Optimal Weight Recovery under Random Projections

Sharan Vaswani
Department of Computer Science
University of British Columbia
sharanv@cs.ubc.ca

Abstract

Random projection has emerged as a powerful technique for dimensionality reduction. It is computationally efficient and has provably good performance under reasonable assumptions. It has consequently been used for a variety of machine learning tasks involving classification, regression and information retrieval. In this paper, we focus on the task of binary classification. There exist a number of theoretical guarantees for classification performance in the low dimensional space. However, the problem of recovering the weight vector in the original space from projected low dimensional space has been studied very recently. In this paper, we empirically evaluate the theoretical bounds derived for weight recovery and study its application for feature selection in high dimensional datasets. We improve the bound on the number of projections required to recover the weights within an ϵ error and outline some promising directions for future work.

1 Introduction

Random projections has proved to be a powerful method for dimensionality reduction. The method involves projecting the original high dimensional data into a lower dimensional space using a random projection matrix. There exist a number of theoretical guarantees (under certain reasonable assumptions on the random matrix used and the number of projections) for the properties in the original space that will be preserved in the projected space. For example, the Johnson-Lindenstrauss lemma [11] derives conditions for preserving pairwise Euclidean distances between points. As compared to PCA, random projections is a computationally efficient way of dimensionality reduction with provably good performance. It has been used in a variety of machine learning tasks such as classification [18] [14], regression [12] and information retrieval [6]. In this paper, we focus on the task of binary classification. A number of theoretical guarantees for classification have been derived [1] [2] [16]. These prove bounds on the classification performance and margin preservation in the projected low dimensional space.However the question whether we can recover the original feature weights from the weights in the low dimensional space has been addressed recently by Zhang et.al [19].

Theoretical guarantees in this context become important for applications like feature selection which use feature weights. This is essential in high dimensional datasets where the number of features is very high as compared to the number of examples. Typical applications involve those in biology such as predicting functional residues in proteins [10] and gene selection for tumour classification [9] [4]. The task of gene selection involves finding from gene expression data a subset of genes which are important for for predicting whether a tumour is malignant or benign. Guyon et.al [9] use support vector machines for this task. A similar feature selection problem arises in high dimensional data from astronomy applications [20].

Guyon et.al summarize a number of feature selection methods in [7]. A common and simple approach for feature selection is to use the weights from a linear classifier to order the features in terms of their importance to the classification problem at hand. [13] [17] use weights from support vector

machines for feature selection. Random projection for dimensionality reduction has been empirically evaluated for practical applications [3]. However to the best of our knowledge, we are the first to empirically evaluate the optimal weight recovery proposed in [19] and explore its use in feature selection for real world applications.

The rest of the paper is structured as follows: Section 2 introduces the notation and presents some of the theorems derived in [19]. We also derive a tighter bound on the number of random projections required to recover the weights to within an ϵ error. In section 3, we present our experiments evaluating the robustness of the weight recovery algorithm and its use in feature selection on two real world datasets. Section 4 concludes our paper and outlines some promising directions for future work.

2 Theoretical Results

2.1 Notation

Let ($\boldsymbol{x_i}$,y_i), i = 1,2....n be the set of training examples where each $\boldsymbol{x_i} \in \mathbb{R}^d$ is the feature vector for the i^{th} example and $y_i \in$ {-1,+1 } is the i^{th} class label. Let $X \in \mathbb{R}^{n \times d} = \{\boldsymbol{x_1}, \boldsymbol{x_2}, ...\}$ be the data matrix consisting of the input examples and let $y \in \mathbb{R}^n$ be the vector of class labels and $\boldsymbol{w} \in \mathbb{R}^d$ represent the weight vector. Let r be the rank of the datamatrix. All the results we present assume a low rank datamatrix X i.e. $r << min(n, d)$. Similar bounds hold for a full rank datamatrix. In the subsequent discussion, we focus on a L2 - regularized cost function. Specifically, we want to minimize the following objective function.

$$\min_{\boldsymbol{w}} \frac{\lambda}{2}||\boldsymbol{w}||^2 + \sum_{i=1}^{n} \ell(y_i x_i \boldsymbol{w}) \tag{1}$$

where $\ell()$ is the loss function (like squared loss) which is a measure of how well the model fits the training data. λ is the regularization parameter and is used to prevent overfitting.

