# VARIANCE REDUCTION WITH NEIGHBORHOOD SMOOTHING FOR LOCAL INTRINSIC DIMENSION ESTIMATION

*Kevin M. Carter*[*], *Alfred O. Hero III*

Department of EECS
University of Michigan
Ann Arbor, MI 48109
{kmcarter,hero}@umich.edu

## ABSTRACT

Local intrinsic dimension estimation has been shown to be useful for many tasks such as image segmentation, anomaly detection, and de-biasing global dimension estimates. Of particular concern with local dimension estimation algorithms is the high variance for high dimensions, leading to points which lie on the same manifold estimating at different dimensions. We propose adding adaptive 'neighborhood smoothing' – filtering over the generated dimension estimates to obtain the most probable estimate for each sample – as a method to reduce variance and increase algorithm accuracy. We present a method for defining neighborhoods using a geodesic distance, which constricts each neighborhood to the manifold of concern, and prevents smoothing over intersecting manifolds of differing dimension. Finally, we illustrate the benefits of neighborhood smoothing on synthetic data sets as well as towards diagnosing anomalies in router networks.

***Index Terms***— Intrinsic dimension, manifold learning, Riemannian manifold, nearest neighbor graph, geodesics

## 1. INTRODUCTION

The field of manifold learning has led to many methods which allow for significant reduction of high dimensional data sets with minor or no loss of information. To perform this dimension reduction, one first needs to know the *intrinsic dimensionality* of the manifold supporting the data. In many problems of practical interest data will exhibit varying dimensionality, as multiple distinct and possibly intersecting manifolds may be represented in a single data set. In [1] we presented a method for adapting global dimension estimation algorithms [2–5] to work in the local sense, obtaining a dimension estimate in the neighborhood of each point within a data set rather than a single estimate for the entire set. We showed that local dimension estimation can be used in problems of practical interest, such as anomaly detection, image segmentation, and for de-biasing global dimension estimation.

In this paper we present a variance reduction method for local dimension estimation. Under the assumption that points which are close in Euclidean distance tend to lie on the same manifold, we are able to filter algorithm results to obtain a new dimension estimate for each sample which is equal to the most represented dimension estimate in a local region about that point. We refer to this process as 'neighborhood smoothing,' as the filter tends to smooth out the highly variable estimates between close samples. We define these local neighborhoods by adapting each neighborhood to the shape of the manifold near the concerned sample point. This constricts each region to consider only points on the same manifold as the sample of interest, and ignore points from disjoint manifolds which may lie close in Euclidean space. We illustrate neighborhood smoothing with local dimension estimation on both synthetic data sets and real data concerning network anomaly detection.

This paper proceeds as follows: In Section 2 we give a review of the $k$-NN dimension estimation algorithm, which is the algorithm we utilize for illustration in this study. Section 3 presents the main contribution of this paper, adaptive neighborhood smoothing as post-processing to local dimension estimation. Experimental results and comparisons are presented in Section 4. Finally, Section 5 presents the conclusions and some possible directions for future improvements.

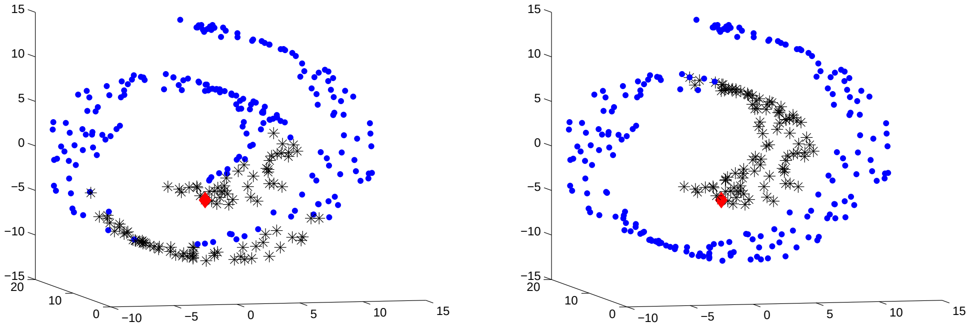
## 2. THE K-NEAREST NEIGHBOR ALGORITHM FOR DIMENSION ESTIMATION

Let $\mathcal{Y}_n = \{\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n\}$ be $n$ independent and identically distributed (i.i.d.) random vectors with values in a compact subset of $\mathbb{R}^d$. The (1-)nearest neighbor of $\boldsymbol{Y}_i$ in $\mathcal{Y}_n$ is given by

$$\arg \min_{\boldsymbol{Y} \in \mathcal{Y}_n \backslash \{\boldsymbol{Y}_i\}} |\boldsymbol{Y} - \boldsymbol{Y}_i|,$$

where $|\boldsymbol{Y} - \boldsymbol{Y}_i|$ is the usual Euclidean ($L_2$) distance in $\mathbb{R}^d$ between vector $\boldsymbol{Y}$ and $\boldsymbol{Y}_i$. For a general integer $k \geq 1$, the $k$-nearest neighbor of a point is defined in a similar way. The $k$-NN graph assigns an edge between each point in $\mathcal{Y}_n$ and

[*]Sponsor: National Science Foundation, grant No. CCR-0325571.



