Data Visualization and Dimensionality Reduction using Kernel Maps with a Reference Point

Johan A.K. Suykens

K.U. Leuven, ESAT-SCD/SISTA
Kasteelpark Arenberg 10
B-3001 Leuven (Heverlee), Belgium
Tel: 32/16/32 18 02 Fax: 32/16/32 19 70
Email: johan.suykens@esat.kuleuven.be

ESAT-SCD/SISTA Technical Report 07-22
submitted for publication

February 1, 2007

Abstract

In this paper a new kernel based method for data visualization and dimensionality reduction is proposed. A reference point is considered corresponding to additional constraints taken in the problem formulation. In contrast with the class of kernel eigenmap methods, the solution (coordinates in the low dimensional space) is characterized by a linear system instead of an eigenvalue problem. The kernel maps with a reference point are generated from a least squares support vector machine core part that is extended with an additional regularization term for preserving local mutual distances together with reference point constraints. The kernel maps possess primal and dual model representations and provide out-of-sample extensions e.g. for validation based tuning. The method is illustrated on toy problems and real life data sets.

Keywords: Dimensionality reduction, data visualization, kernel methods, least squares support vector machines, constrained optimization, feature map, positive definite kernel.

1 Introduction

Traditionally, techniques such as principal component analysis and self organizing maps have been frequently applied for dimensionality reduction and data visualization [13, 15]. In recent years the search for new approaches and solutions to this problem has become an active area of research and a variety of new techniques have been proposed such as isomap, locally linear embedding, Hessian locally linear embedding, diffusion maps, Laplacian eigenmaps and others [2, 6, 9, 14, 25, 28]. Currently, many of these techniques have been characterized under the umbrella of kernel eigenmap methods and manifold learning [3, 7]. Nevertheless, the selection of components and tuning parameters is often difficult to assess. For several existing approaches the characterization of out-of-sample extensions is unclear, though approximate versions have been proposed [4].

In this paper we want to take a fresh look at the problem of data visualization with dimensionality reduction. A main objective here is to design a technique for which the solution follows from solving a linear system instead of an eigenvalue problem. At the same time the method should be able to perform nonlinear dimensionality reduction with preserving local mutual distances and allow for out-of-sample extensions that can be made in an exact way.

Kernel based learning methods have proven to be successful in many applications in different areas, especially also on problems with high-dimensional input spaces. Different methodologies and mathematical frameworks have emerged that are making use of kernels, including support vector machine (SVM) methodologies, function estimation in reproducing kernel Hilbert spaces (RKHS) and Bayesian learning viewpoints with Gaussian processes [10, 19, 20, 24, 26, 27]. In this paper we propose a new kernel based method in the framework of least squares support vector machines (LS-SVMs). In this context kernel based models have been studied for regression, classification, principal component analysis, spectral clustering, canonical correlation analysis, recurrent networks, optimal control and others [1, 17, 22, 23, 24]. The formulations are in terms of constrained optimization problems with use of a feature map in the primal problem and a related positive definite kernel in the dual (which is the problem in the Lagrange multipliers that are related to the constraints). This is similar to standard support vector machine formulations [26] but by making use of an L_2 loss function and equality constraints instead of inequality

constraints. Some advantages of this setting are the systematic and straightforward way of deriving optimal model representations and constraints handling.

The kernel map approach that we propose in this paper makes use of a reference point. Such a reference point is expressed in terms of constraints that are added to the formulation. We show how this mechanism converts the eigenvalue problem into a linear system. As stated in [21], complexity theory results for linear systems are easier to obtain than for eigenvalue problems. Also the development of fast cross-validation techniques and on-line learning methods for the design of adaptive systems becomes easier [5] and both direct and iterative solvers can be used [12]. The solution to the linear system delivers the coordinates in the low dimensional space. The support values follow from a second linear system. The optimal model representation and solution are generated in a straightforward way from a least squares support vector machine core part that maps the input data to the coordinates in the low dimensional space. An additional regularization term that preserves local mutual distances is incorporated. This term is a modified version of the one that is considered in locally linear embedding [16].

The problem statement as a constrained optimization problem admits primal and dual model representations, respectively in terms of the feature map and the associated positive definite kernel function. An important consequence is that the models can immediately be extended for the evaluation at new data points with out-of-sample extensions. Despite the importance of studies in learning theory [8, 26], practitioners often still have to rely on validation based methods, due to the lack of sharp bounds on the generalization error. In this paper we show for a number of examples with toy problems and real-life data sets how the tuning process of regularization constants and kernel parameters can be guided by making use of a validation set. Examples are shown with 2D and 3D visualizations of the data sets. While the reference point enables to achieve a linear system solution, the point has to be sacrificed and omitted in the final data visualization.

This paper is organized as follows. In Section 2 we give preliminary definitions and prepare the context of this paper. In Section 3 we consider kernel maps that lead to an eigenvalue problem, generated from a least squares support vector machine core model. In Section 4 we discuss the introduction of a reference point leading to a linear system solution. First this is explained for projection to a two-dimensional space and next for

the general case. In Section 5 aspects of model selection with use of a validation set are discussed. In Section 6 examples are given on toy problems and real life data sets. In order to enhance the readability of the paper, all proofs are given in Appendix. Supplementary material is provided with links to a Matlab demo file (Swiss roll problem) and with respect to the experimental section.

2 Definitions and context

Consider a given input data set $\{x_i\}_{i=1}^N$ of N data points $x_i \in \mathbb{R}^p$. We aim at realizing a dimensionality reduction to a low dimensional space by $g(\cdot) : \mathbb{R}^p \to \mathbb{R}^d$ with $d = 2$ or $d = 3$ for data visualization of the corresponding points $\{z_i\}_{i=1}^N$ with $z_i \in \mathbb{R}^d$. Let us define the vector $z = [z_1; z_2; ...; z_N] = [z_1^T z_2^T ... z_N^T]^T \in \mathbb{R}^{dN}$ and the following mechanism to select the i-th data point from the vector z for the different components:

$$\begin{aligned} z_{i,1} &= c_{i,1}^T z \\ z_{i,2} &= c_{i,2}^T z \\ &\vdots \\ z_{i,d} &= c_{i,d}^T z \end{aligned} \tag{1}$$

where $z_{i,l}$ denotes the l-th component ($l = 1, .., d$) of the i-th data point vector z_i ($i = 1, ..., N$) with

$$\begin{aligned} c_{1,1} &= [1; 0_{d-1}; 0_d; ...; 0_d] \\ &... \\ c_{1,d} &= [0_{d-1}; 1; 0_d; ...; 0_d] \\ &... \\ c_{N,1} &= [0_d; 0_d; ...; 1; 0_{d-1}] \\ &... \\ c_{N,d} &= [0_d; 0_d; ...; 0_{d-1}; 1] \end{aligned} \tag{2}$$

and the following matrices $V_l \in \mathbb{R}^{dN \times N}$

$$\begin{aligned} V_1 &= [c_{1,1}\, c_{2,1} \, ... \, c_{N,1}] \\ V_2 &= [c_{1,2}\, c_{2,2} \, ... \, c_{N,2}] \\ &\vdots \\ V_d &= [c_{1,d}\, c_{2,d} \, ... \, c_{N,d}]. \end{aligned} \tag{3}$$

A meaningful objective for dimensionality reduction is to find the points z_i as the solution to

$$\min_{z_i \in \mathbb{R}^d} \; -\frac{\gamma}{2} \sum_{i=1}^{N} \|z_i\|_2^2 + \frac{1}{2} \sum_{i=1}^{N} \|z_i - \sum_{j=1}^{N} s_{ij} z_j\|_2^2 \tag{4}$$

where the first term is a regularization term to avoid the trivial solution. The second term minimizes the objective $\sum_{i=1}^{N} \|z_i - \hat{z}_i\|_2^2$ with estimated coordinates

$$\hat{z}_i = \sum_{j=1}^{N} s_{ij} z_j \tag{5}$$

based on similarities s_{ij} defined on pairs of input data x_i, x_j. A typical choice is $s_{ij} = \exp(-\|x_i - x_j\|_2^2/\sigma^2)$. This second term is a similar criterion as used in locally linear embedding methods [16]. Written in terms of the unknown vector z the objective (4) becomes

$$\min_{z \in \mathbb{R}^{dN}} \; J(z) = -\frac{\gamma}{2} z^T z + \frac{1}{2}(z - Pz)^T (z - Pz) \tag{6}$$

where

$$P = \begin{bmatrix} s_{11} I_d & s_{12} I_d & ... & s_{1N} I_d \\ s_{21} I_d & s_{22} I_d & ... & s_{2N} I_d \\ \vdots & & & \vdots \\ s_{N1} I_d & s_{N2} I_d & ... & s_{NN} Id \end{bmatrix}.$$

Setting $\partial J / \partial z = 0$ yields the eigenvalue problem

$$Rz = \gamma z \tag{7}$$

with $R = (I - P)^T (I - P)$ from which one has to select one of the candidate eigenvector solutions. We are interested now in studying the use of kernel maps in order to project the original data with coordinates x_i to the coordinates z_i in the lower dimensional space.