We define the dual optimization problem as

$$\max_{\boldsymbol{\alpha}} -\sum_{i=1}^{n} \ell_*(\alpha_i) - \frac{1}{2\lambda}\alpha^T G \alpha \tag{2}$$

where $\alpha \in \mathbb{R}^n$ is the dual variable and $G = D(y)X^T X D(y)$ is the Gram matrix. ℓ_* is the convex conjugate of l. Let $\boldsymbol{w_*}$ be the optimal solution to the primal problem given by 1 and $\boldsymbol{\alpha_*}$ be the optimal solution to the dual problem defined by equation 2. Zhang et.al in [19] derive the following proposition which connects the optimal primal and dual solutions.

Proposition 1. *Let $\boldsymbol{w_*}$ be the optimal primal solution and $\boldsymbol{\alpha_*}$ be the optimal dual solution. Then,*

$$\begin{aligned} \boldsymbol{w_*} &= -\frac{1}{\lambda} X D(y) \boldsymbol{\alpha_*} \\ \boldsymbol{\alpha_*} &= \nabla \ell(y_i x_i^T \boldsymbol{w_*}) \end{aligned} \tag{3}$$

Let $R \in \mathbb{R}^{d \times m}$ represent the random matrix which projects each high dimensional example $\boldsymbol{x_i} \in \mathbb{R}^d$ to a lower dimension $\hat{\boldsymbol{x_i}} \in \mathbb{R}^m$. Equation 4 describes the projection the i^{th} example.

$$\hat{\boldsymbol{x_i}} = \frac{1}{\sqrt{m}} R^T \boldsymbol{x_i} \tag{4}$$

$\hat{X}$ represents the datamatrix and $\hat{G}$ represents the Gram matrix in the low dimensional space. Ideally we would like $m << d$ and still be able to recover the original weights from the low dimensional space. [19] use a random Gaussian matrix where each entry of the matrix is independently drawn from a standard normal distribution. Let $\boldsymbol{z_*} \in \mathbb{R}^m$ and $\hat{\boldsymbol{\alpha}}_* \in \mathbb{R}^n$ denote the optimal solutions to the primal problem and dual problems in the low dimensional space.

2.2 Naive Recovery

A point $\boldsymbol{x_i}$ is classified by the linear classifier in the original space as $\boldsymbol{w}_*^T x$. In the lower dimensional space, its projection $\hat{\boldsymbol{x_i}}$ is classified as $\boldsymbol{z}_*^T \hat{x}$. Since we want the classification to be consistent in both the spaces, we have from equation 4, $\hat{\boldsymbol{w}} = \frac{1}{\sqrt{m}} R \boldsymbol{z_*}$. This says that we can obtain the high dimensional weights by simply projecting back the low dimensional weights to the original dimension. Henceforth, this method is known as Naive Recovery of weights. Theorem 1 states that naive recovery is a bad approximation of the original weights.

Theorem 1. *For any $0 < \epsilon < 1/3$, with a probability of at least $1 - exp(-(d-r)/32) - exp(-m/32) - \delta$*

$$||\hat{\boldsymbol{w}} - \boldsymbol{w_*}|| \geq \frac{1}{2}\sqrt{\frac{d-r}{m}}\Big(1 - \frac{\epsilon\sqrt{2(1+\epsilon)}}{1-\epsilon}\Big)||\boldsymbol{w_*}||$$

provided m satisfies

$$m \geq \frac{(r+1)log(2r/\delta)}{c\epsilon^2}$$

where $c \geq 1/4$.

2.3 Recovery using Dual Random Projection

We first present an important corollary which is essential for optimal weight recovery using the dual random projection algorithm.

Corollary 1. *Let $A \in \mathbb{R}^{r \times m}$ be a standard Gaussian random matrix. For any $0 < \epsilon \leq 1/2$, with a probability at least $1 - \delta$, we have*

$$\left|\left|\frac{1}{m}AA^T - I\right|\right|_2 \leq \epsilon$$

provided m satisfies

$$m \geq \frac{(r+1)log(2r/\delta)}{c\epsilon^2}$$

where $c = 2 - \sqrt{3}$.

