Comparative study of Kernel Canonical Correlation Analysis and Kernel Partial Least Square with application

Souilem_n@yahoo.com, elaissi_ilyes@yahoo.fr, taoualiokba@yahoo.fr, hassani.messaoud@enim.rnu.tn

Abstract—This paper treats the identification of nonlinear systems using RKHS models, with Kernel Canonical Correlation Analysis (KCCA) technique. KCCA finds common semantic features between mapped input-output data in high dimension nonlinear spaces, and then use the common features to represent the data. We use KCCA and Kernel Partial Least Square (KPLS) technique in nonlinear system identification benchmark, a Wiener-Hammerstein, and we compare results.

Keywords—RKHS, Kernel method, KPLS, KCCA, Wiener-Hammerstein

I. INTRODUCTION

During last years, a lot of researchs were performed in the field of nonlinear system identification using Kernel methods [5, 12], many techniques have been developed and widely used in different applications, such as Support Vector Machines (SVM) [3], Kernel Partial Least Square (KPLS) [10], Kernel Principal Component Analysis (KPCA) [11], Kernel Independent Component Analysis (KICA) [2]. ...

In this paper we are interested in using Kernel Canonical Correlation Analysis (KCCA) in regression problems. KCCA is the kernel extension of Canonical Correlation Analysis (CCA) with positive definite kernels [1] and is mainly used in classification problems [5, 6, 7]. CCA was proposed by Hotelling in 1936 [8], to solve the problem of finding vector bases for two sets of variables such that the correlations between the projections of the variables onto these bases are mutually maximized. The solution then results in solving an eigenvector problem. Kernel methods principle maps the data into a higher dimensional feature space. KCCA is then applied on the mapped input-output data and extract their common information allowing the construction of the RKHS model.

The partial least squares (PLS) [15, 17] creates orthogonal score vectors (component, latent vectors) by using the existing correlations between different sets of variables (blocks of data) while also keeping most of the variance of both sets. Kernel PLS (KPLS) is the kernel extension of PLS and deals with the data mapped into a higher dimensional feature [10, 16].

In this paper we use KCCA and KPLS techniques to identify RKHS models for nonlinear systems, with applications on a Wiener-Hammerstein benchmark [13].

The paper is organized as follows. In sections 2, 3 and 4 we present the Kernel Canonical Correlation Analysis, the eigenvector problem is firstly solved with a complete Cholesky decomposition, and then with a Partial Gram Schmidt Orthogonalisation [3]. In section five, we perform a numerical simulation on benchmark application, and a comparison study on KCCA and KPLS results is presented.

II. KERNEL CANONICAL CORRELATION ANALYSIS (KCCA)

Due to its linearity, CCA may not extract useful descriptors of the data in nonlinear cases. KCCA offers a solution by mapping the data into a higher dimensional feature space. Consider the set of data \( D = \{x_i, y_i\}_{i=1,...,N} \), \( N \) the number of observations. We define \( k_x : E_x^2 \rightarrow IR \) and \( k_y : E_y^2 \rightarrow IR \) to be continuous positive definite kernels[9, 14, 1]. It exists sequences of orthonormal eigenfunctions \( \left( \varphi_{y_1}, \varphi_{y_2}, ..., \varphi_{y_L} \right) \) in \( L^2(E_y) \) and \( \left( \varphi_{x_1}, \varphi_{x_2}, ..., \varphi_{x_l} \right) \) in \( L^2(E_x) \) (where \( l \) and \( l' \) can be infinite) so that:

\[
k_x(x, x') = \sum_{j=1}^l \sigma_j \varphi_{x_j}(x) \varphi_{x_j}(x') \quad x, x' \in E_x
\]

(1)

\[
k_y(y, y') = \sum_{j=1}^{l'} \sigma_j \varphi_{y_j}(y) \varphi_{y_j}(y') \quad y, y' \in E_y
\]

(2)

Let \( F_x \subset L^2(E_x) \) (resp. \( F_y \subset L^2(E_y) \)) be a Hilbert space associated to the kernel \( k_x \) (resp. \( k_y \)) and defined by:

\[
F_x = \left\{ g_x \in L^2(E_x) | g_x = \sum_{i=1}^l w_i \varphi_{x_i} \text{ and } \sum_{j=1}^l \frac{w_j^2}{\sigma_j} < +\infty \right\}
\]

(3)

\[
F_y = \left\{ g_y \in L^2(E_y) | g_y = \sum_{i=1}^{l'} w_i \varphi_{y_i} \text{ and } \sum_{j=1}^{l'} \frac{w_j^2}{\sigma_j} < +\infty \right\}
\]

