Dependence Discovery from Multimodal Data via Multiscale Graph Correlation

Cencheng Shen[1], Carey E. Priebe[2], Mauro Maggioni[3], and Joshua T. Vogelstein[*4]

[1]Department of Statistics, Temple University
[2]Department of Applied Mathematics and Statistics, Johns Hopkins University
[2]Department of Mathematics, Duke University
[4]Department of Biomedical Engineering and Institute for Computational Medicine, Johns Hopkins University

June 23, 2016

Abstract

Understanding and discovering dependence between multiple properties or measurements is a fundamental task not just in science, but also policy, commerce, and other domains. An ideal test for dependence would have the following properties: (1) Theoretical consistency such that the testing power converges to 1 under any dependency structure and dimensionality. (2) Strong empirical performance on a wide variety of low- and high-dimensional simulations. (3) Provides insight into the nature of the dependence, rather than merely a valid p-value. (4) On real data, detects dependence when it exists, and does not detect dependence when it does not exist. No existing test satisfies all of these properties. In this paper we propose a novel dependence test statistic called "Multiscale Graph Correlation" (Mgc), by combining the ideas of distance correlation with nearest-neighbor testing. More specifically, we only use the distance correlations amongst the nearest-neighbors of each data point, yielding a sparse, and therefore regularized, matrix from which we can compute the test statistic. We demonstrate that Mgc has all of the above properties via a series of theoretical proofs, numerical simulations, and real data experiments. Specifically, we applied Mgc in several real applications: (i) detect dependence between brain disorder and hippocampus shape, (ii) determine whether either of two pipelines can detect dependence between brain activity and personality, and (iii) do not inflate non-existent dependence between resting activity and a spurious stimulation. Mgc performs as well or better than previously proposed methods in essentially all theory, low-dimensional and high-dimensional simulations, and real data experiments. Mgc is therefore poised to be useful in a wide variety of applications, requiring only data and a dissimilarity function for both measurement types. Both MATLAB and R code are provided here: https://github.com/jovo/RankdCorr/.

Keywords: testing independence, distance correlation, k-nearest-neighbor, local correlation coefficient, permutation test

*jovo@jhu.edu

Contents

Detecting dependency among multiple data sets is one of the most important and fundamental tasks in computational statistics and data science. Indeed, prior to embarking on a predictive machine learning investigation, one might first check whether any dependence is detectable; if not, high-quality predictions will be unlikely. The founders of statistics first highlighted the importance of this task, starting with Pearson, who developed Pearson's Product-Moment Correlation statistic [1]. Since then, researchers have consistently developed new and improved methods (see [2] for a recent review and discussion).

In the era of big data, several challenges emerge as particularly prevalent and therefore, problematic. First, the dependencies between different modalities of data can be highly **non-linear**. While this has always been the case, the relative abundance of data has led to an increased demand in checking for dependence in many previously uninvestigated settings. Second, the **dimensionality** of individual samples is growing at exponential rates, with genomics and connectomics data, for example, often accruing millions or billions of dimensions per data point. At the same time, the **sample sizes** are not increasing proportionally, meaning that we often have datasets with very high-dimensions and relatively low sample size. Third, the data are often **complicated**: networks, shapes, questionnaires, semi-structured text are all typical examples. For example, we may desire to understand whether brain shape and disease status are related, so that we can develop prognostic biomarkers to combat the deleterious efforts of degenerative neurological disorders [3]. Fourth, because we will often have a data deluge, with myriad different measurements, it is important to be able to compute the results reasonably **efficiently**. Fifth, when working with big data, statistical procedures often have hyper-parameters that require tuning. Many such procedures lack any guidance in choosing the value of those hyper-parameters, thereby requiring users of the procedures to concoct their own heuristics. It is desirable that a procedure is **adaptive**, in that it can automatically set its hyper-parameters in a valid way. Finally, as alluded to above, checking for dependence is rarely the final step in the analysis. Frequently, investigators and analysts desire more than a simple p-value, rather, they desire some insight into the nature of the **dependence structure**, which can then inform them in terms of how to proceed. We desire tests that satisfy the above desiderata, both in theory as well as in extensive simulations and real data problems.

There are two key insights from the literature that we combine to develop our methodology that satisfies the above desiderata. First, a collection of pairwise comparisons suffices to characterize a joint distribution [4]. Second, nonlinear manifolds can be approximated by local linear spaces [5]. Our approach, Multiscale Graph Correlation (`Mgc`), leverages and improves upon recent developments from both subdisciplines of data science.

Interpoint pairwise comparison matrices have been used for over 100 years for various statistical purposes [4]. Perhaps one of the earliest examples comes from Karl Pearson [1], who created a special case of something subsequently called a “generalized correlation coefficient” [6]. Generalized correlation coefficients start with n pairs of observations (x_i, y_i), where x’s and y’s both might vectors, shapes, networks, etc. And then, a comparison function is defined for each. Specifically, let $a_{ij} = \delta_x(x_i, x_j)$, and let $b_{ij} = \delta_y(y_i, y_j)$. Thus, $A = \{a_{ij}\}$ and $B = \{b_{ij}\}$ are the $n \times n$ interpoint comparison matrices for $\boldsymbol{x}$ and $\boldsymbol{y}$, respectively. Without loss of generality, assuming A and B have zero mean, the generalized correlation coefficient can then be written:

$$C = \tfrac{1}{z}\textstyle\sum_{i,j=1}^{n} a_{ij} b_{ij}, \tag{1}$$

where z is proportional to standard deviations of A and B, that is $z = n^2\sigma_a\sigma_b$. In words, C is the correlation across *pairwise comparisons*, rather than the individual data samples. C has many well known special cases historically, including Pearson’s [1], Spearman’s [7], Kendall’s [6], and Mantel’s correlation [8]. Recently, Szekely et al. [9] extended these approaches, choosing δ_x to be the Euclidean distance, followed by subtracting the row means and column means, resulting in “doubly centered” distances. Impressively, they proved that this “distance correlation” (Dcorr) statistic is a consistent test for independence for any joint distribution (under suitable regularity conditions), that is, the Dcorr’s power approaches 1 as sample size approaches infinity, for any joint distribution of finite dimension and finite second moments. Szekely et al. [10] further proposed a modified version called Mcorr, which they prove to be consistent even as the dimensions of x and y increase to infinity as well. Moreover, because these distance based tests merely require a comparison function for both x and y, Lyons was able to prove that they are consistent even in other metric spaces, including certain networks, shapes, and other complicated spaces [11]. Thus, existing generalized correlation coefficient based tests therefore work well in high dimensions and low sample sizes, including in complicated domains [11], and are reasonably computationally efficient. But, empirically, they struggle in various non-linear settings, do not automatically adapt to the data, and do not provide much insight into the nature of the dependence, other than a p-value.

A deep insight that the generalized correlation coefficient tests have yet to capitalized on, that could help address the above described limitations, is that nonlinear shapes can be approximated by **locally** linear ones [5]. Locality has been utilized for classification and regression [12], data compression [13], and recommender systems [14], to name a few of the myriad data science problems for which locality has already reaped benefits. Moreover, it has become an invaluable tool in unfolding nonlinear geometry in many recent development of nonlinear embedding algorithms, dating back to the 1950s [15], and more recently making a resurgence with the advent of Isomap [16, 17], Local Linear Embedding [18, 19], and Laplacien eigenmaps [20], among many others. The concept of locality, while popular within certain fields has only entered into testing very infrequently [21–23]. These approaches, like the distance correlation based ones, have the advantage of naturally operating on complicated data, because they only require a comparison function between observations. They also have strong theoretical guarantees. However, these approaches focus on two-sample testing, rather than dependence testing.

The challenge associated with all of these methods that employ locality is in choosing the appropriate scale [24]. Even those approaches that do provide a mechanism for optimizing scale (or neighborhood size) often do so without any theoretical guarantees, and choose based on some surrogate function, rather than the exploitation task at hand. In either case, changing the neighborhood size for many of these algorithms typically requires running the entire algorithm again, rendering it computationally intractable. Thus, a gap remains in the literature: a dependence test that has all of the desirable properties of the distance based tests, but also performs well in highly nonlinear settings via adapting scale appropriately, thereby providing insight into the most informative neighborhood sizes for understanding and subsequent inference purposes.

and also profitably and efficiently employs locality while providing guidance into which scales are most informative.

Multiscale Graph Correlation

All dependence tests start from the same setting: we observe n pairs of observations $(\boldsymbol{x}_i, \boldsymbol{y}_i)$, and we desire to know whether the $\boldsymbol{x}$'s and $\boldsymbol{y}$'s are independent of one another, and if so, what is the nature of that dependence structure (Figure 1 provides an example where $\boldsymbol{x}$ and $\boldsymbol{y}$ are nonlinearly dependent).

Multiscale Graph Correlation (Mgc) combines generalized correlation coefficients with graph distances, in an effort to efficiently uncover local relationships and optimize the independence test. Specifically, let $R(a_{ij})$ be the "rank" of $\boldsymbol{x}_i$ relative to $\boldsymbol{x}_j$, that is, $R(a_{ij}) = k$ if $\boldsymbol{x}_i$ is the k^{th} closest point (or "neighbor") to $\boldsymbol{x}_j$, starting from 1 to n, and define $R(b_{ij})$ equivalently for the $\boldsymbol{y}$'s. For any neighborhood size k around each $\boldsymbol{x}$ and any neighborhood size l around each $\boldsymbol{y}$, we define the rank-truncated pairwise comparisons:

$$a_{ij}^k = \begin{cases} a_{ij} - \bar{a}^k, & \text{if } R(a_{ij}) \leq k, \\ 0, & \text{otherwise}; \end{cases} \qquad b_{ij}^l = \begin{cases} b_{ij} - \bar{b}^l, & \text{if } R(b_{ij}) \leq l, \\ 0, & \text{otherwise}; \end{cases} \tag{2}$$

where $\bar{a}^k$ and $\bar{b}^l$ are two mean-adjusting scalars such that $\sum_{i,j=1}^n a_{ij}^k = \sum_{i,j=1}^n b_{ij}^l = 0$. Then we can define a *local* variant of any global generalized correlation coefficient, by excluding large distances:

$$C^{kl} = \frac{1}{z_{kl}} \sum_{i,j=1}^n a_{ij}^k b_{ij}^l, \tag{3}$$

where $z_{kl} = n^2 \sigma_a^k \sigma_b^l$, with σ_a^k and σ_b^l being the standard deviations for the truncated pairwise comparisons. There are a maximum of n^2 different local correlations, one for each possible combinations of k and l (more technical details of Mgc are in Appendix B.4). Among all n^2 local statistics, $\{C^{kl}\}$, Mgc selects the best local statistic for testing.

Having defined how to compute Mgc, we face three challenges to make the method practical. First, in addition to the test statistic, we need to compute the null distribution, so that we may find the critical values and p-values. Second, naïvely, computing all local C^{kl} statistics would require an unacceptably large computational budget. Third, having computed all local statistics, we require a method for choosing the optimal neighborhood size, in such a way that the test is still consistent, and not biased (meaning that the resultant p-value is valid).

Computing the p-values from the test statistic is actually straightforward, thanks to the advent of permutation testing [25]. Specifically, we can permute the labels of either the $\boldsymbol{x}_i$'s or the $\boldsymbol{y}_i$'s, and then compute the Mgc statistics on the permuted data. By permuting the labels, we have rendered the two different views of the data independent. Doing so many times yields an empirical estimate of the null distribution, which we can use to compute the critical value and p-value. This procedure is somewhat time consuming, which makes computing the test statistics for all neighborhoods efficiently even more important.

Nearly all algorithms that employ *some* kind of regularization face a similar dilemma: how to efficiently choose the hyper-parameters.

Most manifold learning algorithms require that the user essentially runs the entire algorithm again from scratch for each different hyper-parameter setting, a pursuit that can be exponentially taxing as the number of hyper-parameters increases. In our case, once the rank information is provided, each distance-based local correlation takes $O(n^2)$ to compute (Pseudocode 1 in Appendix C.1), which means a straightforward algorithm to compute all local correlations would take $O(n^4)$.