We use equation 4 to make the following observation regarding the Gram matrix in the low dimensional space.

$$\hat{G} = D(y)\hat{X}\hat{X}^T D(y) \tag{5}$$

$$\hat{G} = D(y)X^T \frac{RR^T}{m} X D(y) \tag{6}$$

Given that m satisfies the condition in corollary 1 we have E[$\frac{RR^T}{m}$] $= I$. This condition holds for other matrices as well. For example, a Bernoulli ensemble matrix with equally probable random {-1, +1} entries also satisfies E[$\frac{RR^T}{m}$] $= I$ and in principle can be used the random projection matrix. This condition coupled with equation 6 implies that $\hat{G} = G$ in expectation. Hence from the dual formulation given in 2, we have in expectation that $\hat{\boldsymbol{\alpha}}_* = \boldsymbol{\alpha}_*$. Thus, the solutions to the dual problem in both spaces are close to each other. We use this fact to recover the weight vector $\tilde{\boldsymbol{w}}$. We now present the dual random projection algorithm.

- Use a random Gaussian / Bernoulli ensemble matrix (R) to project data to low dimensional space.
- Solve the primal classification problem in the low dimensional space and obtain the weights $\boldsymbol{z}_*$.
- Use the relation in proposition 1 to obtain the equivalent dual solution $\hat{\boldsymbol{\alpha}}_*$
- From the previous argument, we know that $\hat{\boldsymbol{\alpha}}_* \approx \boldsymbol{\alpha}_*$. Use the relation in proposition 1 to obtain the primal solution $\tilde{\boldsymbol{w}}$ in the original space.

Theorem 2 gives precise bounds on quality of the solution obtained from the above algorithm.

Theorem 2. *Let $\boldsymbol{w}_*$ be the optimal solution and let $\tilde{\boldsymbol{w}}$ be the solution recovered by the dual random projection algorithm. For any $0 < \epsilon < 1/2$, with a probability at least $1 - \delta$, we have*

$$||\tilde{\boldsymbol{w}} - \boldsymbol{w}_*|| \leq \frac{\epsilon}{1-\epsilon}||\boldsymbol{w}_*||$$

provided

$$m \geq \frac{(r+1)log(2r/\delta)}{c\epsilon^2}$$

where $c \geq 1/4$.

2.4 Tighter bounds for dual random projection

Zhang et.al prove a $\mathcal{O}$(r log r) bound on the number of projections needed to achieve a certain fixed error. Corollary 1 is the key to this proof. We prove an improved version of corollary 1 for any random matrix having entries drawn independently from a distribution with mean zero and unit variance. For this, we use a theorem about the asymptotic properties of the maximum singular value of a random matrix. This theorem was first derived by Geman [5].We first state this theorem as it appears in the survey [15].

Theorem 3 (Theorem 2.1 in [15])**.** *Let $A = A_{N,n}$ be an $N \times n$ random matrix whose entries are independent copies of some random variable with zero mean, unit variance and finite fourth moment. Suppose that the dimensions N and n grow to infinity while the aspect ratio n/N converges to some number $y \in (0, 1]$. Then*

$$\frac{1}{\sqrt{N}}s_{min}(A) \to 1 - \sqrt{y}, \qquad \frac{1}{\sqrt{N}}s_{max}(A) \to 1 + \sqrt{y}$$

almost surely. Moreover, without the fourth moment assumption the sequence $\frac{1}{\sqrt{N}}s_{max}(A)$ is almost surely unbounded.

We use Theorem 3 to prove a corollary which proves a $\mathcal{O}$(r) bound on the number of projections required to achieve a fixed error ϵ in the weights. A proof similar to that for Theorem 2 follows directly from this corollary.

Corollary 2. *Let $A \in \mathbb{R}^{r \times m}$ be a matrix whose entries are independently sampled from a distribution with zero mean and unit variance and has a finite fourth order moment. For any $0 < \epsilon < 1$, we have*

$$\left|\left|\frac{1}{m}AA^T - I\right|\right|_2 \leq \epsilon$$

provided

$$m \geq \frac{3r}{\epsilon}$$

Proof. Let $B = \frac{1}{m}A^TA - I$. We need to find conditions on m such that $||B||_2 \leq \epsilon$. Let s_{max} be the maximum singular value for A and let λ_{max} be its maximum eigenvalue. We know that,

$$||A||_2 = s_{max}(A) = s_{max}(A^T) = \sqrt{\lambda_{max}(A^TA)}$$

We have the following relation,

$$||B||_2 = \lambda_{max}(B) = \frac{1}{m}\lambda_{max}(A^TA) - 1$$

$$||B||_2 = \frac{1}{m}[s_{max}(A^T)]^2 - 1$$

$$||B||_2 = \Big[\frac{1}{\sqrt{m}}s_{max}(A^T)\Big]^2 - 1 \leq \epsilon$$

$A^T \in \mathbb{R}^{m \times r}$ satisfies the properties for Theorem 3. Hence, as $r \to \infty$ and $m \to \infty$, $r/m \to y$ such that $y \in (0, 1]$. From theorem 3 and the above relation.

