Diffusion Bases Dimensionality Reduction

Alon Schclar[1] and Amir Averbuch[2]
[1]*School of Computer Science, The Academic College of Tel-Aviv Yaffo, POB 8401, Tel Aviv, 61083, Israel*
[2]*School of Computer Science, Tel Aviv University, POB 39040, Tel Aviv, 69978, Israel*

Keywords: Dimensionality Reduction, Unsupervised Learning.

Abstract: The overflow of data is a critical contemporary challenge in many areas such as hyper-spectral sensing, information retrieval, biotechnology, social media mining, classification etc. It is usually manifested by a high dimensional representation of data observations. In most cases, the information that is inherent in high-dimensional datasets is conveyed by a small number of parameters that correspond to the actual degrees of freedom of the dataset. In order to efficiently process the dataset, one needs to derive these parameters by embedding the dataset into a low-dimensional space. This process is commonly referred to as *dimensionality reduction* or *feature extraction*. We present a novel algorithm for dimensionality reduction – *diffusion bases* – which explores the connectivity among the coordinates of the data and is dual to the diffusion maps algorithm. The algorithm reduces the dimensionality of the data while maintaining the coherency of the information that is conveyed by the data.

1 INTRODUCTION

High dimensional datasets can be found in many areas such as information retrieval, biotechnology, social media, hyper-spectral sensing, classification etc. These datasets are composed of observations that were acquired by a set of sensors. The dimension of a data observation is the number of values that describe it. A simple example is an ordinary color image where each pixel has 3 values that represent the red, green and blue intensities. In this example, the dimensionality is low (equals to 3). In contrast, the dimensionality of hyper-spectral images may reach a few hundreds or even thousands - according to the number of wavelengths that describe the image.

The main problem of high dimensional data is the so called *curse of dimensionality*, which means that the complexity of many algorithms grows exponentially with the increase of the dimensionality of the input data. Commonly, the acquiring sensors produce data whose dimensionality is much higher than the actual degrees of freedom of the data. Unfortunately, this phenomenon is usually unavoidable due to the inability to produce a special sensor for each application. This can be attributed to the lack of knowledge which sensors are more important for the task at hand. Consider, for example, a task that separates red objects from green objects using an off-the-shelf digital camera. In this case, the camera will produce, in addition to the red and green channels, a blue channel, which is unnecessary for this task.

In order to efficiently process high-dimensional datasets, one must first analyze their geometrical structure and detect the parameters that govern the structure of the dataset. This number of parameters is referred to as the *intrinsic dimension* (ID) of the dataset and is equal to the degrees of freedom that are inherent in the data. Thus, the information that is conveyed by the dataset can be described by a set of vectors whose dimension is equal to the ID of the original dataset. Dimensionality reduction algorithms construct a mapping between high-dimensional datasets and low-dimensional datasets whose dimension is close, or ideally equal, to the ID of the original datasets. The mapping should preserve the geometrical structure of the high-dimensional dataset as much as possible.

We propose a novel algorithm for the reduction of dimensionality which we call *diffusion bases* (DB). The algorithm explored the non-linear variability among the coordinates of the data and is dual to the *diffusion maps* (DM) (Coifman and Lafon, 2006) scheme. Both algorithms employ a manifold learning approach. However, depending on the size and dimensionality of the dataset - the DB algorithm may reduce the dimensionality at a computational cost that

Schclar, A. and Averbuch, A..
Diffusion Bases Dimensionality Reduction.
In *Proceedings of the 7th International Joint Conference on Computational Intelligence (IJCCI 2015) - Volume 3: NCTA*, pages 151-156
ISBN: 978-989-758-157-1

is lower than that of the DM algorithm. The DM algorithms has been successfully applied to the detection of moving vehicles (Schclar et al., 2010) and to the construction of ensembles of classifiers (Schclar et al., 2012).

This paper is organized as follows: in section 2 we present a short survey of related work on dimensionality reduction. The diffusion maps scheme (Coifman and Lafon, 2006) is briefly described in section 3. In section 4 we introduce the Diffusion bases (DB) algorithm. Concluding remarks are given in section 5.