3 Kernel maps and eigenvalue problem

We can realize the nonlinear mapping g through a least squares support vector machine regression combined with (4). This results in the following primal problem

$$\begin{aligned} \min_{z, w_j, e_{i,j}} \quad & J_2 = -\frac{\gamma}{2} z^T z + \frac{1}{2}(z - Pz)^T (z - Pz) + \frac{\nu}{2} \sum_{j=1}^{d} w_j^T w_j + \frac{\eta}{2} \sum_{i=1}^{N} \sum_{j=1}^{d} e_{i,j}^2 \\ \text{such that} \quad & c_{i,j}^T z = w_j^T \varphi_j(x_i) + e_{i,j}, \quad \forall i = 1, ..., N; j = 1, ..., d \end{aligned} \tag{8}$$

with $w_j \in \mathbb{R}^{n_{h_j}}$ and error variables $e_{i,j}$. The regularization constants ν, η are assumed to be positive. Different feature maps $\varphi_j(\cdot) : \mathbb{R}^p \to \mathbb{R}^{n_{h_j}}$ are used in order to realize mappings from the given input space to the different components in the lower dimensional space, where n_{h_j} denote the dimensions of the feature maps. A positive definite kernel function $K(\cdot,\cdot) : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$ is related to a feature map $\varphi(\cdot) : \mathbb{R}^p \to \mathbb{R}^{n_h}$ through $K(x,z) = \varphi(x)^T\varphi(z)$ (which is often called the *kernel trick*). For the commonly used radial basis function kernel $K(x,z) = \exp(-\|x-z\|_2^2/\sigma^2)$ the feature map is infinite dimensional ($n_h \to \infty$). The constrained optimization problem (8) is the primal problem with as unknowns the coordinates z of the data in the lower dimensional space, the error variables $e_{i,j}$ and the vector w_j for the primal representation of the model

$$\hat{z}_{*,j} = w_j^T \varphi_j(x^*) \tag{9}$$

evaluated at any point $x^* \in \mathbb{R}^p$ in the input space. It yields the corresponding predicted coordinates $\hat{z}_{*,j}$. Neural networks interpretations with a hidden layer and hidden units can be given both to this primal representation of the model (in terms of the feature map) and to the dual representation of the model (in terms of the kernel function) [24].

Note that the objective function J from (6) is acting now as an additional regularization term to the least squares support vector machine core part. In a different context, an additional Laplacian based regularization term has also been considered in [3] for Laplacian regularized least squares methods with function estimation in a reproducing kernel Hilbert space. In order to show the difference between (6) and (8) in the solutions we consider the zero bias term case.

In order to fix the ideas let us consider the case $d = 2$ where the given input data are to be projected and visualized in a two-dimensional space:

(**Problem P1**)

$$\begin{aligned} \min_{z,w_1,w_2,e_{i,1},e_{i,2}} \quad & J_3 = -\frac{\gamma}{2} z^T z + \frac{1}{2}(z-Pz)^T(z-Pz) + \frac{\nu}{2}\left(w_1^T w_1 + w_2^T w_2\right) + \frac{\eta}{2}\sum_{i=1}^{N}(e_{i,1}^2 + e_{i,2}^2) \\ \text{such that} \quad & c_{i,1}^T z = w_1^T \varphi_1(x_i) + e_{i,1}, \quad \forall i = 1,...,N \\ & c_{i,2}^T z = w_2^T \varphi_2(x_i) + e_{i,2}, \quad \forall i = 1,...,N. \end{aligned} \tag{10}$$

The solution to this problem can be viewed as a regularized version of (7) in the following sense.

Lemma 1. The solution to the problem (10) is given by the eigenvalue problem

$$\left(R + V_1(\frac{1}{\nu}\Omega_1 + \frac{1}{\eta}I)^{-1}V_1^T + V_2(\frac{1}{\nu}\Omega_2 + \frac{1}{\eta}I)^{-1}V_2^T\right) z = \gamma z \tag{11}$$

with $R = (I-P)^T(I-P)$ and kernel matrices $\Omega_1, \Omega_2 \in \mathbb{R}^{N\times N}$ with ij-th entries $\Omega_{1,ij} = K_1(x_i, x_j) = \varphi_1(x_i)^T\varphi_1(x_j)$, $\Omega_{2,ij} = K_2(x_i, x_j) = \varphi_2(x_i)^T\varphi_2(x_j)$ for $i, j = 1, ..., N$ and positive definite kernel functions $K_1(\cdot,\cdot), K_2(\cdot,\cdot)$.

Proof: see Appendix

□

Corollary 1. The dual representations of the model, expressed in terms of the Lagrange multipliers $\alpha_{i,1}, \alpha_{i,2}$ (corresponding to the constraints in (10)), are given by

$$\begin{aligned} \hat{z}_{*,1} &= \frac{1}{\nu}\sum_{i=1}^{N}\alpha_{i,1}K_1(x_i, x^*) \\ \hat{z}_{*,2} &= \frac{1}{\nu}\sum_{i=1}^{N}\alpha_{i,2}K_2(x_i, x^*) \end{aligned} \tag{12}$$

where $\alpha_{i,1}, \alpha_{i,2}$ are the unique solution to the linear systems

$$(\frac{1}{\nu}\Omega_1 + \frac{1}{\eta}I)\alpha_1 = V_1^T z \text{ and } (\frac{1}{\nu}\Omega_2 + \frac{1}{\eta}I)\alpha_2 = V_2^T z \tag{13}$$

with z a solution to (11) and $\alpha_1 = [\alpha_{1,1}; ...; \alpha_{N,1}]$, $\alpha_2 = [\alpha_{1,2}; ...; \alpha_{N,2}]$.

The eigenvalue problem (11) has $2N$ eigenvector solutions corresponding to the eigenvalues γ. However, it turns out that the selection of the best solution from this pool of $2N$ candidates is not straightforward. Selection of a best solution interacts in fact with the model selection problem for ν, η and the kernel tuning parameters. The best solution is often not corresponding with the largest or smallest eigenvalue γ in (11).

4 Main result: linear system solution obtained from reference point constraints

The constrained optimization problem formulations in least squares support vector machines enable to incorporate additional constraints in a straightforward manner. We will

demonstrate now how adding reference point constraints converts the eigenvalue problem (11) into a new problem where the solution is given by a linear system. We first consider projection of the input data to a two-dimensional space. Next we consider the general case.

4.1 Projection to a two-dimensional space

The main idea is to define one of the data points of the training set as a *reference point* for which the coordinates in the low dimensional space are chosen (approximately) and fixed beforehand. We assume here (without loss of generality) that this is the first data point of the given training set. The new problem formulation becomes

(**Problem P2**)

$$\begin{aligned}
\min_{z,w_1,w_2,b_1,b_2,e_{i,1},e_{i,2}} \quad & J_4 = -\frac{\gamma}{2} z^T z + \frac{1}{2}(z - P_D z)^T (z - P_D z) + \frac{\nu}{2}\left(w_1^T w_1 + w_2^T w_2\right) + \frac{\eta}{2}\sum_{i=1}^{N}(e_{i,1}^2 + e_{i,2}^2) \\
\text{such that} \quad & c_{1,1}^T z = q_1 + e_{1,1} \\
& c_{1,2}^T z = q_2 + e_{1,2} \\
& c_{i,1}^T z = w_1^T \varphi_1(x_i) + b_1 + e_{i,1}, \quad \forall i = 2, ..., N \\
& c_{i,2}^T z = w_2^T \varphi_2(x_i) + b_2 + e_{i,2}, \quad \forall i = 2, ..., N.
\end{aligned} \tag{14}$$

In this constrained optimization problem the core part is the LS-SVM mapping of the input data to the coordinates in the two-dimensional space. The first two terms of the objective function take into account neighborhood preservation. More flexibility is incorporated by modifying (6) into $\sum_{i=1}^{N} \|z_i - \sum_{j=1}^{N} s_{ij} D z_j\|_2^2 = (z - P_D z)^T (z - P_D z)$ with diagonal matrix $D \in \mathbb{R}^{d \times d}$ and

$$P_D = \begin{bmatrix} s_{11}D & s_{12}D & ... & s_{1N}D \\ s_{21}D & s_{22}D & ... & s_{2N}D \\ \vdots & & & \vdots \\ s_{N1}D & s_{N2}D & ... & s_{NN}D \end{bmatrix}.$$

For the first training data point x_1 *fictitious* coordinates of the reference point $q = [q_1; q_2] \in \mathbb{R}^2$ are specified by the user. Technically speaking, this point q makes the problem in fact semi-supervised instead of unsupervised (in an artificial way). On the other hand, this reference point needs to be sacrificed in the final visualization of the coordinates

z_i. Furthermore, additional bias terms b_1, b_2 are taken in the model. The solution is characterized now as follows.