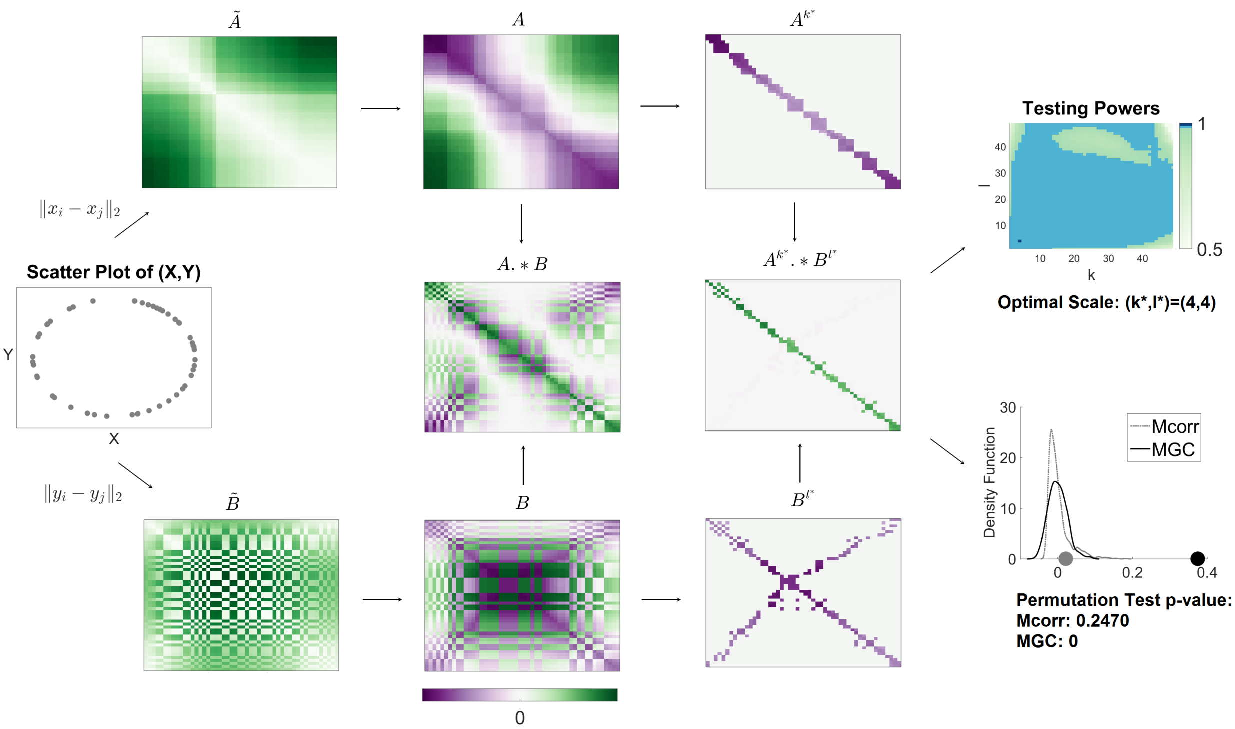

Figure 1: Flowchart for Mgc computation: Column 1: (X, Y) have a circle relationship. Column 2: The heat maps of $\tilde{A}$ and $\tilde{B}$, which are the pairwise Euclidean distance matrices of X and Y. All distance entries are non-negative. Column 3: The top and bottom panels are the heat maps of $A = \{a_{ij}\}$ and $B = \{b_{ij}\}$, which are the properly centered distance matrices of $\tilde{A}$ and $\tilde{B}$. The center panel is the heatmap of the entry-wise products of A and B, summing over which yields the un-normalized Mcorr statistic. As the entries of A and B can be either positive or negative, the entry-wise products can be either positive or negative for nonlinear dependencies, which causes $\texttt{Mcorr}(X, Y)$ to be close to 0 and the p-value to be in-significant, as shown in column 5. Column 4: The top and bottom panels are the heat maps of local A and B, i.e., $A^{k^*} = \{a_{ij}^{k^*}\}$ and $B^{l^*} = \{b_{ij}^{l^*}\}$, where $(k^*, l^*) = (4, 4)$ is the optimal scale for the circle relationship. The center panel is the heat map of the entry-wise products of local A and B, summing over which yields the un-normalized Mgc statistic C^*. Mgc successfully identifies the optimal local structure for correlation testing, and the resulting entry-wise products are dominantly non-negative, which causes $\texttt{Mgc}(X, Y)$ to be much larger than 0 and the p-value to be significant, as shown in column 5. Column 5: The top panel is the testing powers of all local correlations, where the optimal scale is shown as a dark blue point with many adjacent scales being very close to optimal (light blue points). The bottom panel shows $\texttt{Mgc}(X, Y)$ and $\texttt{Mcorr}(X, Y)$ as dark and gray dots on the x-axis, as well as the distribution of the permuted test statistics.

However, we have devised an algorithm for exactly computing *all* local correlations in $\mathcal{O}(n^2 \log n)$, essentially the same running time complexity as the global correlation coefficient (the additional log factor is for sorting to find the neighbors, see Pseudocode 2 in Appendix C.1 for details). We do so by noting that the sufficient statistics for larger neighborhood sizes include those for the smaller sizes, so we can simply keep track of them as we iteratively increase neighborhood size. The end result is `Mgc` can be computed in comparable time as the other leading dependence tests.

Therefore, we can efficiently compute all local correlations for a given pair of data and the permuted data, which yields the p-values for each neighborhood size (see Pseudocode 3 in Appendix C.1 for details). But it does not tell us which neighborhood sizes are optimal.

Our procedure for estimating the optimal scale searches for regions of neighborhood sizes for which p-values are consistently low, guarding against noisy scales that appear optimal, and combating bias added by looking at many different scales. We assert that the optimal scale is the largest neighborhood size in that region. We define the p-value of `Mgc` to be the p-value from the optimal scale, and declare significant dependency when the p-value is less than α, often 0.05 (see Pseudocode 4 and 5 in Appendix C.1 for details).

Theoretical Consistency of `Mgc`

The formal testing scenario is as follows: we observe n pairs of observations, $(\boldsymbol{x}_i, \boldsymbol{y}_i)$, and we desire to know whether the $\boldsymbol{x}$'s are independent of the $\boldsymbol{y}$'s. To cast this problem as a statistical inference query requires specifying a statistical model, that is, a collection of possible distributions from which we may assume the data arise. To make the investigation as general as possible, we consider the largest possible set of distributions: any possible joint distribution f_{xy}. If $\boldsymbol{x}$ and $\boldsymbol{y}$ were independent, then it would follow that $f_{xy} = f_x f_y$; in other words, for independent data, the joint distribution is equal to the product of the marginals. Therefore, we have the following hypothesis testing scenario:

$$H_0 : f_{xy} = f_x f_y,$$
$$H_A : f_{xy} \neq f_x f_y.$$

The power of a test is defined as the probability that it correctly rejects the null when the null is indeed false. As defined above, a test is consistent if its power converges to 1 as sample size increases. Let C_t denote a global generalized correlation coefficient based test, that is, t might indicate Pearson, MANTEL, DCORR, or MCORR, and let $\beta(C_t^*)$ denote the power of the corresponding multiscale version. Recalling from the work Szekeley et al. that DCORR and MCORR are both consistent tests. More specifically, DCORR is consistent whenever f_{xy} has finite dimension and bounded variance, and MCORR is consistent even as dimension increases to infinity. Denote the set of distributions satisfying consistency for a given test by $\mathcal{F}_t$, where t indicates which test we are referring to. Then, we have the following theorem:

Theorem 1. $\beta(C_t^*) \to 1$ *for all* f_{xy} *in* $\mathcal{F}_t$.

Therefore, MGC is consistent against all dependent alternatives for which its global counterpart is. However, asymptotic consistency does not convey to us how quickly MGC achieves optimal power in various settings, and whether it exhibits significant advantage over its global counterpart and other popular methods. For that, we turn to numerical simulations.

Finite Sample Simulation Experiments

From this section onwards, unless mentioned otherwise, our MGC is always implemented for MCORR, due to its theoretical consistency and numerical advantages throughout.

Based on the previous section, we understand MGC can be a consistent tests in a wide variety of settings (all finite dimensional joint distributions with bounded variance) But, our theoretical results do not shed light on the finite sample performance of MGC. Specifically, we are interested in comparing our newly proposed local tests in a comprehensive set of simulations, to previous tests like HHG, MCORR, DCORR, and MANTEL, each of which performs well in a fraction but not all of the simulations.

To do so, we consider 20 different joint distributions f_{xy}. A large fraction of these are taken exactly from existing literature [9, 26–28], and we have added several additional settings. They include: linear and nearly linear (1-5), polynomial (6-12), trigonometric (13-17), uncorrelated but nonlinearly dependent (18-19), and an independent relationship (20). Details for each setting are given in Appendix A, with a visualization of each dependency shown in Supplementary Figure A1.

Figure 2 shows the testing powers of MGC, MCORR, and HHG versus the dimensionality of x (see Methods for details), with the sample sizes fixed at $n = 100$ for each simulation. Note that the dimensionality of y increases in only a subset of the settings.

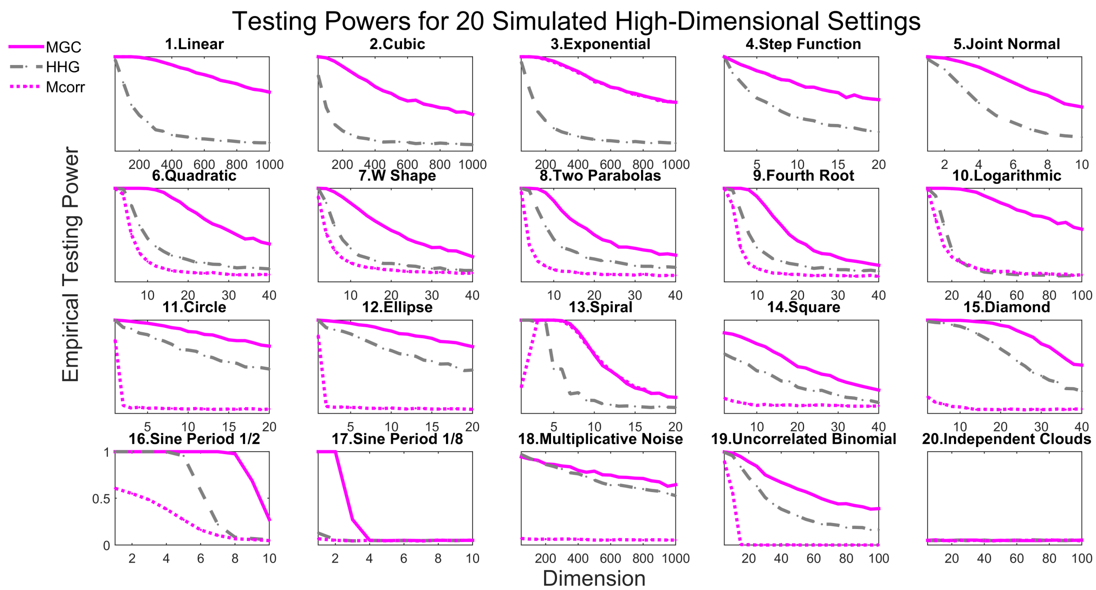

Figure 2: Powers of different methods for 20 different dependence structures, estimated by the empirical distributions of the test statistics under the null and the alternative on the basis of $10{,}000$ Monte-Carlo replicates. $2{,}000$ additional MC replicates are used for optimal scale estimation for Mgc. Each panel shows empirical testing power on the absicca at a significant level $\alpha = 0.05$, and the dimensionality on the ordinate. Mgc empirically achieves similar or better power than the previous state of the art approaches for all sample sizes on all problems.

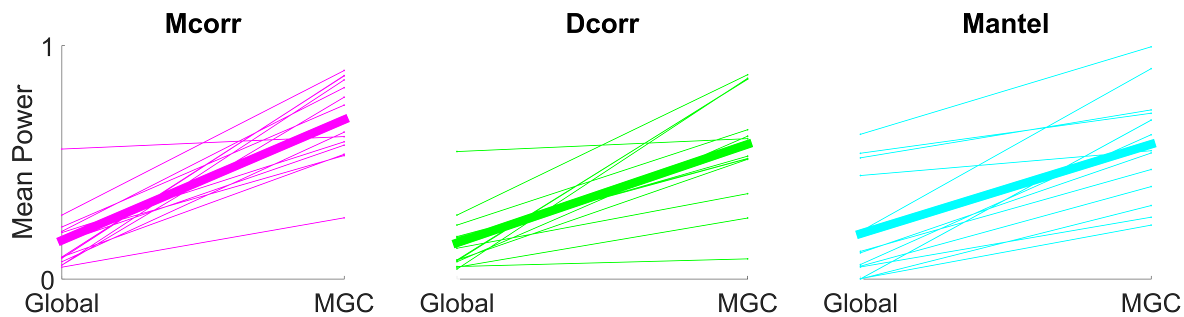

Figure 3: Average powers slopegraphs comparing global and Mgc tests. For each global test, the left side corresponds to the mean power of each simulation in Figure 2, the right side corresponds to the respective Mgc mean power. The thin solid line are shown for 6-19, because Mgc equals the global correlation for 1-5 and 20. Then the thick solid line summarizes how the overall mean power (including 6-19) changes from global to Mgc. It is clear that Mgcalways significantly improves over its global counterpart.

