Stress Functions for Nonlinear Dimension Reduction, Proximity Analysis, and Graph Drawing*

Lisha Chen† and Andreas Buja

Yale University and University of Pennsylvania

This draft: November 16, 2012

Abstract

Multidimensional scaling (MDS) is the art of reconstructing pointsets (embeddings) from pairwise distance data, and as such it is at the basis of several approaches to nonlinear dimension reduction and manifold learning. At present, MDS lacks a unifying methodology as it consists of a discrete collection of proposals that differ in their optimization criteria, called "stress functions". To correct this situation we propose (1) to embed many of the extant stress functions in a parametric family of stress functions, and (2) to replace the ad hoc choice among discrete proposals with a principled parameter selection method. This methodology yields the following benefits and problem solutions: (a) It provides guidance in tailoring stress functions to a given data situation, responding to the fact that no single stress function dominates all others across all data situations; (b) the methodology enriches the supply of available stress functions; (c) it helps our understanding of stress functions by replacing the comparison of discrete proposals with a characterization of the effect of parameters on embeddings; (d) it builds a bridge to graph drawing, which is the related but not identical art of constructing embeddings from graphs.

Key words and phrases: Multidimensional Scaling, Force-Directed Layout, Cluster Analysis, Clustering Strength, Unsupervised Learning, Box-Cox Transformations

1 INTRODUCTION

In the last decade and a half an important line of work in machine learning has been nonlinear dimension reduction and manifold learning. Many approaches used in this area are

*Running title: Stress Functions

†Corresponding author. Statistics Department, Yale University, 24 Hillhouse Ave, New Haven, CT 06511 (lisha.chen@yale.edu).

based on inter-object distances and the faithful reproduction of such distances by so-called "embeddings," that is, mappings of the objects of interest (images, signals, documents, genes, network vertices, ...) to points in a low dimensional space such that the low-dimensional distances mimic the "true" inter-object distances as best as possible. Examples of distance-based methods include, among many others: kernel PCA (KPCA; Schölkopf, Smola, and Müller 1998), "Isomap" (Tenenbaum, Silva, and Langford 2000), kernel-based semidefinite programming (SDP; Lu, Keles, Wright, and Wahba 2005; Weinberger, Sha, Zhu, and Saul 2006), and two very different methods that both go under the name "local multidimensional scaling" by Venna and Kaski (2006) and by the present authors (Chen and Buja 2009). These can all be understood as outgrowths of various forms of multidimensional scaling (MDS).

MDS approaches are divided into two distinct classes: (1) classical scaling of the Torgerson-Gower type (the older approach) is characterized by the indirect approximation of target distances through inner products; (2) distance scaling of the Kruskal-Shepard type is characterized by the direct approximation of target distances. The relative merits are as follows: classical scaling approaches often reduce to eigendecompositions that provide hierarchical solutions (increasing the embedding dimension means adding more coordinates to an existing embedding); distance scaling approaches are non-hierarchical and require high-dimensional optimizations, but they tend to force more information into any given embedding dimension. It is this class of distance scaling approaches for which the present article provides a unified methodology.

Distance scaling approaches differ in their choices of a "stress function", that is, a criterion that measures the mismatch between target distances (the data) and embedding distances. Distance scaling and the first stress function were first introduced by Kruskal (1964a,b), followed with proposals by Sammon (1969), Takane, Young and de Leeuw (ALSCAL, 1977), Kamada and Kaway (1989), among others. The problem with a proliferation of proposals is that proposers invariably manage to find situations in which their methods shine, yet no single method is universally superior to all others across all data situations in any meaningful sense, nor does one single stress function necessarily exhaust all possible insights to be gained even from a single dataset. For example, embeddings from two stress functions on the same data may both be insightful in that one better reflects local structure, the other global structure.

This situation calls for a rethinking that goes beyond the addition of further proposals. Needed is a methodology that organizes stress functions and provides guidance to their specific performance on any given dataset. To satisfy this need we will execute the following program: (1) We embed extant stress functions in a multi-parameter family of stress functions that ultimately extends to incomplete distance data or distance graphs, thereby encompassing "energy functions" for graph drawing; (2) we interpret the effects of some of these parameters on embeddings in terms of a theory that describes how different stress functions entail different compromises in the face of conflicting distance information; (3) we use meta-criteria

to measure the quality of embeddings independently of the stress functions, and we use these meta-criteria to select stress functions that are in well-specified senses (near) optimal for a given dataset. We have used meta-criteria earlier (Chen and Buja 2009) in a single-parameter selection problem, and a variation of the approach proves critical in a multi-parameter setting.

For part (1) of the program we took a page from graph drawing which had been in a situation similar to MDS: a collection of discrete proposals for so-called "energy functions", the analogs of stress functions for graph data. This state of affairs changed with the work by Noack (2003) who embedded extant energy functions in single-parameter families of energy functions. Inspired by this work, the first author (Chen 2006) proposed in her thesis the four-parameter family of distance-based stress functions presented here for the first time. These stress functions are based on Box-Cox transforms and named the "B-C family"; it includes power laws and logarithmic laws for attracting and repulsing energies, a power law for up- or down-weighting of small or large distances, as well as a regularization parameter for incomplete distance data. This family provides an umbrella for several stress functions from the MDS literature as well as energy functions from the graph drawing literature. A related two-parameter family of energy functions for weighted graph data was proposed by Noack (2009), and we study its connection to stress functions for distance data in Section 2.5.

For part (2) of the program, the analysis and interpretation of the stress function parameters, we develop the nucleus of a theory that explains the effects of the some of the parameters on embeddings. Here, too, we looked to Noack (2003, 2007, 2009) for a template of a theory, but it turns out that distance data, considered by us, and weighted graph data, considered by Noack (2009), require different theories. For one thing, distance data, unlike weighted graph data, have a natural concept of "perfect embedding", which is achieved when the target distance data are perfectly matched by the embedding distances. We show that all members in the B-C family of stress functions for complete distance data have the property that they are minimized by perfect embeddings if such exist (Section 2.3) because they satisfy what we call "edgewise unbiasedness". In the general case, when there exists no perfect embedding, a natural question is how the minimization of stress functions creates compromises between conflicting distance information. To answer this question we introduce the notion of "scale sensitivity", which is the degree to which the compromise is dominated by small or large distances through the interaction of two stress function parameters (Section 2.4).

Before we outline step (3) of our program, we make a point that is of interest to machine learning: The B-C family of stress functions encompasses energy functions for graph drawing through an extension from complete to incomplete distance data. First we note that MDS based on complete distance data has been successfully applied to graph drawing through the device of shortest-path length computation for all pairs of nodes in a graph; see, for example, Gansner et al. (2004). Underlying this device is the interpretation of (unweighted) graphs as incomplete

distance data whereby edges carry a distance of +1 and non-edges have missing distances. Similarly, the ISOMAP method of nonlinear dimension reduction relies on a complete distance matrix consisting of shortest path lengths computed from a local distance graph. There exists, however, another device for extending MDS to graphs: It is possible to canonically extend all B-C stress functions from complete to incomplete distance data by constructing a limit whereby intuitively non-edges are imputed with an infinite distance that has infinitesimally small weight, creating a pervasive repulsing energy that spreads out embeddings and prevents them from crumpling up. This limiting process offers up a parameter to control the relative strength of the pervasive repulsion vis-à-vis the partial stress for the known distances, thereby acting as a regularization parameter that stabilizes embeddings by reducing variance at the cost of some bias. — This device, first applied by the authors (Chen and Buja 2009) to Kruskal's stress function, brings numerous energy functions for unweighted graphs under the umbrella of the B-C family of stress functions.

Finally, in step (3) of our program, we turn to the problem of selecting "good embeddings" from the multitude that can be obtained from the B-C family of stress functions. This problem can be approached in a principled way with a method that was first used by the authors again in the case of Kruskal's stress function (Chen and Buja 2009; Chen 2006; Akkucuk and Carroll 2006): We employ "meta-criteria" that judge how well embeddings preserve the input topology in a manner that is independent of the stress function used to create the embedding. These meta-criteria measure the degree to which K-nearest neighborhoods are preserved in the mapping of objects to their images in an embedding. K-NN structure is insensitive to nonlinear monotone transformations of the distances in both domains, implying that the meta-criteria allow even quite biased (distorted) configurations to be recognized as performing well in the minimalist sense of preserving K-NNs. Thus the parameters of the B-C family of stress functions can be chosen to optimize a meta-criterion. In this way we turn the ad hoc trial-and-error search for good embeddings into a parameter selection problem.

This article proceeds as follows: Section 2 introduces the B-C family of stress functions in steps: It first interprets Kruskal's (1964a) stress in the framework of attracting and repulsing energies (Section 2.1); it then generalizes these energies with general power laws (Section 2.2), discusses the notions of edgewise unbiasedness (Section 2.3) and scale sensitivity (Section 2.4), as well as the relation between distance- and weight-based approaches (Section 2.5), and generalizes the family to the case of incomplete distance data (Section 2.6). The section concludes with technical aspects concerning the irrelevance of the relative strengths of attracting and repulsing energies (Section 2.7) and the unit invariance of the repulsion parameter for incomplete distance data (Section 2.8). Subsequently, Section 3 introduces the meta-criteria, and Section 4 illustrates the methodology with simulated examples (Section 4.1), the Olivetti face data (Section 4.2) and the Frey face data (Section 4.3). Section 5 concludes with

a discussion.

2 MDS Stress Functions Based on Power Laws

2.1 Kruskal's Stress as Sum of Attracting and Repulsing Energies

To start we assume a generic MDS situation in which a full set of target distance data $D = (D_{i,j})_{i,j=1,\ldots,N}$ is given for all pairs of objects of interest. We assume $D_{i,i} = 0$ and $D_{i,j} > 0$ for $i \neq j$. MDS solves what we may call the "Rand McNalley Road Atlas problem": Given a table showing the distances between all pairs of cities, draw a map of the cities that reproduces the given distances.

Kruskal's (1964a) original MDS proposal solves the problem by proposing a stress function that is essentially a residual sum of squares (RSS) between the target distances given as data and the distances in the embedding. An embedding (configuration, graph drawing) is a set of points $\mathbf{X} = (\mathbf{x}_i)_{i=1,\ldots,N}$, $\mathbf{x}_i \in \mathbb{R}^p$, so that

$$d_{i,j} \;=\; \|\mathbf{x}_i - \mathbf{x}_j\|$$

are the embedding distances (we limit ourselves to Euclidean distances). The goal is to find an embedding $\mathbf{X}$ whose distances $d_{i,j}$ fit the target distances $D_{i,j}$ as best as possible. Kruskal's stress function is therefore

$$S(\boldsymbol{d}|\boldsymbol{D}) \;=\; \sum_{i,j} (d_{i,j} - D_{i,j})^2,$$

where we let $\boldsymbol{d} = (d_{i,j})_{i,j} = (\|\mathbf{x}_i - \mathbf{x}_j\|)_{i,j}$ and $\boldsymbol{D} = (D_{i,j})_{i,j}$. Optimization is carried out over all $N \times p$ coordinates of the configuration $\mathbf{X}$.

Taking a page from the graph drawing literature, we interpret Kruskal's stress function as composed of an "attracting energy" and a "repulsing energy" as follows:

$$S(\boldsymbol{d}|\boldsymbol{D}) \;=\; \sum_{i,j} ({d_{i,j}}^2 - 2D_{i,j}\, d_{i,j}) \;+\; \text{const.}$$

The term ${d_{i,j}}^2$ represents an "attracting energy" because in isolation it is minimized by $d_{i,j} = 0$. The term $-2D_{i,j}d_{i,j}$ represents a "repulsing energy" because again in isolation it is minimized by $d_{i,j} = \infty$. (The term ${D_{i,j}}^2$ is a constant that does not affect the minimization; it calibrates the minimum energy level at zero.) A stress term $(d_{i,j} - D_{i,j})^2$ is therefore seen to be equivalent to a sum of an attracting and a repulsing energy term that balance each other in such a way that the minimum energy is achieved at $d_{i,j} = D_{i,j}$.

2.2 The B-C Family of Stress Functions

We next introduce a family of stress functions whose attracting and repulsing energies follow power laws, in analogy to Noack's (2003, 2007, 2009) generalized energy functions for graph drawing. However, we would like this family to also include logarithmic laws, as in Noack's (2003, 2007) "LinLog" energy. To accommodate logarithms in the family of power transformations, statisticians have long used the so-called Box-Cox family of transformations, defined for $d > 0$ by

$$BC_\alpha(d) \;=\; \begin{cases} \frac{d^\alpha - 1}{\alpha} & (\alpha \neq 0) \\ \log(d) & (\alpha = 0) \end{cases}$$

This modification of the raw power transformations d^α not only affords analytical fill-in with the natural logarithm for $\alpha = 0$, it also extends the family to $\alpha < 0$ while preserving increasing monotonicity of the transformations: for $\alpha < 0$ raw powers d^α are decreasing while $BC_\alpha(d)$ is increasing. The derivative is

$$BC_\alpha{}'(d) \;=\; d^{\alpha-1} > 0 \quad \forall d > 0,\;\; \forall \alpha \in \mathbb{R}\,.$$

By subtracting the (otherwise irrelevant) constant 1 in the numerator and dividing by α, Box-Cox transformations are affinely matched to the natural logarithm at $d = 1$ for all powers α:

$$BC_\alpha(1) = 0, \quad BC_\alpha{}'(1) = 1.$$

See Figure 1 for an illustration of Box-Cox transformations.

Using Box-Cox transformations we construct a generalization of Kruskal's stress function by allowing arbitrary power laws for the attracting and the repulsing energies, subject to the constraint that the attracting power is greater than the repulsing power to guarantee that the minimum combined energy is finite ($> -\infty$). We denote the attracting power by $\mu + \lambda$ and the repulsing power by μ with the understanding that $\lambda > 0$ and $-\infty < \mu < +\infty$.

Definition: *The **B-C family of stress functions** for complete distance data $\boldsymbol{D} = (Dij)_{i,j}$ is given by*

$$S(\boldsymbol{d}|\boldsymbol{D}) \;=\; \sum_{i,j=1,\ldots,N} {D_{i,j}}^\nu \left(BC_{\mu+\lambda}(d_{i,j}) \;-\; {D_{i,j}}^\lambda \, BC_\mu(d_{i,j}) \right) . \tag{1}$$

As we assume $D_{i,j} > 0$ for $i \neq j$ the weight term ${D_{i,j}}^\nu$ is meaningful for all powers $-\infty < \nu < +\infty$. Thus ${D_{i,j}}^\nu$ upweights the summands for large $D_{i,j}$ when $\nu > 0$ and downweights them when $\nu < 0$; for $\nu = 0$ the stress function is an unweighted sum. The parameter ν allows us to capture a couple of extant stress functions; see Table 1. Kruskal's stress function does not require ν as it arises from $\mu = 1$, $\lambda = 1$ and $\nu = 0$. The idea of

using general power laws in an attraction-repulsion paradigm arose independently in the first author's PhD thesis (Chen 2006) and in Noack (2009). For a discussion of the relationship between the two proposals see Section 2.5.

