CSC338 Numerical Methods

Lecture 12

April 2, 2020

Agenda

- Exam
- Principal Component Analysis
- Course Evaluation

Exam Logistics

We're still working on the exam logistics, here's what we currently have in mind:

- Exam paper will be downloadable on the day of the exam (most likely via a link on Markus). There will be different versions of the exam
- We will log the time that you access your exam script
- We will expect you save your exam script locally and cut internet
- You will have 2 hours to write the exam (e.g. on a piece of paper, type your answers . . .)
- You will write a declaration stating the time you start/complete the exam, and that you did not use any unauthorized aid
- You will have 30 minutes to upload your solutions **and your aid sheet**

Exam Office Hours

Starting April 6th:

- MW 3:00pm-4:00pm

I will wait 15 minutes, and if no one is online the office hours will be cancelled.

If you intend to come later in the office hour time, please email me in advance.

Mock Exam

We'll hold a 30 minute mock exam available between April 9th-April 11th.

- Download the exam script (e.g. from the link on Markus)
- We will log the time that you access your exam script (you won't see this)
- The mock test should take ~30 minutes
- Upload your solutions on Markus

CSC321 Social: Evening of Go

Learn Go with Lisa: Friday, April 3rd, 4pm-5pm on Bb Collaborate

https://ca.bbcollab.com/guest/f3dfb9707f304c0f8f016dcffce03eaf

AlphaGo Documentary:
https://www.youtube.com/watch?v=WXuK6gekU1Y

AlphaGo/Reinforcement Learning Discussions: Friday, April 3rd, 8pm-9pm

CSC338 "Exam Jam"

Is there interest in making April 6th office hours an *Exam Jam* session?

Dimensionality Reduction

We want to represent data in a new coordinate system with fewer dimensions, while preserving as much *information* as possible

Why?

- Can't easily visualize high dimensional data, but can easily plot 2D (and 3D data)
- We want to extract features from the data (e.g. to build a linear regression model like in Hw6)
- We want to compress the data while preserving most of the information

Preserving Information

What does it mean to preserve *as much information* as possible?

- Preserve **distance** between data points
- Preserve **variations** in the data

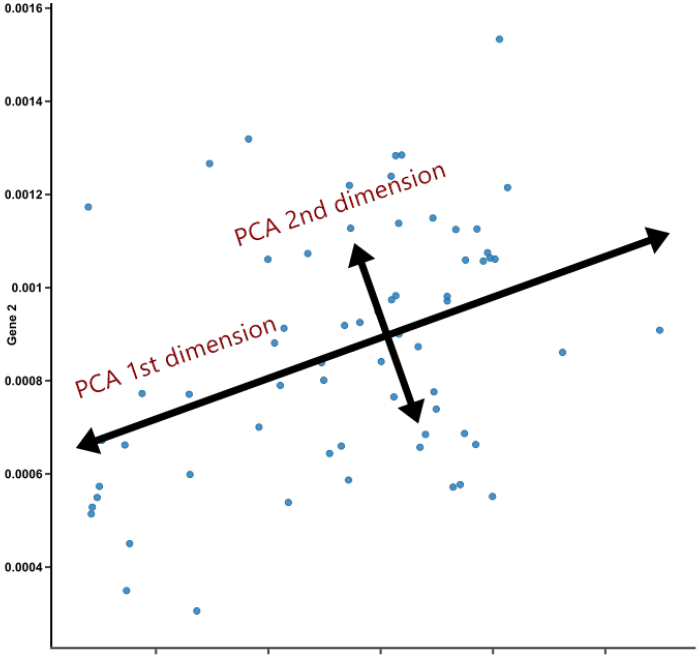
Principal Component Analysis

PCA is a **linear** dimensionality reduction technique

The transformed data is a linear transformation of the original data

We want to find a hyperplane that the data lies in and project the data onto that hyperplane

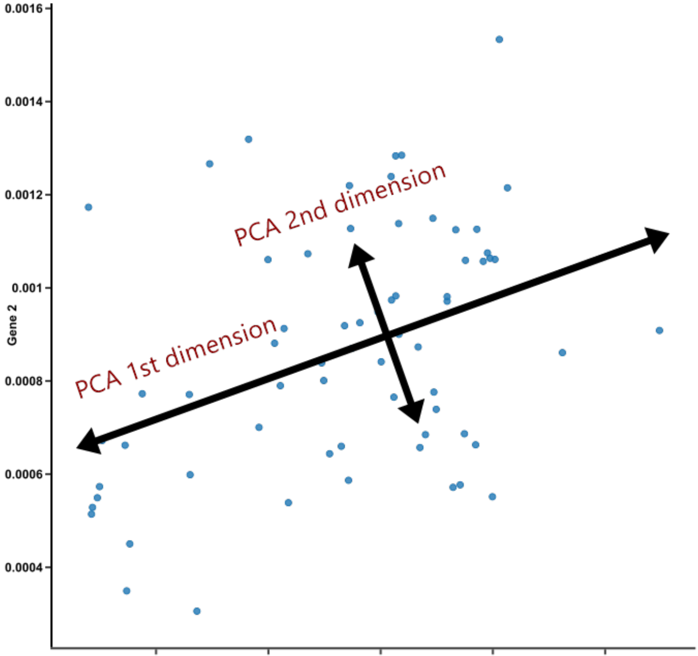
PCA 2D to 1D


0.0016
0.0014
0.0012
0.001
0.0008
0.0006
0.0004
Gene 2
PCA 2nd dimension
PCA 1st dimension