The advantage of Mgc over its global counterpart Mcorr and Hhg is stark. For the nearly linear settings, Mgc and Mcorr are essentially identical and significantly better than Hhg as the dimension increases. For the remaining nonlinear dependencies, Mgc achieves superior power than Hhg and Mcorr for *all* functions, often by a significant margin. For the independent simulation, all tests yield powers at the significance level α, indicating no more false positives than expected according to the theory.

Mgc **dominates Global Counterparts**

To better summarize the advantages of global versus local, Figure 3 shows how the powers change from each global correlation to its Mgc implementation, for each dependency in Figure 2. Indeed, Mgc always improves over its global correlation, regardless of the global correlation that is being used. Note that the actual powers for Dcorr, MANTEL , and their Mgc variants are included in Appendix A, where the same conclusion for Mgc superiority still hold. We also present in Appendix A an additional simulation setting with increasing sample size and fixed dimensionality to observe that the powers of Mgc converge to 1 faster than all the benchmarks for nearly all dependencies.

Discovery of Dependency Across Scales

A multiscale power map is a heatmap of powers for all neighborhood sizes, for a given joint distribution and sample size. Figure 4 provides the multiscale power maps for all 20 different scenarios for a specified dimensionality (see caption for details), illustrating how the powers of local correlations change with respect to increasing neighborhood sizes.

The multiscale power map sheds light into the intrinsic dependency structure. For nearly linear dependencies (1-5), the best neighborhood choice is always the largest scale, i.e., $k = l = n$. For all strongly nonlinear dependencies (6-19), Mgc almost always chooses a smaller scale in a distribution dependent fashion. Furthermore, similar dependencies have similar local correlation structure, and thus similar optimal scales. For example, quadratic (6) and W (7) are both polynomials of degree 2 with different coefficients, and their power maps are quite similar to each other. Similarly, (16) and (17) are the same trigonometry function (sine) with different periods, and they share a narrow range of significant local correlations. Both circle (11) and eclipse (12), as well as square (14) and diamond (15), are closely related functions, and have similar multiscale power maps. Note that for almost all simulations, there exist a large portion of adjacent local neighborhoods that are equally significant, which is an important observation that we use to approximate the optimal Mgc scale for real data.

The above described qualitative descriptions led us to believe the following two conjectures. First, for linear dependencies, the optimal Mgc scale is the global one. Second, under certain nonlinear dependencies, Mgc can achieve a better finite-sample testing power than its corresponding global correlation. Indeed, we were able to prove both of these claims:

Theorem 2. *If x is linearly dependent on y, then for any n it always holds that*

$$\beta(C^{nn}) = \beta(C^{*}) = \beta(C). \tag{4}$$

Thus the optimal scale for Mgc *is the global scale for linearly dependent data.*

On the other hand, for finite sample nonlinear dependencies (which better characterize all real data) we have the following theorem.

Theorem 3. *There exists f_{xy} and n such that*

$$\beta(C^{*}) > \beta(C). \tag{5}$$

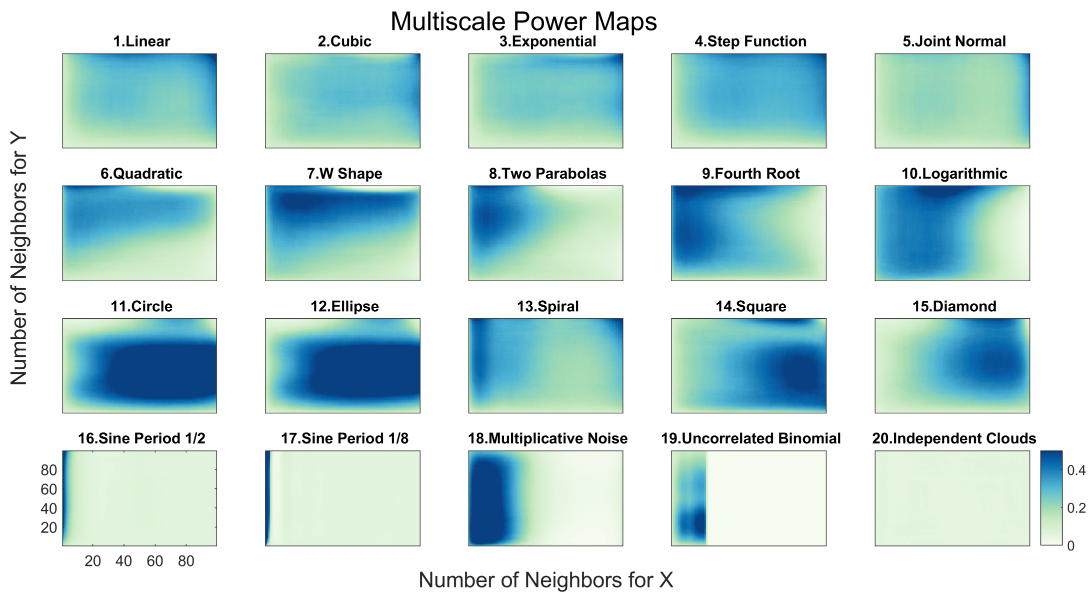

Figure 4: Influence of neighborhood size on testing power of local correlations at $\alpha = 0.05$. For each of the 20 panels, the abscissa denotes the number of neighbors for X (the scale increases from left to right), and the ordinate denotes the number of neighbors for Y (the scale increases from bottom to top). For each simulation, the sample size is $n = 100$, and the dimension is determined by the largest dimension for `Mgc` to have powers exceeding the threshold 0.5. Each different simulation yields a different surface, highlighting the importance of understanding local scale in terms of understanding the data.

Thus multiscale graph correlation can be better than its global correlation coefficient under certain nonlinear dependency, for finite sample.

Note that Theorem 2 and Theorem 3 hold for any of MGC varieties, including DCORR, MCORR, and MANTEL. The proofs of Theorem 2 and 3 are both in Appendix D. The proof of Theorem 2 is straightforward. The proof of Theorem 3 is a constructive one. More specifically, we constructed quadratic function and sampled data a finite number of times and exactly compute the power for both MGC and DCORR, proving that MGC has higher power in this setting. This shows that MGC can outperform its global counterpart even for the most modest nonlinear functions. Because any function can be approximated by a polynomial expansion [29], the proof of Theorem 3 suggests that MGC is able to outperform its corresponding global correlation on a wide variety of nonlinear functions, which is indeed the case throughout the numerical simulations.

Real Data Experiments

Only Local Scales can Detect Dependence

Our first real data experiment investigates whether brain shape and disease status are indeed dependent on one another. Previous investigations have linked major depressive disorder to the hippocampus shape [3, 30], though global tests were unable to detect a statistically significant dependence structure at the $\alpha = 0.05$ level.

This brain shape versus disease dataset consists of $n = 114$ subjects, for each we have an MRI scans as well as a categorical variable indicating whether the subject is clinically depressed, high-risk, or non-affected. From the MRI data, previous work we extracted both the left and right hippocampi. For the brain shape "view" of the data, we compute the interpoint comparison matrices using a nonlinear landmark matching approach [3, 31]. For the categorical disorder variable, we use squared Euclidean distance, then add 1 to every non-diagonal entry (so only the diagonals are of distance 0).

We consider two dependence tests, one for each hemisphere: is hippomcapus shape independent of depressive state. Figure 5A provides the p-value curves for MGC for $k = 2, \ldots, n$ at $l = 4$ (we only show $l = 4$ because the other curves look similar). Many local scales yield significant p-values (around 0.01) for both hemispheres, whereas the global scale does not detect a significant dependence in either hemisphere. None of the previously proposed dependence tests under

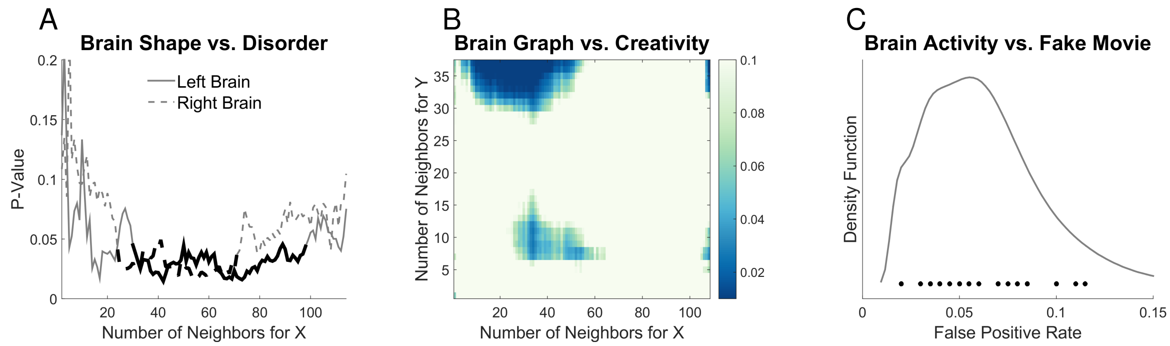

Figure 5: (A) Local correlation p-value curves with respect to $k = 2, \ldots, 114$ at $l = 4$ for brain vs disease. Dark lines correspond to the largest region of significant scales. (B) Local correlation p-value heat map with respect to $k = 2, \ldots, 109$ and $l = 2, \ldots, 38$ for brain Migraine vs CCI. (C) Density estimate for the false positive rates of Mgc on the brain vs noise experiments, with the actual rate of each data shown as dots above the x-axis.

consideration (Mantel, Dcorr, Mcorr, or Hhg)were able to detect dependence for both (not shown).

Mgc can provide insight into the nature of the dependence structure

The next real data experiment investigates whether brain networks and personalities are independent of one another. Previous work [32] investigated whether individual voxels were related to specific dimensions of personality, but were unable to compare entire brain networks to a higher-dimensional characterization of personality. Figure 5B shows that global dependence tests can ascertain whether the whole brain-network is independent of the five-factor personality traits [33]. However, the global test is quite fragile, even ignoring a single subject from the global test can render the test non-significant. On the other hand, Mgc is more robust, there is a whole region of neighborhood sizes such that the test is quite significant. Moreover, that the local tests performs optimally with approximately 30 neighbors suggests that these data have multiple cohorts, for which the dependence structure likely differs. This result therefore suggests the next investigatory steps to take to further understand the nature of the dependence structure between brain networks and personality.

MGC Does Not Inflate False Positive Rates

In the last experiment, MGC is applied to test independence between brain voxel activities and non-existent stimulus similar to [34], by using 26 resting state fMRI data sets from the 1000 functional connectomes project (http://fcon_1000.projects.nitrc.org/). We used CPAC [35] to estimate regional time-series, in particular, using the sequence of pre-processing decisions determined to optimize discriminability [36]. The output for each dataset is the resting state fMRI time-series data containing 200 regions of interest for 200 time-steps. We then also generate an independent stimulus by sampling from a standard normal at each time step. Of course, the brain activity data and the stimuli are independent by construction. For each brain region, we test: is activity of that brain region independent of the time-varying stimuli. We pool brain activity over all of the samples from the population. Any regions that are detected significant are false positives by definition. By testing reach brain region separately, we obtain a distribution of false positive rates. If our test is unbiased, that distribution should be centered around the critical level, which we set at 0.05 for this experiment.

To conduct this test, we must construct a distance matrix for brain region activity, and another for the stimulus. For each brain region, we compute $a_{ij} = \|\boldsymbol{x}_{\cdot i} - \boldsymbol{x}_{\cdot j}\|_2^2$, for all (i, j) pairs, where $\boldsymbol{x}_{\cdot i}$ denotes the observation vector of all subjects at time-step i. For the stimulus, we similarly compute the Euclidean distance between activity at all pairs of time-steps: $b_{ij} = \|y_i - y_j\|_2^2$. Note that the distance matrices at different brain regions are distinct, but the stimulus is the same for all brain regions during the same experiment.

