# Random projection experiments with chemometric data

**Kurt Varmuza[a]*, Peter Filzmoser[b] and Bettina Liebmann[a]**

* Correspondence to: K. Varmuza
Laboratory for Chemometrics, Institute of Chemical Engineering,
Vienna University of Technology, Getreidemarkt 9/166, A-1060 Vienna, Austria
E-mail: kvarmuza@email.tuwien.ac.at

[a] K. Varmuza, B. Liebmann
Laboratory for Chemometrics, Institute of Chemical Engineering,
Vienna University of Technology, Getreidemarkt 9/166, A-1060 Vienna, Austria
E-mails: kvarmuza@email.tuwien.ac.at; Bettina.Liebmann@tuwien.ac.at

[b] P. Filzmoser
Institute of Statistics and Probability Theory, Vienna University of Technology,
Wiedner Hauptstrasse 8-10, A-1040 Vienna, Austria
E-mail: P.Filzmoser@tuwien.ac.at

**Abstract**

Random projection (RP) is a linear method for the projection of high-dimensional data onto a lower dimensional space. RP uses projection vectors (loading vectors) that consist of random numbers taken from a symmetric distribution with zero mean, and many successful applications have been reported for high dimensional data sets. The basic ideas of RP are presented, and tested with artificial data, data from chemoinformatics and from chemometrics. RP's potential in dimensionality reduction is investigated by a subsequent cluster analysis, classification or calibration, and is compared to PCA as a reference method. RP allowed drastic reduction in data size and computing time, while preserving the performance quality. Successful applications are shown in structure similarity searches (53,478 chemical structures characterized by 1233 binary substructure descriptors) and in classification of mutagenicity (6506 chemical structures characterized by 1455 molecular descriptors). Only in calibration tasks with low dimensional data as in many chemical applications, RP showed limited performance. For special applications in chemometrics with very large data sets and/or severe restrictions for hardware and software resources, RP is a promising method.

**Keywords:** dimensionality reduction; PCA; similarity of chemical structures; KNN classification; PLS regression

## 1. Introduction

Many successful methods in chemometrics for the analysis of multivariate data $\boldsymbol{X}$ ($n$ samples/objects/rows × $m$ variables/features/columns) are based on linear latent variables also called the components. A component is defined by a vector $\boldsymbol{b}$ of unit length (loading vector) corresponding to a direction in the $m$-dimensional variable space. The values (scores), $u_i$, of a latent variable for the objects $i$ are obtained by a projection of the $n$ data points on an axis defined by the vector $\boldsymbol{b}$. The score vector for all objects is $\boldsymbol{u} = \boldsymbol{X} \bullet \boldsymbol{b}$. In some methods not a single loading vector is used but $1 \leq a < m$ of vectors $\boldsymbol{b}_g$ ($g = 1, ..., a$) forming a projection matrix $\boldsymbol{B}(m \times a)$, which is orthonormal if all vectors are pairwise orthogonal and have unit length. Because the number of components, $a$, is usually much smaller than the number of variables, $m$, the projection $\boldsymbol{X} \bullet \boldsymbol{B}$ results in a dimensionality reduction yielding a new matrix $\boldsymbol{U}(n \times a)$. Depending on the aim of data analysis and the used method, the loading vectors are derived from $\boldsymbol{X}$ (and a vector $\boldsymbol{y}$ containing object properties) by defining a desired specific mathematical property for the scores. For instance, in principal component analysis (PCA) the scores possess maximum variance; in linear discriminant analysis (LDA) for two classes of objects the scores of the discriminant variable provide maximum separation of the classes; in ordinary least-squares regression (OLS) the scores of the resulting component have maximum correlation coefficient with an object property $y$; and in partial-least squares (PLS) regression, usually several components with maximum covariance between the scores and an object property $y$ are used for regression (remember that the Pearson correlation coefficient and the covariance become identical for variance scaled data). The problem specific definition of an appropriate objective for the scores is essential for all these methods.

A very different strategy for defining latent variables is pursued in "random projection" by using projection vectors with random directions [1]. That means, the data structures of $\boldsymbol{X}$ and $\boldsymbol{y}$ have no influence on the used projection/loading matrix $\boldsymbol{B}$. However, the aim of the projection is still a - usually considerable - dimensionality reduction. Remarkably, in high-dimensional space two vectors with appropriately created random vector components have a high probability for being almost orthogonal [2,3]. Random projection (RP) is a simple and fast technique, and may be an alternative to classical methods for dimensionality reduction for data sets with a very large number of variables and objects, or for specific applications with very limited computer resources such as space experiments. Another potential application exists if the data structure corrupts the defined mathematical property of a classical component score because the general data structure heavily influences the directions of the projection vectors and small - but important - groups of objects are not considered adequately. In such cases a projection to randomly selected axes may have advantages.

Several successful applications of RP are reported, such as for information retrieval in text documents [4-6], for the recognition of handwritten text [7], for (hyper-spectral) image compression [8,9] or face recognition [10], and indexing of audio documents [11]. Theses results indicate that RP preserves distances and has a performance similar to e.g. PCA while being much faster [4]. Dimensionality reduction of text or audio data by RP speeds-up subsequent document categorization and classification procedures such as Latent Semantic Indexing (LSI) [6], also combined with Self-Organizing Maps (SOM) [11].

RP was also used for clustering of artificial data [12], for instance mixtures of high dimensional Gaussians [7] while retaining data separation. Ensembles of RP runs instead of

single runs yield more stable results in clustering of high-dimensional artificial data [13] as well as with gene expression data in a fuzzy ensemble approach [14].

RP was successfully used for nearest neighbor searches in high dimensions [15,16]. Mass spectra from protein samples from cancer cells and from reference materials have been represented by vectors containing 14,936 components and RP was successfully applied for classification [17]. Several reports deal with RP applied to bioinformatics [18], neural networks [19-21], to learning theory [22] or to spatio-temporal data [23].

In this study we apply both random projection and PCA projection for dimensionality reduction of the original data, and then compare their performances in searches for similar objects, KNN classification and PLS regression. PCA is applied either to all objects or to a small random sample of the objects in order to reduce the computation time to a level comparable with RP. The data sets used are (1) artificial data with Gaussian shaped clusters; (2) a set of 53,476 organic chemical structures characterized by 1233 binary substructure descriptors; (3) a set of 6506 organic compounds with Ames test results (active or not active) with each structure characterized by 1455 molecular descriptors; and (4) two data sets of smaller size for multivariate calibration by PLS regression models (166 to 846 objects, 235 to 529 variables).

## 2. Methods

### 2.1. Random projection

Random projection (RP) is a linear method for reducing the dimensionality of data. Given a data matrix $\boldsymbol{X}$ (with $m$ variables), the data points are projected from the originally high-dimensional space ($m$-dimensional) onto a lower-dimensional subspace (with $a$ projection scores) formed by a set of random vectors, i.e. loading vectors with random numbers as vector components.

The motivation of the functionality of RP is given by a lemma by Johnson and Lindenstrauss [24]. It states that distances between points are approximately preserved, if they are projected randomly onto a lower-dimensional subspace. The mathematical formulation and proofs of this lemma are given in [8,25,26].

Different approaches have been suggested for the generation of random projection matrices. In general, each component $b_j$ ($j$ = 1, 2, 3, ..., $m$) of a projection vector $\boldsymbol{b}$ is randomly taken from a distribution which has to be symmetrical to a mean of zero. Some authors [4,13] use a standard normal distribution, N(0,1); others prefer a uniform distribution U[-1, +1] with values between -1 and +1. However, even much simpler approaches have been suggested, for instance only using fixed values with fixed probabilities, e.g. {-1, +1} or {-1, 0, +1} with equal probabilities, or e.g. $\{-(3)^{0.5}, 0, +(3)^{0.5}\}$ with the probabilities 1/6, 2/3, and 1/6, respectively [2,5]. Regardless of the used distribution the projection vectors have to be normalized to unit length in order to make the projection independent from the scale of the projection vectors.