(a) Spherical Neighborhood

(b) Adaptive Neighborhood

**Fig. 1**. Neighborhoods ($\star$) of the sample in question ($\diamond$) defined by a) Euclidean distance and b) geodesic distance.

its $k$-nearest neighbors. Let $\mathcal{N}_{k,i} = \mathcal{N}_{k,i}(\mathcal{Y}_n)$ be the set of $k$-nearest neighbors of $\boldsymbol{Y}_i$ in $\mathcal{Y}_n$. The total edge length of the $k$-NN graph is defined as:

$$L_{\gamma,k}(\mathcal{Y}_n) = \sum_{i=1}^{n} \sum_{\boldsymbol{Y}\in\mathcal{N}_{k,i}} |\boldsymbol{Y} - \boldsymbol{Y}_i|^{\gamma} \,, \qquad (1)$$

where $\gamma > 0$ is a power weighting constant.

For many data sets of interest, the random vectors $\mathcal{Y}_n$ are constrained to lie on an $m$-dimensional Riemannian submanifold $\mathcal{M}$ of $\mathbb{R}^d$ ($m < d$). Under this framework with a large $n$ approximation, the asymptotic behavior of (1) is given by [4] as

$$L_{\gamma,k}(\mathcal{Y}_n) = n^{\alpha}c + \epsilon_n \qquad (2)$$

where $\alpha = (m-\gamma)/m$ and $c$ is a constant with respect to $\alpha$ that depends on the Rènyi entropy of the distribution of the sample on the manifold.

The intrinsic dimension estimate $\hat{m}$ can be found using non-linear least squares (NLS) by calculating graph lengths over varying values of $n$. We solve NLS for $\hat{m}$ by minimizing over both $c$ and integer values of $m \in \mathbb{Z}$. This leads to an estimator

$$\hat{m} = \arg\min_{m\in\mathbb{Z}}\{\min_{c} \sum_{n} (L_n - n^{\alpha(m)}c)^2\}. \qquad (3)$$

Graph lengths $L_n = L_{\gamma,k}(\mathcal{Y}_n)$ for differing sample sizes on the manifold are calculated using a block bootstrapping method, details of which can be found in [1].

### 2.1. Local Dimension Estimation

The $k$-NN algorithm in itself is a global dimension estimator, i.e. it globally fits the $k$-NN graph length functional $L_n$ and solves (3) over the entire sample space. It is transformable as a local dimension estimator by running the algorithm over a smaller neighborhood about each sample point. Intuitively, if an $m$-dimensional manifold $\mathcal{M}$ supports a uniform distribution at the $n$ points, $\mathcal{Y}_n = \{\boldsymbol{Y}_1 \ldots \boldsymbol{Y}_n\}$, then any small sphere or data cluster $\mathcal{C} \subseteq \mathcal{M}$, centered at point $\boldsymbol{Y}_i$ will also support a uniform distribution over $n' \leq n$ data points. As such, the global dimension estimation algorithm can be used on a local subset of the data to estimate the local intrinsic dimension of each sample point.

