

MUTUAL INFORMATION BASED DIMENSIONALITY REDUCTION WITH APPLICATION TO NON-LINEAR REGRESSION

Lev Faivishevsky, Jacob Goldberger

School of Engineering, Bar Ilan University, Israel

ABSTRACT

In this paper we introduce a supervised linear dimensionality reduction algorithm which is based on finding a projected input space that maximizes mutual information between input and output values. The algorithm utilizes the recently introduced MeanNN estimator for differential entropy. We show that the estimator is an appropriate tool for the dimensionality reduction task. Next we provide a nonlinear regression algorithm based on the proposed dimensionality reduction approach. The regression algorithm achieves comparable to state-of-the-art performance on the standard datasets being three orders of magnitude faster. In addition we demonstrate an application of the proposed dimensionality reduction algorithm to reduced-complexity classification.

1. INTRODUCTION

Many real world regression problems deal with analysis of high-dimensional data. The goal of supervised dimensionality reduction is to reduce the dimensionality of the input space while preserving the information about the output values. A common method is Canonical Correlation Analysis (CCA) [1] which is a way of measuring the linear relationship between two multidimensional variables. It finds two bases, one for each variable, that are optimal with respect to correlations and, at the same time, it finds the corresponding correlations. Partial Least Squares (PLS) was developed in econometrics in the 1960s by Herman Wold. PLS is basically the singular-value decomposition of a between-sets covariance matrix; for an overview, see e.g. [2]. In the PLS regression, the principal vectors corresponding to the largest principal values are used as a new, lower dimensional, basis for the signal. A regression of y onto x is then calculated in this new space. See [3] for a recent study in dimensionality reduction using Reproducing Hilbert Kernel Spaces. Other relevant studies include [4], [5].

Recently, methods for feature extraction that are based on mutual information maximization have been proposed, see e.g. [6]. These methods, however, perform dimensionality reduction, only by choosing a subset of given dimensions and apply kNN based mutual information estimators. Torkkola [7] suggested using mutual information for supervised dimensionality reduction in a classification setup by learning a Parzen window approximation for the joint input-target distribution. One of the main difficulties with this kernel based estimation lies in the need for a correct choice of kernel width.

Another approach named Maximum Mutual Information Projection (MMIP) [8] proposed dimensionality reduction based on histogram estimation of mutual information. A significant drawback of the approach is in the treatment of multidimensional result subspace. The histogram estimation of mutual information is

of exponential complexity with respect to the result subspace dimension. Therefore the MMIP finds the subspace in an iterative manner, performing in each step an optimal one-dimensional projection. The next iteration then searches for optimal 1D subspace in the orthogonal complement subspace of previously found 1D projections. This approach is suboptimal because the mutual information is a nonlinear function and the orthogonal projection does not remove dependence on the previously found 1D subspaces. It means that resulting output multidimensional subspace does not necessarily correspond to the maximal information between the projected inputs and targets. Recently proposed Maximal Mutual Information Feature Extractor (MMIFE) [9] is based on a nonlinear entropy estimation of 1D random variable. The MMIFE has a similar drawback as MMIP in the case of multidimensional resulting subspaces since the MMIFE applies the same scheme of iterative 1D projections and orthogonal complements.

In this paper we present an algorithm for supervised linear dimensionality reduction that uses mutual information as a criterion. The advantage of this method is that it preserves the information contained in the input space by searching for optimal linear combinations of existing features. This optimization is efficiently accomplished by conjugated gradients methods applied to the recently introduced MeanNN estimator [10] for the mutual information, that benefits from an analytical expression for the gradient. Based on this estimator we define an efficient nonlinear regression in the extracted linear subspace. The performance of the proposed regression is comparable to state-of-the-art methods while being three orders of magnitude faster at the test stage. The same dimensionality reduction concept is then applied to classification tasks.

The rest of the paper is organized as follows. Section 2 reviews non-parametric k NN and MeanNN estimators for differential entropy. Section 3 introduces the Mutual Information Dimensionality Reduction method. Section 4 presents the non-linear regression method. Section 5 presents an application to classification problems. Section 6 reports experiment results on standard datasets.

