Available online at www.sciencedirect.com

ScienceDirect

AASRI Procedia 8 (2014) 68 – 74

AASRI Procedia

www.elsevier.com/locate/procedia

2014 AASRI Conference on Sports Engineering and Computer Science (SECS 2014)

An Efficient Use of Principal Component Analysis in Workload Characterization-A Study

Jyotirmoy Sarkar[a], Snehanshu Saha[b], Surbhi Agrawal[b]*

[a]*BITS PILANI & TechMahindra,,Bangalore,560100,India*
[b]*CBIMMC & Dept. of Computer Science and Engineering,PESIT-BSC,Bangalore,560100,India*

Abstract

PCA is a useful statistical technique that has found application in fields such as face recognition, image compression, dimensionality reduction, Computer System performance analysis etc. It is a common technique for finding patterns in data of high dimension. In this paper, we present the basic idea of principal component analysis as a general approach that extends to various popular data analysis techniques. We state the mathematical theory behind PCA and focus on monitoring system performance using the PCA algorithm. Next, an Eigen value-Eigenvector dynamics is elaborated which aims to reduce the computational cost of the experiment. The Mathematical theory is explored and validated. For the purpose of illustration we present the algorithmic implementation details and numerical examples over real time and synthetic datasets.

Keywords: PCA; Eigen Value; Eigen Vector,Workload Characterization..

* Corresponding author. Tel.: +91-080-66186622; fax: 91-80-.
E-mail address: snehanshusaha@pes.edu.

doi:10.1016/j.aasri.2014.08.012

1. Introduction

Performance evaluation helps us to give an idea how well a system is performing as compared to other systems. Workload is the most crucial part of any performance evaluation process. The entire process can end up in a wrong conclusion if workload is not chosen in appropriate ways. Therefore, workload selection is an integral part of performance evaluation project. Computer architectures are evaluated by running a workload on the computer and measuring the execution time. New computers are designed the same way. As the newly designed computer does not exist, it is not possible to run any workload. This is where workload characterization comes into the fore. The goal of workload characterization is to describe the properties of a workload in terms of abstract performance metrics, called workload characteristics, which predict the final performance [1].

There are a couple of techniques to classify workload components. One of the widely used techniques is "weighted sum of the parameter values" that uses sum to classify the workload components into classes. But there are proper guidelines to decide the weight of parameters. Before PCA, an analyst running software used to assume the values of weight. Instead, one can use PCA, to calculate the value of weights. PCA is a procedure by which numbers of correlated variables are transformed into a smaller number of uncorrelated variables. It is a data analysis technique traced back to Pearson (1901). It can be used to compress a high dimensional dataset into a lower dimensional dataset. PCA can be derived from a number of starting points and optimization criteria. The most important of these are minimization of the mean square error in data compression, finding mutually orthogonal directions in the data having maximal variances and de-correlation of the data using orthogonal transformation. These uncorrelated variables are called Principal Components.

1.1. How PCA works:

For a given set of **n** parameters $\{x_1, x_2, ...x_n\}$, the Principal Component Analysis will produce a set of principal factors. The following conditions will hold true for the newly produced set -

- The principal factor (y_i) is a linear combination of initial parameters (x_j) .

$$y = \sum_{j=1}^{n} a_{ij} x_j$$

- Principal factor set is an orthogonal set.

$$< y_i, y_j >= \sum_k a_{ik} a_{kj} = 0$$

It is an ordered set $\{y_1, y_2, ...y_n\}$ in the decreasing order of the percentage of variance, with y_1 being the highest percentage of variance and y_n the least. So first few factors can be used to classify the workload components.

We can find the application of Principal Component Analysis in many fields including data compression, image processing, visualization, pattern recognition and time series prediction [2].Sirvich and Kirby had efficiently used PCA in human faces representation [3-4].This approach leads to decomposition of any images into Eigen pictures so that the image can be reconstructed using a portion of the Eigen pictures and the corresponding projection onto the Eigen picture subspace [5]. The PCA method has also been used in handprint recognition, human made object recognition, industrial robotics and mobile robotics etc [6]. In a workload composition the choice of benchmark is very important. The selection of benchmark for inclusion in benchmark suit is called workload composition. Smith [7] used a metric, which is based on dynamic program characteristics for the Fortran language..They used squared Euclidean distance to measure the difference between benchmarks. The shortcoming of this procedure is the use of Euclidean distance for measuring the difference. To overcome this Eeckhout et al. [8] proposed Principal Component Analysis (PCA) to get rid of the correlation and dependence between variables. A number of program characteristics are measured for a number of benchmarks on which PCA was applied.

2. Proposed Work

The most computationally expensive part of PCA is the calculation of Eigen values and Eigen vectors of the dataset. In this paper our main objective is to save the computational time of Principal Component Analysis (PCA) by skipping the Eigen vector calculation. Here we propose to bypass Eigen vector calculation, by rather inspecting the Eigen vectors. The understanding of the dynamics of Eigen values and Eigen vectors in the context of Linear transformations and vector spaces plays a crucial role in improving the efficiency of PCA in the workload characterization problem. This will require us to prove/cite important theorems in Linear Algebra.

