

# On Point Sampling versus Space Sampling for Dimensionality Reduction

Charu C. Aggarwal\*

## Abstract

In recent years, random projection has been used as a valuable tool for performing dimensionality reduction of high dimensional data. Starting with the seminal work of Johnson and Lindenstrauss [8], a number of interesting implementations of the random projection techniques have been proposed for dimensionality reduction. These techniques are mostly *space symmetric* random projections in which random hyperplanes are sampled in order to construct the projection. While these methods can provide effective reductions with worst-case bounds, they are not sensitive to the fact that the underlying data may have much lower implicit dimensionality than the full dimensionality. This may often be the case in many real applications. In this work, we analyze the theoretical effectiveness of *point sampled* random projections, in which the sampled hyperplanes are defined in terms of points sampled from the data. We show that point sampled random projections can be significantly more effective in most data sets, since the implicit dimensionality is usually significantly lower than the full dimensionality. In pathological cases, where space sampled random projections are better, it is possible to use a mixture of the two methods to design a random projection method with excellent average case behavior, while retaining the worst case behavior of space sampled random projections.

**Keywords:** Dimensionality Reduction, Random Projection

## 1 Introduction

Dimensionality Reduction is well known as an effective tool to improve the compactness of the data representation. A well known technique for dimensionality reduction is the method of Singular Value Decomposition [11, 9, 5] (SVD), which projects the data into a lower dimensional subspace. The idea is to transform the data into a new orthonormal coordinate system in which the second order correlations are eliminated. In typical applications, the resulting axis-system has the property that the variance of the data along many of the dimen-

sions in the new coordinate system is very small [9]. These dimensions can then be eliminated, a process resulting in a compact representation of the data with some loss of representational accuracy.

In recent years, the technique of random projection [1, 7, 10] has often been used as an efficient alternative for dimensionality reduction of high dimensional data sets. The idea in random projection is to use spherically symmetric projections, in which arbitrary hyperplanes are sampled repeatedly in order to create a new axis system for data representation. We refer to this technique as a *space sampled random projection*, since the sampled hyperplanes are independent of the underlying data points. To our knowledge, most known results (such as the seminal Johnson-Lindenstrauss result [8], and its subsequent extensions for random projection techniques use space sampled random projections. A different method is that of *point sampled random projections* in which points from the space are sampled in order to create the projections. Specifically, if we sample  $k$  points from the data, it creates a space with dimensionality at most  $(k - 1)$ . We note that the use of point sampled projections automatically eliminates many irrelevant subspaces which would be picked by a space sampled random projection.

In order to intuitively understand this point, we will illustrate with the use of two examples. The first example in Figure 1 illustrates 1-dimensional projections of 2-dimensional data. Consider the data set illustrated in Figure 1 in which we have illustrated two kinds of projections. In Figure 1(a), the data *space* is sampled in order to find a 1-dimensional line along which the projection is performed. The reduced data in this 1-dimensional representation is simply the projection of the data points onto the line, as illustrated in the lower diagram of Figure 1(a). This corresponding 1-dimensional projection is a poor representation of the underlying patterns in the data. This is because space sampled random projections are independent of the underlying data distribution. In Figure 1(b), we have illustrated an example of a point-sampled random projection. In this case, this projection happens to be the 1-dimensional line passing through two

\*IBM T. J. Watson Research Center, charu@us.ibm.com

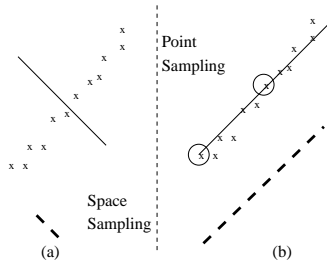


Figure 1: Point Sampled and Space Sampled Random Projections (2-dim. example)

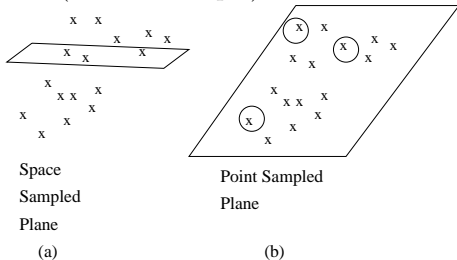


Figure 2: Comparing Point Sampled and Space Sampled Random Projections (3-dim. example)

randomly sampled points in the data. It is clear from Figure 1(b), that this 1-dimensional line points picks up the directions of greater variance more effectively than the space-sampled random projection of Figure 1(a). As a result, the quality of the reduction in Figure 1(b) is significantly more effective than that in Figure 1(a).

A similar behavior is illustrated in Figure 2, in which the space sampled random projection of Figure 2(a) shows very little alignment along the natural subspaces represented by Figure 2(a). This is not the case for point sampled random projections in which any set of 3 randomly picked points can define the subspace along which the data is naturally aligned. Though repeated applications of (space sampled) random projections [1, 8] provide bounds on data reduction quality, it is also evident that space sampled random projections are often wasteful, since they do not use the behavior of the underlying data. In general, the lower the implicit dimensionality (compared to the full dimensionality), the more likely it is that point sampled projections can use the special structure of the underlying data. In this paper, we will analyze the behavior of point sampled random projections and show that it is usually more effective than space sampled random projections. In order to handle the pathological cases in which space-sampled random projections are better, we can use a mixture of the two methods. The mixture can provide results which are significantly better than the space sampled method in most cases, and almost as good in the worst

case. In many cases, much fewer number of samples are required by the point sampled projection process to achieve the same quality. Therefore, the addition of point sampled random projections to the mixture improves the efficiency of the reduction method by more than an order of magnitude.

This paper is organized as follows. In the next section, we will discuss and analyze point sampled random projections. In section 3, we will present the experimental results. The conclusions and summary are discussed in section 4.

## 2 Point Sampled Random Projections: Discussion and Analysis

In this section, we will discuss the process of performing point sampled random projections, and analyze its effectiveness. We will show that while point sampled random projections do not provide (distribution independent) hard bounds like space sampled projections, they can often be a much more effective tool in preserving the dimensionality of the data. First, we will provide some understanding of the philosophy behind space-sampled random projections, and why they work well for the purpose of dimensionality reduction.