For each data set, the above test is carried out for each brain region, and the false positive rates of MGC for each dataset are shown in Figure 5C. MGC false positive rate is centered around the critical level 0.05, as it should be. In contrast, standard methods for fMRI analysis, such as generalized linear models, significantly increase or decrease the false discovery rates, depending on the data [34, 37].

Discussion

We propose multiscale graph correlation to test independence between measurement types. We demonstrate via simulations that MGC empirically performs well in linear and non-linear settings, regardless of the dimension, sample size, and noise. Moreover, it efficiently adapts to the data, to provide not just a valid p-value, but also a picture of which scales contain the dependence structure. We then prove that it achieves optimal power asymptotically no matter what the dependence structure is, even in complicated settings. In real data experiments it revealed dependence where global methods failed, revealed the locality of dependence where global methods succeeded, and did not falsely detect signals when there were none.

A method closely related to distance correlation tests arises from the machine learning community: kernel-based independence test [38–40]. Recent work has demonstrated the equivalence between these kernel tests and the energy statistics work [41, 42]. Thus, we may be able to glean further insights by casting Mgc within the kernel framework. Two other tests merit particular mention at this point. First, Dumcke et al [43] recently proposed a related nearest-neighbor based test. Unfortunately, their proposed test requires estimating relative high-dimensional densities, and therefore, does not perform particularly well, nor does it have strong theoretical support. Finally, Reshef et al [44] is another competing methodology, but does not perform as well as energy based tests in various benchmarks [26], and their actual test is an approximation with unknown error bound relative to their theoretical claims.

Although our definition of local correlation coefficient is fast to implement, generally applicable to any global correlation, and achieves good testing powers, there are multiple ways to combine neighborhood information into a particular global correlation coefficient. So it is possible that the testing performance may be further improved, by tailoring a different centering or ranking scheme for a given global correlation, or by coming up with a different rank-truncated pairwise comparison. Overall, a more thorough investigation on the finite-sample performance of Mgc, its possible extensions, and other existing methods, are much needed in the future to enhance our understanding of dependence discovery.

Furthermore, the optimal scale for Mgc is also of interest, such as how to more accurately select the local scale under unknown models for a particular inference task, and the implication of the optimal scale on the geometry of underlying dependency, etc. Another direction we are investigating is how to choose the optimal metric for given data. Beyond the dependence testing framework, it may also be promising to pursue the applications of Mgc and local correlations in other closely-related subjects, such as dimension reduction, classification, other testing and prediction domains, etc.

References

[1] K. Pearson, *Proceedings of the Royal Society of London* **58**, 240 (1895). 3, 4

[2] M. Reimherr, D. Nicolae, *Statistical Science* **28**, 116 (2013). 3

[3] Y. Park, C. Priebe, M. Miller, N. Mohan, K. Botteron, *Journal of Biomedicine and Biotechnology* p. 694297 (2008). 3, 14

[4] J. Maa, D. Pearl, R. Bartoszynski, *Annals of Statistics* **24**, 1069 (1996). 3, 4

[5] W. K. Allard, G. Chen, M. Maggioni, *Applied and Computational Harmonic Analysis* **32**, 435 (2012). 3, 4

[6] M. G. Kendall, *Rank Correlation Methods* (London: Griffin, 1970). 4

[7] C. Spearman, *The American Journal of Psychology* **15**, 72 (1904). 4

[8] N. Mantel, *Cancer Research* **27**, 209 (1967). 4, 27

[9] G. Szekely, M. Rizzo, N. Bakirov, *Annals of Statistics* **35**, 2769 (2007). 4, 9, 20, 28, 39

[10] G. Szekely, M. Rizzo, *Journal of Multivariate Analysis* **117**, 193 (2013). 4, 29, 39

[11] R. Lyons, *Annals of Probability* **41**, 3284 (2013). 4, 28

[12] C. Stone, *Annals of Statistics* **4**, 595 (1977). 4

[13] I. Daubechies, *Ten lectures on wavelets* (SIAM, 1992). 4

[14] B. Sarwar, G. Karypis, J. Konstan, J. Riedl, *ACM WebKDD 2000 Workshop* (2000). 4

[15] W. Torgerson, *Multidimensional Scaling: I. Theory and method* (Psychometrika, 1952). 5

[16] J. B. Tenenbaum, V. de Silva, J. C. Langford, *Science* **290**, 2319 (2000). 5

[17] V. de Silva, J. B. Tenenbaum, *Advances in Neural Informaiton Processing Systems* **15**, 721 (2003). 5

[18] L. K. Saul, S. T. Roweis, *Science* **290**, 2323 (2000). 5

[19] S. T. Roweis, L. K. Saul, *Journal of Machine Learning Research* **4**, 119 (2003). 5

[20] M. Belkin, P. Niyogi, *Neural Computation* **15**, 1373 (2003). 5

[21] D. Barton, F. David, *Research Papers in Statistics, Wiley, New York* (1966). 5

[22] J. Friedman, L. Rafsky, *Annals of Statistics* **11**, 377 (1983).

[23] M. Schilling, *Journal of the American Statistical Association* **81**, 799 (1986). 5

[24] C. Shen, J. T. Vogelstein, C. Priebe, *Submitted* (2016). 5

[25] P. Good, *Permutation, Parametric, and Bootstrap Tests of Hypotheses* (Springer, 2005). 6

[26] N. Simon, R. Tibshirani, *available at http://arxiv.org/abs/1401.7645* (2012). 9, 17, 20

[27] M. Gorfine, R. Heller, Y. Heller, *available at http://ie.technion.ac.il/ gorfinm/files/science6.pdf* (2012). 20, 30

[28] R. Heller, Y. Heller, M. Gorfine, *Biometrika* **100**, 503 (2013). 9, 30

[29] W. Rudin, *Real and Complex Analysis* (McGraw-Hill Education, 1986), third edn. 14

[30] J. Posener, *et al.*, *American Journal of Psychiatry* **160**, 83 (2003). 14

[31] M. Beg, M. Miller, A. Trouv, L. Younes, *International journal of computer vision* **61**, 139 (2005). 14

[32] J. Adelstein, *et al.*, *PLoS ONE* **6**, e27633 (2011). 15

[33] R. R. Costa, & McCrae, *Neo PI-R professional manual*, vol. 396 (1992). 15

[34] A. Eklund, M. Andersson, C. Josephson, M. Johannesson, H. Knutsson, *NeuroImage* **61**, 565 (2012). 16

[35] C. Craddock, *et al.*, *Frontiers in Neuroinformatics* **42** (2015). 16

[36] S. Wang, C. E. Priebe, M. Maggioni, J. T. Vogelstein, *in preparation* (2016). 16

[37] A. Eklund, T. Nichols, H. Knutsson, *arXiv* (2015). 16

[38] A. Gretton, R. Herbrich, A. Smola, O. Bousquet, B. Scholkopf, *Journal of Machine Learning Research* **6**, 2075 (2005). 17

[39] A. Gretton, L. Gyorfi, *Journal of Machine Learning Research* **11**, 1391 (2010).

[40] A. Gretton, K. Borgwardt, M. Rasch, B. Scholkopf, A. Smola, *Journal of Machine Learning Research* **13**, 723 (2012). 17

[41] D. Sejdinovic, B. Sriperumbudur, A. Gretton, K. Fukumizu, *Annals of Statistics* **41**, 2263 (2013). 17

[42] A. Ramdas, S. J. Reddi, B. Pczos, A. Singh, L. Wasserman, *29th AAAI Conference on Artificial Intelligence* (2015). 17

[43] S. Dmcke, U. Mansmann, A. Tresch, *PLOS ONE* **9**, e107955 (2014). 17

[44] D. Reshef, *et al.*, *Science* **334**, 1518 (2011). 17

Acknowledgment

This work was partially supported by National Security Science and Engineering Faculty Fellowship (NSSEFF), Johns Hopkins University Human Language Technology Center of Excellence (JHU HLT COE), Defense Advanced Research Projects Agency's (DARPA) SIMPLEX program through SPAWAR contract N66001-15-C-4041, and the XDATA program of the Defense Advanced Research Projects Agency (DARPA) administered through Air Force Research Laboratory contract FA8750-12-2-0303. The authors thank Dr. Brett Mensh of Optimize Science for acting as our intellectual consigliere.

A Simulation Functions

We list the distributions of the 20 dependencies used in the simulations, which are based on a combination of the simulations used in [9, 26, 26, 27] but with some changes (such as the inclusion of additional noise and an extra weight vector) to better compare all methods throughout different dimensions and sample sizes.

For each sample $\boldsymbol{x} \in \mathbb{R}^{d_x}$, we denote $\boldsymbol{x}^d, d = 1, \ldots, d_x$ as the dth dimension of $\boldsymbol{x}$. For the purpose of high-dimensional simulations, $w \in \mathbb{R}^{d_x}$ is a decaying vector with $w^d = 1/d$ for each d, such that $w^{\mathsf{T}}\boldsymbol{x}$ is a 1-dimensional weighted summation of all dimensions of $\boldsymbol{x}$, which equals $\boldsymbol{x}$ if $d_x = 1$. Furthermore, $\mathcal{U}$ denotes the uniform distribution, $\mathcal{B}$ denotes the Bernoulli distribution, $\mathcal{N}$ denotes the normal distribution, u and v represent realizations from some auxiliary random variables, c is a scalar constant to control the noise level (which equals 1 for 1-dimensional simulations and 0

otherwise), and ϵ is sampled from an independent standard normal distribution unless mentioned otherwise.

For all of the below equations, $(\boldsymbol{x}, \boldsymbol{y}) \overset{iid}{\sim} f_{xy} = f_{y|x} f_x$. For each setting, we provide the space of $(\boldsymbol{x}, \boldsymbol{y})$, and define each of the above distributions, and any additional auxiliary distributions.

1. Linear $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$,

$$\boldsymbol{x} \sim \mathcal{U}(-1, 1)^{d_x},$$
$$\boldsymbol{y} = w^\mathsf{T} \boldsymbol{x} + c\epsilon.$$

2. Cubic $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$:

$$\boldsymbol{x} \sim \mathcal{U}(-1, 1)^{d_x},$$
$$\boldsymbol{y} = 128(w^\mathsf{T} \boldsymbol{x} - \tfrac{1}{3})^3 + 48(w^\mathsf{T} \boldsymbol{x} - \tfrac{1}{3})^2 - 12(w^\mathsf{T} \boldsymbol{x} - \tfrac{1}{3}) + 80c\epsilon.$$

3. Exponential $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$:

$$\boldsymbol{x} \sim \mathcal{U}(0, 3)^{d_x},$$
$$\boldsymbol{y} = exp(w^\mathsf{T} \boldsymbol{x}) + 10c\epsilon.$$

4. Step Function $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$:

$$\boldsymbol{x} \sim \mathcal{U}(-1, 1)^{d_x},$$
$$\boldsymbol{y} = \boldsymbol{I}(w^\mathsf{T} \boldsymbol{x} > 0) + \epsilon,$$

   where $\boldsymbol{I}$ is the indicator function.

5. Joint normal $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}^{d_x}$: Let $\rho = 1/2d_x$, I_{d_x} be the identity matrix of size $d_x \times d_x$, J_{d_x} be the matrix of ones of size $d_x \times d_x$, and $\Sigma = \begin{bmatrix} I_{d_x} & \rho J_{d_x} \\ \rho J_{d_x} & I_{d_x} \end{bmatrix}$. Then let $(u, v) \sim \mathcal{N}(0, \Sigma)$, $\epsilon \sim \mathcal{N}(0, I_{d_x})$,

$$\boldsymbol{x} = u,$$
$$\boldsymbol{y} = v + 0.5c\epsilon.$$

6. Quadratic $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$:

$$\boldsymbol{x} \sim \mathcal{U}(-1, 1)^{d_x},$$
$$\boldsymbol{y} = (w^\mathsf{T} \boldsymbol{x})^2 + 0.5c\epsilon.$$

7. W Shape $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$: $u \sim \mathcal{U}(-1,1)^{d_x}$,

$$\begin{aligned}\boldsymbol{x} &\sim \mathcal{U}(-1,1)^{d_x},\\ \boldsymbol{y} &= 4\left[\left((w^\mathsf{T}\boldsymbol{x})^2 - \tfrac{1}{2}\right)^2 + w^\mathsf{T}u/500\right] + 0.5c\epsilon.\end{aligned}$$

8. Two Parabolas $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$: $\epsilon \sim \mathcal{U}(0,1)$, $u \sim \mathcal{B}(0.5)$,

$$\begin{aligned}\boldsymbol{x} &\sim \mathcal{U}(-1,1)^{d_x},\\ \boldsymbol{y} &= \left((w^\mathsf{T}\boldsymbol{x})^2 + 2c\epsilon\right) \cdot (u - \tfrac{1}{2}).\end{aligned}$$

9. Fourth Root $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$:

$$\begin{aligned}\boldsymbol{x} &\sim \mathcal{U}(-1,1)^{d_x},\\ \boldsymbol{y} &= |w^\mathsf{T}\boldsymbol{x}|^{\frac{1}{4}} + \frac{c}{4}\epsilon.\end{aligned}$$

10. Logarithmic $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}^{d_x}$: $\epsilon \sim \mathcal{N}(0, I_{d_x})$

$$\begin{aligned}\boldsymbol{x} &\sim \mathcal{N}(0, I_{d_x}),\\ \boldsymbol{y}^d &= 2log(\boldsymbol{x}^d) + 3c\epsilon^d,\end{aligned}$$

for $d = 1, \ldots, d_x$.

