An efficient high dimensional supervised feature selection method based on non-negative matrix factorization

Yi-Hung, Huang[1]
[1] Institute of Information Science
Academic Sinica
Taipei 106, Taiwan
Email: ivano@iis.sinica.edu.tw

Chun-Nan, Hsu[1,2]
[2] Information Science Institute
University of Southern California
Marina del Rey, CA
Email: chunnan@iis.sinica.edu.tw

Abstract — **Feature selection is helpful to extract the important information from a given dataset. In addition to identify the significant features, to extract the mutual information between features is also an important issue in feature selection. Non-negative matrix factorization has been proven capable to extract the mutual information between features. In this paper, we introduce an efficient supervised feature selection method based on NMF.**

Keywords*: *feature selection*, *non-negative matrix facotrizaiton*, *feature extraction*, *feature ranking*, *Fisher discriminant analysis

I. INTRODUCTION

Feature selection is still one of the popular topics in the machine learning in these years. In the feature selection, it is important to identify the most significant features. However, besides the most significant features identification, we are more concerned about identifying the potentially relevant features.

There are two examples to illustrate the importance of identifying the potentially relevant features. One is the text mining problem, and the other one is a critical issue in bioinformatics. In the text mining problem, each document is represented as a vector of keyword frequency. We would like to know that which set of keywords are potentially relevant for corresponding to a specific category label. In the bioinformatics, each case is represented as a vector of gene expression. It is important to identify a set of highly correlated genes which are concerned with a specific phenotype.

For solving this kind of problem, Lee and Seung point out that non-negative matrix factorization (NMF) is capable to extract the mutual information between features from a data matrix [1]. However, as the dimension of data matrix increasing, the mutual information will become weaker. For this reason, we introduce an efficient supervised feature selection method based on NMF in this paper. We will also establish two kinds of simulated data to show that our method is able to identify the potentially relevant features in a high dimensional data matrix.

This paper is organized as follows; the popular NMF algorithm will be presented on section II. Then, we will introduce our supervised NMF on section III. Finally, the performance on two kinds of simulated data is discussed on section IV.

II. NONEGATIVE MATRIX FACTORIZATION

Non-negative matrix factorization (NMF) algorithm is one of the popular matrix decomposition methods. It can factorize a data matrix $V \in R^{m \times n}$ as the product of two smaller matrixes $W \in R^{m \times k}$ and $H \in R^{k \times n}$, where m is the dimension of feature values, n is the number of instances, and the k is a pre-specified value.

$$V \approx WH \tag{1}$$

W is used to characterize the coordinated system and each column of W is represented as the coordinates of the feature space. And H is used to characterize the distribution of the data matrix in the feature space and each row of H is represented as an original case projection in the feature space.

NMF is focus on finding the suitable W and H through minimizing Euclidean distance between V and WH under the constraint that each element of W and H is non-negative.

$$\min_{W,H} \left\| V - WH \right\|^2 = \frac{1}{2} tr((V - WH)^T (V - WH)) \quad \text{subject to } W \geq 0, H \geq 0 \tag{2}$$

In order to solve this optimization problem, Lee and Seung provide the two notable updating algorithms [2]. One is the multiplicative update algorithm and the other is the additive update algorithm. Because we are more concerned about their additive update algorithm, it is introduced as follow,

$$\begin{cases} H' = H - \eta(W^T WH - W^T V) \\ W' = W - \eta(WHH^T - VH^T) \end{cases}, \tag{3}$$

, where η is denoted as the learning rate which is a prespecific value. The additive update algorithm is based on the gradient descent method and W and H are alternately updating.

III. SUPERVISED NON-NEGATIVE MATRIX FACTORIZATION

In this section, we introduce a supervised NMF algorithm which is the standard object function cooperated with the supervised part. Moreover, the one vs. all strategy is adapted to solve the multiclass problem. According to Fisher

discriminate analysis, we hope that the supervised part is capable of increasing the distance between different classes and decreasing the variance within the classes in the feature space. For representing the distance between differrent classes, we design a weight vector $\alpha \in R^{n\times 1}$ and each element α_i is,

$$\alpha_i = \begin{cases} \frac{1}{n_1} & \text{if instance } i \in \text{class+} \\ \frac{-1}{n_2} & \text{if instance } i \in \text{class}- \end{cases}, \quad (4)$$

, where n_1 is denoted as the number of the class+ and n_2 is denoted as the number of the class–. The difference of class+ and class– in the feature space can be represented as $H\alpha$.

$$(H\alpha)^T = [\mu_{H_1}^+ - \mu_{H_1}^-, \mu_{H_2}^+ - \mu_{H_2}^-, \ldots, \mu_{H_k}^+ - \mu_{H_k}^-] \quad (5)$$

, where μ_{Hi}^+/μ_{Hi}^- is represented as the mean of element i of class+/– and k is the dimension of feature space. $(H\alpha)^T(H\alpha)$ is thus equal to the distance between different classes.

