

1

Dependence Discovery from Multimodal Data via 2 Multiscale Graph Correlation

3 Cencheng Shen¹, Carey E. Priebe², Mauro Maggioni³, and Joshua T. Vogelstein*⁴

4 ¹Department of Statistics, Temple University

5 ²Department of Applied Mathematics and Statistics, Johns Hopkins University

6 ²Department of Mathematics, Duke University

7 ⁴Department of Biomedical Engineering and Institute for Computational Medicine, Johns
8 Hopkins University

9 June 23, 2016

10

Abstract

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

33 *Keywords:* testing independence, distance correlation, k-nearest-neighbor, local correlation coef-
34 ficient, permutation test

*jovo@jhu.edu

35 **Contents**

36	A Simulation Functions	20
37	B Supplementary Figures	23
38	C Dependence Measures	23
39	C.1 (Global) MANTEL Test	26
40	C.2 (Global) Distance Correlation	27
41	C.3 (Global) Modified Distance Correlation	28
42	C.4 Multiscale Graph Correlations (Mgc)	29
43	C.5 Heller, Heller & Gorfine (HHG)	29
44	D Mgc Algorithms and Testing Procedures	30
45	D.1 Algorithms	31
46	D.2 Discussions of Optimal Scale Estimation	32
47	E Proofs	38

48 Detecting dependency among multiple data sets is one of the most important and fundamental
49 tasks in computational statistics and data science. Indeed, prior to embarking on a predictive
50 machine learning investigation, one might first check whether any dependence is detectable; if not,
51 high-quality predictions will be unlikely. The founders of statistics first highlighted the importance
52 of this task, starting with Pearson, who developed Pearson's Product-Moment Correlation statistic
53 [1]. Since then, researchers have consistently developed new and improved methods (see [2] for
54 a recent review and discussion).

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

76 There are two key insights from the literature that we combine to develop our methodology that
77 satisfies the above desiderata. First, a collection of pairwise comparisons suffices to characterize
78 a joint distribution [4]. Second, nonlinear manifolds can be approximated by local linear spaces
79 [5]. Our approach, Multiscale Graph Correlation (Mgc), leverages and improves upon recent devel-

80 opments from both subdisciplines of data science.

81 Interpoint pairwise comparison matrices have been used for over 100 years for various statistical
82 purposes [4]. Perhaps one of the earliest examples of using them for dependence testing comes
83 from Karl Pearson [1], who created a special case of something subsequently called a “generalized
84 correlation coefficient” [6]. Generalized correlation coefficients start with n pairs of observations
85 (x_i, y_i) , where x ’s and y ’s both might vectors, shapes, networks, etc. And then, a comparison
86 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, $A =$
87 $\{a_{ij}\}$ and $B = \{b_{ij}\}$ are the $n \times n$ interpoint comparison matrices for x and y , respectively. Without
88 loss of generality, assuming A and B have zero mean, a generalized correlation coefficient can
89 then be written:

$$C = \frac{1}{z} \sum_{i,j=1}^n a_{ij} b_{ij}, \quad (1)$$

90 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
91 correlation across *pairwise comparisons*, rather than the individual data samples. C has many
92 well known special cases historically, including Pearson’s [1], Spearman’s [7], Kendall’s [6], and
93 Mantel’s correlation [8]. Recently, Szekely et al. [9] extended these approaches, letting δ_x and δ_y
94 to be the Euclidean distance, followed by subtracting the row means and column means, resulting
95 in “doubly centered” distances. Impressively, they proved that this “distance correlation” (DCORR)
96 statistic is a consistent test for independence for any joint distribution (under suitable regularity
97 conditions), that is, the DCORR’s power approaches 1 as sample size approaches infinity, for any
98 joint distribution of finite dimension and finite second moments. Szekely et al. [10] further proposed
99 a modified version called MCORR, which they prove to be consistent even as the dimensions of x
100 and y increase to infinity as well. Moreover, because these distance based tests merely require
101 a comparison function for both x and y , Lyons was able to prove that they are consistent even in
102 other metric spaces, including certain networks, shapes, and other complicated spaces [11]. Thus,
103 existing generalized correlation coefficient based tests therefore work well in high dimensions
104 and low sample sizes, including in complicated domains, and are reasonably computationally
105 efficient. But, empirically, they struggle in various non-linear settings, perhaps because they do
106 not automatically adapt to the data. Therefore, they also do provide insight into the nature of the
107 dependence.

108 A deep insight that the generalized correlation coefficient tests have yet to capitalized on, that
109 could help address the above described limitations, is that nonlinear shapes can be approximated
110 by **locally** linear ones [5]. Locality has been utilized for classification and regression [12], data

111 compression [13], and recommender systems [14], to name a few of the myriad data science
112 problems for which locality has already reaped benefits. Moreover, it has become an invaluable
113 tool in unfolding nonlinear geometry in many recent development of nonlinear embedding algo-
114 rithms, dating back to the 1950s [15], and more recently making a resurgence with the advent of
115 Isomap [16, 17], Local Linear Embedding [18, 19], and Laplacien eigenmaps [20], among many
116 others. The concept of locality, while popular within certain fields has only entered into testing very
117 infrequently [21–23]. These approaches, like the distance correlation based ones, have the advan-
118 tage of naturally operating on complicated data, because they only require a comparison function
119 between observations. They can also have strong theoretical guarantees. However, these local
120 testing approaches focus on two-sample testing, rather than dependence testing.

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

130 Multiscale Graph Correlation

131 All dependence tests start from the same setting: we observe n pairs of observations (x_i, y_i) , and
132 we first desire to know whether the x 's and y 's are independent of one another, and if so, we then
133 desire to understand the nature of that dependence structure.

134 Multiscale Graph Correlation (Mgc) combines generalized correlation coefficients with locality.
135 Specifically, let $R(a_{ij})$ be the “rank” of x_i relative to x_j , that is, $R(a_{ij}) = k$ if x_i is the k^{th} clos-
136 est point (or “neighbor”) to x_j , starting from 1 to n , and define $R(b_{ij})$ equivalently for the y 's. For
137 any neighborhood size k around each x and any neighborhood size l around each y , we define

138 the rank-truncated pairwise comparisons:

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

139 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
140 we can define a *local* variant of any global generalized correlation coefficient, by excluding large
141 distances:

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

142 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 com-
143 parisons. There are a maximum of n^2 different local correlations, one for each possible combina-
144 tions of k and l (more technical details of MGC are in Appendix C.4). Among all n^2 local statistics,
145 $\{C^{kl}\}$, MGC selects the best local statistic for testing. Figure 1 schematically illustrates our method
146 on a particular nonlinear dependence structure.

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

153 Computing the p-values from the test statistic is straightforward. Specifically, we can permute the
154 labels of either the x_i 's or the y_i 's, and then compute the MGC statistics on the permuted data
155 [25]. By permuting the labels, we have rendered the two different views of the data independent.
156 Doing so many times yields an empirical estimate of the null distribution, which we can use to
157 compute the critical value and p-value. This procedure is somewhat time consuming, which makes
158 computing the test statistics for all neighborhoods efficiently even more important.

159 Nearly all algorithms that employ regularization (for example, sparse methods, feature selection,
160 dimensionality reduction) face a similar dilemma: how to efficiently choose the hyper-parameters.
161 Most manifold learning algorithms require that the user essentially runs the entire algorithm again
162 from scratch for each different hyper-parameter setting, a pursuit that can be exponentially taxing
163 as the number of hyper-parameters increases. In our case, once the rank information is provided,
164 each distance-based local correlation takes $O(n^2)$ to compute (Pseudocode 1 in Appendix D.1),
165 which means a straightforward algorithm to compute all local correlations would take $O(n^4)$.