2.3 Edgewise Unbiasedness of Stress Functions

The reason for introducing the multiplier ${D_{i,j}}^{\lambda}$ in the repulsing energy is to grant what we call **edgewise unbiasedness**: If there exist only two objects, $N = 2$, with target distance D, then the stress function $S(d) = D^{\nu}\left(BC_{\mu+\lambda}(d) - D^{\lambda}BC_{\mu}(d)\right)$ should be minimized by $d = D$:

$$D \;=\; \mathrm{argmin}_d\, D^{\nu}\left(BC_{\mu+\lambda}(d) - D^{\lambda}BC_{\mu}(d)\right) \tag{2}$$

This property is easily verified using $\lambda > 0$: $S'(d) = D^{\nu+\mu-1}\left(d^{\lambda} - D^{\lambda}\right)$, hence $S'(d) < 0$ for $d \in (0, D)$ and $S'(d) > 0$ for $d \in (D, \infty)$, so that $S(d)$ is strictly descending on $(0, D)$ and strictly ascending on (D, ∞). — This property holds only for this particular choice of the power D^{λ} in the repulsing energy term.

Edgewise unbiasedness is essential to grant the following exact reconstruction property:

Proposition: *If the target data $D_{i,j}$ form a set of Euclidean distances in the embedding dimension, $D_{i,j} = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|$ $(i, j = 1, ..., N)$, then all B-C stress functions are minimized by the embeddings that reproduce the target distances exactly: $d_{i,j} = D_{i,j}$.*

Note that embeddings are unique only up to rotations, translations and reflections. They may have additional non-uniqueness properties that may be peculiar to the data.

2.4 Scale Sensitivity

Next we analyze the role of the parameters ν and λ. As we will see, they determine the degree to which conflicting metric information is decided in favor of small or large target distances. It is a major goal of MDS procedures to reach good compromises to obtain informative embeddings in the general situation when distance data are not perfectly embeddable in a Euclidean space of a given dimension, be it due to error in the target distances, or due to the distance interpretation of what is really just dissimilarity data, or due to intrinsic higher dimensionality of the underlying objects. To gain insight into the nature of the compromises, it is beneficial to construct a simple paradigmatic situation in which contention between conflicting distance data can be analyzed. One such situation is as follows: Assume again that there are only two objects ($N = 2$), but that target distances were obtained twice for this same pair of objects, resulting in different values D_1 and D_2 (due to observation error, say). In practice, one often reduces multiple distances by averaging them, but a more principled approach is to form a

stress function with multiple stress terms per object pair (i, j). In general, if target distances $D_{i,j,k}$ for the object pair (i, j) are observed $K_{i,j}$ times, the B-C stress function will be

$$S \;=\; \sum_{i,j=1,\ldots,N} \sum_{k=1,\ldots,K_{i,j}} {D_{i,j,k}}^{\nu} \Big(BC_{\mu+\lambda}(d_{i,j}) - {D_{i,j,k}}^{\lambda} BC_{\mu}(d_{i,j}) \Big)$$

With this background, the paradigmatic situation of two target distances D_1 and D_2 observed on one object pair is the simplest case that exhibits contention between conflicting distance information. The stress function for the single embedding distance d is

$$S \;=\; {D_1}^{\nu} \Big(BC_{\mu+\lambda}(d) - {D_1}^{\lambda} BC_{\mu}(d) \Big) \;+\; {D_2}^{\nu} \Big(BC_{\mu+\lambda}(d) - {D_2}^{\lambda} BC_{\mu}(d) \Big) \,.$$

It is minimized by

$$d_{\min} \;=\; \Big(\alpha_1 {D_1}^{\lambda} + \alpha_2 {D_2}^{\lambda} \Big)^{1/\lambda}, \quad \text{where } \alpha_1 = \frac{{D_1}^{\nu}}{{D_1}^{\nu} + {D_2}^{\nu}}, \;\; \alpha_2 = \frac{{D_2}^{\nu}}{{D_1}^{\nu} + {D_2}^{\nu}}, \qquad (3)$$

so that $\alpha_1 + \alpha_2 = 1$. Thus $d_{\min}$ is the Lebesgue L_λ norm of the 2-vector (D_1, D_2) with regard to the Bernoulli distribution with probabilities α_1 and α_2 (an improper norm for $0 < \lambda < 1$). However, α_1 and α_2 are also functions of (D_1, D_2), hence the minimizing distance $d_{\min} = d(D_1, D_2)$ is a function of the target distances in a complex way. Yet, the Lebesgue norm interpretation is useful because it allows us to analyze the dependence of d on the parameters λ and ν separately:

- For fixed $D_1 \neq D_2$, the minimizing distance d is a monotone increasing function of ν for $-\infty < \nu < \infty$, and we have

  $$d_{\min} \;=\; \Big(\alpha_1 {D_1}^{\lambda} + \alpha_2 {D_2}^{\lambda} \Big)^{1/\lambda} \quad \begin{cases} \uparrow \;\; \max(D_1, D_2) & \text{as} \quad \nu \uparrow \infty \,, \\ \downarrow \;\; \min(D_1, D_2) & \text{as} \quad \nu \downarrow -\infty \,. \end{cases}$$

  The reason is that if $D_1 > D_2$ we have $\alpha_1 \uparrow 1$ as $\nu \uparrow \infty$, and $\alpha_2 \uparrow 1$ as $\nu \downarrow -\infty$.

- For fixed $D_1 \neq D_2$, the minimizing distance d is a monotone increasing function of λ for $0 < \lambda < \infty$, and we have

  $$d_{\min} \;=\; \Big(\alpha_1 {D_1}^{\lambda} + \alpha_2 {D_2}^{\lambda} \Big)^{1/\lambda} \quad \begin{cases} \uparrow \;\; \max(D_1, D_2) & \text{as} \quad \lambda \uparrow \infty \,, \\ \downarrow \;\; {D_1}^{\alpha_1} {D_2}^{\alpha_2} & \text{as} \quad \lambda \downarrow 0 \,. \end{cases}$$

(These facts generalize in the obvious manner to K distances $D_1, D_2, ..., D_K$ observed on the pair of objects.) While large distances win out in the limit for $\lambda \uparrow +\infty$, fixed small distances > 0 will never win out entirely for $\lambda \downarrow 0$, although for ever smaller λ the compromise will be shifted ever more toward the smaller distance.

Conclusion: *Embeddings that minimize B-C stress compromise ever more in favor of ...*
... larger distances as $\lambda \uparrow \infty$ or $\nu \uparrow \infty$, with full max-dominance in either limit;
... smaller distances as $\lambda \downarrow 0$ or $\nu \downarrow -\infty$, with full min-dominance only in the ν-limit.

We use the term "**small scale sensitivity**" for the behavior of stress functions as $\lambda \downarrow 0$ and/or $\nu \downarrow -\infty$. It has the effect of reinforcing local structure because object pairs with small target distances will preferentially be placed close together in the embedding. A related observation was made by Noack (2003) for $\lambda \downarrow 0$ in graph drawing and called "clustering strength"; this concept is not identical to small distance sensitivity, however; see Section 2.5.

2.5 Distances versus Weights

Noack (2009) presents a family of "energy functions" for weighted graphs/networks that should be discussed here because it might be thought to be identical to the B-C family of stress functions — which it isn't, though there exists a connection. The following discussion is meant to clarify the difference between specifying the relation among object pairs in terms of weights and in terms of distances.

Underlying the idea of mapping weighted graph data to graph drawings is a density paradigm. The intuition is that objects connected by edges with large weights should be represented by embedding points that are near each other so as to form high density areas. Hence large weights play a similar role as small distances in their intended effects on embeddings. Weights and distances are therefore in an inverse relation to each other, a fact that will be made precise below.

Next we follow Noack (2009) and consider data given as edge weights $w_{i,j} \geq 0$ for all pairs (i, j) with the interpretation that an edge in a graph "exists" between objects i and j if $w_{i,j} > 0$. (He also allows node weights w_i, but we set these to 1 as they add no essential freedom of functional form.) The family of energy functions he considers uses a general form of power laws for attracting and repulsing energies:

$$U(\boldsymbol{d}|\boldsymbol{W}) \;=\; \sum_{i,j=1,\ldots,N} \left(w_{i,j} \, \frac{{d_{i,j}}^{a+1}}{a+1} \;-\; \frac{{d_{i,j}}^{r+1}}{r+1} \right), \tag{4}$$

where we write $\boldsymbol{W} = (w_{i,j})_{i,j=1,\ldots,N}$. It is assumed that $a > r$ in order to grant finitely sized minimizing embeddings for connected graphs. In the spirit, though not the letter, of Box-Cox transforms, Noack imputes natural logarithms for $a + 1 = 0$ or $r + 1 = 0$. Unweighted graphs are characterized by $w_{i,j} \in \{0, 1\}$, in which case the total energy (4) amounts to (1) the sum of attracting energies limited to the edges in the graph, and (2) the sum of repulsing energies for *all* pairs of nodes. This functional form is suggested by traditional energy functions in graph drawing where an attracting force holds the embedding points $\mathbf{x}_i$ and $\mathbf{x}_j$ together if there exists an edge between them and where the repulsing force is pervasive and exists for all pairs so as to disentangle the embedding points by spreading them out.

We now ask how the energy functions (4) and the B-C stress functions (1) relate to each other. A simple answer can be given by drawing on the notion of edgewise unbiasedness: in a two-node situation with single weight w, find the embedding distance $d_{\min}$ that minimizes the

energy function (4); this distance $d_{\min} = d(w)$ can be interpreted as the target distance D for which the energy function is edgewise unbiased. Thus the canonical relation between weights and target distances is $D = d(w)$. For an energy function (4) the specialization to two nodes is $U = w\, d^{a+1}/(a+1) - d^{r+1}/(r+1)$, whose stationarity condition is $U' = wd^a - d^r = 0$, hence $w = 1/d^{a-r}$ and $d(w) = 1/w^{1/(a-r)}$, as noted by Noack (2009), eq. (3). Thus the correspondence between w and its edgewise unbiased target distance D is

$$D \;=\; \frac{1}{w^{1/(a-r)}}. \tag{5}$$

Using the translation $w_{i,j} = {D_{i,j}}^{-(a-r)}$ and the convention $w_{i,j} = 0 \Rightarrow D_{i,j} = +\infty \Rightarrow {D_{i,j}}^{-(a-r)} = 0$, we can rewrite the energy function (4) modulo irrelevant constants as

$$U(\boldsymbol{d}|\boldsymbol{D}) \;\sim\; \sum_{i,j=1,\ldots,N} \Big({D_{i,j}}^{-(a-r)}\, BC_{a+1}(d_{i,j}) \;-\; BC_{r+1}(d_{i,j}) \Big) \tag{6}$$

A comparison with (1) shows that the 2-parameter familiy of energy functions (6) forms a subfamily of the 3-parameter family of distance-based B-C stress functions (1) as follows:

$$\nu = -(a-r), \quad \mu = r+1, \quad \lambda = a-r.$$

Thus the essential constraint is that $\lambda = -\nu$, entailing $\nu < 0$. In light of the results of Section 2.4 this constraint implies a counterbalancing of distance sensitivities implied by these parameters: as $\lambda \uparrow \infty$ large distance sensitivity increases, but simultaneously $\nu = -\lambda \downarrow -\infty$ and hence small scale sensitivity increases as well. Full clarity of the interplay is gained by repeating the exercise of Section 2.4 in the case $\nu = -\lambda$: Given two target distances D_1 and D_2 for $N = 2$ objects, the minimizing distance is obtained by specializing (3) to $\nu = -\lambda$:

$$d_{\min} \;=\; \frac{1}{\left(\frac{1}{2}{D_1}^{-\lambda} + \frac{1}{2}{D_2}^{-\lambda}\right)^{1/\lambda}} \quad \begin{cases} \downarrow\; \min(D_1, D_2) & \text{as} \quad \lambda \uparrow \infty\,, \\ \uparrow\; \sqrt{D_1 D_2} & \text{as} \quad \lambda \downarrow 0\,. \end{cases}$$

Thus the minimizing distance $d_{\min}$ is the reciprocal of the Lebesgue L_λ norm of the vector $({D_1}^{-1}, {D_2}^{-1})$ with regard to a uniform distribution $\alpha_1 = \alpha_2 = 1/2$. The identification $\nu = -\lambda$ has therefore a considerable degree of small scale sensitivity for all values of $\lambda > 0$, and counter-intuitively it increases with increasing λ: apparently the increasing small scale sensitivity incurred from the parameter $\nu \downarrow -\infty$ outweighs the diminished small scale sensitivity due to $\lambda \uparrow +\infty$.

It follows that Noack's (2003) notion of "clustering strength" is not identical to our notion of small scale sensitivity because clustering strength increases for $\lambda = -\nu \downarrow 0$. Rather, clustering strength has to do with the implied translation of a fixed weight w to a target distance $D = 1/w^{1/\lambda}$ according to (5): relatively large weights w will result in relatively ever smaller target distances D as $\lambda \downarrow 0$, thus reinforcing the clustering effect by the simple translation $w \mapsto D$. Diminishing small scale sensitivity for $\lambda = -\nu \downarrow 0$ is a lesser effect by comparison.