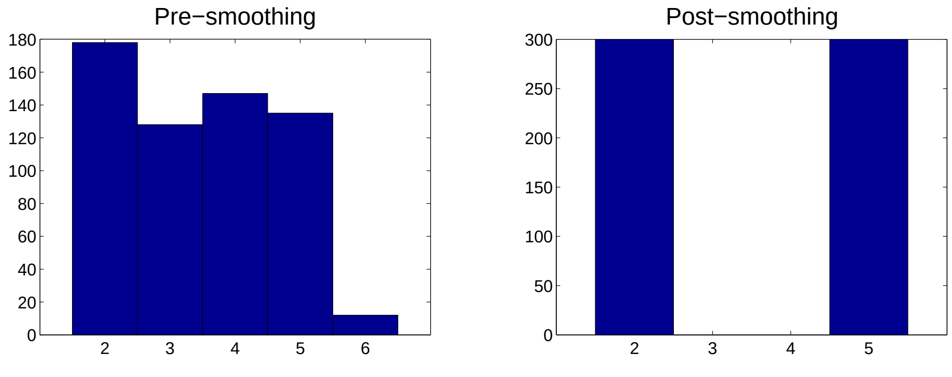
## 3. NEIGHBORHOOD SMOOTHING

In previous work, dimension estimates were solved as described above without any variance reduction, which led to results with a high variability due to the random subsampling. To increase the accuracy of the algorithm, we have added neighborhood smoothing as a post-processing of the results of the $k$-NN dimension estimator. The initial intuition when developing the algorithm was that samples that were "close" tend to lie on the same manifold, and therefore have the same dimension. With that assumption still in place, it follows that filtering by majority vote over the dimension estimates of nearby samples should smooth the estimator and reduce variance. This voting strategy is similar to the methods of bagging [6] and learning by rule ensembles [7]. Smoothing simply looks at the distribution of dimension estimates within each sample point's local neighborhood, and re-assigns each sample a dimension estimate equal to that with the highest probability within its neighborhood. Specifically,

$$\hat{m} = \arg\max_{j} \mathrm{P}_{\mathcal{N}_i}\left[\hat{m} = j\right], \qquad (4)$$

where $\mathrm{P}_{\mathcal{N}_i}$ is the probability over the neighborhood of the current sample $\mathcal{N}_i$.

The key factor to smoothing is defining the neighborhood, $\mathcal{N}_i$. If $\mathcal{N}_i$ is too large, oversmoothing will occur. The variance of the dimension estimates will drastically decrease, but there will be a strong bias which will remove detected anomalies and smaller manifolds. As such, one cannot use a constant,



(a)

(b)

**Fig. 2**. Neighborhood smoothing applied to 7-dimensional data containing two spheres with intrinsic dimensions 2 and 5

spherical region about a point, but must adapt that region to the statistics of the sample.

### 3.1. Non-Spherical Neighborhoods

Rather than defining neighborhoods through Euclidean distance, which will form only spherical regions about each sample point, we will define neighborhoods using a geodesic distance metric. This will adapt the neighborhood to the geometry of the manifold. The geodesic distance between 2 unconnected points in a graph is defined as the shortest path connecting said points. For our purposes, this metric can be determined by taking each point, and creating an edge to the $k$-NN of each point. Then using Dijkstra's shortest path algorithm (or any other algorithm for computing the shortest path), find the geodesic distances to each pair of points in the graph. Any points that remain unconnected are considered to have an infinite geodesic distance.

To define a local neighborhood, we can now simply choose the closest $n_{geo}$ points for which the geodesic distance is not infinite. This forms a non-spherical neighborhood that adapts to the curvature of the manifold, performing much better than spherical neighborhoods. Figure 1 illustrates the difference in the neighborhoods (black stars) that are formed on the 'swiss roll' manifold when using different proximity metrics. The Euclidean distance (Fig. 1(a)) forms a spherical neighborhood, including points that are separated from the sample in question (red diamond). The geodesic distance (Fig. 1(a)), however, forms a neighborhood considering points only in close proximity along the actual manifold. This prevents smoothing across distinct manifolds which may lie closely together in Euclidean space.

## 4. SIMULATION RESULTS

One immediate benefit is that neighborhood smoothing allows us to decrease run time by almost two orders of magnitude. This is a result of the ability to use nominal settings in the estimation algorithm (i.e. less averaging and bootstrapping) which significantly reduces computational complexity.

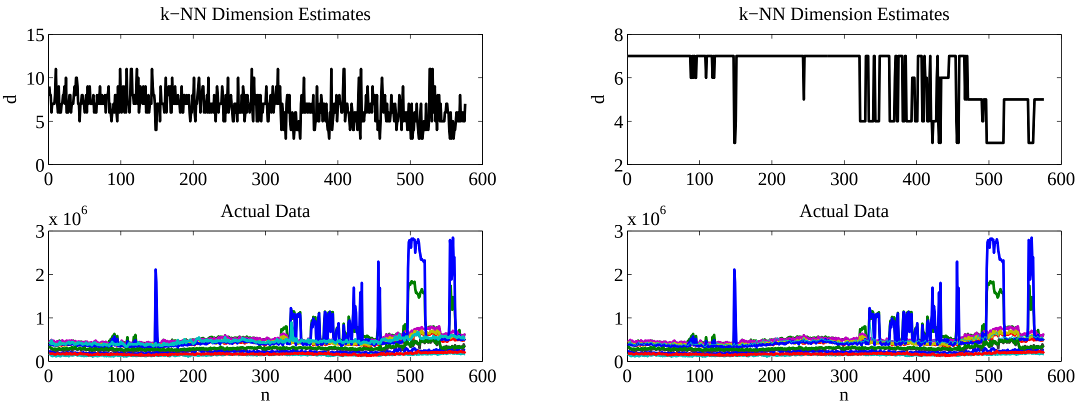
### 4.1. Two Spheres

Illustrating the effects of the adaptive neighborhood smoothing, we create a 7-dimensional data set that includes 2 distinct spheres of intrinsic dimensions 2 and 5, each containing 300 uniformly sampled points. The spheres intersect in three common dimensions. Fig. 2(a) shows the histogram of the local dimension estimates of each sample before any neighborhood smoothing was applied, while Fig. 2(b) shows the results after the smoothing. One can clearly see that the wide histogram was correctly condensed to the proper local dimension estimates, even though the manifolds intersect.

