DE-BIASING FOR INTRINSIC DIMENSION ESTIMATION

Kevin M. Carter, Alfred O. Hero III, and Raviv Raich*

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

ABSTRACT

Many algorithms have been proposed for estimating the intrinsic dimension of high dimensional data. A phenomenon common to all of them is a negative bias, perceived to be the result of undersampling. We propose improved methods for estimating intrinsic dimension, taking manifold boundaries into consideration. By estimating dimension locally, we are able to analyze and reduce the effect that sample data depth has on the negative bias. Additionally, we offer improvements to an existing algorithm for dimension estimation, based on k-nearest neighbor graphs, and offer an algorithm for adapting any dimension estimation algorithm to operate locally. Finally, we illustrate the uses of local dimension estimation with data sets consisting of multiple manifolds, including applications such as diagnosing anomalies in router networks and image segmentation.

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

1. INTRODUCTION

Technological advances in both sensing and media storage have allowed for the generation of massive amounts of high dimensional data and information. While this data may appear to be very complex, much of the information is often concentrated on lower dimensional subsets – manifolds – which allows for significant dimension reduction with minor or no loss of information. To perform dimension reduction, one first needs to know the *intrinsic dimensionality* of the manifold supporting the data. In contrast to model order selection methods such as MDL, AIC, or BIC (see [1]), we consider non-parametric methods of dimension estimation. When the intrinsic dimension is assumed constant over the data set, several algorithms [2–5] have been proposed to estimate the dimensionality of the manifold. In several problems of practical interest, however, data will exhibit varying dimensionality. For example, the intrinsic dimension of a time series data set can vary with time.

To our knowledge, every method of estimating intrinsic dimension has expressed an issue with a negative bias, due to insufficient sampling of the manifold. We propose that a significant portion of the bias is a result of regions on the manifold which may appear to be low dimensional when sampled. Specifically, samples near the boundaries or edges of a manifold contribute a strong negative bias to the global estimate of dimension. In this paper we will show, by using local dimension estimation and data depth analysis, that we are able to isolate those regions of the manifold that contribute to the bias, improving upon global dimension estimation. Furthermore, we will present additional novel uses for local dimension estimation, including network anomaly detection and image segmentation.

For the purposes of this paper, we will be utilizing an improved version of the k-nearest neighbor algorithm for dimension estimation; which was originally presented in [4] and applied locally in [6]. While performing competitively with other algorithms, the variance of the results across simulations on the same data set was high. A significant source of discrepancy was that the algorithm did not take data dependencies into account. We now propose algorithm improvements by reducing the effects of data dependencies through the implementation of a block bootstrap resampling method, and solving only over integer values to improve accuracy.

The organization of the paper is as follows: We give an overview of the k-NN algorithm and its application to local dimension estimation in Section 2. The problem of bias in dimension estimation and a de-biasing framework using data depth analysis are introduced in Section 3. 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

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

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 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 $\mathbb{R}^d$ ($m < d$). A Riemann manifold has an associated metric g [7], which endows $\mathcal{M}$ with both a notion of distance via geodesics and also a measure μ_g via the differential volume element. Under this framework, the asymptotic behavior of (1) is given by the following theorem [4]:

*Acknowledgement: This work is partially funded by the National Science Foundation, grant No. CCR-0325571.

Theorem 1. *Let $(\mathcal{M}, g)$ be a compact Riemann m-dimensional submanifold of $\mathbb{R}^d$. Suppose $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n$ are i.i.d. random vectors of $\mathcal{M}$ with bounded density f relative to μ_g. Assume $m \geq 2$, $1 \leq \gamma < m$ and define $\alpha = (m-\gamma)/m$. Then, with probability 1,*

$$\lim_{n\to\infty} \frac{L_{\gamma,k}(\mathcal{Y}_n)}{n^{(d'-\gamma)/d'}} = \begin{cases} \infty, & d' < m \\ \beta_{m,\gamma,k} \int_{\mathcal{M}} f^{\alpha}(\boldsymbol{y})\, \mu_g(\mathrm{d}\boldsymbol{y}), & d' = m \\ 0, & d' > m \end{cases}, \tag{2}$$

where $\beta_{m,\gamma,k}$ is a constant independent of f and $(\mathcal{M}, g)$. Furthermore, the mean length $E\left[L_{\gamma,k}(\mathcal{Y}_n)\right]/n^{\alpha}$ converges to the same limit.

