and Schneider. ICML, 2012.
- Maximal Information Coefficient (MIC). Reshef et al. Science, 2011.
- Randomized Dependence Coefficient (RDC). Lopez-Paz, Hennig, and Scholkopf. NIPS 2013.

Maximal Information Coefficient (MIC)

- Science, 2011
- "A Correlation for the 21st Century" – Terry Speed



Computing MIC For each pair (X, Y), the MIC algorithm finds the x-by-y grid with the highest induced mutual information.

D N Reshef et al. Science 2011;334:1518-1524

Kendall's τ

- 1938
- Let, $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ be a set of observations of the joint random variables X and Y respectively
- Kendall τ coefficient

$$\tau = \frac{1}{\binom{n}{2}} \sum_{1 \le i < j \le n} \operatorname{sign}[(x_i - x_j)(y_i - y_j)]$$

- $-1 \le \tau \le 1$

Comparison

- Comparison between dependence measures

Name of Coeff.	Comp. cost
Pearson's ρ	n
Spearman's ρ	$n \log n$
Kendall's τ	$n \log n$
CCA	n
KCCA	n^3
ACE	n
MIC	2^n
MMD	n^2
CMMD	n^2
RDC	$n \log n$
dCor	$n^2 \rightarrow n \ \log n$

II. Distance Correlation

1. Distance correlation
2. Sample dCor
3. An example

How to measure statistical dependence?

- Independence: f(x,y)=f(x)f(y)
  - Joint density is the multiplication of two marginal densities
- Hope:
  - X and Y independent if and only if corr(X,Y)=0
  - If X=c_1·Y+c_2, then corr(X,Y)=1
- Pearson's correlation coefficient not effective

Distance covariance

- Gabor J. Szekely, 2005, 2007 (AoS), 2009 (AoAS), 2012 (SPL), 2014 (AoS)
- Distance covariance: (population version)

$$\mathcal{V}^2(X,Y) = \left\|\phi_{X,Y}(t,s) - \phi_X(t)\phi_Y(s)\right\|_w^2$$

$$:= \int_{R^{p+q}} \left|\phi_{X,Y}(t,s) - \phi_X(t)\phi_Y(s)\right|^2 w(t,s)dt\, ds$$

where $\phi_{X,Y}$, ϕ_X, and ϕ_Y are characteristic func.

- Weight $w(t,s) = (|t|_p^{1+p}|s|_q^{1+q})^{-1}$ to ensure the above integral is well defined…

Distance correlation

- Correlation:

$$dCor = \frac{\mathcal{V}\,(X,Y)}{\sqrt{\mathcal{V}(X,X)\cdot\mathcal{V}(Y,Y)}}$$

A Fact

- An equation:

$$2\pi \int_{-\infty}^{\infty} (F(x) - G(x))^2 = \underbrace{\int_{-\infty}^{\infty} \frac{\left|\hat{f}(t) - \hat{g}(t)\right|^2}{t^2} dt}_{\text{Distance correlation}}$$

- Distance correlation ~ difference between cumulative distribution functions.

Sample Distance Covariance

- Pairwise distances: $a_{ij} = \|X_i - X_j\|, 1 \le i, j \le n$
- Similarly, $b_{ij} = \|Y_i - Y_j\|$
- Centered matrix:

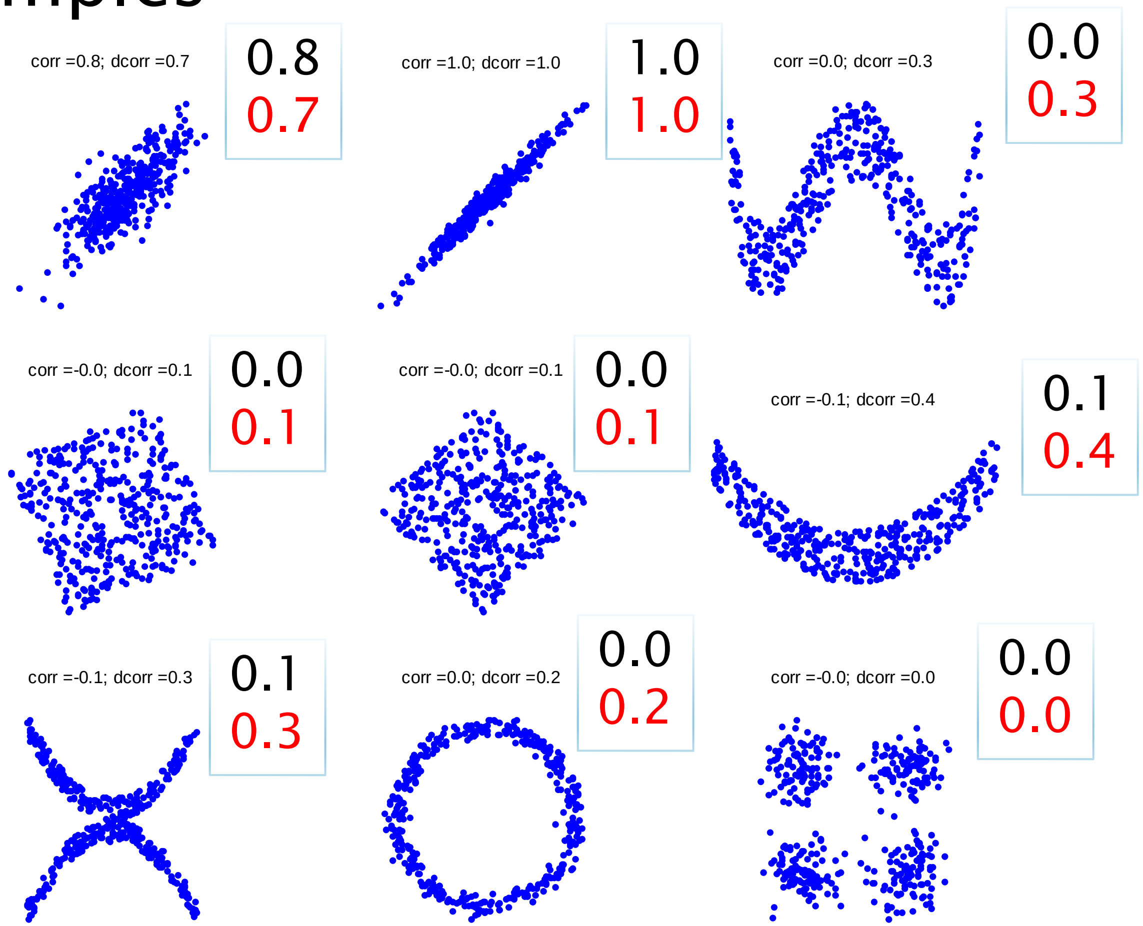
$$A_{ij} = \begin{cases} a_{ij} - \frac{\sum_{\ell=1}^{n} a_{i\ell}}{n-2} - \frac{\sum_{k=1}^{n} a_{kj}}{n-2} + \frac{\sum_{k,\ell=1}^{n} a_{k\ell}}{(n-1)(n-2)}, & i \neq j; \\ 0, & i = j \end{cases}$$

- Similarly, B_{ij}.
- An unbiased estimator of $\mathcal{V}^2(X,Y)$:

$$(A \cdot B) = \frac{\sum_{i \neq j} A_{ij} B_{ij}}{n(n-3)}$$

Compare with Dist. Corr.

- Same examples



III. Fast Algorithm

- Reformulation
- Partials sums
- 2-D dyadic partitioning and updating

Reformulation

- Requiring pairwise distances undesirable (n^2)
- Denote $a_{i\cdot} = \sum_{\ell=1}^{n} a_{i\ell}$, $1 \le i \le n$; similarly for $b_{i\cdot}$
- Denote $a_{\cdot\cdot} = \sum_{i,\ell=1}^{n} a_{i\ell}$; similarly for $b_{\cdot\cdot}$
- We have

$$(A \cdot B) = \frac{\sum_{i\neq j} a_{ij} b_{ij}}{n(n-3)} - \frac{2\sum_{i=1}^{n} a_{i\cdot} b_{i\cdot}}{n(n-2)(n-3)} + \frac{a_{\cdot\cdot} b_{\cdot\cdot}}{n(n-1)(n-2)(n-3)}$$

Main ideas towards an O(n log n) algorithm

- $a_{i\cdot}$ and $b_{i\cdot}$ can be related to partial sums – O(n) algorithm
- We designed a ***dyadic updating*** scheme to compute for

$$\sum_{i \neq j} a_{ij} b_{ij} = \sum_{i \neq j} |x_i - x_j| \cdot |y_i - y_j|$$

- An O(n log n) algorithm

Fast method for ALL $a_{i\cdot}$ ($b_{i\cdot}$)

- Recall $a_{i\cdot} = \sum_{l=1}^{n} a_{il} = \sum_{l=1}^{n} |x_i - x_l|$
- Sorting $x_1, x_2, \ldots, x_n$ takes (on average) $O(n \log n)$
- Computes ALL $\alpha_i = \sum_{l:\, x_l < x_i} 1$ takes $O(n)$
- Computes ALL $\beta_i = \sum_{l:\, x_l < x_i} x_l$ takes $O(n)$
- Note

$$a_{i\cdot} = \sum_{i=1}^{n} x_i + (2\alpha_i - n)x_i - 2\beta_i$$

- Overall, computing for ALL $a_{i\cdot}$'s takes $O(n \log n)$

Fast Method for cross terms

- Recall we want to compute for

$$\sum_{i \neq j} a_{ij} b_{ij} = \sum_{i \neq j} |x_i - x_j| \cdot |y_i - y_j|$$