2 RELATED WORKS

The theoretical foundations of dimensionality reduction were laid in the pioneering work by Johnson and Lindenstrauss (Johnson and Lindenstrauss, 1984) who showed that N points in N dimensional space can almost always be projected to a space of dimension $C\log N$ with control on the ratio of distances and the error (distortion). Bourgain (Bourgain, 1985) showed that any metric space with N points can be embedded by a bi-Lipschitz map into an Euclidean space of $\log N$ dimension with a bi-Lipschitz constant of $\log N$. Randomized versions of this theorem were used for various applications such as protein mapping (Linial et al., 1997), reconstruction of frequency sparse signals (Candes et al., 2006; Donoho, 2006) and construction of ensembles of classifiers (Schclar and Rokach, 2009)

The general problem of dimensionality reduction has been extensively investigated. Classical techniques for dimensionality reduction such as Principal Component Analysis (PCA) and Multidimensional Scaling (MDS), are simple to implement and can be efficiently computed. However, PCA and *classical* MDS can discover the true structure of data only if it lies on or near a linear subspace of the high-dimensional input space (Mardia et al., 1979). PCA finds a low-dimensional embedding of the data points that best preserves their variance as measured in the high-dimensional input space. Classical MDS finds an embedding that preserves the inter-point distances, and is equivalent to PCA when these distances are the Euclidean distances. However, the pitfall of these methods is that they are *global* i.e. they take into account the distances between *every* pair of points. This makes them susceptible to noise and outliers. Furthermore, many datasets contain nonlinear structures that can not be detected by PCA and MDS.

Some dimensionality reduction methods amend this pitfall by considering only the distances to the closest neighboring points of each point. Two algorithms in this category are Local Linear Embedding (LLE) (Roweis and Saul, 2000) and ISOMAP (Tenenbaum et al., 2000). The LLE algorithm attempts to discover nonlinear structure in high dimensional data by exploiting the local symmetries of linear reconstructions. The ISOMAP (Tenenbaum et al., 2000) approach uses classical MDS but seeks to preserve the intrinsic geometry of the data as captured by the geodesic manifold distances between all pairs of data points. Another algorithm that falls into this category is the *Diffusion Maps (DM)* (Coifman and Lafon, 2006) manifold learning scheme. This algorithm uses the random walk distance metric which takes into account all the paths between every pair of points. This distance reflects the connectivity among the points and is more robust to noise. Furthermore, *DM* can provide parametrization of the data when only the point-wise similarity matrix is available. This may occur either when there is no access to the original data or when the original data consists of abstract objects.

3 DIFFUSION MAPS (DM)

This section briefly describes the *DM* (Coifman and Lafon, 2006) algorithm. Given a set of data points

$$\Gamma = \{x_i\}_{i=1}^{m}, x_i \in \mathbb{R}^n \quad (1)$$

the *DM* algorithm includes the following steps:

1. Construction of an undirected graph G on Γ (the vertices correspond to the data points) with a weight function w_ε that corresponds to the *local* point-wise similarity between the points in Γ[1].
2. Derivation of a random walk on G via a Markov transition matrix P that is obtained from w_ε.
3. Eigen-decomposition of P.

By designing a local geometry that reflects quantities of interest, it is possible to construct a diffusion operator whose eigen-decomposition enables the embedding of Γ into a space Y of substantially lower dimension. The Euclidean distance between a pair of points in the reduced space corrsponds to a diffusion metric that measures the proximity of points in terms of their connectivity in the original space. Specifically, the Euclidean distance between a pair of points, in Y, is equal to the random walk distance between the corresponding pair of points in the original space.

The eigenvalues and eigenfunctions of P define an embedding of the data through the diffusion map.

[1] G is sparse since only the points in the local neighborhood of each point are considered. Wider neighborhood are explored via a diffusion process. In case we are only given w_ε, this step is skipped.

3.1 Building the Graph G and the Weight Function w_ε

Let Γ be a set of points in $\mathbb{R}^n$ as defined in Eq. (1). We construct the graph $G(V,E), |V| = m$, on Γ in order to study the intrinsic geometry of this set. A weight function $w_\varepsilon(x_i,x_j)$, which measures the pairwise similarity between the points, is introduced. For all $x_i,x_j \in \Gamma$, the weight function has the following properties:

- symmetry: $w_\varepsilon(x_i,x_j) = w_\varepsilon(x_j,x_i)$
- non-negativity: $w_\varepsilon(x_i,x_j) \geq 0$
- fast decay: given a scale parameter $\varepsilon > 0$, $w_\varepsilon(x_i,x_j) \to 0$ when $\|x_i - x_j\| \gg \varepsilon$ and $w_\varepsilon(x_i,x_j) \to 1$ when $\|x_i - x_j\| \ll \varepsilon$. The sparsity of G is a result of this property.