$$(1 + \sqrt{y})^2 - 1 \leq \epsilon$$

After some algebra, we have $y \leq \frac{\epsilon}{3}$. Hence $m \geq \frac{3r}{\epsilon}$. □

2.5 Iterative Recovery

From Theorem 2, the relative error can be reduced by a constant factor by one iteration of the dual random projection algorithm. This suggests we can develop an algorithm which iteratively reduces the error. [19] propose such an algorithm and prove its correctness.For the sake of brevity, we skip the details of this algorithm.

3 Experiments

In this section, we present our results for the empirical evaluation of the dual random projection algorithm. For our experiments, we use the squared loss function as defined in equation 7

$$\ell(z) = \frac{1}{2}(1-z)^2 \tag{7}$$

Hence our primal objective function defined in equation 1 can be rewritten as

$$\min_{\boldsymbol{w}} \frac{\lambda}{2}||\boldsymbol{w}||^2 + \frac{1}{2}\sum_{i=1}^{n}(y_i - x_i^T\boldsymbol{w})^2 \tag{8}$$

To follow the dual random projection algorithm, we need expressions to find the primal optimal solution and to convert between the primal-dual solutions. For the squared loss function, the optimal primal solution is given by equation 9.

$$\boldsymbol{w_*} = X(\lambda I + X^T X)^{-1} y \tag{9}$$

We can convert between primal and dual solutions using equations 10

$$\begin{aligned} \boldsymbol{\alpha} &= D(y)X^T\boldsymbol{w} - 1 \\ \boldsymbol{w} &= -\frac{1}{\lambda}XD(y)\alpha \end{aligned} \tag{10}$$

Analogous expressions hold for the low dimensional space.

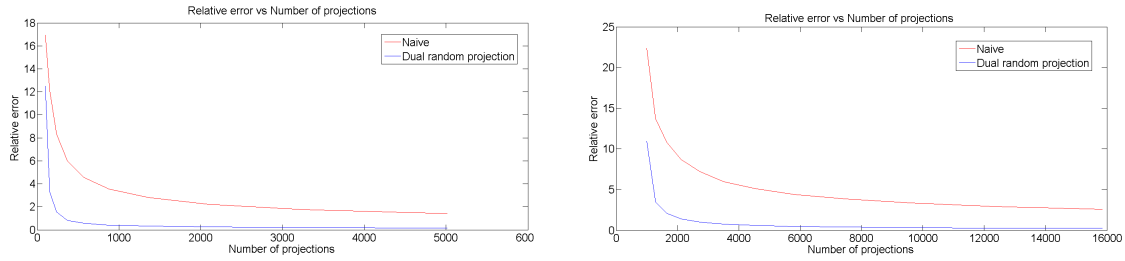
3.1 Datasets

We use the ARCENE and DOROTHEA datasets [8] which were used for the NIPS 2003 feature selection challenge. Both datasets have full rank and possess a greater number of features than input examples. For both these datasets, half of the features are true features whereas half of the features have been artificially added as probes and have no predictive power in the classification task. To conduct experiments to test the dependence on the rank of the datamatrix, we use low rank approximations of the data. The task for ARCENE dataset is to differentiate between cancer and normal patterns based on the mass-spectrometric data. The dataset consists of 10000 features with 100 instances. The DOROTHEA dataset is a drug discovery dataset where the task is to distinguish between active and inactive chemical compounds based on their structural molecular features. This dataset consists of 100000 features with 800 instances.