The fundamental, yet surprising property of random projection vectors is that in high-dimensional space they might be sufficiently close to orthogonal. Furthermore, a much larger number of almost orthogonal than orthogonal directions exist in high-dimensions [3]. To measure the similarity, the cosine of the angle, $\alpha$, between two vectors $\boldsymbol{b}_g$ and $\boldsymbol{b}_h$ is commonly used; it is defined as the normalized inner product

$$\cos \alpha = (\boldsymbol{b}_g^{\mathrm{T}} \bullet \boldsymbol{b}_h) / (\|\boldsymbol{b}_g\| \bullet \|\boldsymbol{b}_h\|) \qquad (1)$$

The same equation holds for the Pearson correlation coefficient of mean-centered data $b_{jg}$ versus $b_{jh}$ ($j = 1 \ldots m$). Note that the randomly chosen vector components are from a symmetric distribution with mean zero, and of course the correlation coefficient of such random data is mostly near zero, that means $\alpha$ is near 90°. Figure 1 shows simulation results for 10,000 pairs of random vectors for $m$ = 100, 500, and 2500 variables; the random number distributions used for the vector components were N(0, 1), U[-1, +1], and two fixed values {-1, +1} with equal probabilities. These three distributions yield very similar results for the density distributions of cos $\alpha$. The means of the density distributions are near zero and the widths become narrower with increasing dimensionality. We define (arbitrarily) two vectors as being "almost orthogonal" if the deviation from $\alpha$ = 90° is within $\pm$3° ($\pm$0.05234 for cos $\alpha$). For all three random number distributions, 99.1 to 99.2% of the randomly generated vector pairs are almost orthogonal in 2500 dimensional space; for 500 dimensions still 75.8 to 77.1%, but for 100 dimensions only about 40%. In this study, all random projection vectors contain random vector components chosen from the uniform distribution U[-1, 1].

More specifically, it has been shown [27] that the distribution of cos $\alpha$ can be approximated by a normal distribution with a mean of zero and a standard deviation of $(1/m)^{0.5}$. This approximation improves with increasing dimensionality. In Figure 1 the normal distributions $N(0, (1/m)^{0.5})$ are drawn in gray; they are not distinguishable from the corresponding empirical distributions.

In many applications of RP, particularly when dealing with more than about 500 variables, a straightforward random generation of the random projection vectors is reasonable because most pairs of projection vectors are almost orthogonal. However, strictly orthogonal axes are necessary, if the variance preserved by the projection is evaluated. A further argument favoring orthogonality is that a new orthogonal direction will capture potentially new information, whereas a non-orthogonal direction "repeats" information that is already included in previous directions.

Orthogonalization requires additional computational effort that may contradict an application of RP in some cases. In this study all random projection vectors are orthogonal to each other. They have been generated one after the other by limiting the space to the orthogonal subspace of the previously chosen random directions (orthogonalization "on the fly"). For a data set with 1000 objects and 1000 variables the computation time for 100 random vectors is about 0.04 s without orthogonalization, and about 7 s with orthogonalization on the fly. For 1000 random vectors the time is about 0.3 s without, and 700 s with orthogonalization. A further possibility to obtain orthogonal random vectors is to perform the orthogonalization after generating all random directions. However, the resulting directions are exactly the same as for orthogonalization "on the fly", and also the computation time is practically the same. The

pseudo code for orthogonalization "on the fly" is as follows, according to the Gram-Schmidt procedure, well-known in matrix algebra [28,29].

(1) First direction $\boldsymbol{b}_1$ is defined by a vector with *m* randomly chosen vector components, with $\|\boldsymbol{b}_1\| = 1$

(2) *j*-th direction, for $j = 2, \ldots, a$:

(a) select $\tilde{\boldsymbol{b}}_j$ randomly (vector with *m* randomly chosen vector components)

(b) $\boldsymbol{b}_j^* = \tilde{\boldsymbol{b}}_j - \left( \sum_{i=1}^{j-1} \boldsymbol{b}_i \boldsymbol{b}_i^T \right) \tilde{\boldsymbol{b}}_j$

(c) $\boldsymbol{b}_j = \frac{\boldsymbol{b}_j^*}{\|\boldsymbol{b}_j^*\|}$

The resulting random directions $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_a$ have norm 1 and are orthogonal to each other, because

$$\boldsymbol{b}_k^T \boldsymbol{b}_j^* = \boldsymbol{b}_k^T \tilde{\boldsymbol{b}}_j - \left( \sum_{i=1}^{j-1} \boldsymbol{b}_k^T \boldsymbol{b}_i \boldsymbol{b}_i^T \right) \tilde{\boldsymbol{b}}_j = \boldsymbol{b}_k^T \tilde{\boldsymbol{b}}_j - \boldsymbol{b}_k^T \tilde{\boldsymbol{b}}_j = 0$$

for any $k < j$. Orthogonalization of the random vectors at the end works very similar; however, step (2a) can be omitted since the directions are already available. Note that both orthogonalization methods are independent from the data, they just depend on the data dimensions.

### 2.2. PCA

Principal component analysis (PCA) was chosen as reference method to compare with RP. Because only a few principal components are required for this purpose, we preferably used the NIPALS algorithm available as function "nipals" in the R package "chemometrics" [30].

PCA has been applied to data with all objects for obtaining directions with maximum variances of the scores. In addition, a random subset of the objects (typically 1-10 %) was selected to reduce the computation time of PCA to a level comparable with RP including orthogonalization.

### 2.3. KNN classification

For k-nearest neighbor (KNN) classification the function “knn.cv” available in the R package “cluster” was applied. The variables were scaled to unit variance of all variables and the Euclidean distance served as an inverse similarity measure. The number of neighbors, *k*, was varied in five steps between 1 and 101. The classification performance was estimated by leave-one-out cross validation and the average predictive ability, $P_{MEAN}$, defined as the arithmetic mean of the predictive abilities (fraction correctly classified) for each class of objects [30].

### 2.4. PLS regression

For modeling an object property *y* from *x*-variables or projection scores, partial least-squares (PLS) regression was used in the version SIMPLS as implemented in the R package "pls" [31]. The performance of PLS models for test set objects was estimated by repeated double cross validation (rdCV) [32] available in the R package "chemometrics" [30]. The performance measure used is SEP (standard error of prediction), equivalent to the standard deviation of the prediction errors for test set objects [30].

### 2.5. Software

All computations were performed with software written for the statistical programming environment R [33]. Several packages available in R have been utilized; they are mentioned in the related Sections before.

## 3. Results

### 3.1. Artificial data

With respect to a criterion like the proportion of the explained total variance, random projection (RP) is expected to perform worse than PCA. This is demonstrated by a simulation study. We generate an *artificial data set 1*, consisting of matrix $\boldsymbol{X}$ with $n = 5000$ objects and $m = 500$ variables, using a so called Gaussian, i.e. a multivariate normal distribution, here with mean zero and a diagonal covariance matrix with diagonal elements $1/j^2$, $j = 1,\ldots, m$. Thus the total variance is $\sum_{j=1}^{m} 1/j^2$. The data are randomly rotated with an orthonormal matrix, resulting in a centered data matrix $\tilde{\boldsymbol{X}}$ with 500 variables.

For dimensionality reduction we then apply three methods: (a) RP with orthogonal projection vectors, (b) PCA using all objects, and (c) PCA using a random sample of 1 % of the objects. For each method we extract sequentially up to $a_{MAX} = 50$ projection directions $\boldsymbol{b}_g$, and measure the computation time. The resulting centered score matrices are $\boldsymbol{U}_a = \tilde{\boldsymbol{X}}\boldsymbol{B}_a$, with $\boldsymbol{B}$ containing

the projection vectors $\boldsymbol{b}_g$, for $g = 1, ..., a$, and $1 \leq a \leq a_{MAX}$. The proportion of explained variance for a certain number of components is thus the sum of the variances of the columns of $\boldsymbol{U}_a$ divided by the total variance. We also measure the distortion from the original space to the lower dimensional space, being defined as

$$\delta_a = \left\| \tilde{\boldsymbol{X}}\tilde{\boldsymbol{X}}^T - \boldsymbol{U}_a\boldsymbol{U}_a^T \right\| \quad (2)$$

for a considered number $a$ of components ($\|.\|$ denotes the Euclidean norm). Let $\tilde{\boldsymbol{x}}_i$ and $\tilde{\boldsymbol{x}}_l$ denote the $i$-th and $l$-th row of $\tilde{\boldsymbol{X}}$, respectively. Then the element $(i,l)$ of $\tilde{\boldsymbol{X}}\tilde{\boldsymbol{X}}^T$ is the inner product $\tilde{\boldsymbol{x}}_i^T\tilde{\boldsymbol{x}}_l$ which is, according to Equation (1), the cosine of the angles between the vectors defined by the two objects through zero, multiplied with their lengths. The distortion thus measures how well on average the angle between all pairs of objects and their distances are preserved in the reduced space.

The simulation was repeated 50 times, and the averages of the results are shown in Figure 2. The proportion of explained variance as well as the average distortion is quite comparable for PCA using all data and PCA with a 1% subset. However, the computation time for PCA with a 1% subset is about 100 times shorter than PCA with all data. In fact, the computation time for PCA on the subset is comparable to RP. For RP, however, the proportion of explained variance and the average distortion are very poor compared to PCA.