Lemma 2. Assuming $\gamma \leq 0$ and $\nu, \eta > 0$, the unique solution to the problem (14) is given by the linear system

$$\left[\begin{array}{c|c|c} U & -V_1 M_1^{-1} 1 & -V_2 M_2^{-1} 1 \\ \hline -1^T M_1^{-1} V_1^T & 1^T M_1^{-1} 1 & 0 \\ \hline -1^T M_2^{-1} V_2^T & 0 & 1^T M_2^{-1} 1 \end{array}\right] \left[\begin{array}{c} z \\ \hline b_1 \\ \hline b_2 \end{array}\right] = \left[\begin{array}{c} \eta(q_1 c_{1,1} + q_2 c_{1,2}) \\ \hline 0 \\ \hline 0 \end{array}\right] \tag{15}$$

with matrices

$$\begin{aligned} U &= (I - P_D)^T (I - P_D) - \gamma I + V_1 M_1^{-1} V_1^T + V_2 M_2^{-1} V_2^T + \eta c_{1,1} c_{1,1}^T + \eta c_{1,2} c_{1,2}^T \\ M_1 &= \frac{1}{\nu}\Omega_1 + \frac{1}{\eta} I \,,\; M_2 = \frac{1}{\nu}\Omega_2 + \frac{1}{\eta} I \\ V_1 &= [c_{2,1} \ldots c_{N,1}] \,,\; V_2 = [c_{2,2} \ldots c_{N,2}] \end{aligned}$$

and kernel matrices $\Omega_1, \Omega_2 \in \mathbb{R}^{(N-1)\times(N-1)}$ with ij-th entries $\Omega_{1,ij} = K_1(x_i, x_j) = \varphi_1(x_i)^T \varphi_1(x_j)$, $\Omega_{2,ij} = K_2(x_i, x_j) = \varphi_2(x_i)^T \varphi_2(x_j)$ for $i, j = 2, ..., N$ and positive definite kernel functions $K_1(\cdot,\cdot), K_2(\cdot,\cdot)$.

Proof: see Appendix

□

Corollary 2. The primal and dual model representations allow making out-of-sample extensions. Evaluated at point $x^* \in \mathbb{R}^p$ the predicted coordinates are

$$\begin{aligned} \hat{z}_{*,1} &= w_1^T \varphi_1(x^*) + b_1 = \frac{1}{\nu} \sum_{i=2}^{N} \alpha_{i,1} K_1(x_i, x^*) + b_1 \\ \hat{z}_{*,2} &= w_2^T \varphi_2(x^*) + b_2 = \frac{1}{\nu} \sum_{i=2}^{N} \alpha_{i,2} K_2(x_i, x^*) + b_2 \end{aligned} \tag{16}$$

where b_1, b_2 are the solution to (15) and $\alpha_1, \alpha_2 \in \mathbb{R}^{N-1}$ are the unique solutions to the linear systems

$$M_1 \alpha_1 = V_1^T z - b_1 1_{N-1} \;\text{ and }\; M_2 \alpha_2 = V_2^T z - b_2 1_{N-1} \tag{17}$$

with z the solution to (15) and $\alpha_1 = [\alpha_{2,1}; ...; \alpha_{N,1}]$, $\alpha_2 = [\alpha_{2,2}; ...; \alpha_{N,2}]$, $1_{N-1} = [1; 1; ...; ; 1]$. In the sequel in all experiments $\gamma = 0$ will be taken.

4.2 General case

In the general case with dimensionality reduction to a d-dimensional space the problem statement is

(**Problem P3**)

$$\begin{aligned} \min_{z,w_j,b_j,e_{i,j}} \quad & J_5 = -\frac{\gamma}{2} z^T z + \frac{1}{2}(z - P_D z)^T (z - P_D z) + \frac{\nu}{2} \sum_{j=1}^{d} w_j^T w_j + \frac{\eta}{2} \sum_{i=1}^{N} \sum_{j=1}^{d} e_{i,j}^2 \\ \text{such that} \quad & c_{1,j}^T z = q_j + e_{1,j}, \quad \forall j = 1, ..., d \\ & c_{i,j}^T z = w_j^T \varphi_j(x_i) + b_j + e_{i,j}, \quad \forall i = 2, ..., N \,;\, j = 1, ..., d. \end{aligned} \tag{18}$$

The form of the solution is similar to the two-dimensional case.

Lemma 3. Assuming $\gamma \leq 0$ and $\nu, \eta > 0$, the unique solution to the problem (18) is given by the linear system

$$\left[\begin{array}{c|c|c|c} U & -V_1 M_1^{-1} 1 & ... & -V_d M_d^{-1} 1 \\ \hline -1^T M_1^{-1} V_1^T & 1^T M_1^{-1} 1 & 0 & 0 \\ \hline \vdots & 0 & \ddots & 0 \\ \hline -1^T M_d^{-1} V_d^T & 0 & 0 & 1^T M_d^{-1} 1 \end{array}\right] \left[\begin{array}{c} z \\ \hline b_1 \\ \hline \vdots \\ \hline b_d \end{array}\right] = \left[\begin{array}{c} \eta \sum_{j=1}^{d} q_j c_{1,j} \\ \hline 0 \\ \hline \vdots \\ \hline 0 \end{array}\right] \tag{19}$$

with

$$U = (I - P_D)^T (I - P_D) - \gamma I + \sum_{j=1}^{d} V_j M_j^{-1} V_j^T + \eta \sum_{j=1}^{d} c_{1,j} c_{1,j}^T$$

$$M_j = \frac{1}{\nu} \Omega_j + \frac{1}{\eta} I \,,\, V_j = [c_{2,j} \, ... \, c_{N,j}] \,,\, j = 1, ..., d$$

and kernel matrices $\Omega_j \in \mathbb{R}^{(N-1)\times(N-1)}$ with ij-th entries $\Omega_{j,kl} = K_j(x_k, x_l) = \varphi_j(x_k)^T \varphi_j(x_l)$, for $j = 1, ..., d$ and $k, l = 2, ..., N$ and positive definite kernel functions $K_j(\cdot, \cdot)$.

Proof: see Appendix

□

Corollary 3. The primal and dual model representations allow to make out-of-sample extensions. Evaluated at point $x^* \in \mathbb{R}^p$ the predicted coordinates are

$$\hat{z}_{*,j} = w_j^T \varphi_j(x^*) + b_j = \frac{1}{\nu} \sum_{i=2}^{N} \alpha_{i,j} K_j(x_i, x^*) + b_j \tag{20}$$

where the bias terms b_j are the solution to (19) and $\alpha_j \in \mathbb{R}^{N-1}$ are the unique solutions to the linear systems

$$M_j \alpha_j = V_j^T z - b_j 1_{N-1} \tag{21}$$

with z the solution to (19) and $\alpha_j = [\alpha_{2,j}; ...; \alpha_{N,j}]$ for $j = 1, ..., d$.

For moderate size data sets the linear systems can be solved using direct methods [12]. For larger scale problems iterative solvers such as conjugate gradient methods can be used [12]. However, for conjugate gradient methods the matrix of the problem needs to be positive definite. The problems (15) and (19) can be converted then into problems with a positive definite matrix [24], similar as for LS-SVM models in classification and regression with bias terms. The computational complexity is well understood in terms of the condition number of the matrix and the required precision [21].

5 Model selection by validation

Let us discuss now the selection of tuning parameters in the context of validation for use of a single validation set or cross-validation. The tuning parameters are determined here according to the criterion

$$\min \sum_{i,j} \left(\frac{\hat{z}_i^T \hat{z}_j}{\|\hat{z}_i\|_2 \|\hat{z}_j\|_2} - \frac{x_i^T x_j}{\|x_i\|_2 \|x_j\|_2} \right)^2 \tag{22}$$

evaluated on validation data.

While the coordinates z, the support values α and bias terms b follow from solving linear systems, the following tuning parameters are determined based on (22):

- Kernels tuning parameters in s_{ij}, $K_1, K_2, (K_3)$:

  When using radial basis functions in $s_{ij} = \exp(-\|x_i - x_j\|_2^2/\sigma^2)$, $K_j(x, z) = \exp(-\|x - z\|_2^2/\sigma_j^2)$ for $j = 1, .., d$ the kernel parameters σ, σ_j need to be determined. Especially the determination of a good range for these values turns out to be relevant as well as the values relative with respect to each other.

- Regularization constants ν, η:

The determination of a good range for the values ν, η is needed. However, experiments indicate that the values are usually not very critical. The regularization constant $\gamma = 0$ can be set in practice.

- Choice of the diagonal matrix D:

  The results are usually considerably improved by taking the diagonal matrix D different from the identity matrix. This scaling aspect is relevant for achieving a good visualization.

- Choice of the reference point q:

  The choice of the point q is like choosing a geometric perspective on an object. In the 2D case we recommend testing the following candidate reference points $q \in \{[+1; +1], [+1; -1], [-1; +1], [-1, -1]\}$. A suitable choice of this point may lead to a better visualization and improved generalization results.

It is also recommended to scale the input data properly before applying the method.

6 Examples

6.1 Spiral problem

In this first toy problem we illustrate the good performance of the kernel map reference point method in terms of generalization after determination of the tuning parameters on a single validation set.