2. NONPARAMETRIC ESTIMATORS FOR DIFFERENTIAL ENTROPY

Our dimensionality reduction method is based on a smooth non-parametric approximation of differential entropy that is reviewed below. The differential entropy of X is defined as:

$$H(X) = - \int f(x) \log f(x) dx \quad (1)$$

We describe the derivation of the Shannon differential entropy estimate of [11], [12]. Our aim is to estimate $H(X)$ from a random sample (x_1, \dots, x_n) of n random realizations of a d -dimensional

random variable X with an unknown density function $f(x)$. The entropy is the average of $-\log f(x)$. If there were unbiased estimators for $\log f(x_i)$, this would yield an unbiased estimator for the entropy. We estimate $\log f(x_i)$ by considering the probability density function $P_{ik}(\epsilon)$ for the distance between x_i and its k -th nearest neighbor (the probability is computed over the positions of all other $n - 1$ points, with x_i kept fixed). The probability $P_{ik}(\epsilon)d\epsilon$ is equal to the chance that there is one point within distance $r \in [\epsilon, \epsilon + d\epsilon]$ from x_i , that there are $k - 1$ other points at smaller distances, and that the remaining $n - k - 1$ points have larger distances from x_i . Denote the mass of the ϵ -ball centered at x_i by $p_i(\epsilon)$, i.e. $p_i(\epsilon) = \int_{\|x - x_i\| < \epsilon} f(x)dx$. Applying the trinomial formula we obtain:

$$P_{ik}(\epsilon) = \frac{(n-1)!}{1!(k-1)!(n-k-1)!} \frac{dp_i(\epsilon)}{d\epsilon} p_i^{k-1} (1-p_i)^{n-k-1} \quad (2)$$

It is easy to verify that $\int P_{ik}(\epsilon)d\epsilon = 1$. Hence, the expected value of the function $\log p_i(\epsilon)$ according to the distribution $P_{ik}(\epsilon)$ is:

$$E_{P_{ik}(\epsilon)}(\log p_i(\epsilon)) = \int_0^\infty P_{ik}(\epsilon) \log p_i(\epsilon) d\epsilon = \quad (3)$$

$$k \binom{n-1}{k} \int_0^1 p^{k-1} (1-p)^{n-k-1} \log p dp = \psi(k) - \psi(n)$$

where $\psi(x)$ is the digamma function (the logarithmic derivative of the gamma function). To verify the last equality, differentiate the identity $\int_0^1 x^{a-1} (1-x)^{b-1} = \Gamma(a)\Gamma(b)/\Gamma(a+b)$ with respect to parameter a and recall that $\Gamma'(x) = \psi(x)\Gamma(x)$. The expectation is taken over the positions of all other $n - 1$ points, with x_i kept fixed. Assuming that $f(x)$ is almost constant in the entire ϵ -ball around x_i , we obtain:

$$p_i(\epsilon) \approx c_d \epsilon^d f(x_i) \quad (4)$$

where d is the dimension of x and c_d is the volume of the d -dimensional unit ball ($c_d = \pi^{d/2}/\Gamma(1+d/2)$ for the Euclidean norm). Substituting Eq. (4) into Eq. (3), we obtain:

$$-\log f(x_i) \approx \psi(n) - \psi(k) + \log(c_d) + dE(\log(\epsilon)) \quad (5)$$

which leads to the unbiased k NN estimator for the entropy [11]:

$$H_k(X) = \psi(n) - \psi(k) + \log(c_d) + \frac{d}{n} \sum_{i=1}^n \log \epsilon_i \quad (6)$$

where ϵ_i is the distance from x_i to its k -th nearest neighbor. An alternative proof of the asymptotic unbiasedness and consistency of the k NN estimator can be found in [13]. This estimation forms a connection between information theory and nearest-neighbor concepts. Unlike previously suggested approximations, there are no parameters to be tuned.

Being non-parametric, the k NN estimator (6) relies on the order statistics. This makes the analytical calculation of the gradient virtually impossible. Also it leads to a certain lack of smoothness of the estimator value as a function of the sample coordinates. Finally, finding the k -nearest neighbor is a computationally intensive problem. It becomes practically obligatory to use involved approximate nearest neighbor techniques for large data sets.

