Learning a kernel matrix for nonlinear dimensionality reduction
or
The semidefinite Teapot

O. Elshenawy
BIRTHDATE1
MAIL1@kth.se

E. Wärnberg
BIRTHDATE2
MAIL2@kth.se

M. Poletti
BIRTHDATE3
MAIL3@kth.se

A. Pettersson
BIRTHDATE4
MAIL4@kth.se

J. Bütepage
900924-T085
butepage@kth.se

Abstract

In this report, we present the replication and analysis of Weinberger et al. [1]. In [1] the authors describe a novel method for dimensionality reduction using Kernel Principal Component Analysis (KPCA). Instead of using a predetermined Kernel function, such as e.g. a Gaussian Kernel, the authors propose to learn a Kernel matrix from the data. The construction of this matrix is based on neighbourhood relations and computed using semidefinite programming.

1 Introduction

In an age of Big Data and high dimensional data sets the topic of dimensionality reduction is essential for efficient data representation and machine learning. The basic idea behind dimensionality reduction methods is that the inherent structure of high dimensional data can be captured by a low dimensional representation without the loss of substantial information.

In mathematical terms, the problem can be stated as follows. Given a number of data points $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ...\mathbf{x}_m\} \in \mathcal{R}^n$, find a d dimensional representation, with $d < n$, which sufficiently captures the information content of the data.

Next to linear approaches such as Principal Component Analysis (PCA) and Multidimensional Scaling (MDS), several methods for learning non-linear submanifolds have been developed. One Kernel based method, Kernel Principal Component Analysis (KPCA), derives the eigenvectors of a Kernel matrix, which maps the data into a, possibly infinite-dimensional, space. Problematic is here, the choice of an appropriate Kernel function, such as the squared exponential, which results in poor performance for submanifold learning. In graph-based methods, the data is represented as a graph, in which data points resemble the nodes and distance relationships between these points constitute the links. As distances on a manifold can reliably be computed only for local neighbourhoods, the graph connects each node to its k nearest neighbours and is not fully connected. After deriving a matrix representation of this graph, the submanifold if given by a number of eigendirections of this matrix.

The approach presented in this report, Maximum variance unfolding (MVU), combines Kernel-based and graph-based methods by learning a positive semidefinite matrix constrained by the relations contained in the graph. According to "Mercers theorem", this matrix can be viewed as the Kernel matrix needed for KPCA. Thus, instead of using a fixed Kernel function, the learning of the Kernel matrix preserves the local structure of the original manifold while unraveling its global complex properties.

In the following, we will shortly present the basic concepts of different methods for dimensionality reduction. Especially, we will focus on MVU and how it can be related to MDS and KPCA. After describing our implementation of MVU, we will describe and present a number of experiments. At last, we will discuss the method and its results.

2 Methods for dimensionality reduction

2.1 Linear Methods

One of the basic approaches is PCA. In order to maximize the information represented by the subspace, it is constructed of d basis vectors that capture the directions of maximal variance in the original data. The solution to this problem are d eigenvectors corresponding to the d largest eigenvalues of the covariance matrix corresponding to $\mathbf{X}$.

Another linear approach is MDS which attempts to find a representation that preserves distance relationships between data points in the original space. As the aim is to minimize the squared difference of dot products of $\mathbf{X}$ in both the original and the low dimensional representation, the solution is given by the spectral decomposition of the Gram matrix $\mathbf{G}$, with $\mathbf{G}_{ij} = \mathbf{x}_i \dot{\mathbf{x}}_j$. This matrix can be derived by a transformation of the distance matrix S, with $\mathrm{S}_{ij} = ||\mathbf{x}_i - \mathbf{x}_j||^2$, as $\mathbf{G}= -\frac{1}{2}\mathbf{HXH}$, where $\mathbf{H} = \mathbf{I} - \frac{1}{n}(1, 1, ..., 1)^T(1, 1, ..., 1)$. In the following, the d dimensional subspace, $d <= n$, is constructed of $\{\sqrt{\lambda_i}\mathbf{e}_i\}_{i=\{1,..d\}}$, where $\mathbf{e}_i$ is the ith eigenvector of $\mathbf{G}$ corresponding to the ith eigenvalue λ_i and d is the number of large eigenvalues.

