UNIVERSITY OF TARTU
Faculty of Science and Technology
Institute of Computer Science
Computer Science Curriculum

Savelii Vorontcov

Effects of Data Distributions and Distance Measures in Representational Similarity Analysis

Bachelor's Thesis (9 ECTS)

Supervisor(s): Raul Vicente Zafra, PhD

Tartu 2024

Effects of Data Distributions and Distance Measures in Representational Similarity Analysis

Abstract:

Representational Similarity Analysis (RSA) is an analysis technique often used in Computational Neuroscience. In the context of measured brain data, it allows us to get representations of various stimuli in the brain and compare these representations between different brain regions, between different species and different modalities of measured data. Comparing data gathered using different modalities is a particularly challenging task in neuroscience because it would require us to perform mapping between modalities in question, which in some cases can be ill-defined. The task of comparing brain-activity data with computational or behavioral models might be even more challenging. RSA addresses all mentioned issues.

One question that arises is how much linear correlations get distorted after applying RSA, which is addressed in this study. We consider in detail how correlations between two arrays of underlying data influence correlations between corresponding representations after applying RSA.

Results show that in all cases rank correlations in processed data are lower or equal than linear correlations in initial data. This effect is particularly noticeable for intermediate values of linear correlation (0.3-0.6). The implication is that RSA underestimates linear correlations captured by underlying data. In other words, correlations in initial data tend to be higher or equal compared to the ones calculated through RSA. Since some brain studies involving RSA make conclusions about dependence structure in data based on correlations between calculated representations, it is be useful to know how the real correlation structure gets distorted. In a broader perspective, it might influence what we consider a "high" or "low" correlation in the context of RSA and when correlation is significant enough for us to conclude that two arrays of data are interdependent.

Keywords:
Representational similarity analysis, generation of correlated data, probability distributions, distance measures

CERCS: P170 Computer science, numerical analysis, systems, control

Andmete jäotuste ja kauguste mõõdute mõjud esindusliku sarnasuse analüüsis

Lühikokkuvõte:

Esindusliku sarnasuse analüüsis (RSA) on analüüsimise töörist sageli kasutatav arvutuslikus neuroteaduses. Ajust mõõdetud andmete kontekstis see annab meile võimaluse saada erinevate stiimulite esindusi ajus ning võrrelda neid esindusi erinevate aju alade, erinevate loomuliikide ja erinevate modaalsuste vahel. Erinevate modaalsuste kaudu kogutud andmete võrdlemine on eriti raske neuroteaduse ülesanne sest see vajab meilt mingil viisil seostada neid moodalsusi omavahel, mis mõnikord võib olla ebatriviaalne ülesanne. Ajust kogutud andmete arvutusliku või käitumise mudelitega võib olla veelgi raskem. RSA tegeleb mainitud probleemidega.

Üks küsimus mis tekkib on kui palju lineaarsed korrelatsioonid muutuvad pärast RSA kasutamist, mis on selles tööd uuritud. Meie vaatleme detailselt kuidas korrelatsioonid kahe algmassiivi vahel mõjutavad korrelatsioone vastavate esinduste vahel pärast RSA kasutamist.

Tulemused näitavad, et kõigil juthudel järgu korrelatsioonid töödeldud andmetel on väiksem või samad kui lineaarsed korrelatsioonid algandmetes. See effekt on eritiv nähtav kui lineaarne korrelatsioon kuulub vahepealsete väärtuste hulka (0.3-0.6). Järeldus on see, et RSA hindab alla lineaarseid korrelatsioone algandmetes. Teistes sõnades, korrelatsioonid algandmetes on tavaliselt suurem või samad võrreldes RSA kaudu arvutatud korrelatsioonidega. Sellepärast, et mõned aju uuritused mis kasutavad RSA teevad järeldusi sõltuvuse struktuurist võrreldes arvutatud representatsioone, teadmine sellest, kuidas tegelik korrelatsioonide struktuur moonutatakse oleks väga kasulik. Laiemas perspektiivis, see võib mõjutada meie arusaamist sellest, mis on suur või väike korrelatsioon RSA kontekstis ning millal korrelatsioon on piisavalt märkimisväärne selleks, et järeldada et kaks andmete massiivi are omavahel sõltuvad.

Võtmesõnad:
esinduslik sarnasuse analüüs, korreleeritud andmete generatsioon, tõenäosusjäotused, kauguse mõõdud

CERCS: P170 Arvutiteadus, arvutusmeetodid, süsteemid, juhtimine (automaatjuhtimisteooria)

Contents

1 Introduction

Representational Similarity Analysis is a popular framework for analyzing data in neuroscience. RSA was developed as a way to relate data from different branches of neuroscience: measured data, computational models and behavior (figure 1). Even if we consider only brain activity measurement, RSA can be very useful by giving us a possibility to easily relate data gathered between different brain regions, between different individuals, between different species and between different modalities (such as singe-cell recordings, fMRI, EEG and others).

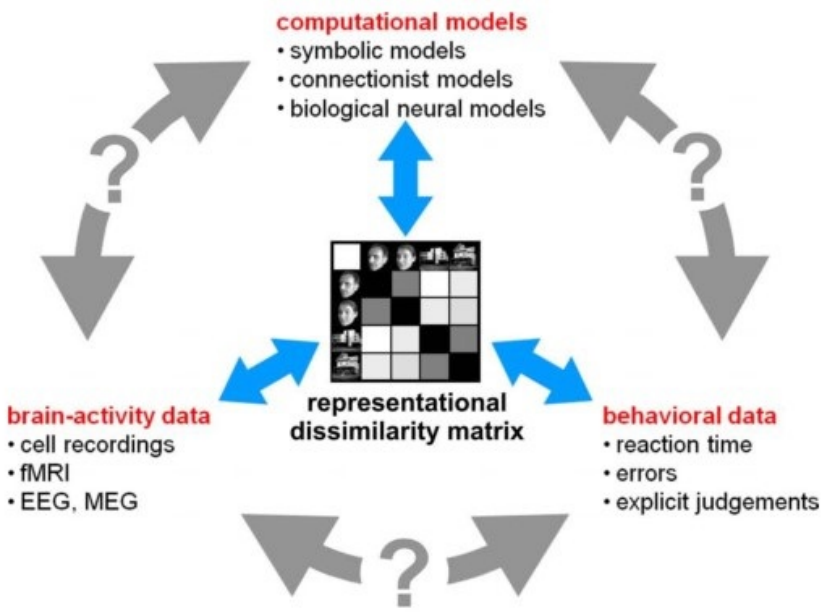

Figure 1. Problems that RSA was designed to address. Image taken from [KMB08].

RSA is based on a principle of a second-order dissimilarity: instead of comparing data gathered under two different circumstances directly, perhaps by creating some correspondence between them, we instead compare the differences between each pair of experimental conditions. Practically speaking, if we have a series of arrays of data with each array corresponding to an experimental condition, we compare how these arrays are different for given circumstances, and then compare it with differences in arrays collected under different circumstances. By calculating these differences between each pair of experimental conditions, we get a symmetric $n \times n$ matrix, which is called Representational Dissimilarity Matrix (figure 2). RDM captures how a series of experimental conditions is represented in a region of interest in a brain.

By applying RSA, we can compare representations of data across different brain

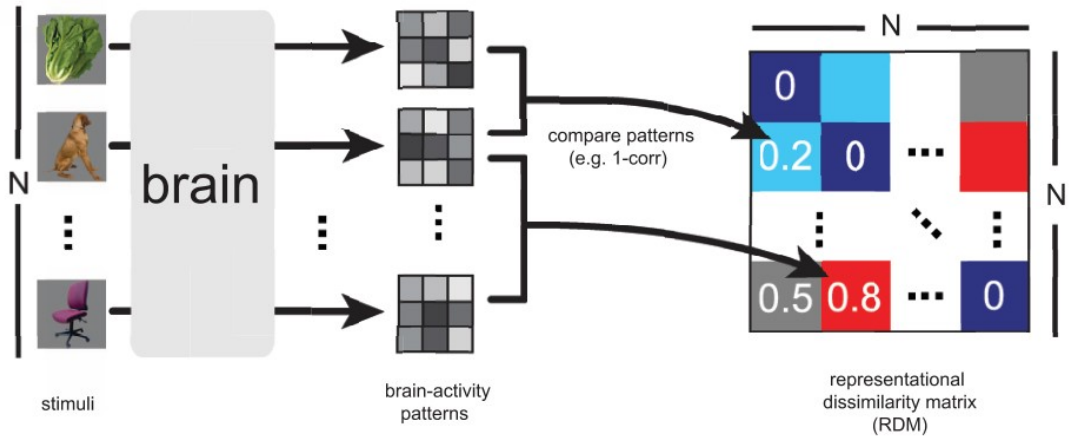

Figure 2. Second-order dissimilarity and construction of RDM. Image taken from [NWW+14].

regions or across species. However, application of RSA may bring changes to correlations between two arrays of initially measured data.

An example: the original RSA article mentions a comparison of signals of different modalities elicited in human and monkey ITs [KMB08]. The resulting correlation between the two RDMs is 0.49. Is neuroscience, such correlation is considered high enough to make conclusions that two regions are somehow related. However, if "real" correlation between two datasets (implying one of them is mapped to another in some way so that we have same dimensionality of data) is lower, meaning that application of RSA results in inflated correlations, then there will be a need to reassess previous applications of RSA to correlated data.

In this thesis, we check how Pearson (linear) correlation between two arrays of data can influence Kendall-tau (rank) correlation in processed data. We do this by generating correlated data by sampling it from different distributions, varying dimensionality of data and changing distance measures when calculating each RDM entry. The work includes implementing generation of data with given Pearson correlation coefficient and distribution of each of these correlated arrays, implementing the pipeline of RSA from scratch, implementing calculation of various distance measures, and finally, checking how varying initial distributions, distance measures and dimensionality of initial data changes final correlation between RDMs.

An important part is that in this thesis data is generated with ground truth already known. The analysis won't be influenced by possible noise in measured data.

Section "Methods and Materials" describes the process of generating data with predefined correlation and the details of applying RSA. In section "Results", the figures,

made by varying initial distributions of data, dissimilarity measures and dimensionality of initial data, are analyzed, and influence of correlations between final RDMs is checked. In the "Discussion" section, the results are interpreted and possiblities of further research are discussed.