11. Circle $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$: $u \sim \mathcal{U}(-1,1)^{d_x}$, $\epsilon \sim \mathcal{N}(0, I_{d_x})$, $r = 1$,

$$\begin{aligned}\boldsymbol{x}^d &= r\left(\sin(\pi u^{d+1})\prod_{j=1}^{d}\cos(\pi u^j) + 0.4\epsilon^d\right) \text{ for } d = 1, \ldots, d_x - 1,\\ \boldsymbol{x}^{d_x} &= r\left(\prod_{j=1}^{d_x}\cos(\pi u^j) + 0.4\epsilon^{d_x}\right),\\ \boldsymbol{y} &= \sin(\pi u^1).\end{aligned}$$

12. Ellipse $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$: Same as above except $r = 5$.

13. Spiral $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$: $u \sim \mathcal{U}(0,5)$, $\epsilon \sim \mathcal{N}(0,1)$,

$$\begin{aligned}\boldsymbol{x}^d &= u\sin(\pi u)[\cos(\pi u)]^d \text{ for } d = 1, \ldots, d_x - 1,\\ \boldsymbol{x}^{d_x} &= u[\cos(\pi u)]^{d_x},\\ \boldsymbol{y} &= u\sin(\pi u) + 0.4(d_x - 1)\epsilon.\end{aligned}$$

14. Square $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}^{d_x}$: Let $u \sim \mathcal{U}(-1,1)$, $v \sim \mathcal{U}(-1,1)$, $\epsilon \sim \mathcal{N}(0,1)^{d_x}$, $\theta = -\frac{\pi}{8}$. Then

$$\begin{aligned}\boldsymbol{x}^d &= u\cos\theta + v\sin\theta + 0.05d_x\epsilon^d,\\ \boldsymbol{y}^d &= -u\sin\theta + v\cos\theta,\end{aligned}$$

for $d = 1, \ldots, d_x$.

15. Diamond $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}^{d_x}$: Same as above except $\theta = -\frac{\pi}{4}$.
16. Sine Period 1/2 $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$: $u \sim \mathcal{U}(-1, 1)$, $v \sim \mathcal{N}(0, 1)^{d_x}$, $\theta = 4\pi$,

$$\boldsymbol{x}^d = u + 0.02 d_x v^d \text{ for } d = 1, \dots, d_x,$$
$$\boldsymbol{y} = \sin(\theta x) + c\epsilon.$$

17. Sine Period 1/8 $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$: Same as above except $\theta = 16\pi$ and the noise is changed to $0.5c\epsilon$.
18. Multiplicative Noise $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}^{d_x}$: $u \sim \mathcal{N}(0, I_{d_x})$, $\epsilon \sim \mathcal{N}(0, I_{d_x})$,

$$\boldsymbol{x} \sim \mathcal{N}(0, I_{d_x}),$$
$$\boldsymbol{y}^d = u^d \boldsymbol{x}^d + 0.5\epsilon^d,$$

for $d = 1, \dots, d_x$.
19. Uncorrelated Binomial $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$: $u \sim \mathcal{B}(0.5)$,

$$\boldsymbol{x} \sim \mathcal{B}(0.5)^{d_x},$$
$$\boldsymbol{y} = (2u - 1) w^{\mathsf{T}} \boldsymbol{x} + 0.6\epsilon.$$

20. Independent Clouds $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}^{d_x}$: Let $u \sim \mathcal{N}(0, I_{d_x})$, $v \sim \mathcal{N}(0, I_{d_x})$, $u' \sim \mathcal{B}(0.5)^{d_x}$, $v' \sim \mathcal{B}(0.5)^{d_x}$. Then

$$\boldsymbol{x} = u/3 + 2u' - 1,$$
$$\boldsymbol{y} = v/3 + 2v' - 1.$$

For each distribution, $\boldsymbol{x}$ and $\boldsymbol{y}$ are clearly dependent except (20); for some settings (11-15) they are conditionally independent upon conditioning on the respective auxiliary variables, while for others they are "directly" dependent. Then we can independently generate $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ from $(\boldsymbol{x}, \boldsymbol{y})$ for $i = 1, \dots, n$, set $X = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{d_x \times n}$ and $Y = [\boldsymbol{y}_1, \cdots, \boldsymbol{y}_n] \in \mathbb{R}^{d_y \times n}$, and calculate local / global correlations for the sample data. A visualization of each dependency is shown in Figure A1.

For the increasing dimension simulation in the main paper, we always set $c = 0$ and $n = 100$, with d_x increasing while $d_y = d_x$ for type $5, 10, 14, 15, 18, 20$ and $d_y = 1$ otherwise. The decaying vector w is utilized for $d_x > 1$ to treat higher dimensions as small perturbations, which creates a meaningful setting for testing power comparison. The powers of all three MGC implementations in this setting are provided in Figure A2, where we denote MGC_D as the MGC for DCORR, MGC_M as the MGC for MCORR, MGC_P as the MGC for MANTEL.

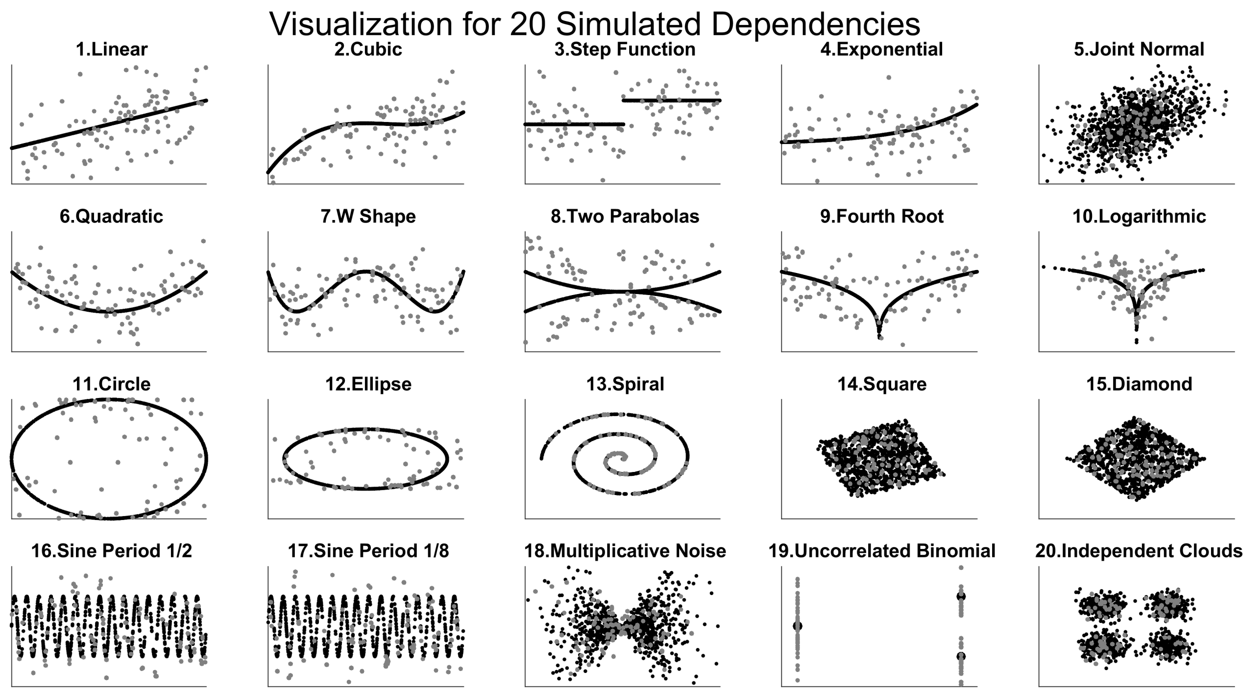

Figure A1: Visualization of the 20 dependencies for 1-dimensional simulations. The blue points are generated with noise (c=1) at $n = 100$ to show the actual sample data in testing, and the red points are generated without noise at $n = 1000$ to highlight each underlying dependency.

Here we also present an additional setting, which sets $d_x = d_y = 1$ and $c = 1$ with the sample size n increasing from 5 to 100. The parameter before c (e.g., there is a 80 before c in type 2) is a tuned noise parameter for some dependencies, so the testing powers can be compared meaningfully for each simulation, i.e., in the absence of noise, the testing powers may converge to 1 at very small n for some trivial dependencies like linear; and it is also more meaningful to consider noisy simulations in practice. The powers of all methods in this setting are provided in Figure A3, with the multiscale power maps shown in Figure A4.

Clearly Mgc always improves over its global counterpart, and always has a large advantage regardless of the underlying dependency structure, the dimensionality, the sample size, or noise.

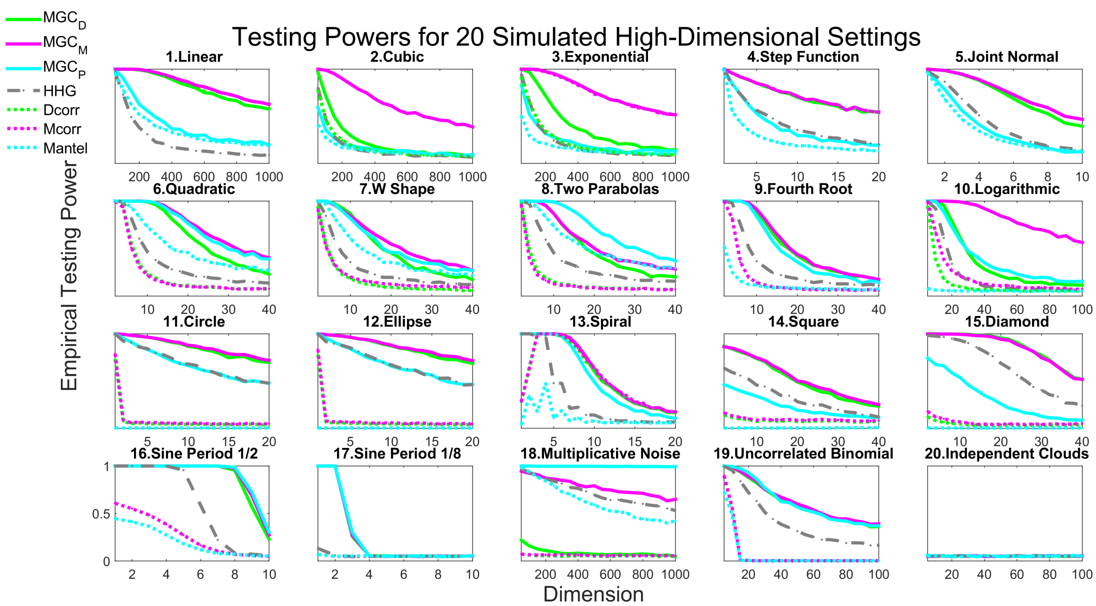

Figure A2: Same as Figure 2 but includes all three different Mgc implementations.

B Dependence Measures

In this section, we review the Mantel test, distance correlation, modified distance correlation, the Mgc statistic, and the Hhg statistic in order. Note that for Dcorr / Mcorr, we implement them in a slightly different but equivalent way from the original definition.