Recently [10] proposed a new smooth estimator for the entropy evaluation as a function of sample coordinates. It is based

on the fact that the k NN estimator (6) is valid for every k . Therefore the differential entropy can be also extracted from a mean of several estimators corresponding to different values of k . Next we consider all the possible values of order statistics k from 1 to $n - 1$:

$$H_{mean} = \frac{1}{n-1} \sum_{k=1}^{n-1} H_k = \quad (7)$$

$$= \log(c_d) + \psi(n) + \frac{1}{n-1} \sum_{k=1}^{n-1} (-\psi(k)) + \frac{d}{n} \sum_{i=1}^n \log \epsilon_{i,k}$$

where $\epsilon_{i,k}$ is the k -th nearest neighbor of x_i . Consider the double-summation last term in Eq. (7). Exchanging the order of summation, the last sum adds for each sample point x_i the sum of the log of its distances to all its nearest neighbors in the sample. It is of course equivalent to the sum of the logs of its distances to all other points in the sample set. Hence the mean estimator (7) for the differential entropy can be written as:

$$H_{mean} = \frac{d}{n(n-1)} \sum_{i \neq j} \log \|x_i - x_j\| + \text{const} \quad (8)$$

Note that, unlike the k NN based estimator, this entropy estimator is a smooth function of the given data points and is not sensitive to small perturbations in the values of x_1, \dots, x_n .

Next assume that additional to input vectors $x_1, \dots, x_n \in \mathcal{R}^D$ we have also target values $y_1, \dots, y_n \in \mathcal{R}$. We can express the mutual information between X and Y by means of joint and marginal entropies. Using the MeanNN entropy estimator we get a MeanNN estimator for the mutual information:

$$I_{mean}(X; Y) = H_{mean}(X) + H_{mean}(Y) - H_{mean}(X, Y) \quad (9)$$

3. MI LINEAR DIMENSIONAL REDUCTION

In this study we address the problem of supervised dimensionality reduction. Our goal is to utilize the smooth entropy estimator, reviewed in the previous section, to form an information-theoretic criterion that can be easily optimized. Given n vectors $X = \{x_1, \dots, x_n\}$ in \mathcal{R}^D and corresponding target values $Y = \{y_1, \dots, y_n\}$ in \mathcal{R} we want to find a linear transformation $A : \mathcal{R}^D \rightarrow \mathcal{R}^d$ that maximizes the mutual information $I(AX; Y)$. Since we want to predict the target in the projected space, we search for features that are most correlated with the target. The mutual information criterion is a way to quantify this correlation. We search for a matrix A that maximizes the mutual information between the targets and transformed inputs.

To estimate the mutual information between a one-dimensional random variable Y and the d -dimensional random vector AX , with no prior information about their joint distribution, we apply the MeanNN estimator of mutual information (see Section 2). We want to maximize the mutual information as a function of the matrix A . An information theory relation reveals that: $I(AX; Y) = H(AX) + H(Y) - H(AX, Y)$. Since Y does not depend on A , to maximize the mutual information we need compute:

$$I_{mean}(AX; Y) = \text{const} + \frac{d}{2n(n-1)} \sum_{i \neq j} \log(\|A(x_i - x_j)\|^2) - \frac{d+1}{2n(n-1)} \sum_{i \neq j} \log(\|A(x_i - x_j)\|^2 + \|y_j - y_i\|^2) \quad (10)$$

To find the best linear dimensionality reduction we have to solve the optimization problem:

$$\hat{A} = \arg \max_A I_{mean}(AX; Y)$$

Such an optimization can be done using conjugate gradient techniques. The smoothness of the MeanNN entropy estimator enables its gradient to be analytically computed. Differentiating $I_{mean}(AX; Y)$ with respect to the transformation matrix A yields a gradient rule which we can use for learning:

$$\begin{aligned} \frac{\partial I_{mean}(AX; Y)}{\partial A} &= \frac{d}{n(n-1)} \sum_{i \neq j} \frac{A(x_i - x_j)(x_i - x_j)^\top}{\|A(x_i - x_j)\|^2} \\ &- \frac{d+1}{n(n-1)} \sum_{i \neq j} \frac{A(x_i - x_j)(x_i - x_j)^\top}{\|A(x_i - x_j)\|^2 + \|y_j - y_i\|^2} \quad (11) \end{aligned}$$