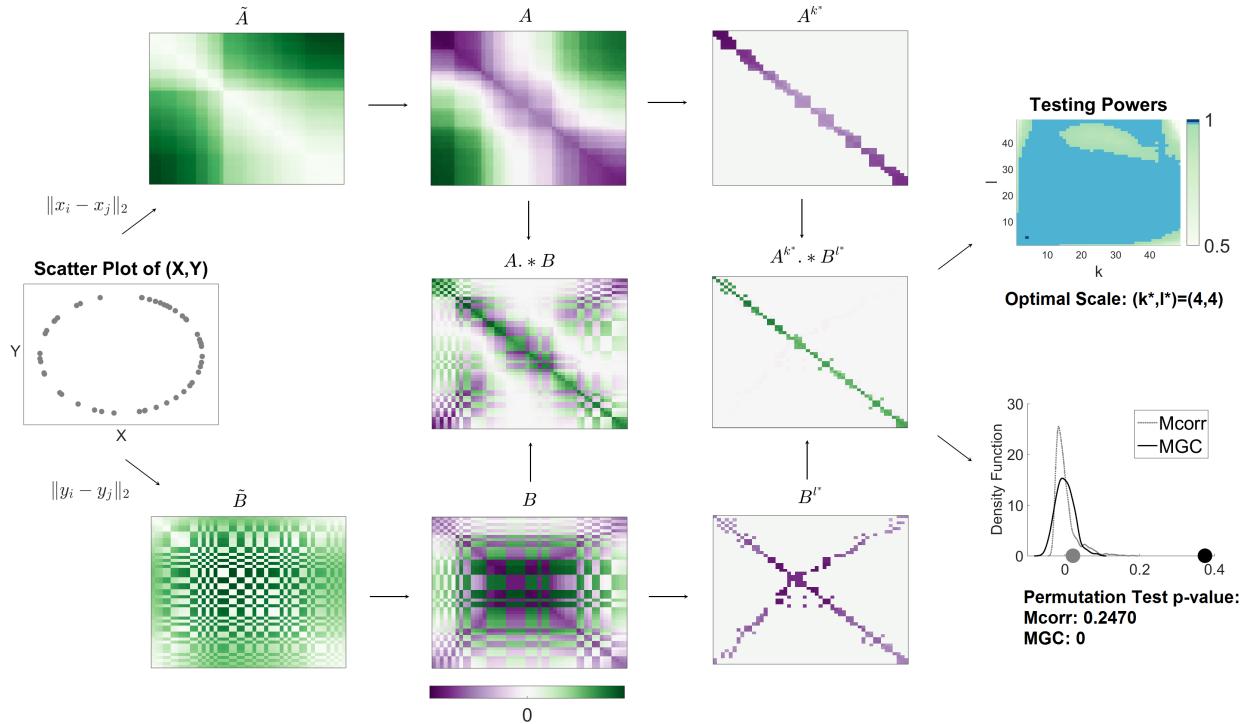


Figure 1: Flowchart for Mgc computation: Column 1: (X, Y) have a circle relationship. Column 2: The heat maps of \tilde{A} and \tilde{B} , which are the pairwise Euclidean distance matrices of X and Y . All distance entries are non-negative. Column 3: The top and bottom panels are the heat maps of $A = \{a_{ij}\}$ 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 $\text{Mcorr}(X, Y)$ 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 heatmap 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 $\text{Mgc}(X, Y)$ 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 $\text{Mgc}(X, Y)$ and $\text{Mcorr}(X, Y)$ as dark and gray dots on the x-axis, as well as the distribution of the permuted test statistics.

166 However, we have devised an algorithm for exactly computing *all* local correlations in $\mathcal{O}(n^2 \log n)$,
167 essentially the same running time complexity as global correlation coefficients (the additional log
168 factor is for sorting to find the neighbors, see Pseudocode 2 in Appendix D.1 for details). We do
169 so by noting that the sufficient statistics for larger neighborhood sizes include those for the smaller
170 sizes, so we can simply keep track of them as we iteratively increase neighborhood size. The end
171 result is M_{GC} can be computed in comparable time as the other leading dependence tests (see
172 Pseudocode 3 in Appendix D.1 for details on computing all n^2 p-values efficiently).

173 Finally, we must find an optimal scale. Our procedure for estimating the optimal scale searches
174 for regions of neighborhood sizes for which p-values are consistently low, guarding against noisy
175 scales that appear optimal, and combating bias added by looking at many different scales. We
176 assert that the optimal scale is the largest neighborhood size in that region. We define the p-value
177 of M_{GC} to be the p-value from the optimal scale, and declare significant dependency when the
178 p-value is less than α , often 0.05 (see Pseudocode 4 and 5 in Appendix D.1 for details).

179 Finite Sample Simulation Experiments

180 We are interested in assessing the performance of our newly proposed multiscale tests in a com-
181 prehensive set of simulations, to better understand which the tests, and gain insight into which to
182 use in different settings. We therefore consider 20 different joint distributions f_{xy} . A large fraction
183 of these are taken exactly from existing literature [9, 26–28], and we have added several additional
184 settings. They include: linear and nearly linear (1-5), polynomial (6-12), trigonometric (13-17),
185 uncorrelated but nonlinearly dependent (18-19), and an independent relationship (20). Details for
186 each setting are given in Appendix A, with a visualization of each dependency shown in Supple-
187 mentary Figure A1.

188 Figure 2 shows the testing powers versus the dimensionality of x (the dimensionality of y increases
189 in only a subset of the settings; see Methods for details), with the sample sizes fixed at $n = 100$
190 for each simulation. We compare our novel test, M_{GC} , with two previously proposed state-of-the-
191 art tests: MCORR [10] and HHG [28]. HHG has previously been demonstrated to perform very well
192 on all sorts of nonlinear dependencies, especially in low-dimensional settings, and enjoys strong
193 theoretical guarantees. More exhaustive benchmark experiments, including focusing on the one-
194 dimensional scenarios, are qualitatively similar, and are therefore relegated to Appendix B, in
195 which we also compare to MANTEL and DCORR, as well as our novel local variants of both MANTEL and

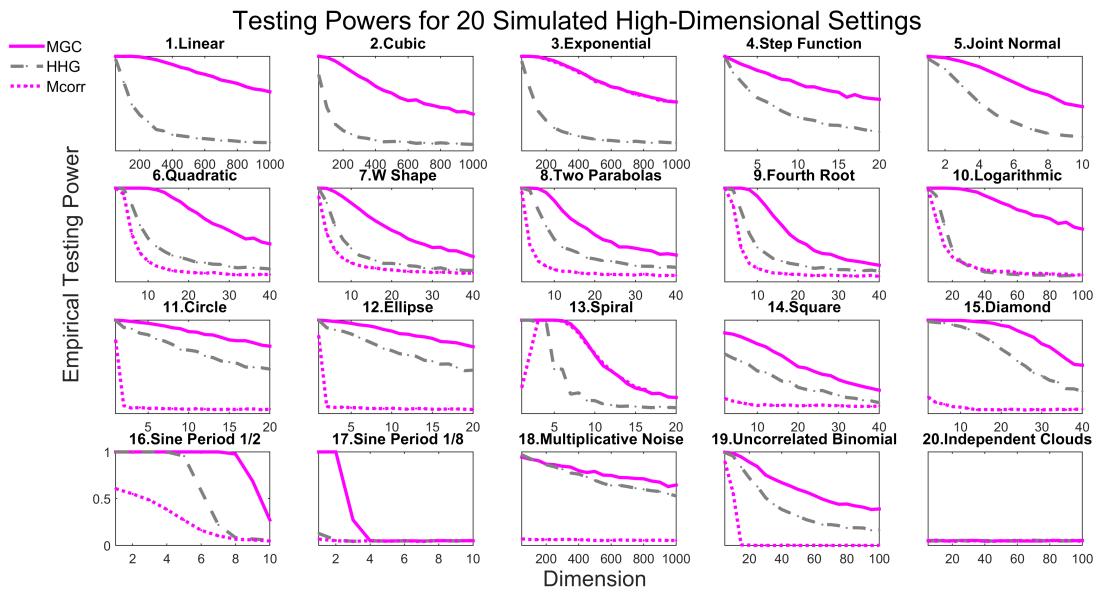


Figure 2: Powers of different methods for 20 different dependence structures, estimated by the empirical distributions of the test statistics under the null and the alternative on the basis of 10,000 Monte-Carlo replicates. 2,000 additional MC replicates are used for optimal scale estimation for Mgc. Each panel shows empirical testing power on the abscissa 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.

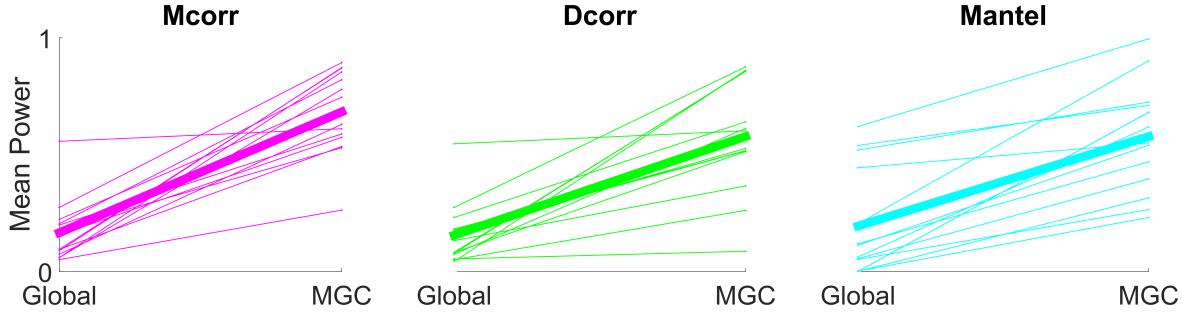


Figure 3: Average powers slopegraphs comparing global and MGC tests. For each global test, the left side corresponds to the mean power of each simulation in Figure 2, the right side corresponds to the respective MGC mean power. The thin solid lines are shown for 6-19, because MGC equals the global correlation 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 MGC always significantly improves over its global counterpart.