2 Methods and Materials

The main workflow of this thesis is split into two parts – generating correlated data and applying RSA to it. The workflow diagrams are presented on figures 3 and 4.

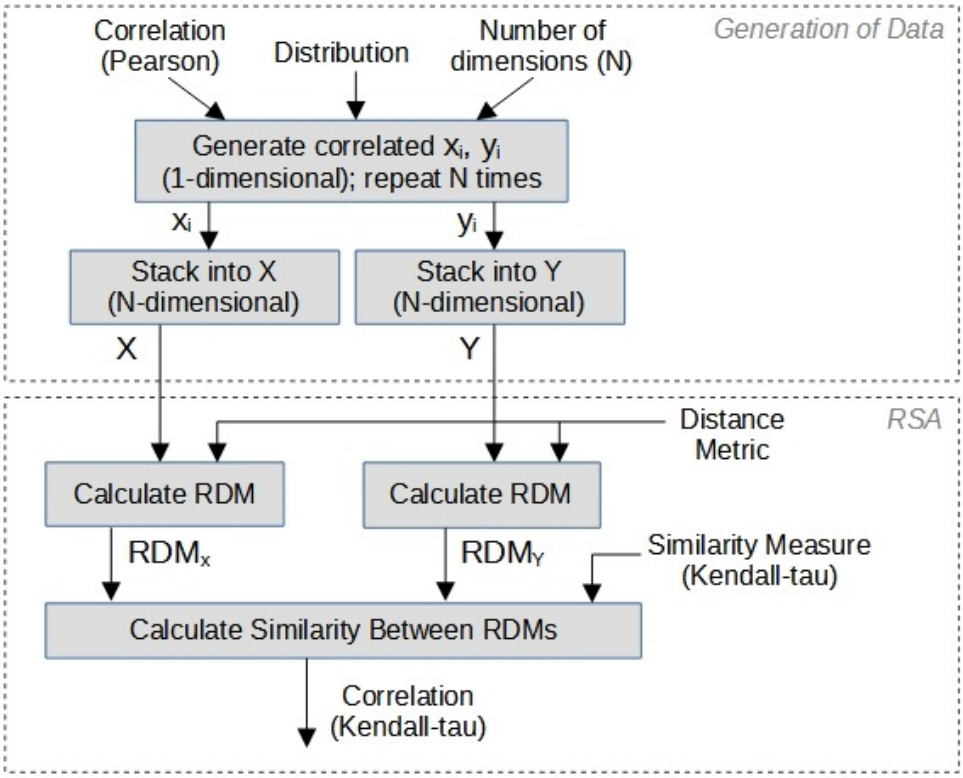

Figure 3. **Workflow diagram of a single experiment.** We define desired correlation, distribution, distance measure and dimensionality of data. Two correlated one-dimensional samples are generated from distribution of our choice. In case data is multidimensional, we generate several one-dimensional arrays and stack them together. Two arrays of data are then used to construct RDMs using desired dissimilarity measure. Then, similarity between RDMs is calculated.

In terms of algorithms, generation of data and RSA are independent from each other. To generate two arrays of data, we need to choose Pearson correlation coefficient, dimensionality of data n, size of a sample m and a distribution. Size of a sample and parameters of distributions are fixed in advance and remain the same for each experiment. As a result, we get two arrays $X, Y \in \mathbb{R}^{m \times n}$. To apply RSA, we need two arrays of data, choose a dissimilarity metric for RDM construction and a similarity metric to calculate

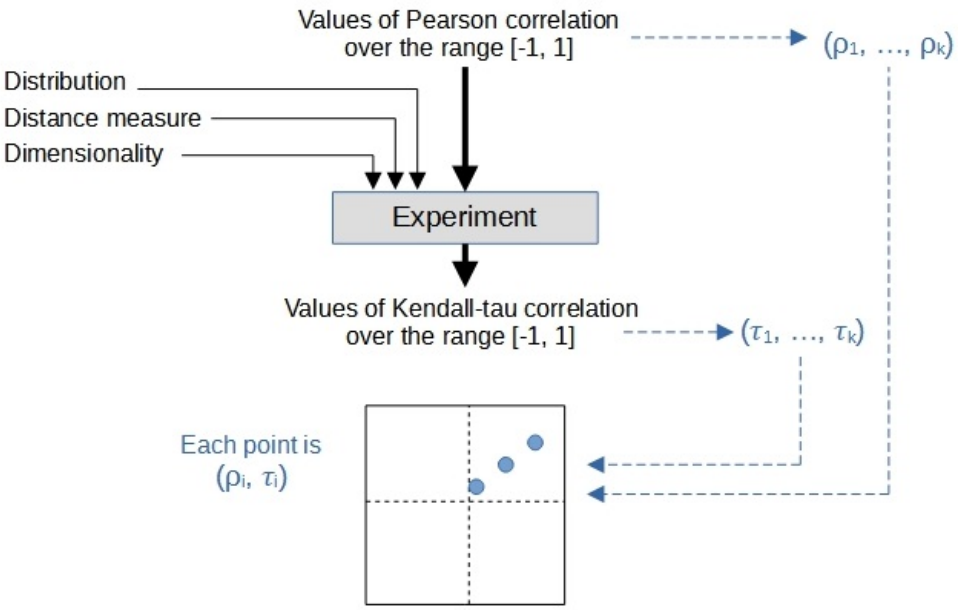

Figure 4. **Workflow diagram for a series of experiments.** We vary Pearson correlation coefficient, while keeping other parameters fixed. As a result, we get several corresponding values of Kendall-tau correlations after applying RSA. Each pair of values results in a point on the plot. After plot is constructed, we change one of the parameters and repeat the procedure, which allows us to see how different parameters influence the graph.

similarity between RDMs. As a similarity measure for comparison of RDMs, we use Kendall-tau correlation; this measure remains fixed over the course of experiments. We perform RSA on each of arrays of data and compare two resulting RDMs. As a result, we get Kendall-tau correlation τ.

The workflow described forms an algorithm for one *experiment*. By fixing distribution of initial data, its dimensionality and distance measure and then varying input Pearson correlation and getting different corresponding values for output Kendall-tau correlation, we perform a *series of experiments*. For such series we get a graph, on which we can see how linear correlations in initial data and rank correlations obtained through RSA are related.

2.1 Generation of Data With Given Correlation

We want to test how sampling data from different distributions affects the result after applying RSA. In each case we generate two arrays of data. In each case we have some desired correlation between two arrays. In this thesis, we check the influence of two

normally distributed correlated samples, two uniformly distributed correlated samples and of two bimodal distributed samples.

Inversion method Typically every case of random number generation is based on assumption that we can have a source of uniform random variable $U(0, 1)$. If we want to sample data from other distributions, we first draw a sample from uniform distribution and then apply various transformations. A method which is considered to be universal for generation of random variables with custom distributions is inversion method.

Assume that we want to generate a sample from random variable X with CDF $F_X(p)$. Then we can first draw a sample $(y_1, y_2, \dots y_m)$ from uniform distribution $U(0, 1)$ and then calculate values $x_i = F_X^{-1}(y_i)$, which will have desired distribution of X. $F^{-1}(p)$ is called *inverse distirbution function*, also known as *quantile function* or *percent-point function* (PPF). Proof of the method and more details can be found in [Dev86].

The problem with inversion method is that not it's not always possible to calculate inverse of a CDF in a closed form. In fact, even normal distribution doesn't have inverse CDF in closed form and relies on so-called error function, which can't be solved analytically. Often numerical solutions or approximations by other functions are applied.

In this thesis, we use inversion method explicitly only during generation of bimodal data. This method (or other methods) can be used without our knowledge when using numerical packages for generation of non-uniform data.

Generation of multidimensional data To generate two arrays of data with dimensionality n, we simply generate two one-dimensional arrays n times, followed by stacking data together over dimensions for each of those arrays. Such procedure is equivalent to sampling from $2n$-dimensional random variable, which covariance matrix is a block-diagonal matrix (equation 1), and then using odd marginal samples as marginals of the first array and even ones as the second array.

$$\Sigma = \begin{pmatrix} \Sigma_1 & O & \dots & O \\ O & \Sigma_2 & \dots & O \\ \vdots & \vdots & \ddots & \vdots \\ O & O & \dots & \Sigma_n \end{pmatrix} \in \mathbb{R}^{(2n \times 2n)}, \quad \Sigma_i = \begin{pmatrix} \rho_i & 0 \\ 0 & \rho_i \end{pmatrix}, \quad O = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \tag{1}$$

Assume that $Z \in \mathbb{R}^{2n \times m}$ is a sample of size m from $2n$-dimensional random variable with covariance matrix Σ, defined as in equation 1. Then, two arrays $X, Y \in \mathbb{R}^{n \times m}$ will be constructed as:

$$X^{(i)} = Z^{(2i-1)}, Y^{(i)} = Z^{(2i)}; i = 1 \dots n/2 \tag{2}$$

,

where $S^{(i)}$ is an i-th row vector of sample S.

Once again, note that in our construction we explicitly define correlations only between row vectors $X^{(i)}, Y^{(i)}, i = 1 \dots n$; we do not define correlations between different dimensions (when $i \neq j$). However, since we do not define any dependence between them, we can safely assume that these correlations are expected to be zero.

Kolmogorov-Smirnov test To verify that two arrays of data indeed have the desired correlation, we calculate correlation as a value between samples. To verify that an array has the desired distribution, we use *Kolmogorov-Smirnov test* (KS-test) [Mas51]. It is a nonparametric test of statistical significance that allows us to check whether a given sample is taken from specified distribution (one-sided test) or, alternatively, whether two samples are taken from the same, undefined, distribution (two-sided test). In this thesis we check only whether samples have required distribution, so we use only one-sided test.