The learning algorithm therefore is: maximize the above objective (10) using a gradient-based optimizer such as delta-bar-delta or conjugate gradients. Of course, as the cost function above is not convex, some care must be taken to avoid local maxima during training. We dub the proposed method Mutual Information Dimensionality Reduction (MIDR). It is summarized in Figure 1. A standard information theory exercise reveals that $I(AX; Y)$ is invariant to any invertible transformation on either AX or Y . Since $\|A(x_i - x_j)\|^2 = (x_i - x_j)^\top A^\top A (x_i - x_j)$, our optimization criterion depends only on $A^\top A$. Hence, every orthogonal matrix $\mathbf{R}_{d \times d}$ yields a solution $\mathbf{R} \cdot A$ that is completely equivalent to A . Therefore our cost function $I_{mean}(AX; Y)$ is rotation invariant. We note in passing that we can make the MI approximation scale invariant by applying it to $I(AX; \|A\|Y) = I(AX; Y)$ such that $\|\cdot\|$ is the Frobenius norm. However, taking this approach the cost we optimize is no longer rotation invariant since the Frobenius norm is not rotation invariant. We choose to take the approach described above that yields a rotation invariant score. In our approach, to control the matrix scale we can penalize large-norm transformations A by adding a regularization term $-\lambda \|A\|^2$ to the cost function we are maximizing such that λ is a pre-specified positive constant that can be set in a cross validation step.

4. AN APPLICATION TO NON-LINEAR REGRESSION

To demonstrate the level of performance of the proposed dimensionality reduction method we next apply it to the problem of non-linear regression. Consider a fixed sample of n input points $X = \{x_1, \dots, x_n\}$ in R^D , along with target values $Y = \{y_1, \dots, y_n\}$ in \mathbf{R} . Our goal is to estimate a functional dependence: $y = f(x)$ in a way, that allows efficient computation of a predicted output $y_{test} = f(x_{test})$ for an input x_{test} at the testing stage.

We consider here the dimensionality reduction as a way to cope with challenges of high dimensional inputs space of regression tasks. The basic idea is to perform the dimensionality reduction of the input space as a preprocessing step that preserves peculiar information about the target function. A proper dimensionality reduction results in a low dimensional space such that the target function still may be predicted based on the features lying in that subspace. Such a prediction should then be performed by a regression technique. This prediction will be computationally efficient because it runs in the low dimensional subspace. Yet a high accuracy may be achieved with a good combination of a dimensionality reduction and regression method. For instance, a

linear dimensionality reduction may be performed at a first step to achieve an informative low dimensional subspace in which a more involved nonlinear prediction technique will be applied. Such two steps algorithm constitutes a nonlinear regression method that benefits from fast linear operation in initial input space and a precise regression in the reduced subspace that still may be performed fast and accurately.

Here we describe a simple nonlinear regression approach which shows high accuracy in terms of testing set error while remaining highly computationally efficient. We achieve this by running the MIDR to reduce the input dimensionality so as to obtain a smaller subspace, that still retains maximal information about the target values.

Afterwards, function g is approximated by a multinomial function P of degree l :

$$g(x) \approx P(Ax) = P(w)$$

$$P(w) = \sum_{i_1+i_2+\dots+i_d \leq l} c_{i_1 i_2 \dots i_d} w_1^{i_1} w_2^{i_2} \dots w_d^{i_d}$$

The approximation is done by fitting coefficients in the L_2 norm:

$$\hat{P} = \arg \min \|Y - P(w)\|_2$$

This minimization problem is a linear problem as a function of the polynomial coefficients and therefore it can be easily solved. Values of the intended intrinsic dimension d and the multinomial degree l are determined by the cross-validation.

Of course, the regression in the resulted linear subspace may be performed by other than polynomial regression approach. More powerful techniques such as GPR or SVM may be applied. However usually they are more computationally hard. In this performance vs complexity tradeoff we have chosen the polynomial regression because it is a well performing method yet it may be computed in a fast manner. In particular it does not require any subset from the training set to be stored for the test stage computations. Below we empirically demonstrate that in the output subspace of MIDR a simple polynomial regression achieves roughly the same or better results as an involved GPR technique.

The MIDR in this case may be viewed as a generalization of Projection Pursuit [14]. In fact, in the projection pursuit regression (PPR) the residual variance is brought to minimum whereas MIDR maximizes the mutual information between the predictor function and target values. Therefore the PPR method best suits for Gaussian variables as opposite to MIDR that has not inherent limitations for the explored variables distributions. We denote the non-linear regression algorithm based on the MIDR dimensionality reduction method followed by a polynomial regression model as ‘Mutual Information Polynomial Regression’ (MIPR) algorithm.