2.6 B-C Stress Functions for Incomplete Distance Data or Distance Graphs

In order to arrive at stress functions for non-full graphs, we extend a device we used previously to transform Kruskal-Shepard MDS into a localized or graph version called "local MDS" or "LMDS" (Chen and Buja 2009). We now assume target distances $D_{i,j}$ are given only for edges $(i,j) \in E$ in a graph. Starting with stress functions (1) for full graphs, we replace the dissimilarities $D_{i,j}$ for non-edges $(i,j) \notin E$ with a single large dissimilarity D_∞ which we let go to infinity. We down-weight these terms with a weight w in such a way that ${wD_\infty}^{\lambda+\nu} = t^{\lambda+\nu}$ is constant:

$$\begin{aligned} S &= \sum_{(i,j)\in E} {D_{i,j}}^{\nu} \Big(BC_{\mu+\lambda}(d_{i,j}) \;-\; {D_{i,j}}^{\lambda}\, BC_{\mu}(d_{i,j}) \Big) \\ &+ \; w \sum_{(i,j)\notin E} {D_\infty}^{\nu} \Big(BC_{\mu+\lambda}(d_{i,j}) \;-\; {D_\infty}^{\lambda}\, BC_{\mu}(d_{i,j}) \Big) \end{aligned}$$

As $D_\infty \to \infty$, we have $w = (t/D_\infty)^{\nu+\lambda} \to 0$ and ${wD_\infty}^{\nu} \to 0$, hence in the limit we obtain:

$$S \;=\; \sum_{(i,j)\in E} {D_{i,j}}^{\nu} \Big(BC_{\mu+\lambda}(d_{i,j}) \;-\; {D_{i,j}}^{\lambda}\, BC_{\mu}(d_{i,j}) \Big) \;-\; t^{\nu+\lambda} \sum_{(i,j)\notin E} BC_{\mu}(d_{i,j})\,. \qquad (7)$$

This procedure justifies wiping out the attracting energy outside the graph. We call (7) the *B-C family of stress functions* for distance graphs. The parameter t balances the relative strength of the combined attraction and repulsion inside the graph with the repulsion outside the graph. For completeness, we list the assumed ranges of the parameters:

$$t \geq 0\,, \quad \lambda > 0\,, \quad -\infty < \mu < \infty\,, \quad -\infty < \nu < \infty.$$

An interesting variation of the idea of pervasive repulsion is proposed by Koren and Civril (2009) who use finite rather than limiting energies.

2.7 An Irrelevant Constant: Weighting the Attraction

Noack (2003, Sec. 5.5) observed that for his LinLog energy function the relative weighting of the attracting energy relative to the repulsing energy is irrelevant in the sense that such weighting would only change the scale of the minimizing layout but not the shape. A similar statement can be made for all members of the B-C family of stress functions. To demonstrate this effect, we introduce B-C stress functions whose attraction is weighted by a factor c^λ ($c > 0$):

$$S_c(\boldsymbol{d}) \;=\; \sum_{(i,j)\in E} {D_{i,j}}^{\nu} \Big(c^{\lambda}\, BC_{\mu+\lambda}(d_{i,j}) \;-\; {D_{i,j}}^{\lambda}\, BC_{\mu}(d_{i,j}) \Big) \;-\; t^{\nu+\lambda} \sum_{(i,j)\notin E} BC_{\mu}(d_{i,j})\,,$$

Choice of Parameters	Special Cases
$E = V^2,\ \lambda = 1,\ \mu = 1,\ \nu = 0$	MDS (Kruskal 1964a; Kruskal & Seery 1980)
$E = V^2,\ \lambda = 2,\ \mu = 2,\ \nu = 0$	ALSCAL (Takane, Young and de Leeuw 1977)
$E = V^2,\ \lambda = 1,\ \mu = 1,\ \nu = -2$	Kamada & Kawai (1989)
$E = V^2,\ \lambda = 1,\ \mu = 1,\ \nu = -1$	Sammon (1969)
$E \subset V^2,\ \lambda = 1,\ \mu = 1,\ \nu = 0, \quad t > 0$	LMDS (Chen and Buja 2009)
$E \subset V^2,\ \lambda = 3,\ \mu = 0, \quad D_{i,j} = 1,\ t = 1$	Fruchterman & Reingold (1991)
$E \subset V^2,\ \lambda = 4,\ \mu = -2, D_{i,j} = 1,\ t = 1$	Davidson & Harel (1996)
$E \subset V^2,\ \lambda = 1,\ \mu = 0, \quad D_{i,j} = 1,\ t = 1$	Noack's LinLog (2003)
$E \subset V^2,\ \lambda = 1,\ \mu = 1, \quad D_{i,j} = 1,\ t = 1$	Noack's QuadLin (2003)
$E \subset V^2,\ \lambda > 0,\ \mu = 0, \quad D_{i,j} = 1,\ t = 1$	Noack's PolyLog family (2003; his $r = \lambda$)

Table 1: *Some special cases of stress functions and their parameters in the B-C family. The first four entries refer to stress functions for complete distance data; the last five entries refer to energy functions for plain graphs (in which case $D_{i,j} = 1$ for all edges and hence ν is vacuous). LMDS applies to incomplete distance data or distance graphs, as do all members of the B-C family. (Not included is Noack's (2009) family of power laws for weighted graphs because they become stress functions for distance graphs only after a mapping of weights to distances.)*

where $\boldsymbol{d} = (d_{i,j})$ is the set of all configuration distances for all pairs (i, j), including those not in the graph E. The repulsion terms are still differentially weighted depending on whether (i, j) is an edge of the graph E or not, which is in contrast to most energy functions proposed in the graph layout literature where invariably $t = 1$.

In analogy to Noack's argument, we observe the following form of scale equivariance:

$$S_1(c\,\boldsymbol{d}) \;=\; c^{\mu}\; S_c(\boldsymbol{d}) \;+\; \text{const} \tag{8}$$

As a consequence, if $\boldsymbol{d}$ is a minimizing set of configuration distances for $S_c(\cdot)$, then the distances $c\,\boldsymbol{d}$ of the scaled embedding $c\,\mathbf{X}$ minimize the original unweighted B-C stress function $S_1(\cdot)$.

It is in this sense that Noack's PolyLog family of stress functions can be considered as a special case of the B-C family: PolyLog energies agree with B-C stress functions for unweighted graphs ($D_{i,j} = 1$) for $\mu = 0$ and $t = 1$ up to a multiplicative factor in the attracting energy.

2.8 Unit-Invariant Forms of the Repulsion Weight

In the B-C family of stress functions (7), the relative strength of attracting and repulsing forces is balanced by the parameter t. This parameter, however, has two deficiencies: (1) It suffers from a lack of invariance under a change of units in the target distances $D_{i,j}$; (2) it has stronger effects in sparse graphs than dense graphs because the number of terms in the summations over E and $V \setminus E$ vary with the size of the graph E. Both deficiencies can be corrected by reparametrizing t in terms of a new parameter τ as follows:

$$t^{\lambda+\nu} \;=\; \frac{|E|}{|V^2| - |E|} \cdot \left(\mathrm{median}_{(i,j)\in E}\; D_{i,j}\right)^{\lambda+\nu} \cdot \tau^{\lambda+\nu} \,. \tag{9}$$

This new parameter τ is unit free and adjusted for graph size. (Obviously the median can be replaced with any other statistic $S(D)$ that is positively homogeneous of first order: $S(cD) = cS(D)$ for $c > 0$.) These features enable us to formulate past experience in a problem-independent fashion as follows: in the examples we have tried, $\tau = 1$ has yielded satisfactory results. In light of this experience, there may arise few occasions in practice where there is a need to tune τ. As users work with different units in $D_{i,j}$ or different neighborhood sizes when defining NN-graphs, the recommendation $\tau = 1$ stands. Just the same, we will illustrate the effect of varying τ in an artificial example (Section 4.1).

3 Meta-Criteria For Parameter Selection

Following Chen and Buja (2009) and Akkucuk and Carroll (2006), we describe "meta-criteria" to measure the quality of configurations independently of the primary stress functions. The main purpose of these meta-criteria is to guide the selection of parameters such as those in the B-C family, λ, μ and τ. The idea is to compare "input neighborhoods" defined in terms of $D_{i,j}$ with "output neighborhoods" defined in terms of $d_{i,j}$ by measuring the size of their overlaps. Such neighborhoods are typically constructed as K-NN sets or, less frequently, in metric terms as ϵ-neighborhoods. In a dimension reduction setting one may define for the i'th point the input neighborhood $\mathcal{N}_D(i)$ as the set of K-NNs with regard to $D_{i,j}$ and similarly the ouput neighborhood $\mathcal{N}_d(i)$ as the set of K-NNs with regard to $d_{i,j}$. In an unweighted graph setting, one may define $\mathcal{N}_D(i)$ as the metric $\epsilon = 1$ neighborhood, that is, the set of points connected with the i'th point in the graph E, and hence the neighborhood size $K(i) = |\mathcal{N}_D(i)|$ is the degree of the i'th point in the graph E and will vary from point to point. The corresponding output neighborhood $\mathcal{N}_d(i)$ can then be defined as the $K(i)$-NN set with regard to $d_{i,j}$. The pointwise meta-criterion at the i'th point is defined as size of the overlap between $\mathcal{N}_d(i)$ and $\mathcal{N}_D(i)$, hence it is in frequency form

$$N_d(i) \;=\; |\mathcal{N}_d(i) \cap \mathcal{N}_D(i)| \,,$$

and in proportion form, using $|\mathcal{N}_D(i)|$ as the baseline,

$$M_d(i) \;=\; \frac{|\mathcal{N}_d(i) \cap \mathcal{N}_D(i)|}{|\mathcal{N}_D(i)|}\,.$$

The global meta-criteria are simply the averages over all points:

$$N_d \;=\; \frac{1}{|V|}\sum_i N_d(i) \qquad \text{and} \qquad M_d = \frac{1}{|V|}\sum_i M_d(i)\,.$$

Only when all input neighborhood sizes are equal, $|\mathcal{N}_D(i)| = K$, is there a simple relationship between N_d and M_d: $M_d = \frac{1}{K}N_d$. We subscript these quantities with d because they serve to compare different outputs $(\mathbf{x}_i)_{i=1\ldots N}$ (configurations, embeddings, graph drawings), but all that is used are the interpoint distances $d_{i,j} = \|\mathbf{x}_i - \mathbf{x}_j\|$. The proportion form M_d is obviously advantageous because it allows comparisons across different K (or ϵ).

Whether the meta-criterion values are small or large should be judged not against their possible ranges ($[0,1]$ for M_d) but against the possibility that $d_{i,j}$ (hence the embedding) and $D_{i,j}$ are entirely unrelated and generate only random overlap in their respective neighborhoods $\mathcal{N}_d(i)$ and $\mathcal{N}_D(i)$. The expected value of random overlap is not zero, however; rather, it is $\mathrm{E}[|\mathcal{N}_d(i) \cap \mathcal{N}_D(i)|] = |\mathcal{N}_d(i)| \cdot |\mathcal{N}_D(i)|/(|V|-1)$ because random overlap should be modeled by a hypergeometric distribution with $|\mathcal{N}_D(i)|$ "defectives" and $|\mathcal{N}_d(i)|$ "draws" from a total of $|V|-1$ "items." The final adjusted forms of the meta-criteria are therefore:

$$\begin{aligned}
N_d^{adj}(i) \;&=\; |\mathcal{N}_d(i) \cap \mathcal{N}_D(i)| \;-\; \frac{1}{|V|-1}\,|\mathcal{N}_d(i)| \cdot |\mathcal{N}_D(i)|\,,\\
M_d^{adj}(i) \;&=\; \frac{|\mathcal{N}_d(i) \cap \mathcal{N}_D(i)|}{|\mathcal{N}_D(i)|} \;-\; \frac{1}{|V|-1}\,|\mathcal{N}_d(i)|\,,\\
N_d^{adj} \;&=\; \frac{1}{|V|}\sum_i N_d^{adj}(d)\,, \qquad M_d^{adj} \;=\; \frac{1}{|V|}\sum_i M_d^{adj}(d)\,.
\end{aligned}$$

When the neighborhoods are all K-NN sets, $|\mathcal{N}_d(i)| = |\mathcal{N}_D(i)| = K$, these expressions simplify:

$$\begin{aligned}
N_d^{adj}(i) \;&=\; |\mathcal{N}_d(i) \cap \mathcal{N}_D(i)| \;-\; \frac{K^2}{|V|-1}\,,\\
M_d^{adj}(i) \;&=\; \frac{|\mathcal{N}_d(i) \cap \mathcal{N}_D(i)|}{K} \;-\; \frac{K}{|V|-1} \;=\; \frac{N_d^{adj}(i)}{K}\,,\\
N_d^{adj} \;&=\; N_d \;-\; \frac{K^2}{|V|-1}, \qquad M_d^{adj} \;=\; M_d \;-\; \frac{K}{|V|-1} \;=\; \frac{N_d^{adj}}{K}\,.
\end{aligned}$$

An important general observation is that if the neighborhoods are defined as K-NN sets, the meta-criteria are invariant under monotone transformations of both inputs $D_{i,j}$ and outputs $d_{i,j}$. Methods that have this invariance are called "non-metric" in proximity analysis/multidimensional scaling because they depend only on the ranks and not the actual values of the distances.

In what follows, we will report M_d^{adj} for each configuration shown in the figures, and we will also use the pointwise values $M_d(i)$ as a diagnostic by highlighting points with $M_d(i) < 1/2$ as problematic in some of the figures.

Remark: Venna and Kaski (2006, and references therein) introduce an interesting distinction between "trustworthiness" and "continuity" measurement. In our notation the points in $\mathcal{N}_d(i) \setminus \mathcal{N}_D(i)$ violate trustworthiness because they are shown near but are not near in truth (near = being in the $K(i)$-NN), whereas the points in $\mathcal{N}_D(i) \setminus \mathcal{N}_d(i)$ violate continuity because they are near in truth but not shown as near. Venna and Kaski (2006) measure both violations separately based on distance-ranks. We implicitly also measure both, but more crudely by unweighted counting of violations. It turns out, however, that the two violation counts are the same: $|\mathcal{N}_d(i) \setminus \mathcal{N}_D(i)| = |\mathcal{N}_D(i) \setminus \mathcal{N}_d(i)| = K(i) - |\mathcal{N}_d(i) \cap \mathcal{N}_D(i)|$. Thus our meta-criterion is simultaneously a measure of trustworthiness and of continuity. — Lee and Verleysen (2008) introduce a larger class of potentially interesting meta-criteria that include ours and Venna and Kaski's (2006) as special cases.

4 B-C Stress Functions Applied

4.1 Simulated Data

We introduced three parameters in the B-C stress functions for complete distance data, namely, λ, μ and ν, and a fourth parameter, τ, in the B-C stress functions for incomplete distance graph data. In this subsection, we will examine how three of the four parameters affect configurations in terms of their local and global structure by experimenting with an artificial example. We will simplify the task and eliminate the parameter ν by setting it to zero, so that the weight ${D_{i,j}}^{\nu} = 1$ disappears from the stress functions. The reason for doing so is that both parameters ν and λ play a role in determining scale sensitivity, and, while they are not redundant, the weighting power ν is the more dangerous of the two because it can single-handedly destabilize stress functions as $\nu \downarrow -\infty$ through unlimited outweighting of large distances. By comparison, the small scale sensitivity caused by small values of the parameter $\lambda > 0$ is limited as the analysis of Section 2.4 shows.