- Define $S_{ij} = \text{sign}\big[(x_i - x_j)(y_i - y_j)\big]$
- We have

$$\begin{aligned}
\sum_{i \neq j} a_{ij} b_{ij} &= \sum_{i \neq j} |x_i - x_j| \cdot |y_i - y_j| \\
&= \sum_{i=1}^{n} \sum_{j:j \neq i} (x_i y_i + x_j y_j - x_i y_j - x_j y_i) S_{ij} \\
&= \sum_{i=1}^{n} \left[x_i y_i \sum_{j:j \neq i} S_{ij} + \sum_{j:j \neq i} x_j y_j S_{ij} - x_i \sum_{j:j \neq i} y_j S_{ij} - y_i \sum_{j:j \neq i} x_j S_{ij} \right]
\end{aligned}$$

Fast Method (2)

- We need to compute for all $1 \leq i \leq n$,

$$\sum_{j:j\neq i} c_j S_{ij}$$

- For an i, we have

Partial sum

$$\sum_{j:j\neq i} c_j S_{ij}$$

$$= \sum_{j} c_j - c_i \boxed{- 2 \sum_{j:j<i}} c_j \boxed{- 2 \sum_{j:\, y_j<y_i}} c_j$$

$$\boxed{+ 4 \sum_{j:j<i, y_j<y_i}} c_j$$

Need some efforts!

Relation to Fast Kendall's τ

Recall Kendall τ coefficient

$$\tau = \frac{1}{\binom{n}{2}} \sum_{1 \le i < j \le n} \operatorname{sign}[(x_i - x_j)(y_i - y_j)] = \frac{1}{\binom{n}{2}} \sum_{1 \le i < j \le n} S_{ij}$$

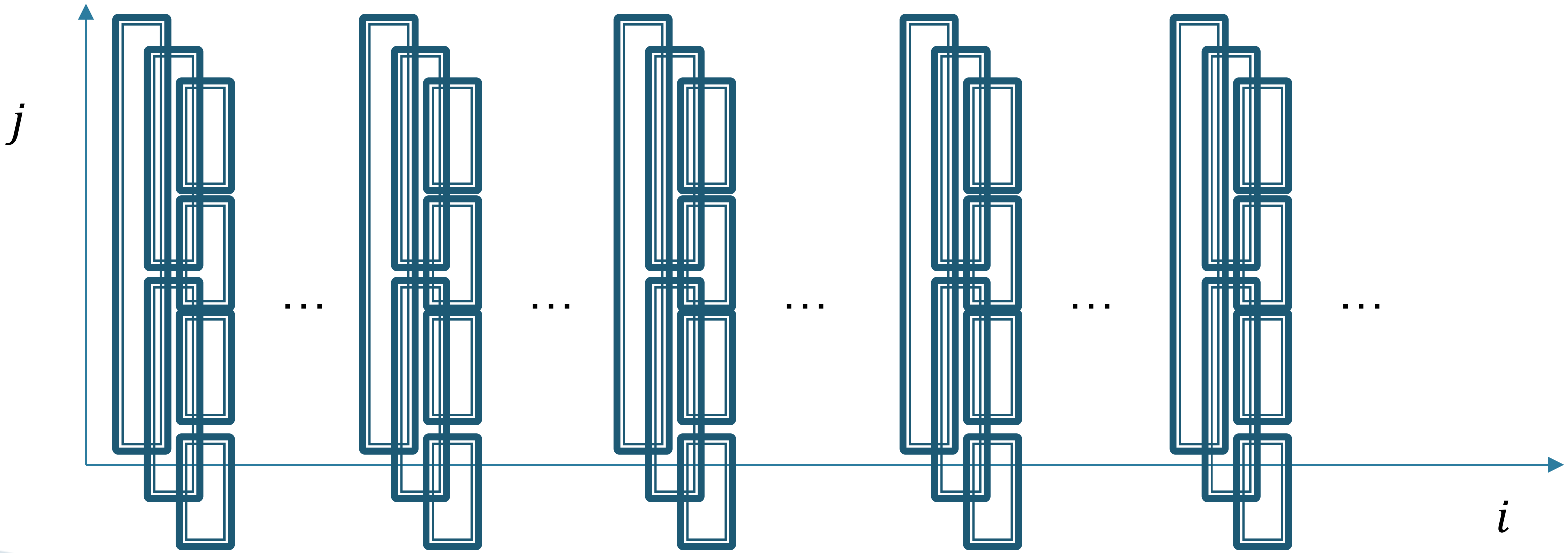
- Equivalent to $c_j \equiv 1$
- Knight (JASA 1966) and Christensen (2005)
- AVL tree structure (Adelson-Velskii and Landis 1962)