We note that in random projection, we attempt to create a *distance-preserving* transform which may require a linear scaling of the data. The basic idea behind the approach is that the  $k$ -dimensional random projection of the distance between two points onto a random set of vectors is equivalent to the projection of a random vector (of the same length as the distance between the points) on to any  $k$ -coordinates from a fixed  $d$ -dimensional orthogonal coordinate system. It can be shown that a random vector of length  $L$  in  $d$ -dimensional space (when projected onto  $k$  dimensions) has an expected length of  $\sqrt{k/d} \cdot L$ , and this length is “sharply concentrated” around the mean. This sharp concentration is defined in terms of Chernoff bounds, which provide probabilistic guarantees on the length lying between  $\sqrt{k/d} \cdot L \cdot (1 - \epsilon)$  and  $\sqrt{k/d} \cdot L \cdot (1 + \epsilon)$ . By scaling the vector by a factor of  $\sqrt{d/k}$ , it is possible to obtain a vector which is within a pre-defined tolerance of the original vector length  $L$  using certain probabilistic guarantees. Thus, in order to preserve pair-wise distances in the random projection technique, we project all points onto a randomly chosen  $k$ -dimensional hyperplane, and then scale all data points by the factor  $\sqrt{d/k}$ . By picking  $k = O(\log(N)/\epsilon^2)$  and repeating the random projection process  $N$  times, it is possible to show that pairwise distances are preserved by at least one of these projections to within a tolerance of  $\epsilon$  with fixed probability. We note that the above description of why random projection works is a concise

explanation of the simplification of the original proof [8] in [4].

**2.1 How Strong are the Johnson-Lindenstrauss Bounds Really?** Consider a data set containing only  $N = 2$  points, and let us examine the effect of the random projection process on the relative distance of these two points from the origin. According to the Johnson-Lindenstrauss bound, the error tolerance  $\epsilon$  of the distance of each of this pair of points from the origin in the projected data grows with  $\sqrt{1/k}$  where  $k$  is the dimensionality of the projection. This tolerance needs to be examined in the context of the overall behavior of high dimensional data. Recent results [6] show that in high dimensional space, all pairwise distances become identical because of data sparsity. Under certain distribution assumptions [6], the proportionate difference between the maximum and minimum distances from any target point (such as the origin) grow with  $1/\sqrt{d}$ , where  $d$  is the overall dimensionality of the data.

**LEMMA 2.1.** [6] *Consider a distribution of  $N = 2$  data points drawn from the  $d$  dimensional space  $\mathcal{F}^d$  with i.i.d. dimensions, where  $\mathcal{F}$  is any 1-dimensional distribution with non-zero variance. Let  $D_{max}$  be the maximum distance of this pair from the origin, and  $D_{min}$  is the minimum distance. Then, we have the following result:*

$$(2.1) \quad \frac{D_{max} - D_{min}}{D_{max}} \rightarrow_p 1/\sqrt{d}$$

In the above expression, the symbol  $\rightarrow_p$  corresponds to convergence in probability with dimensionality  $d$ . Thus, it has been shown in [6] that  $(D_{max} - D_{min})/D_{min}$  grows (shrinks) proportionally to  $1/\sqrt{d}$  with increasing  $d$ . While the distribution assumptions in [6] rely on i.i.d. dimensions, it can generally be assumed that this behavior of the expression  $(D_{max} - D_{min})/D_{min}$  may grow proportionally with  $1/\sqrt{d^*}$ , where  $d^* \leq d$  is the *implicit* dimensionality of the data. Therefore, if the dimensionality of the projection  $k$  is chosen less than  $d^*$ , the Johnson-Lindenstrauss tolerance guarantees (which grow with  $\sqrt{1/k}$ ) are asymptotically larger than  $(D_{max} - D_{min})/D_{min}$ . Therefore, the nearest of the pair of data points may become the furthest pair (after projection) and vice-versa. Even when  $k$  is chosen to be larger than  $d^*$ , the tolerance may be a large fraction  $\sqrt{d^*/k}$  of  $(D_{max} - D_{min})/D_{min}$ . This also has implications for the meaningfulness of worst-case bounds in nearest neighbor techniques [7], which use random projection techniques to provide such guarantees. While much has been made of such bounds in the random projection technique, the above argument shows that the Johnson-Lindenstrauss bounds are actually quite weak

when viewed in context of the *overall sparsity behavior* of high dimensional data. This does not mean that random projection is a poor approach in the average case. In fact, recent empirical results show [3] that random projection is indeed a useful tool which provides a high level of retrieval effectiveness in real applications. However, it does not leverage the behavior of the underlying data effectively. It is the aim of this paper to develop a random projection technique which can leverage the lower implicit dimensionality of real data sets in order to further improve both the effectiveness and efficiency of the projection process in the average case. We will show that point sampled random projections are much more effective than space sampled random projections, when the underlying implicit dimensionality of the data is small compared to the full dimensionality.

**2.2 Random Projection and PCA** We note that while random projection is often used as an alternative to other dimensionality reduction techniques such as Principal Component Analysis (PCA) [9], the two methods are quite different in many respects. The Principal Component Analysis technique uses the covariance behavior of the data to optimize the direction of the projection, so that the least amount of variance is lost. On the other hand, the random projection approach does *not* attempt to optimize the direction of the projection, but depends upon the fact that pair-wise *proportionate* distances are maintained between different points by a randomly chosen projection. By choosing an appropriate scaling factor ( $\sqrt{d/k}$ ), *absolute* distances can also be maintained within the same factor. After scaling, the new data set may have more or less variance than the original data. The inability of space sampled random projections to use the underlying distribution of the data leads us to naturally explore the possibility that *point sampled random projections* may lead to a much more effective dimensionality reduction, since it uses the underlying distribution of the data. In the following description, we will discuss our proposed implementation of point sampled random projections, and its application to dimensionality reduction.

Another key difference between random projection and PCA is that random projection should be viewed as a *distance preserving embedding*, whereas PCA should be viewed as a pure axis-rotational transformation. The reason for this key difference is that random projection preserves the distance bounds only after multiplicatively scaling by a factor  $\sqrt{d/k}$ , whereas the PCA approach is a pure axis-rotational transformation without any kind of scaling. (In practice, the multiplicative scaling never needs to be performed since most data analysis applications only require preservation of *proportional*

**Algorithm** *PointSampledProject*(Data:  $\mathcal{D}$ , MaxProjected: $k$ ,  
MaxSamples:  $numsamp$ );

**begin**

Determine centroid  $\bar{x}$  and variance  $v$   
of database  $\mathcal{D}$ ;  
Determine  $numsamp$  sets of points with  $k$  points  
 $\mathcal{S}_1 \dots \mathcal{S}_{numsamp}$ ;  
 $\forall i \in \{1 \dots numsamp\}$  orthogonalize each set  $\mathcal{S}_i$   
to determine the set  $\mathcal{E}_i$ ;  
 $\forall i \in \{1 \dots numsamp\}$  project database  $\mathcal{D}$   
onto set  $\mathcal{E}_i$  to determine  $\mathcal{D}_i$   
while computing centroid  $\bar{x}_i$  and variance  $v_i$   
of projected database;  
 $\forall i \in \{1 \dots numsamp\}$  multiply each entry in  $\mathcal{D}_i$   
by  $\sqrt{v/v_i}$  for normalization while computing the  
centroid-distance error;  
Pick the projection  $\mathcal{D}_i$  with the least error;

**end**

Figure 3: The Point Sampled Random Projection Algorithm

distances.)