To illustrate the effects of the remaining parameters λ, μ and τ on embeddings, we constructed an artificial data example consisting of 83 points that form a geometric shape represented in Figure 2 (top). The design was inspired by a simulation example used by Trosset (2006). The distance between any pair of adjacent points is set to 1. To define an initial local graph, as input to the stress functions, we connected each point with its adjacent neighbors with distance 1 (Figure 2, bottom). That is, we used metric nearest neighborhoods with radius 1. Thus, interior points have node degree 4, corner points and connecting points have

node degree 2, and the remaining peripheral points have node degree 3. The geometry of this input graph is intended to represent three connected clusters with an internal structure that is relatively tight compared to the connecting structure.

We produced 2-D configurations using B-C stress functions, and to explore the effect of the parameters on the configurations, we varied each of the three parameters one at a time. For each combination of parameters, we used two different starting configurations: the inputs from Figure 2, and a random start, respectively. Starting from the true input configurations is of course not actionable in practice, but it serves a purpose: it demonstrates the biases and distortions implied by minimization of the stress function under the best of circumstances. Starting from a random configuration, on the other hand, gives indications about the stability of the solutions in terms of local minima, as well as the effort required to get from an uninformed starting configuration to a meaningful local minimum. (In practice, one never knows how truly optimal any configuration is that has been obtained by a numerical algorithm, and a better sense of the issue is often obtained only by analyzing the solutions obtained from multiple restarts.)

For starts from the input configurations, the results are shown in Figures 3, 5, and 7, and for starts from random configurations they are shown in Figures 4, 6, and 8, along with their M^{adj} values that measure the local faithfulness of the configuration. We also colored red the points whose neighborhood structure is not well preserved in terms of a proportion $< 1/2$ of shared neighbors between input and output configurations ($M(i) < 1/2$).

Parameter λ: We set $\mu = 0$ and $\tau = 1$ and let λ vary as follows: $\lambda = 5,\ 2,\ 1,\ 0.5$. The resulting configurations are shown in Figures 3 and 4. The overall observation from both figures is that for smaller λ the greater small-scale sensitivity causes the configurations to cluster more strongly. We also notice in both figures that M^{adj} increases with decreasing λ, which indicates local structure within clusters is better recovered when the clusters are well separated. To confirm this, we show a zoom on the nearly collapsed points in the bottom right configurations and observe the square structure in the input configuration is almost perfectly recovered. This indicates that by tuning λ properly the resulting configurations can reveal both macro structure in terms of relative cluster placement as well as micro structure within each cluster.

Parameter μ: In Figures 5 and 6, we examine the effect of μ. We fix $\lambda = 5$ and $\tau = 1$ and let μ vary as follows: $\mu = -1,\ 0,\ 1,\ 2$. An overall observation is that the larger μ, the more spread out are the points. This is consistent with the interpretation of μ as the power law of repulsion. With smaller μ (such as $\mu = -1$) the stress function flattens out as the distance increases (Figure 1) and the points are subject to very weak repulsion. The top left plots ($\mu = -1$) in both figures show that weak repulsion is suitable for generating locally faithful structure. However, the repulsion can be too weak to generate globally meaningful

structure, as illustrated by the configurations obtained from random starts (Figure 6). In the top left plot of Figure 6, the three clusters are not aligned properly due to the weak repulsion. With stronger repulsion the points are placed in a globally more correct position, as shown in bottom two plots in Figure 6, with a sacrifice of local faithfulness (as reflected by the lower value of M^{adj}). The distortion of local structure is not surprising considering the fact that the repulsion is stronger in the direction in which points line up, which in this case is the horizontal direction. By comparison, points are squeezed flat in the vertical direction because repulsion has no traction vertically.

Parameter τ: We fix $\lambda = 5$ and $\mu = 0$ and vary τ as follows: $\tau = 0.01,\ 1,\ 10^3,\ 10^5$. The configurations starting from the original design and from a random start are shown in Figures 7 and 8. Figure 7 shows that the configuration closest to the input configuration is achieved with relatively small τ ($\tau = 0.01$). This indicates that the B-C stress functions for small τ are quite successful in recreating local distances as they are supposed to. From a random start, however, the configuration can be easily trapped in a local minimum with small τ (Figure 8, top two plots), which indicates that the relative weight of repulsion to attraction controlled by τ plays an essential role in achieving a stable configuration. With relatively larger τ, the points are more spread-out and configurations reveal the underlying structure more faithfully, both locally and globally (bottom two plots, Figure 8).

4.2 Olivetti Faces Data

This dataset, published on Sam Roweis' website, contains 400 facial images of 40 people, 10 images for each person. All images are of size 64 by 64. Mathematically, each image can be represented by a long vector, each element of which records the light intensity of one pixel in the image. Given this representation, we treat each image as a data point lying in a 4096-dimensional space ($64 \times 64 = 4096$). For visualization purposes we reduce the dimension from 4096 to 2. As the 40 faces form natural groups, we would expect effective dimension reduction methods to show clusters in their configurations. If this expectation is correct, then this dataset provides an excellent test bed for the effect of the clustering power λ.

We first centered the data, a 400×4096 matrix, at their row means to adjust the brightness of the images to the same level. We constructed a pairwise distance matrix using Euclidean distances in the original high dimensional space R^{4096}. We then defined a local graph using 4-NN, that is, we connected each point to its four nearest neighbors. In the resulting graph five small components were disconnected from the main component of the graph. Each of them contained images from a single person, with 5 images for one person and 5 images for another four persons. Since the disconnected components are trivially pushed away from the main component in any embedding due to the complete absence of attraction, we discarded

them and kept the 355 images representing 36 people for further analysis. We created for each person a unique combination of color and symbol to code the points representing it.

Figure 9 shows 2D configurations generated by different stress functions (7) with different clustering powers, $\lambda = 2,\ 1,\ 2/3,\ 1/2$, while the other parameters are fixed at $\mu = 0$ and $\tau = 1$. For the largest value, $\lambda = 2$, we do not see any clusters; for $\lambda = 1$, we see some fuzzy clusters forming; as λ decreases the clusters become clearer. The colors and symbols show that these clusters are not artifactual but real, mostly representing the images of the same person. The configurations do not produce exactly 36 clusters: some images of different people are not quite distinguishable in the configurations and some images of the same person are torn apart. The former could really present similar images; the latter could be due to the placement of images in the random start. However, the overall impression is to confirm the clustering effect due to small scale sensitivity for small values of λ. An interesting observation is that the meta-criterion M_k^{adj} increases as the small scale sensitivity strengthens, which assures us of the faithfulness of local topology.

For comparison, Figure 10 shows 2D configurations generated from four popular dimension reduction methods: PCA, MDS, Isomap and LLE. PCA and MDS did not find any clusters. Isomap and LLE did reveal a few clusters, but not as many nor as clearly as some of those obtained from the BC-family, even though we tuned the neighborhood size for Isomap and LLE to achieve the best visualization. For example, we chose a different neighborhood size $K = 8$ for LLE and Isomap $K = 4$; LLE configurations degenerated to lines or lines and a big cluster when $K = 4$ and 6, respectively.

4.3 Frey Face Data

In the Olivetti data we were able to show successful use of the small scale sensitivity for small values of λ. In the present example, the Frey face data, we study the effect of μ and its interaction with λ. The data were originally published with the LLE article (Roweis and Saul, 2000). We studied its low dimensional configurations from various dimension reduction methods in Chen and Buja (2009). The data contains 1965 facial images of "Branden Frey," which are stills of a short video clip recorded when Frey was making different facial expressions. Each image is of size 20×28 which can be thought of as a data point in 560-dimensional space. In our experiments we use a subset of 500 images in order to save on computations and in order to obtain less cluttered low dimensional embeddings. The fact is that the intrinsic structure of the full dataset is well preserved in this subset, partly due to the inherent redundancies in video sequences: the images close in order are very similar because the stills are taken more frequently than Frey's facial expression changes.

In Figure 11 we show the first two principal components of the 3D configurations for

varying μ as λ and τ are fixed at one. The neighborhood size is $K = 6$. The coloring scheme is adapted from the LMDS configurations of Chen and Buja (2009) where the points were colored to highlight the clustering structure found in those configurations. The bottom left configuration with parameters $\lambda = \mu = 1$ is an LMDS configuration. We observe that the small scale sensitivity of a fixed value $\lambda = 1$ varies as μ varies. With smaller μ such as -1 and 0, the small scale sensitivity is most pronounced: we see bigger clusters in the configurations with larger μ are split into smaller ones. On the other hand, larger values such as $\mu = 1,\ 2$ clearly provide connectivity between clusters and therefore better capture the global structure in the data. The local neighborhood structure, though, is better reflected in the configurations with smaller values of μ, as suggested by the values of meta-criteria.

5 Summary and Discussion

Our work contributes to the literature on proximity analysis, nonlinear dimension reduction and graph drawing by systematizing the class of distance-based approaches whose commonality is that they generate embeddings (maps, configurations, graph drawings) of objects in such a way that given input distances between the objects are well-approximated by output distances in the embedding. The systematization consists of devising a multi-parameter family of stress functions that comprises many published proposals from the literatures on proximity analysis (MDS) and graph drawing. A benefit is that the seemingly arbitrary selection of a loss function is turned into a parameter selection problem based on external "meta-criteria" that measure the quality of embeddings independently of the stress functions.

The parameters of the proposed family have the following interpretations:

- λ: This parameter determines the relative strengths of the attracting and repulsing forces to each other, while maintaining "edgewise unbiasedness" of the stress function. In practical terms, this parameter strongly influences "small scale sensitivity": For decreasing λ, it increases the small-scale sensitivity, that is, the tendency to group together nearby points in the embedding. Range: $\lambda > 0$.
- μ: This parameter is the power law of the repulsing energy. The greater μ, the greater is the tendency to suppress large discrepancies between inputs $D_{i,j}$ and outputs $d_{i,j}$. Range: $-\infty < \mu < +\infty$.
- ν: This is a weighting parameter that allows up- and down-weighting of pairs of objects as a function of the input distance $D_{i,j}$. For example, as ν decreases below zero, stress terms for large distances $D_{i,j}$ will be progressively down-weighted. Range: $-\infty < \nu < +\infty$.

- τ: A regularization parameter that stabilizes configurations for incomplete distance data, that is, distance graphs, at the cost of some bias (stretching of configurations), achieved by imputing infinite input distances with infinitesimal repulsion. Range: $\tau > 0$.

The power laws for attracting and repulsing energies are interpreted as Box-Cox transformations, which has two benefits: (1) Box-Cox transformations encompass a logarithmic attracting law for $\mu + \lambda = 0$ and a logarithmic repulsing law for $\mu = 0$; (2) they permit negative powers for both laws because the Box-Cox transformations are monotone increasing for powers in the whole range of real numbers. — The regularization parameter τ plays a role only when the input distance matrix is incomplete, as in the case of a distance graph or in the case of localization by restricting the loss function to small scale (as in LMDS; Chen and Buja 2009).

The problem of incomplete distance information is often solved by completing it with additive imputations provided by the shortest-path algorithm, so that MDS-style stress functions can be used — the route taken by Isomap (Tenenbaum et al. 2000). The argument against such completion is that stress functions tend to be driven by the largest distances, which are imputed and hence noisy. Conversely the argument against not completing is that the use of pervasive repulsion to stabilize configurations amounts to imputation also, albeit of an uninformative kind. A full understanding of the trade-offs between completion and repulsion is currently lacking, but practitioners can meanwhile experiment with both approaches and compare them on their data. In both cases the family of loss functions proposed here offers control over the scale sensitivity parameter λ, the repulsion power μ, and the weighting power ν.

Another issue with distance-based approaches is that there is often much freedom in choosing the distances, in particular when applied to dimension reduction. There is therefore a need to systematize the choices and provide guidance for "distance selection."

6 APPENDIX: STRESS MINIMIZATION

Minimizing stress functions can be a very high-dimensional optimization problem involving all coordinates of all points in an embedding, amounting to Np parameters. For this reason, minimization algorithms tend to be based on simple gradient descent (Kruskal 1964b) or on majorization (Borg and Groenen 2005). We limit ourselves in this appendix to providing gradients, though with one innovation to solve the following problem: optimization of stress functions tends to spend much effort on getting the size of the embedding right, which is not only unnecessary but also may cause delay of convergence when in fact the shape of the embedding is already optimized, or misjudgement of convergence when the size has been gotten right but the shape has not. This appendix proceeds therefore in three steps: Section 6.1 provides gradients for plain stress functions as presented in the body of this article; Section 6.2 derives size-invariant versions of stress functions; Section 6.3 provides gradients for the latter. (In order to make the formulas more readable, we set the parameter ν to zero and hence ignore it; it would be a simple matter to put it back in the formulas.)

6.1 Gradients for Stress Functions

Let the $N \times p$ matrix $\mathbf{X} = (x_1, \cdots, x_N)^T$ represent the embedding consisting of n points in d dimensions. As always let $D_{i,j}$ be the input distances and $d_{i,j} = \|\mathbf{x}_i - \mathbf{x}_j\|$ the output distances. The B-C stress function for $\mu \neq 0$ and $\mu + \lambda \neq 0$ (but $\nu = 0$) is

$$S(\mathbf{x}_1, \cdots, \mathbf{x}_N) = \sum_{(i,j)\in E} \left(\frac{{d_{i,j}}^{\mu+\lambda} - 1}{(\mu+\lambda)} - {D_{i,j}}^{\lambda} \frac{{d_{i,j}}^{\mu} - 1}{\mu} \right) - t^{\lambda} \sum_{(i,j)\in E^C} \frac{{d_{i,j}}^{\mu} - 1}{\mu} \tag{10}$$

Let $\nabla S = (\nabla_1, \cdots, \nabla_N)^T$ be the gradient of the stress function with respect to $\mathbf{X}$:

$$\nabla_i = \frac{\partial S}{\partial \mathbf{x}_i} = \sum_{j\in\mathcal{N}_D(i)} \left({d_{i,j}}^{\mu+\lambda-2} - {D_{i,j}}^{\lambda} {d_{i,j}}^{\mu-2} \right) (\mathbf{x}_i - \mathbf{x}_j) - t^{\lambda} \sum_{j\in\mathcal{N}_D^c(i)} {d_{i,j}}^{\mu-2} (\mathbf{x}_i - \mathbf{x}_j)$$

Define a $N \times N$ matrix M as follows:

$$M_{ij} = \begin{cases} {d_{i,j}}^{\mu+\lambda-2} - {D_{i,j}}^{\lambda} {d_{i,j}}^{\mu-2} & \text{if } j \in E(i) \\ -t^{\lambda} {d_{i,j}}^{\mu-2} & \text{if } j \notin E(i) \end{cases}$$

Note that M is symmetric. The gradient can be simplified to

$$\begin{aligned}\nabla_i = \frac{\partial S}{\partial \mathbf{x}_i} &= \sum_j M_{ji}(\mathbf{x}_i - \mathbf{x}_j) \\ &= (\sum_j M_{ji})\mathbf{x}_i - \sum_j M_{ji} x_j \ ,\end{aligned}$$

and

$$\nabla S = \mathbf{X} * (M \cdot E) - M \cdot \mathbf{X} \ ,$$

where E is a $N \times d$ matrix with all elements being 1. The symbol '$*$' represents elementwise multiplication of the two matrices of the same size, and the symbol '$\cdot$' stands for regular matrix multiplication.

