Int. J. Advance Soft Compu. Appl, Vol. 10, No. 2, July 2018
ISSN 2074-8523

The combination of Sparse Principle Component Analysis and Kernel Ridge Regression methods applied to speech recognition problem

Loc Hoang Tran, Linh Hoang Tran

Thu Dau Mot University
e-mail: tran0398@umn.edu
e-mail: linhtran.cntt@tdmu.edu.vn

Abstract

Speech recognition is the important problem in pattern recognition research field. In this paper, the combination of the Sparse Principle Component Analysis method and the kernel ridge regression method will be applied to the MFCC feature vectors of the speech dataset available from IC Design lab at Faculty of Electricals-Electronics Engineering, University of Technology, Ho Chi Minh City. Experiment results show that the combination of the Sparse Principle Component Analysis method and the kernel ridge regression method outperforms the current state of the art Hidden Markov Model method and the kernel ridge regression method alone in speech recognition problem in terms of sensitivity performance measure.

Keywords: *kernel ridge regression, HMM, speech recognition, MFCC, PCA, Sparse PCA.*

1 Introduction

In this paper, we will present the kernel ridge regression method and apply this method to automatic speech recognition problem. To the best of our knowledge, this work has not been investigated. Researchers have worked in automatic speech recognition for almost six decades. The earliest attempts were made in the 1950's. In the 1980's, speech recognition research was characterized by a shift in technology from template-based approaches to statistical modeling methods, especially Hidden Markov Models (HMM). Hidden Markov Models (HMM) have been the core of most speech recognition systems for over a decade and is

considered the current state of the art method for automatic speech recognition system [1]. Second, to classify the speech samples, a graph (i.e. kernel) which is the natural model of relationship between speech samples can also be employed. In this model, the nodes represent speech samples. The edges represent for the possible interactions between nodes. Then, machine learning methods such as Support Vector Machine [2], kernel ridge regression [3], Artificial Neural Networks [4], or nearest-neighbor classifiers [5] can be applied to this graph to classify the speech samples. The nearest-neighbor classifiers method labels the speech sample with the label that occurs frequently in the speech sample's adjacent nodes in the network. Hence neighbor counting method does not utilize the full topology of the network. However, the Artificial Neural Networks, Support Vector Machine, kernel ridge regression, and graph based semi-supervised learning methods utilize the full topology of the network. Moreover, the Artificial Neural Networks, kernel ridge regression, Support Vector Machine are supervised learning methods. Please note that the kernel ridge regression method is the simplest form of the Support Vector Machine method.

While nearest-neighbor classifiers method, the Artificial Neural Networks, and the graph based semi-supervised learning methods are all based on the assumption that the labels of two adjacent speech samples in graph are likely to be the same, SVM and kernel ridge regression methods do not rely on this assumption. Graphs used in nearest-neighbor classifiers method, Artificial Neural Networks, and the graph based semi-supervised learning method are very sparse. However, the graph (i.e. kernel) used in SVM and kernel ridge regression methods is fully-connected.

In the last two decades, the SVM learning method has successfully been applied to some specific classification tasks such as digit recognition, text classification, and protein function prediction and automatic speech recognition problem [2]. However, the kernel ridge regression method (i.e. the simplest form of the SVM method) has not been applied to any practical applications. Hence in this paper, we will use the kernel ridge regression method applied to the automatic speech recognition problem as the baseline method.

Next, we will introduce the Principle Component Analysis. Principle Component Analysis (i.e. PCA) is one of the most popular dimensionality reduction techniques [6]. It has several applications in many areas such as pattern recognition, computer vision, statistics, and data analysis. It employs the eigenvectors of the covariance matrix of the feature data to project on a lower dimensional subspace. This will lead to the reduction of noises and redundant features in the data and the low time complexity of the Kernel Ridge Regression approach solving speech recognition problem.

In detail, PCA method convert the original set of features to a different and more compact representation keeping as much information as possible and to try to increase the performance of the Kernel Ridge Regression approach, especially the accuracy of the Kernel Ridge Regression approach. The dimensional reduction stage is achieved by retaining only the relevant dimensions according to one

specific criteria which is maximizing the variance. This stage helps solve the problem called the curse of dimensionality. Therefore, reducing the dimensionality of the dataset is the most direct way solving the problems caused by high dimensionalities.