2.1. Algorithm

- Compute the mean and standard deviation of the parameters. $\overline{x} = \frac{1}{n}\sum_{i=1}^{n} a_i$, $s_x^2 = \frac{1}{n-1}\sum_{i=1}^{n} (x_i - \overline{x})$

- Compute the correlation of the parameters. $R_{x_a x_b} = \dfrac{\frac{1}{n}\sum_{i=1}^{n} (x_{a_i} - \overline{x_a})(x_{b_i} - \overline{x_b})}{s_{x_a} s_{x_b}}$

- Compute QR Decomposition of correlation matrix at every step $A_k = Q_k R_k$ (starting with $k = 0$), where Q_k is an orthogonal matrix and R_k is an upper triangle matrix. $A_{k+1} = R_k Q_K$
- The matrix will converge to a triangular matrix, known as the Schur form. Find out the eigen values of the matrix from the diagonal.
- Apply the results of the theorem to choose the eigen vectors by inspection. This is possible since the matrices obtained are symmetric and the Eigen values are real and distinct.
- Use the Eigen vectors to compute the principal factors.

Next, we prove a theorem related to eigen values and eigen vectors. Let us take a linear map $T : u \rightarrow v \ni T(\alpha x + \beta y) = \alpha T(x) + \beta T(y)$; where $x, y \in u$; u, v are vector spaces of certain dimensions, $n \,\&\, m$ say where $n \neq m$ necessarily; e.g. $u = R^n, v = R^m$ $\exists \lambda \in R \ni Tx = \lambda x$, then λ is an Eigen value of $T \,\&\, x$ is a corresponding Eigen vector.

2.2. Proposition 1:

For the linear map $T : u \rightarrow v$, if the Eigen values “ λ “are distinct, then T admits of linearly independent Eigen vectors.

Proof: Linear Independence: A set of vectors $\{v_1, v_2, v_3, \dots v_n\}$ is linearly independent if $\exists$ scalars $(\alpha_1, \alpha_2, \dots \alpha_n) \ni \sum_{i=1}^{n} \alpha_i v_i = 0$ implies $\alpha_i \equiv 0 \forall i = 1,..n$ i.e. v_i; any vector in the set is NOT a linear combination of any of the other vectors in the same set e.g. $\begin{bmatrix}1\\1\end{bmatrix}$ & $\begin{bmatrix}1\\-1\end{bmatrix}$ are linearly independent.

Proof of the theorem: (Using the Principle of Mathematical Induction)

Basis Step: $n = 2;$ NTS $a_1v_1 + a_2v_2 = 0 \Rightarrow a_1 = 0 = a_2$

Apply $T \Rightarrow T(a_1v_1 + a_2v_2) = T(0)$; T linear map

$$\Rightarrow a_1T(v_1) + a_2T(v_2) = 0;$$

$$\Rightarrow a_1\lambda_1v_1 + a_2\lambda_2v_2 = 0 \qquad (1)$$

Also

$$a_1\lambda_1v_1 + a_2\lambda_1v_2 = 0 \qquad (2)$$

Therefore *(1)* − *(2)* $\Rightarrow a_2(\lambda_2 - \lambda_1)v_2 = 0 \Rightarrow a_2 = 0 \;\; (\because \lambda_1 \neq \lambda_2; v_2 \neq 0)$

$a_2 = 0 \Rightarrow a_1 = 0$; Induction hypothesis: Assume the proposition is true for $n = m$; Induction steps: on $n = m + 1$ i.e. NTS $a_1v_1 + ... + a_mv_m + a_{m+1}v_{m+1} = 0$ => $a_1 = a_2 = ... = a_m = a_{m+1} = 0.$

Let $a_1v_1 + ... + a_mv_m + a_{m+1}v_{m+1} = 0$ i.e. $T(a_1v_1 + ... + a_mv_m + a_{m+1}v_{m+1}) = T(0)$

so,

$$a_1\lambda_1v_1 + ... + a_m\lambda_mv_m + a_{m+1}\lambda_{m+1}v_{m+1} = 0 \qquad (3)$$

Also,

$$a_1\lambda_1v_1 + ... + a_m\lambda_1v_m + a_{m+1}\lambda_1v_{m+1} = 0 \qquad (4)$$

(3) − *(4)* $\Rightarrow \; a_2(\lambda_2 - \lambda_1)v_2 + ... + a_{m+1}(\lambda_{m+1} - \lambda_1)v_{m+1} = 0$

By the hypothesis, $\{v_1, ...v_m\}$ linear independent $\Rightarrow a_1 = a_2 = ...a_m = a_{m+1} = 0$

$$\therefore \; a_{m+1}(\lambda_{m+1} - \lambda_1)v_{m+1} = 0 \;\; (\because (\lambda_{m+1} - \lambda_1) \neq 0 \,\&\, v_{m+1} \neq 0)$$

Therefore, the Eigen vectors $\{v_1, ...v_m, v_{m+1}\}$ are Linearly Independent.

Proposition 2: *A real, symmetric linear map T (matrix) admits of orthogonal Eigenvectors.*

Proof: Well established result [9].