3.2 Experiment 1

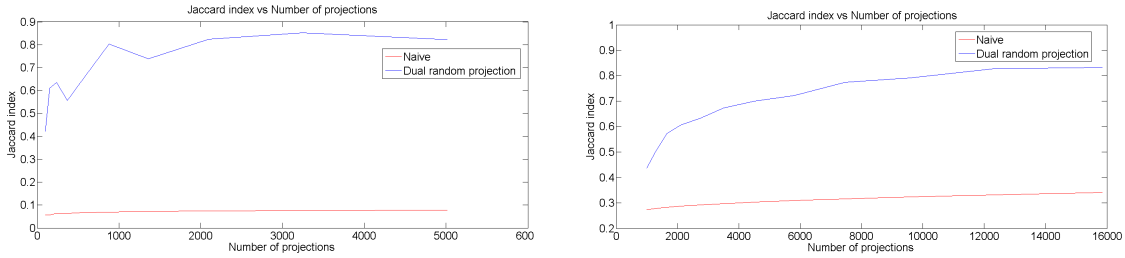
We reconstruct the original weight vector using both naive recovery and the dual random projection methods. We measure the relative error with respect to the true weight vector obtained by solving the problem in the high dimensional space. We evaluate the dependence between the relative error and the number of projections. The experiments use the random Gaussian matrix. Using the Bernoulli ensemble matrix leads to similar results. For all the experiments, we fix the regularization parameter λ as 1, use LBFGS to minimize the primal objective function and average the results across 10 runs. Figures 1(a) and 1(b) present plots for both datasets with full rank.

The weights of irrelevant features are forced to zero because of the L2 regularization. Hence the features with nonzero weights are predictive for the classification task. For feature selection , we need to find the predictive features and are not concerned with the absolute values of the weights. Hence we propose to use another metric to measure efficiency of our weight recovery. We calculate



(a) (b)

Figure 1: Relative error vs number of random projections for (a)ARCENE and (b) DOROTHEA dataset. The weights recovered using the dual random projection method are more accurate than those recovered by the naive method.



(a) (b)

Figure 2: Jaccard index vs number of random projections for (a)ARCENE and (b) DOROTHEA dataset. As expected, as the number of random projections increases, the jaccard index improves and saturates around 0.8 for both the datasets at relatively fewer number of projections

the Jaccard index between sets of features with non-zero weights in the original and recovered space. Figures 2(a) and 2(b) show these plots for the two datasets.

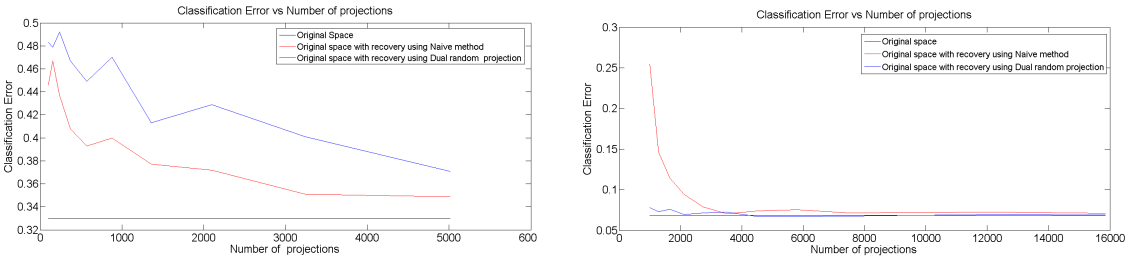
We see that the dual random projection algorithm is able to find the correct subset of important features even when the number of projections is relatively less. This implies that it is possible to quickly solve the problem in the low dimensional space and recover the weights and still find relevant features with reasonable accuracy.

The next set of figures 3(a) and 3(b) show that we get a good classification performance for both recovery methods. The weights are recovered using the two methods and then used to evaluate the classification performance on the test set. [19] proves this fact and reconciles their results with the existing guarantees on classification performance in the projected space. This proves that the conditions for optimal weight recovery subsume those for good classification performance.

Figures 4(a) and 4(b) show the time required for classification in the original space and sum of the times for classification in the projected low dimensional space and weight recovery for each of the methods. These results proves that dimensionality reduction coupled with appropriate weight recovery can lead to reduction in classification time. This becomes important especially for large high dimensional datasets.