However, the PCA has two major disadvantages which are the lack of sparsity of the loading vectors and each principle component is the linear combination of all variables. From data analysis viewpoint, sparsity is necessary for reduced computational time and better generalization performance. From modeling viewpoint, although the interpretability of linear combinations is usually easy for low dimensional data, it could become much harder when the number of variables becomes large. To overcome this hardness and to introduce sparsity, many methods have been proposed such as [7,8,9,10].

In this paper, we will introduce new approach for sparse PCA using Alternating Direction Method of Multipliers (i.e. ADMM method) [11]. Then, we will try to combine the sparse PCA dimensional reduction method and the Kernel Ridge Regression method and applied this combination approach to the speech recognition problem. This work, to the best of our knowledge, has not been investigated.

We will organize the paper as follows: Section II will present the Alternating Direction Method of Multipliers. Section III will derive the sparse PCA method using the ADMM method in detail. Section IV will present the sparse PCA algorithm. Section V will introduce kernel ridge regression algorithms in detail. Section VI will present the detailed derivation of the kernel ridge regression method. In section VII, we will apply this combination of sparse PCA algorithm and Kernel Ridge Regression algorithm to speech samples available from the IC Design lab at Faculty of Electricals-Electronics Engineering, University of Technology, Ho Chi Minh City. Section VIII will conclude this paper and discuss the future directions of researches of this automatic speech recognition problem.

2 Alternating Direction Method of Multipliers

In this section, we will introduce the Alternating Direction Method of Multipliers. The detailed information about the Alternating Direction Method of Multipliers can be found in [10]. First, assume that we want to solve the following problem:

$$minimize f(x) + g(z) \tag{1}$$

$$subject\ to\ Ax + Bz = c \tag{2}$$

with variables $x \in R^n$ and $z \in R^m$, where $A \in R^{p*n}, B \in R^{p*m}$.

Next, we will form the augmented Lagrangian

$$L_\rho(x,z,y) = f(x) + g(z) + y^T(Ax + Bz - c) + \frac{\rho}{2}||Ax + Bz - c||_2^2 \quad (3)$$

Finally, $x^{k+1}, z^{k+1}, \text{ and } y^{k+1}$ can be solved as the followings

$$x^{k+1} = argmin_x L_\rho(x, z^k, y^k) \quad (4)$$

$$z^{k+1} = argmin_z L_\rho(x^{k+1}, z, y^k) \quad (5)$$

$$y^{k+1} = y^k + \rho(Ax^{k+1} + Bz^{k+1} - c), \quad (6)$$

where $\rho > 0$.

3 Sparse Principle Component Analysis Derivation

Assume that we are given the data matrix $X \in R^{p*n}$. Next, we will formulate our sparse PCA problem. This problem is in fact the following optimization problem:

$$minimize_{v,z}||X - \sigma uv^T||_F^2 + \frac{\mu}{2}||v||_2^2 + \lambda||z||_1 \quad (7)$$

$$such\ that\ v = z, \quad (8)$$

where σ, u, v are the singular value, the left singular vector, and the right singular vector of the Singular Value Decomposition (i.e. SVD) of X respectively. Information about the SVD and its relationship to PCA can be found in [6]. In the above optimization problem, $\sigma\ and\ u$ are fixed. Our objective is to find the sparse loading vectors v.

First, the augmented Lagrangian of the above optimization problem can be derived as the following:

$$L_\rho(x,z,y) = ||X - \sigma uv^T||_F^2 + \frac{\mu}{2}||v||_2^2 + \lambda||z||_1 + y^T(v - z) + \frac{\rho}{2}||v - z||_2^2 \quad (9)$$

