Lecture 20 Outline

Monday, November 4, 2019 1:12 PM

Singular Value Decomposition

Solution Methods - Summary

Scalar Differential Equations

SVD (continued)

Monday, November 4, 2019 1:05 PM

Uses of SVD:

① Pseudo-Inverse

All matrices have $\underline{A} = \underline{U}\,\underline{\Sigma}\,\underline{V}^T$

Define the pseudo-inverse as

$$\underline{A}^+ \underline{A} = \underline{I} = \underline{A}\,\underline{A}^+$$

Note: $\underline{A}^{-1}$ might not exist

Let $\underline{A}^+ = \underline{V}\,\underline{\Sigma}^{-1}\underline{U}^T$ with

$$\underline{\Sigma}^{-1} = \begin{bmatrix} \sigma_1^{-1} & & 0 \\ & \sigma_2^{-1} & \\ 0 & & \ddots \end{bmatrix}$$

Then

$$\underline{A}^+ \underline{A} = (\underline{V}\,\underline{\Sigma}^{-1}\underline{U}^T)(\underline{U}\,\underline{\Sigma}\,\underline{V}^T)$$

$$= \underline{V}\,\underline{\Sigma}^{-1}\underline{\Sigma}\,\underline{V}^T = \underline{V}\,\underline{V}^T = \underline{I} \qquad (a)$$

What happens if one or more $\sigma = 0$?

Then the corresponding diagonal elements in both $\underline{\Sigma}$ and $\underline{\Sigma}^{-1}$ are set to zero

Equation (a) no longer holds. Instead,

$$\underline{A}\,\underline{A}^+\underline{A} = \underline{A} \quad + \quad \underline{A}^+\underline{A}\,\underline{A}^+ = \underline{A}^+$$

Example: Image compression with focus on gray-scale

An image is just a matrix with values between 0 and 255, with

$0 = \text{black}, \quad 255 = \text{white}$

Consider a 256×512 pixel image

Storing the full image takes

$256 \times 512 = 131072$ pixels (or data points)

Instead, store only the 5 largest singular values

$$\underline{A} \to \text{Image} \approx \sigma_1 \underline{u}_1 \underline{v}_1^T + \sigma_2 \underline{u}_2 \underline{v}_2^T + \dots + \sigma_5 \underline{u}_5 \underline{v}_5^T$$

Size of compressed image

$5 + 5(256 + 512) = 3845$ data points

Compression ratio: $\dfrac{131072}{3845} \sim \dfrac{34}{1}$

Even more dramatic for larger images

Full image storage $\mathcal{O}(mn)$

Compressed image storage $\mathcal{O}(\nu(m+n))$

Computing SVD

Let $\underline{A} \in \mathbb{R}^{m \times n}$

Naive Method:

① Compute eigendecomposition of $\underline{A}^T \underline{A}$

$$\underline{A}^T \underline{A} = \underline{V} \underline{\Lambda} \underline{V}^T$$

② $\underline{\Sigma} = \underline{\Lambda}^{1/2}$

③ Solve $\underline{U} \underline{\Sigma} = \underline{A} \underline{V}$ for $\underline{U}$

$\Rightarrow$ Condition number of $\underline{A}^T \underline{A}$ is large

$\Rightarrow$ Loss of accuracy, if $\sigma_k \ll \| \underline{A} \|$

Instead, use iterative methods

Two-step process

① Convert $\underline{A}$ to bi-diagonal form

② Convert bi-diagonal to diagonal

$$\begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \end{bmatrix} \xrightarrow{①} \begin{bmatrix} \times & \times & 0 \\ 0 & \times & \times \\ 0 & 0 & \times \\ 0 & 0 & 0 \end{bmatrix} \xrightarrow{②} \begin{bmatrix} \times & 0 & 0 \\ 0 & \times & 0 \\ 0 & 0 & \times \\ 0 & 0 & 0 \end{bmatrix}$$