Fast Method (3)

- For an i, need to compute for

$$\sum_{j:j<i,y_j<y_i} c_j$$

- An *dyadic partitioning/updating* scheme:

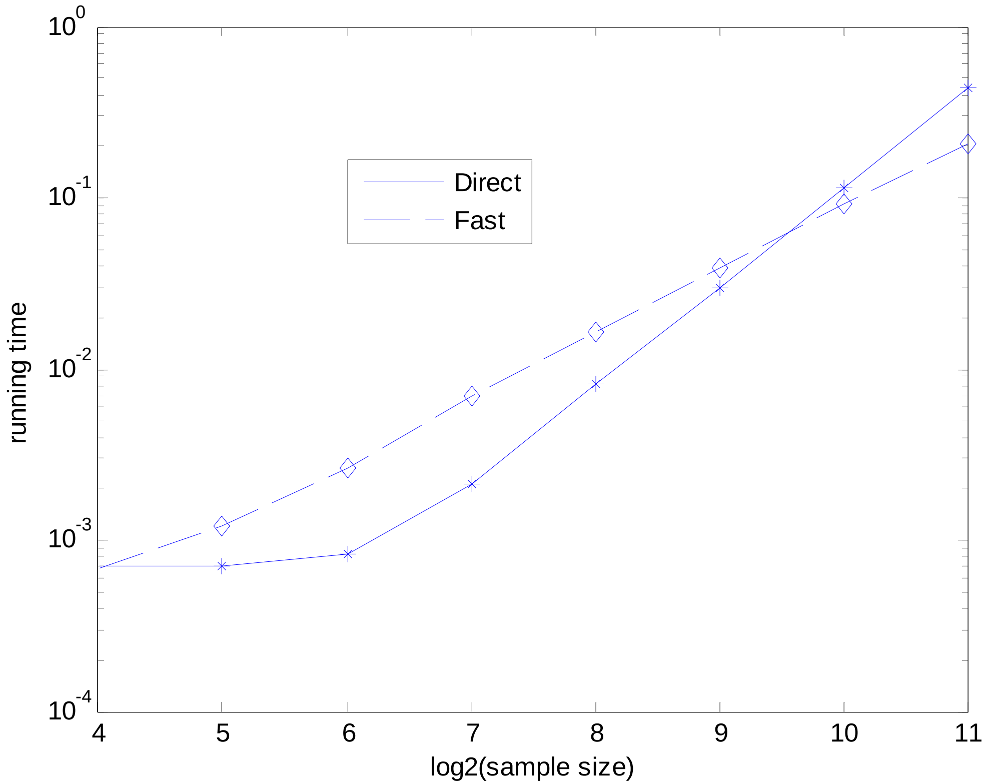


IV. Numerical Experiments

- Implementation
- Effects of large sample simulation
- Convergence

MATLAB implementation