Assume that distribution X has CDF $F_X(x)$ and a sample has empirical CDF calculated as $S_N(x) = k/N$, where N is size of a sample and k is amount of observations less than or equal to x. Then KS-test statistic is calculated as:

$$d = \max_x |F_X(x) - S_N(x)| \tag{3}$$

The intuition behind it is that we find the highest deviation of distribution's CDF from empirical (sample's) CDF and use it as test statistic. This is depicted on figure 5.

p-value can be calculated as $p = 1 - F_K(d\sqrt{n})$, where $F_K(x)$ is CDF of Kolmogorov distribution. If $p < \alpha$, where α is level of significance, we reject the null-hypothesis (we conclude that distribution of a sample is different from distribution we were checking for).

There are existing packages which implement calculation of test statistic and p-value. We use implementation from `scipy` package.

For each statistical hypothesis test in this thesis, we use statistical significance level $\alpha = 0.01$. That is, we reject the null-hypothesis only if $p \leq 0.01$.

2.1.1 Normal Distribution

Multivariate Normal Distribution is a generalization of normal distribution onto several dimensions. n-dimensional distribution of such type has two variables – mean $\mu \in \mathbb{R}^n$ and covariance $\mathbf{\Sigma} \in \mathbb{R}^{n \times n}$. The special case when $n = 2$ is called *bivariate normal distribution*. If in addition to only having two dimensions we have $\mu = 0$ and $\sigma = 1$, we have a *standard* bivariate normal distribution. The density functon is that case is

$$f(x_1, x_2) = \frac{1}{2\pi\sqrt{1-\rho^2}} \exp\left\{ -\frac{1}{2(1-\rho^2)}(x_1^2 + 2\rho x_1 x_2 + x_2^2) \right\}$$

, where ρ is a correlation between its two components.

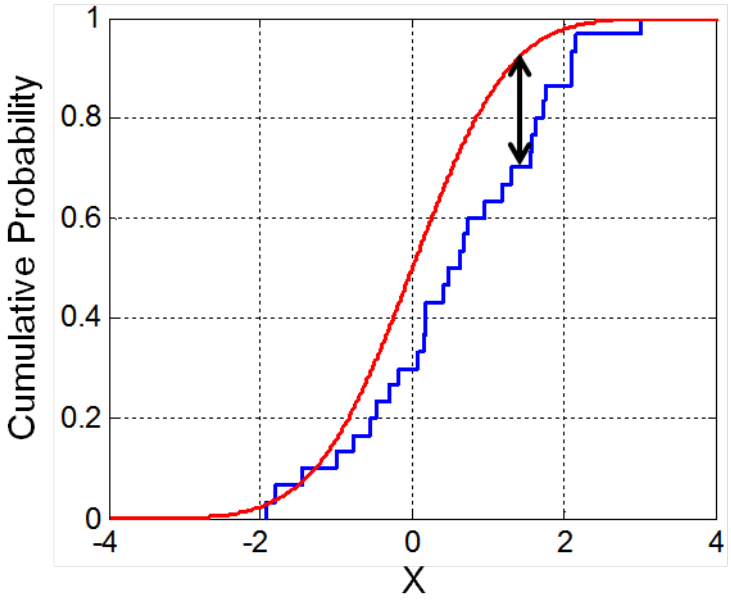

Figure 5. Illustration of Kolmogorov-Smirnov test. Red line depicts distribution's CDF, blue – sample's CDF; distance depicted by black is test statistic.

Each component (marginal distribution) is distributed normally. In addition, parameter ρ allows us to define correlation between marginal distributions. That means that by fixing ρ and drawing m samples from bivariate standard normal distribution, we get two arrays of length m with mean 0 and variance 1, which are also correlated.

To generate a sample from multivariate normal distribution using uniform distribution $U(0, 1)$, we need to use *Cholesky decomposition*. Given a symmetric positive-definite matrix $A \in Mat(\mathbb{R})$, its Cholesky decomposition is defined as $A = LL^T$, where L is a lower triangular matrix with positive diagonal entries. Such decomposition can be calculated, for example, through eigendecomposition:

$$A = Q\Lambda Q^{-1} = Q\Lambda^{\frac{1}{2}}\Lambda^{\frac{1}{2}}Q^{-1} = Q\Lambda^{\frac{1}{2}}(\Lambda^{\frac{1}{2}})^{-1}Q^{-1} = Q\Lambda^{\frac{1}{2}}(\Lambda^{\frac{1}{2}})^{T}Q^{T} = Q\Lambda^{\frac{1}{2}}(Q\Lambda^{\frac{1}{2}})^{T} = LL^{T}$$

, which holds because Λ is diagonal and Q is unitary.

Decomposing covariance matrix $\Sigma = LL^T$ and then multiplying L by a vector of independent random variables, which are normally distributed, gives us a vector of correlated random variables, i.e. a multivariate random variable. In two-dimensional case:

$$X = LZ = \begin{pmatrix} 1 & 0 \\ \rho & \sqrt{1-\rho^2} \end{pmatrix} Z$$

, i.e. $x_1 = z_1$, $x_2 = \rho z_1 + \sqrt{1-\rho^2} z_2$. Correlation between x_1 and x_2 in this case is ρ.

In this thesis, we use function `multivariate_normal` from Python's `numpy` package. This way, we only need to define distribution's parameters to draw samples.

An example of generated data is shown on figure 6.

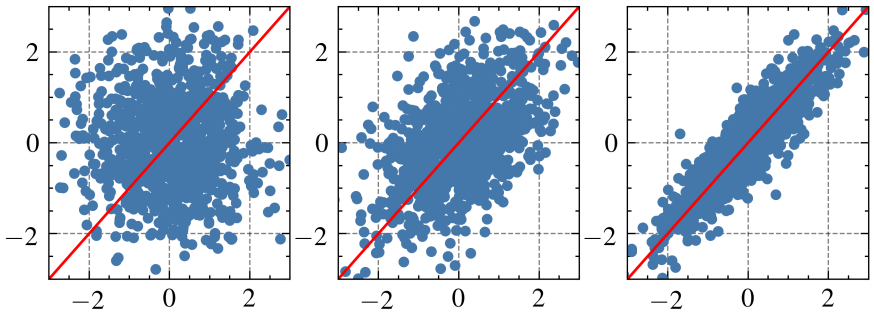

Figure 6. An example of normally generated data for different values of preferred correlation (0, 0.5 and 0.9, correspondingly)

To test how close sample correlations of generated arrays would be to the desired correlation, the generator was tested for values of correlation between 0 and 1 with step 0.1. On each run, 500 points were generated and for each value of desired correlation real correlation was averaged over 400 attempts. To test the spread of correlations between generated samples, the generator was launched 500 times, with 500 points generated for each pair of arrays. Results are presented on figures 8 and 7.

2.1.2 Uniform Distribution

Generating two arrays such that both would be uniformly distributed and correlated to a predefined degree is more challenging, compared to the normal distribution case. To do that, we would need to sample data from a *copula* ([Sch07]). Copulas are useful when we want to handle marginal distributions and dependence structure between then independently, which fits the goal of this thesis perfectly.

Copula is a function $C : [0,1]^n \to [0,1]$, which satisfies properties of cumulative distribution function and each marginals of which is distributed uniformly.

Sklar's theorem states that for any n-dimensional CDF F with marginal distributions $F_1, \ldots, F_n$ there exists a copula C, such that

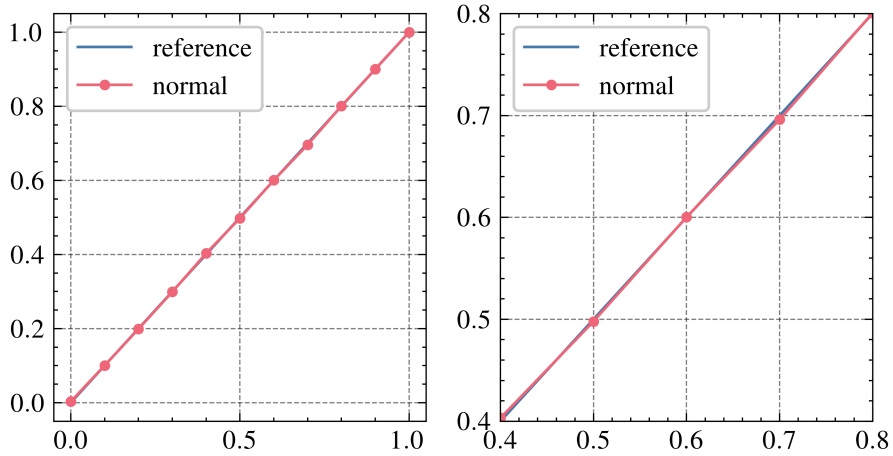

Figure 7. Comparison of generated and desired correlation. On the left – overview for all tested values. On the right – enlarged image for intermediate values of correlation (0.4-0.8).

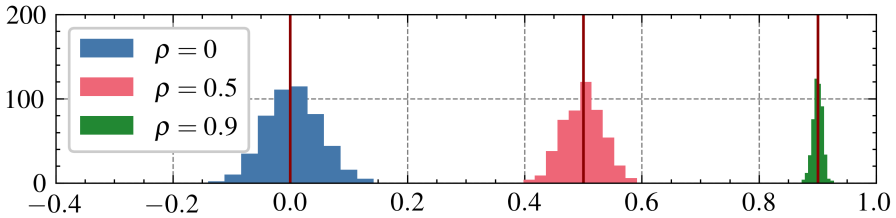

Figure 8. Distribution of sample correlation for normally distributed arrays for different values of preferred correlation

$$F(x_1, \ldots, x_n) = C(F_1(x_1), \ldots, F_n(x_n))$$

By defining $u_i = F_i(x_i)$ and taking into account that $x_i = F^{-1}(u_i)$ we get the following formula:

$$C(u_1, \ldots, u_n) = F(F_1^{-1}(u_1), \ldots, F_n^{-1}(u_n))$$

Note that $u_i \in [0, 1], i = 1, \ldots, n$.

There are different copulae which are generated using different means. In this thesis we use *Gaussian copula*.