### 4.2. Abilene Network Data

Anomalies can be detected in router networks through the use of local dimension estimation [1]. Specifically, when only a few of the routers contribute disproportionably large amounts of traffic, the intrinsic dimension of the entire network decreases. Using neighborhood smoothing as a form of post-processing, we are better able to locate the traffic anomalies, as the variance of the estimates is reduced. Fig. 3 illustrates the usage of neighborhood smoothing on the results of local dimension estimation for anomaly detection. The data used is the number of packets counted on each of the 11 routers on the Abilene network, on January 1-2, 2005. Each sample is taken every 5 minutes, leading to 576 samples with an extrinsic dimension of $d = 11$.

Figure 3(b) illustrates that neighborhood smoothing is able to preserve both the visually obvious ($n = 148$, $n > 300$) and non-obvious ($n = 87 - 120$) changes in network complexity. A detailed investigation of time $n = 244$, for example, reveals that the Sunnyvale router (SNVA) showed increased contribution from a single IP address, and multiple routers showed increased activity on a single port. This change in di-



(a) Before Smoothing

(b) After Smoothing

**Fig. 3**. Neighborhood smoothing applied to Abilene Network traffic data dimension estimation results.

mensionality indicating anomalous activity would generally go unnoticed with the raw results of local dimension estimation due to the high variance (Fig. 3(a)).

We note the results shown in Fig. 3 are performed using nominal algorithm settings which allows the algorithm to run quickly and accurately with neighborhood smoothing. We are able to generate results with much less variance than Fig. 3(a) by applying more averaging and bootstrapping, but this increases computation time by over an order of magnitude, while still producing results with much more variance than Fig. 3(b).

## 5. CONCLUSIONS

We have presented a form of post-processing for local intrinsic dimension estimation that reduces the variability of algorithm results. By maintaining the original assumption that points that are close in Euclidean space tend to lie on the same manifold, we are able to perform local neighborhood smoothing by assigning each sample point a dimension estimate which is most probable in its local neighborhood. By utilizing the geodesic distance rather than the Euclidean distance, the constructed neighborhoods adaptively mold to the shape of the manifold. This prevents smoothing over disjoint manifolds which may lie close in Euclidean space. The use of neighborhood smoothing as post-processing enables the $k$-NN dimension estimation algorithm to run with over an order of magnitude less complexity, due to the necessity for only nominal averaging and bootstrapping.

We have shown that smoothing can significantly improve the ability to use local intrinsic dimension estimation as a means for anomaly detection in router networks. By reducing the variance of the results, anomalies clearly stand out. We note that while we utilize the $k$-NN dimension estimation algorithm for this study, neighborhood smoothing may be utilized with any method of local dimension estimation. Future work includes using neighborhood smoothing for other learning tasks (e.g. classification) and studying the effects smoothing has on de-biasing for global dimension estimation.

## 6. ACKNOWLEDGEMENTS

We would like to thank Bobby Li from the University of Michigan for isolating the source of the anomalies we discovered in the Abilene data.

## 7. REFERENCES

[1] K. M. Carter, A. O. Hero, and R. Raich, "De-biasing for intrinsic dimension estimation," in *Proc. IEEE Statistical Signal Processing Workshop*, August 2007, pp. 601–605.

[2] F. Camastra and A. Vinciarelli, "Estimating the intrinsic dimension of data with a fractal-based method," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 10, pp. 1404–1407, October 2002.

[3] B. Kégl, "Intrinsic dimension estimation using packing numbers," in *Neural Information Processing Systems: NIPS*, Vancouver, CA, Dec. 2002.

[4] J. Costa and A. O. Hero, *Statistics and analysis of shapes*, chapter Learning Intrinsic Dimension and Entropy of High-Dimensional Shape Spaces, pp. 231–252, Birkhauser, 2006.

[5] E. Levina and P. Bickel, "Maximum likelihood estimation of intrinsic dimension," in *Neural Information Processing Systems: NIPS*, Vancouver, CA, Dec. 2004.

[6] Leo Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.

[7] J. H. Friedman and B. E. Popescu, "Predictive learning via rule ensembles," Tech. Rep., Stanford University, 2005.