196 D_{CO}R_R. The advantage of MGC over its global counterpart M_CO_RR and H_HG is stark. For the nearly
 197 linear settings, MGC and M_CO_RR are essentially identical and significantly better than H_HG as the
 198 dimension increases. For the remaining nonlinear dependencies, MGC achieves superior power
 199 than H_HG and M_CO_RR for *all* functions, often by a significant margin. For the independent simulation,
 200 all tests yield powers at the significance level α , indicating no more false positives than expected
 201 according to the theory.

202 MGC dominates Global Counterparts

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

210 **Theoretical Consistency of MGC**

The formal testing scenario is as follows: we observe n pairs of observations, (x_i, y_i) , and we desire to know whether the x 's are independent of the 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 x and 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:

$$H_0 : f_{xy} = f_x f_y,$$

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

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

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

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

225 **Discovery of Dependency Across Scales**

226 A multiscale power map is a heatmap of powers for all neighborhood sizes, for a given joint distribu-
227 tion and sample size. Figure 4 provides the multiscale power maps for all 20 different scenarios for

228 a specified dimensionality (see caption for details), illustrating how the powers of local correlations
 229 change with respect to increasing neighborhood sizes.
 230 The multiscale power map sheds light into the intrinsic dependency structure. For nearly linear
 231 dependencies (1-5), the best neighborhood choice is always the largest scale, i.e., $k = l = n$.
 232 For all strongly nonlinear dependencies (6-19), Mgc almost always chooses a smaller scale in a
 233 distribution dependent fashion. Furthermore, similar dependencies have similar local correlation
 234 structure, and thus similar optimal scales. For example, quadratic (6) and W (7) are both polyno-
 235 mials of degree 2 with different coefficients, and their power maps are quite similar to each other.
 236 Similarly, (16) and (17) are the same trigonometry function (sine) with different periods, and they
 237 share a narrow range of significant local correlations. Both circle (11) and eclipse (12), as well
 238 as square (14) and diamond (15), are closely related functions, and have similar multiscale power
 239 maps. Note that for almost all simulations, there exist a large portion of adjacent local neighbor-
 240 hoods that are equally significant, which is an important observation that we use to approximate
 241 the optimal Mgc scale for real data.

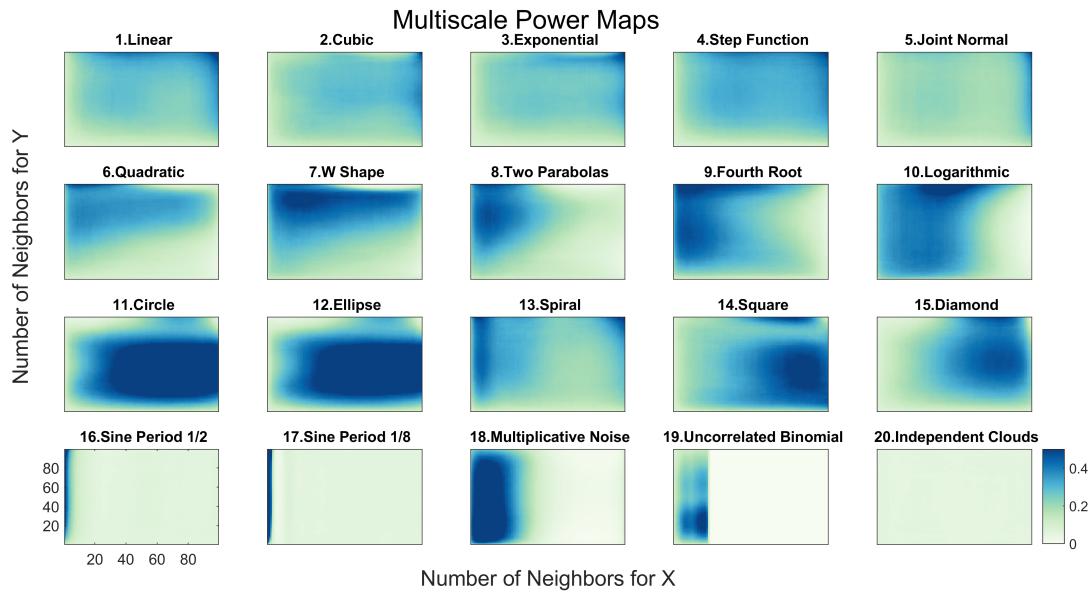


Figure 4: Influence of neighborhood size on testing power of local correlations at $\alpha = 0.05$. For each of the 20 panels, the abscissa denotes the number of neighbors for X (the scale increases from left to right), and the ordinate denotes the number of neighbors for Y (the scale increases from 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.

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

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

$$\beta(C^{mn}) = \beta(C^*) = \beta(C). \quad (4)$$

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

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

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

$$\beta(C^*) > \beta(C). \quad (5)$$

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

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

262 Real Data Experiments

263 Only Local Scales can Detect Dependence

264 Our first real data experiment investigates whether brain shape and disease status are indeed
265 dependent on one another. Previous investigations have linked major depressive disorder to the

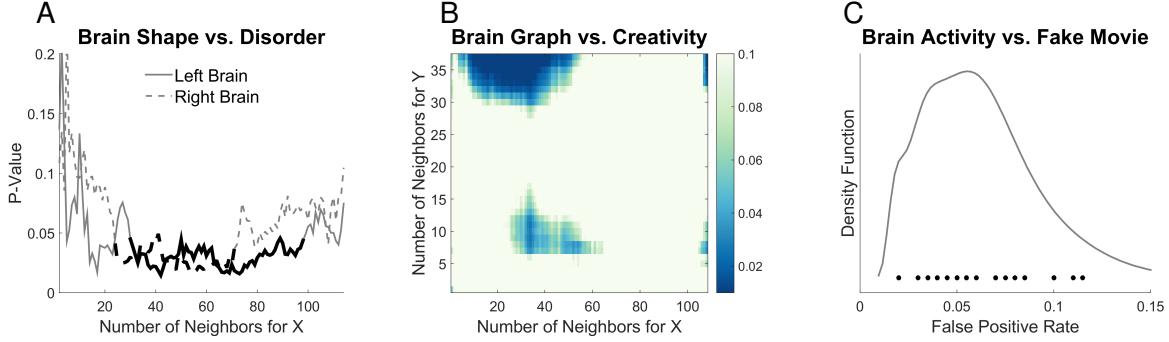


Figure 5: (A) Local correlation p-value curves with respect to $k = 2, \dots, 114$ at $l = 4$ for brain vs disease. Dark lines correspond to the largest region of significant scales. (B) Local correlation p-value heat map with respect to $k = 2, \dots, 109$ and $l = 2, \dots, 38$ for brain MIGRAINE vs 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.

266 hippocampus shape [3, 30], though global tests were unable to detect a statistically significant
 267 dependence structure at the $\alpha = 0.05$ level.

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

275 We consider two dependence tests, one for each hemisphere: is hippocampus shape independent
 276 of depressive state. Figure 5A provides the p-value curves for Mgc for $k = 2, \dots, n$ at $l = 4$ (we
 277 only show $l = 4$ because the other curves look similar). Many local scales yield significant p-
 278 values (around 0.01) for both hemispheres, whereas the global scale does not detect a significant
 279 dependence in either hemisphere. None of the previously proposed dependence tests under
 280 consideration (MANTEL, DCORR, MCORR, or HHG) were able to detect dependence for both (not shown).

281 **MGC can provide insight into the nature of the dependence structure**

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

294 **Mgc Does Not Inflate False Positive Rates**

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

308 To conduct this test, we must construct a distance matrix for brain region activity, and another for
309 the stimulus. For each brain region, we compute $a_{ij} = \|\mathbf{x}_{\cdot i} - \mathbf{x}_{\cdot j}\|_2^2$, for all (i, j) pairs, where $\mathbf{x}_{\cdot i}$
310 denotes the observation vector of all subjects at time-step i . For the stimulus, we similarly compute

311 the Euclidean distance between activity at all pairs of time-steps: $b_{ij} = \|y_i - y_j\|_2^2$. Note that the
312 distance matrices at different brain regions are distinct, but the stimulus is the same for all brain
313 regions during the same experiment.

314 For each data set, the above test is carried out for each brain region, and the false positive rates of
315 Mgc for each dataset are shown in Figure 5C. Mgc false positive rate is centered around the critical
316 level 0.05, as it should be. In contrast, standard methods for fMRI analysis, such as generalized
317 linear models, significantly increase or decrease the false discovery rates, depending on the data
318 [34, 37].

319 Discussion

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

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

338 Although our definition of local correlation coefficient is fast to implement, generally applicable
339 to any global correlation, and achieves good testing powers, there are multiple ways to combine
340 neighborhood information into a particular global correlation coefficient. So it is possible that

341 the testing performance may be further improved, by tailoring a different centering or ranking
342 scheme for a given global correlation, or by coming up with a different rank-truncated pairwise
343 comparison. Overall, a more thorough investigation on the finite-sample performance of Mgc, its
344 possible extensions, and other existing methods, are much needed in the future to enhance our
345 understanding of dependence discovery.