(4)

Let the applications \( \Phi_x \) and \( \Phi_y \):
\( \Phi : E \rightarrow \mathbb{R}^l \)

\[
x \mapsto \Phi(x) = \begin{bmatrix} \varphi_1(x) \\ \vdots \\ \varphi_l(x) \end{bmatrix}
\]

(5)

\( \Phi : E \rightarrow \mathbb{R}^l \)

\[
y \mapsto \Phi(y) = \begin{bmatrix} \varphi_1(y) \\ \vdots \\ \varphi_l(y) \end{bmatrix}
\]

(6)

We have then:

\[
k_x(x, x') = \langle \Phi(x), \Phi(x') \rangle, \quad x, x' \in E_x
\]

\[
k_y(y, y') = \langle \Phi(y), \Phi(y') \rangle, \quad y, y' \in E_y
\]

Let the Gram matrices \( K_x \in \mathbb{R}^{N \times N} \) and \( K_y \in \mathbb{R}^{N \times N} \):

\[
(K_x)_{ij} = k_x(x_i, x_j) \quad \text{and} \quad (K_y)_{ij} = k_y(y_i, y_j)
\]

We consider \( u_x, v_y \) defined by:

\[
u_x = \langle \Phi_x(x), f_x \rangle = f_x^T \Phi_x(X)
\]

(7)

\[
u_y = \langle \Phi_y(y), f_y \rangle = f_y^T \Phi_y(Y)
\]

(8)

With \( \Phi_x(X) \) and \( \Phi_y(Y) \) are the mappings of \( X \) and \( Y \) with the applications \( \Phi_x \) and \( \Phi_y \) respectively. We suppose that \( \Phi_x(X) \) and \( \Phi_y(Y) \) are centered, and with \( f_x \in F_x \), \( f_y \in F_y \) are defined as:

\[
f_x = \sum_{i=1}^{N} \alpha_i \Phi_x(x_i)
\]

(9)

\[
f_y = \sum_{i=1}^{N} \beta_i \Phi_y(y_i)
\]

(10)

\( \alpha_i, \beta_i \in \mathbb{R} \), \( i = 1, \ldots, N \)

The correlation between \( u_x \) and \( v_y \) is defined by

\[
\rho = \text{corr}(\langle \Phi_x(x), f_x \rangle, \langle \Phi_y(y), f_y \rangle)
\]

(11)

\[
= \frac{\text{cov}(\langle \Phi_x(x), f_x \rangle, \langle \Phi_y(y), f_y \rangle)}{\text{var}(\langle \Phi_x(x), f_x \rangle) \text{var}(\langle \Phi_y(y), f_y \rangle)} = \frac{\text{cov}(u_x, v_y)}{\text{var}(u_x) \text{var}(v_y)}
\]

(12)

with:

\[
\text{cov}(u_x, v_y) = \frac{1}{N} \sum_{i=1}^{N} \left\langle \Phi_x(x_i), \sum_{i=1}^{N} \alpha_i \Phi_x(x_i) \right\rangle \left\langle \Phi_y(y_i), \sum_{i=1}^{N} \beta_i \Phi_y(y_i) \right\rangle
\]

\[
\text{var}(u_x) = \frac{1}{N} \sum_{i=1}^{N} \left\langle \Phi_x(x_i), \sum_{i=1}^{N} \alpha_i \Phi_x(x_i) \right\rangle \left\langle \Phi_x(x_i), \sum_{i=1}^{N} \alpha_i \Phi_x(x_i) \right\rangle
\]

Let

\[
\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_N)^T
\]

(12)

\[
\beta = (\beta_1, \beta_2, \ldots, \beta_N)^T
\]

(13)

Using the kernel trick, and performing some easy calculus, we have:

\[
\text{cov}(u_x, v_y) = \frac{1}{N} \alpha^T K_x K_y \beta
\]

(14)

\[
\text{var}(u_x) = \frac{1}{N} \alpha^T K_x K_x \alpha
\]

(15)

\[
\text{var}(v_y) = \frac{1}{N} \beta^T K_y K_y \beta
\]

(16)

KCCA then solves the following problem:

\[
\max_{\alpha, \beta} \frac{\alpha^T K_x K_x \beta}{\sqrt{\alpha^T K_x \alpha} \sqrt{\beta^T K_y \beta}}
\]

(17)

subject to:

\[
\alpha^T K_x \alpha = 1 \quad \text{and} \quad \beta^T K_y \beta = 1
\]