If we define $F_i(x_i) = \Phi(x_i)$ (CDF of standard normal distribution) and $F(x_1, \ldots, x_n) = \Phi_\Sigma(x_1, \ldots, x_n)$ (CDF of multivaraite standard normal distribution with

correlation matrix Σ), then C is called Gaussian copula. Formula for such copula is defined by

$$C_{\Sigma}^{Gauss}(u_1, \ldots, u_n) = \Phi_{\Sigma}(\Phi^{-1}(u_1), \ldots, \Phi^{-1}(u_n)) \tag{4}$$

Since in bivariate case correlation matrix is defiend by

$$\Sigma = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$$

, we Gaussian copula for bivariate case can be defined by

$$C_{\rho}^{Gauss}(u_1, u_2) = \Phi_{\Sigma}(\Phi^{-1}(u_1), \Phi^{-1}(u_2)) \tag{5}$$

, where ρ is the only parameter in correlation matrix Σ.

Required function for sampling from Gaussian copula was found in `statsmodels` package, which is used in this thesis. The function accepts correlation correlation value as an input, simplifying our calculations.

An example of arrays generated from Gaussian copula is shown on figure 9. Kolmogorov-Smirnov test has shown that both arrays are indeed distributed uniformly.

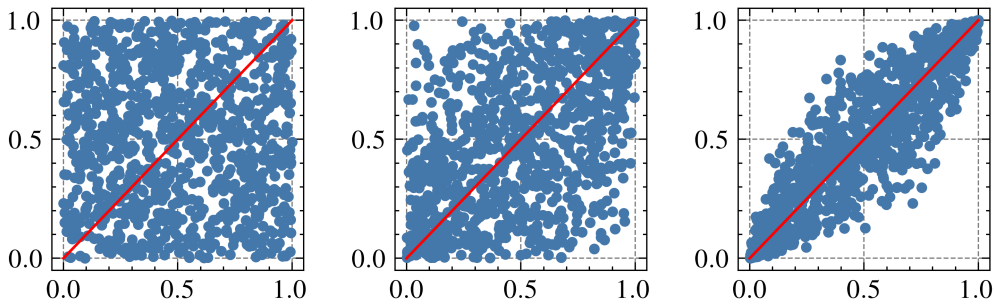

Figure 9. An example of uniformly generated data for different values of preferred correlation (from left to right: 0, 0.5, 0.9)

To test how close sample correlations of generated arrays would be to the desired correlation, the generator was tested for values of correlation between 0 and 1 with step 0.1. On each run, 500 points were generated and for each value of desired correlation real correlation was averaged over 400 attempts. To test the spread of correlations between generated samples, the generator was launched 500 times, with 500 points generated for each pair of arrays. Results are presented on figures 11 and 10.

KS-test has shown that both components are indeed distributed uniformly. Calculating Pearson correlation between them has revealed that correlations are close to the ones defined in correlation matrix.

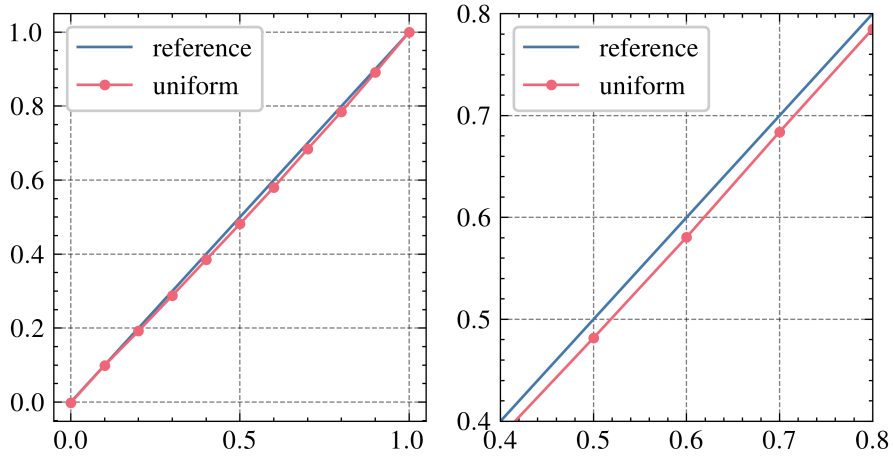

Figure 10. Comparison of generated and desired correlation. On the left – overview for all tested values. On the right – enlarged image for values of correlation (0.4-0.8).

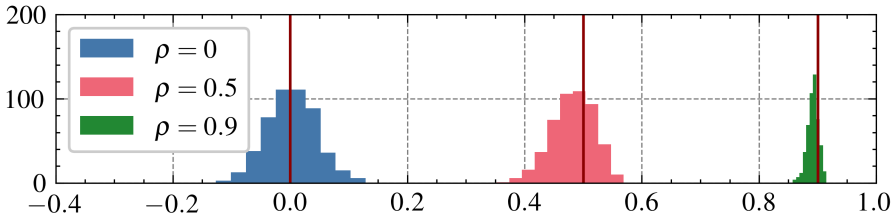

Figure 11. Distribution of sample correlation for uniformly distributed arrays for different values of preferred correlation

2.1.3 Bimodal Distribution

Multimodal distribution is a distribution with more than one mode. If a distribution has exactly two modes, it is called *bimodal distribution*.

Bimodal distribution used in this thesis is *bimodal normal distribution* [GDSCO21]. Its PDF is given by

$$f(x, \mu, \sigma, \alpha) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2 - \frac{\alpha^2}{2}\right] \cosh\left[\alpha\left(\frac{x-\mu}{\sigma}\right)\right]$$

Distribution's CDF is given by

$$F(x;\mu,\sigma,\alpha) = \frac{1}{4}\left[2 + erf\left(\frac{x-\mu-\alpha\sigma}{\sigma\sqrt{2}}\right) + erf\left(\frac{x-\mu+\alpha\sigma}{\sigma\sqrt{2}}\right)\right]$$

This distribution has two modes when $|\alpha| > 1$. In this thesis, we use parameters $\mu = 0, \sigma = 1, \alpha = 2$.

Despite this distribution having a bivariate version, there is no closed form for inducing a correlation structure between marginal distributions. Therefore, we use an inversion method to generate two correlated samples, each having bimodal normal distribution.

Since there is no easy way to find the inverse of bivariate normal distribution's CDF, we will approximate it numerically.

First, we calculate values of CDF at large number of points. Then, by inverting the axes, we have values of the quantile function. Then, we fit a polynomial curve onto these points (in our case, we used polynomial of degree 17). The resulting polynomial is an explicit approximation of quantile function. Analytic and induced PDFs are presented on figure 12.

Polynomial that approximates the quantile function is defined in standard basis and can be calculated as:

$$F_{BN}^{-1}(x) \approx p(x) = \sum_{i=0}^{n} c_i x^i$$

Similar to the case with uniform distribution, we can sample data from Gaussian copula. That way, we have two ararys of uniformly distributed data with correlation close to the desired one. By taking each of resulting marginal samples (which are uniform distributions $U[0,1]$) and calculating values of the approximated quantile function at these values, we get two samples which have bimodal normal distribution and are correlated.

Examples of generated data are presented on figure 13.

To test how close sample correlations of generated arrays would be to the desired correlation, the generator was tested for values of correlation between 0 and 1 with step 0.1. On each run, 500 points were generated and for each value of desired correlation real correlation was averaged over 400 attempts. To test the spread of correlations between generated samples, the generator was launched 500 times, with 500 points generated for each pair of arrays. Results are presented on figure s15 and 14.

As we can see, the correlation in generated data is slightly lower than the desired one. According to [Sch07], it is to be expected since using Pearson correlation as a measure of dependence for non-elliptic distributions (such as normal distribution or mixtures of normal distributions, to which our transformed distribution apparently doesn't belong to) leads to many fallacies. It is also stated in [KLW23] that in general case Pearson correlation is not invariate under marginal transforms, specifically under non-linear

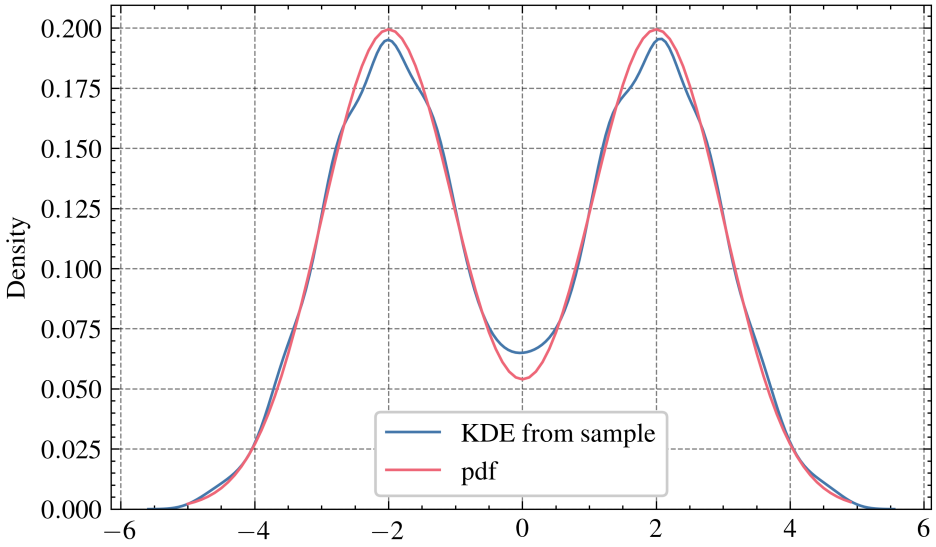

Figure 12. PDFs of BN normal distribution built analytically and induced from an approximated quantile function. Blue graph is built using kernel density estimation on generated data. Orange is an analytical PDF.

strictly increasing transformations. Possible workaround is to calculate a predistorted correlation and generate a sample from bivariate normal distribution with this value, which will be higher, in which case we'll get the desired correlation after the transform [LH75]. In this thesis, however, we do not need strictly matching correlations to make required figures, so we do not perform these calculations.