Note that the parameter ε defines a notion of neighborhood. In this sense, w_ε defines the local geometry of Γ by providing a first-order pairwise similarity measure for ε-neighborhoods of every point x_i. Higher order similarities are derived through a diffusion process. A common choice for w_ε is the Gaussian kernel $w_\varepsilon(x_i,x_j) = \exp\left(-\frac{\|x_i-x_j\|^2}{2\varepsilon}\right)$. However, other weight functions can be used and the choice of the weight function essentially depends on the application at hand.

Successful dimensionality reduction which preserves the geometry of the original dataset strongly depends on the choice of ε. In the Appendix we discuss the choice of ε and rigorously define the range from which ε should to be selected.

3.2 Construction of the Normalized Graph Laplacian

The non-negativity property of w_ε allows to normalize it into a Markov transition matrix P where the states of the corresponding Markov process are the data points. This enables to analyze Γ via a random walk.

Formally, $P = (p(x_i,x_j))_{i,j=1,\ldots,m}$ is constructed as follows:

$$p(x_i,x_j) = \frac{w_\varepsilon(x_i,x_j)}{d(x_i)} \tag{2}$$

where

$$d(x_i) = \sum_{j=1}^{m} w_\varepsilon(x_i,x_j) \tag{3}$$

is the degree of x_i. If we let $D = (d_{ij})$ be a $m \times m$ diagonal matrix where $d_{ii} = d(x_i)$, and we let W_ε be the weight matrix that corresponds to the weight function w_ε, P can be derived by

$$P = D^{-1} W_\varepsilon. \tag{4}$$

P is a Markov matrix since the sum of each row in P is 1 and $p(x_i,x_j) \geq 0$. Thus, $p(x_i,x_j)$ can be viewed as the probability to move from x_i to x_j in *a single* time step. By raising P to the power t, this probability is propagated to nodes in the neighborhood of x_i and x_j and the result is the probability for this move in t time steps. We denote this probability by $p_t(x_i,x_j)$. These probabilities measure the connectivity of the points within the graph. The parameter t controls the scale of the neighborhood in addition to the scale which is provided by ε.

3.3 Eigen-decomposition

The close relation between the asymptotic behavior of P, i.e. the properties of its eigen-decomposition and the clusters that are inherent in the data, was explored in (Chung, 1997; Fowlkes et al., 2004). We denote the left and the right bi-orthogonal eigenvectors of P by $\{\mu_k\}_{k=1,\ldots,m}$ and $\{\nu_k\}_{k=1,\ldots,m}$, respectively. Let $\{\lambda_k\}_{k=1,\ldots,m}$ be the eigenvalues of P where $|\lambda_1| \geq |\lambda_2| \geq \ldots \geq |\lambda_m|$.

It is well known that $\lim_{t\to\infty} p_t(x_i,x_j) = \mu_1(x_j)$. Coifman et al. (Coifman et al., 2005) show that for finite time t we have

$$p_t(x_i,x_j) = \sum_{k=1}^{m} \lambda_k^t \nu_k(x_i) \mu_k(x_j). \tag{5}$$

A fast decay of $\{\lambda_k\}$ is achieved by an appropriate choice of ε. Thus, to achieve a relative accuracy $\delta > 0$, only a few terms $\eta(\delta)$ are required in the sum in Eq. (5).

Coifman and Lafon (Coifman and Lafon, 2006) introduced the *diffusion distance*

$$D_t^2(x_i,x_j) = \sum_{k=1}^{m} \frac{(p_t(x_i,x_k) - p_t(x_k,x_j))^2}{\mu_1(x_k)}.$$

This formulation is derived from the known random walk distance in Potential Theory: $D_t^2(x_i,x_j) = p_t(x_i,x_i) + p_t(x_j,x_j) - 2p_t(x_i,x_j)$ where the factor 2 is due to the fact that G is undirected.