For this spiral problem the data were generated in Matlab (see supplementary material), scaled to ranges $[-1, 1]$ and made zero mean. The given input data are in a 3-dimensional space. The problem P2 was considered for data visualization in 2D. The tuning parameter selection was done interactively based on (22) resulting into the kernel parameters $\sigma^2 = 100$, $\sigma_1^2 = 80$, $\sigma_2^2 = 0.1\sigma_1^2$, the regularization constants $\eta = 1$, $\nu = 0.5$ (and $\gamma = 0$ fixed beforehand), diagonal matrix $D = \text{diag}\{10, 1\}$ and reference point $q = [+1; -1]$.

Figure 1 shows the out-of-sample abilities on a validation and a test set with excellent generalization and a faithful representation in the lower dimensional space. The first

training data point x_1 has to be omitted for the visualization, because its projection lies outside the range of values of all other points.

Also the problem P1 with the solution characterized by the eigenvalue problem has been tested. All experiments indicated that the component selection was difficult in that case: the best solutions were often not corresponding to the largest or smallest eigenvalue in (11). As a result an exhaustive search through all $2N$ eigenvector solution candidates may be needed then together with a validation procedure for the optimal component selection (or by any other suitable model selection procedure), which is computationally much harder. Moreover, the conditioning of the numerical algorithms for computing the eigenvector solutions is inferior (even if one specifies to compute only a subset of the eigenvectors) with respect to the well-conditioned linear systems to be solved in problems P2 and P3.

6.2 Swiss roll data

A well-known example is the swiss roll problem [16]. We generated the input data in Matlab as shown in the supplementary material, scaled to ranges $[-1, 1]$ and made zero mean. The first 100 point are chosen as the validation set, the next 600 as the training data.

The problem P2 was considered for data visualization in 2D. The tuning parameter selection was done interactively based on (22) resulting into the kernel parameters $\sigma^2 = 200$, $\sigma_1^2 = 200$, $\sigma_2^2 = \sigma_1^2$, regularization constants $\eta = 1$, $\nu = 0.5$, diagonal matrix $D = \text{diag}\{10, 1\}$ and reference point $q = [+1; -1]$.

Figure 2 shows the 2D visualization with dimensionality reduction containing the training and validation part. Notice that the visualization result reveals *both* the rolling structure and the parallel aspects. Comparisons with other existing techniques were done in Matlab (using the software mentioned in the supplementary material) including MDS, PCA, isomap, LLE, Hessian LLE, Laplacian, diffusion map, LTSA. The result from diffusion maps looks closest to the kernel map with reference point method. Most other methods do not reveal the rolling structure in the visualization. The kernel map with reference point method realizes an optimality principle in the sense of generalization, corresponding to a solution that also visually provides the best results in terms of the revealed structure.

6.3 Alon colon cancer microarray data set

In this example we aim at visualizing the gene distribution for the Alon colon cancer microarray data set (for further information see the supplementary material). In total 1500 genes are taken: the first 500 ones for the validation part, the next 500 for training and the last 500 for testing. For the given microarray data set the dimension of the input space is $p = 62$.

Both problems P2 and P3 are considered with visualization in 2D and 3D. Figures 3 and 4 reveal the shape of the underlying distribution where the training, validation and test genes are shown. For problem P2 the following tuning parameters were obtained from (22): kernel parameters $\sigma^2 = 10^4$, $\sigma_1^2 = 10^3$, $\sigma_2^2 = 0.1\sigma_1^2$, the regularization constants $\eta = 1$, $\nu = 100$, diagonal matrix $D = \text{diag}\{10, 1\}$ and reference point $q = [+1; -1]$. For problem P3 the following tuning parameters were obtained: kernel parameters $\sigma^2 = 10^4$, $\sigma_1^2 = 10^3$, $\sigma_2^2 = 0.5\sigma_1^2$, $\sigma_3^2 = 0.1\sigma_1^2$, the regularization constants $\eta = 1$, $\nu = 100$, diagonal matrix $D = \text{diag}\{10, 5, 1\}$, and reference point $q = [+1; -1; -1]$.

6.4 Santa Fe chaotic laser data

In this example we visualize the time-series $\{y_t\}_{t=1}^T$ of the Santa Fe chaotic laser data set by defining the vectors $y_{t|t-m} = [y_t; y_{t-1}; y_{t-2}; ...; y_{t-m}]$ with $m = 9$. The data set consists then of the points $\{y_{t|t-m}\}_{t=m+1}^{t=m+N_{tot}}$ with 200 validation data (first part) and 700 training data points (next part) (see Figure 5) in a $p = 10$ dimensional input space.

Both problems P2 and P3 are considered with visualization in 2D and 3D. The results in Figures 6 and 7 show the underlying attractor. For problem P2 the following tuning parameters were obtained from (22): kernel parameters $\sigma^2 = 400$, $\sigma_1^2 = 9$, $\sigma_2^2 = 0.1\sigma_1^2$, the regularization constants $\eta = 1$, $\nu = 10$, diagonal matrix $D = \text{diag}\{10, 1\}$ and reference point $q = [+1; -1]$. For problem P3 the following tuning parameters were obtained: kernel parameters $\sigma^2 = 400$, $\sigma_1^2 = 9$, $\sigma_2^2 = 0.5\sigma_1^2$, $\sigma_3^2 = 0.1\sigma_1^2$, the regularization constants $\eta = 1$, $\nu = 10$, diagonal matrix $D = \text{diag}\{10, 5, 1\}$, and reference point $q = [+1; -1; -1]$.

7 Conclusions

In recent years considerable progress has been made in the area of data visualization and dimensionality reduction using kernel eigenmap methods. In this paper we have shown how to characterize solutions by linear systems instead of eigenvalue problems. For this purpose we proposed a new method of kernel maps with a reference point. The framework of least squares support vector machines is adopted which naturally allows to incorporate reference point constraints in the primal problem within the constrained optimization problem. Kernel model representations are generated in a systematic and straightforward way in the dual. The optimal coordinates for visualization in the low dimensional space are computed from a linear system in a numerically reliable way. The framework immediately allows for out-of-sample extensions. Validation based learning of the tuning parameters has been demonstrated with good results on toy problems and real-life data sets.

Acknowledgements

Research supported by Research Council K.U. Leuven: GOA AMBioRICS, CoE EF/05/006, OT/03/12, PhD/postdoc & fellow grants; Flemish Government: FWO PhD/postdoc grants, FWO projects G.0499.04, G.0211.05, G.0226.06, G.0302.07; Research communities (ICCoS, ANMMM, MLDM); AWI: BIL/05/43, IWT: PhD Grants; Belgian Federal Science Policy Office: IUAP P5/22, IUAP DYSCO.

Supplementary material

A Matlab demo file that shows the application of the kernel map reference point method to the Swiss roll problem can be downloaded from:

http://www.esat.kuleuven.ac.be/sista/lssvmlab/KMref/demoswissKMref.m.

For comparisons with other methods including MDS, PCA, ISOMAP, LLE, Hessian LLE, Laplacian, diffusion map, LTSA one can consult:

http://www.math.umn.edu/∼wittman/mani/

http://www.cs.toronto.edu/∼roweis/lle/code.html.

Spiral data Matlab code:

t=0.01:0.01:4;x=cos(7*t);y=sin(7*t);z=t

Swiss roll data Matlab code:

Ntot=700;tt =(3*pi/2)*(1+2*(1:1:Ntot)/Ntot);height=2*rand(1,Ntot);X=[tt.*cos(tt);height;tt.*sin(tt)]

http://www.cs.toronto.edu/∼roweis/lle/code/swissroll.m

Alon colon cancer microarray data set:

information and links at http://homes.esat.kuleuven.be/∼npochet/Bioinformatics/

Santa Fe laser data:

http://www-psych.stanford.edu/∼andreas/Time-Series/SantaFe.html

Appendix

Proof of Lemma 1

The solution (11) is obtained in a straightforward way. First construct the Lagrangian

$$\mathcal{L}(z, w_1, w_2, e_{i,1}, e_{i,2}; \alpha_{i,1}, \alpha_{i,2}) = J_3 + \sum_{i=1}^{N} \alpha_{i,1}(c_{i,1}^T z - w_1^T \varphi_1(x_i) - e_{i,1}) + \sum_{i=1}^{N} \alpha_{i,2}(c_{i,2}^T z - w_2^T \varphi_2(x_i) - e_{i,2}).$$