PCA is the optimal method with respect to variance and distance preservation, and RP is unable to compete with. However, for classification or calibration tasks it might be other

criteria than variance that are important. Before working with real data problems, a second simulation study is consulted, which compares RP and PCA for their abilities in dimensionality reduction in a classification problem. Accordingly, we generate an *artificial data set 2* with two Gaussians in 100 dimensions, one set with 800 objects, and the other with 200 objects. The covariance matrices of both sets are the same, a diagonal matrix with values 100 for the first 10 diagonal elements, and values uniformly taken from the interval [0, 1] for the remaining diagonal elements. Thus, the essential information is in 10 dimensions. The first set is centered at zero, and the second at a point with a value 0 for the first 10 coordinates and a value of 20 for the other coordinates. The centers of the two object groups are separated only in the variables with low variance, and the groups have a severe overlap.

The joined data are randomly rotated using a random orthonormal rotation matrix, and centered afterwards. For identifying the two data groups we use model-based clustering with two clusters [34]. Figure 3 shows the average of the predictive abilities, $P_{MEAN}$ (Section 2.3) for 50 different simulations of the original data, for RP, PCA, and for PCA with a 1% random subset, for up to 10 projection directions. The average predictive ability for the original data is 0.53; this low value is caused by the fact that the prominent variables do not contain information about class separation. With 10 projection directions RP is considerably better with an averaged predictive ability of 0.97 than PCA. PCA focuses on a good representation of variables with high variances and thus ignores the separating variables; randomly chosen projection directions by RP, however, contain more class information than principal components.

### 3.2. Similarity of chemical structures

The crucial prerequisite for any dimensionality reduction method is the preservation of the similarity of objects in their low dimensional representation. With the Euclidean distance as a measure of dissimilarity, the optimum method is PCA, having projection axes that preserve maximum variance, and therefore also the relative distances between the objects.
However, it has been shown for artificial data (Section 3.1, [7,13]) as well as for real data [4,5,7,10,13] that random projection (RP) is capable to keep the general cluster structure. We present an example from chemoinformatics [35] that deals with the similarity of chemical structures.

Chemical structures of organic compounds are often characterized by binary substructure descriptors, collected in vectors $\boldsymbol{x}_i$ for structures $i$ = 1 to $n$. Each vector component $x_{ij}$ corresponds to a substructure $j$ ($j$ = 1 to $m$), and is either 1 or 0 for substructure $j$ being either present or not in the molecular structure $i$. Typically, $m > 1000$ substructures are used, and the similarity of two structures A and B is prevalently calculated by the Tanimoto index [36], $t_{AB}$, also called Jaccard similarity coefficient [37].

$$t_{AB} = \Sigma(\mathrm{AND}(x_{Aj}, x_{Bj}) / \Sigma(\mathrm{OR}(x_{Aj}, x_{Bj}) \qquad \text{sum for } j = 1 \text{ to } m \qquad (3)$$

$\Sigma(\mathrm{AND}(x_{Aj}, x_{Bj})$ is the number of variables (descriptors) with a "1" in both vectors (logical AND), and $\Sigma(\mathrm{OR}(x_{Aj}, x_{Bj})$ is the number of variables with a "1" in at least one of the vectors (logical OR). The Tanimoto index ranges between 0 and 1; the value 1 is reached if all descriptors are pairwise equal; however, then the structures are not necessarily identical. The

Tanimoto index considers the high diversity of chemical structures, which requires a large number of descriptors, of which many are zero for a given structure. Thus the Tanimoto distance is successfully used in searches for similar chemical structures in a database, but also for characterizing the structural diversity [38,39]. We applied RP to a binary matrix $\boldsymbol{X}$ (for $n$ structures and $m$ binary variables) and compared the structural similarity in the original space (measured by the Tanimoto index) with the similarity of vectors in low dimensional spaces (measured by the Euclidean distance). For comparison, projections to low dimensional spaces are performed using PCA with all objects, and with a small random sample of the objects. The size of the small random sample was chosen such that the computation time for PCA was similar to RP.

The $x$-data were generated from $n$ = 53,478 randomly selected chemical structures (about a half of the structures in the mass spectral database NIST [40]). 1365 binary molecular descriptors [41], of which 132 had the constant value zero for all structures, were calculated by Software SubMat [42], finally resulting in $m$ = 1233 binary variables.

A set of 1000 query structures were selected randomly and for each query the 20 most similar structures (according to highest Tanimoto indices) among the other structures were searched. This structure similarity search resulted in 1000 hitlists, each of which contained the database structures most similar to the query structure, sorted by decreasing similarity (Tanimoto index).

Dimensionality reduction of the 1233-dimensional space was performed by three methods: (a) RP with orthogonal projection vectors; (b) PCA using all objects; (c) PCA using a random

sample of 2% of the structures. The number of dimensions in the low-dimensional space was 10, 30, and 100, respectively. Hitlists for the 1000 query structures were based on the Euclidean distance (calculated from projection scores) as similarity measure. A projection is considered to be successful if the hitlist obtained from binary substructure descriptors and the hitlist obtained from projection scores have a large "number of common hits", NCH. For this comparison we considered three sizes of the hitlist, the first 5, first 10 and first 20 hits, corresponding to different sizes of neighborhoods around the query structure (however, not considering the absolute values of the distances). Each of the 1000 queries yields a NCH, and the distribution of these numbers are represented by a box plot in Figure 4. For instance, with 10 hits and 30 RP projection directions, NCH is between 0 and 10, with a median of 6, the first quartile at 4, and the third quartile at 8. That means, in 50% of the queries the first 10 neighbors found in the 1233-dimensional binary space and the first 10 neighbors found in the 30-dimensional RP-space have six ore more structures in common.

PCA scores for the low-dimensional space (either calculated from all objects or a small random sample of the objects) yield very similar results as RP but narrower distributions of NCH. The boxplots in Figure 4 indicate an increase of NCH with increasing dimensionality of the projection space. For 30 projection directions the median of NCH is about 60% of the length of the hitlist, and reaches 75 to 87% for 100 directions. For 30 and 100 projection dimensions, RP gives similar results as PCA with a small random sample, but has for 10 dimensions a lower performance than the compared methods.

PCA based on only 2% of the data leads to very similar results as PCA based on all data. Note that PCA for the reduced data is comparable in computation time of RP with

orthogonalization (about 3 minutes to find 100 directions). For RP without orthogonalization, the directions are found in a fraction of a second, and the results for NCH are practically the same as for the orthogonalized RP method. This is plausible because the random directions are almost orthogonal in the space of dimension 1233. The gain in computational speed is, however, not so relevant since computing the similarities between the structures again takes several minutes.

A hitlist obtained from projection scores (RP or PCA) contains most of the hits that are in a hitlist obtained from binary descriptors. Next we investigate the distribution of the structural similarities between hits and queries, using the Tanimoto index as a measure for similarity. The search strategy with binary substructure descriptors results in hits with maximum possible Tanomito indices. In Figure 5 the distribution of the Tanimoto indices for the first 10 hits times 1000 queries is shown as solid line. The dashed lines are the distributions of Tanimoto indices for hitlists obtained from RP with 30 and 100 dimensions, respectively. Using only 30 RP dimensions gives hits with rather small Tanimoto indices (poor structural similarity); actually about 5% of these hits have a smaller Tanimoto index than the smallest one obtained from the original substructure descriptors. With 100 RP dimensions both distributions become very similar and only a very few hits obtained from projection data have a poor structure similarity with the query.

Concluding the results of this example, a reduction of the 1233-dimensional space with binary variables to a space with 30 to 100 real value variables seems acceptable for structure similarity searches. The similarity hitlists obtained from the low-dimensional space contain about 3/4 of the structures present in the hitlists obtained from the high-dimensional space.

Advantages of RP are lowered by the fact that binary numbers require small storage capabilities (if packed). Potential applications of RP, however, depend on the number of structures in the database and the available computer resources. Computing time for a projection of the 53,478 chemical structures encoded by 1233 binary descriptors to 100 dimensions was 180 s for RP, 200 s for PCA with 2% of the objects, and 12,000 s for PCA with all objects. Computation of the hitlists (first 20 hits) for 1000 queries required 13,000 s for 1233 binary substructure descriptors (Tanimoto index), 700 s for 30 projection scores (Euclidean distance), and 1000 s for 100 projection scores (Euclidean distance).