From (2), we can make a large n approximation for the total graph length as follows:

$$L_{\gamma,k}(\mathcal{Y}_n) = n^{\alpha} c + \epsilon_n \tag{3}$$

where

$$c = \beta_{m,\gamma,k} \int_{\mathcal{M}} f^{\alpha}(\boldsymbol{y})\, \mu_g(\mathrm{d}\boldsymbol{y}) \tag{4}$$

The estimate of the intrinsic dimension $\hat{m}$ can be found using a non-linear least squares solution, by calculating graph lengths over varying values of n. Since c is dependent on m, it is necessary to solve for the minimum mean squared error by minimizing over both c and integer values of $m \in \mathbb{Z}$ (5). We solve over integer values of m as we do not consider fractal dimensions for this algorithm. This improves accuracy by constraining the estimation space to discrete values, rather than discretizing estimates in a continuous space.

$$\hat{m} = \arg\min_{m\in\mathbb{Z}} \{ \min_{c} \sum_{n} (L_n - n^{\alpha(m)} c)^2 \} \tag{5}$$

In order to calculate graph lengths for differing sample sizes on the manifold, we randomly subsample from the full set. Using an i.i.d. bootstrap - randomly selecting individual points - is sufficient when the data is independent. However, when dependencies lie in the data the i.i.d. bootstrap breaks down, as the random subsampling can remove all temporal and/or spatial correlation between the data points. In these cases, a better subsampling method is the block bootstrap for dependent data, many types of which are described in [8]. This leads to more consistent results in the k-NN algorithm.

For our purposes, we will utilize the non-overlapping block bootstrapping method on the data set $\mathcal{Y}_n = \{\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n\}$. Specifically, let $p_1, \ldots, p_Q$, $1 \leq p_1 < \ldots, < p_Q \leq n$, be Q integers and let w be an integer satisfying $w < n/Q$. Let $\mathcal{Y}'_n = \{\boldsymbol{Y}_{(1)}, \ldots, \boldsymbol{Y}_{(n)}\}$ be a spatially or temporally sorted version of $\mathcal{Y}_n$. Define the blocks $\mathcal{B}_i = (\boldsymbol{Y}_{(i-1)w+1}, \ldots, \boldsymbol{Y}_{iw})$, $i = 1, \ldots, n/w$. For each value of $p \in \{p_1, \ldots, p_Q\}$ randomly draw N bootstrap datasets $\mathcal{B}_p^j$, $j = 1, \ldots, N$, without replacement, where the p data points within each $\mathcal{B}_p^j$ are chosen from the entire data set $\mathcal{B}_n$ independently.

2.1. Local Dimension Estimation

The k-NN algorithm in itself is a global dimension estimator. We are able to adopt it (and any other dimension estimation algorithm) as a local dimension estimator by running the algorithm over a smaller neighborhood about each sample point. Intuitively, if an m-dimensional manifold, $\mathcal{M}$, has a uniform distribution over n points, $\mathcal{Y}_n = \{\boldsymbol{Y}_1 \ldots \boldsymbol{Y}_n\}$, then any small sphere or data cluster $\mathcal{S} \subseteq \mathcal{M}$, centered at point $\boldsymbol{Y}_i$ will also have uniform distribution over $n' \leq n$ data points. As such, we can use the global dimension estimation algorithm on a local subset of the data to estimate the local intrinsic dimension of each sample point. This can be performed as described in Algorithm 1, where 'dimension($\mathcal{C}$)' refers to applying any method of dimension estimation (such as the k-NN algorithm) to the data set $\mathcal{C}$.

Algorithm 1 Local dimension estimation

Input: Data set $\mathcal{Y} = \{\boldsymbol{Y}_1, \boldsymbol{Y}_2, \ldots, \boldsymbol{Y}_n\}$
1: **for** $i = 1$ to n **do**
2: Initialize cluster $\mathcal{C} = \boldsymbol{Y}_i$
3: **for** $k = 1$ to n' **do**
4: Find the k-th NN, $\boldsymbol{Y}_{k,i}$, of $\boldsymbol{Y}_i$
5: $\mathcal{C} \leftarrow \mathcal{C} \cup \boldsymbol{Y}_{k,i}$
6: **end for**
7: $\hat{m}(\boldsymbol{Y}_i) = \text{dimension}(\mathcal{C})$
8: **end for**
Output: Local dimension estimates $\hat{m}$

One of the keys to local dimension estimation is defining a value of n'. There must be a significant number of samples in order to obtain a proper estimate, but it is also important to keep a small sample size as to (ideally) only include samples which lie on the same manifold. Currently we arbitrarily choose n' based on the size of the data set. However, a more definitive method of choosing n' is grounds for future work.