For characterization of the local minima, take the conditions for optimality $\partial\mathcal{L}/\partial z = 0, \partial\mathcal{L}/\partial w_1 = 0, \partial\mathcal{L}/\partial w_2 = 0, \partial\mathcal{L}/\partial e_{i,1} = 0, \partial\mathcal{L}/\partial e_{i,2} = 0, \partial\mathcal{L}/\partial\alpha_{i,1} = 0, \partial\mathcal{L}/\partial\alpha_{i,2} = 0$. This leads to the following set of equations which should be satisfied simultaneously

$$\begin{cases} \frac{\partial\mathcal{L}}{\partial z} &= -\gamma z + Rz + \sum_{i=1}^{N} \alpha_{i,1} c_{i,1} + \sum_{i=1}^{N} \alpha_{i,2} c_{i,2} = 0 \\ \frac{\partial\mathcal{L}}{\partial w_1} &= \nu w_1 - \sum_{i=1}^{N} \alpha_{i,1} \varphi_1(x_i) = 0 \\ \frac{\partial\mathcal{L}}{\partial w_2} &= \nu w_2 - \sum_{i=1}^{N} \alpha_{i,2} \varphi_2(x_i) = 0 \\ \frac{\partial\mathcal{L}}{\partial e_{i,1}} &= \eta e_{i,1} - \alpha_{i,1} = 0\,,\ i = 1, ..., N \\ \frac{\partial\mathcal{L}}{\partial e_{i,2}} &= \eta e_{i,2} - \alpha_{i,2} = 0\,,\ i = 1, ..., N \\ \frac{\partial\mathcal{L}}{\partial \alpha_{i,1}} &= c_{i,1}^T z - w_1^T \varphi_1(x_i) - e_{i,1}\,,\ i = 1, ..., N \\ \frac{\partial\mathcal{L}}{\partial \alpha_{i,2}} &= c_{i,2}^T z - w_2^T \varphi_2(x_i) - e_{i,2}\,,\ i = 1, ..., N. \end{cases}$$

After elimination of $w_1, w_2, e_{i,1}, e_{i,2}$ and application of the kernel trick the set of equations can be expressed in terms of z, α_1, α_2. One obtains

$$-\gamma z + Rz + V_1\alpha_1 + V_2\alpha_2 = 0$$

and

$$\begin{aligned} V_1^T z - \tfrac{1}{\nu}\Omega_1\alpha_1 - \tfrac{1}{\eta}\alpha_1 = 0 \\ V_2^T z - \tfrac{1}{\nu}\Omega_2\alpha_2 - \tfrac{1}{\eta}\alpha_2 = 0. \end{aligned}$$

The dual model representation follows from the conditions for optimality.

Proof of Lemma 2

The solution (15) is obtained in a straightforward way. Construct the Lagrangian

$$\begin{aligned} \mathcal{L}(z, w_1, w_2, b_1, b_2, e_{i,1}, e_{i,2}; \beta_{1,1}, \beta_{1,2}, \alpha_{i,1}, \alpha_{i,2}) = J_4 + \beta_{1,1}(c_{1,1}^T z - q_1 - e_{1,1}) + \beta_{1,2}(c_{1,2}^T z - q_2 - e_{1,2}) \\ + \sum_{i=2}^{N} \alpha_{i,1}(c_{i,1}^T z - w_1^T \varphi_1(x_i) - b_1 - e_{i,1}) + \sum_{i=2}^{N} \alpha_{i,2}(c_{i,2}^T z - w_2^T \varphi_2(x_i) - b_2 - e_{i,2}). \end{aligned}$$

Taking the conditions for optimality [11] $\partial\mathcal{L}/\partial z = 0, \partial\mathcal{L}/\partial w_1 = 0, \partial\mathcal{L}/\partial w_2 = 0, \partial\mathcal{L}/\partial b_1 = 0, \partial\mathcal{L}/\partial b_2 = 0, \partial\mathcal{L}/\partial e_{i,1} = 0, \partial\mathcal{L}/\partial e_{i,2} = 0, \partial\mathcal{L}/\partial \beta_{1,1} = 0, \partial\mathcal{L}/\partial \beta_{1,2} = 0, \partial\mathcal{L}/\partial \alpha_{i,1} = 0, \partial\mathcal{L}/\partial \alpha_{i,2} = 0$ this gives the following set of equations which should be satisfied simultaneously

$$\begin{cases} \frac{\partial\mathcal{L}}{\partial z} &= -\gamma z + (I - P_D)^T(I - P_D)z + \beta_{1,1}c_{1,1} + \beta_{1,2}c_{1,2} + \sum_{i=2}^N \alpha_{i,1}c_{i,1} + \sum_{i=2}^N \alpha_{i,2}c_{i,2} = 0 \\ \frac{\partial\mathcal{L}}{\partial w_1} &= \nu w_1 - \sum_{i=2}^N \alpha_{i,1}\varphi_1(x_i) = 0 \\ \frac{\partial\mathcal{L}}{\partial w_2} &= \nu w_2 - \sum_{i=2}^N \alpha_{i,2}\varphi_2(x_i) = 0 \\ \frac{\partial\mathcal{L}}{\partial b_1} &= \sum_{i=2}^N \alpha_{i,1} = 1_{N-1}^T\alpha_1 = 0 \\ \frac{\partial\mathcal{L}}{\partial b_2} &= \sum_{i=2}^N \alpha_{i,2} = 1_{N-1}^T\alpha_2 = 0 \\ \frac{\partial\mathcal{L}}{\partial e_{1,1}} &= \eta e_{1,1} - \beta_{1,1} = 0 \\ \frac{\partial\mathcal{L}}{\partial e_{1,2}} &= \eta e_{1,2} - \beta_{1,2} = 0 \\ \frac{\partial\mathcal{L}}{\partial e_{i,1}} &= \eta e_{i,1} - \alpha_{i,1} = 0\,,\ i = 2, ..., N \\ \frac{\partial\mathcal{L}}{\partial e_{i,2}} &= \eta e_{i,2} - \alpha_{i,2} = 0\,,\ i = 2, ..., N \\ \frac{\partial\mathcal{L}}{\partial \beta_{1,1}} &= c_{1,1}^T z - q_1 - e_{1,1} = 0 \\ \frac{\partial\mathcal{L}}{\partial \beta_{1,2}} &= c_{1,2}^T z - q_2 - e_{1,2} = 0 \\ \frac{\partial\mathcal{L}}{\partial \alpha_{i,1}} &= c_{i,1}^T z - w_1^T\varphi_1(x_i) - b_1 - e_{i,1} = 0\,,\ i = 2, ..., N \\ \frac{\partial\mathcal{L}}{\partial \alpha_{i,2}} &= c_{i,2}^T z - w_2^T\varphi_2(x_i) - b_2 - e_{i,2} = 0\,,\ i = 2, ..., N. \end{cases}$$

After elimination of $w_1, w_2, e_{i,1}, e_{i,2}$ and application of the kernel trick the set of equations can be expressed in terms of $z, b_1, b_2, \alpha_1, \alpha_2$. One obtains

$$-\gamma z + (I - P_D)^T(I - P_D)z + V_1\alpha_1 + V_2\alpha_2 + \eta c_{1,1}c_{1,1}^T z + \eta c_{1,2}c_{1,2}^T z = \eta(q_1 c_{1,1} + q_2 c_{1,2})$$

and

$$\begin{aligned} &V_1^T z - \tfrac{1}{\nu}\Omega_1\alpha_1 - \tfrac{1}{\eta}\alpha_1 - b_1 1_{N-1} = 0 \\ &V_2^T z - \tfrac{1}{\nu}\Omega_2\alpha_2 - \tfrac{1}{\eta}\alpha_2 - b_2 1_{N-1} = 0 \\ &\beta_{1,1} = \eta(c_{1,1}^T z - q_1) \\ &\beta_{1,2} = \eta(c_{1,2}^T z - q_2). \end{aligned}$$

The dual model representation follows from the conditions for optimality.

Proof of Lemma 3

The proof is similar to the proof of Lemma 2.

References

[1] C. Alzate, J.A.K. Suykens, "A Weighted Kernel PCA Formulation with Out-of-Sample Extensions for Spectral Clustering Methods," in *Proc. of the 2006 International Joint Conference on Neural Networks* (IJCNN'06), Vancouver, Canada, pp. 138-144, 2006.

[2] M. Belkin, P. Niyogi, "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation," *Neural Computation*, 15(6): 1373-1396, June 2003.

[3] M. Belkin, P. Niyogi, V. Sindhwani, "Manifold Regularization: A Geometric Framework for Learning from Labeled and Unlabeled Examples," *Journal of Machine Learning Research*, 7: 2399–2434, 2006.

[4] Y. Bengio, J.-F. Paiement, P. Vincent, O. Delalleau, N. Le Roux and M. Ouimet, "Out-of-Sample Extensions for LLE, Isomap, MDS, Eigenmaps, and Spectral Clustering," In *Advances in Neural Information Processing Systems*, 16, 2004.

[5] G.C. Cawley, N.L.C. Talbot, "Fast exact leave-one-out cross-validation of sparse least-squares support vector machines," *Neural Networks*, 17(10): 1467-1475, 2004.

[6] R.R. Coifman, S. Lafon, A.B. Lee, M. Maggioni, B. Nadler, F. Warner, S.W. Zucker, "Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps," *Proc. Natl. Acad. Sci. U.S.A.*, 102(21): 7426-7431, May 2005.

[7] R.R. Coifman, S. Lafon, "Diffusion maps," *Applied and Computational Harmonic Analysis*, 21(1): 5-30, 2006.

[8] F. Cucker, S. Smale, "On the mathematical foundations of learning theory," *Bulletin of the AMS*, 39, 1–49, 2002.

[9] D.L. Donoho, C. Grimes, "Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data," *Proc. Natl. Acad. Sci. U.S.A*, 100(10): 5591-5596, May 2003.

[10] T. Evgeniou, M. Pontil, T. Poggio, "Regularization networks and support vector machines," *Advances in Computational Mathematics,* 13(1), 1–50, 2000.