### 3.3. Classification of activity

Random projection (RP) has often been used for dimensionality reduction in classification problems with large data sets. Data used in our example are from $n$ = 6506 chemical structures with the mutagenicity given by Ames tests [43]. A set of $n_1$ = 3502 compounds are active, and $n_2$ = 3004 are inactive. Each structure has been characterized by $m$ = 1455 molecular descriptors calculated by software Dragon [44] from approximated 3D-structures and all H-Atoms explicitly given (calculated by software Corina [45]). For this binary classification problem we applied KNN classification with a leave-one-out cross validation for estimating the classification performance. The Euclidean distance was used with variance scaled variables; the number of neighbors, $k$, was 1, 3, 5, 11, 31, and 101 with a simple majority voting.

KNN classifications have been performed with (a) the original 1455-dimensional data; (b) scores from RP; (c) PCA scores obtained from a small random sample of 4% of the objects;

(d) PCA scores obtained from all objects. The number of dimensions used in the projections was 3, 5, 10, 20, 50, 100, and 200. In Figure 6 the predictive ability $P_{MEAN}$ (Section 2.3) is displayed as a function of the number of neighbors with a separate line for each number of dimensions. Maximum performance with the original 1455-dimensional data is about 75% correctly classified with 3 to 5 neighbors. KNN classification with projection scores requires 50 to 100 dimensions for a similar result. The three projection methods applied yield very similar results with these numbers of dimension, and again 3 to 5 neighbors are optimal. Only for a small number of projection directions, the percentage of correctly classified objects was considerably lower for RP compared to the PCA-based methods. Computing time for the projection of the original $\boldsymbol{X}$-matrix (6506×1455) to 200 dimensions was 100 s for RP, 100 s for PCA using 4% of the objects, and 50 s for PCA using all objects (the reason for this short computation time is the optimized C code for SVD-based PCA used in this example). Computation time for leave-one-out cross validation was 1100 s for the original data, but only 120 s for 200 projection scores.

A significant reduction of computing time is principally also possible by a variable selection. However, only supervised problems allow a reasonable variable selection and for a large number of original variables a fast selection strategy must be applied. Fast selection methods typically investigate each variable separately and are known to be inefficient for most data sets. For a comparison with a dimensionality reduction by PCA or RP the variables with highest *t*-values (according to a *t*-test of the class means) have been selected and then used in KNN classification. Results are presented in Figure 6 as for the other methods; the classification performance with 20 or more selected variables is very similar to that with 20 or

more projection scores. However, variable selection used class information but PCA or RP did not.

Concluding the results of this example, a reduction of the 1455 variables (descriptors) to 100 projection scores requires only about 7% of the original data storage and reduces computation time for kNN classification by a factor of nine. A dimensionality reduction by RP or PCA, using a random sample of 4% of the objects, yields similar performances for this data set.

### 3.4. Calibration with PLS

Partial least-squares (PLS) regression is a widely applied linear method for multivariate calibration. For most data sets in chemometrics PLS regression does not necessarily require any preceding dimensionality reduction. However, dimensionality reduction is an essential part in the related method principal component regression (PCR), because the PCA scores are used as regressor variables – mainly in ordinary least-squares (OLS) regression. The following two examples investigate principally the combination of random projection (RP) with a subsequent PLS regression. For comparison we also use PCA scores in a following PLS regression. A pre-selected number of PCA scores with highest variances is computed from all objects, and from a random subset of 10 % requiring for PCA a similar computation time as for RP.

The data set TOX/GC is from $n$ = 846 organic, mostly toxic compounds relevant in forensic chemistry [46], characterized by $m$ = 529 molecular descriptors calculated by software Dragon [44] from approximated 3D-structures and all H-Atoms explicitly given (calculated

by software Corina [45]). The modeled property *y* is the gas chromatographic Kovats retention index (range 1110 - 3870, median 2000).

The data set GLC/NIR is from $n = 166$ centrifuged fermentation mashes, characterized by near infrared (NIR) spectra. In the wavelength range 1100–2300 nm 241 absorbances have been measured in 5 nm increments. Then the first derivatives of the spectra were computed by Savitzky-Golay differentiation [47] with a window size of seven data points and a second-order polynomial, resulting in $m = 235$ variables used for regression models [48]. The modeled property *y* is the concentration of glucose (range 0.32 - 54.4 g/L, median 15.6 g/L).

PLS regression has been used within the evaluation strategy repeated double cross validation (rdCV); the performance measure SEP has been estimated from 100 repetitions within rdCV yielding $100n$ prediction errors from $n$ test set samples [32]. Table I compares the results obtained by PLS with all original variables and PLS with different numbers of RP or PCA scores. For all considered methods and both data sets the model performance increases with increasing number of projection axes. For the TOX/GC data PCA scores (derived from all original variables) give considerably better models than RP scores, however, both are worse than simply using all variables. For the GLC/NIR data set a dimensionality reduction to 100 RP scores or 100 PCA scores is possible without a loss of performance. PCA with a random subset of 10% of the samples requires about the same computation time as RP; of course the number of PCA scores is limited in this case by the small number of objects. PCA scores obtained from the 10% sets give a similar performance of the PLS models as PCA scores obtained from all variables.

Concluding the results of the calibration examples, a reduction of an *x*-space with some hundred variables by RP or PCA to 100 projection scores is not recommended if PLS can be performed with all variables. However, for some data sets a fast and simple dimensionality reduction by RP can be applied if necessary without a decrease of the performance of PLS models.

## 4. Discussion and conclusions

The random projection (RP) method proved itself a very simple and fast method for dimensionality reduction. Projection vectors (loading vectors) in RP consist of random numbers taken from a distribution with zero mean; the shape or type of the used distribution has little influence. Pairs of random vectors in a high dimensional space, generated in this way, are almost orthogonal with a high probability. Consequently, an orthogonalization of the random vectors is not essential for most applications, although it can be easily realized. RP cannot compete with PCA in terms of variance preservation or small distortion of the data structure. However, RP scores yield similar performances in clustering and classification problems as PCA scores or the original high dimensional data set. The number of RP projection vectors required for good results was 5 - 10% of the number of the variables in the investigated data sets. In general PCA with a random sample of 1 - 10% of the objects required the same computation time as RP and yielded similar results as RP. The projection matrix in RP only depends on the dimensionality of the original data but not on the data itself. Thus, no information included in the data can be utilized by RP, but on the other hand artifacts in the data have no influence. Furthermore, a projection matrix can be defined even before the acquisition of data, which might be useful for very remote experiments.

A data set with more than 50,000 chemical structures, characterized by more than 1000 binary substructure descriptors has been used to evaluate the capability of RP in structure similarity searches. A set of 100 RP scores gave structure similarity hitlists that contain about 3/4 of the hits obtained by a search with all binary descriptors.

Classification of the mutagenicity based on molecular descriptors showed that a dimensionality reduction from 1455 original variables to 100 RP scores gave very similar KNN classification results as the original variables, but required only 7% of the data storage and 11% of the computation time.

Application of RP for dimensionality reduction prior to PLS regression was successful for one of two investigated data sets. However, most calibration data sets in chemometrics are rather small and PLS can easily handle many and correlating variables; a powerful variable selection - e.g. by a genetic algorithm or by O-PLS - is more recommended for such data sets than RP or PCA.

Random projection has already been successfully applied in various areas of machine learning, especially for very large data sets; here the applicability for data from chemoinformatics and chemometrics was demonstrated. The usefulness of RP particularly opens up in situations with high dimensional data combined with possible restrictions in data storage and computing time.

**Acknowledgements**

We thank Ulrich Omasits (ETH Zürich) for work with simulated data, Anton Friedl (Vienna University of Technology) for continuous support in this project, and Katja Hansen (Technical University of Berlin) for providing the Ames data. Johan Silen (Finnish Meteorological Institute, Helsinki) brought random projection to our knowledge in connection with a planned application to data from a space experiment.

**Legends of figures**

**Figure 1.** Distribution of the cosine of the angle $\alpha$ between two random projection vectors. The vector components have been drawn from three different distributions; N(0,1), the standard normal distribution; U[-1, 1], a uniform distribution between -1 and +1; [-1, 1], values -1 and +1 with equal probabilities. The number of vector components (dimensions), *m*, is 2500 (solid lines), 500 (dashed lines), and 100 (dotted lines). From 10,000 vector pairs the probability densities are estimated at 512 equally spaced values in the cos $\alpha$ range from -0.2 to +0.2. The approximations of the empirical distributions by N(0, $(1/m)^{0.5}$) are shown in gray.

**Figure 2.** *Artificial data set 1.* Proportion of explained variance (left), computation time (middle) and average distortion (right) for RP, PCA and PCA using only 1% of the data, based on simulated normally distributed data.

**Figure 3.** *Artificial data set 2.* Average predictive abilities $P_{MEAN}$ for the projected and the original data.

**Figure 4.** Box plots for NCH, the number of common hits in hitlists obtained from 1233 binary descriptors (using the Tanimoto index) and hitlists obtained from projection scores with 10, 30, and 100 dimensions (Euclidean distance for real value variables). The size of the hitlist was 5, 10 and 20. Each box plot is computed from 1000 NCH values for 1000 randomly selected query structures.