$$(H\alpha)^T(H\alpha) = \sum_{i=1}^{k}(\mu_{H_i}^+ - \mu_{H_i}^-)^2 \quad (6)$$

For representing the variance within classes, we design a matrix $D \in R^{n\times n}$ and each element D_{ij} of D is as follow,

$$D_{ij} = \begin{cases} \frac{1}{n_1} & \text{if instance } i \in \text{class+ and instance } j \in \text{class+} \\ \frac{1}{n_2} & \text{if instance } i \in \text{class}- \text{ and instance } j \in \text{class}- \\ 0 & \text{if the others} \end{cases} \quad (7)$$

The HD is represented as the mean-matrix. Furthermore, tr((*H*–*HD*)^T(*H*–*HD*)) is proportional to the variance within the classes.

$$tr((H-HD)^T(H-HD)) \propto \sum_{i=1}^{k}(\text{var}_i^+ + \text{var}_i^-) \quad (8)$$

After the supervised part has been introduced, the supervised NMF cost function can be written as follow,

$$\min_{W,H} \frac{1}{2}\|V-WH\|^2 - \gamma\sum_{i=1}^{k}(\mu_{H_i}^+ - \mu_{H_i}^-)^2 + \lambda\sum_{i=1}^{k}(\text{var}_i^+ + \text{var}_i^-)$$
$$\text{subject to } W \geq 0, H \geq 0 \quad (9)$$

, where γ and λ are denoted as the learning rates of the supervised part. We then implement the gradient descent method to derive the updating rules for W and H in (10).

$$\begin{cases} H' = H - \eta(W^TWH - W^TV - 2\gamma H\alpha\alpha^T + 2\lambda(H - 2HD + HDD^T)) \\ W' = W - \eta(WHH^T - VH^T) \end{cases} \quad (10)$$

Since the updating way is the matrix-by-matrix multiplication, it is more efficient than the existing pointwise updating rules introduced in [3]. Besides, the columns of W become meaningful through project the original data to the desired feature space H. At the end of training, the sort value of each column in W can be used to rank the features.

IV. EXPERIMENT

We now present two experiments for evaluating the supervised NMF method. In the first experiment, we random generate an 800 × 1000 data matrix. In the data matrix, each column indicates the feature value and each row indicates the instance. The values in each column are randomly permuted as 50%: 1 and 50%: 0. The first 400 instances are denoted as the class+ and the other 400 instances are the class–. We then compare our results with the Fisher score introduced in [4]. The Fisher score is used to rank the single significant for each feature value. We can find that features with highly Fisher score are distributed on the two ends of the sort value of each column in W in Fig. 1.

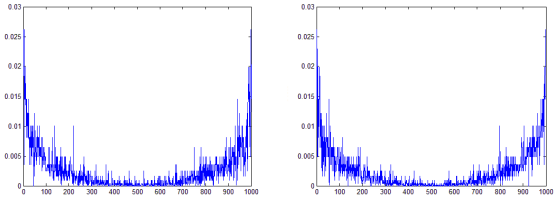

Fig. 1. The x-axis is the sort value of each column in W and the y-axis is the corresponding Fisher score. In this experiment, the value k is set as 2. The left figure is represented as the sort value of 1st column in W and the right figure is represented as the sort value of 2nd column in W.

In the second experiment, the data matrix is constructed similar to the first experiment. However, the size of the data matrix is expanded to 800 × 100,000 and we modify that the first 3 columns are duplicates of next 3 columns. For this reason, the first 6 features are the predefined potentially relevant features. In each instance, if the sum of the first 3 features is equal to i, then this instance is label as class i, there are total 4 classes in this data matrix. The value k is set as 4 in the training. At the end of supervised NMF training, each column of W is normalized to [0, 1].

Table 1. The normalized *W* matrix.

	row1	row2	row3	row4	row5	row6	row7 ~ row 100000
column1	1.000	1.000	1.000	1.000	1.000	1.000	mean: 0.843 variance: 0.011
column2	1.000	1.000	1.000	1.000	1.000	1.000	mean: 0.863 variance: 0.012
column3	0.485	0.490	0.528	0.485	0.490	0.528	mean: 0.330 variance: 0.034
coulmn4	1.000	1.000	1.000	1.000	1.000	1.000	mean: 0.848 variance: 0.014

In this table, it is show that the weight values of relevant feature are the highest in 3 columns of W and the others are distributed on the lower range with a very small variance. It reveals that our supervised NMF is able to identify 6 relevant features from 100,000 features. However, an appropriate learning rate is needed and we will develop an adaptive learning rate in the next work.

REFERENCES

[1] D. D. Lee and H. S. Seung. "Learning the parts of objects with non-negative matrix factorization", *Nature*, vol. 401, pp. 788-791, 1999.

[2] D. D. Lee and H. S. Seung. "Algorithms for non-negative matrix factorization", *NIPS*, pp. 556-562, 2001.

[3] D. Kim, S. Y. Lee, and S. Amari, "Representative and discriminate feature extraction based on NMF for emotion recognition in speech", *Lecture Notes in Computer Science*, vol. 5863, pp. 649-656, 2009.

[4] Y. W. Chang and C. J. Lin. "Feature ranking using linear SVM", *JMLR*, pp. 53-64, 2008.