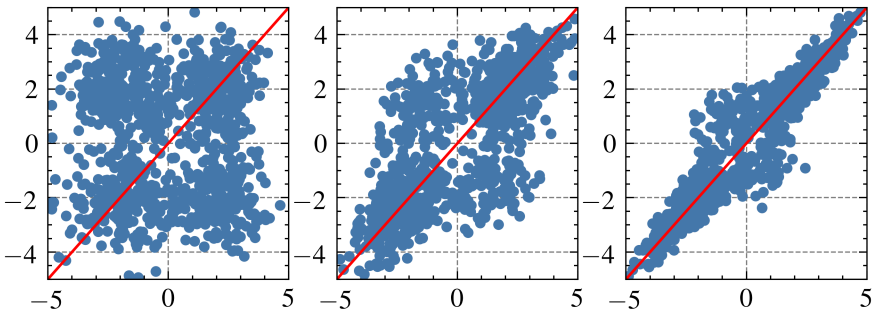

Figure 13. An example of generated data with bimodal distribution for different values of preferred correlation (from left to right: 0, 0.5, 0.9)

2.2 Applying RSA to generated data

After generating two correlated arrays from chosen distribution, we apply RSA to each array and receive two Representational Dissimilatiry Matrices (RDM). While during data generation, we vary distributions and dimensionality, in the process of RSA itself we vary dissimilarity measures and see how they influence the results.

The code for RSA that we used was written from scratch. Later we compare how it behaves compared to an existing RSA toolbox [NWW+14].

2.2.1 Dissimilarity measures

In the context of RSA, "dissimilarity" is often synonymous with the term "distance", sometimes also called "distance metric". However, some dissimilarities are not qualified as distances. It is also worth noting that quite often distances have equivalent norms, but having an equivalent norm is not a necessary requirement for a distance measure.

Mathematically speaking, if $M \subset \mathbb{R}^n$ and $x, y, z \in M$ and $d : M \times M \to \mathbb{R}$ and d satisfies the axioms:

1. $d(x, y) \geq 0$ with $d(x, y) = 0 \Leftrightarrow x = y$ (non-negativity)
2. $d(x, y) = d(y, x)$ (symmetry)
3. $d(x, z) \leq d(x, y) + d(y, z)$ (triangle inequality)

then d is a distance measure.

In this thesis, we consider three distance measures: Euclidean, cosine and geodesic graph distance. Other distances often used in neuroscience include correlation distance and Mahalanobis distance.

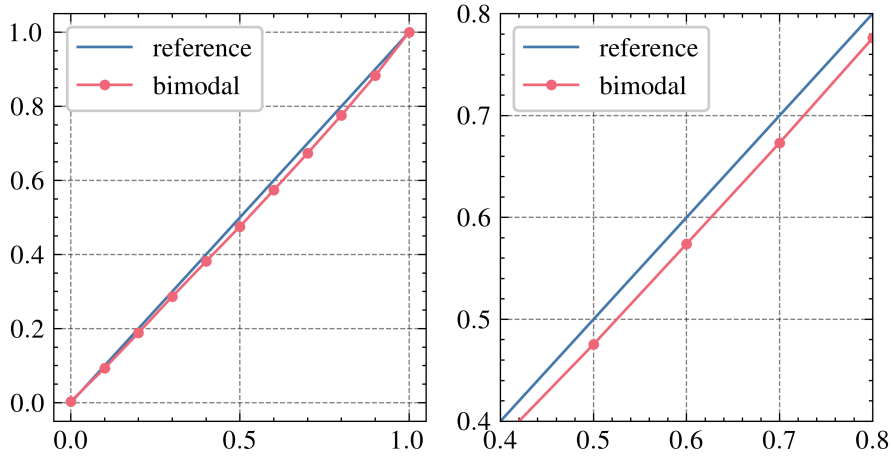

Figure 14. Comparison of generated and desired correlation. On the left – overview for all tested values. On the right – enlarged image for values of correlation (0.4-0.8).

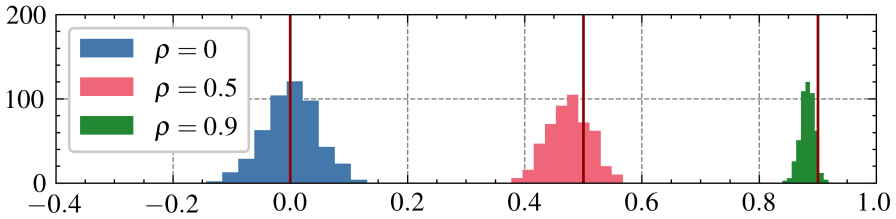

Figure 15. Distribution of sample correlation for uniformly distributed arrays for different values of preferred correlation

Euclidean distance Euclidean distance is a widely used metric and what is often thought of intuitively when using the term "distance". If $x, y \in \mathbb{R}^n$, then Euclidean distance is defined by

$$d_{Eucl}(x,y) = \sqrt{(x_1 - y_1)^2 + \ldots + (x_n - y_n)^2} = \left(\sum_{i=1}^{n}(x_i - y_i)^2\right)^{\frac{1}{2}}$$

where x_i, y_i are i-th components of vectors x and y, correspondingly.

Euclidean distance is often used in Machine Learning to measure distances between embeddings.

Cosine distance Cosine distance is another distance metric, which is notorious for its use in Natural Language Processing. In this thesis, we tested its application in RSA. If

$x, y \in \mathbb{R}^n$, then the formula for cosine distance is:

$$d(x,y) = 1 - s(x,y) = 1 - \frac{x \cdot y}{||x||_2 ||y||_2} = 1 - \frac{\sqrt{x_1 y_1 + \ldots x_n y_n}}{\sqrt{x_1^2 + \ldots + x_n^2}\sqrt{y_1^2 + \ldots + y_n^2}} \quad (6)$$

where $x \cdot y$ is a dot product of x, y and $|| \cdot ||_2$ is a Euclidean norm of a vector. Quantity $s(x, y)$ is called *cosine similarity* and represents an angle between vectors x and y.

Geodesic graph distance Geodesic graph distance is a distance measure derived from the first step of Isomap algorithm [TdSL00]. The principle is illustrated on figure 16. First, we have a set of points (block A). For each point, we either find k nearest points, where k is some fixed number, or find all points in some fixed radius from given point, after which we calculate Euclidean distances from given point to all nearest points (block B). After that, we can build a weighted graph and calculate distance between any two points given initially by using any algorithm for calculating graph distance (such as Floyd-Warshall algorithm [Flo62] [War62]).

Calculating geodesic graph distance is a first step of popular dimensionality reduction algorithms such as Isomap. Therefore, Geodesic graph distance is especially useful in cases when we have data which is located on some manifold. For example, if we have n-dimensional set of points that are located on some manifold (i.e., their positions can be defined parametrically by less than n numbers), then calculating the distance across the manifold (plain blue line, figure 16) would make more sense than calculating direct Euclidean distance (dashed blue line, figure 16). This distance is approximated by building a graph using points presumably located on a manifold (red line, blocks B and C, figure 16).

In this thesis, building the graph is performed by connecting each vertex to its $k = 4$ neighbors. Algorithm that calculates shortest paths between each pair of vertices is Floyd-Warshall algorithm. Distance between each pair of points (i.e. weights of a graph) is measured by Euclidean distance.

2.2.2 Comparing RDMs

After calculating dissimilarity between each pair of experimental conditions, for both of correlated arrays, we get two RDMs. We want to compare how these two matrices are correlated, and ultimately, check how correlation between matrices is different from correlation between initial arrays of observed (or, in our case, generated) data.

When we have two RDMs on our hands, one possible way to compare them is by first normalizing them (e.g. using rank-transform) and then using a common distance measure (e.g. Euclidean). However, we can calculate correlation between directly – in that case, normalization is done implicitly. If we expect linear correspondence between

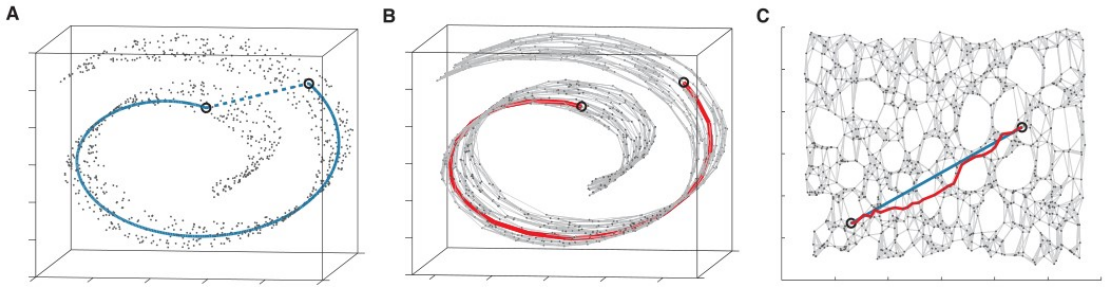

Figure 16. Visualization of Isomap algorithm (taken from [TdSL00])

dissimilarity values in the two matrices, we can use Pearson correlation coefficient. If we can expect only ordinal correspondence or a nonlinear but monotonic relationship, it is better to use rank correlation coefficient. [KMB08]

Further research had shown that for comparing RDMs not all rank correlation coefficients are equally good either. It is preferred to use Kendall-tau correlation coefficient to Spearman correlation [NWW+14]. Despite the fact that Spearman correlation coefficient is still a good alternative if there are no conceptual models in use (meaning there are no ties), we use Kendall-tau coefficient for consistency and to avoid possible confusion.

3 Results

In this thesis, *an experiment* is defined as a process that includes generating data with chosen desired correlation, performing RSA and then measuring Kendall-tau correlation between resulting RDMs. Such procedure is done when three properties – distribution of initial data, distance measure and number of dimensions – are fixed. *A series of experiments* includes varying correlation in input data and calculating corresponding output correlations, i.e. each series is represented by one graph. Each figure in this section has multiple series' of experiments (typically 2 or 3), which allows us to compare how varying different properties influences dependence between the input correlation (linear dependency of the generated data) and output correlation (as measured by RSA). In a sense, we are interested in measuring how the data processing in RSA distorts the linear dependencies between original data, and how those effects depend on data distribution and dimensionality.

Each point on the graph represents an arithmetic mean among 5 experiments, for both coordinates. For each point the first coordinate is average for sample mean and the second is an average for the resulting Kendall-tau correlation between RDMs. The desired correlation was changed in steps of 0.1, in range $[-1, 1]$.

3.1 Varying Distribution of Initial Data

Here the dissimilarity is fixed to be either Euclidean or cosine distance and then 10-dimensional data is generated for various initial distributions. The results are represented on figures 17 and 18. Varying distribution of initial data doesn't seem to affect correlation much. However, it seems that for bimodal distribution there are slightly higher losses in correlation structure compared to other distributions.