6.2 Size-Invariant Forms of B-C Stress Functions

As mentioned, it is a common experience that algorithms for minimizing stress functions spend much effort on getting the size of the embedding right. Size, however, is not of interest — shape is. We have therefore a desire to re-express stress in a manner that is independent of size. Fortunately, there exists a general method that achieves this goal: For any configuration, minimize stress with regard to size and replace the original stress with its size-minimized value. This works because the minimization with regard to size can be carried out explicitly with elementary calculus. The result is a new form of stress that is minimized by the same shapes as the original stress, but it is independent of size and hence purely driven by shape. The computational advantage of size-invariant stress is that gradient-based optimization descends along directions that change shape, not size. — We sketch the derivation (again for $\nu = 0$ for less unwieldy formulas).

It is convenient to collect the repulsion terms inside and outside the graph because they share the power law:

$$S \ = \ \sum_{(i,j)\in E} BC_{\mu+\lambda}(d_{i,j}) \ - \ \sum_{(i,j)\in V^2} \tilde{D}_{i,j}^{\lambda} \, BC_{\mu}(d_{i,j}) \, .$$

where

$$\tilde{D}_{i,j} = \begin{cases} D_{i,j}, & (i,j) \in E \\ t, & (i,j) \notin E \end{cases}$$

Next, consider a configuration $\mathbf{X} = (x_i)_{i=1\ldots N}$ and resized versions $s\mathbf{X} = (s\, x_i)_{i=1\ldots N}$ thereof ($s > 0$). The configuration distances scale along with size: $d_{i,j}(s\mathbf{X}) = s\, d_{i,j}(\mathbf{X})$. To find the stationary size factor s of the stress as a function of s, $S = S(s)$, we observe that

$$\frac{\partial}{\partial s} BC_{\mu}(s\, d_{i,j}) \ = \ s^{\mu-1} {d_{i,j}}^{\mu} \qquad (\forall \mu \in \mathbb{R}).$$

In particular, this holds even for $\mu = 0$. Next we solve the stationary equation and check second derivatives:

$$S(s\boldsymbol{d}) \;=\; \sum_E BC_{\mu+\lambda}(s\, d_{i,j}) \;-\; \sum_{V^2} \tilde{D}^{\lambda}_{i,j}\; BC_{\mu}(s\, d_{i,j})\,, \tag{11}$$

$$S'(s\boldsymbol{d}) \;=\; s^{\mu+\lambda-1}\, T_{den} \;-\; s^{\mu-1} T_{num}\,, \tag{12}$$

$$S''(s\boldsymbol{d}) \;=\; (\mu + \lambda - 1)\; s^{\mu+\lambda-2}\, T_{den} \;-\; (\mu-1)\, s^{\mu-1} T_{num}\,, \tag{13}$$

where $T_{den} = T_{den}(\boldsymbol{d})$ and $T_{num} = T_{num}(\boldsymbol{d})$ are defined for $\boldsymbol{d} = (d_{i,j})$ by

$$T_{den} = \sum_E {d_{i,j}}^{\mu+\lambda}\,, \qquad T_{num} = \sum_{V^2} \tilde{D}^{\lambda}_{i,j}\; {d_{i,j}}^{\mu},$$

and again (12) and (13) hold even for $\mu = 0$ and $\mu + \lambda = 0$. The stationary size factor s_* that satisfies $S'(s_*) = 0$ is

$$s_* \;=\; \left(\frac{T_{num}}{T_{den}}\right)^{\lambda} \qquad (\forall \mu \in \mathbb{R},\;\; \lambda > 0). \tag{14}$$

The factor s_* is a strict minimum:

$$S''(s_*\boldsymbol{d}) \;=\; \lambda \frac{T_{num}^{\lambda\mu+1-2\lambda}}{T_{den}^{\lambda\mu-2\lambda}} \;>\; 0 \qquad (\forall \mu \in \mathbb{R},\; \lambda > 0).$$

Evaluating $S(s_*)$ we arrive at a **size-invariant** yet **shape-equivalent** form of the stress function. For the evaluation we need to separate power laws from the two logarithmic cases:

$$S(s\boldsymbol{d}) \;\approx\; \begin{cases} \frac{1}{\mu+\lambda}\, s^{\mu+\lambda}\, T_{den} \;-\; \frac{1}{\mu}\, s^{\mu}\, T_{num} & (\mu+\lambda \neq 0,\; \mu \neq 0) \\ |E| \log(s) \;+\; \sum_E \log(d_{i,j}) \;-\; \frac{1}{\mu}\, s^{\mu}\, T_{num} & (\mu+\lambda = 0) \\ \lambda\, s^{\lambda}\, T_{den} \;-\; \sum_{V^2} \tilde{D}^{\lambda}_{i,j}\, (\log(s) + \log(d_{i,j})) & (\mu = 0) \end{cases}$$

where "$\approx$" means "equal up to additive constacts that are irrelevant for optimization." We calculate $\tilde{S} = S(s_*)$ separately in the three cases with s_* from (14):

- $\mu + \lambda \neq 0,\; \mu \neq 0$: Several algebraic simplifications produce the following.

$$\tilde{S} \;=\; \left(\frac{1}{\mu+\lambda} - \frac{1}{\mu}\right) \frac{T_{num}^{\lambda\mu+1}}{T_{den}^{\lambda\mu}} \;=\; \left(\frac{1}{\mu+\lambda} - \frac{1}{\mu}\right) \frac{\left(\sum_{V^2} \tilde{D}^{\lambda}_{i,j}\, {d_{i,j}}^{\mu}\right)^{\lambda\mu+1}}{\left(\sum_E {d_{i,j}}^{\mu+\lambda}\right)^{\lambda\mu}}\,. \tag{15}$$

  [Note that $\tilde{S}$ gets minimized, hence the ratio on the right gets minimized when the left factor is positive (i.e., $\mu < 0 < \mu + 1/\lambda$), and it gets maximized when the left factor is negative (i.e., $\mu > 0$ or $\mu + 1/\lambda < 0$).]

- $\mu + \lambda = 0$: We take advantage of the fact that $T_{den} = |E|$ and $\mu = -\lambda$.

$$\tilde{S} \;\approx\; |E|\, \lambda\, \log \sum_{V^2} \left(\frac{\tilde{D}_{i,j}}{d_{i,j}}\right)^{\lambda} \;+\; \sum_E \log(d_{i,j})$$

- $\mu = 0$: We take advantage of the fact that $T_{den} = \sum_E {d_{i,j}}^\lambda$ and also that $T_{num} = \sum_{V^2} \tilde{D}_{i,j}^\lambda$ is a constant for optimization with regard to $d = (d_{i,j})$, and any additive term that is just a function of $\tilde{D}_{i,j}$ but not $d_{i,j}$ can be neglected.

$$\tilde{S} \approx \lambda \left(\sum_{V^2} \tilde{D}_{i,j}^\lambda \right) \log(\sum_E {d_{i,j}}^\lambda) - \sum_{V^2} \tilde{D}_{i,j}^\lambda \log(d_{i,j})$$

Even though size-invariance holds by construction, one checks it easily in all three cases: $\tilde{S}(s\, d) = \tilde{S}(d)$.

6.3 Gradients for Size-Invariant Stress Functions

To describe the gradient of the size-invariant stress function $\tilde{S}\,(\boldsymbol{d})$. We only consider the case $\mu + \lambda \neq 0$ and $\mu \neq 0$, which is shown in equation (15).

Let $\nabla S = ((\nabla S)_1, ..., (\nabla S)_n)^T$ be the gradient the $\tilde{S}\,(d)$ with respect to configuration $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_n)^T$

$$(\nabla S)_i = \left(\frac{1}{\mu + \lambda} - \frac{1}{\mu} \right) \left[(\lambda\mu + 1) \left(\frac{T_{num}}{T_{den}} \right)^{\lambda\mu} (\nabla T_{num})_i - (\lambda\mu) \left(\frac{T_{num}}{T_{den}} \right)^{\lambda\mu+1} (\nabla T_{den})_i \right]$$

$$(\nabla T_{num})_i = \frac{\partial T_{num}\,(d)}{\partial \mathbf{x}_i} = \mu \sum_j \tilde{D}_{i,j} d_{i,j}^{\mu-2} (\mathbf{x}_i - \mathbf{x}_j)$$

$$(\nabla T_{den})_i = \frac{\partial T_{den}\,(d)}{\partial \mathbf{x}_i} = (\mu + \lambda) \sum_{j \in E(i)} d_{i,j}^{\mu+\lambda-2} (\mathbf{x}_i - \mathbf{x}_j)$$

Plug $(\nabla T_{num})_i$ and $(\nabla T_{den})_i$ into the $(\nabla S)_i$

$$\begin{aligned}(\nabla S)_i &= \left(\frac{T_{num}}{T_{den}} \right)^{\lambda\mu} \left(\frac{T_{num}}{T_{den}} \sum_{j \in E(i)} d_{i,j}^{\mu+\lambda-2} (\mathbf{x}_i - \mathbf{x}_j) - \sum_j \tilde{D}_{i,j} d_{i,j}^{\mu-2} (\mathbf{x}_i - \mathbf{x}_j) \right) \\ &= \left(\frac{T_{num}}{T_{den}} \right)^{\lambda\mu} \left(\sum_{j \in E(i)} \left(\frac{T_{num}}{T_{den}} d_{i,j}^{\mu+\lambda-2} - D_{ji}^\lambda d_{i,j}^{\mu-2} \right) (\mathbf{x}_i - \mathbf{x}_j) - \sum_{j \in E^c(i)} t\, d_{i,j}^{\mu-2} (\mathbf{x}_i - \mathbf{x}_j) \right)\end{aligned}$$

Define a $N \times N$ matrix M by

$$M_{ij} = \left\{ \begin{array}{c} \left(\frac{T_{num}}{T_{den}} d_{i,j}^{\mu+\frac{1}{\lambda}-2} - {D_{i,j}}^\lambda d_{i,j}^{\mu-2} \right) \text{ for } j \in E\,(i) \\ t\, d_{i,j}^{\mu-2} \text{ for } j \notin E\,(i) \end{array} \right\}$$

The gradient can be simplified to

$$\begin{aligned}(\nabla S)_i &= \left(\frac{T_{num}}{T_{den}} \right)^{\lambda\mu} \sum_j M_{ij} (\mathbf{x}_i - \mathbf{x}_j) \\ &= \left(\frac{T_{num}}{T_{den}} \right)^{\lambda\mu} \left(\sum_j M_{ij} \mathbf{x}_i - \sum_j M_{ij} \mathbf{x}_j \right)\end{aligned}$$

and

$$\nabla S = \left(\frac{T_{num}}{T_{den}}\right)^{\lambda\mu} (\mathbf{X} * (M \cdot E) - M \cdot \mathbf{X})$$

We did the calculation separately for $\mu = 0$ and $\lambda + \mu = 0$ which resulted in the following:

$$\mu = 0: \qquad \tilde{S}(d) \sim \lambda \left(\sum_{(i,j)\in V^2} \tilde{D}_{ij} \right) \log \left(\sum_{(i,j)\in E} {D_{i,j}}^{\lambda} \right) - \sum_{(i,j)\in V^2} \tilde{D}_{ij} \log D_{i,j}$$

$$\mu + \lambda = 0: \qquad \tilde{S}(d) \sim \lambda \left(\sum_{(i,j)\in E} \right) \log \left(\sum_{(i,j)\in V^2} \tilde{D}_{ij} {D_{i,j}}^{\mu} \right) + \sum_{(i,j)\in E} \log D_{i,j}$$

REFERENCES

Akkucuk, U. and Carroll, J.D., 2006, PARAMAP vs. Isomap: A Comparison of Two Nonlinear Mapping Algorithms. *Journal of Classification*, **23** (2), 221-254.

Borg, I., and Groenen, P. 2005, *Modern Multidimensional Scaling: Theory and Applications*, New York: Springer-Verlag.

Chen, L., 2006, Local Multidimensional Scaling for Nonlinear Dimension Reduction, Graph Layout and Proximity Analysis. Ph.d. Thesis, University of Pennsylvania.

Chen, L. and Buja, A., 2009, Local Multidimensional Scaling for Nonlinear Dimension Reduction, Graph Drawing and Proximity Analysis. *Journal of American Statistical Association*, **104** (485), 209-219.

Davidson, R. and Harel, D., 1996, Drawing Graphs Nicely Using Simulated Annealing. *ACM Transactions on Graphics*, **15**(4), 301-331.

Fruchterman, T. M. J., and Reingold, E. M., 1991, Graph Drawing by Force-Directed Placement. *Software-Practice and Experience*, **21** (11), 1129-1164.

Gansner, E., Koren, Y., and North, S., 2004, Graph Drawing by Stress Majorization. *Graph Drawing*, 239-250.

Kamada, T., and Kawai, S., 1989, An Algorithm for Drawing General Undirected Graphs. *Information Processing Letters*, **31**, 7-15.

Koren, Y., and Civril, A., 2009, The Binary Stress Model for Graph Drawing, *Proceedings of the 16th International Symposium in Graph Drawing* (GD' 08), Vol. 5417, Springer Lecture Notes in Computer Science.

Kruskal, J. B., 1964a, Multidimensional Scaling by Optimizing Goodness of Fit to A Nonmetric Hypothesis. *Psychometrika*, **29**, 1-27.

Kruskal, J. B., 1964b, Nonmetric Multidimensional Scaling: A Numerical Method, *Psychometrika*, **29**, 115-129.

Kruskal, J. B. and Seery, J. B., 1980, Designing Network Diagrams. Technical Memorandum, Bell Laboratories, Murray Hill, NJ.

