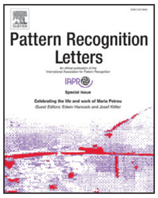
Pattern Recognition Letters 74 (2016) 68–73

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec



Extreme learning machine for out-of-sample extension in Laplacian eigenmaps[☆]

Arturo Mendoza Quispe, Caroline Petitjean[*], Laurent Heutte

LITIS EA 4108, Université de Rouen, 76800 Saint-Etienne-du-Rouvray, France

ARTICLE INFO

Article history:
Received 9 July 2015
Available online 9 February 2016

Keywords:
Dimensionality reduction
Manifold learning
Laplacian eigenmaps
Out-of-sample extension
Extreme learning machine

ABSTRACT

Manifold learning techniques have shown a great potential for computer vision problems; however, they do not extend easily to points different from the ones on which they were trained (out-of-sample). On the other hand, extreme learning machine (ELM) is a powerful method that allows to perform nonlinear, multivariate regression. This paper discusses the effectiveness of ELM for the out-of-sample problem and compares it to the state-of-the-art solution : the Nyström extension. Both methods are evaluated through the reconstruction of the manifold learnt using Laplacian eigenmaps, via experiments on a wide range of publicly available image datasets. We show that when reducing the data dimension to its intrinsic dimension, the ELM offers a better approximation of the embedded coordinates, also with reduced computational costs during testing.

1. Introduction

In computer vision and pattern recognition, many problems are high dimensional, meaning that observations may be described by a large number of measures or variables. In order to find a discriminative description of the data and to get rid of the curse of dimensionality, one may search to reduce its dimension. Dimensionality reduction, also known as Manifold Learning, is also useful for data denoising and visualization. Spectral methods are a family of techniques based on the spectral decomposition (i.e. into eigenvectors and eigenvalues) of the affinity matrix between the N points (or observations) of the dataset. Linear methods, such as principal component analysis (PCA) or multidimensional scaling, consider that data live in a linear subspace and implicitly assume a multivariate gaussian distribution. In practice this may not always be the case, and these assumptions are ignored by nonlinear approaches, such as spectral methods. Spectral methods are a family of non linear dimension reduction techniques, which are based on a feature matrix, whose spectral decomposition yields the reduced dimensionality dataset. These methods can be viewed as kernel PCA described on specially constructed Gram matrices [11]. In image analysis, these nonlinear methods have shown their potential in, among others, facial recognition [41], hyperspectral image classification [24], gait recognition [7], hand-written character recognition [32], and several medical imaging tasks such as segmentation or registration [1,26], data clustering [23]. Example of spectral methods include Isomap, Laplacien eigenmaps, Local Linear Embedding (LLE), diffusion maps, Local Tangent Space Alignment (LTSA), Maximum Variance Unfolding (MVU) [21,28,36]. The most popular for image applications are Isomap and Laplacian eigenmaps and in the following we will restrict our experiments to Laplacian eigenmaps.

One of the drawbacks of nonlinear spectral method is that they do not allow to embed points which are not part of the initial set, contrary to linear methods such as PCA. Out-of-sample projection is necessary when the affinity matrix is built offline for example, but also when the number of observations is very high, an increasingly common situation with today's large amount of data and streaming requirements. In this case, data may be split into two subsets, one whose reduced coordinates are computed with the dimensionality reduction technique (this part plays the role of a training set), and a bigger one, whose reduced coordinates are estimated using an out-of-sample projection. Computing the embedding on the smaller subset allows to reduce computation costs linked to the spectral decomposition [44].

Various solutions have been proposed for out-of-sample [2,5,12,20,31,35,45,46]. Bengio et al's influential paper established much of the framework for this area [4]. This work makes use of the Nyström extension, a method used to speed up kernel methods computations. It is thus based on the assumption that the similarity measure used to compute the embedding may be expressed as a kernel function. The out of sample projection is computed as a linear combination of the eigenvectors of the

[☆] This paper has been recommended for acceptance by J. Yang.