The Lagrangian is

\[
L(\lambda, \lambda', \alpha, \beta) = \alpha^T K_x \beta - \frac{\lambda}{2}(\alpha^T K_x \alpha - 1) - \frac{\lambda'}{2}(\beta^T K_y \beta - 1)
\]

The derivative of the Lagrangian:

34
\[
\frac{\partial L}{\partial \alpha} = K_x K_x \beta - \lambda_\alpha K_{x}^2 \alpha = 0
\]
(18)
\[
\frac{\partial L}{\partial \beta} = K_x K_x \alpha - \lambda_\beta K_{x}^2 \beta = 0
\]
(19)

Multiplying relation (18) by \( \alpha \) and (19) by \( \beta \), and subtracting them, we find
\[
\alpha \beta K_{x} K_{y} \lambda \alpha \beta K_{x}^2 \alpha - \beta \alpha K_{y} K_{x} \lambda \beta \alpha K_{x}^2 \beta = 0
\]

Or: \( \lambda_\alpha \beta K_{x} K_{y} \lambda \alpha \beta K_{x}^2 \alpha = 0 \)

Then \( \lambda_\beta = \lambda_\alpha \), let \( \lambda_\beta = \lambda_\alpha = \lambda \)

We consider \( K_x \) and \( K_y \) are non singular, we have:
\[
\beta = \frac{K_x K_x \alpha}{\lambda}
\]
(20)

From relation (20) and (18) we find
\[
K_x K_y K_x^{-1} K_x \lambda \lambda K_x K_y \lambda \lambda K_x \alpha = 0
\]

Hence \( K_x K_y K_x^{-1} K_x \lambda \lambda K_x, K_y \alpha = 0 \)

Then
\[
I \alpha = \lambda \lambda \alpha
\]
(21)

From equation (21), we can deduce that \( \lambda = 1 \) for every vector \( \alpha \), hence if \( K_x \) and \( K_y \) are invertible, then perfect correlation can be formed, suggesting learning is trivial and applying KCCA in this fashion will not then provide useful results.

Instead of solving (17), we solve a regularized form, we introduce a control on the flexibility of the projection mapping by penalizing the norms of the associated weights, and we convexly combine the CCA term with the regularization term, we obtain:
\[
\max_{\alpha, \beta} \frac{\alpha \beta K_{y} K_{y} \beta}{\sqrt{\alpha \beta K_{x} K_{x} \alpha + \tau \| \alpha \|}} - \frac{\beta \alpha K_{x} K_{x} \beta}{\sqrt{\beta \alpha K_{y} K_{y} \beta + \tau \| \beta \|}} > 0
\]
(22)

According to the definitions of \( f_x \) and \( f_y \), in (9) and (10):
\[
\| f_x \| = \langle f_x, f_x \rangle = \left\{ \sum_{i=1}^{N} \alpha_i \Phi_i(x_i), \sum_{j=1}^{N} \alpha_j \Phi_j(x_j) \right\} = \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j k(x_i, x_j) = \alpha \beta K_{x} K_{y}
\]

Similarly \( \| f_y \| = \langle f_y, f_y \rangle = \beta \beta K_{y} K_{x} \beta \)

Then the KCCA regularized problem becomes
\[
\max_{\alpha, \beta} \frac{\alpha \beta K_{y} K_{y} \beta}{\sqrt{\alpha \beta K_{x} K_{x} \alpha + \tau \| \alpha \|}} - \frac{\beta \alpha K_{x} K_{x} \beta}{\sqrt{\beta \alpha K_{y} K_{y} \beta + \tau \| \beta \|}}
\]
(23)

Subject to
\[
\alpha \beta K_{y} K_{y} \lambda \alpha \beta K_{x}^2 \alpha + \beta \alpha K_{x} K_{y} \lambda \beta \alpha K_{y}^2 \beta = 1
\]

The Langrangien is
\[
L(\lambda_\alpha, \lambda_\beta, \alpha, \beta) = \frac{\lambda_\alpha}{\lambda_\beta} (\alpha \beta K_{y} K_{y} \lambda \beta \alpha K_{y}^2 \beta + \beta \alpha K_{x} K_{y} \lambda \alpha \beta K_{x}^2 \alpha) = \frac{\lambda_\alpha}{\lambda_\beta} (\alpha \beta K_{y} K_{y} \lambda \beta \alpha K_{y}^2 \beta + \beta \alpha K_{x} K_{y} \lambda \alpha \beta K_{x}^2 \alpha) - 1
\]