### 2.3 A Simple Implementation of Point Sampled Random Projections

In this section, we will discuss a simple implementation of point-sampled random projections for dimensionality reduction. As discussed earlier, the point sampled random projection technique requires  $(k+1)$  points from the data in order to generate a projection of dimensionality at most  $k$ . In practice, we sample the centroid of the data along with  $k$  other random points from the data. This provides us with  $(k+1)$  data points, which we denote by  $\bar{y}_1 \dots \bar{y}_{k+1}$ . We need to find an orthogonal axis-system  $\mathcal{E} = \{\bar{e}_1 \dots \bar{e}_k\}$  corresponding to the plane on which these  $k+1$  data points may be found. The first step is to initialize a set of vectors  $\bar{f}_1 \dots \bar{f}_k$  as follows:

$$(2.2) \quad \bar{f}_i = (\bar{y}_{i+1} - \bar{y}_1) / \|\bar{y}_{i+1} - \bar{y}_1\|$$

This ensures that  $\mathcal{E} = \{\bar{f}_1 \dots \bar{f}_k\}$  is a set of vectors parallel to the plane defined by  $\bar{y}_1 \dots \bar{y}_{k+1}$ . However, the vectors  $\bar{f}_1 \dots \bar{f}_k$  will typically not be orthogonal to one another. These vectors can be orthogonalized efficiently in  $k$  iterations by iteratively subtracting out the components of  $\bar{f}_i$  onto the current orthogonal set  $\bar{e}_1 \dots \bar{e}_{i-1}$ . Therefore, we recursively define the new set of vectors  $\bar{e}_1 \dots \bar{e}_k$  as follows:

$$\bar{e}_i = (\bar{f}_i - \sum_{j=1}^{i-1} [\bar{f}_i \cdot \bar{e}_j] \bar{e}_j) / \|\bar{f}_i - \sum_{j=1}^{i-1} [\bar{f}_i \cdot \bar{e}_j] \bar{e}_j\|$$

It is easy to verify by induction that the set of vectors  $\bar{e}_1 \dots \bar{e}_k$  form an orthonormal axis system.

We note that the time-complexity of performing the orthogonalization is asymptotically small as compared to the complexity of performing the random projection itself. In a later subsection, we will show that the time complexity of performing a space sampled and point sampled random projection is asymptotically the same. Next, we will discuss a straightforward point sampled random projection algorithm using the above discussion.

The overall algorithm for performing the random projection is illustrated in Figure 3. We assume that the centroid and variance of the original database  $\mathcal{D}$  are denoted by  $\bar{x}$  and  $v$  respectively. In order to perform the projection, we pick samples of  $k$  data points along with the centroid  $\bar{x}$  of the original data in order to create  $(k+1)$  representative data points. The  $i$ th set of  $(k+1)$  representative points is denoted by  $\mathcal{S}_i$ . We first use the orthogonalization process discussed above to create the orthogonal set of vectors  $\mathcal{E}_i$  from  $\mathcal{S}_i$ . We find all the orthogonalized subspace representations from the different samples before actually performing the projection. This is done so that the final projections can be performed using a single pass over the data. Once all the orthogonalized subspace representations have been computed, we determine the projections of the original database onto these subspace representations. During the projection process, the variance  $v_i$  of the projected database  $\mathcal{D}_i$  is computed. We note that this can be done during the projection process itself since the variance can be computed in a single scan of the data. Each subspace representation is normalized with the factor  $\sqrt{v/v_i}$ , which is analogous to the normalization of space sampled random projections with the factor  $\sqrt{d/k}$ . Then, we compute the *normalized average error* of the projection, which is defined in terms of how much the distance of each data point to the centroid has changed because of the transformation. Let  $dist(\mathcal{D}_i, \bar{x}, \bar{X}_j)$  denote the distance of the data points  $\bar{X}_j \in \mathcal{D}$  from from  $x$ , in the projected and normalized representation corresponding to database  $\mathcal{D}_i$ . Then, the centroid error  $CE(\mathcal{D}_i, \mathcal{D})$  for the database  $\mathcal{D}_i$  is defined as follows:

$$CE(\mathcal{D}_i, \mathcal{D}) = \sum_{\bar{X}_j \in \mathcal{D}} \frac{\|dist(\mathcal{D}, \bar{x}, \bar{X}_j) - dist(\mathcal{D}_i, \bar{x}, \bar{X}_j)\|}{(N \cdot dist(\mathcal{D}, \bar{x}, \bar{X}_j))}$$

Note that we have chosen to define the error in terms of intra-point distances, because unlike PCA (which is an energy preserving transform), both space sampled and point sampled random projections are actually embeddings which preserve intra-point distances. However, instead of measuring worst-case intra-point distances (as in the Johnson-Lindenstrauss result), we have used the *average* fractional error of the distance to the centroid

as a more stable representative of the qualitative results in real applications. This error quantification provides an idea of the proportion of the error in distances which are maintained by the reduction process.

**2.4 Computational Complexity** The computational complexity of point sampled random projections is asymptotically the same as that of space sampled random projections. The space sampled random projection process requires us to project the data onto each of  $numsamp$   $k$ -dimensional projections. Therefore, the space sampled random projection has a computational complexity of  $O(k \cdot N \cdot numsamp)$   $d$ -dimensional vector operations for a database with  $N$  points. This projection process needs to be performed for the point sampled random projection process as well, except that we need to perform an additional orthogonalization process, which requires  $k$  iterations, and the  $i$ th iteration requires  $i$   $d$ -dimensional vector operations. Therefore, the overall complexity of the orthogonalization process is  $O(k^2 \cdot numsamp)$   $d$ -dimensional vector operations. Thus, the overall time complexity of point sampled random projections is given by  $O(k \cdot (k + N) \cdot numsamp)$  vector operations. We note that the dimensionality of the projection  $k$  is typically negligible compared to the number of data points  $N$ , and therefore the overall complexity of point sampled random projections is given by  $O(k \cdot N \cdot numsamp)$  vector operations, which is the same as that of space sampled random projections. Furthermore, our subsequent analysis and experimental results will show that since point sampled random projections leverage the behavior of the underlying data, they typically require orders of magnitude fewer projection samples to achieve the same or better qualitative results. This will mean that in practice, the point sampled random projection process will have significantly better computational complexity.

**2.5 Theoretical Analysis of Point Sampled Random Projections** In this section, we will analyze the theoretical effectiveness of point sampled random projections. We note that the effectiveness of point sampled random projections depends upon the fact that the data is often embedded in a much lower dimensional subspace than the full dimensional space. Thus, we will try to analyze the effectiveness of the process in such situations. To begin, we make the following straightforward observation about point sampled random projections.