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

352 References

- 353 [1] K. Pearson, *Proceedings of the Royal Society of London* **58**, 240 (1895). [3](#), [4](#)
- 354 [2] M. Reimherr, D. Nicolae, *Statistical Science* **28**, 116 (2013). [3](#)
- 355 [3] Y. Park, C. Priebe, M. Miller, N. Mohan, K. Botteron, *Journal of Biomedicine and Biotechnol-*
356 *ogy* p. 694297 (2008). [3](#), [14](#)
- 357 [4] J. Maa, D. Pearl, R. Bartoszynski, *Annals of Statistics* **24**, 1069 (1996). [3](#), [4](#)
- 358 [5] W. K. Allard, G. Chen, M. Maggioni, *Applied and Computational Harmonic Analysis* **32**, 435
359 (2012). [3](#), [4](#)
- 360 [6] M. G. Kendall, *Rank Correlation Methods* (London: Griffin, 1970). [4](#)
- 361 [7] C. Spearman, *The American Journal of Psychology* **15**, 72 (1904). [4](#)
- 362 [8] N. Mantel, *Cancer Research* **27**, 209 (1967). [4](#), [26](#)
- 363 [9] G. Szekely, M. Rizzo, N. Bakirov, *Annals of Statistics* **35**, 2769 (2007). [4](#), [8](#), [20](#), [27](#), [38](#)
- 364 [10] G. Szekely, M. Rizzo, *Journal of Multivariate Analysis* **117**, 193 (2013). [4](#), [8](#), [28](#), [38](#)
- 365 [11] R. Lyons, *Annals of Probability* **41**, 3284 (2013). [4](#), [27](#)
- 366 [12] C. Stone, *Annals of Statistics* **4**, 595 (1977). [4](#)

- 367 [13] I. Daubechies, *Ten lectures on wavelets* (SIAM, 1992). 5
- 368 [14] B. Sarwar, G. Karypis, J. Konstan, J. Riedl, *ACM WebKDD 2000 Workshop* (2000). 5
- 369 [15] W. Torgerson, *Multidimensional Scaling: I. Theory and method* (Psychometrika, 1952). 5
- 370 [16] J. B. Tenenbaum, V. de Silva, J. C. Langford, *Science* **290**, 2319 (2000). 5
- 371 [17] V. de Silva, J. B. Tenenbaum, *Advances in Neural Information Processing Systems* **15**, 721
372 (2003). 5
- 373 [18] L. K. Saul, S. T. Roweis, *Science* **290**, 2323 (2000). 5
- 374 [19] S. T. Roweis, L. K. Saul, *Journal of Machine Learning Research* **4**, 119 (2003). 5
- 375 [20] M. Belkin, P. Niyogi, *Neural Computation* **15**, 1373 (2003). 5
- 376 [21] D. Barton, F. David, *Research Papers in Statistics*, Wiley, New York (1966). 5
- 377 [22] J. Friedman, L. Rafsky, *Annals of Statistics* **11**, 377 (1983).
- 378 [23] M. Schilling, *Journal of the American Statistical Association* **81**, 799 (1986). 5
- 379 [24] C. Shen, J. T. Vogelstein, C. Priebe, *Submitted* (2016). 5
- 380 [25] P. Good, *Permutation, Parametric, and Bootstrap Tests of Hypotheses* (Springer, 2005). 6
- 381 [26] N. Simon, R. Tibshirani, available at <http://arxiv.org/abs/1401.7645> (2012). 8, 16, 20
- 382 [27] M. Gorfine, R. Heller, Y. Heller, available at <http://ie.technion.ac.il/~gorfinm/files/science6.pdf>
383 (2012). 20, 29
- 384 [28] R. Heller, Y. Heller, M. Gorfine, *Biometrika* **100**, 503 (2013). 8, 29
- 385 [29] W. Rudin, *Real and Complex Analysis* (McGraw-Hill Education, 1986), third edn. 13
- 386 [30] J. Posener, et al., *American Journal of Psychiatry* **160**, 83 (2003). 14
- 387 [31] M. Beg, M. Miller, A. Trouv, L. Younes, *International journal of computer vision* **61**, 139 (2005).
388 14
- 389 [32] J. Adelstein, et al., *PLoS ONE* **6**, e27633 (2011). 15
- 390 [33] R. R. Costa, & McCrae, *Neo PI-R professional manual*, vol. 396 (1992). 15

- 391 [34] A. Eklund, M. Andersson, C. Josephson, M. Johannesson, H. Knutsson, *NeuroImage* **61**, 565
392 (2012). **15, 16**
- 393 [35] C. Craddock, *et al.*, *Frontiers in Neuroinformatics* **42** (2015). **15**
- 394 [36] S. Wang, C. E. Priebe, M. Maggioni, J. T. Vogelstein, *in preparation* (2016). **15**
- 395 [37] A. Eklund, T. Nichols, H. Knutsson, *arXiv* (2015). **16**
- 396 [38] A. Gretton, R. Herbrich, A. Smola, O. Bousquet, B. Scholkopf, *Journal of Machine Learning*
397 *Research* **6**, 2075 (2005). **16**
- 398 [39] A. Gretton, L. Gyorfi, *Journal of Machine Learning Research* **11**, 1391 (2010).
- 399 [40] A. Gretton, K. Borgwardt, M. Rasch, B. Scholkopf, A. Smola, *Journal of Machine Learning*
400 *Research* **13**, 723 (2012). **16**
- 401 [41] D. Sejdinovic, B. Sriperumbudur, A. Gretton, K. Fukumizu, *Annals of Statistics* **41**, 2263
402 (2013). **16**
- 403 [42] A. Ramdas, S. J. Reddi, B. Pcos, A. Singh, L. Wasserman, *29th AAAI Conference on Artifi-*
404 *cial Intelligence* (2015). **16**
- 405 [43] S. Dmcke, U. Mansmann, A. Tresch, *PLOS ONE* **9**, e107955 (2014). **16**
- 406 [44] D. Reshef, *et al.*, *Science* **334**, 1518 (2011). **16**

407 **Acknowledgment**

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

415 A Simulation Functions

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

420 For each sample $\mathbf{x} \in \mathbb{R}^{d_x}$, we denote $\mathbf{x}^d, d = 1, \dots, d_x$ as the d th dimension of \mathbf{x} . For the purpose
 421 of high-dimensional simulations, $w \in \mathbb{R}^{d_x}$ is a decaying vector with $w^d = 1/d$ for each d , such
 422 that $w^\top \mathbf{x}$ is a 1-dimensional weighted summation of all dimensions of \mathbf{x} , which equals \mathbf{x} if $d_x = 1$.
 423 Furthermore, \mathcal{U} denotes the uniform distribution, \mathcal{B} denotes the Bernoulli distribution, \mathcal{N} denotes
 424 the normal distribution, u and v represent realizations from some auxiliary random variables, c is
 425 a scalar constant to control the noise level (which equals 1 for 1-dimensional simulations and 0
 426 otherwise), and ϵ is sampled from an independent standard normal distribution unless mentioned
 427 otherwise.

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

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

$$\begin{aligned}\mathbf{x} &\sim \mathcal{U}(-1, 1)^{d_x}, \\ \mathbf{y} &= w^\top \mathbf{x} + c\epsilon.\end{aligned}$$

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

$$\begin{aligned}\mathbf{x} &\sim \mathcal{U}(-1, 1)^{d_x}, \\ \mathbf{y} &= 128(w^\top \mathbf{x} - \frac{1}{3})^3 + 48(w^\top \mathbf{x} - \frac{1}{3})^2 - 12(w^\top \mathbf{x} - \frac{1}{3}) + 80c\epsilon.\end{aligned}$$

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

$$\begin{aligned}\mathbf{x} &\sim \mathcal{U}(0, 3)^{d_x}, \\ \mathbf{y} &= \exp(w^\top \mathbf{x}) + 10c\epsilon.\end{aligned}$$

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

$$\begin{aligned}\mathbf{x} &\sim \mathcal{U}(-1, 1)^{d_x}, \\ \mathbf{y} &= \mathbf{I}(w^\top \mathbf{x} > 0) + \epsilon,\end{aligned}$$

430 where \mathbf{I} is the indicator function.

5. Joint normal $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}^{d_x}$: 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 $(u, v) \sim \mathcal{N}(0, \Sigma)$, $\epsilon \sim \mathcal{N}(0, I_{d_x})$,

$$\mathbf{x} = u,$$

$$\mathbf{y} = v + 0.5c\epsilon.$$

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

$$\mathbf{x} \sim \mathcal{U}(-1, 1)^{d_x},$$

$$\mathbf{y} = (w^\top \mathbf{x})^2 + 0.5c\epsilon.$$

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

$$\mathbf{x} \sim \mathcal{U}(-1, 1)^{d_x},$$

$$\mathbf{y} = 4 \left[\left((w^\top \mathbf{x})^2 - \frac{1}{2} \right)^2 + w^\top u / 500 \right] + 0.5c\epsilon.$$

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

$$\mathbf{x} \sim \mathcal{U}(-1, 1)^{d_x},$$

$$\mathbf{y} = \left((w^\top \mathbf{x})^2 + 2c\epsilon \right) \cdot (u - \frac{1}{2}).$$

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

$$\mathbf{x} \sim \mathcal{U}(-1, 1)^{d_x},$$

$$\mathbf{y} = |w^\top \mathbf{x}|^{\frac{1}{4}} + \frac{c}{4}\epsilon.$$

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

$$\mathbf{x} \sim \mathcal{N}(0, I_{d_x}),$$

$$\mathbf{y}^d = 2\log(\mathbf{x}^d) + 3c\epsilon^d,$$

431 **for** $d = 1, \dots, d_x$.