5. AN APPLICATION TO CLASSIFICATION

In this paper we focus on efficient non-linear regression algorithms. The proposed mutual information based linear dimensionality reduction can be also utilized for classification. The problem of classification differs from regression in terms of the domain of the target values. Namely, consider for a fixed sample of n input points $X = \{x_1, \dots, x_n\}$ in R^D : their target values are $C = \{c_1, \dots, c_n\}$ in a discrete set Ω , as opposed to \mathbf{R} in the case of regression. Here also our goal is to estimate a functional dependence: $c = f(x)$ in a way, that allows for efficient computation of a predicted output $c_{test} = f(x_{test})$ for an input x_{test} at

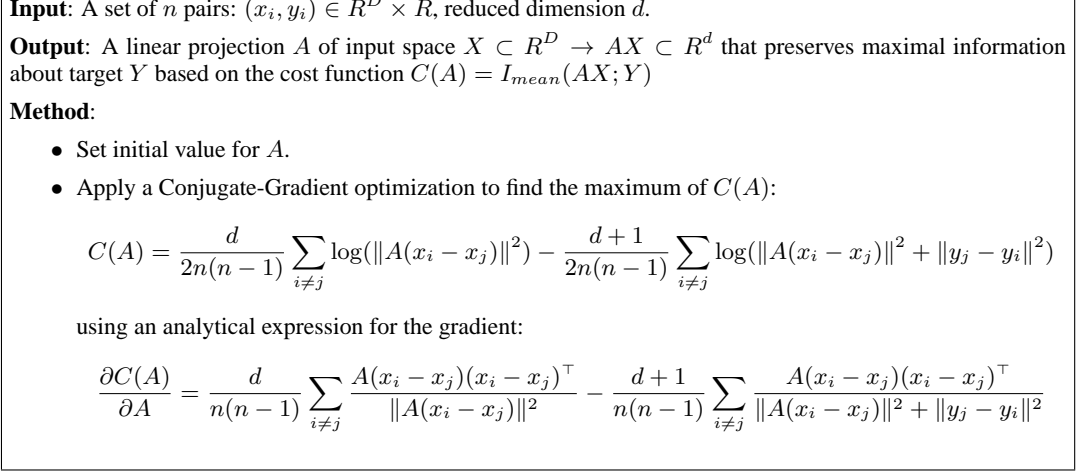


Fig. 1. The Mutual Information Dimensionality Reduction (MIDR) algorithm.

the testing stage. A standard method for classification is the kNN technique, in which a test input x_{test} is classified by finding its k -th nearest neighbor out of the train inputs \hat{x}_{train} and assigning $c_{test} = c_{train}(\hat{x}_{train})$. However, if the inputs belong to high-dimensional space the nearest neighbor search becomes computationally prohibitive. A useful technique to overcome the curse of dimensionality is to perform a dimensionality reduction in the input subspace prior to the NN classification. In fact, the LDA technique does exactly the same by projecting the inputs in the subspace spanned by eigenvectors of a combination of intra-inter covariance matrices [15]. In order to apply MIDR for the task of classification we need to estimate the mutual information between the continuous inputs and discrete target values. Fortunately, this can be done easily by means of conditional entropies:

$$I(X; C) = H(X) - H(X|C) = H(X) - \sum_{c \in \Omega} p(c)H(X|c) \quad (12)$$

In the above equation c denotes each possible value of the target variable, $p(c)$ is the probability for the target variable to have value c , and $H(X|c)$ is the entropy of the input random vector restricted to values such that their output equals c , so-called in-class entropy. So the MeanNN estimator can be applied to estimate the mutual information between the continuous and discrete variables. Therefore optimal linear dimensionality reduction matrix A is found by maximizing the MeanNN estimate of the mutual information.