**OBSERVATION 2.1.** *If all data points in  $\mathcal{D}$  are embedded in a  $k$ -dimensional linearly independent subspace  $\mathcal{H}$ , then any set of  $(k + 1)$  sampled linearly independent points from  $\mathcal{D}$  will define  $\mathcal{H}$ .*

In many cases, the data sets may show this kind of behavior because of particular domain specific characteristics which constrain the data to a very low dimensional projection. In such cases, point sampling is a straightforward way to discover the underlying subspaces. In other cases however, this may only be approximately true. For example, it may be possible to find a  $k$ -dimensional hyperplane in  $\mathcal{H}$  from which all data points in  $\mathcal{D}$  lie at a distance of only  $\epsilon > 0$  from  $\mathcal{H}$ . In many practical scenarios, such a  $k$ -dimensional hyperplane can be found that the value of  $\epsilon$  is orders of magnitude smaller than the data variance, and the value of  $k$  is significantly smaller than the full dimensionality  $d$ . In such cases, it is interesting to analyze the effectiveness of point sampled random projections. We make the following claim:

**LEMMA 2.2.** *Let  $\mathcal{H}$  be a hyperplane such that all data points in  $\mathcal{D}$  lie at a distance of at most  $\epsilon$  from  $\mathcal{H}$ . Let  $S$  be a set of randomly sampled  $k$  linearly independent points from  $\mathcal{D}$ , and  $S(\mathcal{H})$  be the projection of all data points onto  $\mathcal{H}$ . Let  $x$  be any data point from  $\mathcal{D}$  and  $x_{\mathcal{H}}$  be the projection of  $x$  onto  $\mathcal{H}$ . Let  $L$  be any line passing through  $x_{\mathcal{H}}$  and the convex hull of  $S(\mathcal{H})$ . Let  $p$  be the length of the segment in  $L$  corresponding to the two points of intersection of  $L$  with the convex hull of  $S(\mathcal{H})$ , and let  $q$  be the smallest distance along  $L$  from  $x_{\mathcal{H}}$  to the convex hull of  $S(\mathcal{H})$ . Let  $\mathcal{HS}$  be the hyperplane passing through  $S$ . Then, the projection of  $x_{\mathcal{H}}$  onto  $\mathcal{HS}$  is at a distance of at most  $2 \cdot \epsilon \cdot (p + q)/q$  from  $x_{\mathcal{H}}$ .*

*Proof.* Let the two points of intersection of  $L$  with the convex hull of  $S(\mathcal{H})$  be  $P$  and  $Q$  respectively. Let the set of  $k$  points in  $S$  be denoted by  $Z_1 \dots Z_k$  respectively. Let the projections of  $P, Q, x_{\mathcal{H}}, Z_1 \dots Z_k$  onto  $\mathcal{H}$  be denoted by  $P', Q', x'_{calH}, Z'_1 \dots Z'_k$  respectively. Let the linear transformation corresponding to this projection be denoted by  $f(\cdot) : R^d \rightarrow R^d$ . Since  $P$  and  $Q$  lie on the convex hull of  $S\mathcal{H}$ , there must exist sets of scalars  $\lambda_1 \dots \lambda_k$ , and  $\mu_1 \dots \mu_k$  satisfying the following:

$$\begin{aligned} P &= \sum_{i=1}^k \lambda_i \cdot Z_i \\ \sum_{i=1}^k \lambda_i &= 1 \\ Q &= \sum_{i=1}^k \mu_i \cdot Z_i \\ \sum_{i=1}^k \mu_i &= 1 \end{aligned}$$

By applying the linear transformation to both sides, we

have:

$$f(P) = f\left(\sum_{i=1}^k \lambda_i \cdot Z_i\right) = \sum_{i=1}^k \lambda_i \cdot f(Z_i)$$

$$f(Q) = f\left(\sum_{i=1}^k \mu_i \cdot Z_i\right) = \sum_{i=1}^k \mu_i \cdot f(Z_i)$$

We note that the linear decomposability follows from the linearity of the transformation  $f(\cdot)$ . Since  $P' = f(P)$ ,  $Z'_i = f(Z_i)$ , and  $Q' = f(Q)$ , we have:

$$P' = \sum_{i=1}^k \lambda_i \cdot Z'_i$$

$$\sum_{i=1}^k \lambda_i = 1$$

$$Q' = \sum_{i=1}^k \mu_i \cdot Z'_i$$

$$\sum_{i=1}^k \mu_i = 1$$

Therefore, we have:

$$P - P' = \sum_{i=1}^k \lambda_i \cdot (Z_i - Z'_i)$$

$$\sum_{i=1}^k \lambda_i = 1$$

$$Q - Q' = \sum_{i=1}^k \mu_i \cdot (Z_i - Z'_i)$$

$$\sum_{i=1}^k \mu_i = 1$$

Therefore, we have:

$$\|P - P'\| \leq \sum_{i=1}^k \lambda_i \cdot \|Z_i - Z'_i\|$$

$$\leq \epsilon \cdot \left(\sum_{i=1}^k \lambda_i\right) = \epsilon \|Q - Q'\|$$

$$\leq \sum_{i=1}^k \mu_i \cdot \|Z_i - Z'_i\|$$

$$\leq \epsilon \cdot \left(\sum_{i=1}^k \mu_i\right) = \epsilon$$

The above result follows from the triangle inequality. Now let us examine the line  $L'$  passing through  $P'$  and

$Q'$ . This is essentially the projection of the line  $L$  onto hyperplane  $H$ , and it will also contain the projection  $x'_{\mathcal{H}}$  of  $x_{\mathcal{H}}$ . Since each of  $P$  and  $Q$  are perturbed at most a distance of  $\epsilon$  during the projection process of this line  $L$ , it follows from proportionate distance scaling that the point  $x_{\mathcal{H}}$  is perturbed by a distance of no more than  $2 \cdot \epsilon \cdot (p + q)/q$ .