PCA: Key Idea

1. **Rotate** the data with some rotation matrix R so that the new features are uncorrelated
2. **Keep** only the dimensions with the **highest variance** (same as preserving distance)

PCA Rotation Picture



PCA Derivation

We want to identify directions in our data dimension with high variance

Let A be our data, where A is $m \times n$, and A is normalized (so the mean of each column of A is zero)

Look at the covariance matrix $E = A^T A$. The spectral theorem says that we can diagonalize E:

$$E = RDR^T$$

Where D is diagonal and R is orthogonal. The diagonals of D are the eigenvalues. The columns of R are eigenvectors of E.

Do you remember what eigenvectors are?

Why PCA

$$\begin{aligned} E &= RDR^T \\ A^T A &= R^T D R \\ R^T A^T A R &= D \\ (RA)^T (AR) &= D \end{aligned}$$

So if we rotate A using R, the covariance of the transformed data will be diagonal.

The columns of AR is uncorrelated.

Numerical Problem

Given $E = A^T A$, find orthogonal R, diagonal D such that

$$E = RDR^T$$

Finding one eigenvector/eigenvalues

Finding the largest eigenvalue and the corresponding eigenvector is straightforward using **Power Iteration**.

Start with a random vector $\mathbf{x}$ in $\mathbb{R}^n$, and repeatedly compute

$$\mathbf{x} = A\mathbf{x}$$

(Does this remind you of fixed-point iteration?)

Finding one eigenvector/eigenvalues

Note: the norm of $\mathbf{x}$ might grow, so normalize instead

$$\mathbf{x} = A\mathbf{x}$$
$$\mathbf{x} = \frac{\mathbf{x}}{||\mathbf{x}||}$$

Finding many eigenvectors/eigenvalues

In linear algebra class, you might have used the characteristic polynomial

In numerical computing, we use *simnultaneous iteration*: similar idea to *power iteration*, but we try to find multiple eigenvectors/eigenvalues at the same time! Use a matrix X instead of $\mathbf{x}$

We need to make sure that we find different eigenvectors/eigenvalues, so we want the columns of X to be orthogonal!

Compute QR factorization of X in each iteration

Finding many eigenvectors/eigenvalues

In each iteration:

- Compute the QR factorization of X
- Replace X with AQ

We can find *all* the eigenvectors/eigenvalues in the same way.

(We'll skip the discussion on sensitivity and conditioning. Generally, the problem becomes ill-conditioned when you have eigenvalues that are close to each other)

Singular Value Decomposition

Instead of computing the eigenvalue decomposition of $A^T A$, computing the **singular value decomposition** of A is a better conditioned problem.

$A = U\Sigma V^T$

Where U is $m \times m$ and orthogonal, V is $n \times n$ and orthogonal, and Σ is $m \times n$ and diagonal.

The diagonal entries of Σ are called *singular values*

SVD vs Eigendecomposition

The eigenvalues of $A^T A$ are squares of the singular values of A. If we have the SVD of $A = U\Sigma V^T$ then

$$\begin{aligned} A^T A &= (U\Sigma V^T)^T (U\Sigma V^T) \\ &= V\Sigma^T U^T U\Sigma V^T \\ &= V\Sigma^T \Sigma V^T \end{aligned}$$

Which gives us the eigenvalue decomposition $A^T A = RDR^T$

SVD and QR Decomposition

QR Decomposition:

$$A = Q \begin{bmatrix} R \\ O \end{bmatrix}$$

Where Q is $m \times m$ and orthogonal, F is $n \times n$ and upper triangular.

Singular Value Decomposition:

$$A = U \Sigma V^T$$

Where U is $m \times m$ and orthogonal, V is $n \times n$ and orthogonal, and Σ is $m \times n$ and diagonal.

Demo

1. SVD on MNIST digits
2. Visualize Eigenvalues & Eigenvectors
3. Eigenvalues

Eigencaces


eigenface 0
eigenface 1
eigenface 2
eigenface 3
eigenface 4
eigenface 5
eigenface 6
eigenface 7
eigenface 8
eigenface 9
eigenface 10
eigenface 11
eigenface 12
eigenface 13
eigenface 14
eigenface 15

Course Evaluations