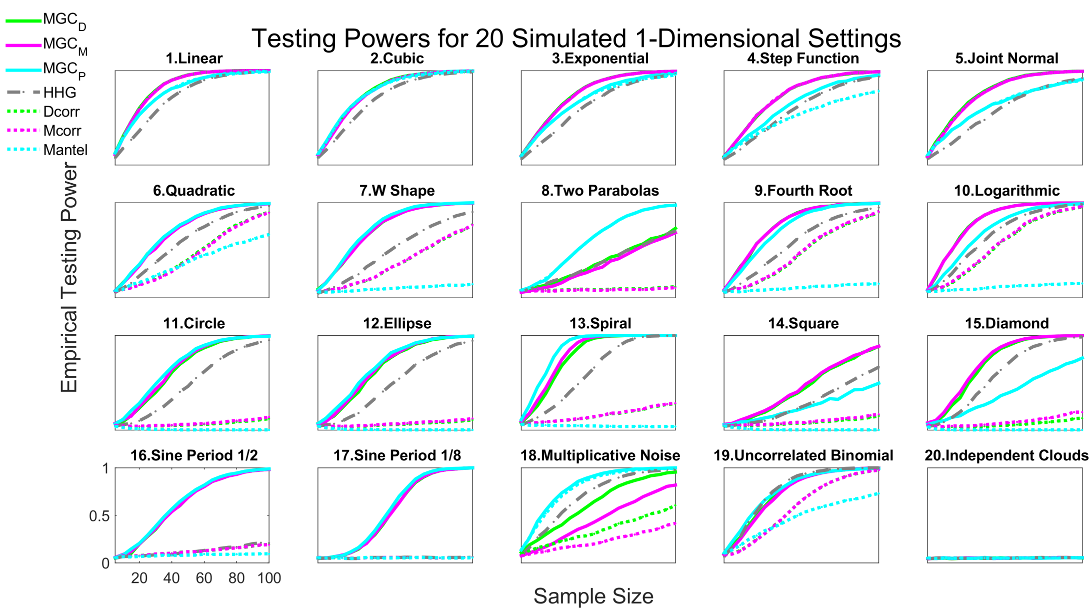

Figure A3: Powers of different methods for 20 different 1-dimensional dependence structures, estimated by the empirical distributions of the test statistics under the null and the alternative on the basis of $10{,}000$ Monte-Carlo replicates. $2{,}000$ additional MC replicates are used for optimal scale estimation for Mgc. Each panel shows empirical testing power on the absicca at a significant level $\alpha = 0.05$, and sample size on the ordinate. Mgc empirically achieves similar or better power than the previous state of the art approaches for all sample sizes on nearly all problems.

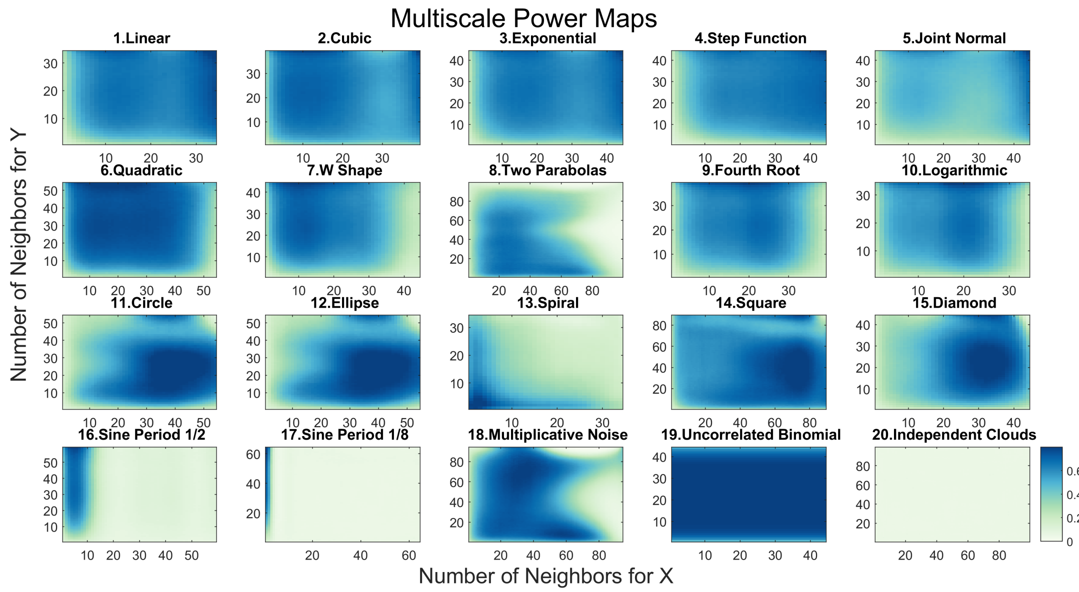

Figure A4: Influence of neighborhood size on testing power of local correlations. For each simulation, the dimension is 1, and the sample size is determined by the first sample size n for MGC to have powers exceeding the threshold 0.8.

B.1 (Global) MANTEL Test

Given the Euclidean distance matrices $\tilde{A}$ and $\tilde{B}$, the MANTEL coefficient [8] is defined as

$$\texttt{Mantel}(X,Y) = \frac{\sum_{i\neq j}^{n}(a_{ij}-\bar{a})(b_{ij}-\bar{b})}{\sqrt{\sum_{i\neq j}^{n}(a_{ij}-\bar{a})^2 \sum_{i\neq j}^{n}(b_{ij}-\bar{b})^2}}, \tag{1}$$

where $A=\tilde{A}$, $B=\tilde{B}$, $\bar{a}=\frac{1}{n(n-1)}\sum_{i\neq j}^{n}(a_{ij})$ and similarly for $\bar{b}$. Then the MANTEL test is carried out by the permutation test.

Unlike distance correlation and HHG, the MANTEL test is not consistent against all dependent alternatives, but it has been a very popular method in biology and ecology due to its simplicity. It is clear from Figure A2 and A3 that global MANTEL is sub-optimal and appears to be not consistent for many dependencies, yet MGC_P achieves comparable performances as other variants of MGC, which implies that MGC_P may be consistent against most, if not all dependent alternatives.

B.2 (Global) Distance Correlation

Given two distance matrices $\tilde{A}$ and $\tilde{B}$ of the sample data X and Y, the sample distance covariance is defined by doubly centering the distance matrices:

$$dcov(X,Y) = \frac{1}{n^2} \sum_{i,j=1}^{n} a_{ij} b_{ij}, \tag{2}$$

where $A = H\tilde{A}H$, $B = H\tilde{B}H$ with $H = I_n - \frac{J_n}{n}$. Then the sample distance variance is defined as

$$dvar(X) = \frac{1}{n^2} \sum_{i,j=1}^{n} a_{ij}^2,$$
$$dvar(Y) = \frac{1}{n^2} \sum_{i,j=1}^{n} b_{ij}^2,$$

and the sample distance correlation equals

$$\texttt{Dcorr}(X,Y) = \frac{dcov(X)}{\sqrt{dvar(X) \cdot dvar(Y)}}. \tag{3}$$

It is shown in [9] that as $n \to \infty$, $\texttt{Dcorr}(X,Y) \to \texttt{Dcorr}(\boldsymbol{x},\boldsymbol{y}) \geq 0$, where $\texttt{Dcorr}(\boldsymbol{x},\boldsymbol{y})$ denotes the population distance correlation between the underlying random variable $\boldsymbol{x}$ and $\boldsymbol{y}$. The population distance correlation is defined by the characteristic functions, which is 0 if and only if $\boldsymbol{x}$ and $\boldsymbol{y}$ are independent. Thus the sample distance correlation is a consistent statistic for testing independence, i.e., the testing power $\beta_\alpha(\texttt{Dcorr}(X,Y))$ converges to 1 as n increases, at any type 1 error level α. Note that all of $dcov, dvar$, Dcorr are always non-negative; and the consistency result assumes finite second moments of $\boldsymbol{x}$ and $\boldsymbol{y}$, which holds for a family of metrics not limited to the Euclidean distance [11]. Also note that the Dcorr above is actually the square of distance correlation in [9], but for ease of presentation the square naming is dropped here.

Alternatively, calculating the distance covariance by $A = H\tilde{A}$ and $B = \tilde{B}H$ gives the same statistic as in Equation 2, i.e., instead of using doubly centered distance matrices, it is the same to singly center one distance matrix by row and the other distance matrix by column. Then Dcorr by singly centered distance matrices has the same testing power as the original Dcorr, because distance covariance is equivalent to distance correlation in the permutation test (note that the actual Dcorr statistic by single centering is different from the original Dcorr, as using single centering changes the distance variances).

In our implementation of global / local Dcorr, we always use singly centered distance matrices rather than doubly centered distance matrices. Although they are equivalent for the testing power of global Dcorr, our alternative implementation improves the testing power of local Dcorr and Mgc. This is because the ranking information of $\tilde{A}$ and $\tilde{B}$ are better preserved in singly centered distance matrices, so that Mgc is more effective in excluding far-away points that exhibit insignificant dependency. This applies to Mcorr as well.

B.3 (Global) Modified Distance Correlation

In case of high-dimensional data where the dimension d_x or d_y increases with the sample size n, the sample distance correlation may no longer be appropriate. For example, even for independent Gaussian distributions, $\texttt{Dcorr}(X,Y) \to 1$ as $d_x, d_y \to \infty$, which may severally impair the testing power of sample Dcorr in high-dimensional simulations.

The modified distance correlation is proposed in [10] to tackle the bias of sample Dcorr. Denote the Euclidean distance matrices as $\tilde{A}$ and $\tilde{B}$, the doubly centered distance matrices as $\hat{A}$ and $\hat{B}$, the modified distance covariance is defined as

$$mcov(X,Y) = \frac{n}{(n-1)^2(n-3)}(\sum_{i\neq j}^{n} a_{ij}b_{ij} - \frac{2}{n-2}\sum_{j=1}^{n} a_{jj}b_{jj}), \tag{4}$$

where A modifies the entries of $\hat{A}$ by

$$a_{ij} = \begin{cases} \hat{a}_{ij} - \frac{\tilde{a}_{ij}}{n}, & \text{if } i \neq j, \\ \frac{n\sum_i \tilde{a}_{ij} - \sum_{i,j}\tilde{a}_{ij}}{n^2}, & \text{if } i = j, \end{cases}$$

and so is B. Then $mvar(X)$ and $mvar(Y)$ can be similarly defined.

If $mvar(X) \cdot mvar(Y) \leq 0$, the modified distance correlation is set to 0 (negativity can only occur when $n \leq 2$, equality can only happen in some special cases); otherwise it is defined as

$$\texttt{Mcorr}(X,Y) = \frac{mcov(X,Y)}{\sqrt{mvar(X) \cdot mvar(Y)}}. \tag{5}$$

It is shown in [10] that $\texttt{Mcorr}(X,Y)$ is an unbiased estimator of the population distance correlation $\texttt{Dcorr}(\boldsymbol{x},\boldsymbol{y})$ for all d_x, d_y, n; and Mcorr is approximately normal even if $d_x, d_y \to \infty$. Thus it is a consistent statistic for testing independence, but may work better than Dcorr under high-dimension dependencies.

Similar to the alternative implementation of Dcorr, we can also use singly centered distance matrices for $\hat{A}$ and $\hat{B}$ in defining Mcorr, which does not alter the theoretical advantages of original Mcorr. We further set $A_{ii} = B_{ii} = 0$ for all i, which simplifies the expression of Mcorr and is asymptotically equivalent for the testing purpose.

B.4 Multiscale Graph Correlations (Mgc)

For any generalized correlation coefficient, its local correlations can be directly implemented as in Equation 3, by plugging in the respective a_{ij} and b_{ij} from Equation 1 and sorting the distance matrices column-wise as in Equation 2.

In particular, Mantel sets a_{ij} and b_{ij} as the respective entry of $\tilde{A}$ and $\tilde{B}$ (the Euclidean distances). Dcorr lets a_{ij} and b_{ij} be the respective matrix entry of A and B (the doubly centered distance matrices), then the sample means $\bar{a}, \bar{b}$ are automatically 0. Mcorr slightly modifies a_{ij} and b_{ij} of Dcorr to adjust their high-dimensional bias. As discussed already, our version of Mgc$_M$ is based on single centering throughout: we take $a_{ij} = b_{ij} = 0$ when $i = j$, otherwise set a_{ij} as the matrix entry of $H\tilde{A} - \tilde{A}/n$, and set b_{ij} as the entry of $\tilde{B}H - \tilde{B}/n$. Then the local version of Mcorr follows by Equation 3.