A simple corollary of the above result is the following:

**COROLLARY 2.1.** *Let  $\mathcal{H}$  be a hyperplane such that all data points in  $\mathcal{D}$  lie at a distance of at most  $\epsilon$  from  $\mathcal{H}$ . Let  $S$  be a set of randomly sampled  $k$  linearly independent points from  $\mathcal{D}$ , and  $S(\mathcal{H})$  be the projection of all data points onto  $\mathcal{H}$ . Let  $x$  be any data point from  $\mathcal{D}$  and  $x_{\mathcal{H}}$  be the projection of  $x$  onto  $\mathcal{H}$ . Let  $L$  be any line passing through  $x_{\mathcal{H}}$  and the convex hull of  $S(\mathcal{H})$ . Let  $p$  be the length of the segment in  $L$  corresponding to the two points of intersection of  $L$  with the convex hull of  $S(\mathcal{H})$ , and let  $q$  be the smallest distance along  $L$  from  $x_{\mathcal{H}}$  to the convex hull of  $S(\mathcal{H})$ . Let  $\mathcal{HS}$  be the hyperplane passing through  $S$ . Then, the projection of  $x$  onto  $\mathcal{HS}$  is at a distance of at most  $2 \cdot \epsilon \cdot (p + q)/q + 2 \cdot \epsilon$  from  $x$ .*

We note that Lemma 2.2 is different from Corollary 2.1 only in the last line, in which we prove the result with respect to  $x$  rather than  $x_{\mathcal{H}}$ , and modify the maximum distance by  $2 \cdot \epsilon$ . The truth of this corollary follows from the simple fact that the distance between  $x$  and  $x_{\mathcal{H}}$  is at most  $\epsilon$ .

We note that the results of Lemma 2.2 and Corollary 2.1 provide some intuition on the nature of the distance between a data point  $x$  and its projection onto the point sampled hyperplane  $\mathcal{HS}$ . The results show that if a hyperplane  $\mathcal{H}$  can be found such that all data points are at a distance of at most  $\epsilon$  from it, then any set of linearly independent points  $S$  will define a hyperplane  $\mathcal{HS}$  such that the distance between  $x$  and its projection onto  $\mathcal{HS}$  depends upon  $\epsilon$ . The exact distance depends upon the nature and size of the convex hull of the set  $S$  of sampled points. Thus, the results provide the intuition that as long as a hyperplane  $\mathcal{H}$  exists which defines the distribution of the points in database  $\mathcal{D}$ , the use of point sampled random projections is likely to yield accurate results.

**2.6 Pathological Cases: The Problem and a Solution** Our discussion in the previous section leads us to the following question: are there cases in which space sampling is better than point sampling? In this section, we will show that there are indeed such cases, though we will also show in the empirical section that they rarely arise in the context of real data sets.

Furthermore, we will show that even in such cases, a mixture of point and space sampling can provide almost comparable results to the best of the two methods.

In order to find pathological cases, we need to find a scenario in which the preconditions of Lemma 2.2 are not met. We note that the precondition of Lemma 2.2 assumes that a global hyperplane  $H$  of lower dimensionality is available along which the data points in  $\mathcal{D}$  can be approximately reduced. A counter example to this case is one in which the data has full global dimensionality, but the local behavior of the data is very different in different regions. We note that while local implicit dimensionality is often lower than global implicit dimensionality [2], the global implicit dimensionality is usually much lower than the full dimensionality. This is because the global reduction subspaces usually subsume the local subspaces. In such cases, point sampled global random projections continue to work quite well. However, in the unusual case that such a correlation does not exist and the data has full implicit dimensionality, we do not expect point sampled random projections to work very well. This is because in such cases, the local subspaces do not share global defining characteristics. Hence, there is no global direction of correlation. A point sampled hyperplane will typically contain some of the local directions of correlation, and completely miss the others. On the other hand, a space sampled random projection is likely to be less unbiased in representing the different directions.

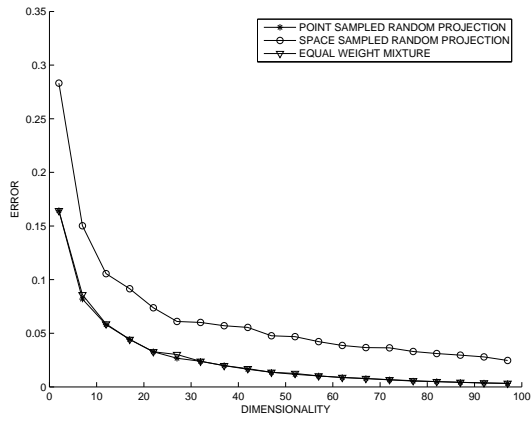
For example, consider the case when the  $d$ -dimensional data is partitioned into  $d$  different clusters, each of which is distributed along a 1-dimensional line. We also assume that the  $d$  lines are orthogonal to one another. In such a case, the data has full implicit dimensionality, but the local correlations are not similar to the (non-existent) directions of global correlation. We will show in the empirical section that in such cases, space sampled random projection may provide superior results. However, even in these cases, it is possible to obtain reasonable results by using an equally weighted mixture of space and point sampled random projections. The final representation is the best reduction among all the different point and space sampled projections. By doing so, we can obtain the best of the two methods by using twice the number of samples. For the same number of samples, the mixture provides results which are only slightly worse than the better of the two methods. In the experimental section, we will show that in both cases of pure point and space sampled random projections, the best sampling results are obtained within the first few iterations. Therefore, when a large number of samples are used, the difference in quality between

the better of the two (pure) methods and the mixture is small. It also retains the excellent average case behavior of point sampled random projections at the expense of a little reduction in quality. This provides excellent average-case behavior without compromising on the worst-case behavior in pathological instances.

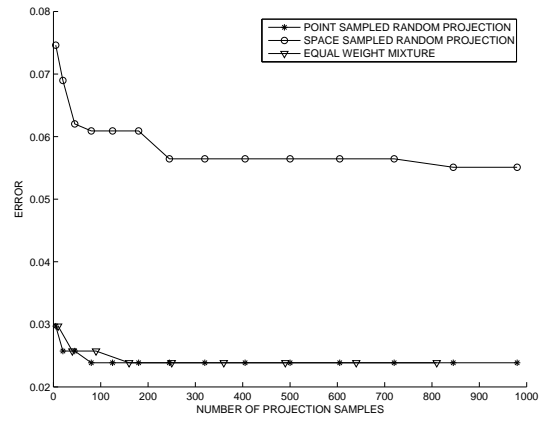
### 3 Experimental Results

In this section, we will analyze the effectiveness of point sampled and space sampled random projections. We will show that point sampled random projections are significantly more effective than point sampled random projections in a variety of circumstances. We will also show that a mixture of point and space sampled random projections provides results which are almost as good as the best of the two methods. We will show the results on both synthetic and real data sets. While the results on real data sets show that point sampled random projections can provide significantly more effective results in practical situations, the synthetic data sets can be used to illustrate the behavior of the underlying data on the effectiveness of point sampled random projections. The real data sets were obtained from the UCI machine learning repository. The aim of the testing process was to show that the point sampled random projection process was significantly more effective than the method of space sampled random projections. In general, the point sampled random projection process was not only able to achieve a superior qualitative reduction, but it was also able to do so in a far fewer number of projection samples. Furthermore, the mixture of the two methods provided almost comparable results to the best of the two methods, while retaining robustness in reduction quality even in pathological cases.