Conclusion of proposition: The aforementioned matrix (or the linear map, T) has distinct eigen values (as always the case will be) and is real, symmetric. Therefore, the corresponding eigenvectors will be linearly independent & orthogonal to each other. This enables us to find the eigenvectors by inspection rather than computing step by step via set of simultaneous equations. This saves O (n) computations, crucial computation cost!

2.3. Implication

The paper aims to improve time complexity of PCA algorithm. The following example will illustrate the principle behind PCA from initial parameters.We have collected synthetic data of the number of packets lost on two different network links. x_a is the number of packets lost on link A and x_b is the number of packets lost on link B

Table 1. Data for Principal Component Analysis Example 1

Observation Number	x_a (Variables)	x_b (Variables)	y_a (Principal factors)	y_b (Principal factors)
1	300	400	-0.0027	-0.0014
2	510	330	-0.0014	0.0028
3	212	547	-0.0021	0.0049
4	309	690	0.0028	-0.0070
5	610	410	0.0014	0.0028
6	910	150	0.0007	0.0133
7	540	320	-0.0014	0.0042
8	440	540	0.0007	-0.0021
9	219	440	-0.0037	-0.0023
10	510	779	0.0070	-0.0054

First we have to compute the mean and standard deviation using the formulas given in *Algorithm 2*

$$\overline{x_a} = \frac{4560}{10} = 456 \;;\; \overline{x_b} = \frac{4600}{10} = 460 \;; s_{x_a}^2 = \frac{2483986 - 10 \times 456^2}{9} = 44958.4 \;;$$

$$s_{x_b}^2 = 34805.5$$

Correlation among the variables as $R_{x_a x_b} = -0.486$ and hence the correlation matrix will be

$$C = \begin{bmatrix} 1.000 & -0.486 \\ -0.486 & 1.000 \end{bmatrix}$$

(5)

Now we will compute the Eigen values from the above correlation matrix using characteristic equation

$$|C - \lambda I| = \begin{vmatrix} 1-\lambda & -0.486 \\ -0.486 & 1-\lambda \end{vmatrix} = 0 \Rightarrow (1-\lambda)^2 - 0.486^2 = 0$$

The Eigen values are 1.486 and 0.514.

3. Results and Discussion

Now, the correlation matrix in (5) is both real and symmetric and is a candidate for the theorems to be applied. Figure1 below shows the average execution time of PCA algorithm over a sample of 5 different datasets. The execution time has been recorded in milliseconds.

4. Conclusion

PCA is the simplest of the true Eigenvector-based multivariate analyses. It can be used to reveal internal structure of data in a way that best explains the variance in data. PCA is sensitive to outliers in the data that produce large number of errors. So, before applying PCA it is expected to remove outliers. As a limitation the result of PCA depend on the scaling of variables. The applicability of PCA constrained by certain assumption made in derivation. Our work explores the underlying principles of PCA and exploits the inherent mathematical theory for efficient computation. The figure below conclusively shows that computation time has been reduced to achieve the same results.

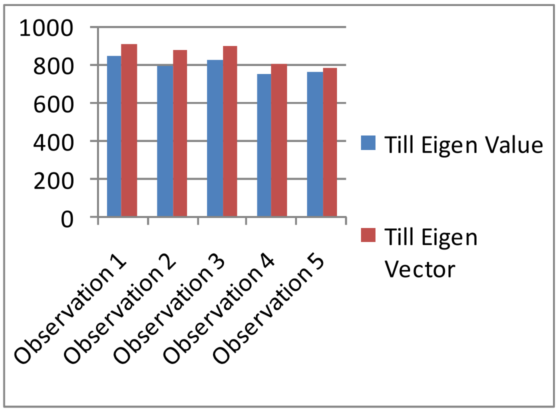

Fig.1. execution times of Eigen values and Eigen vectors and the time saved.

References

[1] T. M. Conte and W. Hwu, Benchmark Characterization,"IEEE Computer, vol. 24, no. 1, pp. 48-56, Jan. 1991.
[2] Raj Jain, The Art of Computer Systems Performance Analysis, Techniques for Experimental Design, Measurement, Simulation, and Modeling.
[3] Kirby and Sirovich, 1990. Application of Karhunen-Loeve Procedure for the Characterization of Human Faces. IEEE
[4] Taranpreet Singh, Face Recognition Based on PCA Algorithm.
[5] S Ekhe, Y Chincholkar, Improved Face Recognition using PCA & LDA
[6] R Gottumukkal, V K Asari, An Improved Face Recognition Technique Based on Modular PCA Approach.

[7] R. H. Saavedra and A. J. Smith, “Analysis of Benchmark Characteristics and Benchmark Performance Prediction,” ACM TOCS, vol. 14, no. 4, pp. 344–384, Nov. 1996.
[8] L. Eeckhout, H. Vandierendonck, and K. De Bosschere, “Quantifying the Impact of Input Data Sets on Program Behavior and its Applications,” JILP, vol. 5, Feb. 2003, http://www.jilp.org/vol5
[9] Gilbert Strang, “Introduction to Linear Algebra”, 4th Edition, SIAM, 2009.