[11] R. Fletcher, *Practical Methods of Optimization,* Chichester and New York: John Wiley & Sons, 1987.

[12] G.H. Golub, C.F. Van Loan, *Matrix Computations*, Baltimore MD: Johns Hopkins University Press, 1989.

[13] I.T. Jolliffe, *Principal Component Analysis*, Springer Series in Statistics, Springer-Verlag, 1986.

[14] G.E. Hinton, R.R. Salakhutdinov, "Reducing the Dimensionality of Data with Neural Networks," *Science*, 313(5786): 504–507, July 2006.

[15] T. Kohonen, "The self-organizing map," *Proceedings of the IEEE*, 78(9): 1464–1480, 1990.

[16] S. Roweis, L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, 290(5500), 2323–2326, Dec. 2000.

[17] C. Saunders, A. Gammerman, V. Vovk, "Ridge regression learning algorithm in dual variables," *Proc. of the 15th Int. Conf. on Machine Learning (ICML-98)*, Madison-Wisconsin, 515–521, 1998.

[18] B. Schölkopf, A. Smola, K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation,* 10, 1299–1319, 1998.

[19] B. Schölkopf, A. Smola, *Learning with Kernels*, MIT Press, Cambridge, MA, 2002.

[20] J. Shawe-Taylor, N. Cristianini, *Kernel Methods for Pattern Analysis*, Cambridge University Press, June 2004.

[21] S. Smale, "Complexity theory and numerical analysis," *Acta Numerica*, 523–551, 1997.

[22] J.A.K. Suykens, J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, 9(3), 293–300, 1999.

[23] J.A.K. Suykens, T. Van Gestel, J. Vandewalle, B. De Moor, "A support vector machine formulation to PCA analysis and its kernel version," *IEEE Transactions on Neural Networks*, 14(2): 447–450, March 2003.

[24] J.A.K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, J. Vandewalle, *Least Squares Support Vector Machines*, World Scientific, Singapore, 2002.

[25] J.B. Tenenbaum, V. De Silva, J.C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, 290(5500): 2319–2323, 2000.

[26] V. Vapnik, *Statistical Learning Theory,* John Wiley & Sons, New York, 1998.

[27] G. Wahba, *Spline Models for Observational Data*, Series in Applied Mathematics, 59, SIAM, Philadelphia, 1990.

[28] K.Q. Weinberger, L.K. Saul, "Unsupervised learning of image manifolds by semidefinite programming," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (CVPR-04), Washington D.C., 2004.

Captions of Figures

Figure 1: (Top) given input data of a spiral data set in a 3-dimensional space (training data (blue *), validation data (magenta o), test data (red +)); (Bottom) visualization with dimensionality reduction to a 2-dimensional space using the kernel map with reference point.

Figure 2: (Top) Swiss roll data set (containing 600 training data points and 100 validation set points); (Bottom) visualization with dimensionality reduction to a 2-dimensional space using the kernel map with reference point.

Figure 3: Alon colon cancer microarray data set: 2D visualization of the gene distribution obtained by the kernel map reference point method (training data (blue *), validation data (black o), test data (red +)).

Figure 4: Alon colon cancer microarray data set: 3D visualization of the gene distribution obtained by the kernel map reference point method (training data (blue *), validation data (black o), test data (red +)). Shown are two different perspectives (with the same reference point).

Figure 5: Santa Fe chaotic laser data: shown is the part used for training.

Figure 6: Santa Fe laser data: 2D visualization obtained by the kernel map reference point method.

Figure 7: Santa Fe laser data: 3D visualization obtained by the kernel map reference point method. Shown are two different perspectives (with the same reference point).

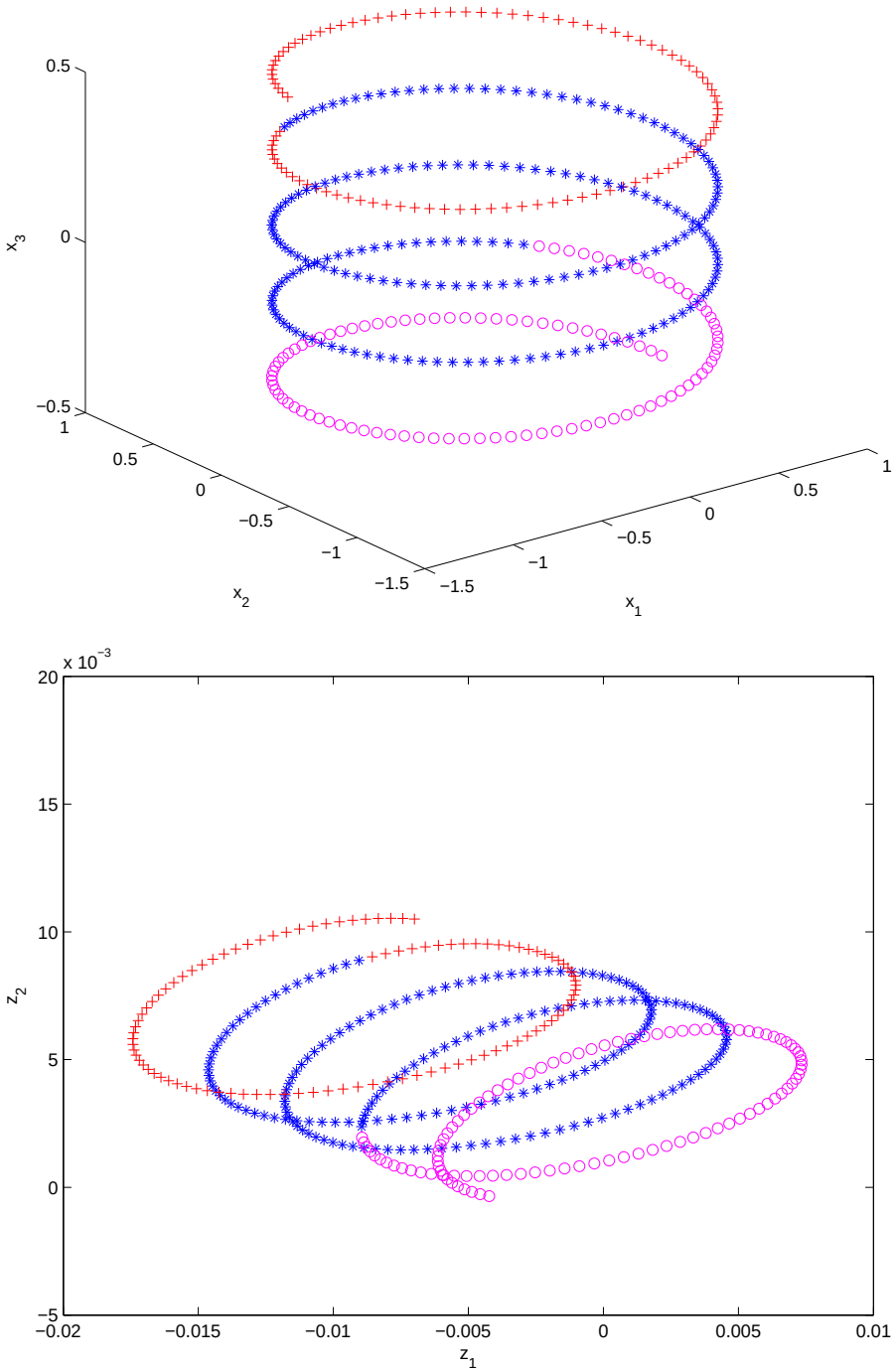

0.5
0
−0.5
x3
1
0.5
0
−0.5
−1
−1.5
x2
−1.5
−1
−0.5
0
0.5
1
x1
x 10−3
20
15
10
5
0
−5
z2
−0.02
−0.015
−0.01
−0.005
0
0.005
0.01
z1