- Compare with direct implementation

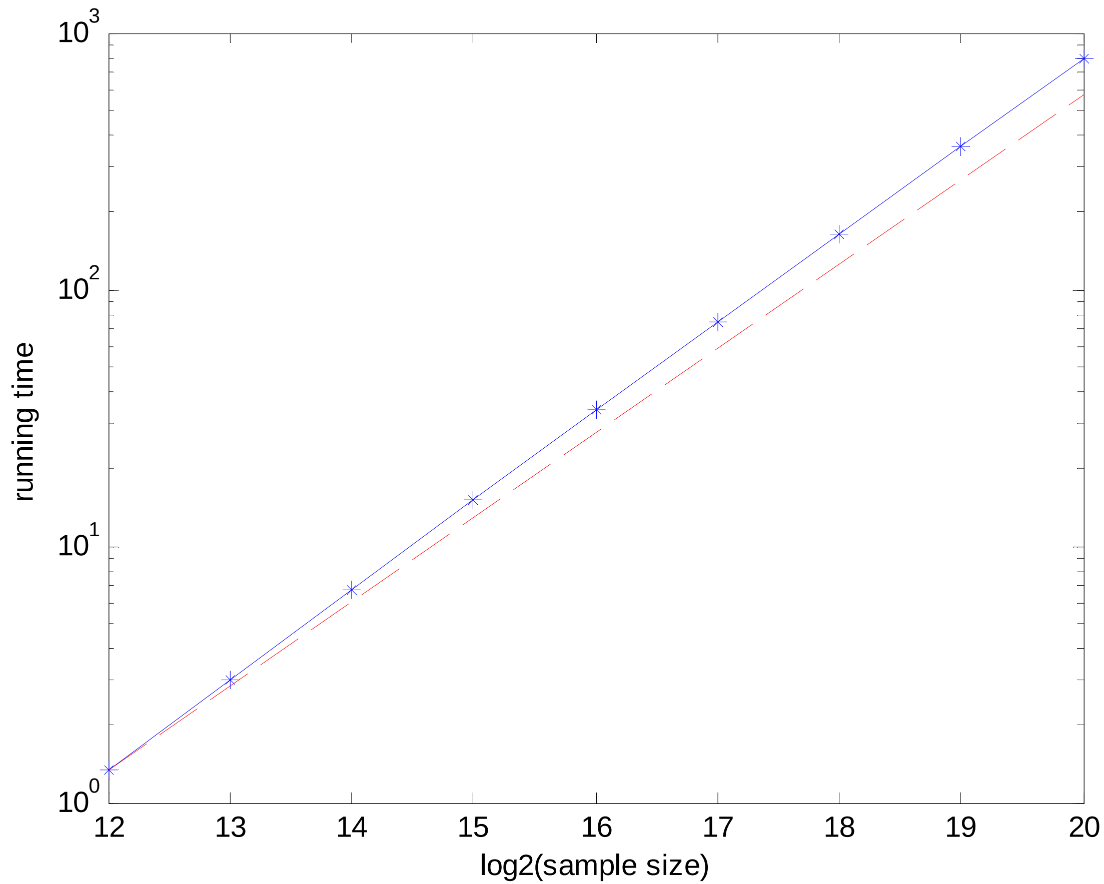


- When sample size > 2000, MATLAB is out of memory

MATLAB implementation (2)

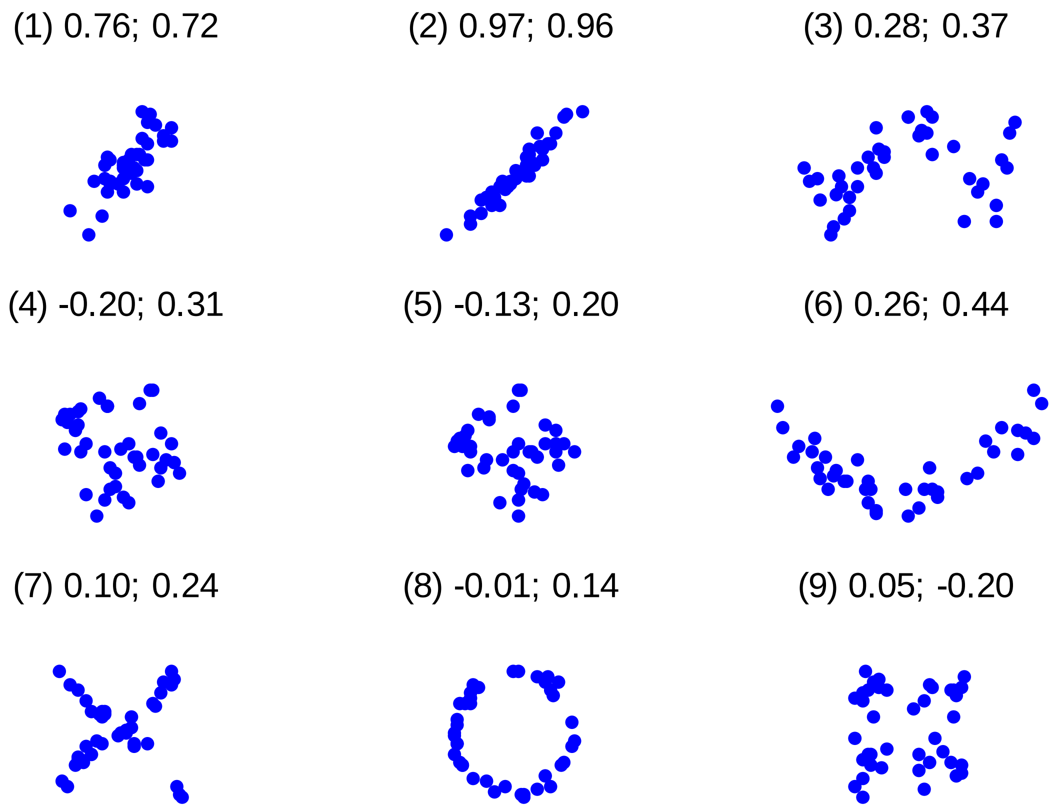
- Running time vs sample size (n)
- For n=1M, about 3min. on a laptop
- Dash line:

  O(n log n) complexity



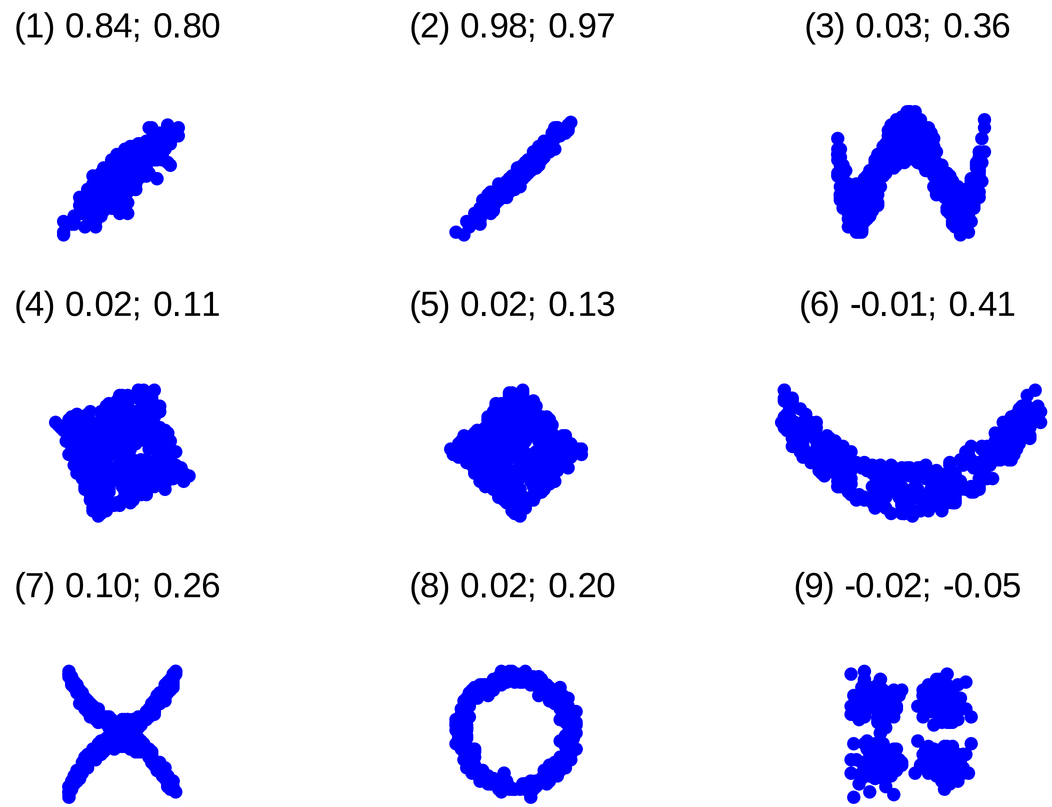
Effect of Large Sample Simulations

- Small sample (n=40)