2.2 Kernel PCA

2.3 Maximum Variance Unfolding

3 Our approach

We re-represented the constraints for the Kernel matrix in matrices that can be used with the SeDuMi toolbox.

4 Experimental results

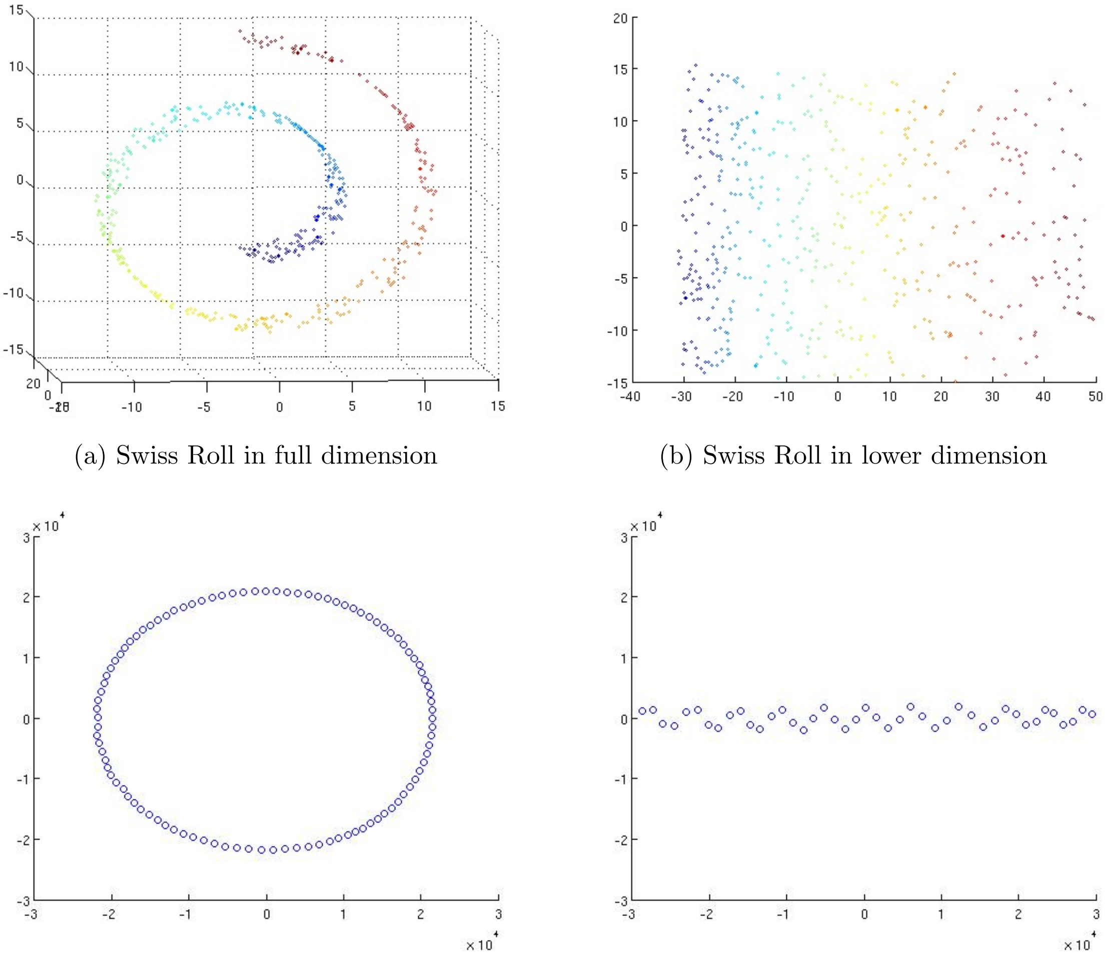
4.1 Setup

4.2 Results

5 Summary and Conclusions

References

[1] K.Q. Weinberger, F. Sha, and L. K. Saul. Learning a kernel matrix for nonlinear dimensionality reduction. In *Proceedings of the Twenty First International Conference on Machine Learning (ICML-04)*, pages 839–846, 2004.



(a) Swiss Roll in full dimension

(b) Swiss Roll in lower dimension

(c) All teapots (0-360°)

(d) Half of the teapots (0-180°)