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

$$\mathbf{x}^d = r \left(\sin(\pi u^{d+1}) \prod_{j=1}^d \cos(\pi u^j) + 0.4\epsilon^d \right) \text{ for } d = 1, \dots, d_x - 1,$$

$$\mathbf{x}^{d_x} = r \left(\prod_{j=1}^{d_x} \cos(\pi u^j) + 0.4\epsilon^{d_x} \right),$$

$$\mathbf{y} = \sin(\pi u^1).$$

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

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

$$\mathbf{x}^d = u \sin(\pi u) [\cos(\pi u)]^d \text{ for } d = 1, \dots, d_x - 1,$$

$$\mathbf{x}^{d_x} = u [\cos(\pi u)]^{d_x},$$

$$\mathbf{y} = u \sin(\pi u) + 0.4(d_x - 1)\epsilon.$$

14. Square $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}^{d_x}$: 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

$$\mathbf{x}^d = u \cos \theta + v \sin \theta + 0.05d_x \epsilon^d,$$

$$\mathbf{y}^d = -u \sin \theta + v \cos \theta,$$

⁴³³ for $d = 1, \dots, d_x$.

⁴³⁴ 15. Diamond $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}^{d_x}$: Same as above except $\theta = -\frac{\pi}{4}$.

16. Sine Period 1/2 $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$: $u \sim \mathcal{U}(-1, 1)$, $v \sim \mathcal{N}(0, 1)^{d_x}$, $\theta = 4\pi$,

$$\mathbf{x}^d = u + 0.02d_x v^d \text{ for } d = 1, \dots, d_x,$$

$$\mathbf{y} = \sin(\theta x) + c\epsilon.$$

⁴³⁵ 17. Sine Period 1/8 $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$: Same as above except $\theta = 16\pi$ and the noise is changed
to $0.5c\epsilon$.

18. Multiplicative Noise $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}^{d_x}$: $u \sim \mathcal{N}(0, I_{d_x})$, $\epsilon \sim \mathcal{N}(0, I_{d_x})$,

$$\mathbf{x} \sim \mathcal{N}(0, I_{d_x}),$$

$$\mathbf{y}^d = u^d \mathbf{x}^d + 0.5\epsilon^d,$$

⁴³⁷ for $d = 1, \dots, d_x$.

19. Uncorrelated Binomial $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}$: $u \sim \mathcal{B}(0.5)$,

$$\mathbf{x} \sim \mathcal{B}(0.5)^{d_x},$$

$$\mathbf{y} = (2u - 1)w^\top \mathbf{x} + 0.6\epsilon.$$

20. Independent Clouds $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}^{d_x}$: 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

$$\mathbf{x} = u/3 + 2u' - 1,$$

$$\mathbf{y} = v/3 + 2v' - 1.$$

438 For each distribution, x and y are clearly dependent except (20); for some settings (11-15) they
439 are conditionally independent upon conditioning on the respective auxiliary variables, while for
440 others they are "directly" dependent. Then we can independently generate (x_i, y_i) from (x, y) for
441 $i = 1, \dots, n$, set $X = [x_1, \dots, x_n] \in \mathbb{R}^{d_x \times n}$ and $Y = [y_1, \dots, y_n] \in \mathbb{R}^{d_y \times n}$, and calculate local /
442 global correlations for the sample data. A visualization of each dependency is shown in Figure A1.

443 For the increasing dimension simulation in the main paper, we always set $c = 0$ and $n = 100$,
444 with d_x increasing while $d_y = d_x$ for type 5, 10, 14, 15, 18, 20 and $d_y = 1$ otherwise. The decaying
445 vector w is utilized for $d_x > 1$ to treat higher dimensions as small perturbations, which creates a
446 meaningful setting for testing power comparison. The powers of all three Mgc implementations in
447 this setting are provided in Figure A2, where we denote Mgc_D as the Mgc for Dcorr, Mgc_M as the
448 Mgc for Mcorr, Mgc_P as the Mgc for Mantel.

449 B Supplementary Figures

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

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

459 C Dependence Measures

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

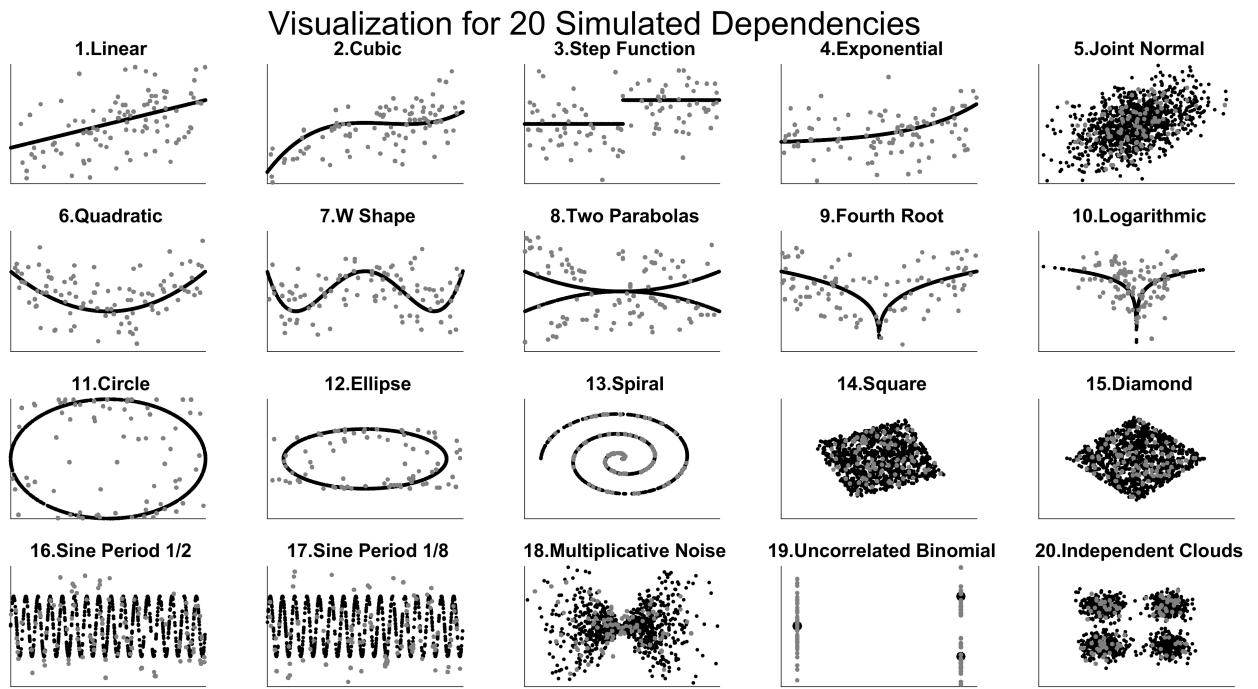


Figure A1: Visualization of the 20 dependencies for 1-dimensional simulations. The blue points are generated with noise ($c=1$) at $n = 100$ to show the actual sample data in testing, and the red points are generated without noise at $n = 1000$ to highlight each underlying dependency.

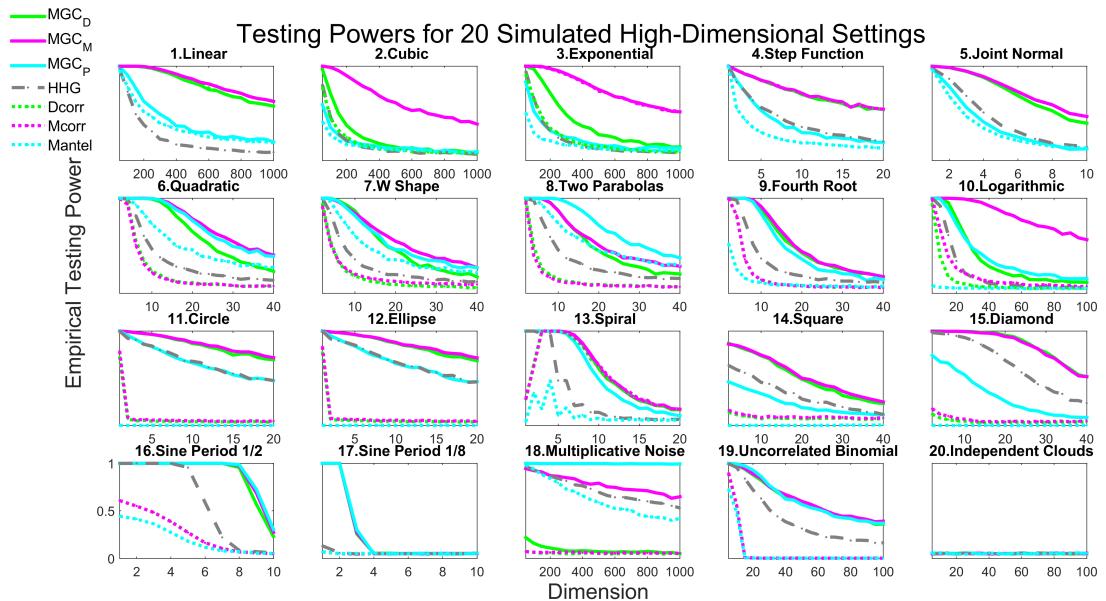


Figure A2: Same as Figure 2 but includes all three different Mgc implementations.