We calculate \( \frac{\partial L}{\partial \alpha} \) and \( \frac{\partial L}{\partial \beta} \) we obtain:
\[
\frac{\partial L}{\partial \alpha} = K_x K_y \beta - \lambda_\alpha (K_x^2 \alpha + \tau K_x \alpha) = 0
\]
(24)
\[
\frac{\partial L}{\partial \beta} = K_y K_x \alpha - \lambda_\beta (K_y^2 \beta + \tau K_y \beta) = 0
\]
(25)

We multiply (24) by \( \alpha \beta \) and (25) by \( \beta \alpha \) and subtracting both equations, we obtain:
\[
\alpha \beta K_x K_y \beta - \lambda_\alpha (K_x^2 \alpha + \tau K_x \alpha) - \beta \alpha K_y K_x \alpha + \lambda_\alpha (K_y^2 \beta + \tau K_y \beta) = 0
\]

Then
\[
\lambda_\alpha (K_x^2 \alpha + \tau K_x \alpha) - \lambda_\alpha (K_y^2 \beta + \tau K_y \beta) = 0
\]

Then \( \lambda_\alpha = \lambda_\beta \), let \( \lambda_\alpha = \lambda_\beta = \lambda \)

We consider \( K_x \) and \( K_y \) non singular, then equation (25) gives:
\[ \beta = \frac{(K_x + \tau I)^{-1} K_x \alpha}{\lambda} \]  

(26)

The equations (24) and (26) give
\[ K_x (K_x + \tau I)^{-1} K_x \alpha = \lambda \tilde{K}_x (K_x + \tau I) \alpha \]

Then
\[ K_x (K_x + \tau I)^{-1} K_x \alpha = \lambda \tilde{K}_x (K_x + \tau I) \alpha \]

we finally obtain the eigenvector problem
\[ (K_x + \tau I)^{-1} K_x (K_x + \tau I)^{-1} K_x \alpha = \lambda \tilde{K}_x (K_x + \tau I) \alpha \]

(27)

we obtain an eigenvector problem of the form
\[ A_1 x = \tilde{\lambda} \tilde{x} \]

(28)

Unfortunately, the matrix \( A_1 \) in (28) is not symmetric and this may yield imaginary eigenvalues. Besides, matrix inversion in the case of large \( K_x \) training sets may lead to computational problems.

III. FIRST LEVEL HEADING KCCA REGULARISATION WITH PARTIAL GRAM CHMIDT ORTHOGONALISATION

To overcome the problem we apply Partial Gram Schmidt Orthogonalisation (PGSO) \[3\] or equivalently incomplete Cholesky decomposition to reduce the dimensionality of the kernel matrices. We will decompose the kernel matrices \( K_x \) and \( K_y \) via the complete Cholesky decomposition as:
\[ K_x = R_x R_x^T \]
\[ K_y = R_y R_y^T \]

where \( R_x \) and \( R_y \) is a lower triangular matrices, gives

the relation (24) et (25) can be written as
\[ R_x R_x^T R_x R_y^T \beta - \lambda R_x R_x^T (R_x R_x^T R_x R_y^T + \tau R_x R_y^T) \alpha = 0 \]

(29)

\[ R_y R_y^T R_y R_y^T \alpha - \lambda R_y R_y^T (R_y R_y^T R_y R_y^T + \tau R_y R_y^T) \beta = 0 \]

(30)

Multiplying (29) with \( R_x^T \) and (30) with \( R_y^T \), we obtain
\[ R_x^T R_x R_x^T R_y R_y^T \beta - \lambda R_x^T R_x R_x^T (R_x R_x^T R_x R_y^T + \tau R_x R_y^T) \alpha = 0 \]

(31)

\[ R_y^T R_y R_y^T R_y R_y^T \alpha - \lambda R_y^T R_y R_y^T (R_y R_y^T R_y R_y^T + \tau R_y R_y^T) \beta = 0 \]

(32)

we define \( Z_{xx} = R_x^T R_x \), \( Z_{xy} = R_y^T R_y \), \( Z_{yx} = R_y^T R_y \), \( Z_{yy} = R_y^T R_y \), \( \tilde{\alpha} = R_x^T \alpha \), \( \tilde{\beta} = R_y^T \beta \)

the relation (31) and (32) can be written as
\[ Z_{xx} \tilde{\beta} - \lambda Z_{xx} (Z_{xx} + \tau I) \tilde{\alpha} = 0 \]

(33)

\[ Z_{yx} \tilde{\alpha} - \lambda Z_{yx} (Z_{yx} + \tau I) \tilde{\beta} = 0 \]