**Figure 5.** Distribution of Tanimoto indices between 1000 query structures and the corresponding 10 hits. Solid line for hitlists obtained from 1233 binary substructure descriptors; dashed lines for hitlists obtained from 30 and 100 RP scores, respectively.

**Figure 6.** KNN classification of mutagenicity from molecular descriptors. $P_{MEAN}$ is the mean of the predictive abilities (% correctly classified) of the active and the inactive class, as obtained by leave-one-out cross validation. The number of neighbors, *k*, was varied between 1 and 101. The dashed line is for the original 1455 descriptors. The solid lines are for lower dimensional spaces obtained by RP, PCA projection with 4% of the objects, PCA with all objects, and variable selection. The reduced dimensionality was varied between 3 and 200.

**Table I.** Dimensionality reduction by random projection (RP) and principal component analysis (PCA) before PLS regression. PCA has been used will all objects and with a 10% random sample.

| Data set | *n* | Dimensionality reduction | *p* | SEP (rdCV) | *a* |
|---|---|---|---|---|---|
| TOX/GC | 846 | no | 529 | 86 | 14 |
| | 846 | RP | 3 | 308 | 2 |
| | 846 | RP | 20 | 170 | 3 |
| | 846 | RP | 100 | 124 | 12 |
| | 84 | PCA (10 %) | 3 | 215 | 2 |
| | 84 | PCA (10 %) | 20 | 170 | 4 |
| | 84 | PCA (10 %) | 84 | 108 | 11 |
| | 846 | PCA (all) | 3 | 208 | 2 |
| | 846 | PCA (all) | 20 | 171 | 4 |
| | 846 | PCA (all) | 100 | 92 | 13 |
| GLC/NIR | 166 | no | 235 | 6.5 | 9 |
| | 166 | RP | 3 | 13.0 | 1 |
| | 166 | RP | 20 | 11.2 | 3 |
| | 166 | RP | 100 | 6.8 | 14 |
| | 16 | PCA (10 %) | 3 | 12.2 | 1 |
| | 16 | PCA (10 %) | 16 | 9.8 | 4 |
| | 166 | PCA (all) | 3 | 12.3 | 1 |
| | 166 | PCA (all) | 20 | 9.0 | 5 |
| | 166 | PCA (all) | 100 | 6.6 | 9 |

*n* is the number of objects; *p* is the number of variables (original variables or projection scores) used as regressor variables in PLS; *a* is the estimated optimum number of PLS components.

**References**


1. Vempala SS. The random projection method. American Mathematical Society: Providence, RI, 2004.
2. Achlioptas D. Database-friendly random projections: Johnson-Lindenstrauss with binary coins. *Journal of Computer and System Sciences* 2003; **66**: 671-687.
3. Hecht-Nielsen R. Context vectors; general purpose approximate meaning representations self-organized from raw data. In: Zurada JM, Marks RJ, Robinson CJ, editors. Computational intelligence: imitating life: IEEE Press; 1994.
4. Bingham E, Mannila H. Random projection in dimensionality reduction: Applications to image and text data. Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco, CA: Association for Computing Machinery (ACM); 2001. p 245-250. www.cis.hut.fi/ella/publications/randproj_kdd.pdf
5. Lin J, Gunopulos D. Dimensionality reduction by random projection and latent semantic indexing. Text Mining Workshop, at the 3rd SIAM International Conference on Data Mining. San Francisco, CA; 2003. www.cs.ucr.edu/~jessica/RP_LSI_SDM03.pdf
6. Papadimitriou CH, Raghavan P, Tamaki H, Vempala S. Latent semantic indexing: A probabilistic analysis. Proc. 17th ACM Symp on the Principles of Database Systems; 1998.
7. Dasgupta S. Experiments with random projection. Sixteenth Conference on Uncertainty in Artificial Intelligence (UAI). San Francisco, CA: Morgan Kaufmann Publishers Inc.; 2000. p 143-151. www-cse.ucsd.edu/~dasgupta/papers/randomf.pdf

8. Amador JJ. Random projection and orthonormality for lossy image compression. *Image and Vision Computing* 2007; **25**(5): 754-766.

9. Fowler JE. Compressive-projection principal component analysis. *IEEE Transactions on Image Processing* 2009; **18**(10): 2230-2242.

10. Goel N, Bebis G, Nefian A. Face recognition experiments with random projection. Proceedings SPIE. Volume 5776; 2005. p 426. www.anefian.com/research/goel05_face.pdf

11. Kurimo M. Indexing audio documents by using latent semantic analysis and SOM. In: Oja E, Kaski S, editors. Kohonen maps. Amsterdam, The Netherlands: Elsevier; 1999.

12. Lonardi S, Szpankowski W, Yang Q. Finding biclusters by random projections. *Theoretical Computer Science* 2006; **368**(3): 217-230.

13. Fern XZ, Brodley CE. Random projection for high dimensional data clustering: A cluster ensemble approach. In: Fawcett T, Mishra N, editors. Proceedings of 20th International Conference on Machine Learning (ICML2003), ISBN 978-1-57735-189-4,. Washington, DC; 2003. www.aaai.org/Press/Proceedings/icml03.php

14. Avogadri R, Valentini G. Fuzzy ensemble clustering based on random projections for DNA microarray data analysis. *Artificial Intelligence in Medicine* 2009; **45**: 173-183.

15. Indyk P, Motwani R. Approximate nearest neighbors: towards removing the curse of dimensionality. Proceedings 30th Symposium on Theory of Computing: ACM; 1998. p 604-613.

16. Kleinberg JM. Two algorithms for nearest-neighbor search in high dimensions. Proceedings of the twenty-ninth annual ACM symposium on theory of computing. El Paso, Texas, United States: ACM; 1997.

17. Cuesta-Albertos JA, Fraiman R, Ransford T. Random projections and goodness-of-fit tests in infinite-dimensional spaces. *Bull. Braz. Math. Soc., New Series* 2006; **37**(4): 1-25.

18. Buhler J, Tompa M. Finding motifs using random projections. *J. Computational Biology* 2002; **9**: 225-242.

19. Huang GB, Zhu QY, Siew CK. Extreme learning machine: Theory and applications. *Neurocomputing* 2006; **70**: 489-501.

20. Miche Y, Bas P, Jutten C, Simula O, Lendasse A. A methodology for building regression models using extreme learning machine. European Symposium on Artificial Neural Networks, OP-ELM, ESANN 2007. Bruges, Belgium; 2008. p 457-462.

21. Verstraeten D, Schrauwen B, D'Haene M, Stroobandt D. An experimental unification of reservoir computing methods. *Neural Networks* 2007; **20**: 391-403.

22. Arriaga RI, Vempala S. An algorithmic theory of learning: Robust concepts and random projection. *Machine Learning* 2006; **63**(2): 161-182.

23. Al-Naymat G, Chawla S, Gudmundsson J. Dimensionality reduction for long duration and complex spatio-temporal queries. *Proceedings of the ACM Symposium on Applied Computing* 2007: 393-397.

24. Johnson WB, Lindenstrauss J. Extensions of Lipschits mapping into a Hilbert space. *Contemp. Math.* 1984; **26**: 189-206.

25. Dasgupta S, Gupta A. An elementary proof of the Johnson-Lindenstrauss lemma. Technical report 99-006 Berkeley: U.C. Berkeley, CA, www-cse.ucsd.edu/~dasgupta/papers/jl-tr.ps; 1999.

26. Frankl P, Maehara H. The Johnson-Lindenstrauss Lemma and the sphericity of some graphs. *J. Comb. Theory Ser. A* 1987; **44**(3): 355-362.

27. Kaski S. Dimensionality reduction by random mapping: Fast similarity computation for clustering. Proceedings of the 1998 Int. Joint Conf. on Neural Networks, Piscataway, NJ; 1998. p 413-418. ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=682302&isnumber=15009

28. Johnson RA, Wichern DW. Applied multivariate statistical analysis. Prentice Hall: Upper Saddle River, NJ, USA, 2002.

29. Massart DL, Vandeginste BGM, Buydens LCM, De Jong S, Smeyers-Verbeke J. Handbook of chemometrics and qualimetrics: Part A. Elsevier: Amsterdam, The Netherlands, 1997.

30. Varmuza K, Filzmoser P. Introduction to multivariate statistical analysis in chemometrics. CRC Press: Boca Raton, FL, 2009.

31. Mevik BH, Wehrens R. The pls package: Principal component and partial least squares regression in R. *J. Stat. Software* 2007; **18**(2): 1-24.