3.2 Varying Dissimilarity Measures

Here we fix distribution as normal and data as 10-dimensional and varied dissimilarity measures. The results are represented on figures 19, 20 and 21. We can see an abnormality on figure 20 – when using uniform distribution and cosine distance, corresponding output correlation for negative values of input correlation are much lower than for Euclidean distance. The graph for cosine distance is also non-symmetric. Another finding is that application of geodesic graph distance in RSA severely distorts correlation values, making them lower (specifically for intermediate values between 0 and 1).

3.3 Varying Dimensionality of Initial Data

For each figure in this section we fix distribution of initial data and distance measure and plot how different dimensionalities lead to different results (figures 22, 23, 24). Varying

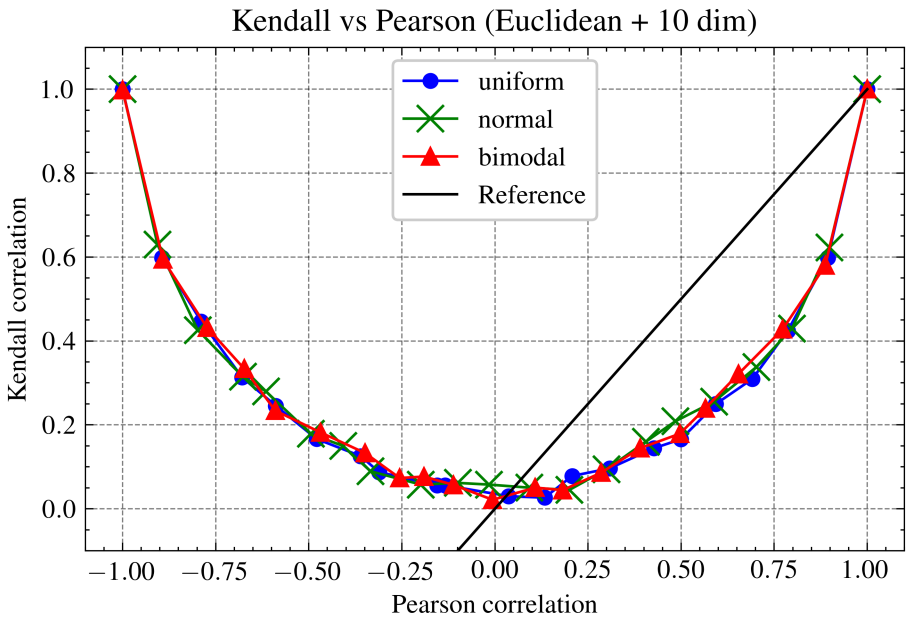

Figure 17. Comparison of correlations in initial and processed data (Euclidean + 10 dimensions)

dimensionality of data doesn't seem to affect correlation much, except that data of lower dimensionality seems to be worse at preserving correlation structure.

3.4 Deeper Analysis

The majority of functions for generation of correlated data and performing RSA were written from scratch. External functions include those for numerical computations (numpy), plotting (matplotlib), statistics (scipy).

Comparison of results for functions written from scratch with those from the existing toolbox [NWW+14] had been performed. Comparison of final Kendall-tau correlation for functions from toolbox and functions written from scratch is plotted on figure 28.

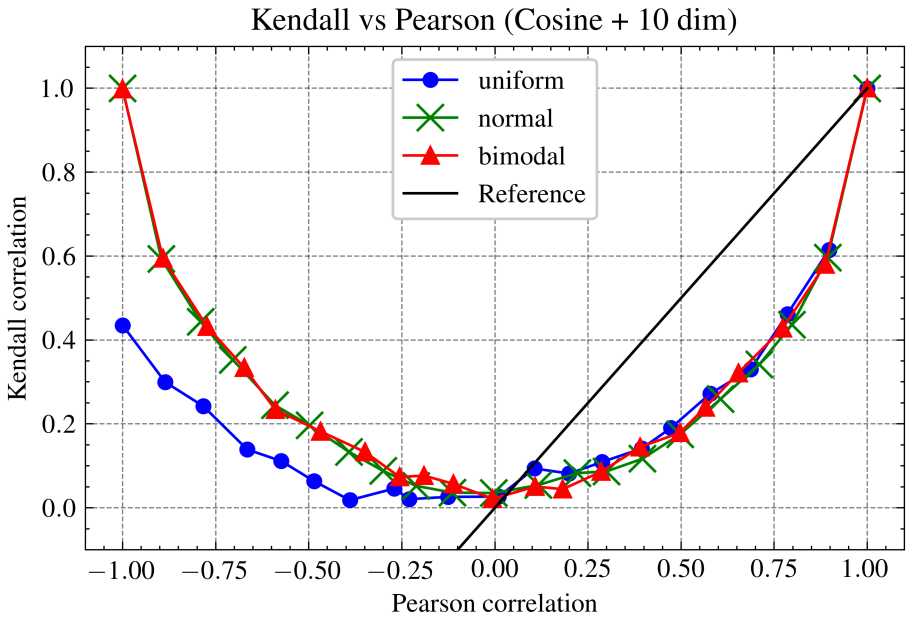

Figure 18. Comparison of correlations in initial and processed data (cosine + 10 dimensions)

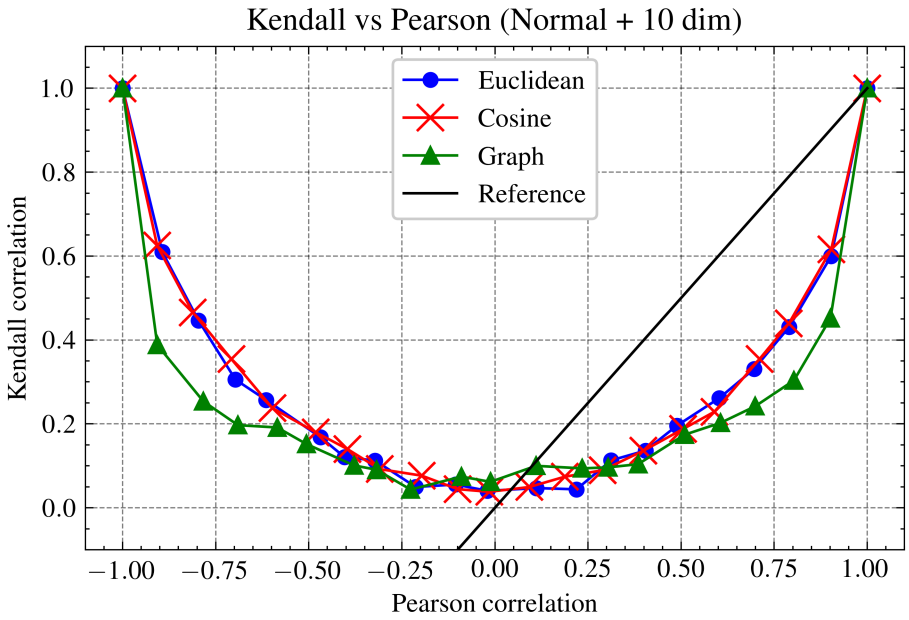

Figure 19. Comparison of correlations in initial and processed data (normal distribution + 10 dimensions)

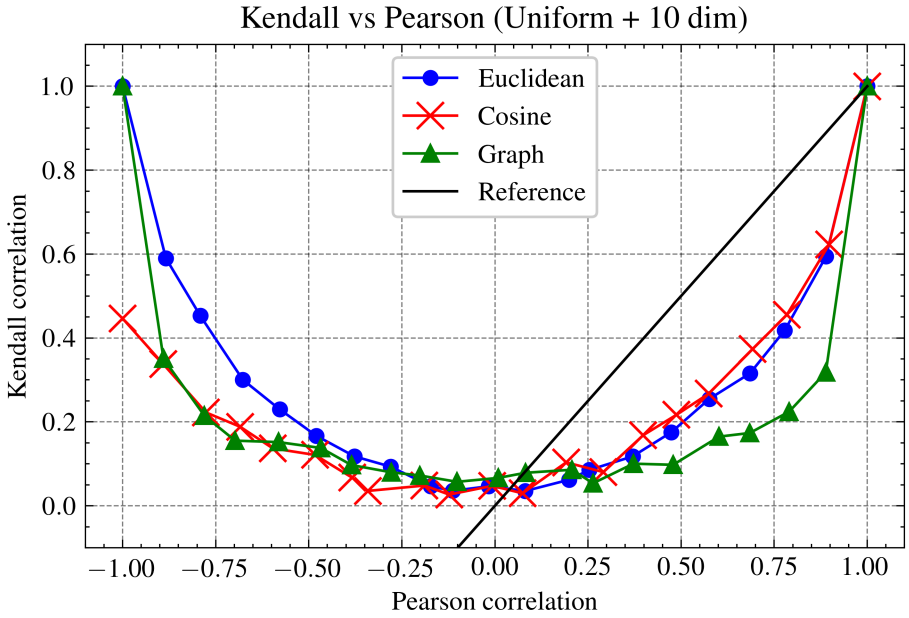

Figure 20. Comparison of correlations in initial and processed data (uniform distribution + 10 dimensions)

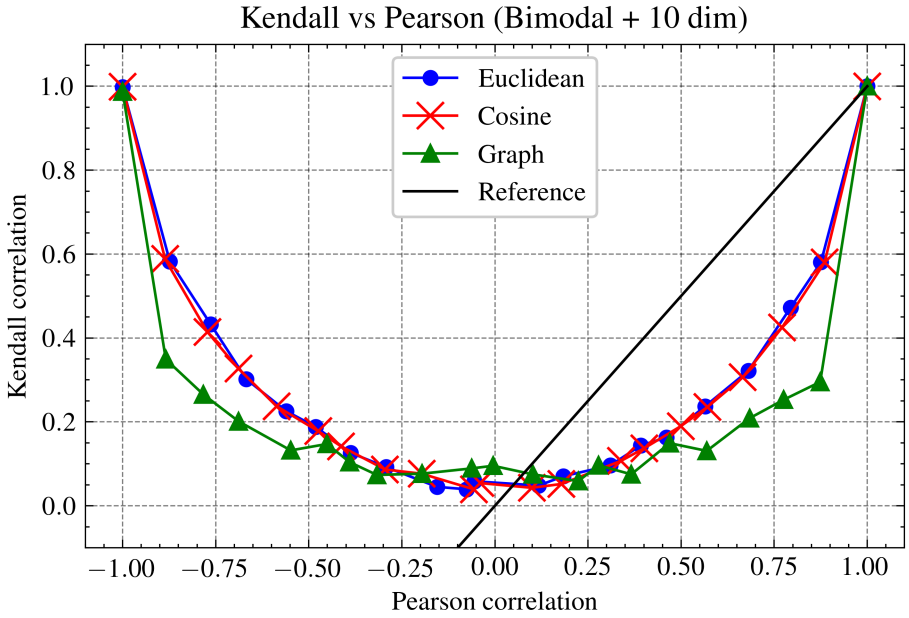

Figure 21. Comparison of correlations in initial and processed data (bimodal distribution + 10 dimensions)