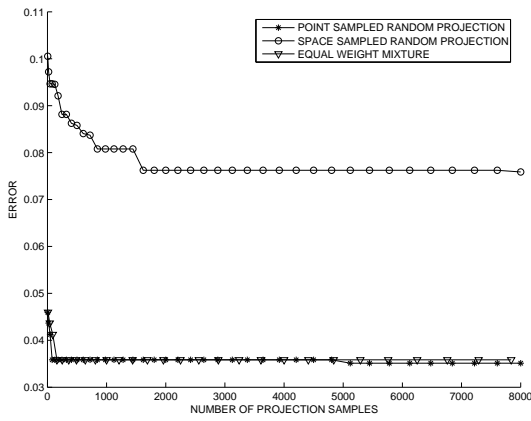
The first data set tested was the musk data set, which had 160 dimensions. In Figure 4(a) we have illustrated the average error behavior of space sampled projections, the point sampled random projections, and a mixture of the two methods. On the X-axis, we have illustrated the dimensionality of the projection, whereas on the Y-axis, we have illustrated the average error-metric for both methods as defined earlier. In each case, the value of *numsamp* was chosen to be 200. It is clear that the point sampled random projection process had a significantly lower error than the space sampled random projection process. Even for a projection dimensionality of 97, the space sampled random projection process continued to have 3 – 4% distance errors. Such errors can be significant for high dimensional applications. On the other hand, the point sampled random projection technique had a much smaller level of error across the board. We also note that the equally weighted mixture of point and space sampled random projections had



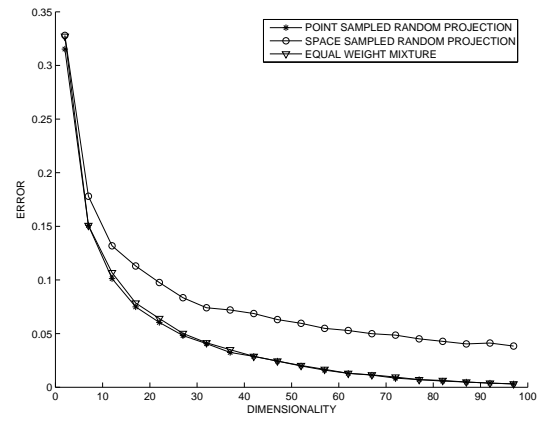
(a) Musk (w.r.t dimensionality)



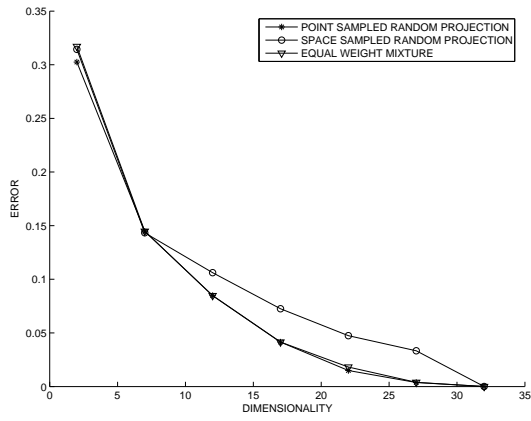
(a) Corel (w.r.t sample size)



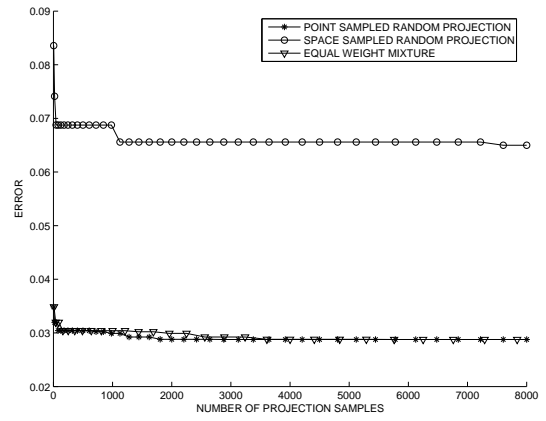
(b) Musk (w.r.t sample size)



(b) Arrhythmia (w.r.t dim.)



(c) Corel (w.r.t. dimensionality)

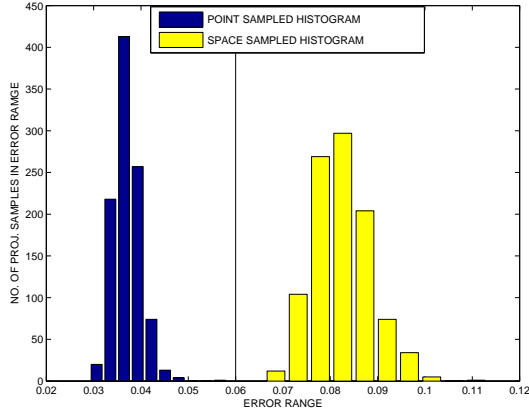


(c) Arrhythmia (w.r.t sample size)

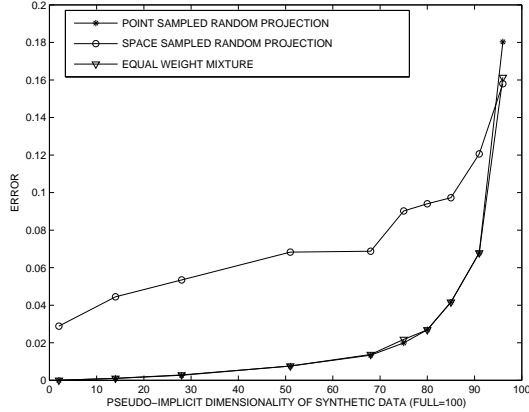
Figure 4:

Figure 5:

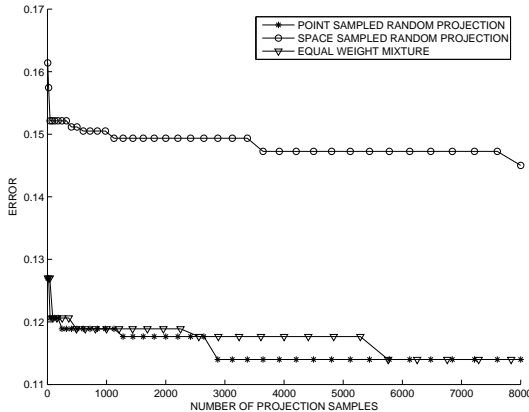




(a) Arrythmia (Error Distribution of 1000 samples)



(b) Implicit Dimensionality Effects on Error



(c) Slightly Correlated Data (w.r.t. samplesize)

Figure 6:

an error which was almost comparable to the point sampled random projection method in each case. This illustrates that the use of a mixture can provide similar average case behavior, while preserving the worst-case behavior in some pathological cases. Even further insight was obtained by examining the behavior of the reduction with an increasing number of projection samples  $numsamp$ , when the projection dimensionality was fixed at 20. The results are illustrated in Figure 4(b). *It is interesting to see that even when the value of  $numsamp$  is chosen at 7,500 in the space-sampled random projection process, the errors are significantly greater than those of point sampled or mixture based random projections with  $numsamp \leq 10$ .* We will see that this behavior is repeated for the other real data sets. Since the computational complexities of each sample in both methods are exactly the same, this translates to not only a qualitative edge, but also *orders of magnitude* improvements of efficiency with the use of point sampled random projections.