32. Filzmoser P, Liebmann B, Varmuza K. Repeated double crosss validation. *J. Chemometr.* 2009; **23**: 160-171.

33. R. A language and environment for statistical computing. R Development Core Team, Foundation for Statistical Computing, www.r-project.org: Vienna, Austria, 2009.

34. Fraley C, Raftery A. How many clusters? Which clustering method? Answers via model-based cluster analysis. *The Computer Journal* 1998; **41**: 578-588.

35. Gasteiger J, editor. Handbook of chemoinformatics - From data to knowledge (4 volumes). Weinheim, Germany: Wiley-VCH; 2003.

36. Willett P. Similarity and clustering in chemical information systems. Research Studies Press: Letchworth, United Kingdom, 1987.

37. Vandeginste BGM, Massart DL, Buydens LCM, De Jong S, Smeyers-Verbeke J. Handbook of chemometrics and qualimetrics: Part B. Elsevier: Amsterdam, The Netherlands, 1998.

38. Demuth W, Karlovits M, Varmuza K. Spectral similarity versus structural similarity: mass spectrometry. *Anal. Chim. Acta* 2004; **516**: 75-85.

39. Scsibrany H, Karlovits M, Demuth W, Müller F, Varmuza K. Clustering and similarity of chemical structures represented by binary substructure descriptors. *Chemom. Intell. Lab. Syst.* 2003; **67**: 95-108.

40. NIST. Mass Spectral Database 98. National Institute of Standards and Technology, www.nist.gov/srd/nist1a.htm: Gaithersburg, MD, USA, 1998.

41. Varmuza K, Demuth W, Karlovits M, Scsibrany H. Binary substructure descriptors for organic compounds. *Croatica Chimica Acta* 2005; **78**: 141-149.

42. SubMat. Software. Scsibrany H, Varmuza K, Laboratory for Chemometrics, Vienna University of Technology, www.lcm.tuwien.ac.at.: Vienna, Austria, 2004.

43. Hansen K, Mika S, Schroeter T, Sutter A, ter Laak A, Steger-Hartmann T, Heinrich N, Müller KR. Benchmark data set for in silico prediction of Ames mutagenicity. *J. Chem. Inf. Model.* 2009; **49**: 2077-2081.

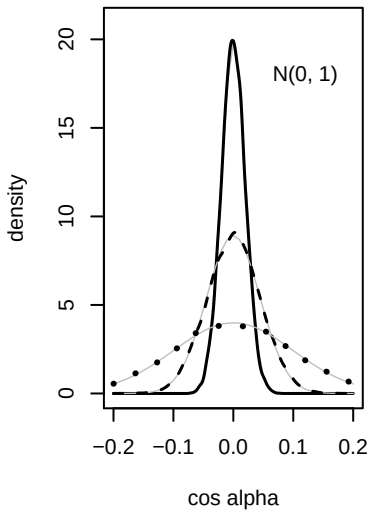
44. Dragon. Software for calculation of molecular descriptors, by Todeschini R, Consonni V, Mauri A, Pavan M. Talete srl, www.talete.mi.it: Milan, Italy, 2004.

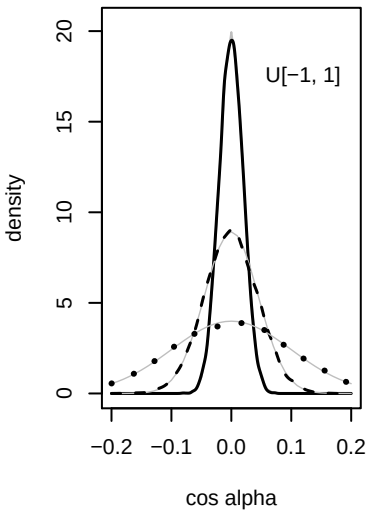
45. Corina. Software for the generation of high-quality three-dimensional molecular models, by Sadowski J, Schwab CH, Gasteiger J. Molecular Networks GmbH Computerchemie, www.mol-net.de: Erlangen, Germany, 2004.

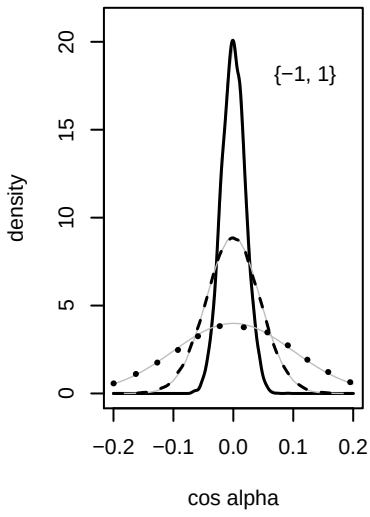
46. Garkani-Nejad Z, Karlovits M, Demuth W, Stimpfl T, Vycudilik W, Jalali-Heravi M, Varmuza K. Prediction of gas chromatographic retention indices of a diverse set of toxicologically relevant compounds. *J. Chromatogr. B* 2004; **1028**: 287-295.

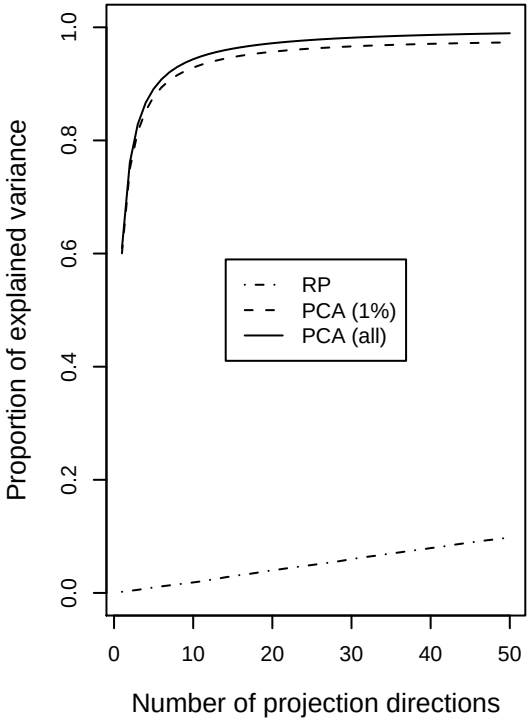
47. Naes T, Isaksson T, Fearn T, Davies T. A user-friendly guide to multivariate calibration and classification. NIR Publications: Chichester, United Kingdom, 2004.

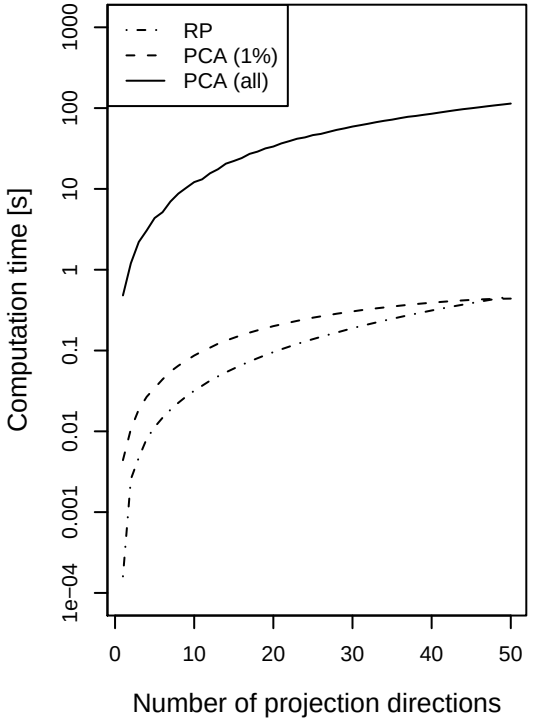
48. Liebmann B, Friedl A, Varmuza K. Determination of glucose and ethanol in bioethanol production by near infrared spectroscopy and chemometrics. *Anal. Chim. Acta* 2009; **642**: 171-178.