Figure 1

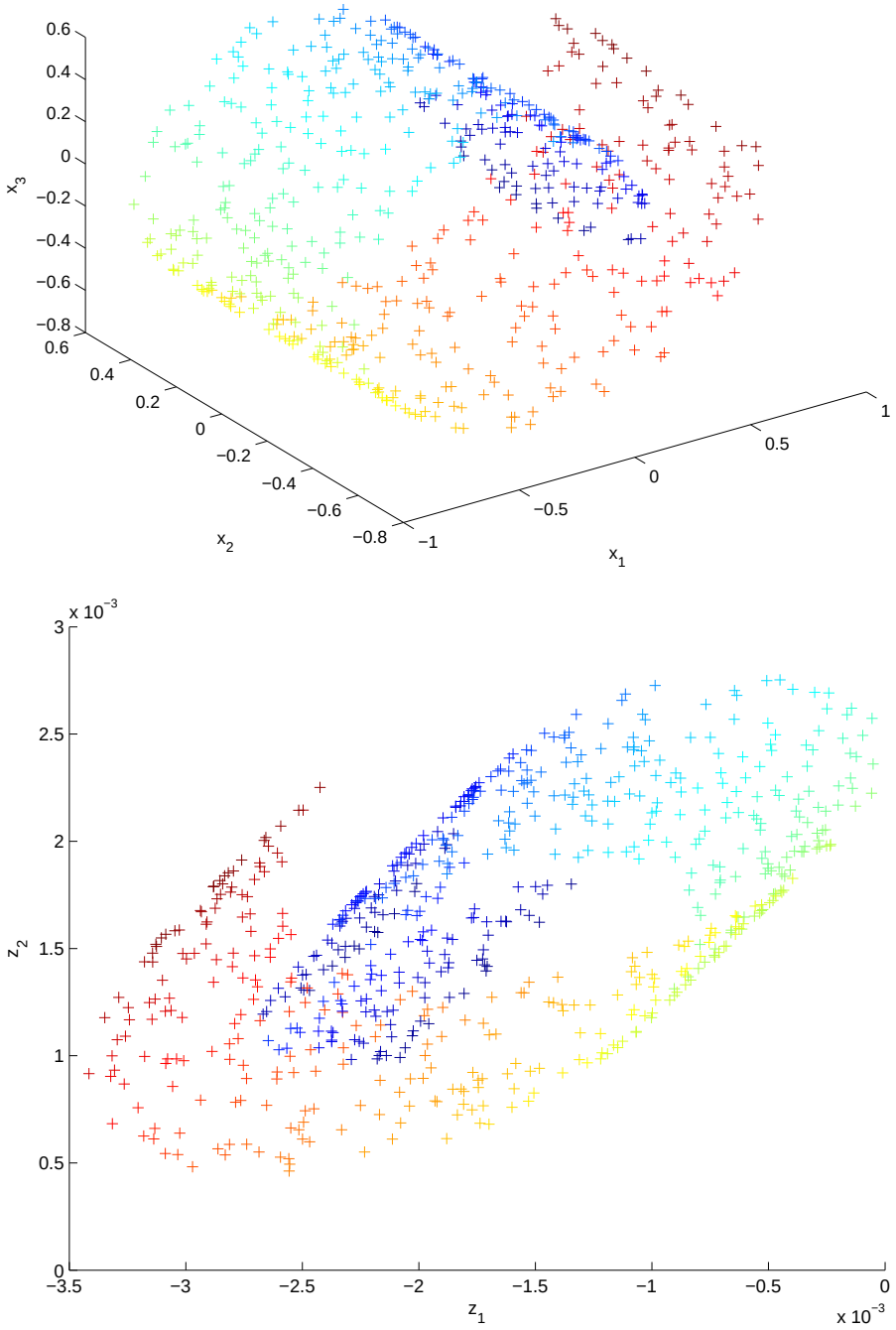

0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−0.8
x3
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−0.8
x2
−1
−0.5
0
0.5
1
x1
x 10−3
3
2.5
2
1.5
1
0.5
0
z2
−3.5
−3
−2.5
−2
−1.5
−1
−0.5
0
z1
x 10−3

Figure 2

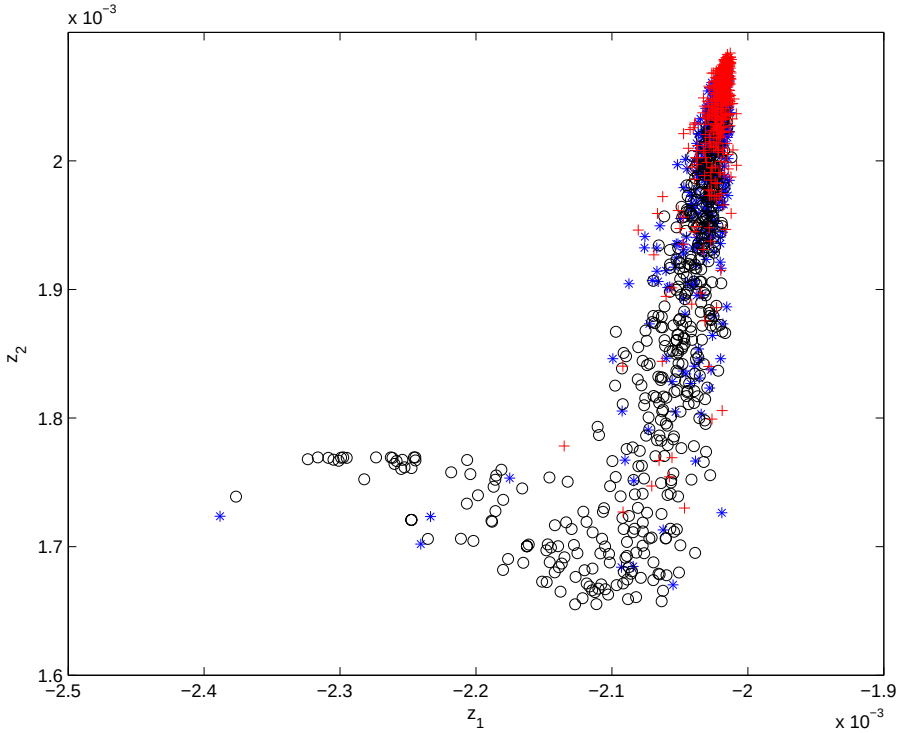


Figure 3

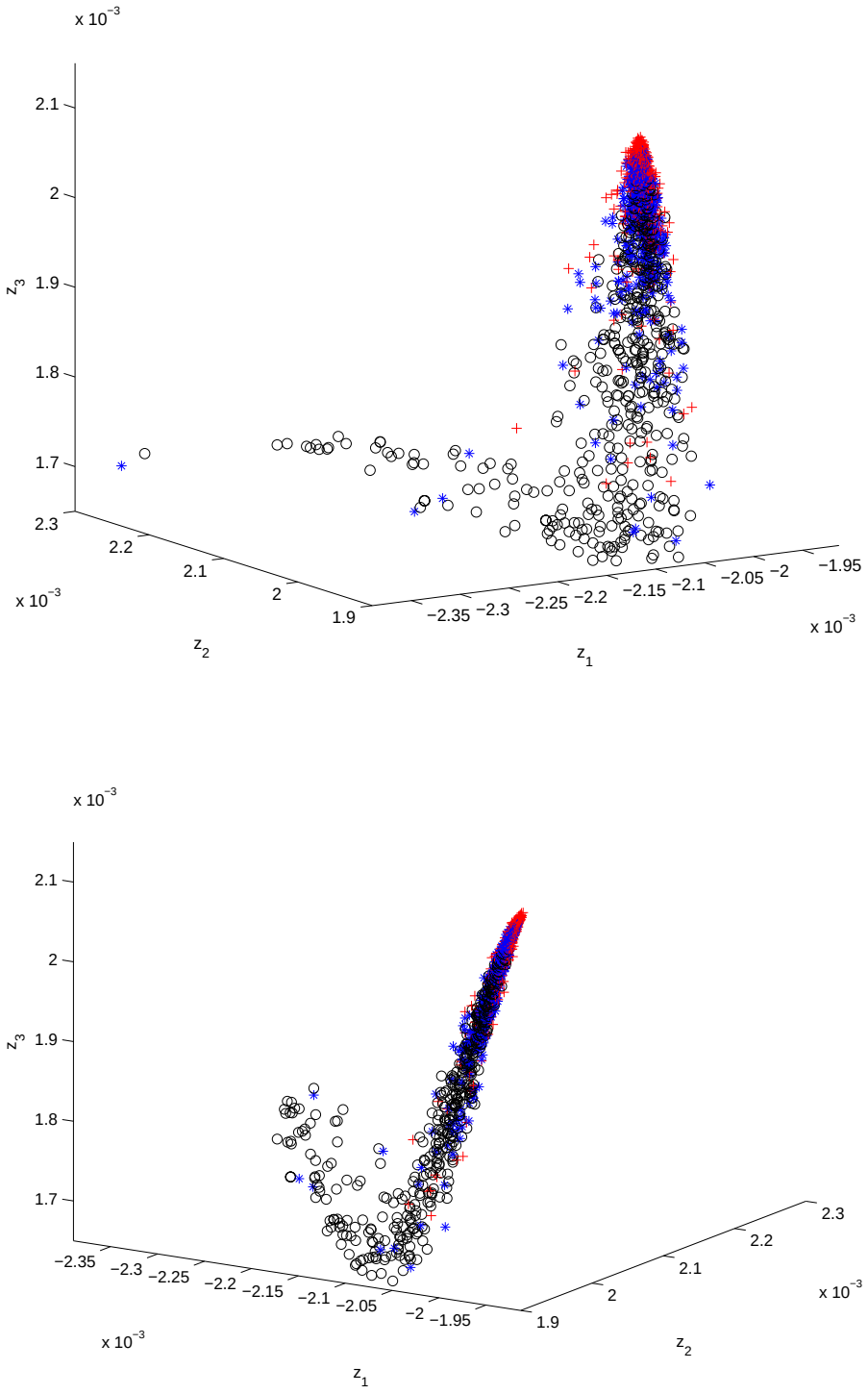

x 10^−3
2.1
2
1.9
1.8
1.7
z3
2.3
2.2
2.1
2
1.9
x 10^−3
z2
−2.35 −2.3 −2.25 −2.2 −2.15 −2.1 −2.05 −2 −1.95
x 10^−3
z1
x 10^−3
2.1
2
1.9
1.8
1.7
z3
−2.35 −2.3 −2.25 −2.2 −2.15 −2.1 −2.05 −2 −1.95
x 10^−3
z1
1.9
2
2.1
2.2
2.3
x 10^−3
z2