Then $v^{k+1}, z^{k+1}, and\ y^{k+1}$ can be solved as the followings:

$$v^{k+1} = argmin_v L_\rho(v, z^k, y^k) \quad (10)$$

Hence

$$\frac{dv^{k+1}}{dv} = \frac{d}{dv}(||X - \sigma uv^T||_F^2 + \frac{\mu}{2}||v||_2^2 + y^{kT}(v - z^k) + \frac{\rho}{2}||v - z^k||_2^2) \quad (11)$$

$$= \frac{d}{dv}(\sum_{i,j}(X_{ij} - (\sigma uv^T)_{ij})^2 + \frac{\mu}{2}||v||_2^2 + y^{kT}(v - z^k) + \frac{\rho}{2}||v - z^k||_2^2) \quad (12)$$

$$= \frac{d}{dv}(\sum_{i,j}(X_{ij}^2 - 2X_{ij}(\sigma uv^T)_{ij} + (\sigma uv^T)_{ij}^2) + \frac{\mu}{2}||v||_2^2 + y^{kT}(v - z^k) + \frac{\rho}{2}||v - z^k||_2^2) \quad (13)$$

$$= \frac{d}{dv}\left(||X||_F^2 - 2\sum_{ij} X_{ij}(\sigma u v^T)_{ij} + ||\sigma u v^T||_F^2 + \frac{\mu}{2}||v||_2^2 + y^{kT}(v - z^k) + \frac{\rho}{2}||v - z^k||_2^2 \right) \quad (14)$$

$$= \frac{d}{dv}(-2\sigma \sum_j \sum_i X_{ij} u_i v_j + \sigma^2 trace(vv^T) + \frac{\mu}{2}||v||_2^2 + y^{kT}(v - z^k) + \frac{\rho}{2}||v - z^k||_2^2)$$

$$= \frac{d}{dv}(-2\sigma \sum_j (X^T u)_j v_j + \sigma^2 trace(vv^T) + \frac{\mu}{2}||v||_2^2 + y^{kT}(v - z^k) + \frac{\rho}{2}||v - z^k||_2^2) \quad (15)$$

$$= \frac{d}{dv}(-2\sigma v^T X^T u + \sigma^2 trace(vv^T) + \frac{\mu}{2}||v||_2^2 + y^{kT}(v - z^k) + \frac{\rho}{2}||v - z^k||_2^2) \quad (16)$$

$$= -2\sigma X^T u + 2\sigma^2 v + \mu v + y^k + \rho(v - z^k) \quad (17)$$

Next, we solve $\frac{dv^{k+1}}{dv} = 0 \Leftrightarrow (2\sigma^2 + \mu + \rho)v = 2\sigma X^T u - y^k + \rho z^k$ (18)

Thus, $v^{k+1} = \frac{2\sigma X^T u - y^k + \rho z^k}{(2\sigma^2 + \mu + \rho)}$ (19)

Next, we have

$$z^{k+1} = argmin_v L_\rho(v^{k+1}, z, y^k) \quad (20)$$

Hence

$$\frac{dz^{k+1}}{dz} = \frac{d}{dv}(\lambda||z||_1 + y^{kT}(v^{k+1} - z) + \frac{\rho}{2}||v^{k+1} - z||_2^2) \quad (21)$$

$$= \lambda\xi - y^k + \rho(v^{k+1} - z)(-1) \quad (22)$$

$$= \lambda\xi - y^k + \rho(z - v^{k+1}), \quad (23)$$

where

$$\xi_i = \begin{cases} 1 \ if \ z_i > 0 \\ [-1,1] \ if \ z_i = 0 \\ -1 \ if \ z_i < 0 \end{cases} \quad (24)$$

Solve $\frac{dz^{k+1}}{dz} = 0$, we have

$$z_i^{k+1} = v_i^{k+1} + \frac{1}{\rho} y_i^k - \frac{\lambda}{\rho}\xi_i \quad (25)$$

If $z_i^{k+1} > 0, \xi_i = 1$, then

$$v_i^{k+1} + \frac{1}{\rho} y_i^k - \frac{\lambda}{\rho} > 0 \Rightarrow v_i^{k+1} + \frac{1}{\rho} y_i^k > \frac{\lambda}{\rho} \quad (26)$$

If $z_i^{k+1} < 0, \xi_i = -1$, then

$$v_i^{k+1} + \frac{1}{\rho} y_i^k + \frac{\lambda}{\rho} < 0 \Rightarrow v_i^{k+1} + \frac{1}{\rho} y_i^k < -\frac{\lambda}{\rho} \tag{27}$$