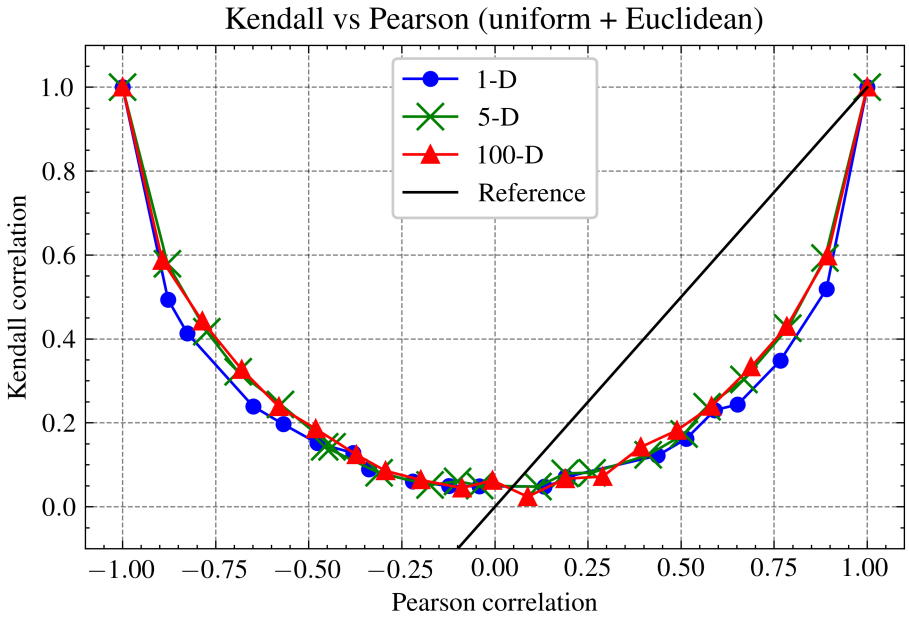

Figure 22. Comparison of correlations in initial and processed data (uniform + Euclidean)

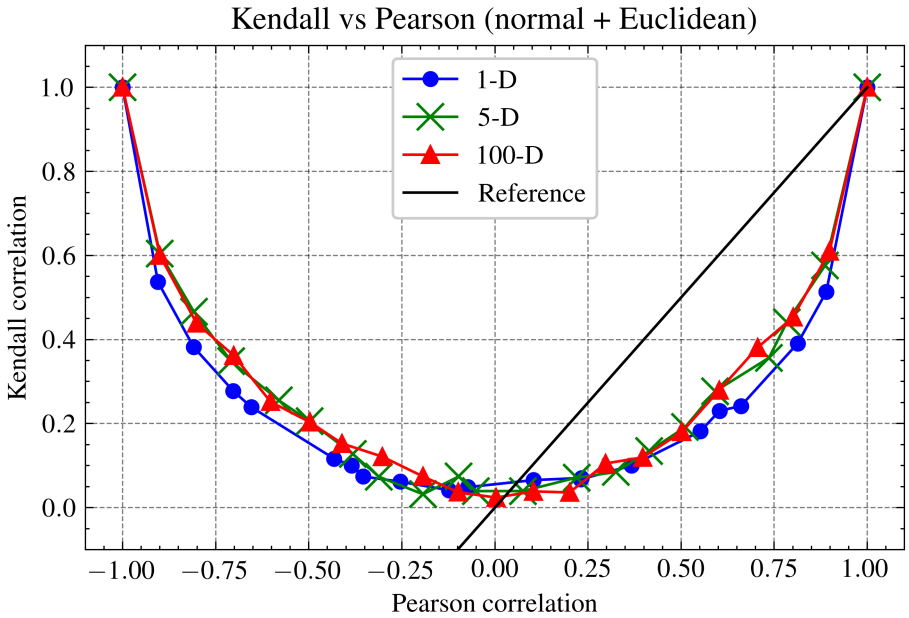

Figure 23. Comparison of correlations in initial and processed data (normal + Euclidean)

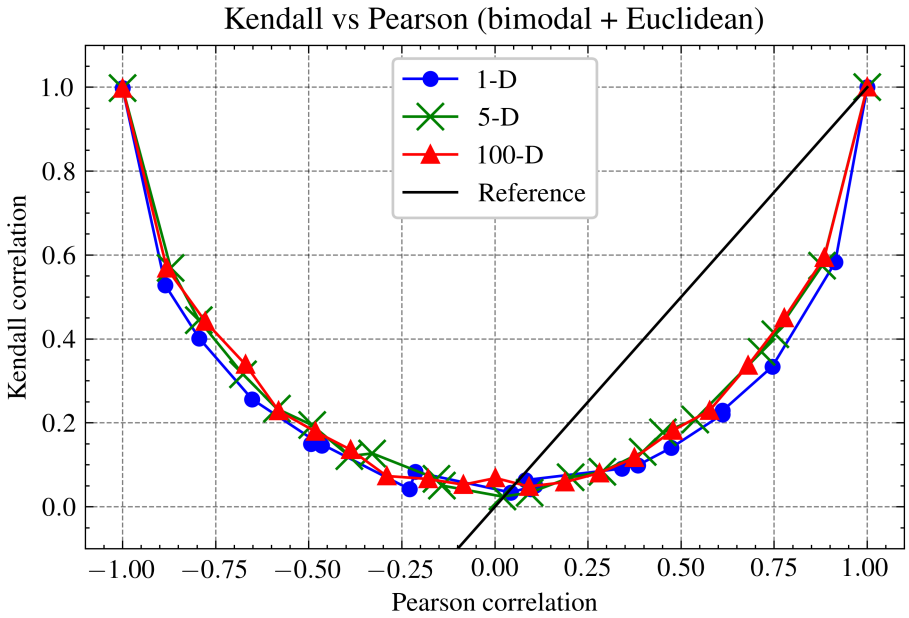

Figure 24. Comparison of correlations in initial and processed data (bimodal + Euclidean)

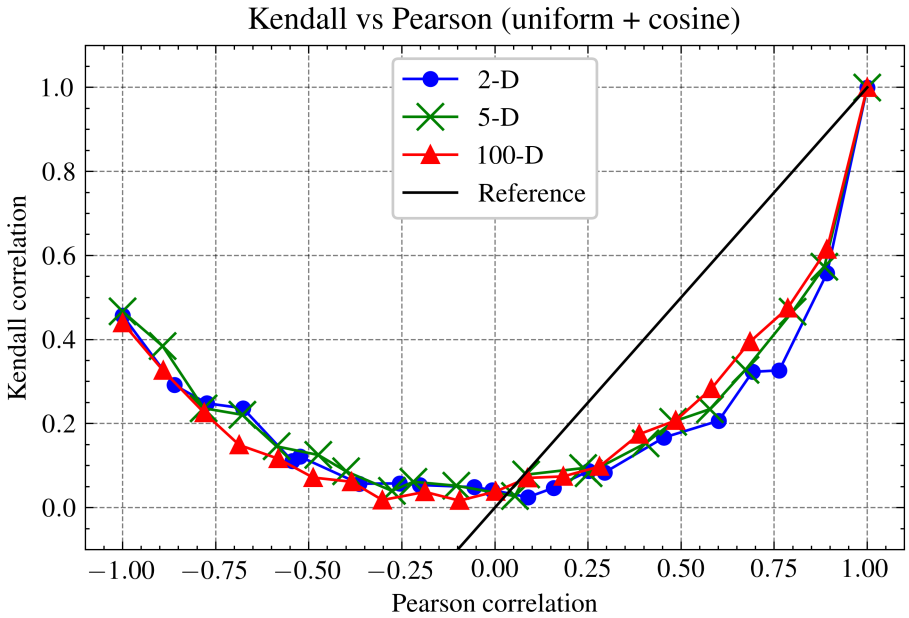

Figure 25. Comparison of correlations in initial and processed data (uniform + cosine)

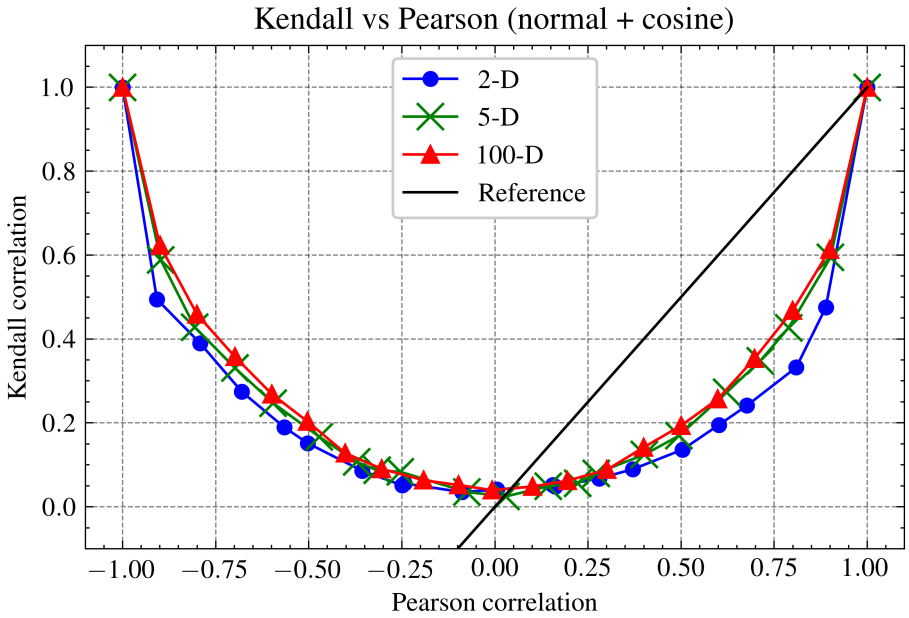

Figure 26. Comparison of correlations in initial and processed data (normal + cosine)