[*] Corresponding author. Tel.: +33 23 295 5215; fax: +33 23 295 5022.
E-mail address: caroline.petitjean@univ-rouen.fr (C. Petitjean).

http://dx.doi.org/10.1016/j.patrec.2016.01.024
0167-8655/© 2016 Elsevier B.V. All rights reserved.

feature matrix, weighted with a kernel function expressing the similarities between the out-of-sample point and the points in the training set. The problem is the tuning of the parameters, usually done in a heuristic way. In [31], authors propose to use sparse grid functions to approximate the eigenfunctions corresponding to the Laplacian eigenmap embedding. The proposed framework is independent from the number of training data points but is dedicated to the Laplacian eigenmap embedding. In [33], the data are represented with sparse coding. While no hyperparameter needs to be tuned, computation time is very high due to L1 minimization.

Nonparametric out-of-sample methods, such as Nyström's, require access to the data, which can be costly for large datasets. Parametric solutions have been developed, that derive an explicit mapping function between the high-dimensional space and the low-dimensional space: some approaches for example integrate several local feature extractor into a global representation [37,39], others propose a nonlinear dimension reduction algorithm is proposed, that learns a parametric mapping to recover a global low dimensional space [40]. Machine learning approaches have also been investigated, such as in [10], where a three-layered perceptron network is trained on the embedding. Whereas it is a generic approach that can suit any manifold learning technique, it suffers from some limitations (e.g. iterative tuning of the parameters, long convergence times) and is not shown to outperform linear fitting error, in terms of reconstruction accuracy.

Some neural network methods are more computationally efficient, such as the family of randomness-based learning networks, among which QuickNet [42,43], Random Vector Feature Link (RVFL) [6,19,30], Random Neural Networks (RNN) [34] and Extreme Learning Machine (ELM) [13,18]. All these methods differ by the way parameters are optimized, among others [14]. In particular, ELM is a learning method to train single-hidden layer feedforward neural networks (SLFN) that does not require iterative tuning, which results in a high learning speed. The ELM method has universal approximation capability of approximating any target continuous function, and of classifying [15,17]. In this respect, ELM can be applied to any approximation problem, and in particular to out-of-sample approximation. However (to the best of our knowledge) ELM has never been investigated for this task. Our aim is thus to show that ELM can be used in practice for out-of-sample approximation, under its multivariate regression form, and what its usage implies in terms of accuracy and computation time. This is why we explore in this paper the capabilities of ELM as an out-of-sample method and compare it to the classical Nyström extension, on embedded spaces built with the Laplacian eigenmaps.

When assessing an out-of-sample method, the protocol is usually as follows: the N points of the datasets are split into a training set of $N - m$ points and the out-of-sample group of m points. An embedding is computed with the whole dataset (in our case with Laplacian eigenmaps), and will serve as reference. Another embedding is computed with only the $N - m$ points of the training set. The remaining m points are then projected using an out-of-sample method (either Nyström or ELM), and their estimated coordinates are compared to those obtained with the reference embedding. Our comparison of Nyström vs ELM measures the projection accuracy for the out-of-sample points onto the manifold and the influence of the number of samples in the training set, the number of dimensions to be reduced, and computation time, both for testing and training.

The remainder of the paper is as follows. The Nyström extension and ELM theoretical background adapted to out-of-sample projection are presented in Sections 2 and 3, respectively. The experimental protocol is described in Section 4, followed by results and discussions which are reported in Section 5. We conclude in Section 6.

2. Nyström extension

In the following, let D denote the dimension of the initial set, N the number of points, $\mathbf{x}_i$ a point included in $\mathbb{R}^D$ and $\mathbf{X}$ the $D \times N$ training matrix containing the points. Let $\mathbf{y}_i$ denote the coordinates in the embedded space, included in $\mathbb{R}^d$ where d is the reduced dimension that corresponds to $\mathbf{x}_i$. At last let $\mathbf{x}_{N+1}$ denote a point not belonging to the initial set of points, i.e. an out-of-sample point; the goal is to estimate its reduced coordinates $\mathbf{y}_{N+1}$.