$$\begin{aligned} \arg \max_A \left(H(AX) - \sum_{c \in \Omega} p(c)H(AX|c) \right) &\approx \\ \arg \max_A \left(\frac{1}{n(n-1)} \sum_{i \neq j} \log(\|A(x_i - x_j)\|^2) \right. & \\ \left. - \sum_{c \in \Omega} \frac{p(c)}{n_c(n_c-1)} \sum_{i_c \neq j_c} \log(\|A(x_{i_c} - x_{j_c})\|^2) \right) & \end{aligned} \quad (13)$$

where n_c is the number of samples having target value c . Clearly $n = \sum_{c \in \Omega} n_c$. The gradient of the MeanNN estimator of the

mutual information with respect to A is given by

$$\begin{aligned} \frac{\partial I}{\partial A} = & \left(\frac{2}{n(n-1)} \sum_{i \neq j} \frac{A(x_i - x_j)(x_i - x_j)^\top}{\|A(x_i - x_j)\|^2} - \right. \\ & \left. \sum_{c \in \Omega} \frac{2p(c)}{n_c(n_c-1)} \sum_{i_c \neq j_c} \frac{A(x_{i_c} - x_{j_c})(x_{i_c} - x_{j_c})^\top}{\|A(x_{i_c} - x_{j_c})\|^2} \right) \end{aligned}$$

A INN classification constitutes the last stage of the classification approach, here dubbed 'Mutual Information Classification' (MIC).

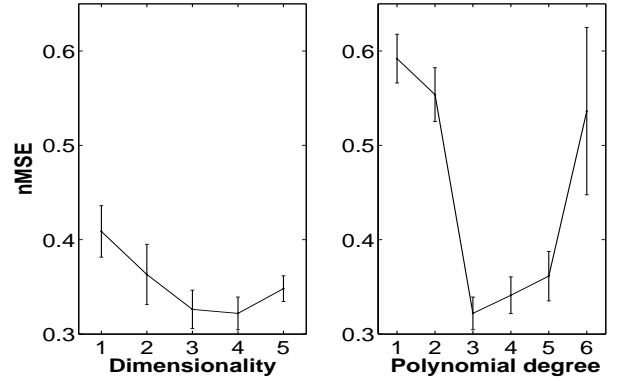


Fig. 3. MIPR parameters sensitivity. Variability of nMSE on the test set due to dimensionality of the reduced subspace (left), multinomial degree (right). Puma-8nh data, 1000 training samples, 1000 testing samples. Statistics are shown for 10 repetitions.

6. EXPERIMENTAL RESULTS

First, we compared MIPR performance with other possibilities for the regression in the resulting subspace, such as Gaussian Process Regression, considered today to be the state-of-the-art in regression. The GPR technique models the outputs as Gaussian processes with a few hyperparameters, see e.g. [16]. We also used

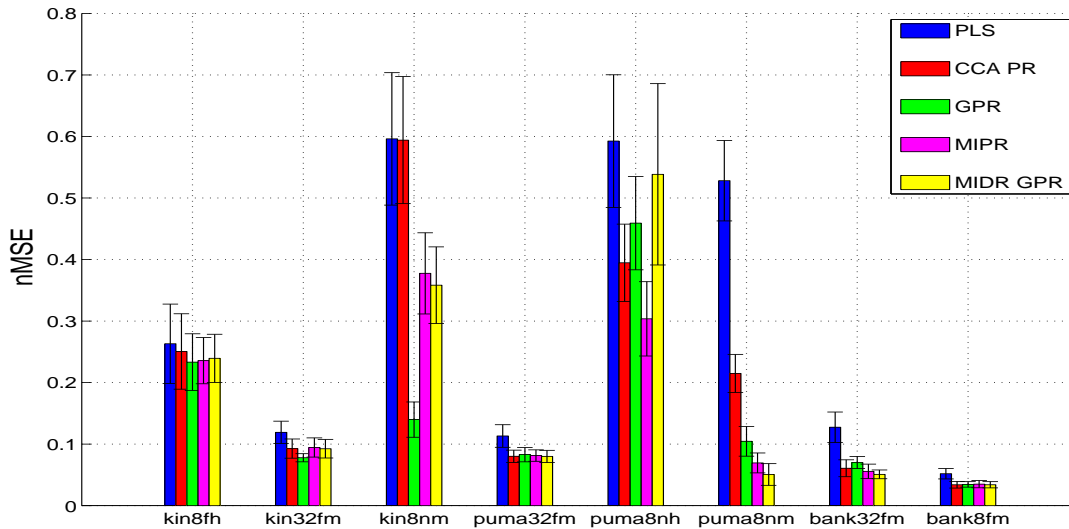


Fig. 2. Performance of several regression methods on Delve datasets. 1000 training samples, 1000 validation samples, 1000 testing samples. Statistics are shown for 10 repetitions.

the GPR on the full input space to evaluate the possible loss of accuracy due to dimensionality reduction. We applied Canonical Correlation Analysis as a representative alternative supervised linear dimensionality reduction method. The CCA method finds an optimal subspace by maximization of the correlation between the input and output spaces [1]. In addition we compared the proposed approach with the PLS regression.