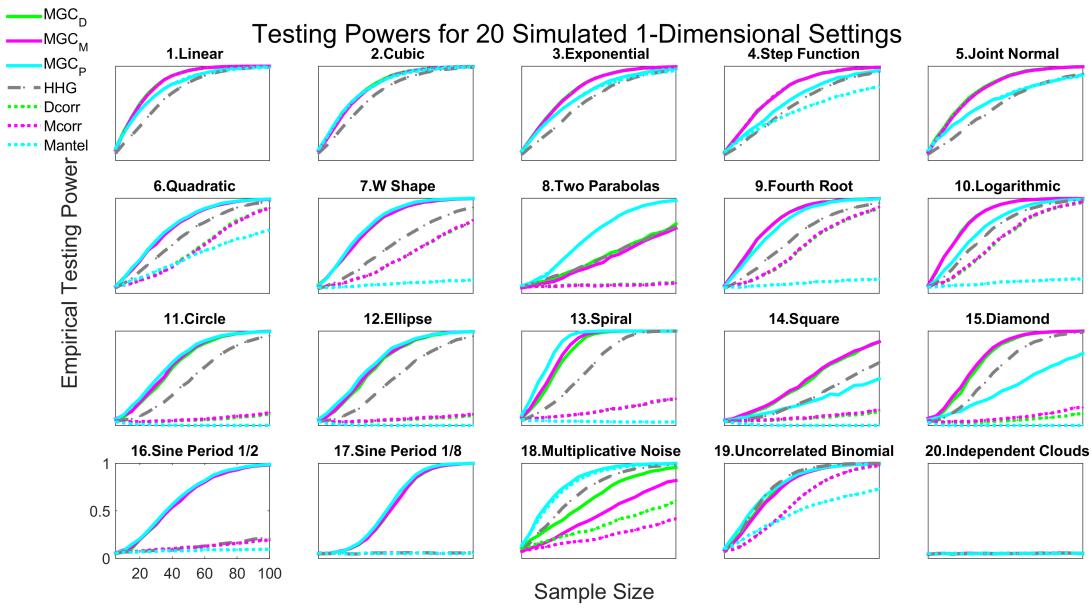


Figure A3: Powers of different methods for 20 different 1-dimensional dependence structures, estimated by the empirical distributions of the test statistics under the null and the alternative on the basis of 10,000 Monte-Carlo replicates. 2,000 additional MC replicates are used for optimal scale estimation for Mgc . Each panel shows empirical testing power on the abscissa 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.

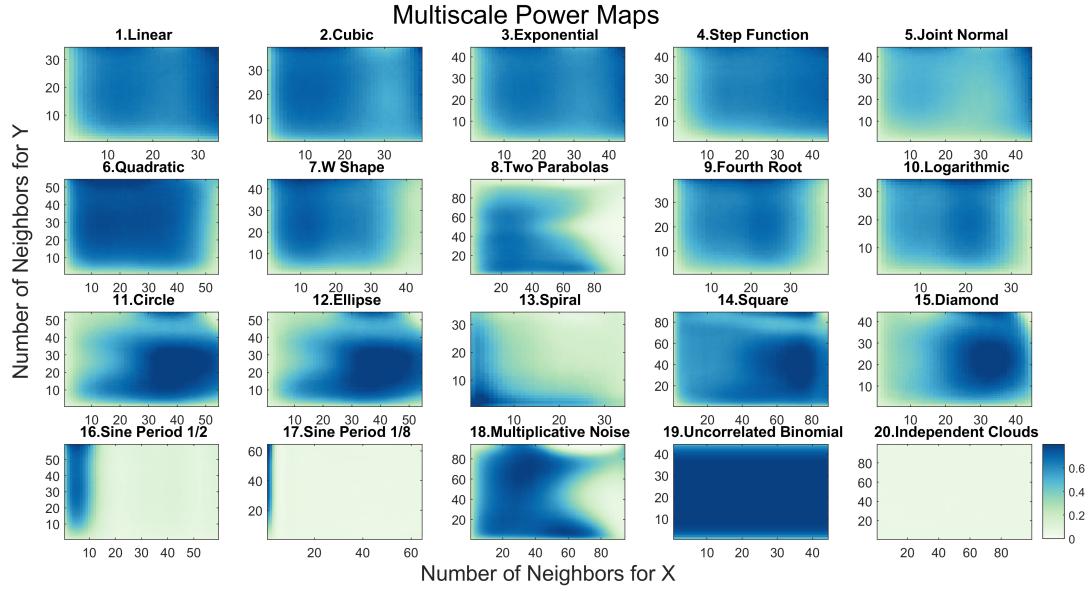


Figure A4: Influence of neighborhood size on testing power of local correlations. For each simulation, the dimension is 1, and the sample size is determined by the first sample size n for MGC to have powers exceeding the threshold 0.8.

463 C.1 (Global) MANTEL Test

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

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

465 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
466 by the permutation test.

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

472 C.2 (Global) Distance Correlation

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

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

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

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

475 and the sample distance correlation equals

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

476 It is shown in [9] that as $n \rightarrow \infty$, $Dcorr(X, Y) \rightarrow Dcorr(\mathbf{x}, \mathbf{y}) \geq 0$, where $Dcorr(\mathbf{x}, \mathbf{y})$ denotes
 477 the population distance correlation between the underlying random variable \mathbf{x} and \mathbf{y} . The pop-
 478 ulation distance correlation is defined by the characteristic functions, which is 0 if and only if \mathbf{x}
 479 and \mathbf{y} are independent. Thus the sample distance correlation is a consistent statistic for testing
 480 independence, i.e., the testing power $\beta_\alpha(Dcorr(X, Y))$ converges to 1 as n increases, at any type
 481 1 error level α . Note that all of $dcov$, $dvar$, $Dcorr$ are always non-negative; and the consistency
 482 result assumes finite second moments of \mathbf{x} and \mathbf{y} , which holds for a family of metrics not limited
 483 to the Euclidean distance [11]. Also note that the $Dcorr$ above is actually the square of distance
 484 correlation in [9], but for ease of presentation the square naming is dropped here.

485 Alternatively, calculating the distance covariance by $A = H\tilde{A}$ and $B = \tilde{B}H$ gives the same statis-
 486 tic as in Equation 2, i.e., instead of using doubly centered distance matrices, it is the same to
 487 singly center one distance matrix by row and the other distance matrix by column. Then $Dcorr$ by
 488 singly centered distance matrices has the same testing power as the original $Dcorr$, because dis-
 489 tance covariance is equivalent to distance correlation in the permutation test (note that the actual
 490 $Dcorr$ statistic by single centering is different from the original $Dcorr$, as using single centering
 491 changes the distance variances).

492 In our implementation of global / local $Dcorr$, we always use singly centered distance matrices
 493 rather than doubly centered distance matrices. Although they are equivalent for the testing power

494 of global `Dcorr`, our alternative implementation improves the testing power of local `Dcorr` and
 495 `Mcc`. This is because the ranking information of \tilde{A} and \tilde{B} are better preserved in singly centered
 496 distance matrices, so that `Mcc` is more effective in excluding far-away points that exhibit insignificant
 497 dependency. This applies to `Mcorr` as well.

498 C.3 (Global) Modified Distance Correlation

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

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

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

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

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

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

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

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

510 It is shown in [10] that $Mcorr(X, Y)$ is an unbiased estimator of the population distance correlation
 511 $Dcorr(x, y)$ for all d_x, d_y, n ; and `Mcorr` is approximately normal even if $d_x, d_y \rightarrow \infty$. Thus it is a
 512 consistent statistic for testing independence, but may work better than `Dcorr` under high-dimension
 513 dependencies.

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

518 **C.4 Multiscale Graph Correlations (MGC)**

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

522 In particular, **MANTEL** sets a_{ij} and b_{ij} as the respective entry of \tilde{A} and \tilde{B} (the Euclidean distances).
523 **Dcorr** lets a_{ij} and b_{ij} be the respective matrix entry of A and B (the doubly centered distance
524 matrices), then the sample means \bar{a}, \bar{b} are automatically 0. **Mcorr** slightly modifies a_{ij} and b_{ij} of
525 **Dcorr** to adjust their high-dimensional bias. As discussed already, our version of MGC_M is based
526 on single centering throughout: we take $a_{ij} = b_{ij} = 0$ when $i = j$, otherwise set a_{ij} as the matrix
527 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
528 by Equation 3.