Generally, there are a total of $\max(R(a_{ij})) \times \max(R(b_{ij}))$ local correlations, which equals n^2 when there exists no repeating data. Note that we use minimal ranks in sorting when ties occur, which indexes all local correlations more conveniently than breaking ties randomly or using average / max ranks.

Among all possible local correlations, MGC picks the optimal local correlation that yields the best testing power. The optimal scale clearly exists, but is distribution dependent and is almost always non-unique. Among all local correlations, it suffices to exclude C^{1l} and C^{k1} for testing and optimal scale estimation: since $C^{1l} = C^{k1} = C^{11}$, they do not include any neighbor other than each observation itself, merely count the diagonal terms in the distance matrices, and are not meaningful for the testing purpose.

B.5 Heller, Heller & Gorfine (Hhg)

The Hhg statistic applies Pearson's chi-square test to ranks of distances within each column, and is shown to be better than many global tests including Dcorr under common nonlinear dependencies in [27, 28]. Like Dcorr and Mcorr, Hhg is distance-based and consistent, but not in the form of the generalized correlation coefficient; and like our Mgc, it makes use of the rank information, but in a distinct manner.

Given the Euclidean distance matrices $\tilde{A} = [\tilde{a}_{ij}]$ and $\tilde{B} = [\tilde{b}_{ij}]$, we denote

$$
\begin{aligned}
H_{11}(i,j) &= \sum_{q=1, q\neq i,j}^{n} I(\tilde{a}_{ik} \leq \tilde{a}_{ij}) I(\tilde{b}_{ik} \leq \tilde{b}_{ij}) \\
H_{12}(i,j) &= \sum_{q=1, q\neq i,j}^{n} I(\tilde{a}_{ik} \leq \tilde{a}_{ij}) I(\tilde{b}_{ik} > \tilde{b}_{ij}) \\
H_{21}(i,j) &= \sum_{q=1, q\neq i,j}^{n} I(\tilde{a}_{ik} > \tilde{a}_{ij}) I(\tilde{b}_{ik} \leq \tilde{b}_{ij}) \\
H_{22}(i,j) &= \sum_{q=1, q\neq i,j}^{n} I(\tilde{a}_{ik} > \tilde{a}_{ij}) I(\tilde{b}_{ik} > \tilde{b}_{ij}),
\end{aligned}
$$

and the Hhg statistic is defined as

$$
\texttt{Hhg}(X,Y) = \sum_{i=1, j\neq i}^{n} \frac{(n-2)(H_{12}(i,j)H_{21}(i,j) - H_{11}(i,j)H_{22}(i,j))^2}{H_{1\cdot}(i,j)H_{2\cdot}(i,j) - H_{\cdot 1}(i,j)H_{\cdot 2}(i,j)},
$$

where $H_{1\cdot} = H_{11} + H_{12}$, $H_{2\cdot} = H_{21} + H_{22}$, $H_{\cdot 1} = H_{11} + H_{21}$, and $H_{\cdot 2} = H_{12} + H_{22}$. It is clear that Hhg is structurally different from Dcorr / Mcorr / Mantel, cannot be conveniently expressed by Equation 1, and there is no direct extension of local correlation to Hhg.

The permutation test using the Hhg statistic is consistent against all dependent alternatives. In our numerical simulations, Hhg falls a bit short when testing against high-dimensional and noisy linear dependencies, but is often more advantageous than global correlations under nonlinear dependencies, which makes it a strong competitor in general.

C Mgc Algorithms and Testing Procedures

In this section we elaborate on the algorithms for computing local correlation and Mgc, as well as their testing procedures in simulations and real data experiment.

Five algorithms are presented in section C.1: given the choice of a global correlation coefficient, algorithm 1 computes one local correlation coefficient at a given (k,l); then algorithm 2 shows how to compute all local correlations simultaneously; algorithm 3 computes the p-values of all local correlation by the random permutation test; algorithm 4 approximates the optimal scale for Mgc based on the p-values of all local correlations, and outputs the approximated p-value of Mgc; algorithm 5 estimates the testing powers of all local statistics based on a given joint distribution or multiple pairs of data, which can be used to more accurately estimate the optimal scale for

Mgc when the underlying model is known or training data are given. More detailed discussions regarding the optimal scale approximation is offered in section C.2.

C.1 Algorithms

All algorithms are implemented in Matlab and R with the pseudo-code shown below. For ease of presentation, we assume there are no repeating data and take Dcorr as the global correlation in the pseudo-code.

Algorithm 1 shows a straightforward computation of one local correlation coefficient, which requires $O(n^2)$ once the rank information is provided. This is suitable for Mgc computation when the optimal local scale is known or already estimated. But using algorithm 1 to compute all local correlations would require iterating through all possible neighborhoods (k,l), which takes $O(n^4)$ and would make the optimal scale estimation computationally inefficient.

To facilitate the optimal scale estimation, algorithm 2 provides a fast method to compute all local correlations in $O(n^2)$. An important observation is that each product $a_{ij}b_{ij}$ is included in C^{kl} if and only if (k,l) satisfies $k \leq R(a_{ij})$ and $l \leq R(b_{ij})$, so it suffices to iterate through $a_{ij}b_{ij}$ for $i,j = 1,\ldots,n$, and add the product simultaneously to all C^{kl} whose scales are no more than $(R(a_{ij}), R(b_{ij}))$. However, accessing and adding multiple C^{kl} at the same time is not computationally efficient; instead, for each product, we only add it to C^{kl} at $(k,l) = (R(a_{ij}), R(b_{ij}))$ (so only one local scale is accessed for each operation), iterate through all products for $i,j = 1,\ldots,n$, then add up adjacent C^{kl} for $k,l = 1,\ldots,n$. Thus all local correlations can be computed in $O(n^2)$, which has the same running time complexity as the global distance correlation. There are two additional overheads: sorting the distance matrices column-wise takes $O(n^2 \log n)$, and properly centering the distance matrices takes $O(n^2)$.

Algorithm 3 computes the p-values of all local correlation by the permutation test with r random permutations, which takes $O(rn^2 \log n)$.

Algorithm 4 approximates the optimal scale (k^*, l^*) from the p-values of all local correlations, and outputs the approximated Mgc p-value. This is necessary for testing on one pair of data with unknown model, while algorithm 5 is more appropriate for known model. Conceptually, the algorithm first searches for a set of "valid" adjacent rows $\mathcal{K} = \{k_1, k_1+1, \ldots, k_2-1, k_2\}$ such that the median p-value of $\{p_{kl}, k \in \mathcal{K}, l = 2, \ldots, n\}$ is no larger than $\alpha/(n-1) * |\mathcal{K}|$, otherwise we take

$\mathcal{K} = \{n\}$; and similarly determine the set of valid columns $\mathcal{L}$. Once $\mathcal{K}$ and $\mathcal{L}$ are determined, the optimal scale (k^*, l^*) is found by the scale that minimizes the p-value within $\{p_{kl}, k \in \mathcal{K}, l \in \mathcal{L}\}$. Clearly if the majority p-values of all local correlations are less than α, then $\mathcal{K} = \mathcal{L} = \{1, \ldots, n\}$, and the optimal scale equals the scale that minimizes the p-values among all local correlations; if there is no valid rows and columns, then Mgc takes the largest scale and equals the global correlation. Note that the actual algorithm is a simpler version of the above description: instead of considering all possible sets of rows and check the validity, we limit the check to the most likely set of rows, by first looking for the row scale of the smallest p-value, then including all adjacent rows whose minimal p-value on the row is no larger than α; similarly for the set of columns.

Algorithm 5 computes the testing powers of all local correlations by repeated simulating samples generated from the joint distribution f_{xy}. Sample data under the null and the alternative are repeatedly generated for r Monte-Carlo replicates, and algorithm 2 is applied to compute the sample local correlations under the null and the alternative. Then the testing power at each local correlation can be estimated, and the Mgc optimal scale can be found by maximizing the powers. This algorithm is also applicable if there exists multiple pairs of data with unknown model but similar dependency structure, then the alternative statistic can be computed from each data pair while the null statistic can be computed from each data pair under permutation. The running time is $O(rn^2 \log n)$.

C.2 Discussions of Optimal Scale Estimation

To evaluate Mgc in simulations or real data, the optimal scale for Mgc always needs to be estimated first. Algorithm 5 computes the testing powers of all local correlations for known model, so the optimal scale (k^*, l^*) can be directly estimated by maximizing the testing powers (if there are more than one optimal scales, one may pick the scale that maximizes the mean difference of the test statistic under the null and the alternative). Once the optimal scale is determined, the testing power of Mgc under the given model can be quickly determined by algorithm 5, and its p-value for testing on a particular pair of data can be determined by algorithm 3.

If there is only one pair of data (X, Y) with unknown distributions, we have to approximate the optimal scale by algorithm 4. It makes use of Bonferroni correction to separately verify the set of rows and columns, which guarantees the false positive rate to be no higher than α; otherwise the scale is set to the largest, which guarantees the approximated Mgc is at least as powerful as the

Algorithm 1 Local Correlation Computation for One Scale

Input: A pair of distance matrices $(\tilde{A}, \tilde{B}) \in \mathbb{R}^{n \times n} \times \mathbb{R}^{n \times n}$, and the given local scale $(k, l) \in \mathbb{R} \times \mathbb{R}$.

Output: The local correlation coefficient $C^{kl} \in [-1, 1]$ at the given (k, l).

function LOCALCORR($\tilde{A}$,$\tilde{B}$,k,l)

  initialize C^{kl}, V_k^A, V_l^B, E_k^A, E_l^B as 0.

  for $Z := A, B$ **do** $R^Z = $ SORT($\tilde{Z}$) **end for** ▷ column-wise sorting and assume no ties

  for $Z := A, B$ **do** $Z = $ CENTER($\tilde{Z}$) **end for** ▷ proper centering of the distance matrices

  for $i, j = 1, \ldots, n$ **do**

    $C^{kl} = C^{kl} + A_{ij} B_{ij} \boldsymbol{I}(R_{ij}^A \leq k) \boldsymbol{I}(R_{ij}^B \leq l)$ ▷ store local distance covariance

    $V_k^A = V_k^A + A_{ij}^2 \boldsymbol{I}(R_{ij}^A \leq k)$ ▷ store local distance variance for X

    $V_l^B = V_l^B + B_{ij}^2 \boldsymbol{I}(R_{ij}^B \leq l)$ ▷ store local distance variance for Y

    $E_k^A = E_k^A + A_{ij} \boldsymbol{I}(R_{ij}^A \leq k)$ ▷ store the sample means

    $E_l^B = E_l^B + B_{ij} \boldsymbol{I}(R_{ij}^B \leq l)$

  end for

  $C^{kl} = \left(C^{kl} - E_k^A E_l^B / n^2\right) / \sqrt{\left(V_k^A - {E_k^A}^2 / n^2\right) \left(V_l^B - {E_l^B}^2 / n^2\right)}$ ▷ normalize the local covariances

end function

Algorithm 2 $O(n^2 \log n)$ Algorithm for Computing All Local Correlations

Input: A pair of distance matrices $(\tilde{A}, \tilde{B}) \in \mathbb{R}^{n \times n} \times \mathbb{R}^{n \times n}$.

Output: All local correlation coefficients $C^{kl} \in [-1, 1]^{n \times n}$ for $k, l = 1, \ldots, n$.

function LOCALCORR($\tilde{A}$,$\tilde{B}$)

  initialize C as a zero matrix of size $n \times n$; V^A, V^B, E^A, E^B as zero vectors of size n.

  for $Z := A, B$ **do** $R^Z = \text{SORT}(\tilde{Z})$ **end for**

  for $Z := A, B$ **do** $Z = \text{CENTER}(\tilde{Z})$ **end for**

  for $i, j = 1, \ldots, n$ **do**

    $k = R^A_{ij}$

    $l = R^B_{ij}$

    $C^{kl} = C^{kl} + A_{ij} B_{ij}$

    $V^A_k = V^A_k + A^2_{ij}$

    $V^B_l = V^B_l + B^2_{ij}$

    $E^A_k = E^A_k + A_{ij}$

    $E^B_l = E^B_l + B_{ij}$

  end for

    ▷ the next two for loops with respect to the scales guarantee the computation of all local covariance / variance in $O(n^2)$

  for $k = 1, \ldots, n-1$ **do**

    $C^{1,k+1} = C^{1,k} + C^{1,k+1}$

    $C^{k+1,1} = C^{k+1,1} + C^{k+1,1}$

    for $Z := A, B$ **do** $V^Z_{k+1} = V^Z_k + V^Z_{k+1}$ **end for**

    for $Z := A, B$ **do** $E^Z_{k+1} = E^Z_k + E^Z_{k+1}$ **end for**

  end for

  for $k, l = 1, \ldots, n-1$ **do**

    $C^{k+1,l+1} = C^{k+1,l} + C^{k,l+1} + C^{k+1,l+1} - C^{k,l}$

  end for

  for $k, l = 1, \ldots, n$ **do** ▷ normalize all local covariances

    $C^{kl} = \left(C^{kl} - E^A_k E^B_l / n^2\right) / \sqrt{\left(V^A_k - {E^A_k}^2 / n^2\right)\left(V^B_l - {E^B_l}^2 / n^2\right)}$

  end for

end function

Algorithm 3 P-value Computation for All Local Correlations

Input: A pair of distance matrices $(\tilde{A}, \tilde{B}) \in \mathbb{R}^{n \times n} \times \mathbb{R}^{n \times n}$, the number of permutations r.

