Approximated PCA

Rodrigo Arias Mallo

February 28, 2017

1 Introduction

The PCA method transforms a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components using an orthogonal transformation (rotations and reflections).

The transformation, projects the data in a new subspace, in which each new variable it's now uncorrelated. That means that the covariance of each pair of new variables is zero. To compute the transformation, different approaches can be taken. In a first attempt, the covariance matrix will be used. Let x_{ij} be the observation j of the variable i. Let n be the number of variables and m the number of observations. Each element s_{ij} in the covariance matrix S is computed by

$$s_{ij} = \frac{\sum x_{ik}x_{jk} - \sum x_{ij}x_{jk}}{n(n-1)}$$

Once the covariance matrix S in computed, it can be used to find his eigenvalues and eigenvectors. One method to compute those values, is a combination of a Householder transformation, followed by the QR transformation. The first will transform S in a product of two matrix Q and R.

$$S = QR$$

Such that R contains 3 diagonals (tridiagonal) with elements and zeros in the rest. The QR transformation then takes this two matrices, and computes iteratively a new diagonal matrix $A^{(i+1)} = R^{(i)}Q^{(i)}$. Finally, the eigenvalues are in the diagonal of A and the eigenvectors are computed from these.

2 Householder tridiagonalization

The Householder tridiagonalization it's a process where a matrix A is transformed by multiplying with an orthogonal matrix $P^{(k)}$: $P^{(k)} = I - 2ww^T$ Such matrix $P^{(k)}$ has been prepared, so that $P^{(k)}A$ is a new matrix, with zeros below the $k+1$ element in the k column. This new matrix, has the same eigenvalues as the provious A. The step is repeated until the final matrix has only elements in the diagonal, and the two sub-diagonals. The process is similar to a Gaussian elimination.

3 Eigenvalue sensitivity

Corolary 8.1.6: If A and $A+E$ are n-by-n symmetric matrices, then

$$|\lambda_k(A+E) - \lambda_k(A)| \leq \|E\|_2$$

for $k = 1 : n$.

Then, the difference between the eigenvalue of a noisy matrix, and the original, can be bounded by the 2-norm of E, also the maximum eigenvalue of E.

4 First approach

The first experiment to be carried out consists in determining which parts of the algorithm can be suitable for approximate computing. The main steps of PCA can be summarized as follow:

1. Take a dataset X of n variables.
2. Scale and center the variables.
3. From X compute the $n \times n$ covariance matrix S.
4. **Compute the eigenvalues and eigenvectors of S.**
5. *Optional: Ignore some eigenvectors.*
6. Generate a new basis from the selected eigenvectors.
7. Project X into the new basis.

Computing the eigenvectors is the principal step of PCA.

4.1 Computing eigenvalues and eigenvectors

1. Take the $n \times n$ target matrix $A = S$.
2. Compute a tridiagonal matrix T: $A = PTP^T$.
3. From T compute a diagonal matrix D: $T = QDQ^T$.
4. The eigenvalues of A are in the diagonal of D.
5. Compute the eigenvectors from D.

Computing a tridiagonal matrix is called **tridiagonalization**. For the diagonal, **diagonalization**. Several algorithms exists for both steps.

4.2 Tridiagonalization algorithms

These algorithms transform a **symmetric** matrix A into a new pair of matrices P and T such that P is orthogonal, T is tridiagonal, and $A = PTP^T$

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix} = P \begin{pmatrix} t_{11} & t_{12} & & \\ t_{21} & t_{22} & t_{23} & \\ & t_{32} & t_{33} & t_{34} \\ & & t_{43} & t_{44} \end{pmatrix} P^T$$

Some algorithms can be used for this factorization:

Algorithm	Complexity	Iterative	Stability
Householder	$O(4n^3/3)$	No	Great
Givens	$O(kn^3)$	No	Good
Lanczos	$O(kpn^2)$	Yes	Bad
Others			

Where $n \times n$ is the dimension of the matrix A, k is some constant, and p the number of iterations.

4.3 Diagonalization algorithms

These algorithms take a **tridiagonal** matrix T into a new pair of matrices Q and D such that Q is orthogonal, D is diagonal, and $T = QDQ^T$

$$\begin{pmatrix} t_{11} & t_{12} & & \\ t_{21} & t_{22} & t_{23} & \\ & t_{32} & t_{33} & t_{34} \\ & & t_{43} & t_{44} \end{pmatrix} = Q \begin{pmatrix} d_{11} & & & \\ & d_{22} & & \\ & & d_{33} & \\ & & & d_{44} \end{pmatrix} Q^T$$

The matrix D contains the **eigenvalues** in the diagonal. Some algorithms can be used to compute the diagonalization:

Algorithm	Complexity	Convergence
QR	$O(6n^3)$	Cubic
Divide and conquer	$O(8n^3/3)$	Quadratic
Jacobi	$O(n^3)$	Quadratic
Power iteration	$O(n^3)$	Linear
Inverse iteration	$O(n^3)$	Linear
Others		

All these algorithms are iterative and $n \times n$ is the dimension of the matrix A.