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

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

539 **C.5 Heller, Heller & Gorfine (HHG)**

540 The **HHG** statistic applies Pearson's chi-square test to ranks of distances within each column, and is
541 shown to be better than many global tests including **Dcorr** under common nonlinear dependencies
542 in [27, 28]. Like **Dcorr** and **Mcorr**, **HHG** is distance-based and consistent, but not in the form of the
543 generalized correlation coefficient; and like our **MGC**, it makes use of the rank information, but in a
544 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 H_{HG} statistic is defined as

$$\text{H}_{\text{HG}}(X, Y) = \sum_{i=1, j \neq i}^n \frac{(n-2)(H_{12}(i, j)H_{21}(i, j) - H_{11}(i, j)H_{22}(i, j))^2}{H_{1.}(i, j)H_{2.}(i, j) - H_{.1}(i, j)H_{.2}(i, j)},$$

- 545 where $H_{1.} = H_{11} + H_{12}$, $H_{2.} = H_{21} + H_{22}$, $H_{.1} = H_{11} + H_{21}$, and $H_{.2} = H_{12} + H_{22}$. It is clear
 546 that H_{HG} is structurally different from **Dcorr** / **Mcorr** / **MANTEL**, cannot be conveniently expressed by
 547 Equation 1, and there is no direct extension of local correlation to H_{HG} .
 548 The permutation test using the H_{HG} statistic is consistent against all dependent alternatives. In
 549 our numerical simulations, H_{HG} falls a bit short when testing against high-dimensional and noisy
 550 linear dependencies, but is often more advantageous than global correlations under nonlinear
 551 dependencies, which makes it a strong competitor in general.

552 D Mgc Algorithms and Testing Procedures

- 553 In this section we elaborate on the algorithms for computing local correlation and **Mgc**, as well as
 554 their testing procedures in simulations and real data experiment.
 555 Five algorithms are presented in section D.1: given the choice of a global correlation coefficient,
 556 algorithm 1 computes one local correlation coefficient at a given (k, l) ; then algorithm 2 shows
 557 how to compute all local correlations simultaneously; algorithm 3 computes the p-values of all
 558 local correlation by the random permutation test; algorithm 4 approximates the optimal scale for
 559 **Mgc** based on the p-values of all local correlations, and outputs the approximated p-value of **Mgc**;
 560 algorithm 5 estimates the testing powers of all local statistics based on a given joint distribution
 561 or multiple pairs of data, which can be used to more accurately estimate the optimal scale for

562 MGC when the underlying model is known or training data are given. More detailed discussions
563 regarding the optimal scale approximation is offered in section D.2.

564 D.1 Algorithms

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

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

573 To facilitate the optimal scale estimation, algorithm 2 provides a fast method to compute all lo-
574 cal correlations in $O(n^2)$. An important observation is that each product $a_{ij}b_{ij}$ is included in C^{kl}
575 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
576 $i, j = 1, \dots, n$, and add the product simultaneously to all C^{kl} whose scales are no more than
577 $(R(a_{ij}), R(b_{ij}))$. However, accessing and adding multiple C^{kl} at the same time is not computa-
578 tionally efficient; instead, for each product, we only add it to C^{kl} at $(k, l) = (R(a_{ij}), R(b_{ij}))$ (so only one
579 local scale is accessed for each operation), iterate through all products for $i, j = 1, \dots, n$, then add
580 up adjacent C^{kl} for $k, l = 1, \dots, n$. Thus all local correlations can be computed in $O(n^2)$, which
581 has the same running time complexity as the global distance correlation. There are two additional
582 overheads: sorting the distance matrices column-wise takes $O(n^2 \log n)$, and properly centering
583 the distance matrices takes $O(n^2)$.

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

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

591 $\mathcal{K} = \{n\}$; and similarly determine the set of valid columns \mathcal{L} . Once \mathcal{K} and \mathcal{L} are determined, the
592 optimal scale (k^*, l^*) is found by the scale that minimizes the p-value within $\{p_{kl}, k \in \mathcal{K}, l \in \mathcal{L}\}$.
593 Clearly if the majority p-values of all local correlations are less than α , then $\mathcal{K} = \mathcal{L} = \{1, \dots, n\}$,
594 and the optimal scale equals the scale that minimizes the p-values among all local correlations;
595 if there is no valid rows and columns, then Mgc takes the largest scale and equals the global
596 correlation. Note that the actual algorithm is a simpler version of the above description: instead of
597 considering all possible sets of rows and check the validity, we limit the check to the most likely set
598 of rows, by first looking for the row scale of the smallest p-value, then including all adjacent rows
599 whose minimal p-value on the row is no larger than α ; similarly for the set of columns.

600 Algorithm 5 computes the testing powers of all local correlations by repeated simulating samples
601 generated from the joint distribution f_{xy} . Sample data under the null and the alternative are re-
602 peatedly generated for r Monte-Carlo replicates, and algorithm 2 is applied to compute the sample
603 local correlations under the null and the alternative. Then the testing power at each local corre-
604 lation can be estimated, and the Mgc optimal scale can be found by maximizing the powers. This
605 algorithm is also applicable if there exists multiple pairs of data with unknown model but similar
606 dependency structure, then the alternative statistic can be computed from each data pair while
607 the null statistic can be computed from each data pair under permutation. The running time is
608 $O(rn^2 \log n)$.

609 D.2 Discussions of Optimal Scale Estimation

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

617 If there is only one pair of data (X, Y) with unknown distributions, we have to approximate the
618 optimal scale by algorithm 4. It makes use of Bonferroni correction to separately verify the set of
619 rows and columns, which guarantees the false positive rate to be no higher than α ; otherwise the
620 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) .

```
1: function LOCALCORR( $\tilde{A}, \tilde{B}, k, l$ )
2:   initialize  $C^{kl}, V_k^A, V_l^B, E_k^A, E_l^B$  as 0.
3:   for  $Z := A, B$  do  $R^Z = \text{SORT}(\tilde{Z})$  end for            $\triangleright$  column-wise sorting and assume no ties
4:   for  $Z := A, B$  do  $Z = \text{CENTER}(\tilde{Z})$  end for        $\triangleright$  proper centering of the distance matrices
5:   for  $i, j = 1, \dots, n$  do
6:      $C^{kl} = C^{kl} + A_{ij}B_{ij}\mathbf{I}(R_{ij}^A \leq k)\mathbf{I}(R_{ij}^B \leq l)$            $\triangleright$  store local distance covariance
7:      $V_k^A = V_k^A + A_{ij}^2\mathbf{I}(R_{ij}^A \leq k)$            $\triangleright$  store local distance variance for  $X$ 
8:      $V_l^B = V_l^B + B_{ij}^2\mathbf{I}(R_{ij}^B \leq l)$            $\triangleright$  store local distance variance for  $Y$ 
9:      $E_k^A = E_k^A + A_{ij}\mathbf{I}(R_{ij}^A \leq k)$            $\triangleright$  store the sample means
10:     $E_l^B = E_l^B + B_{ij}\mathbf{I}(R_{ij}^B \leq l)$ 
11:   end for
12:    $C^{kl} = (C^{kl} - E_k^A E_l^B / n^2) / \sqrt{(V_k^A - E_k^{A2} / n^2)(V_l^B - E_l^{B2} / n^2)}$        $\triangleright$  normalize the local covariances
13: 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 $k, l = 1, \dots, n$.