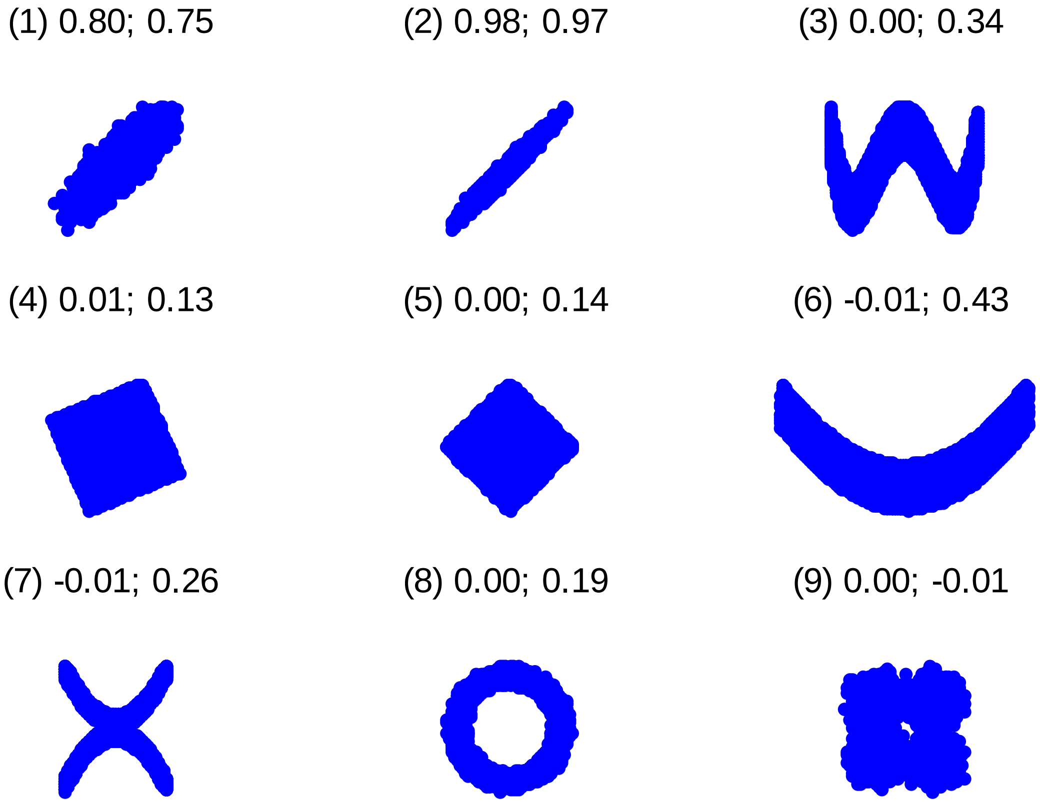
Effect of Large Sample Simulations

- Small sample (n=400)



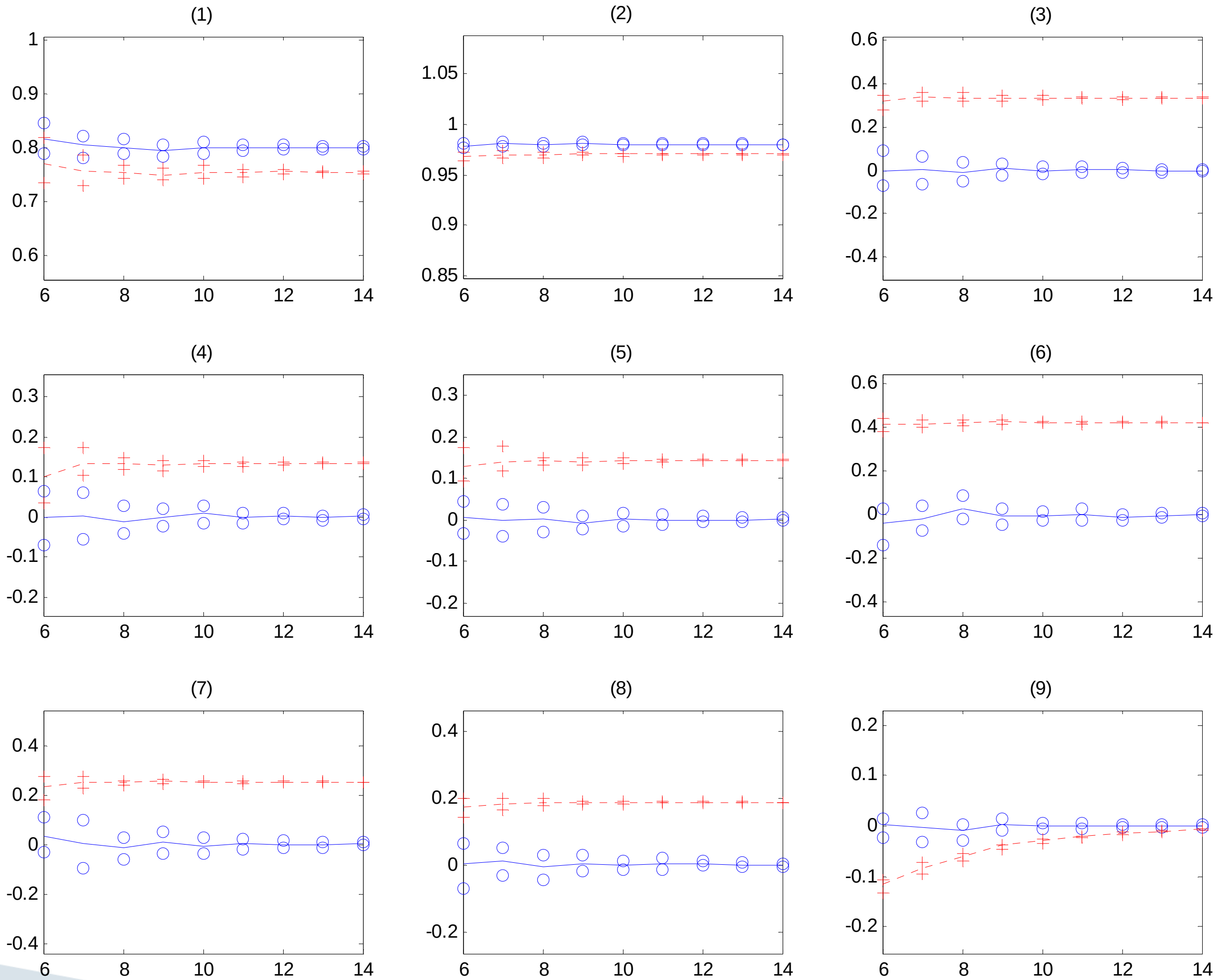
Effect of Large Sample (2)