We tested the method on a number of standard datasets from the *Data for Evaluating Learning in Valid Experiments (Delve)* collection¹. We used eight different data sets which belong to three different dataset families. The first family, called *bank*, describes queues of customers in a series of banks. The other two families *kin* and *pumadyn* were generated by two synthetic robotic arms. Each family contains 8-dimensional and 32-dimensional input spaces and the output space is 1-dimensional. In our experiments we used training sets of size 1000. For each run we used different splits of the data in the training, validating (1000 samples) and testing (1000 samples) datasets. The results for the test set appear in Figure 2.

The MIPR algorithm produces comparable results to the GPR technique using the full input space and in general performs better than other methods. In particular, mutual information linear dimensionality reduction leads to a smaller error than the CCA method followed by the same polynomial regression (the method is referenced in the graph as CCA PR). On the other hand, application of a more elaborate and computationally intensive GPR technique after the MIDR stage does not improve the results.

We next present various performance aspects and a parameter sensitivity analysis of the proposed MIPR algorithm on the 'puma-8nh' dataset. The optimal dimensional reduction was $d = 4$, see Figure 3 (top). The optimal multinomial degree was 3 in all runs, see Figure 3 (bottom). Note that the computational advantage of the MIPR method is significant. For the size 1000 test set the run-

ning time for the proposed scheme was approximately three orders of magnitude faster than the GPR: 0.0021 seconds per run for the method vs. 1.83 seconds per run for the GPR. It is worth pointing out that we compared the running times using our matlab non-optimized code vs. the GPR function from the standard optimized gpml package². The training time for all the methods we examined was comparable: the average training time for our method was 1 min and for GPR it was 6 min. We ran our comparison on a computer with an Intel(R) Xeon(R) 2.67 GHz processor, 3 GB RAM. It is clear that by making a set of routine optimizations even faster execution times can be achieved in the case of the MIPR method and its computational advantage over GPR will be even more significant.

Finally we made numerical experiments in the field of reduced-complexity classification. We evaluated the performance of the MIC algorithm on the standard datasets from the UCI repository [17]. To assess a contribution of the MIDR to the classification accuracy we applied two other mutual information based dimensionality reduction techniques MMIP [8] and MMIFE [9] followed by 1NN classification. In addition we applied LDA algorithm. The dimensionality reduction to two-dimensional subspace was done ($d = 2$). Finally we applied 1NN classification in the full input space to provide a baseline of methods accuracy assessment. To illustrate the ability of the proposed algorithm to utilize fully the information contained in the training set we used a relatively small portion (10%) of data for training and the testing was carried out on the rest of points. The results appear in Figure 4.

The proposed algorithm MIC outperforms other classification schemes based on the dimensionality reduction such as LDA, MMIP, MMIFE for all the datasets. Moreover, for the most of the datasets MIC performs not worse than 1NN classification that benefits from the full input space data. All the above emphasizes the fact that the MeanNN estimator for the entropy (and hence for the mutual infor-

¹<http://www.cs.toronto.edu/~delve>

²<http://www.gaussianprocess.org/gpml/code/>

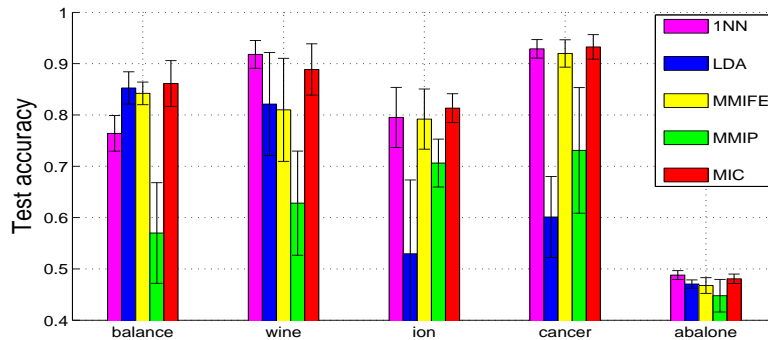


Fig. 4. Comparison of classification test accuracy. UCI classification data sets: balance, wine, ionosphere, cancer (breast) and abalone. Training is 10% of a data set, testing is 90%. Statistics are shown for 100 random repetitions. Reduction to two-dimensional subspace ($d = 2$).

mation) produces a qualitative measure of the differential entropy leading to an optimal linear dimensionality reduction.