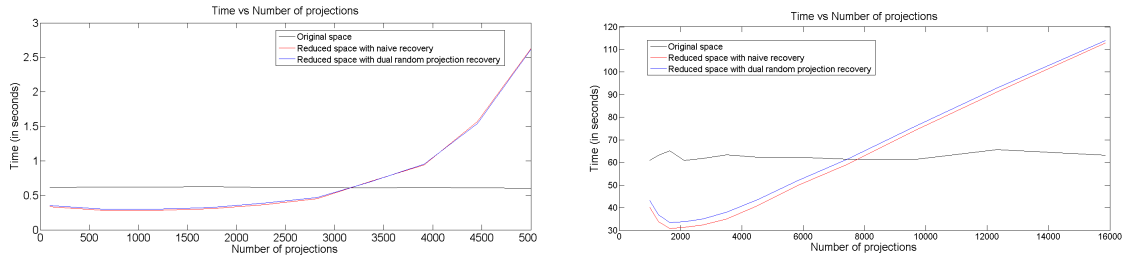
3.3 Experiment 2

For the subsequent experiments, we show results only for the ARCENE dataset. The results for the DOROTHEA dataset are similar. The objective of this experiment is used to evaluate the dependence of the weight recovery error on the rank of the matrix. Figure 5(a) shows a linear dependence of the recovery error on rank for a fixed number of projections. This agrees with the our improved bounds presented in the previous section.



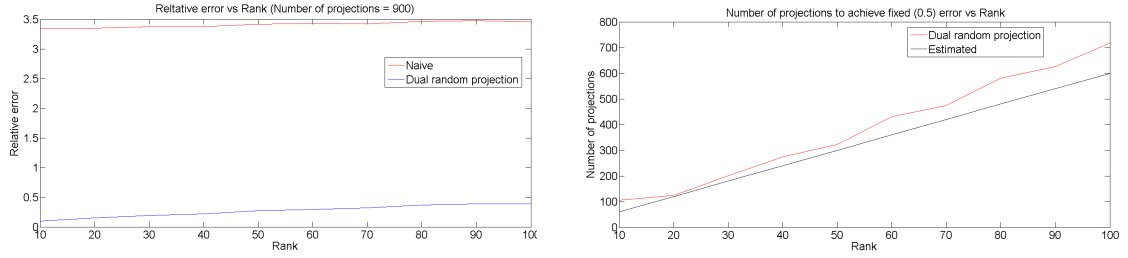
(a) (b)

Figure 3: Classification Error vs number of random projections for (a)ARCENE and (b) DOROTHEA dataset. The results show good classification performance for both recovery methods.



(a) (b)

Figure 4: Time taken for classification vs number of random projections for (a)ARCENE and (b) DOROTHEA dataset. The time taken for classification and weight recovery is smaller than the classification time in the original space when the number of projections is around 3000 for the ARCENE and 7000 for the DOROTHEA dataset.The error in weight recovery with these number of random projections is sufficiently low and the jaccard index is almost maximum. As the number of random projections increases further, the weight recovery time increases and it is more efficient to solve the problem in the original space.



(a) (b)

Figure 5: (a)Recovery error vs rank of the datamatrix with the number of projections fixed at 900.(b) Number of projections required to achieve a fixed error (0.5) vs rank of the matrix. This dependence is linear as estimated by our improved bound. The original estimate is much higher than the number of projections required in practice.

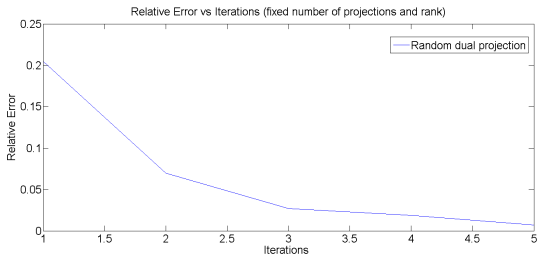


Figure 6: Relative error vs number of iterations of the dual random projection method. The number of projections is fixed at 2500 and the datamatrix is of full rank.

3.4 Experiment 3

This experiment verifies the dependence on the number of projections required to achieve a constant error with increasing rank of the data matrix. The original proof in [19] derives this dependence to be $\mathcal{O}$(r log(r) where r is the rank of the matrix. In section 2 we improve this bound to a linear dependence between the number of projections and rank. Figure 5(b) provides empirical proof of this fact. Hence, we are able to empirically verify the correctness of our bound.

3.5 Experiment 4

The last set of experiments verify the iterative reduction in error by the dual random projection algorithm. Figure 3.5 shows this result. We can in principle go beyond 5 iterations to reduce the error further until round-off errors start to dominate. However, we lose the gain in time required for classification if we use a larger number of iterations.