- Large sample (n=10,000)



Convergence Study

- Convergence of sample correlations



V. Apply in Variable Screening

- Sure independence screening
- DC-SIS
- Improvement

dCor in Sure Screening

- Sure screening
- Li. et al JASA 2012, propose dCor in SURE screening
- More extensive simulation studies with sample size from n=200 to n=20,000…

SURE Variable Screening

- Setting

$$y \sim X_1, X_2, \ldots, X_p$$

- p is large; # of observation, small
- sure independence screening (SIS), Fan and Lv (JRSS-B 2008):
  - Compute marginal utility function
  - Retain the largest few
  - Asymptotic theory: “sure screening property”
  - marginal utility function: the Pearson's correlation

DC-SIS

- sure independence screening procedure based on the distance correlation
- marginal utility function: distance correlation
- Li. et al (JASA 2012)

Improvements via distCorr

- Experiment setting

$$(1.a):\ Y = c_1\beta_1 X_1 + c_2\beta_2 X_2 + c_3\beta_3 1(X_{12} < 0) + c_4\beta_4 X_{22} + \varepsilon,$$

$$(1.b):\ Y = c_1\beta_1 X_1 X_2 + c_3\beta_2 1(X_{12} < 0) + c_4\beta_3 X_{22} + \varepsilon,$$

$$(1.c):\ Y = c_1\beta_1 X_1 X_2 + c_3\beta_2 1(X_{12} < 0) X_{22} + \varepsilon,$$

$$(1.d):\ Y = c_1\beta_1 X_1 + c_2\beta_2 X_2 + c_3\beta_3 1(X_{12} < 0) + \exp(c_4|X_{22}|)\varepsilon,$$

- $p = 2000, 5000.$ $n = 200 \rightarrow 20{,}000.$

Improvements via distCorr (2)

- When $n = 200$

Table 1. The 5%, 25%, 50%, 75%, and 95% quantiles of the minimum model size $\mathcal{S}$ out of 500 replications in Example 1

$\mathcal{S}$	SIS					SIRS					DC-SIS				
Model	5%	25%	50%	75%	95%	5%	25%	50%	75%	95%	5%	25%	50%	75%	95%
Case 1: $p = 2000$ and $\sigma_{ij} = 0.5^{\lvert i-j \rvert}$															
(1.a)	4.0	4.0	5.0	7.0	21.2	4.0	4.0	5.0	7.0	45.1	4.0	4.0	4.0	6.0	18.0
(1.b)	68.0	578.5	1180.5	1634.5	1938.0	232.9	871.5	1386.0	1725.2	1942.4	5.0	9.0	24.5	73.0	345.1
(1.c)	395.9	1037.2	1438.0	1745.0	1945.1	238.5	805.0	1320.0	1697.0	1946.0	6.0	10.0	22.0	59.0	324.1
(1.d)	130.5	611.2	1166.0	1637.0	1936.5	42.0	304.2	797.0	1432.2	1846.1	4.0	5.0	9.0	41.0	336.2

- When $n = 20{,}000$

$\mathcal{S}$	SIS					SIRS					DC-SIS				
Model	5%	25%	50%	75%	95%	5%	25%	50%	75%	95%	5%	25%	50%	75%	95%
Case 1: $p = 2000$ and $\sigma_{ij} = 0.5^{\lvert i-j \rvert}$															
(1.a)	4	4	6	10	22	4	5	6	10	20	4	5	6	9	20
(1.b)	76	551	1180	1592	1918	237	814	1269	1789	1959	4	6	8	11	14
(1.c)	591	922	1364	1781	1941	342	827	1354	1637	1930	6	6	6	8	11
(1.d)	8	237	726	1310	1827	58	273	919	1444	1878	4	4	6	8	1001

Conclusion

- A fast algorithm to measure dependence: fast dCor
- Nearly linear complexity: O(n log n)
- Reference: "Fast Computing for Distance Covariance", Technometrics 2016

Name of Coeff.	Comp. cost
Pearson's ρ	n
Spearman's ρ	$n \log n$
Kendall's τ	$n \log n$
CCA	n
KCCA	n^3
ACE	n
MIC	2^n
MMD	n^2
CMMD	n^2
RDC	$n \log n$
dCor	$n^2 \rightarrow n \log n$