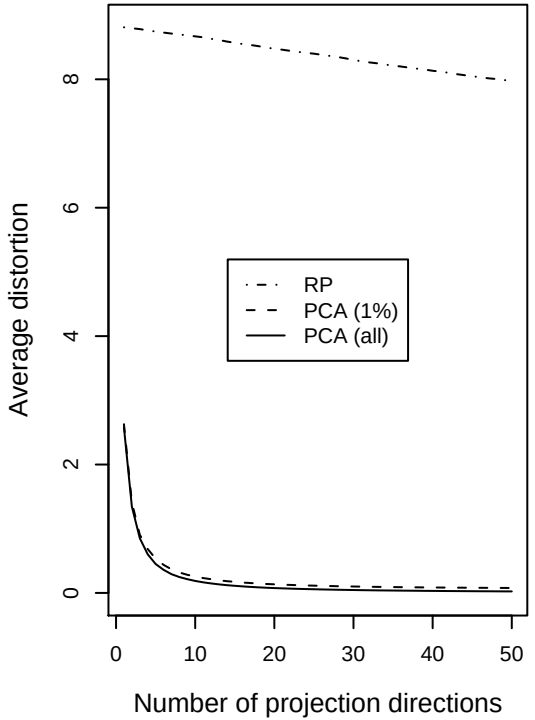

N(0, 1)
density
cos alpha
0
5
10
15
20
−0.2
−0.1
0.0
0.1
0.2

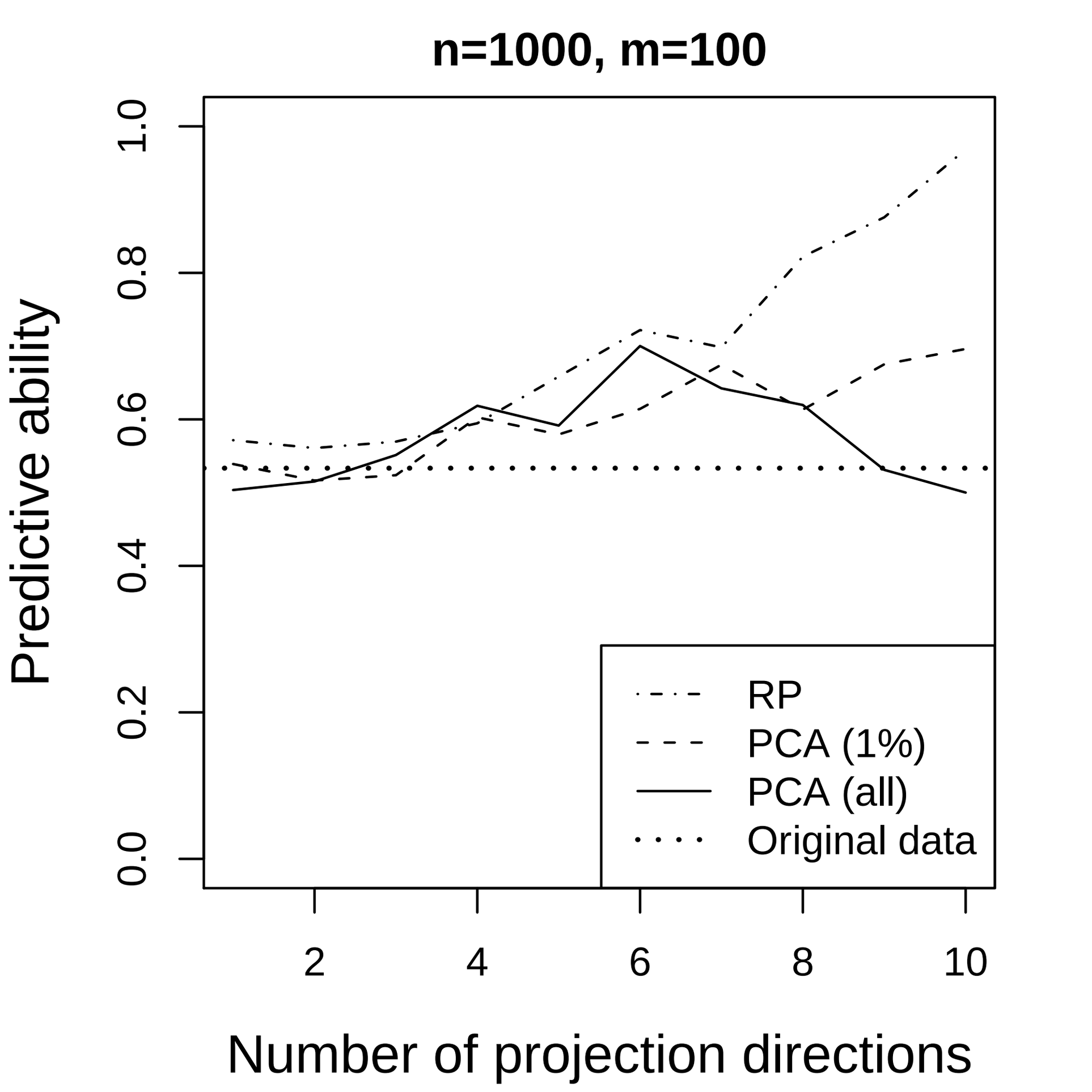

U[−1, 1]
density
cos alpha
0
5
10
15
20
−0.2
−0.1
0.0
0.1
0.2

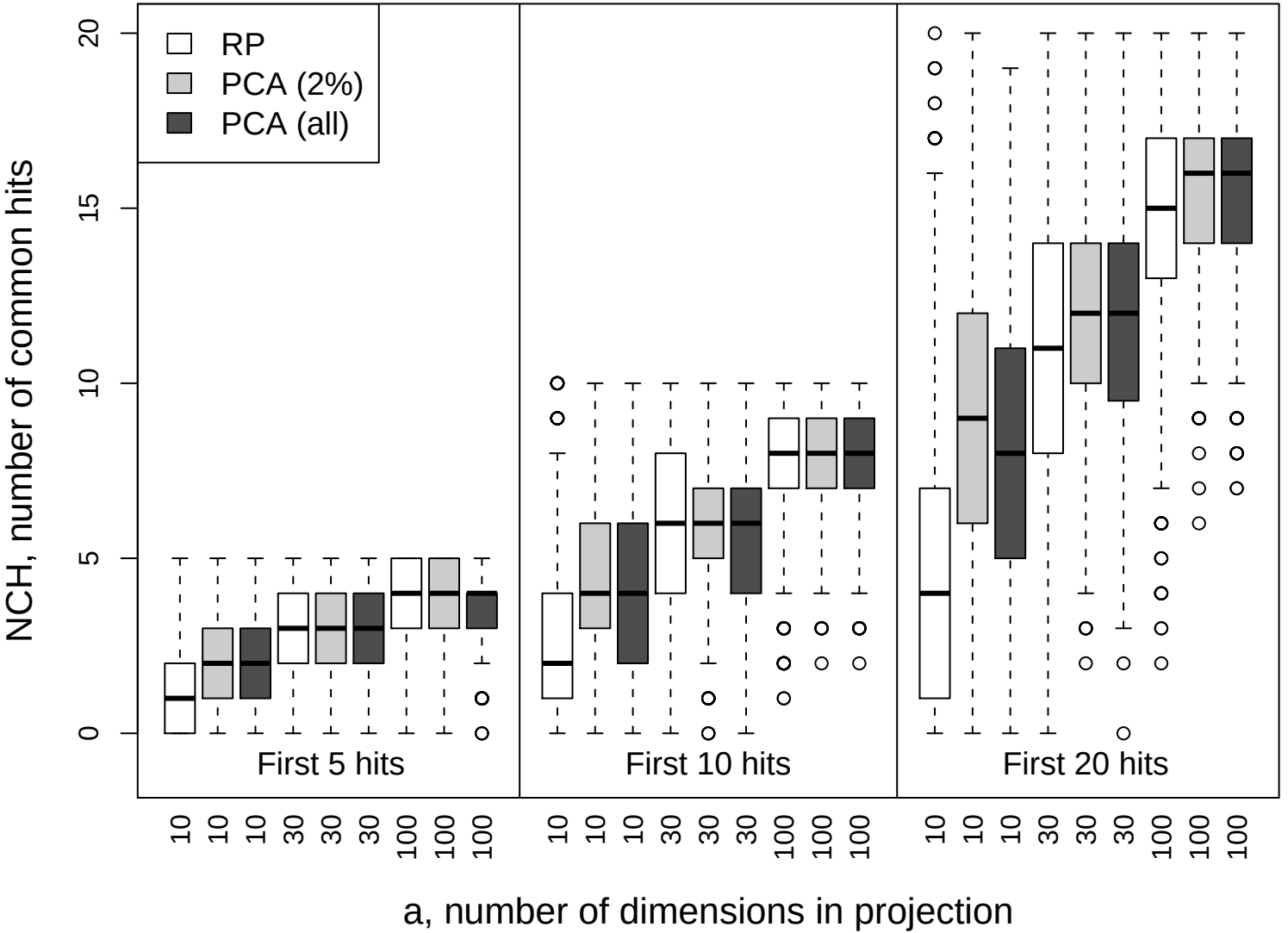

{−1, 1}
density
cos alpha
0
5
10
15
20
−0.2
−0.1
0.0
0.1
0.2

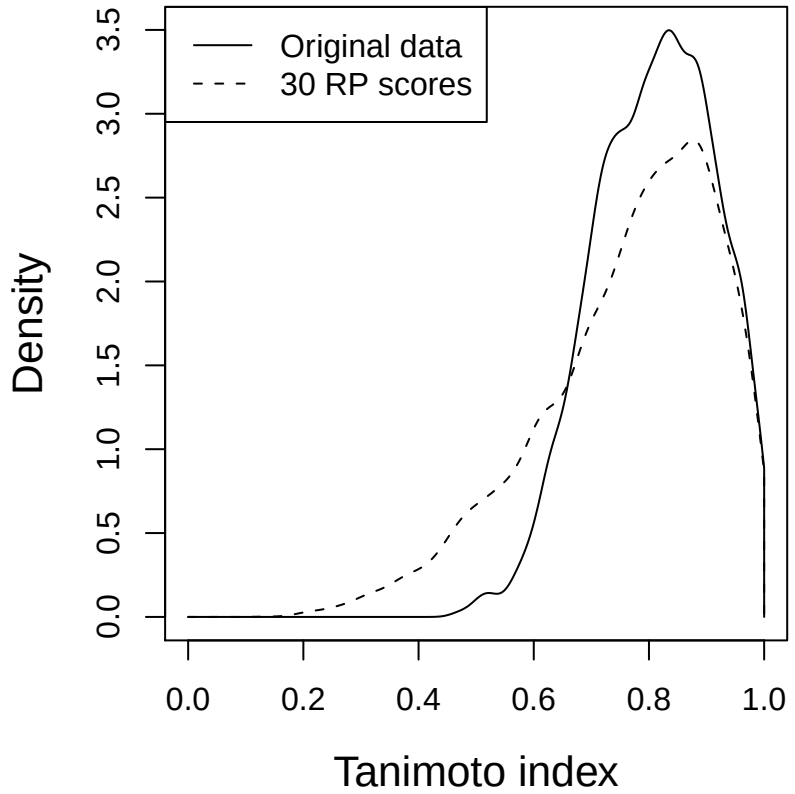

Proportion of explained variance
0.0
0.2
0.4
0.6
0.8
1.0
RP
PCA (1%)
PCA (all)
0
10
20
30
40
50
Number of projection directions

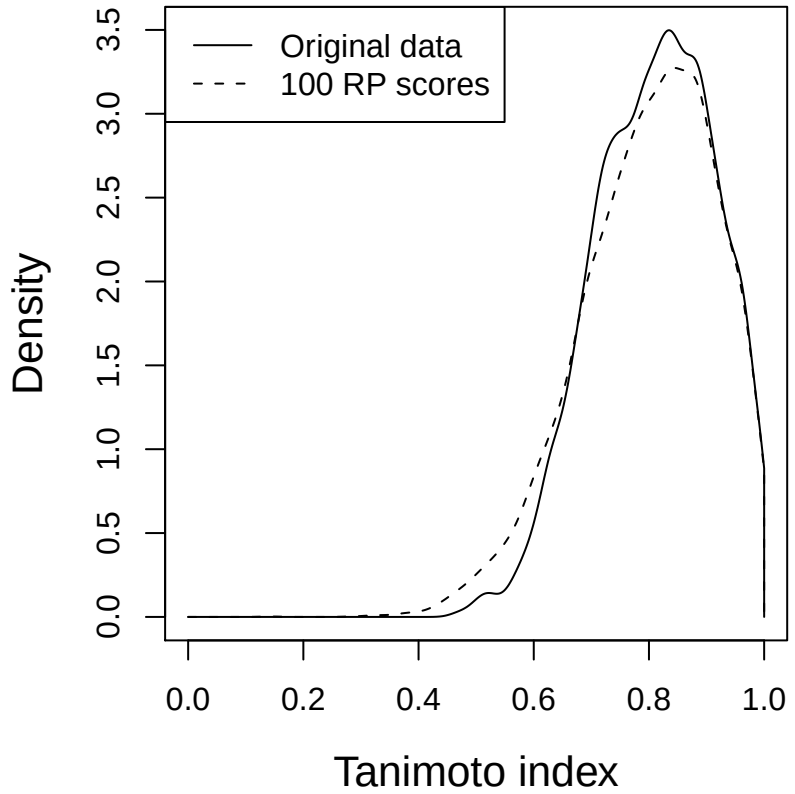

Computation time [s]
1e−04
0.001
0.01
0.1
1
10
100
1000
RP
PCA (1%)
PCA (all)
0
10
20
30
40
50
Number of projection directions

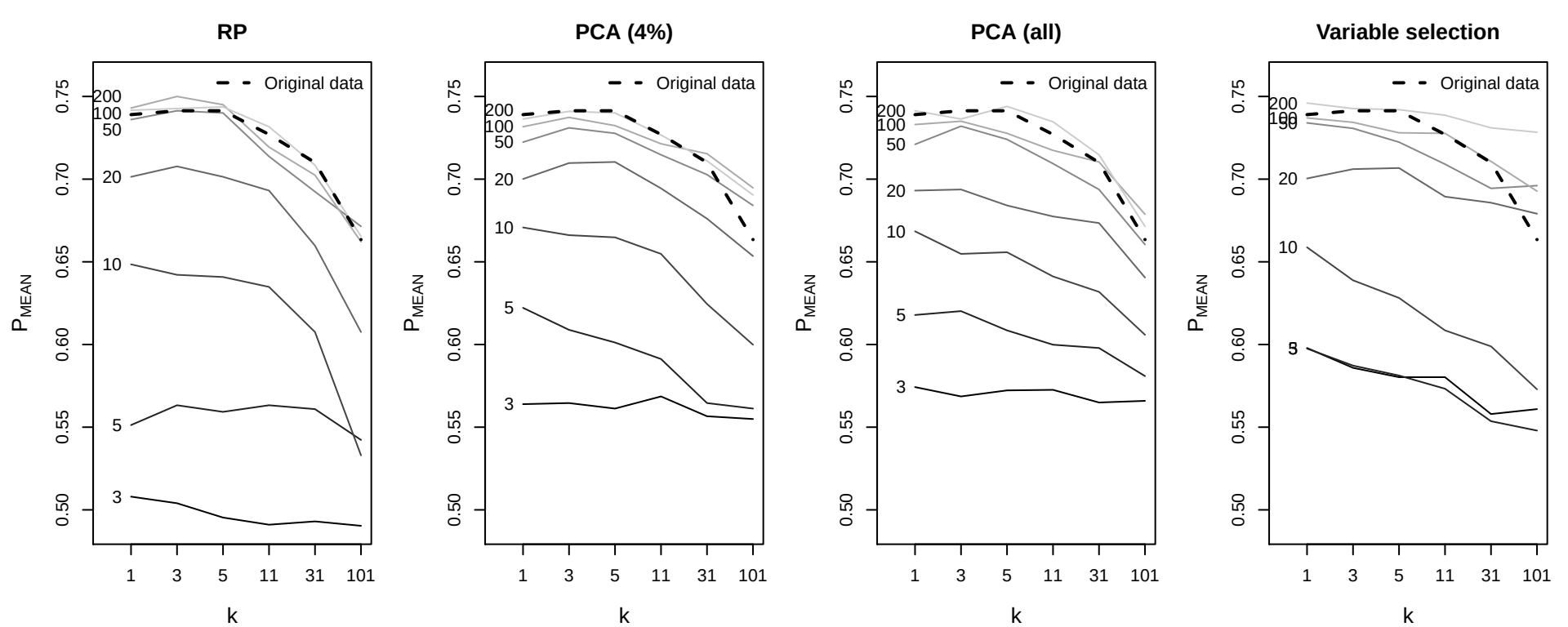

Average distortion
0
2
4
6
8
RP
PCA (1%)
PCA (all)
0
10
20
30
40
50
Number of projection directions


n=1000, m=100
Predictive ability
0.0
0.2
0.4
0.6
0.8
1.0
2
4
6
8
10
Number of projection directions
RP
PCA (1%)
PCA (all)
Original data


RP
PCA (2%)
PCA (all)
NCH, number of common hits
0
5
10
15
20
First 5 hits
First 10 hits
First 20 hits
10 10 10 30 30 30 100 100 100
10 10 10 30 30 30 100 100 100
10 10 10 30 30 30 100 100 100
a, number of dimensions in projection


Original data
30 RP scores
Density
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
0.0
0.2
0.4
0.6
0.8
1.0
Tanimoto index


Original data
100 RP scores
Density
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
0.0
0.2
0.4
0.6
0.8
1.0
Tanimoto index


RP
PCA (4%)
PCA (all)
Variable selection
Original data
200
100
50
20
10
5
3
$P_{MEAN}$
0.50
0.55
0.60
0.65
0.70
0.75
1
3
5
11
31
101
k