If $z_i^{k+1} = 0$, then

$$-\frac{\lambda}{\rho} \leq v_i^{k+1} + \frac{1}{\rho} y_i^k \leq \frac{\lambda}{\rho} \tag{28}$$

Thus,

$$z_i = sign(v_i^{k+1} + \frac{1}{\rho} y_i^k) \max(\left| v_i^{k+1} + \frac{1}{\rho} y_i^k \right| - \frac{\lambda}{\rho}, 0) \tag{29}$$

Finally, we have

$$y^{k+1} = y^k + \rho(v^{k+1} - z^{k+1}) \tag{30}$$

4 Sparse Principle Component Analysis algorithm

In this section, we will present the sparse PCA algorithm

Algorithm 1: Sparse PCA algorithm

1. Input: The dataset $X \epsilon R^{p*n}$, where *p* is the dimension of the dataset and *n* is the total number of observations in the dataset
2. Compute $\tilde{X} = [x_1 - \mu | x_2 - \mu | \dots | x_n - \mu]$, where $\mu = \frac{1}{n} \sum_{i=1}^{n} x_i$ be the mean vector of all column vectors $x_1, x_2, \dots, x_n$ of *X*
3. Randomly select parameters ρ, μ, λ.
4. Set $V = zeros(n, dim)$
5. for $i = 1{:}\, dim$
    i. Compute the SVD of $\tilde{X}$
    ii. Initialize $v^0,\ z^0, y^0$
    iii. Set $k = 0$
    iv. do

a. Compute $v^{k+1} = argmin_v L_\rho(v, z^k, y^k)$

b. Compute $z^{k+1} = argmin_v L_\rho(v^{k+1}, z, y^k)$

c. Compute $y^{k+1} = y^k + \rho(v^{k+1} - z^{k+1})$

d. $k = k + 1$

v. while $||v^{k+1} - v^k|| > 10^{-10}$

vi. $v = \frac{v^{k+1}}{norm(v^{k+1})}$

vii. $V(:, i) = v$

viii. $\tilde{X} = \tilde{X}(I - vv^T)$

6. End
7. Output: The matrix *V*.

---

5 Kernel Ridge Regression Algorithm

Given a set of feature vectors of speech samples $\{x_1, \ldots, x_l, x_{l+1}, \ldots, x_{l+u}\}$ where $n = l + u$ is the total number of speech samples.
Please note that $\{x_1, \ldots, x_l\}$ is the set of all labeled points and $\{x_{l+1}, \ldots, x_{l+u}\}$ is the set of all un-labeled points. The way constructing the feature vectors of speech samples will be discussed in Section 7.
Let K represents the kernel matrix of the set of labeled points.
Let *c* be the total number of words.
Let $Y \in R^{l*c}$ the initial label matrix for *l* labeled speech samples be defined as follows

$$Y_{ij} = \begin{cases} 1 \text{ if } x_i \text{ belongs to word } j \\ 0 \text{ if } x_i \text{ does not belong to word } j \end{cases} \quad (31)$$

Our objective is to predict the labels of the un-labeled points $x_{l+1}, \ldots, x_{l+u}$.
Let the matrix $F \in R^{u*c}$ be the estimated label matrix for the set of feature vectors of speech samples $\{x_{l+1}, \ldots, x_{l+u}\}$, where the point x_i is labeled as $max_j(F_{ij})$ for each word *j* ($1 \leq j \leq c$).

Kernel Ridge Regression

In this section, we will give the brief overview of the original kernel ridge regression method [3]. The outline of this algorithm is as follows

1. Form the kernel matrix K. The way constructing *K* will be discussed in section 7.
2. Compute $\alpha = (K + \lambda I)^{-1} Y$
3. For each speech sample *k* belong to the set of un-labeled points, compute the vector $K(k,:) \in R^{1*l}$, where the value of each element of this vector is defined as the similarity of feature vector of speech sample *k* and feature vector of speech sample of the set of *l* labeled points.
4. Compute $F(k,:) = K(k,:) * \alpha$. Label each speech samples $x_k (l+1 \leq k \leq l+u)$ as the integer value result of $max_j F_{kj}$

6 Detailed Derivation of Kernel Ridge Regression

Consider the problem of finding a homogeneous real-valued linear function

$$g(x) = w^T x \tag{32}$$

that best interpolates a given training set

$$S = \{(x_1, y_1), \dots (x_l, y_l)\} \tag{33}$$

of point $x_i \in R^{d*1}$ with corresponding label $y_i \in \{0,1\}$ and *d* is the dimension of the feature vector of the speech samples.