The second data set was the 32-dimensional corel-histogram data set containing 68040 records. We stripped out the first field in the data which only contained the line number. In Figure 4(c) we have illustrated the behavior of the different kinds of reduction on the data sets with varying projection dimensionality. We used  $numsamp = 100$  in this case. As in the previous cases, the point sampled random projection process is more effective than that of space sampled random projections for different projection dimensionalities. The mixture of point and space sampled random projections showed an effectiveness which almost overlapped with that of the effectiveness of point sampled random projections. In Figure 5(a), we have also illustrated the behavior of the two methods for different number of samples, when an 20-dimensional projection was used. As in the previous case, it turns out that the error of the point sampled method with 1 sample is much lower than the error of the space sampled method with even a thousand samples. This again illustrates the tremendous benefits of using point sampling for dimensionality reduction. Furthermore, the equally weighted mixture of point and space sampled random projection almost matched the effectiveness of the best of the two methods.

The results for the 279-dimensional arrythmia data set are illustrated in Figures 5(b) and 5(c) respectively. In the case of Figure 5(b), we have used  $numsamp = 200$ , whereas in the case of Figure 5(c), we have used a projection dimensionality of 40. As earlier, the results of Figure 5(c) show that even the use of 10,000 space sampled random projections cannot match the behavior of a small number of point sampled

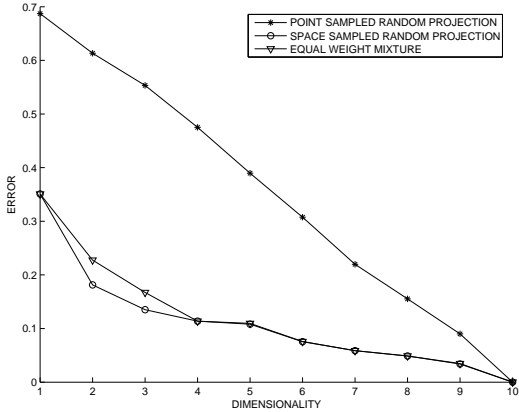


Figure 7: Pathological Data Set (w.r.t. Dim.)

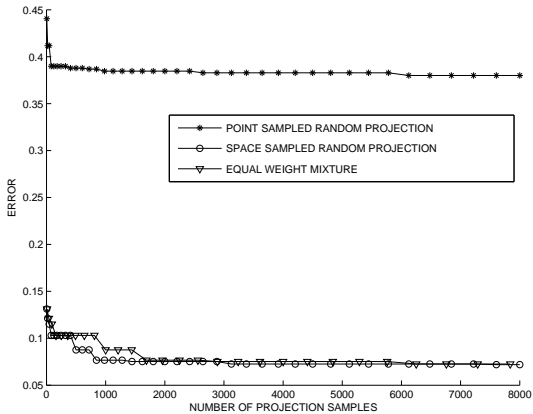


Figure 8: Pathological Data Set (w.r.t. samplesize)

projections. As in the previous cases, the behavior of the equally weighted point and space sampled random projections was almost the same as the effectiveness of point sampled random projections. In order to explore this point further, we picked 1000 samples of the 40-dimensional random projection in both cases, and plotted a histogram of the distribution of errors over these projection samples. The results are illustrated in Figure 6(a). The results show that even the *best* of 1000 space sampled random projections (right of vertical line in Figure 6(a) has a higher error than the *worst* of 1000 point sampled random projections (left of vertical line in Figure 6(a) Furthermore, the variation in the error over different space sampled random projections is higher than the variation in error over different point sampled projections. This point conclusively demonstrates the relative robustness of the point sampled random projection process.

**3.1 Some Interesting Cases with Varying Implicit Dimensionality** In this section, we will discuss the relative behavior of point sampled and space sampled random projections with varying implicit dimensionality. This section will illustrate the fact that the advantages of the point sampled random projection process arise from the fact that real data often has lower implicit dimensionality than the full dimensionality. When the data is uniformly distributed, the point sampled random projection process has no advantages over space sampled random projection, and therefore both methods are expected to perform similarly. In fact, the space sampled random projection process has a slight advantage, since it has better flexibility in picking the projections.

In order to test the effects of implicit dimensionality, we generated a series of data sets with varying levels of correlation in the data. In order to generate such a series of data sets, we first generated an axis system with random orientation. This axis system represents the directions of correlation. The level of correlation can be varied by changing the variances along the different axis directions. Note that in a data set with low implicit dimensionality, most of the variance is concentrated along a few of the axis-directions which are also referred to as principal components. Therefore, in order to create skew in the variance along the different principal components, we determined the standard-deviation along the  $i$ th axis direction using the Zipf- distribution  $1/i^\theta$ . Therefore, the implicit dimensionality can be varied by changing the value of  $\theta$ . A choice of  $\theta = 0$  corresponds to a uniform distribution, whereas the implicit dimensionality rapidly reduces with increasing values of  $\theta$ . However, we first need to create a crisp definition

of the implicit dimensionality for experimental testing purposes. This term is generally loosely used to refer to the number of significant principal components in the data, but we are not aware of a more concrete definition. Therefore, for experimental testing purposes, we defined the *pseudo-implicit* dimensionality of a data set as the number of axis directions in the optimal principal component transform [9] which retains 98% of the variance in the data. This turns out to be a fairly intuitive definition for testing purposes.