3. DATA DEPTH AND DIMENSION ESTIMATION

To our knowledge, a phenomenon common to all algorithms of intrinsic dimension estimation is a negative bias in the dimension estimate. It is believed that this is an effect of undersampling the high dimensional manifold. While the bias due to lack of sufficient samples is inherent, we offer that the sample size is not the only source of bias; a significant portion is related to the depth of the data.

Specifically, as data samples approach the boundaries of the manifold, they exhibit a lower intrinsic dimension. Consider the m-dimensional unit hypercube $\mathcal{A} = [0,1]^m$. One can define the interior as the set $\mathcal{I} = \{x \mid \frac{\epsilon}{2} \leq x_i \leq 1 - \frac{\epsilon}{2}\}$. The ϵ-boundary is therefore $\partial\mathcal{A} = \mathcal{A}/\mathcal{I}$. The following statement can be made:

Proposition 1. *With probability of at least $1-\delta$, a uniformly selected x from $\mathcal{A}$ is contained in the boundary $\partial\mathcal{A}$, i.e., $x \in \partial\mathcal{A}$ and $\epsilon = \frac{\log(1/\delta)}{m}$.*

Proof. Since x is uniform in $\mathcal{A}$, its components are *i.i.d.* uniform random variables $U[0,1]$. The probability of x being in the interior $\mathcal{I}$ is therefore given by the product

$$P(x \in \mathcal{I}) = \prod_{i=1}^{m} P\left(\frac{\epsilon}{2} \leq x_i \leq 1 - \frac{\epsilon}{2}\right) = (1-\epsilon)^m.$$

Therefore, the probability of $x \in \partial\mathcal{A}$ is

$$\begin{aligned} P(x \in \partial\mathcal{A}) &= 1 - (1-\epsilon)^m \\ &= 1 - \exp(m \log(1-\epsilon)). \end{aligned}$$

Since $\log(1+t) \leq t$, we have $\exp(m\log(1-\epsilon)) \leq \exp(-m\epsilon)$ and therefore

$$P(x \in \partial\mathcal{A}) \geq 1 - \exp(-m\epsilon).$$

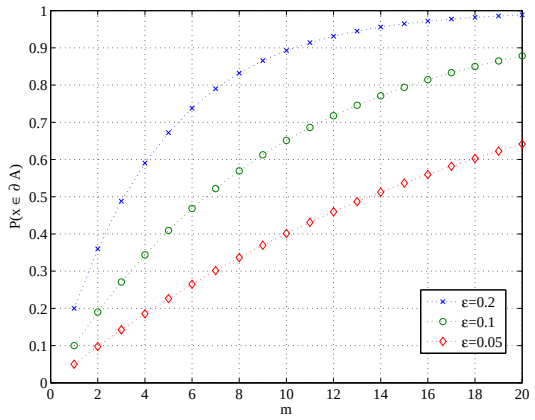


Fig. 1. The probability of randomly selecting a point on the boundary of an m-dimensional hypercube for $\epsilon = 0.2$ ($\times$), $\epsilon = 0.1$ ($\circ$), and $\epsilon = 0.05$ ($\diamond$).

For $\epsilon = \frac{\log(1/\delta)}{m}$, we have

$$P(x \in \partial\mathcal{A}) \geq 1 - \exp\left(-m\frac{\log(1/\delta)}{m}\right) = 1 - \delta.$$

□

This result suggests that at least $1-\delta$ of the entire points in the hypercube are concentrated in a boundary with $\epsilon \to 0$ as $m \to \infty$. Alternatively, for large m most points in a hypercube will concentrate on its boundary (see Fig. 1).

We proceed by suggesting that the boundary of the m-dimensional hypercube can be approximated as an $(m-1)$-dimensional manifold and hence should produce a lower dimension estimate. Clearly, a simple average of the dimension estimate over the manifold will consider many more points $(1-\delta)$ on the boundary with a lower dimension as compared with the number of points in the interior of the hypercube (δ), leading to a lower dimension estimate.