Averaging along all the paths from x_i to x_j results in a distance measure that is more robust to noise and topological short-circuits than the geodesic distance (Tenenbaum et al., 2000). Finally, the diffusion distance can be expressed in terms of the right eigenvectors of P (see (Keller and Coifman, 2006) for a proof):

$$D_t^2(x_i,x_j) = \sum_{k=1}^{m} \lambda_k^{2t} (\nu_k(x_i) - \nu_k(x_j))^2.$$

It follows that in order to compute the diffusion distance, one can simply use the right eigenvectors of P. Moreover, this facilitates the embedding of the original points in a Euclidean space $\mathbb{R}^{\eta(\delta)-1}$ by:

$$\Xi_t : x_i \to \left(\lambda_2^t \nu_2(x_i), \lambda_3^t \nu_3(x_i), \ldots, \lambda_{\eta(\delta)}^t \nu_{\eta(\delta)}(x_i)\right).$$

The first eigenvector ν_1 is not used since it is constant. This also endows coordinates on the set Γ. Essentially, $\eta(\delta) \ll n$ due to the fast decay of the eigenvalues of P. Furthermore, $\eta(\delta)$ depends only on the primary intrinsic variability of the data as captured by the random walk and not on the original dimensionality of the data. This data-driven method enables the parametrization of any set of points - abstract or not - provided the similarity matrix w_ε of the points is available.

4 DIFFUSION BASES (DB)

Diffusion bases (*DB*) is a dual algorithm to the *DM* algorithm in the sense that it explores the connectivity among the *coordinates* of the original data instead of the connectivity among the data points. Both algorithms share a graph Laplacian construction, however, the *DB* algorithm uses the Laplacian eigenvectors as an orthonormal system and projects the original data on it.

Let $\Gamma = \{x_i\}_{i=1}^m, x_i \in \mathbb{R}^n$, be the original dataset as defined in Eq. (1) and let $x_i(j)$ denote the j-th coordinate of x_i, $1 \le j \le n$. We define the vector $x'_j \triangleq (x_1(j), \ldots, x_m(j))$ to be the j-th coordinate of all the points in Γ. We construct the set

$$\Gamma' = \{x'_j\}_{j=1}^n. \tag{6}$$

The *DM* algorithm is applied to the set Γ'. The right eigenvectors of P constitute an orthonormal basis $\{\nu_k\}_{k=1,\ldots,n}, \nu_k \in \mathbb{R}^n$. This bares some similarity to PCA, however, the diffusion process has the potential to achieve better dimensionality reduction due to: (a) its ability to capture non-linear manifolds within the data by local exploration of each coordinate; (b) its robustness to noise. Furthermore, this process is more general than PCA and it produces similar results to PCA when the weight function w_ε is *linear* e.g. the inner product, Euclidean distance.

Next, we use the eigenvalue decay property of the eigen-decomposition to extract only the first $\eta(\delta)$ eigenvectors $B \triangleq \{\nu_k\}_{k=1,\ldots,\eta(\delta)}$ (we do not exclude the first eigenvector as mentioned in section 3.3).

We project the original data Γ onto the basis B. Let Γ_B be the set of these projections which is defined as follows: $\Gamma_B = \{g_i\}_{i=1}^m, g_i \in \mathbb{R}^{\eta(\delta)}$, where $g_i = \left(x_i \cdot \nu_1, \ldots, x_i \cdot \nu_{\eta(\delta)}\right)$, $i = 1, \ldots, m$ and $\cdot$ denotes the inner product operator. Γ_B contains the coordinates of the original points in the orthonormal system whose axes are given by B. Alternatively, Γ_B can be interpreted in the following way: the coordinates of g_i contain the correlation between x_i and the directions given by the vectors in B. A summary of the *DiffusionBases* procedure is given in Algorithm 1.

The duality connection between the DB and DM algorithms can be demonstrated, for example, when the weight function is defined by the dot product, i.e. $w(x_i, x_j) = \langle x_i, x_j \rangle$. In this case DM and DB are connected through the singular value decomposition of the weight matrix $W = BSR^T$. Namely, $WW^T = BSR^T RSB^T = BS^2B^T$ and $W^TW = RSB^T BSR^T = RD^2R^T$ and thus the results of the eigen-decomposition steps in the DM and DB algorithms are given by B and R, respectively.

Algorithm 1: The Diffusion Bases Algorithm.