(34)

From equation (34) we write
\[ \tilde{\beta} = \frac{(Z_{yx} + \tau I)^{-1} Z_{yx}^{-1} \tilde{Z}_{xy} \tilde{Z}_{yx} \tilde{\alpha}}{\lambda} \]

(35)

Then
\[ \tilde{\beta} = \frac{(Z_{yx} + \tau I)^{-1} Z_{yx}^{-1} \tilde{Z}_{yx} \tilde{Z}_{yx} \tilde{\alpha}}{\lambda} \]

(36)

Using the relation (33) and (35), we obtain
\[ Z_{xx} \tilde{\beta} - \lambda Z_{xx} (Z_{xx} + \tau I) \tilde{\alpha} = \lambda^2 Z_{xx} (Z_{xx} + \tau I) \tilde{\alpha} \]

(37)

Let \( B \) be the lower triangular matrix of the complete Cholesky decomposition of \( Z_{xx} + \tau I \) such that
\[ Z_{xx} + \tau I = B B^T \] and let \( \tilde{\alpha} = B^T \tilde{\alpha} \), the equation (27) then becomes
\[ B^{-1} Z_{xx} (Z_{yx} + \tau I)^{-1} Z_{yx} (B^{-1})^T \tilde{\alpha} = \lambda^2 \tilde{\alpha} \]

(38)

It is a symmetric eigenvector problem of the form
\[ A_2 x = \lambda x \] with
\[ A_2 = B^{-1} Z_{xx} (Z_{yx} + \tau I)^{-1} Z_{yx} (B^{-1})^T \]. The eigenvector problem (38) is equivalent to the eigenvector problem (27), with \( \tilde{\alpha} = B^T R_x^T \alpha \).
From the equations (7), (8) we have $u_i = f_i^T \Phi_i(X)$ and $v_i = f_i^T \Phi_i(Y)$, the problem is to find matrix $A$, such that

$$\min_A \| f_i^T \Phi_i(Y) - A f_i^T \Phi_i(X) \|$$

(39)

Using relations (12) and (13) we can write

$$f_i^T \Phi_i(X) = \alpha_i^T K_i$$

(40)

$$f_i^T \Phi_i(Y) = \beta_i^T K_i$$

(41)

The problem can now be written as follows:

$$\min_A \| \beta_i^T K_i - A \alpha_i^T K_i \|^2$$

(42)

The matrix $K_i = YY^T$, we obtain

$$Y Y^T = (\beta \beta^T)^{-1} \beta A \alpha^T K_i$$

(43)

The estimate output can be found from the equation (43).

IV. SIMULATIONS

We proceed to the identification of a Wiener Hammerstein using KCCA and KPLS methods, and we compare the performances of the RKHS models obtained with the both methods. The vector $x$ of RKHS model have the structure:

$$x = (u(i-1), u(i-2), y(i-1))^T$$

(44)

The system to be modelled is sketched by Figure 1. It consists on an electronic nonlinear system with a Wiener Hammerstein structure that was built by Gerd Vendesteen [13]. This process was adopted as a nonlinear system benchmark in SYSID 2009.

In simulation of both methods KCCA and KPLS, we have used 300 observations in the identification phase and 1600 other observations in the validation one.

In figure 2 we plot the RKHS model outputs using KCCA methods and the output process in the validation phase. As previously we notice the concordance between the outputs. The Means Square Error (MSE) % is equal to 0.1 %.

In figure 3 we plot the RKHS model outputs using KPLS methods and the output process in the validation phase. As previously we notice the concordance between the outputs. The Means Square Error (MSE) % is equal to 0.01 %.

In Table I we present the performances of both kernel methods. The first performance is the generalization ability evaluated by the Means Square Error (MSE) % in the identification phase and the validation one and the second concerns the compute time.
Table 1. Performances of kernel methods KCCA and KPLS

<table>
<thead>
<tr>
<th></th>
<th>KCCA</th>
<th>KPLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSE learning %</td>
<td>0.003</td>
<td>0.0034</td>
</tr>
<tr>
<td>MSE validation %</td>
<td>0.1</td>
<td>0.0161</td>
</tr>
<tr>
<td>Compute time (seconds)</td>
<td>1.8218</td>
<td>0.8281</td>
</tr>
</tbody>
</table>

VI. CONCLUSION

In this paper we have presented the KCCA method, it is a kernel method used to extract common features between highly non linear mapped data. We have used KCCA in the regression case to identify benchmark system. A comparative study with KPLS method has been achieved and shows the effectiveness of the method.

REFERENCES