The Nyström method is a method to speed up kernel methods computations, by performing the eigendecomposition on a subset of examples. It was used in [4] to propose an out-of-sample extension to kernel-based spectral methods. Let us recall the general framework in which spectral dimension reduction techniques can be cast. Let $\mathbf{W}$ be a symmetric matrix of size $N \times N$, expressing the affinity between the N points of the training set. Let $K(\cdot, \cdot)$ denote a data-dependant kernel function giving rise to matrix $\mathbf{W}$ with $W_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$.

Let $(\mathbf{v}_k, \lambda_k)$ denote the eigenvector and eigenvalue pairs such that $\mathbf{W}\mathbf{v}_k = \lambda_k \mathbf{v}_k$. For dimensionality reduction, retain the d largest (or smallest, depending on the method) eigenvalues and their associated eigenvectors. The embedding (or reduced coordinates) of each training sample $\mathbf{x}_i$ is the ith line of a matrix $\mathbf{U}$ that contains the d eigenvectors in columns.

The Nyström extension for an out-of-sample point is only a weighted sum of the previously calculated eigenvectors and eigenvalues. More precisely the kth reduced coordinate of the out-of-sample point is approximated as:

$$y_{N+1,k} = \frac{1}{\lambda_k} \sum_{i=1}^{N} v_{ki} K(\mathbf{x}_{N+1}, \mathbf{x}_i) \text{ for all } k = 1, .., d \quad (1)$$

Or, in matrix form,

$$\hat{\mathbf{y}}_{N+1} = \frac{1}{\sqrt{\Lambda}} \mathbf{U}^T \mathbf{K}_{N+1} \quad (2)$$

where $\frac{1}{\sqrt{\Lambda}} = \mathrm{diag}(\frac{1}{\sqrt{\lambda_1}}, \cdots, \frac{1}{\sqrt{\lambda_d}})$, $\mathbf{U}$ is the matrix whose columns are the eigenvectors, and $\mathbf{K}_{N+1} = [K(\mathbf{x}_{N+1}, \mathbf{x}_1) \cdots K(\mathbf{x}_{N+1}, \mathbf{x}_N)]$. In [4], Bengio et al. have designed a formulation of $K(\cdot, \cdot)$ for MDS, Laplacian eigenmaps [3], Isomap and LLE. The Nyström extension is applicable to any technique that kernel function. This method requires some parameter choice for kernel $K(\cdot, \cdot)$, usually made heuristically.

3. Extreme learning machine for out-of-sample approximation

We first present ELM basics and then how to use it for out-of-sample extension. Contrary to the Nyström extension that relies on the embedding obtained from the training set, the ELM just trains on the data, as any multivariate regression.

The output function of ELM for an input $\mathbf{x}$ can be written as [13]:

$$f_L(\mathbf{x}) = \sum_{i=1}^{L} \beta_i h_i(\mathbf{x}) = \mathbf{h}(\mathbf{x})\beta \quad (3)$$

where $\beta = [\beta_1 ... \beta_L]$ is the output weight vector between the L-neuron layer and the output nodes, and $\mathbf{h}(\mathbf{x}) = [h_1(\mathbf{x}) ... h_L(\mathbf{x})]$ is a nonlinear feature mapping, with each function h_i defined as:

$$h_i(\mathbf{x}) = G(\mathbf{a}_i, b_i, \mathbf{x}) \quad (4)$$

where G is for example a sigmoid function defined as $G(\mathbf{a}_i, b_i, \mathbf{x}) = \frac{1}{1+\exp(-\mathbf{a}_i \cdot \mathbf{x} + b_i)}$; and the vector $\mathbf{a}_i$ of length L and the bias term b_i are the randomly generated parameters of the hidden node i. Training the ELM includes two stages, the random feature mapping and linear parameter solving. The parameters ($\mathbf{a}_i$ and b_i) of