DiffusionBases(Γ', w_ε, ε, δ)

1. Calculate the weight function $w_\varepsilon\left(x'_i, x'_j\right)$, $i, j = 1, \ldots n$.
2. Construct a Markov transition matrix P by normalizing the sum of each row in w_ε to be 1:

$$p\left(x'_i, x'_j\right) = \frac{w_\varepsilon\left(x'_i, x'_j\right)}{d\left(x'_i\right)}$$

   where $d\left(x'_i\right) = \sum_{j=1}^n w_\varepsilon\left(x'_i, x'_j\right)$.
3. Perform eigen-decomposition of $p\left(x'_i, x'_j\right)$

$$p\left(x'_i, x'_j\right) \equiv \sum_{k=1}^n \lambda_k \nu_k\left(x'_i\right) \mu_k\left(x'_j\right)$$

   where the left and the right eigenvectors of P are given by $\{\mu_k\}$ and $\{\nu_k\}$, respectively, and $\{\lambda_k\}$ are the eigenvalues of P in descending order of magnitude.
4. Project the original data Γ onto the orthonormal system $B \triangleq \{\nu_k\}_{k=1,\ldots,\eta(\delta)}$:

$$\Gamma_B = \{g_i\}_{i=1}^m, g_i \in \mathbb{R}^{\eta(\delta)}$$

   where

$$g_i = \left(x_i \cdot \nu_1, \ldots, x_i \cdot \nu_{\eta(\delta)}\right),$$
$$i = 1, \ldots, m, \nu_k \in B, 1 \le k \le \eta(\delta)$$

   and $\cdot$ is the inner product.
5. **return** Γ_B.

5 FUTURE RESEARCH

It was shown in (Coifman and Lafon, 2006) that any positive semi-definite kernel may be used for the dimensionality reduction. Rigorous analysis of families of kernels to facilitate the derivation of an optimal kernel for a given set Γ is an open problem.

The parameter $\eta(\delta)$ determines the dimensionality of the diffusion space. A rigorous method for choosing $\eta(\delta)$ will facilitate an automatic embedding of the data. Naturally, $\eta(\delta)$ is *data driven* (similarly to ε) i.e. it depends on the set Γ at hand.

Finally, various applications of the diffusion bases scheme are currently being investigated by the authors - namely, video segmentation and construction of ensembles of classifiers.

REFERENCES

Bourgain, J. (1985). On lipschitz embedding of finite metric spaces in hilbert space. *Israel Journal of Mathematics*, 52:46–52.

Candes, E., Romberg, J., and Tao, T. (2006). Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509.

Chung, F. R. K. (1997). *Spectral Graph Theory*. AMS Regional Conference Series in Mathematics, 92.

Coifman, R. R. and Lafon, S. (2006). Diffusion maps. *Applied and Computational Harmonic Analysis: special issue on Diffusion Maps and Wavelets*, 21:5–30.

Coifman, R. R., Lafon, S., Lee, A., Maggioni, M., Nadler, B., Warner, F., and Zucker, S. (2005). Geometric diffusions as a tool for harmonics analysis and structure definition of data: Diffusion maps. In *Proceedings of the National Academy of Sciences*, volume 102, pages 7432–7437.

Donoho, D. (2006). Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306.

Fowlkes, C., Belongie, S., Chung, F., and Malik, J. (2004). Spectral grouping using the nyström method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):214–225.

Hein, M. and Audibert, Y. (2005). Intrinsic dimensionality estimation of submanifolds in euclidean space. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 289–296.

Johnson, W. B. and Lindenstrauss, J. (1984). Extensions of lipshitz mapping into hilbert space. *Contemporary Mathematics*, 26:189–206.

Keller, S. L. Y. and Coifman, R. R. (2006). Data fusion and multi-cue data matching by diffusion maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1784–1797.

Linial, M., Linial, N., Tishby, N., and Yona, G. (1997). Global self-organization of all known protein sequences reveals inherent biological signatures. *Journal of Molecular Biology*, 268(2):539–556.

Mardia, K. V., Kent, J. T., and Bibby, J. M. (1979). *Multivariate Analysis*. Academic Press, London.

Roweis, S. T. and Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326.

Schclar, A., Averbuch, A., Hochman, K., Rabin, N., and Zheludev, V. (2010). A diffusion framework for detection of moving vehicles. *Digital Signal Processing,*, 20(1):111–122.

Schclar, A. and Rokach, L. (ICEIS 2009). Random projection ensemble classifiers. *Lecture Notes in Business Information Processing, Proceedings of the 11th Conference on Enterprise Information System*.

Schclar, A., Rokach, L., and Amit, A. (2012). Diffusion ensemble classifiers. In *Proceedings of the 4th International Conference on Neural Computation Theory and Applications (NCTA 2012)*, Barcelona, Spain.

Tenenbaum, J. B., de Silva, V., and Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2323.