4 Conclusion and Future Work

In this paper, we empirically evaluate the dual random projection algorithm for optimal weight recovery proposed in [19]. We see that it is possible to recover the weights with a sufficiently low error using only a relatively small number of random projections. We conclude that dimensionality reduction coupled with the proposed algorithm can be used for feature selection and lead to gains in classification time especially for large high-dimensional datasets. We prove a tighter bound on the number of projections required to achieve a certain error and empirically verify the bound. We also verify it is possible to iteratively reduce the weight recovery error using the dual random projection method.

Since support vector machine have been proved to the most efficient classifiers for feature selection [13], we plan to extend the formalism to handle weight recovery using support vector machine as the classifier. L1 regularization is a more effective way of enforcing sparsity in the feature weights and hence more useful for feature selection. We plan to devise a similar algorithm with L1 regularization on the weights. Since it is the order and number of non-zero weights which affect the feature selection rather than the exact weight values, we can relax the problem and derive conditions such that the order of the weight values are preserved in the low dimensional space.

References

[1] Rosa I Arriaga and Santosh Vempala. An algorithmic theory of learning: Robust concepts and random projection. In *Foundations of Computer Science, 1999. 40th Annual Symposium on*, pages 616–623. IEEE, 1999.

[2] Maria-Florina Balcan, Avrim Blum, and Santosh Vempala. Kernels as features: On kernels, margins, and low-dimensional mappings. *Machine Learning*, 65(1):79–94, 2006.

[3] Ella Bingham and Heikki Mannila. Random projection in dimensionality reduction: applications to image and text data. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 245–250. ACM, 2001.

[4] Terrence S Furey, Nello Cristianini, Nigel Duffy, David W Bednarski, Michel Schummer, and David Haussler. Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 16(10):906–914, 2000.

[5] Stuart Geman et al. A limit theorem for the norm of random matrices. *The Annals of Probability*, 8(2):252–261, 1980.

[6] Navin Goel, George Bebis, and Ara Nefian. Face recognition experiments with random projection. In *Defense and Security*, pages 426–437. International Society for Optics and Photonics, 2005.

[7] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *The Journal of Machine Learning Research*, 3:1157–1182, 2003.

[8] Isabelle Guyon, Steve R Gunn, Asa Ben-Hur, and Gideon Dror. Result analysis of the nips 2003 feature selection challenge. In *NIPS*, volume 4, pages 545–552, 2004.

[9] Isabelle Guyon, Jason Weston, Stephen Barnhill, and Vladimir Vapnik. Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1-3):389–422, 2002.

[10] Chris Haddow, Justin Perry, Marcus Durrant, and Joe Faith. Predicting functional residues of protein sequence alignments as a feature selection task. *International journal of data mining and bioinformatics*, 5(6):691–705, 2011.

[11] William B Johnson and Joram Lindenstrauss. Extensions of lipschitz mappings into a hilbert space. *Contemporary mathematics*, 26(189-206):1, 1984.

[12] Odalric-Ambrym Maillard and Rémi Munos. Linear regression with random projections. *The Journal of Machine Learning Research*, 13(1):2735–2772, 2012.

[13] Dunja Mladenić, Janez Brank, Marko Grobelnik, and Natasa Milic-Frayling. Feature selection using linear classifier weights: interaction with classification models. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 234–241. ACM, 2004.

[14] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *NIPS*, volume 3, page 5, 2007.

[15] Mark Rudelson and Roman Vershynin. Non-asymptotic theory of random matrices: extreme singular values. *arXiv preprint arXiv:1003.2990*, 2010.

[16] Qinfeng Shi, Chunhua Shen, Rhys Hill, and Anton van den Hengel. Is margin preserved after random projection? 2012.

[17] Alexander Statnikov, Douglas Hardin, and Constantin Aliferis. Using svm weight-based methods to identify causally relevant and non-causally relevant variables. *sign*, 1:4, 2006.

[18] Santosh S. Vempala. *The random projection method*, volume 65. American Mathematical Soc., 2004.

[19] Lijun Zhang, Mehrdad Mahdavi, Rong Jin, Tianbao Yang, and Shenghuo Zhu. Recovering the optimal solution by dual random projection. In *Conference on Learning Theory*, pages 135–157, 2013.

[20] Hongwen Zheng and Yanxia Zhang. Feature selection for high-dimensional data in astronomy. *Advances in Space Research*, 41(12):1960–1964, 2008.