Lee, A. J., and Verleysen, M. (2008), Rank-Based Quality Assessment of Nonlinear Dimensionality Reduction. *ESANN '2008* (European Symposium on Artificial Neural Networks), proceedings, 49-54, ISBN 2-930307-08-0.

Lu, F., Keles, S., Wright, S. J., and Wahba, G., 2005, Framework for kernel regularization with application to protein clustering, *PNAS* **102**, (35), 12332-12337

Noack, A., 2003, Energy Models for Drawing Clustered Small-World Graphs. Computer Science Reports, I-07/2003, Brandenburg Technical University at Cottbus, Germany.

Noack, A., 2007, Energy Models for Graph Clustering, *Journal of Graph Algorithms and Applications*, **11**(2), 453-480.

Noack, A., 2009, Modularity Clustering is Force-Directed Layout. *Physical Review E*, **79** (2), 026102.

Roweis, S. T. and Saul, L. K., 2000, Nonlinear Dimensionality Reduction by Local Linear Embedding. *Science*, **290**, 2323-2326.

Sammon, J., 1969, A Nonlinear Mapping for Data Structure Analysis. *IEEE Transactions on Computers*, **C-18** (5), 401-409.

Schölkopf, B., Smola, A. J., and Müller, K.-R., 1998, Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Computation*, **10**, 1299-1319.

Saul, L. K. and Roweis, S. T., 2003, Think Globally, Fit Locally: Unsupervised Learning of Low Dimensional Manifolds. *J. of Machine Learning*, **4**, 119-155.

Takane, Y., Young, F. W. and de Leew, J., 1977, Nonmetric Individual Differences Multidimensional Scaling: An Alternating Least Squares Method with Optimal Scaling Features. *Psychometrika*, **42**, 7-67.

Tenenbaum, J. B., Silva, V. d., and Langford, J. C., 2000, A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, **290**, 2319-2323.

Trosset, M.W., 2006, Classical Multidimensional Scaling and Laplacian Eigenmaps. Presentation given at the 2006 Joint Statistical Meeting (Session 411), Seattle, WA.

Venna, J., and Kaski, S., 2006, Local Multidimensional Scaling. *Neural Networks*, **19**, 889-899.

Weinberger, K. Q., Sha, F., Zhu, Q., and Saul, L. K., 2006, Graph Laplacian Regularization for Large-Scale Semidefinite Programming. *NIPS 2006*, 1489-1496.

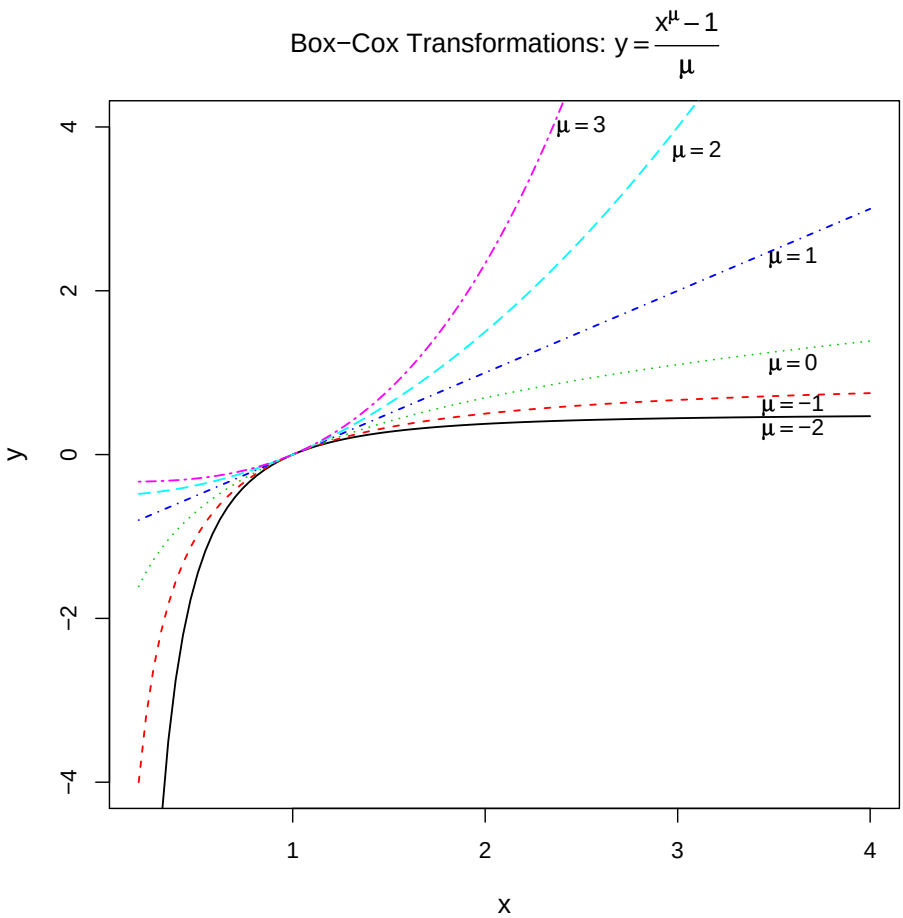

Figure 1: *Box-Cox Transformations:* $y = \frac{x^{\mu}-1}{\mu}$

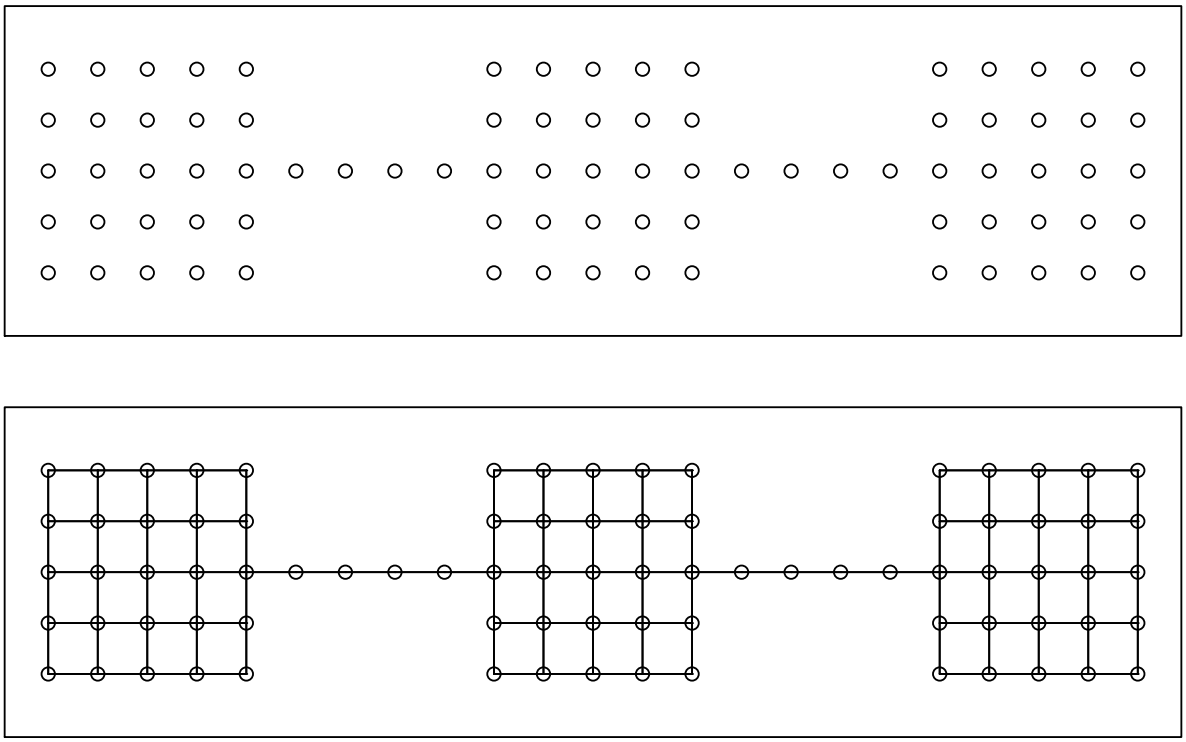

Figure 2: *The original configuration (top) and initial local graph (bottom)*

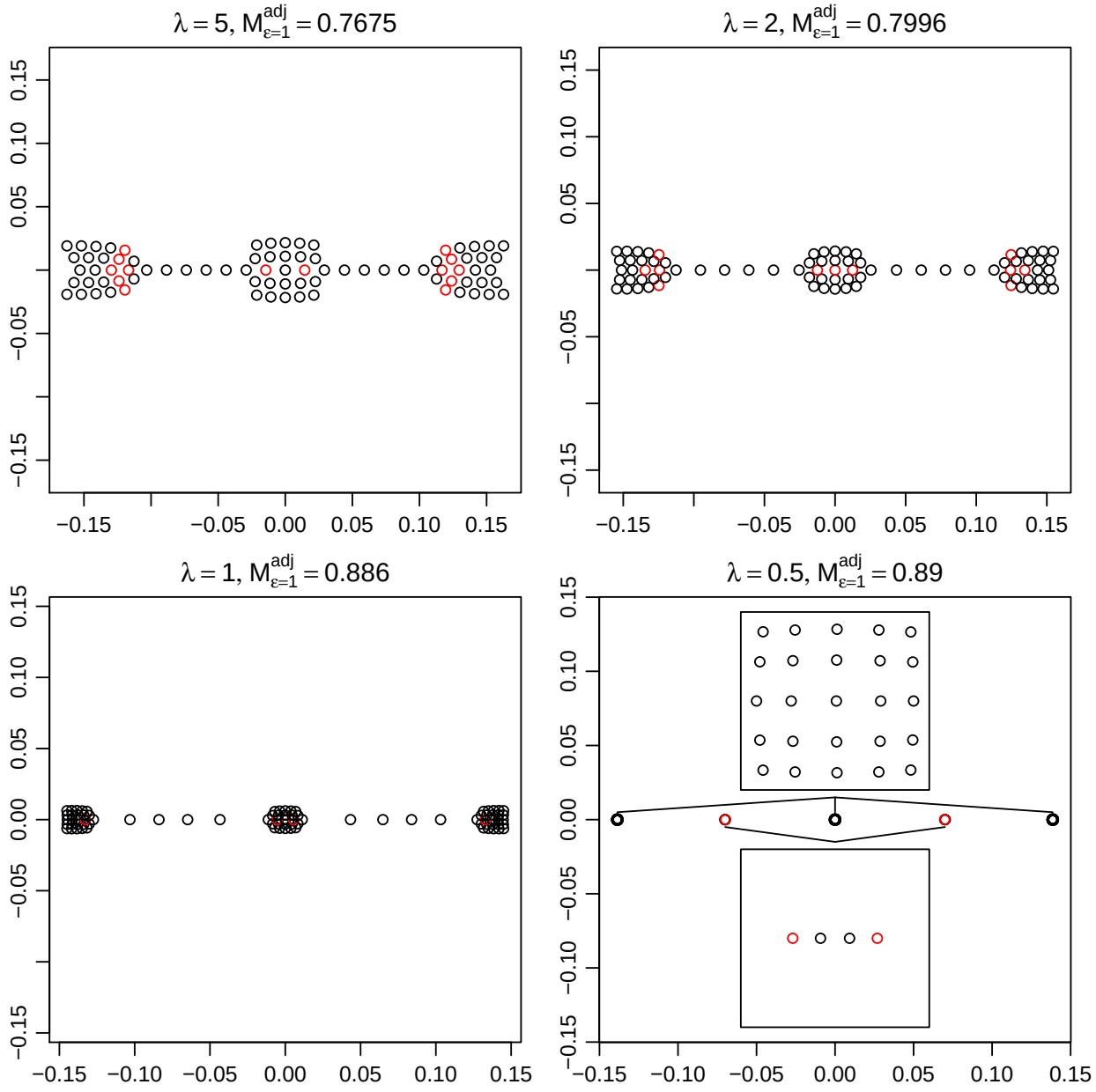

Figure 3: *The configurations with varying λ with other parameters fixed at $\mu = 0,\ \tau = 1$, starting from the input configuration (Figure 2).*

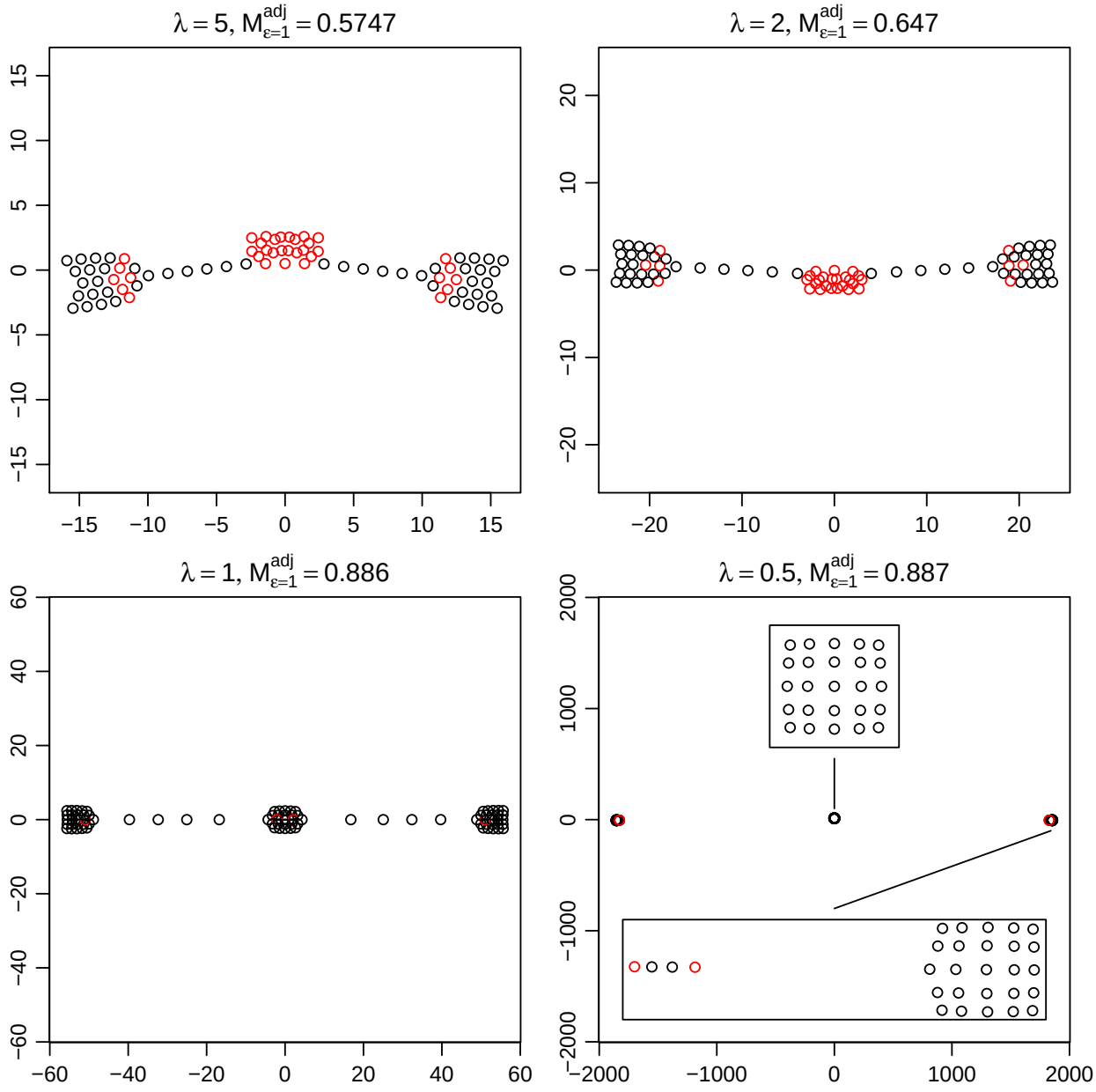

Figure 4: *The configurations with varying λ with other parameters fixed at $\mu = 0,\ \tau = 1$, starting from a random configuration.*