APPENDIX: CHOOSING ε

The choice of ε is critical to achieve the optimal performance of the DM and DB algorithms since it defines the size of the local neighborhood of each point. On one hand, a large ε produces a coarse analysis of the data as the neighborhood of each point will contain a large number of points. In this case, the similarity weight will be close to one for most pairs of points. On the other hand, a small ε might produce neighborhoods that contain only one point. In this case, the similarity will be zero for most pairs of points. Clearly, an adequate choice of ε lies between these two extreme cases and should be derived from the data.

In the following, we derive the range from which ε should be chosen when a Gaussian weight function is used and when the dataset Γ approximately lies near a low dimensional manifold. We denote by d the intrinsic dimension of M. Let $L = I - P = I - D^{-1}W$ be the *normalized graph Laplacian* (Chung, 1997) where P was defined in Eq. (4) and I is the identity matrix. The matrices L and P share the same eigenvectors. Furthermore, Singer (2006) proved that if the points in Γ are independently uniformly distributed over M then with high probability

$$\frac{1}{\varepsilon}\sum_{j=1}^{m} L_{ij} f(x_j) = \frac{1}{2}\triangle_M f(x_i) + O\left(\frac{1}{m^{1/2}\varepsilon^{1/2+d/4}}, \varepsilon\right) \tag{7}$$

where $f : M \to \mathbb{R}$ is a smooth function and $\triangle_M$ is the continuous Laplace-Beltrami operator of the manifold

M. The error term is composed of a variance term $O\left(\frac{1}{m^{1/2}\varepsilon^{1/2+d/4}}\right)$, which is minimized by a large value of ε, and a bias term $O(\varepsilon)$, which is minimized by a small value of ε.

We utilize the scheme that was proposed in (Hein and Audibert, 2005) and examine the sum of the weight matrix elements

$$S_\varepsilon = \sum_{i=1}^{m}\sum_{j=1}^{m} w_\varepsilon(x_i,x_j) = \sum_{i=1}^{m}\sum_{j=1}^{m} \exp\left(-\frac{\left\|x_i-x_j\right\|^2}{2\varepsilon}\right) \tag{8}$$

as a function of ε. Let $Vol(M)$ be the volume of the manifold M. The sum in Eq. (8) can be approximated by its mean value integral

$$S_\varepsilon \approx \frac{m^2}{Vol^2(M)} \int_M \int_M \exp\left(-\frac{\|x-x'\|^2}{2\varepsilon}\right) dx dx' \tag{9}$$

provided the variance term in Eq. (7) is sufficiently small.

Moreover, we use the fact that for small values of ε the manifold locally looks like its tangent space $\mathbb{R}^d$ and thus

$$\int_M \exp\left(-\frac{\|x-x'\|^2}{2\varepsilon}\right) dx \approx$$

$$\int_{\mathbb{R}^d} \exp\left(-\frac{\|x-x'\|^2}{2\varepsilon}\right) dx = (2\pi\varepsilon)^{d/2}. \tag{10}$$

Combining Eqs. (8)-(10), we get

$$S_\varepsilon \approx \frac{m^2}{Vol(M)} (2\pi\varepsilon)^{d/2}.$$

Applying logarithm on both sides yields

$$\log(S_\varepsilon) \approx \frac{d}{2}\log(\varepsilon) + \log\left(\frac{m^2(2\pi)^{d/2}}{Vol(M)}\right).$$

Consequently, the slope of S_ε as a function of ε on a log-log scale is $\frac{d}{2}$. However, this slope is only linear in a limited subrange of ε since $\lim_{\varepsilon\to\infty} S_\varepsilon = m^2$ and $\lim_{\varepsilon\to 0} S_\varepsilon = m$ as illustrated in Fig. 1. In this subrange, the error terms in Eq. (7) are smaller than they are in the rest of the ε range. Thus, an adequate ε should be chosen from this linear subrange.

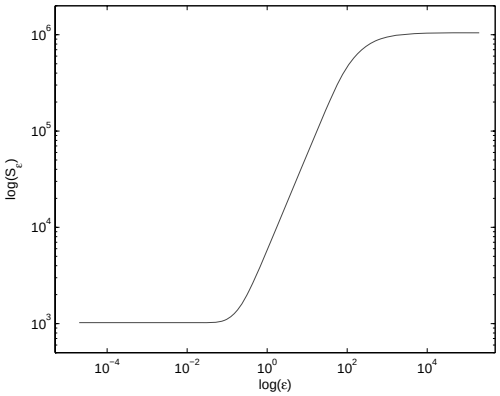

Figure 1: A plot of S_ε as a function of ε on a log-log scale.