We are able to further justify the effect of data depth on dimension estimation by calculating the depth of each sample and analyzing the relationship between depth and dimension. We utilize the L_1-data depth algorithm developed in [9], which calculates depth as the sum of all the unit vectors between the interested sample $y \in \boldsymbol{X}$ and the rest of the data set, $\boldsymbol{X} = \{x_1, \ldots, x_n\}$. Specifically,

$$D_n(y) = 1 - \max\Big(0, \Big\|\sum_{x_i \neq y} e(x_i - y)/n\Big\| - \sum_{x_i = y}\frac{1}{n}\Big) \qquad (6)$$

where $e(x_i - y) = (x_i - y)/\|x_i - y\|$ is the unit vector in the direction of $(x_i - y)$. This depth metric assigns the most interior points in the data set a depth value approaching 1, while samples along the boundaries approach a depth of 0.

Using this metric, we illustrate the effect of data depth on dimensional estimation in Fig. 2. The data set of use was of 3000 points uniformly sampled on a 6-dimensional hyperplane. We utilize the maximum likelihood method for dimension estimation [5] to demonstrate that the negative bias is inherent to dimension estimation, and not specific to a given algorithm. Figure 2 illustrates the distribution of data depths for samples that estimate at different dimensions. It is clear that the samples with more depth estimate at a high dimension, while the points closer to the boundaries estimate at lower dimensions. When developing a global dimension estimate,

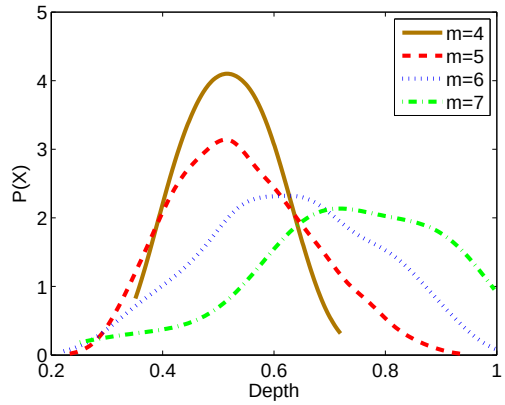
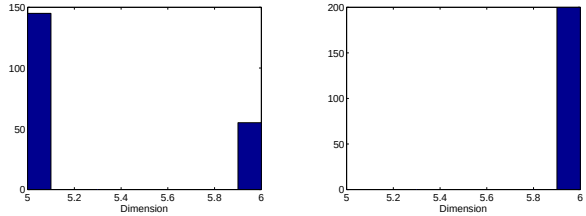


Fig. 2. Analysis of the effect of data depth on local dimension estimation. Points with less depth estimate at a lower dimension, contributing to the overall negative bias.



(a) Biased Results (b) De-biased Results

Fig. 3. Developing a de-biased global dimension estimate by averaging over the 50% of points with the greatest depth on the manifold

these points will contribute heavily to the negative bias. As such, when estimating the global dimension of a data set, one can substantially reduce the negative bias by considering the local dimension of those points away from the boundaries, as these points are more indicative of the true dimension of the manifold. This is illustrated in Fig. 3, in which we estimated the global dimension (by averaging local dimension estimates) of the 6-dimensional hypercube over 200 unique trials. Figure 3(a) shows the histogram of biased dimension estimates obtained by using the entire set for dimension estimation, while Fig. 3(b) obtains correct dimension estimate each trial by using our de-biasing method. It is clear that our method has a strong effect on removing the bias. While we only averaged over the deepest 50% of the samples for this example, the optimal depth at which to consider samples for a dimension estimate is still an open problem.