Figure 4

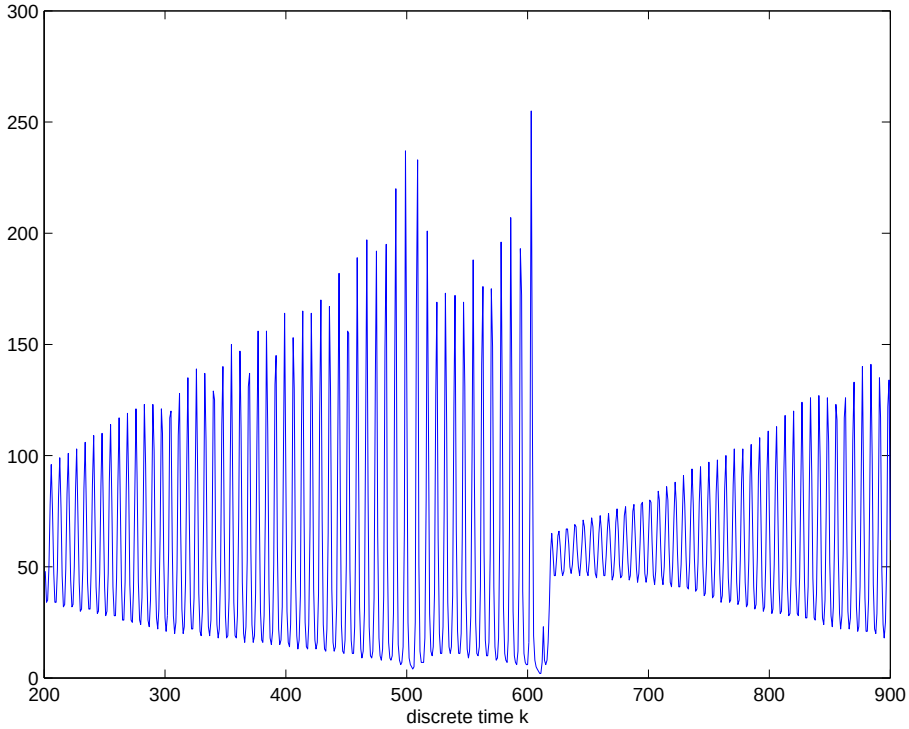


Figure 5

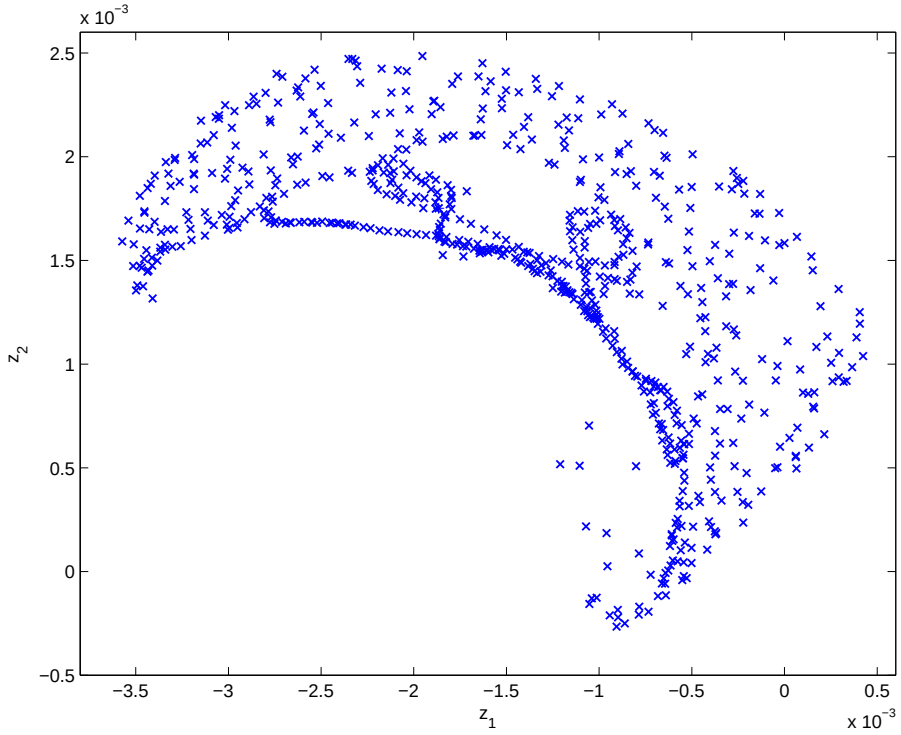
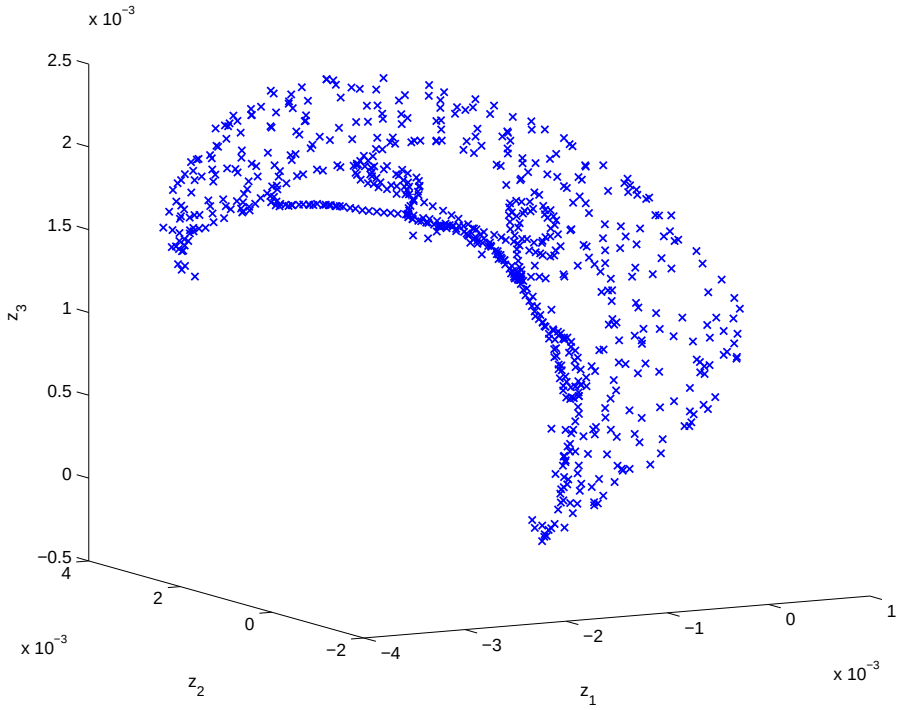


Figure 6


x 10^−3
2.5
2
1.5
1
0.5
0
−0.5
z3
4
2
0
−2
x 10^−3
z2
−4
−3
−2
−1
0
1
x 10^−3
z1

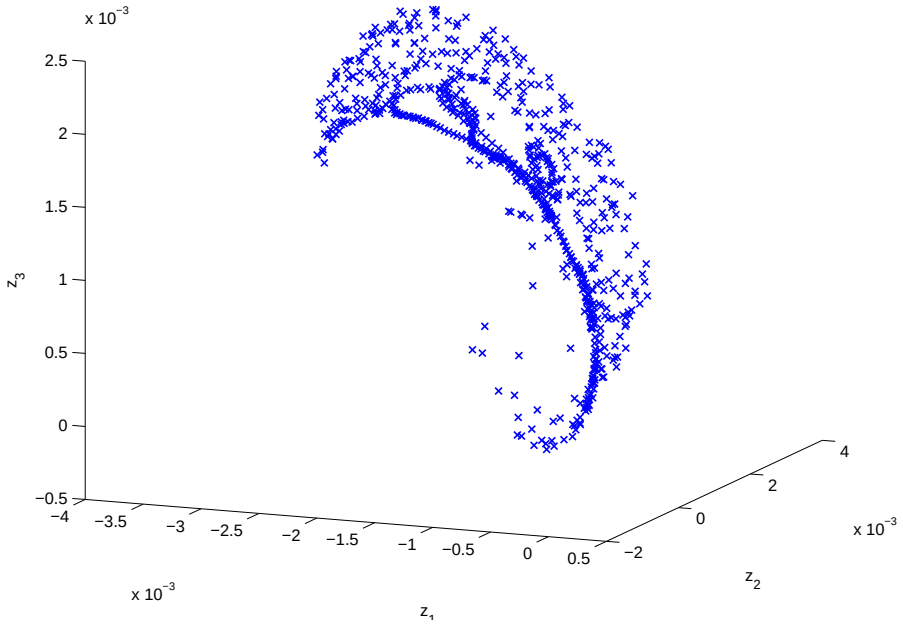

x 10^−3
2.5
2
1.5
1
0.5
0
−0.5
z3
−4
−3.5
−3
−2.5
−2
−1.5
−1
−0.5
0
0.5
x 10^−3
z1
4
2
0
−2
x 10^−3
z2

Figure 7