In practice, a choice of  $\theta = 3$  can concentrate all the variance in only 2 or 3 axis directions. For example, when we generated a series of data sets with  $N = 1000$  points in  $d = 100$  dimensions, the implicit dimensionalities of data sets with choices of  $\theta = 0, 0.8, 1.0, 1.12, 1.22, 1.32, 1.5, 1.75, 2,$  and  $3$  correspond to data sets with implicit dimensionalities 96, 91, 85, 80, 75, 68, 51, 28, 14 and 2 respectively. We used this series of data sets in order to test the effectiveness of point sampled and space sampled random projections. In Figure 6(b), we have illustrated the error behavior of this series of data sets with varying implicit dimensionality, when a 10-dimensional projection is picked from the transformed data with  $numsamp = 200$  samples. It is clear that the point sampled random projection process has a great advantage over the space sampled random projection process when the implicit dimensionality is very low compared to the full dimensionality. The most interesting special case is that when  $\theta = 0$ . This corresponds to the uniformly distributed data set in which the point sampled random projection process has no special advantage over space sampled random projections. We note that this is a pathological case which is never encountered in real data sets. This corresponds to the rightmost point in Figure 6(b) with a pseudo-implicit dimensionality<sup>1</sup> of 96. In this case, the error behavior of both methods are almost the same. In fact, the space sampled random projection process is slightly better, which is possibly because of the greater flexibility of picking the projection during repeated sampling. The other very interesting cases are the extreme ones in which the data sets have extremely low implicit dimensionality compared to the full dimensionality. We note that since we are picking a projection with a dimensionality of 10, an effective reduction approach should have negligible errors for data sets with implicit dimensionalities which are less than 10. In order to examine what happens in this case, we look at the leftmost

point in Figure 6(b). In this case, the 100-dimensional data set is (almost) embedded on a plane with only 2-dimensions. The interesting result is that even a 10-dimensional space-sampled random projection continues to have greater than 3% distance errors. Therefore, even a choice of projection dimensionality significantly greater than the pseudo-implicit dimensionality is not able to reduce the error level to a negligible level for the space sampled random projections. On the other hand, the point sampled random projection process continues to have very little error for data sets of implicit dimensionality which are less than 15. This shows that the space sampled random projection process often misses obvious reductions in the data, because it is blind to the underlying distribution. The results also show that this can be leveraged by the point sampled random projection process. The results of Figure 6(b) show that even for data sets with slight correlations (implicit dimensionality greater than 85), the point sampled random projection process has significantly lower error. In Figure 6(c), we have illustrated the variation in error-behavior of the 10-dimensional random projection (with different values of  $numsamp$ ) for an instantiation of the 100-dimensional synthetic data set with pseudo-implicit dimensionality of 94. The results in Figure 6(c) show that the point sampled random projection process is significantly more effective even for this relatively uncorrelated data set. Furthermore, the quality of the point sampled random projection with the use of 5 samples is significantly better than the space sampled projection process with a choice of even 10,000 samples. This is consistent with our observations on real data sets in which point sampled random projections process are significantly superior to space sampled projections.

We also tested our algorithm on the pathological case discussed in Section 2.6. In this case we generated a 10-dimensional instantiation of such a data set in the unit cube with 1000 data points. In Figure 7, we have illustrated the error of the projection for different values of the projection dimensionality when we used  $numsamp = 200$ . Since this data set has full implicit dimensionality but misleading local variations in the data behavior it resulted in the point sampling approach to pick projections which were sometimes orthogonal to many of the true directions of local correlation. As a result, some of the points had large errors. We also note that the data was specifically generated in a particular way so that the different local correlations were orthogonal to one another. This particular pathological structure resulted in the point sampling not being as effective as space sampling. *However, even in this pathological case, the mixture method continued to be almost as effective as the best of*

<sup>1</sup>Note that the pseudo-implicit dimensionality is always likely to be less than 100 even for the uniform distribution, when the data set is of finite size. Therefore, the pseudo-implicit dimensionality of the uniformly distributed data set is 96, and not 100.

*the two methods.* We have also illustrated the behavior of the methods for different numbers of projection samples in Figure 8 when a projection dimensionality of 5 was used. These results also show that while space sampling was better in this case, the mixture method continued to provide very robust results. These results show that even in the contrived cases in which space sampling is superior, the mixture method continues to provide robust results.

#### 4 Conclusions and Summary

In this paper, we presented methods for using point sampled random projections for dimensionality reduction. Our results show that point sampled random projections can perform the dimensionality reduction effectively when the underlying data has low implicit dimensionality compared to the full set of dimensions. We also provide theoretical results which show that point sampled random projections are very effective at preserving the underlying variance of the data. The point sampled dimensionality reduction is not only more accurate, but can significantly improve the efficiency of the reduction process by requiring a number of projections which are orders of magnitude fewer. In addition, the point sampled random projection process can achieve qualitative results which cannot be achieved by a practical number of iterations in the space sampled random projection process. We also present empirical results which show the effects of the underlying implicit dimensionality on the relative effectiveness of point sampled and space sampled random projections. The results show that the relative effectiveness of the point sampled random projection process is particularly high when the implicit dimensionality of the data is low compared to the full dimensionality. Even in pathological cases, in which the space sampling method has an advantage, we discussed the robustness of using a mixture of point and space sampled random projections for dimensionality reduction. This mixture typically provides results which are competitive with the best of the two methods across a wide spectrum of data sets.

#### References

- [1] D. Achlioptas. *Database-friendly Random Projections*. ACM PODS Conference, 2001.
- [2] C. C. Aggarwal. *Hierarchical Subspace Sampling: A Unified Approach to High Dimensional Data Reduction, Selectivity Estimation, and Nearest Neighbor Search*. ACM SIGMOD Conference, 2002.
- [3] E. Bingham, and H. Mannila. *Random Projection in Dimensionality Reduction: Applications to Image and Text Data*. ACM KDD Conference Proceedings, 2001.
- [4] S. Dasgupta, and A. Gupta. *An Elementary Proof of the Johnson-Lindenstrauss Lemma*. Technical Report, International Computer Science Institute, California, Berkeley, 1999.
- [5] C. Faloutsos, and K.-I. Lin. *FastMap: A Fast Algorithm for Indexing, Data-Mining and Visualization of Traditional and Multimedia Datasets*. ACM SIGMOD Conference, 1995.
- [6] A. Hinneburg, C. C. Aggarwal, and D. A. Keim. *What is the nearest neighbor in high dimensional space?* Proceedings of the VLDB Conference, 2000.
- [7] P. Indyk, and R. Motwani. *Approximate Nearest Neighbors: Towards Removing the Curse of Dimensionality*. ACM STOC Proceedings, pages 604-613, 1998.
- [8] W. Johnson, and J. Lindenstrauss. *Extensions of Lipschitz mapping into a Hilbert space*. Conference in modern analysis and probability, pages 189-206, American Math Society, 1984.
- [9] I. T. Jolliffe. *Principal Component Analysis*, Springer-Verlag, New York, 1986.
- [10] C. H. Papadimitriou, P. Raghavan, H. Tamaki, and S. Vempala. *Latent Semantic Indexing: A Probabilistic Analysis*. ACM PODS Conference, 1998.
- [11] K. V. Ravi Kanth, D. Agrawal, A. Singh. *Dimensionality Reduction for Similarity Searching in Dynamic Databases*. SIGMOD Conference, 1998.