4. SIMULATION RESULTS

4.1. Algorithm Comparison

To illustrate the improvements to the k-NN algorithm for dimension estimation, we compare the versions on a data set consisting of two distinct manifolds; 300 points uniformly sampled on a "swiss roll", which has an intrinsic dimension of 2, and 150 points uniformly sampled on a hyper-sphere with intrinsic dimension of 3. Both manifolds were embedded into the same 5-dimensional space. We estimated the local dimension of each sample in the data set, and calculated the probability of error (P_e). This experiment was run 10 times,

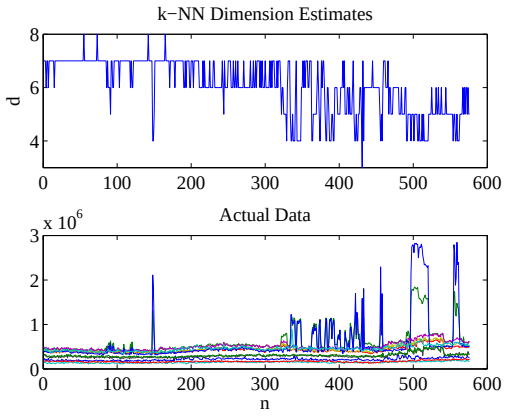


Fig. 4. The k-NN Algorithm applied to network traffic data

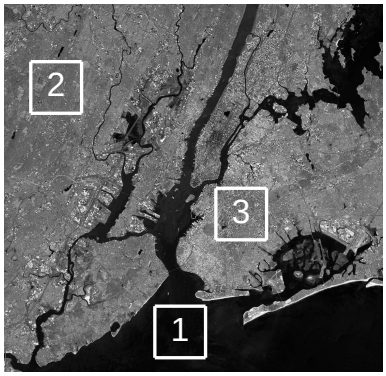


Fig. 5. Satellite image of New York City; three regions of differing complexity are noted

with different points sampled on the manifolds in each run, and we show the mean and variance of P_e in Table 1. It is clearly shown that the methods discussed in this paper have a significant improvement on both the probability of error and the variance of estimation results across simulations.

k-NN Version	Mean(P_e)	Var(P_e)
New k-NN	0.014	0.0001
Old k-NN	0.092	0.0025

Table 1. Comparison of k-NN algorithm versions

4.2. Abilene Network Data

The Abilene network is the set of routers which is the backbone of the '.edu' network. When an anomaly occurs on the network, there are changes in the correlation between traffic traces at different points in the system, imposing nonlinear constraints on the observed data. We believe that during an anomaly the intrinsic dimension of the data will change. This was discussed with respect to analysis of individual routers in [10]. We apply the theory to the network as a whole. Specifically, we hypothesize that when only a few of the routers contribute disproportionately large amounts of traffic, the intrinsic dimension of the entire network should decrease.

The data set used in Fig. 4 is the number of packets counted during 5 minute intervals on each of the 11 Abilene routers. Using $d = 11$ as the extrinsic dimension, we applied the k-NN algorithm to estimate the intrinsic dimension of the network at each time sample. The results of the algorithm illustrate that our hypothesis was correct. In the time instances in which routers displayed increased and disproportional contributions to the overall network traffic, the intrinsic dimension decreased. Figure 4 shows that we are able to detect the anomalous activity, such as at the visually obvious $n = 148$. Moreover, the k-NN algorithm is able to pick out the non-obvious complexity changes as well. This is illustrated with the change in dimension at the time instance $n = 244$. A detailed investigation reveals that the Sunnyvale router showed increased contribution from a single IP address. Large percentages (over half) of the overall packets had both source and destination IP 128.223.216.xxx within port 119. The same port showed increased activity on the Atlanta router during this time period as well. Without a tool such as the k-NN algorithm, these changes in the network topology would most likely go unnoticed by strictly viewing the plot of actual data in Fig. 4.

4.3. Image Segmentation

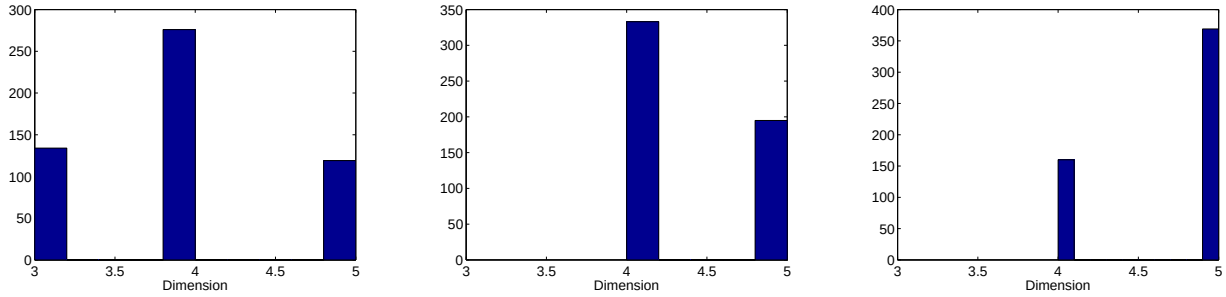
There are many problems in which knowing the exact intrinsic dimension is unnecessary, as there may be no real life interpretation of the value. Instead, a measure of *complexity* is desired to distinguish between data. In these situations, dimension estimation is can be used as a means of differentiation by complexity. As an illustration, let us consider an image which contains regions of varying textures. It is desirable to segment this image into the various regions, based on some measure of complexity. We will briefly demonstrate this ability with dimension estimation.