```

1: function LOCALCORR( $\tilde{A}, \tilde{B}$ )
2:   initialize  $C$  as a zero matrix of size  $n \times n$ ;  $V^A, V^B, E^A, E^B$  as zero vectors of size  $n$ .
3:   for  $Z := A, B$  do  $R^Z = \text{SORT}(\tilde{Z})$  end for
4:   for  $Z := A, B$  do  $Z = \text{CENTER}(\tilde{Z})$  end for
5:   for  $i, j = 1, \dots, n$  do
6:      $k = R_{ij}^A$ 
7:      $l = R_{ij}^B$ 
8:      $C^{kl} = C^{kl} + A_{ij}B_{ij}$ 
9:      $V_k^A = V_k^A + A_{ij}^2$ 
10:     $V_l^B = V_l^B + B_{ij}^2$ 
11:     $E_k^A = E_k^A + A_{ij}$ 
12:     $E_l^B = E_l^B + B_{ij}$ 
13:   end for
      ▷ the next two for loops with respect to the scales guarantee the computation of all local
      covariance / variance in  $O(n^2)$ 
14:   for  $k = 1, \dots, n - 1$  do
15:      $C^{1,k+1} = C^{1,k} + C^{1,k+1}$ 
16:      $C^{k+1,1} = C^{k+1,1} + C^{k+1,1}$ 
17:     for  $Z := A, B$  do  $V_{k+1}^Z = V_k^Z + V_{k+1}^Z$  end for
18:     for  $Z := A, B$  do  $E_{k+1}^Z = E_k^Z + E_{k+1}^Z$  end for
19:   end for
20:   for  $k, l = 1, \dots, n - 1$  do
21:      $C^{k+1,l+1} = C^{k+1,l} + C^{k,l+1} + C^{k+1,l+1} - C^{k,l}$ 
22:   end for
23:   for  $k, l = 1, \dots, n$  do                                ▷ normalize all local covariances
24:      $C^{kl} = (C^{kl} - E_k^A E_l^B / n^2) / \sqrt{(V_k^A - E_k^A)^2 / n^2 (V_l^B - E_l^B)^2 / n^2}$ 
25:   end for
26: 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.

```
1: function PERMUTATIONTEST( $\tilde{A}, \tilde{B}, r$ )
2:    $C^{kl} = \text{LOCALCORR}(\tilde{A}, \tilde{B})$                                  $\triangleright$  calculate the observed local correlations
3:   for  $j = 1, \dots, r$  do
4:      $\pi = \text{RANDPERM}(n)$                                           $\triangleright$  generate a random permutation of size  $n$ 
5:      $C_0^{kl}[j] = \text{LOCALCORR}(\tilde{A}, \tilde{B}(\pi, \pi))$            $\triangleright$  calculate the permuted test statistics
6:   end for
7:   for  $k, l = 1, \dots, n$  do
8:      $P_{kl} = \sum_{j=1}^r (C^{kl} < C_0^{kl}[j]) / r$                    $\triangleright$  get the p-value at each local scale
9:   end for
10:  end function
```

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

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

635 Note that it is tempting to directly use the optimal scale that minimizes all local p-values without
636 the validation by algorithm 4, or generate random samples based on the given data pair and use
637 algorithm 5 by bootstrap. However, both approaches are biased such that the false positive rate will
638 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 .

```
1: function MGCSCALEVERIFY( $P, \alpha$ )
2:    $\mathcal{K} = \text{VERIFYRow}(P, \alpha)$                                  $\triangleright$  search for a set of valid row indices
3:    $\mathcal{L} = \text{VERIFYRow}(P^T, \alpha)$                              $\triangleright$  search for a set of valid column indices
4:    $[k^*, l^*] = \arg \min_{\{k \in \mathcal{K}, l \in \mathcal{L}\}} P_{kl}$            $\triangleright$  find the optimal scale within the valid range
5:    $p = P_{k^*l^*}$ 
6: end function
```

Input: Same as MGCSCALEVERIFY.

Output: The indices of valid rows.

```
1: function VERIFYRow( $P, \alpha$ )
2:   initialize  $\mathcal{K}$  as an empty set
3:    $[k^*, l^*] = \arg \min_{k,l} \{P_{kl}, k, l = 2, \dots, n\}$ 
4:   for  $k = k^*, \dots, 2$  do                                 $\triangleright$  check all row scales no larger than  $k^*$ 
5:     if  $\min\{P_{kl}, l = 2, \dots, n\} > \alpha$  then
6:       break
7:     end if
8:      $\mathcal{K} = [k, \mathcal{K}]$ 
9:   end for
10:  for  $k = k^* + 1, \dots, m$  do                          $\triangleright$  check all row scales larger than  $k^*$ 
11:    if  $\min\{P_{kl}, l = 2, \dots, n\} > \alpha$  then
12:      break
13:    end if
14:     $\mathcal{K} = \{\mathcal{K}, k\}$ 
15:  end for
16:  if  $\text{MEDIAN}(P_{kl}, k \in \mathcal{K}, l = 2, \dots, n) > \alpha * \frac{|\mathcal{K}|}{n-1}$  then
17:     $\mathcal{K} = \{n\}$             $\triangleright$  take the largest scale if the median p-value is not sufficiently small
18:  end if
19: 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 $(k^*, l^*) \in \mathbb{R} \times \mathbb{R}$.

```
1: function TESTINGPOWERS( $f_{xy}, n, r, \alpha$ )
2:   for  $j = 1, \dots, r$  do
3:     for  $i := [n]$  do  $(X_i^1, Y_i^1) \stackrel{iid}{\sim} f_{xy}$  end for            $\triangleright$  generate dependent samples
4:     for  $i := [n]$  do  $X_i^0 \stackrel{iid}{\sim} f_x$  end for                    $\triangleright$  generate independent samples
5:     for  $i := [n]$  do  $Y_i^0 \stackrel{iid}{\sim} f_y$  end for
6:     for  $Z := A, B$  do  $\tilde{Z}_1 = \text{DIST}(Z_1)$  end for     $\triangleright$  the distance matrices under the alternative
7:     for  $Z := A, B$  do  $\tilde{Z}_0 = \text{DIST}(Z_0)$  end for       $\triangleright$  the distance matrices under the null
8:      $C_1^{kl}[j] = \text{LOCALCORR}(\tilde{A}_1, \tilde{B}_1)$         $\triangleright$  calculate all local correlations under the alternative
9:      $C_0^{kl}[j] = \text{LOCALCORR}(\tilde{A}_0, \tilde{B}_0)$         $\triangleright$  calculate all local correlations under the null
10:   end for
11:   for  $k, l = 1, \dots, n$  do
12:      $c_\alpha = \text{CDF}_{1-\alpha}(C_{kl}^0[j], j \in [r])$            $\triangleright$  get the critical value by the empirical cumulative
           distribution under the null at each scale
13:      $\beta_\alpha^{kl} = \sum_{j=1}^r (C_{kl}^1[j] > c_\alpha) / r$          $\triangleright$  estimate the power
14:   end for
15:    $(k^*, l^*) = \arg \max(\beta_\alpha^{kl})$                        $\triangleright$  find the optimal local scale
16: end function
```

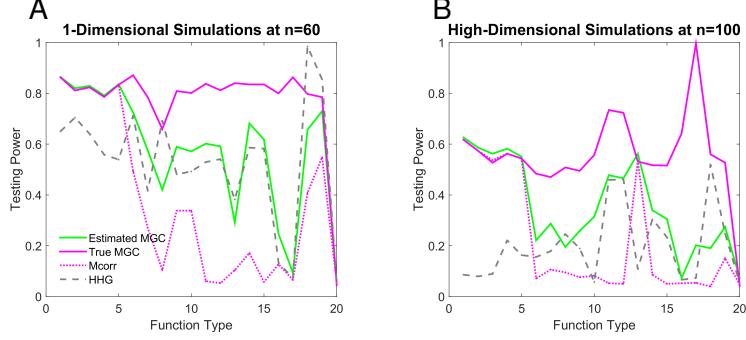


Figure A5: Comparing estimated MGC power to true MGC power, for the 1-dimensional and high-dimensional simulations. (A) 1-dimensional simulations, where $d_x = 1$ and the sample size is chosen by the power threshold 0.8 as in Figure A4. (B) High-dimensional simulations, where $n = 100$ and the dimension 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 M_{corr} and H_{HG}, combines the better performance of the two benchmarks, is quite close to the true MGC power, and does not inflate false signals.

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

644 E Proofs

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

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

$$\beta(C^*) = \max_{\mathbf{x}, l} \{\beta(C^{kl})\} \geq \beta(C), \quad (6)$$

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

648 Therefore $\beta(C_t^*) \rightarrow 1$ for all f_{xy} in \mathcal{F}_t . In particular, M_{GCD} and M_{GCM} are consistent against all alter-
 649 native of finite second moments, because D_{corr} and M_{corr} are consistent against all alternatives
 650 of finite second moments by [9, 10]. \square

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

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

652 Thus the optimal scale for MGC is the global scale for linearly dependent data.

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

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

$$1 = C(X, Y) \geq C(X, YQ) \quad (8)$$

657 for any permutation matrix Q , where the equality holds if and only if X is a scalar multiple of YQ .

658 It follows that the p-value of C is 0, which is at the minimal.

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

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

$$\beta(C^*) > \beta(C). \quad (9)$$

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

664 *Proof.* We give a simple discrete example of f_{xy} at $n = 7$, such that the p-value of MGC_M is strictly
665 lower than the p-value of MCORR.

Suppose under the alternative, each pair of observation (x, y) is sampled as follows:

$$\begin{aligned} x &\in \{-1, -\frac{2}{3}, -\frac{1}{3}, 0, \frac{1}{3}, \frac{2}{3}, 1\} \text{ without replacement,} \\ y &= x^2, \end{aligned}$$

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

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

669 case k is bounded above by $n = 7$ while l is bounded above by 4 due the the repeating points in
670 Y .

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

673 Note that we can always consider sample points in $[-1, 1]$ for X , increase n and reach the same
674 conclusion with more significant p-values; but the computation of all possible permuted test statis-
675 tics becomes more time-consuming as n increases. The same conclusion also holds for Mgc_D and
676 Mgc_P using the same example. □