Measures discrepancy between function output and correct output (squared to ensure always positive):

$$f(x,y) = (g(x) - y)^2 \tag{34}$$

We introduce notation: matrix X has rows of *l* labeled points. Hence we can write

$$\varepsilon = y - Xw \tag{35}$$

for the vector of differences between $g(x_i)$ and y_i.

We need to ensure that flexibility of *g* is controlled (controlling the norm of *w* proves effective)

$$min_w L(w,X) = min_w (\lambda \|w\|^2 + \|\varepsilon\|^2) \tag{36}$$

where we can compute

$$\|\varepsilon\|^2 = y^T y - 2w^T X^T y + w^T X^T X w \tag{37}$$

Setting the derivative of *L(w,X)* equal to zero gives

$$X^T X w + \lambda w = X^T y \tag{38}$$

We get the primal solution weight vector

$$w = (X^T X + \lambda I)^{-1} X^T y \tag{39}$$

and the regression function

$$g(x) = x^T w = x^T (X^T X + \lambda I)^{-1} X^T y \tag{40}$$

A dual solution expresses the weight vector as a linear combination of the training examples can be obtained from (1). We have

$$X^T Xw + \lambda w = X^T y \quad (41)$$

This implies that

$$w = X^T \frac{1}{\lambda}(y - Xw) = X^T \alpha \quad (42)$$

where $\alpha = \frac{1}{\lambda}(y - Xw)$ (43)

The vector α is the dual solution.

Substitute $w = X^T \alpha$ into equation (2), we obtain

$$\lambda \alpha = y - XX^T \alpha. \quad (44)$$

Thus we can get the dual solution and the regression function as the followings

$$\alpha = (XX^T + \lambda I)^{-1} y \quad (45)$$

$$g(x) = x^T w = x^T X^T \alpha = \alpha^T Xx \quad (46)$$

7 Experiments and Results

In this paper, the set of 4,500 speech samples recorded of 50 different words (90 speech samples per word) are used for training. Then another set of 500 speech samples of these words are used for testing the sensitivity measure. This dataset is available from the IC Design lab at Faculty of Electricals-Electronics Engineering, University of Technology, Ho Chi Minh City. After being extracted from the conventional MFCC feature extraction method, the column sum of the MFCC feature matrix of the speech sample will be computed. The result of the column sum which is the R^{26*1} column vector will be used as the feature vector of the combination of the sparse PCA and the kernel ridge regression algorithms.

First, the sparse PCA algorithm will be applied to 5,000 speech samples to transform this original dataset to the new dataset. Finally, the kernel ridge regression algorithm will be applied to this new dataset. Please note that 4,500 speech samples are still used for training and 500 speech samples are still used for testing.

There is one way to construct the kernel matrix from these feature vectors (of the set of labeled points): The fully connected network (i.e. all speech samples in the labeled set are connected).

In this paper, the similarity function (i.e. the kernel value of speech samples *i* and speech sample *j*) is the value of the dot product of feature vector of speech sample *i* and feature vector of speech sample *j*.

In this section, we experiment with the above kernel ridge regression method in terms of sensitivity measure. All experiments were implemented in Matlab 6.5 on virtual machine. The sensitivity measure Q is given as follows:

$$Q = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (46)$$

True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) are defined in the following Table 1

Table 1: Definitions of TP, TN, FP, and FN

		Predicted Label	
		Positive	Negative
Known Label	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

For this dataset, the second table shows the sensitivity measures of the kernel ridge regression method alone, the HMM method (i.e. the current state of the art method of speech recognition application), the combination of the PCA and the kernel ridge regression method, and the combination of the sparse PCA and the kernel ridge regression method.