7. CONCLUSION

This paper makes several contributions. First, we introduced a supervised linear dimensionality reduction algorithm MIDR based on maximization of mutual information between the subspace of inputs and the outputs values that produces optimal multidimensional result subspaces. We demonstrated a simple nonlinear regression algorithm MIPR that is based on MIDR. The regression method achieves essentially the same performance as the state-of-the-art regression algorithm GPR and in general performs better than other methods. The MIPR algorithm has a significant computational advantage in the test stage, being three orders of magnitude faster than GPR. The MIPR structure lends itself well to fast implementation. Finally, we provided an application for reduced-complexity classification. It leads to superior results compared to the classically employed methods such as LDA. The classification framework emphasizes the advantage MIDR on other mutual information based dimensionality reduction schemes MMIP and MMIFE.

8. REFERENCES

- [1] H. Hotelling, "Relations between two sets of variates," *Biometrika*, pp. 321–377, 1936.
- [2] P. Geladi and B. R. Kowalski, "Partial least-squares regression: a tutorial," *Analytica Chimica Acta*, 185, pp. 1–17, 1986.
- [3] K. Fukumizu, F. Bach, and M. I. Jordan, "Dimensionality reduction for supervised learning with reproducing kernel hilbert spaces," *J. Mach. Learn. Res.*, vol. 5, pp. 73–99, 2004.
- [4] A. Globerson and S. Roweis, "Metric learning by collapsing classes," *Advances in Neural Information Processing Systems 18*, 2006.
- [5] "Distance metric learning for large margin nearest neighbor classification," *Advances in Neural Information Processing Systems 18*, 2006.
- [6] F. Rossi, A. Lendasse, D. François, V. Wertz, and M. Verleysen, "Mutual information for the selection of relevant variables in spectrometric nonlinear modelling," *Chemometrics and Intelligent Laboratory Systems*, pp. 215–226, 2007.
- [7] K. Torkkola, "Feature extraction by non-parametric mutual information maximization," *Journal of Machine Learning Research*, pp. 1415–1438, 2003.
- [8] K.D. Bollacker and J. Ghosh, "Mutual information feature extractors for neural classifiers," *Proc. Int'l Conf. Neural Networks (ICNN '96)*, pp. 1528–1533, 1996.
- [9] J. M. Leiva-Murillo and A. Artes-Rodríguez, "Maximization of mutual information for supervised linear feature extraction," *IEEE Transactions on Neural Networks*, pp. 1433–1441, 2007.
- [10] L. Faivishevsky and J. Goldberger, "ICA based on a smooth estimation of the differential entropy," *Advances in Neural Information Processing Systems 21*, 2009.
- [11] L. Kozachenko and N. Leonenko, "On statistical estimation of entropy of random vector," *Problems Infor. Transmiss.*, vol. 23 (2), pp. 95–101, 1987.
- [12] J. D. Victor, "Binless strategies for estimation of information from neural data," *Physical Review E*, p. 051903, 2002.
- [13] H. Singh, N. Misra, V. Hnizdo, A. Fedorowicz, and E. Demchuk, "Nearest neighbor estimates of entropy," *American Journal of Mathematical and Management Sciences*, pp. 301–321, 2003.
- [14] J. H. Friedman and J. W. Tukey, "A projection pursuit algorithm for exploratory data analysis," *IEEE Transactions on Computers*, pp. 881–890, 1974.
- [15] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annual of Eugenic*, pp. 179–188, 1936.
- [16] C. E. Rasmussen and C. K.I. Williams, *Gaussian Processes for Machine Learning*, The MIT Press, Cambridge, MA, USA, 2006.
- [17] A. Asuncion and D.J. Newman, "UCI machine learning repository," 2007.