Consider Fig. 5, which is a satellite image of New York City[1]. We have identified three regions with varying complexities, and we will illustrate the uses of dimension estimation for distinguishing between them. We hypothesize that as the complexity increases, so will the estimate of the dimension. For our purposes, we segmented each region into 3x3 pixel blocks, and considered each block as a 9-dimensional vector. Each region is described as $\boldsymbol{X}_i = \{x_1, \ldots, x_n\}$ where $x_j \in \mathbb{R}^9$ and n is the number of blocks in the region. We then used the maximum likelihood method [5] to estimate the local dimension of each block. In Fig. 6 we plot the histogram results of the local dimension estimates of each block, for each region. As we expected, the histogram of the dimension estimates increases from the region with the least visual complexity (region 1) to the region with the most visual complexity (region 3). This type of analysis could be used to segment the entire image into different regions.

5. CONCLUSIONS

We have shown that the negative bias in dimension estimation is strongly influenced by the data depth of the samples on the manifold. As samples approach the boundaries of the manifold, they perceive the local intrinsic dimension to be lower than that of the entire manifold, contributing to a negative bias on the global dimension estimate. While this issue is somewhat alleviated with increased

[1] http://newsdesk.si.edu/photos/sites_earth_from_space.htm



(a) Region 1 (b) Region 2 (c) Region 3

Fig. 6. Dimension estimation can be used for image segmentation by observing the difference in estimates of region *complexity*

sampling, that is usually not a legitimate option. As such, we propose de-biasing the dimension estimate of a manifold by considering the local dimension of those points significantly 'deep' into the manifold. We point out that as the dimension increases, the number of interior point decreases (holding total number of points constant). As such, using only the interior points in averaging over local dimensions may result in large variance of the dimension estimate due to a small sample size. The bias-variance trade-off and its optimization is of great importance, and should be considered an area for future work.

Additionally, we proposed improvements to the algorithm described in [6], better distinguishing disjoint manifolds in a global space. The new k-NN algorithm offers a dramatic improvement over the previous work on real and synthetic data sets. Dimension estimation has many uses, and we have shown practical applications using intrinsic dimension in the analysis of network traffic and image segmentation. Future improvements include adaptively building the k-NN graphs by adjusting the sample neighborhoods according to properties of the data set, as well as continued use of local dimension estimation as an anomaly detection method for use in time-series analysis.

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, as well as Eric Kolaczyk from Boston University for discussions on the block bootstrap.

7. REFERENCES

[1] A.D. Lanterman, "Schwarz, Wallace, and Rissanen: Intertwining themes in theories of model order estimation," *International Statistical Review*, vol. 69, pp. 185–212, August 2001.

[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. A. Costa and A. O. Hero, "Geodesic entropic graphs for dimension and entropy estimation in manifold learning," *IEEE Trans. on Signal Processing*, vol. 52, no. 8, pp. 2210–2221, August 2004.

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

[6] J. Costa, A. Girotra, and A. Hero, "Estimating local intrinsic dimension with k-nearest neighbor graphs," *IEEE Workshop on Statistical Signal Processing (SSP)*, pp. 417–422, July 2005.

[7] M. Carmo, *Riemannian geometry*, Birkhäuser, Boston, 1992.

[8] S.N. Lahiri, *Resampling Methods for Dependent Data*, Springer, NY, USA, 2003.

[9] Y. Vardi and C.-H Zhang, "The multivariate $L1$-median and associated data depth," *Proceedings of the National Academy of Science USA*, vol. 97, pp. 1423–1426, 2000.

[10] A. Lakhina, M. Crovella, and C. Diot, "Diagnosing network-wide traffic anomalies," *Proceedings of ACM SIGCOMM*, pp. 219–230, Aug. 2004.