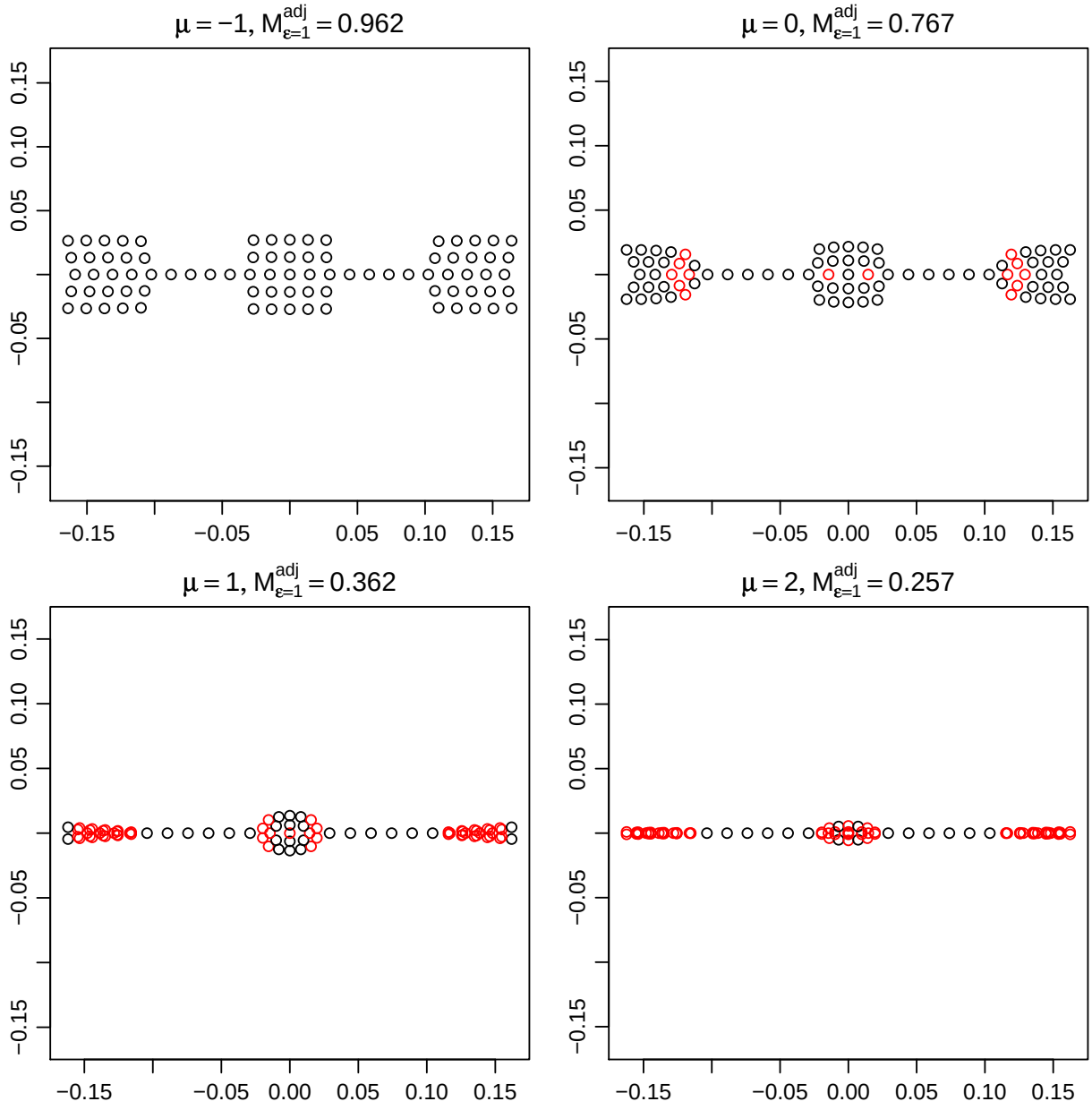

Figure 5: *The configurations with varying μ with other parameters fixed at $\lambda = 5,\ \tau = 1$, starting optimization from the input configuration (Figure 2).*

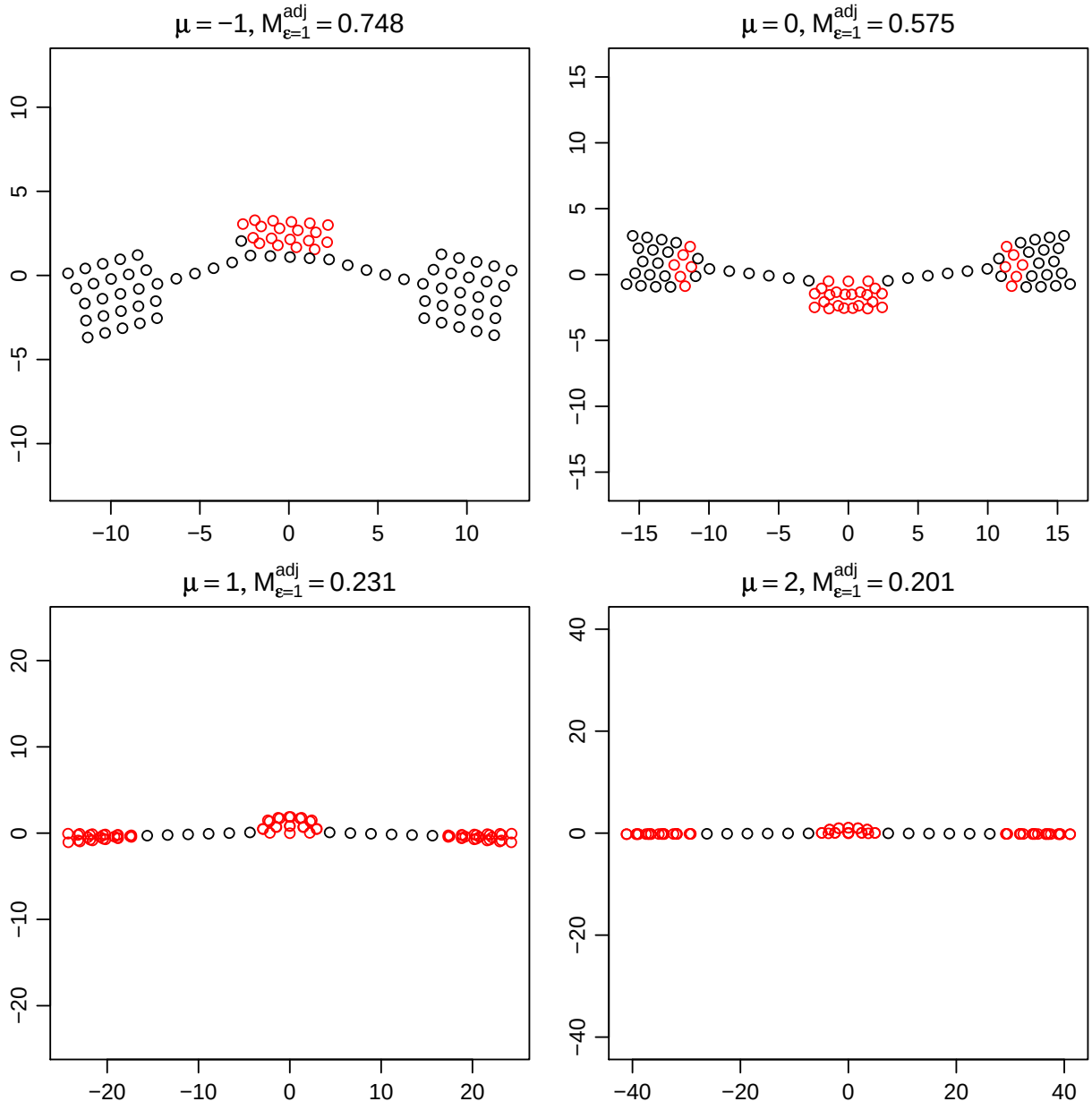

Figure 6: *The configurations with varying μ with other parameters fixed at $\lambda = 5,\ \tau = 1$, starting from a random configuration.*

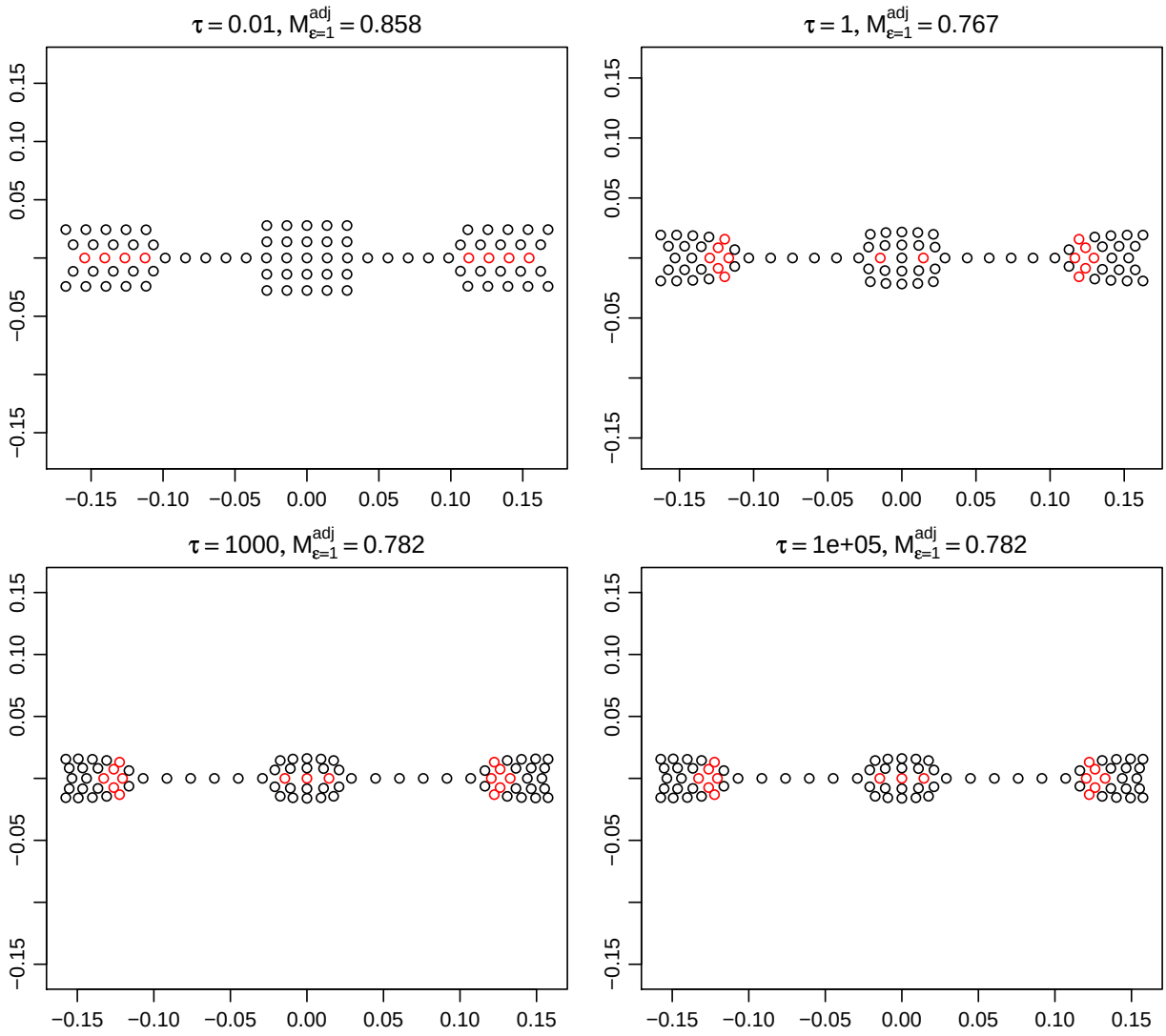

Figure 7: *The configurations with varying τ with other parameters fixed at $\lambda = 5,\ \mu = 0$, starting from the input configuration (Figure 2).*