Table 2: Comparisons of the kernel ridge regression method, the HMM method, the combination of the PCA and the kernel ridge regression method, and the combination of the sparse PCA and the kernel ridge regression method

Sensitivity Measures (%)	
HMM method (4 mixtures, 8 states)	89%
Kernel Ridge Regression (dot product)	94%
PCA *(d=19)* + Kernel Ridge Regression (dot product)	94%
PCA *(d=20)* + Kernel Ridge Regression (dot product)	95%
PCA *(d=21)* + Kernel Ridge Regression (dot product)	95.4%
PCA *(d=22)* + Kernel Ridge Regression (dot product)	94.8%
Sparse PCA *(d=19)* + Kernel Ridge Regression (dot product)	94%
Sparse PCA *(d=20)* + Kernel Ridge Regression (dot product)	95%
Sparse PCA *(d=21)* + Kernel Ridge Regression (dot product)	95.4%
Sparse PCA *(d=22)* + Kernel Ridge Regression (dot product)	95%

From the above Table 2, we recognized that the combination of the sparse PCA and the kernel ridge regression method outperforms the current state of the art HMM method and the Kernel Ridge Regression method alone in terms of sensitivity measures in speech recognition problem. Moreover, the combination of the sparse PCA and the kernel ridge regression method is at least as good as the combination of the PCA and the kernel ridge regression method but sometimes leads to better sensitivity performance measures.

8 Conclusion

The detailed algorithm combining the sparse PCA method and the kernel ridge regression method applying to the speech recognition problem has been developed. We easily recognized that this combination of the sparse PCA method and the kernel ridge regression method outperform the kernel ridge regression method alone and the current state of the art method for speech recognition which is the HMM method.

In the future, the other dimensional reduction methods will be explored such as Local Linear Embedding [12] and Laplacian Eigenmaps methods [13]. The combinations of these dimensional reduction methods and the Kernel Ridge Regression method have not yet been investigated, to the best of our knowledge. Those methods' performances will be compared to the sparse PCA method's performances.

ACKNOWLEDGEMENTS.
This work is funded by the Ministry of Science and Technology, State-level key program, Research for application and development of information technology and communications, code KC.01.23/11-15.

References

[1] Lawrence Rabiner, Biing Hwan Juang, Fundamentals of speech recognition, AT&T, 1993, 507 pages.

[2] Ganapathiraju, Aravind. Support vector machines for speech recognition Diss. *Mississippi State University*, 2002

[3] Zhang, Yuchen, John Duchi, and Martin Wainwright. Divide and conquer kernel ridge regression Conference on Learning Theory. 2013

[4] Marshall, Austin. *Artificial Neural Network for Speech Recognition 2nd Annual Student Research Showcase* (2005)

[5] J. Labiak and K. Livescu. Nearest neighbor classifiers with learned distances for phonetic frame classification. *In Proceedings of Interspeech*, 2011

[6] Kokiopoulou, E., and Y. Saad. PCA and kernel PCA using polynomial filtering: a case study on face recognition. *Technical Report umsi-2004-213, Minnesota Supercomputer Institute, University of Minnesota, Minneapolis*, MN, 2004. submitted, 2004.

[7] R.E. Hausman. Constrained multivariate analysis. *Studies in the Management Sciences*, 19 (1982), pp. 137–151
[8] Vines, S. K. Simple principal components. Journal of the Royal Statistical Society: Series C (Applied Statistics) 49.4 (2000): 441-451.
[9] Jolliffe, Ian T., Nickolay T. Trendafilov, and Mudassir Uddin. A modified principal component technique based on the LASSO. *Journal of computational and Graphical Statistics* 12.3 (2003): 531-547.
[10] Zou, Hui, Trevor Hastie, and Robert Tibshirani. Sparse principal component analysis. Journal of computational and graphical statistics 15.2 (2006): 265-286.
[11] Boyd, Stephen, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning* 3.1 (2011): 1-122.
[12] Roweis, Sam T., and Lawrence K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science* 290.5500 (2000): 2323-2326.
[13] Belkin, Mikhail, and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation* 15.6 (2003): 1373-1396.