Output: The p-value matrix $P \in [0,1]^{n \times n}$ for all local distance correlations.

function PermutationTest($\tilde{A}$,$\tilde{B}$,r)
    $C^{kl} =$ LocalCorr$(\tilde{A}, \tilde{B})$ ▷ calculate the observed local correlations
    for $j = 1, \ldots, r$ **do**
        $\pi =$ RandPerm(n) ▷ generate a random permutation of size n
        $C_0^{kl}[j] =$ LocalCorr$(\tilde{A}, \tilde{B}(\pi, \pi))$ ▷ calculate the permuted test statistics
    end for
    for $k, l = 1, \ldots, n$ **do**
        $P_{kl} = \sum_{j=1}^{r} (C^{kl} < C_0^{kl}[j])/r$ ▷ get the p-value at each local scale
    end for
end function

global correlation. Still, algorithm 4 is a heuristic approach to approximate the optimal local scale, which does not guarantee the optimal local correlation to be always correctly identified.

To better justify algorithm 4, we compare the estimated Mgc power by algorithm 4 to the true Mgc power by algorithm 5, with the global Mcorr and Hhg as benchmarks. For each type of dependency in the simulation section, we generate $1{,}000$ pairs of dependent data by the same low- and high-dimensional settings as in Figure A3 and A2; and for each pair of data, all local p-values are calculated by $1{,}000$ random permutations. By using the true optimal scale (from the simulation section) consistently for each data pair, the true Mgc p-value can be computed; by using algorithm 4 to approximate the optimal scale for each pair of data separately, the estimated Mgc p-value can be computed; and the p-values of global Mcorr and Hhg can also be derived. The null is rejected when the p-value is less than 0.05, and the power equals the percentage of correct rejection. Based on the powers of true Mgc / estimated Mgc / Mcorr / Hhg shown in Figure A5, we observe that although the estimated Mgc power by algorithm 4 can be lower than the true Mgc power, it is almost always better than global Mcorr and Hhg, and combines the better performance of the two benchmarks.

Note that it is tempting to directly use the optimal scale that minimizes all local p-values without the validation by algorithm 4, or generate random samples based on the given data pair and use algorithm 5 by bootstrap. However, both approaches are biased such that the false positive rate will be higher than the type 1 error in the absence of dependency. This is because for a given pair of

Algorithm 4 Optimal Local Scale Approximation by P-values

Input: The p-value matrix $P \in \mathbb{R}^{n \times n}$ of all local distance correlations, the type 1 error level α.

Output: The approximated Mgc optimal scale (k^*, l^*), and the approximated Mgc p-value p.

function MGCScaleVerify(P, α)
    $\mathcal{K} = \text{VerifyRow}(P, \alpha)$ ▷ search for a set of valid row indices
    $\mathcal{L} = \text{VerifyRow}(P^{\mathsf{T}}, \alpha)$ ▷ search for a set of valid column indices
    $[k^*, l^*] = \arg\min_{\{k \in \mathcal{K}, l \in \mathcal{L}\}} P_{kl}$ ▷ find the optimal scale within the valid range
    $p = P_{k^* l^*}$
end function

Input: Same as MGCScaleVerify.

Output: The indices of valid rows.

function VerifyRow(P, α)
    initialize $\mathcal{K}$ as an empty set
    $[k^*, l^*] = \arg\min_{k,l}\{P_{kl}, k, l = 2, \ldots, n\}$
    for $k = k^*, \ldots, 2$ **do** ▷ check all row scales no larger than k^*
        if $\min\{P_{kl}, l = 2, \ldots, n\} > \alpha$ **then**
            break
        end if
        $\mathcal{K} = [k, \mathcal{K}]$
    end for
    for $k = k^* + 1, \ldots, m$ **do** ▷ check all row scales larger than k^*
        if $\min\{P_{kl}, l = 2, \ldots, n\} > \alpha$ **then**
            break
        end if
        $\mathcal{K} = \{\mathcal{K}, k\}$
    end for
    if $\text{Median}(P_{kl}, k \in \mathcal{K}, l = 2, \ldots, n) > \alpha * \frac{|\mathcal{K}|}{n-1}$ **then**
        $\mathcal{K} = \{n\}$ ▷ take the largest scale if the median p-value is not sufficiently small
    end if
end function

Algorithm 5 Testing Powers Computation for All Local Correlations

Input: A joint distribution f_{xy}, the sample size n, the number of MC replicates r, and the type 1 error level α.

Output: The power matrix $\beta_\alpha \in [0,1]^{n \times n}$ for all local correlations, and the `Mgc` optimal scale $(k^*, l^*) \in \mathbb{R} \times \mathbb{R}$.

function TESTINGPOWERS(f_{xy},n, r,α)
  for $j = 1, \ldots, r$ **do**
    for $i := [n]$ **do** $(X_i^1, Y_i^1) \overset{iid}{\sim} f_{xy}$ **end for** ▷ generate dependent samples
    for $i := [n]$ **do** $X_i^0 \overset{iid}{\sim} f_x$ **end for** ▷ generate independent samples
    for $i := [n]$ **do** $Y_i^0 \overset{iid}{\sim} f_y$ **end for**
    for $Z := A, B$ **do** $\tilde{Z}_1 = \text{DIST}(Z_1)$ **end for** ▷ the distance matrices under the alternative
    for $Z := A, B$ **do** $\tilde{Z}_0 = \text{DIST}(Z_0)$ **end for** ▷ the distance matrices under the null
    $C_1^{kl}[j] = \text{LOCALCORR}(\tilde{A}_1, \tilde{B}_1)$ ▷ calculate all local correlations under the alternative
    $C_0^{kl}[j] = \text{LOCALCORR}(\tilde{A}_0, \tilde{B}_0)$ ▷ calculate all local correlations under the null
  end for
  for $k, l = 1, \ldots, n$ **do**
    $c_\alpha = \text{CDF}_{1-\alpha}(C_{kl}^0[j], j \in [r])$ ▷ get the critical value by the empirical cumulative distribution under the null at each scale
    $\beta_\alpha^{kl} = \sum_{j=1}^{r} (C_{kl}^1[j] > c_\alpha)/r$ ▷ estimate the power
  end for
  $(k^*, l^*) = \arg\max(\beta_\alpha^{kl})$ ▷ find the optimal local scale
end function

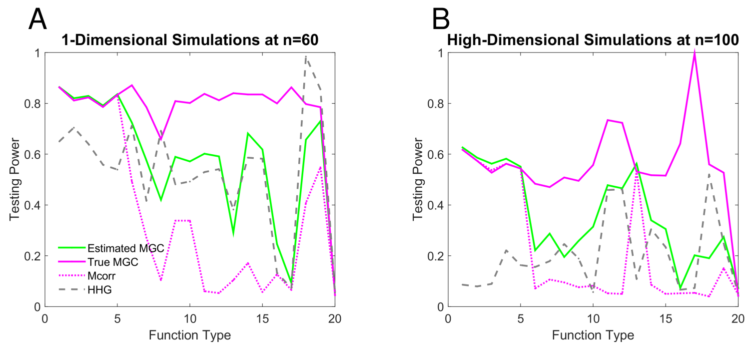

Figure A5: Comparing estimated Mgc power to true Mgc power, for the 1-dimensional and high-dimensional simulations. (A) 1-dimensional simulations, where $d_x = 1$ and the sample size is chosen by the power threshold 0.8 as in Figure A4. (B) High-dimensional simulations, where $n = 100$ and the dimension is chosen by the power threshold 0.5 as in Figure 4. The estimated Mgc power by the approximated optimal scale is almost always better than global Mcorr and Hhg, combines the better performance of the two benchmarks, is quite close to the true Mgc power, and does not inflate false signals.

data, a non-optimal scale can happen to have a significant p-value, which may be falsely identified as optimal if we directly minimize all local p-values. Those erroneous scales often still exist after a straightforward re-sampling, so random samples have the same problem. More investigations into the bias and better methods for searching the optimal scale are two worthwhile directions for future works.

D Proofs

Theorem 1. $\beta(C_t^*) \to 1$ *for all* f_{xy} *in* $\mathcal{F}_t$.

Proof. For any f_{xy}, the power of multiscale graph correlation satisfies

$$\beta(C^*) = \max \boldsymbol{x}_{k,l}\{\beta(C^{kl})\} \geq \beta(C), \tag{6}$$

at any type 1 error level α. So $\beta(C^*) \to 1$ if $\beta(C) \to 1$.

Therefore $\beta(C_t^*) \to 1$ for all f_{xy} in $\mathcal{F}_t$. In particular, Mgc$_D$ and Mgc$_M$ are consistent against all alternative of finite second moments, because Dcorr and Mcorr are consistent against all alternatives of finite second moments by [9, 10]. □

Theorem 2. *If x is linearly dependent on $\boldsymbol{y}$, then for any n it always holds that*

$$\beta(C^{nn}) = \beta(C^{*}) = \beta(C). \tag{7}$$

Thus the optimal scale for $\texttt{Mgc}$ *is the global scale for linearly dependent data.*

Proof. To show that $\texttt{Mgc}$ is equivalent to the global correlation coefficient, it suffices to show the p-value of C^{kl} is always no less than the p-value of C for all k, l under linear dependence.

Under linear dependency, for any global correlation coefficient satisfying Equation 1, by Cauchy-Schwarz inequality it follows that

$$1 = C(X, Y) \geq C(X, YQ) \tag{8}$$

for any permutation matrix Q, where the equality holds if and only if X is a scalar multiple of YQ. It follows that the p-value of C is 0, which is at the minimal.

Therefore the p-value of C^{kl} cannot be less than the p-value of C under linear dependency, such that the global correlation is the optimal scale for $\texttt{Mgc}$ under linear dependency. $\square$

Theorem 3. *There exists f_{xy} and n such that*

$$\beta(C^{*}) > \beta(C). \tag{9}$$

Thus multiscale graph correlation can be better than its global correlation coefficient under certain nonlinear dependency, for finite sample.

Proof. We give a simple discrete example of f_{xy} at $n = 7$, such that the p-value of $\texttt{Mgc}_M$ is strictly lower than the p-value of $\textsc{Mcorr}$.

Suppose under the alternative, each pair of observation $(\boldsymbol{x}, \boldsymbol{y})$ is sampled as follows:

$$\boldsymbol{x} \in \{-1, -\frac{2}{3}, -\frac{1}{3}, 0, \frac{1}{3}, \frac{2}{3}, 1\} \text{ without replacement,}$$
$$\boldsymbol{y} = \boldsymbol{x}^2,$$

which is a discrete version of the quadratic relationship in the simulations.

At $n = 7$, we can directly calculate $C^{kl}(X, Y)$ and $\{C^{kl}(X, YQ)\}$ for all permutation matrices Q. It follows that the p-value of $\textsc{Mcorr}$ is $\frac{151}{210}$, while $C^{kl}(X, Y) = \frac{17}{70}$ at $(k, l) = (2, 4)$. Note that in this case k is bounded above by $n = 7$ while l is bounded above by 4 due the the repeating points in Y.

Then by choosing $\alpha = 0.25$, Mgc has power 1 while global Mcorr has power 0, i.e., Mgc successfully identifies the dependency in this example while global Mcorr fails.

Note that we can always consider sample points in $[-1, 1]$ for X, increase n and reach the same conclusion with more significant p-values; but the computation of all possible permuted test statistics becomes more time-consuming as n increases. The same conclusion also holds for $\texttt{Mgc}_D$ and $\texttt{Mgc}_P$ using the same example. $\square$