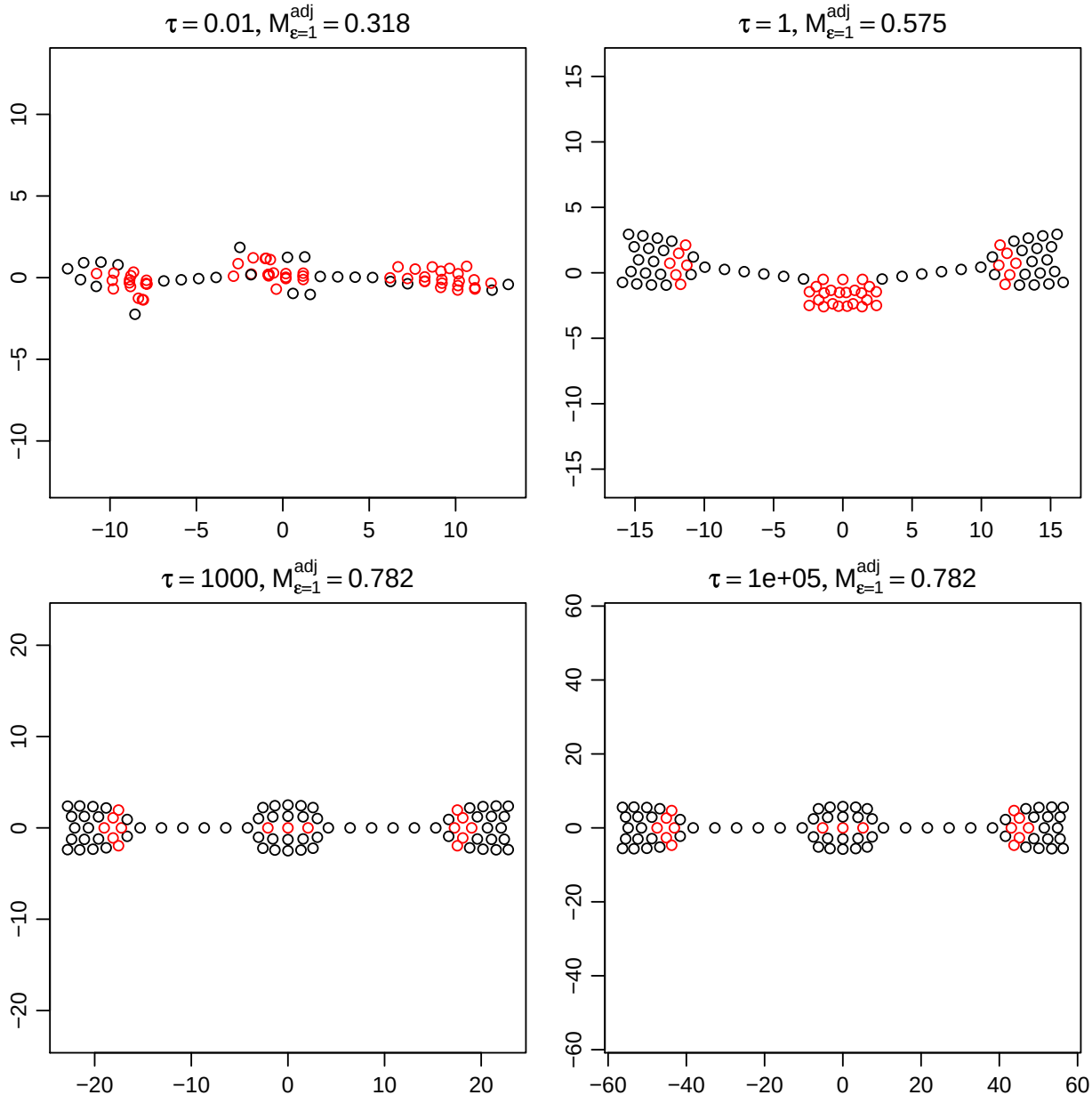

Figure 8: *The configurations with varying τ with other parameters fixed at $\lambda = 5,\ \mu = 0$, starting from a random configuration.*

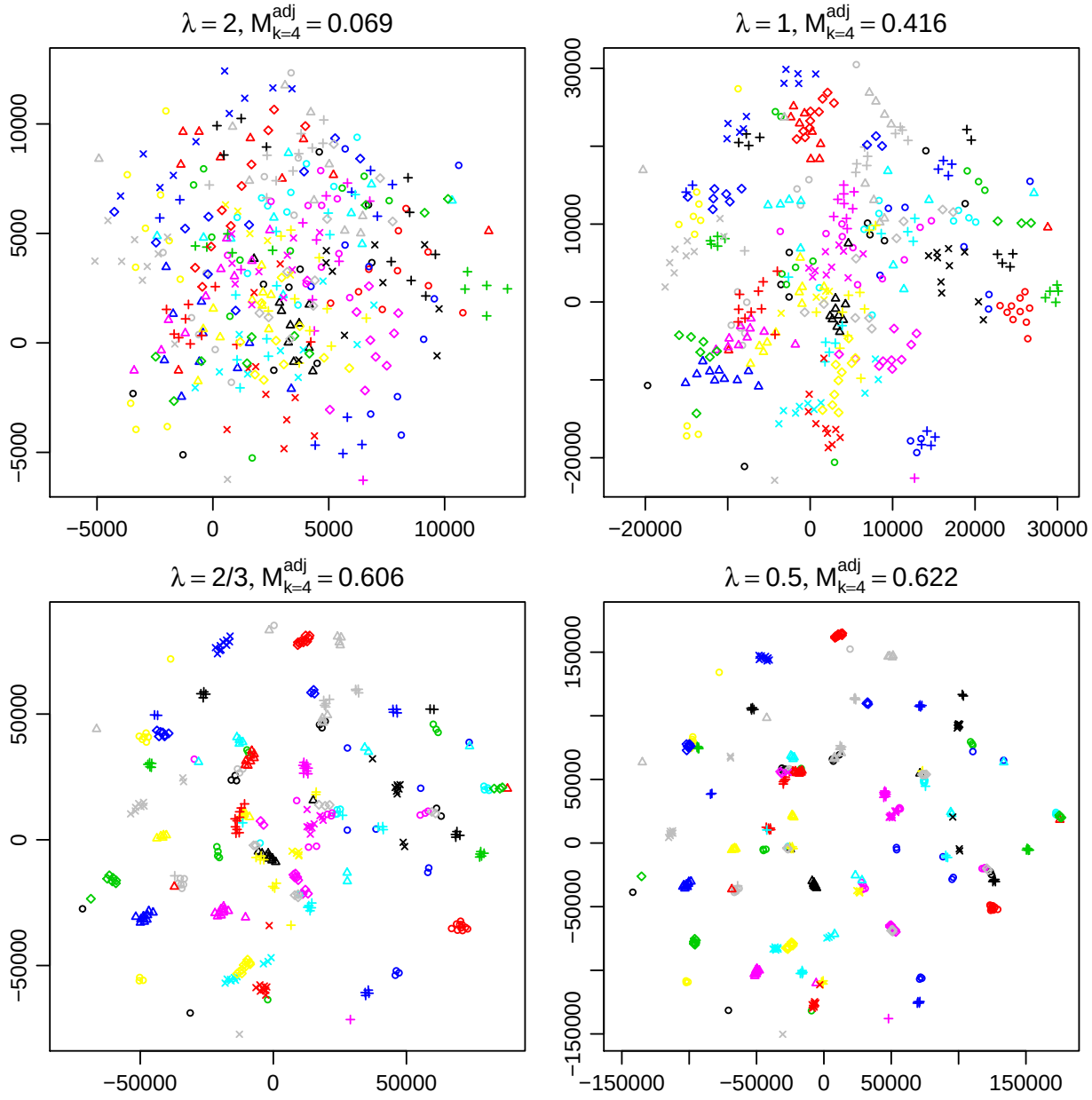

Figure 9: *Olivetti Faces: Configurations with varying λ with other parameters fixed at $\mu = 0,\ \tau = 1$.*

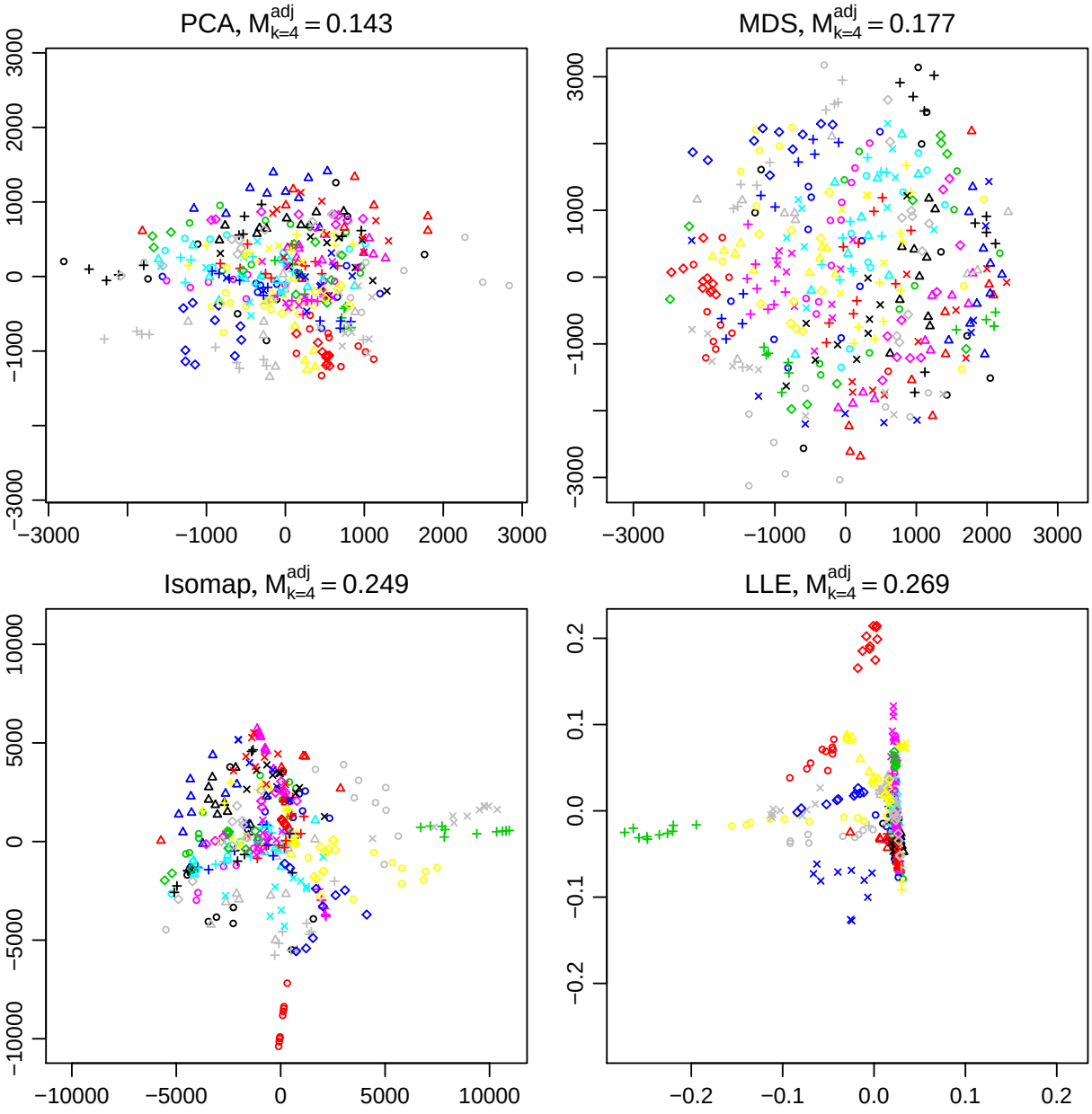

Figure 10: *Olivetti Faces: Configurations from PCA, MDS, Isomap and LLE.*

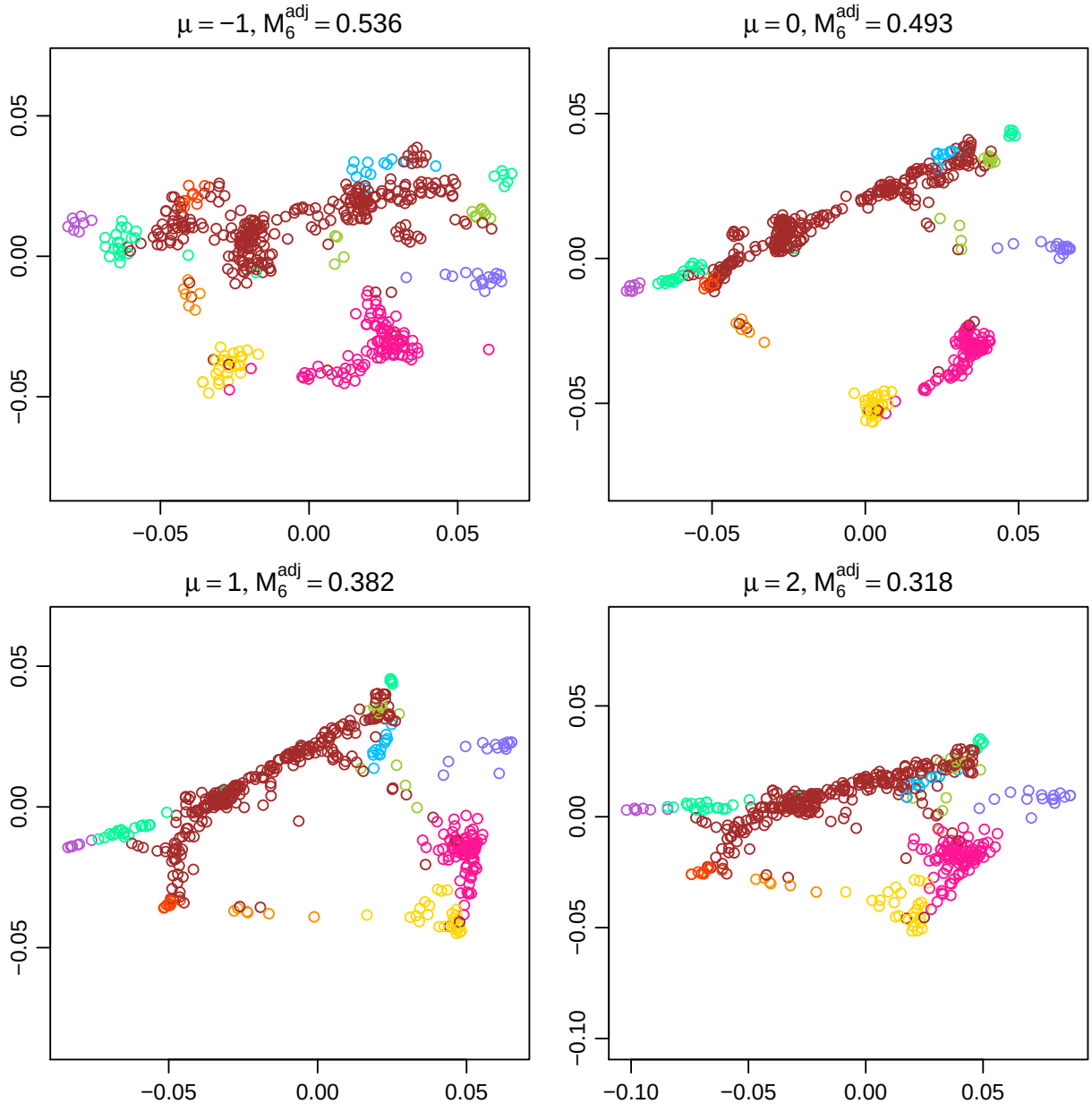

Figure 11: *Frey Face Data: Configurations with varying μ when $\lambda = 1$.*