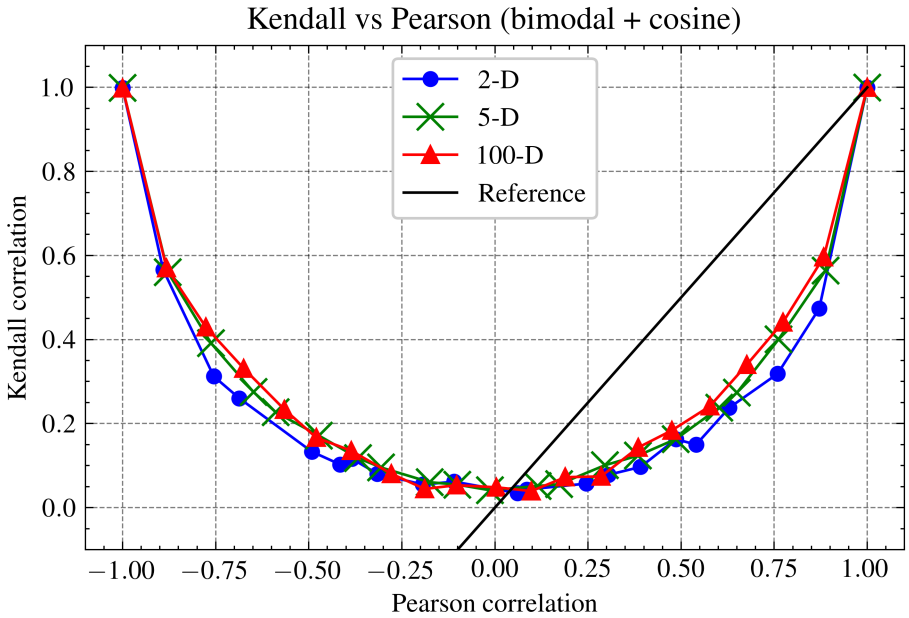

Figure 27. Comparison of correlations in initial and processed data (bimodal + cosine)

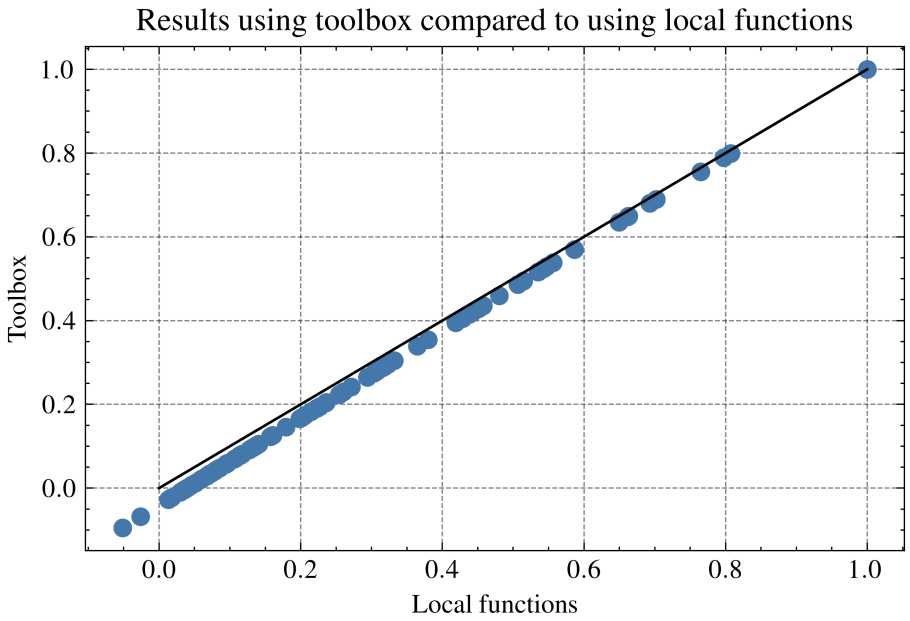

Figure 28. Comparison of final Kendall-tau correlations after using local functions and after using the toolbox

4 Discussion

Results show that correlations between two arrays of initial data becomes lower after processing them using RSA. In particular, the effect is more noticeable for intermediate values of the linear correlation coefficient (0.3-0.6). The relation between initial correlation (as predefined in the generated data) and the correlation between RDMs after processing forms a convex function. Generally, this indicates that RSA will tend to underestimate linear dependencies in the data.

It is also worth noting that if correlation between two initial arrays is negative, it becomes positive after applying RSA. One possible explanation is that distance measures used in RSA are always non-negative, but for a definitive answer further investigation is required.

Another finding of current work is how linear correlation in underlying data becomes distorted when two arrays are distributed uniformly and negatively correlated, while using cosine distance as dissimilarity measure. Rank correlation after application of RSA in that case becomes much lower compared cases with positive correlation, in which case figures of rank correlation in processed data plotted against linear correlations in underlying data become severely asymmetric. At the moment, it is unknown what causes this phenomenon and further investigation is required.

An interesting future direction might be to study the effect that noise can have on our observed relations between linear dependencies in the underlying data and the correlation captured by RSA. In particular, since RSA understimates linear correlations it seems crucial to test whether noise can hide the linear correlation to the point that is not detectable by RSA (although present in the underlying noisy data).

Another possible direction for further research is applying ideas from this thesis to other dissimilarity measures between activity patterns (such as Mahalanobis distance or correlation distance) and checking other distributions of initial data.

One particular research ([SSAN21]) suggests that under certain circumstances, RDMs can lie on a Riemann mainfold, in which case it might be a good idea to approximate Riemann distances using geodesic graph distance mentioned in this thesis.

5 Conclusion

In this thesis, we researched how correlations and data distribution between two arrays of initial data influence correlations between two corresponding RDMs after application of RSA. The work included generation of correlated data and writing RSA functions from scratch.

Our results indicate that RSA generally underestimates linear dependencies in the underlying data. This might be also natural given that most distance measures include some non-linearity in their definition which might distort linear measures of dependencies. Further analysis is needed to fully explore the relation between the dependencies in the original data and the correlations captured by RSA.

References

[Dev86] Luc Devroye. *Non-Uniform Random Variate Generation*. Springer-Verlag, New York, NY, USA, 1986.

[Flo62] Robert W. Floyd. Algorithm 97: Shortest path. *Commun. ACM*, 5(6):345, jun 1962.

[GDSCO21] E. Gómez-Déniz, J.M. Sarabia, and E. Calderín-Ojeda. Bimodal normal distribution: Extensions and applications. *Journal of Computational and Applied Mathematics*, 388:113292, 2021.

[KLW23] Takaaki Koike, Liyuan Lin, and Ruodu Wang. Invariant correlation under marginal transforms, 2023.

[KMB08] Nikolaus Kriegeskorte, Marieke Mur, and Peter Bandettini. Representational similarity analysis - connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience*, 2, 2008.

[LH75] Shing Ted Li and Joseph L. Hammond. Generation of pseudorandom numbers with specified univariate distributions and correlation coefficients. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-5(5):557–561, 1975.

[Mas51] Frank J. Massey. The kolmogorov-smirnov test for goodness of fit. *Journal of the American Statistical Association*, 46(253):68–78, 1951.

[NWW+14] Hamed Nili, Cai Wingfield, Alexander Walther, Li Su, William Marslen-Wilson, and Nikolaus Kriegeskorte. A toolbox for representational similarity analysis. *PLOS Computational Biology*, 10(4):1–11, 04 2014.

[Sch07] Thorsten Schmidt. Coping with copulas. *Copulas-From theory to application in finance*, 3:1–34, 2007.

[SSAN21] Mahdiyar Shahbazi, Ali Shirali, Hamid Aghajan, and Hamed Nili. Using distance on the riemannian manifold to compare representations in brain and in models. *NeuroImage*, 239:118271, 2021.

[TdSL00] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[War62] Stephen Warshall. A theorem on boolean matrices. *J. ACM*, 9(1):11–12, jan 1962.

Appendix

I. Glossary

Dissimilarity Measure – a mathematical measure that satisfies several axioms; intuitively, an inverse of a similarity measure – the less similar two entities are, the higher this measure is

Experimental Environment – a combination of circumstances that theoretically, when unchanged, under the same experimental conditions, should produce the same response patterns; in our setting, it is a combination of a predefined brain region, measuring modality and an individual undergoing the experiment

Experimental Condition – a distinct and discrete part of an experiment which elicits a distinct response pattern in a brain; for example, when an idea of experiment is to show subject a series of images and measure responses in the brain, an experimental condition would be one image

Experiment – a set of experimental conditions

RDM – see Representational Dissimilarity Matrix

Representation – how some real-life object or phenomenon is represented in a brain; in our setting representation is synonymous with representational dissimilarity matrix, which utilizes so-called second-order dissimilarity

Representational Dissimilarity Matrix – a square matrix which entries show dissimilarity between two experimental conditions on intersection of each row and column; a matrix shows how a given set of experimental conditions is represented in a given brain region measured by some given modality

RSA – see Representational Similarity Analysis

Representational Similarity Analysis – a technique in neuroscience that allows us to compare representations using second-order dissimilarities

Second-Order Dissimilarity – an idea that allows us, instead of comparing two sets of response patterns elicited by two different experimental conditions directly, to calculate dissimilarity among these patterns under each set of experimental conditions and then compare the dissimilarities themselves

II. Repository

Repository with code and supplementary notebook used for calculations in this thesis will be available at the following link:

`https://github.com/Corvu/rsa-thesis`

III. Licence

Non-exclusive licence to reproduce thesis and make thesis public

I, **Savelii Vorontcov**,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to

   reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

   Effects of Data Distributions and Distance Measures in Representational Similarity Analysis,

   supervised by Raul Vicente Zafra.

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 4.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Savelii Vorontcov
04/01/2024