Golub-Kahan (G-K) Bidiagonalization

Use Householder on both left and right

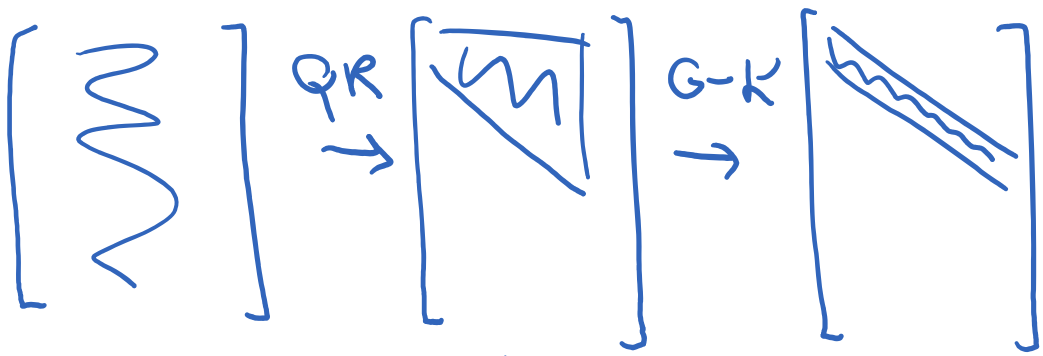
$$\underset{\underline{A}}{\begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \end{bmatrix}} \xrightarrow{\underline{U}_1^T} \underset{\underline{U}_1^T \underline{A}}{\begin{bmatrix} \times & \times & \times \\ 0 & \times & \times \\ 0 & \times & \times \\ 0 & \times & \times \end{bmatrix}} \xrightarrow{\underline{V}_1} \underset{\underline{U}_1^T \underline{A}\, \underline{V}_1}{\begin{bmatrix} \times & \times & 0 \\ 0 & \times & \times \\ 0 & \times & \times \\ 0 & \times & \times \end{bmatrix}}$$

Operation count $\sim 4mn^2 - \frac{4}{3}n^3$

$\sim$ 2x that of QR decomposition, but result is close to final $\underline{R}$

Alternative: Lawson-Hanson-Chan (LHC)

Do QR on $\underline{A}$ first, then G-K on the R matrix



$\underline{A}$ $\quad\quad$ $\underline{Q}^T\underline{A}$ $\quad\quad$ $\underline{U}^T\underline{Q}^T\underline{A}\,\underline{V}$

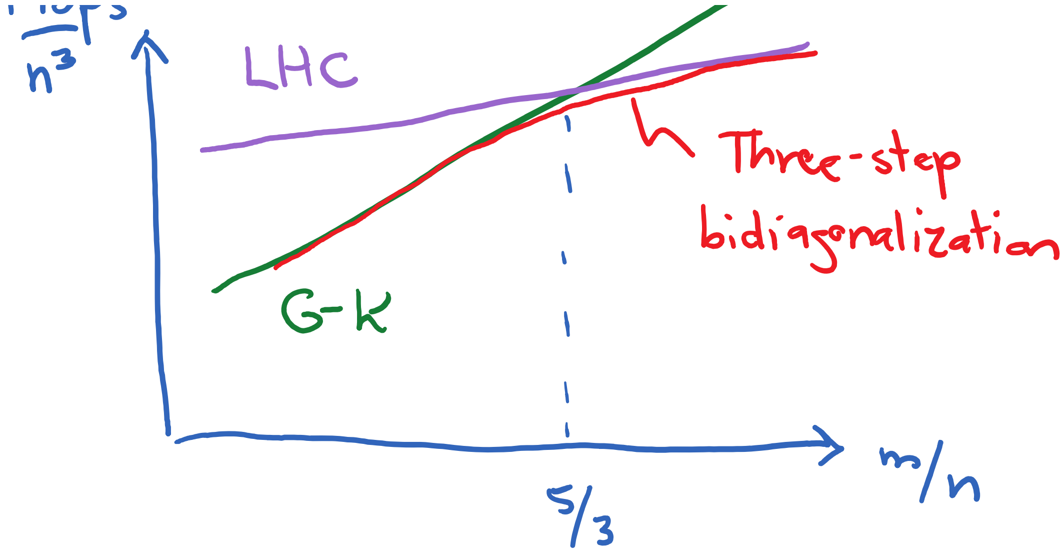
Operation count for LHC

$$QR \sim 2mn^2 - \frac{2}{3}n^3$$

$$G\text{-}K \sim \frac{8}{3}n^3$$

$$\text{Total} \sim 2mn^2 + 2n^3$$

$\therefore$ LHC is cheaper than straight G-K, if $m > \frac{5}{3}n$



See Trefethen & Bau, Lecture 31

for more detail

```

% clear space
clc
clear

% define A matrix
A = [ 1 2 3; 4 5 6; 2 4 6 ]
% perform svd of A
[U,S,V] = svd(A)

% check inverse of S
Sinv = inv(S)
Spinv = pinv(S)

% form pseudoinverse of A
Apinv1 = V*Spinv*U'

% check A times Apseudoinverse
A*Apinv1

% check matrix products
Apinv1 - Apinv1*A*Apinv1
A - A*Apinv1*A

```

Solution Methods - Summary

Monday, November 4, 2019 9:42 AM

$\underline{A}\,\underline{x} = \underline{b}$ might arise from linear systems of equation, regression, etc.

① LU Decomposition $\underline{A} = \underline{L}\,\underline{U}$

$\underline{L}$: lower triangular

$\underline{U}$: upper triangular

To decompose $\underline{A} = \underline{L}\,\underline{U}$ used Gaussian elimination

$\underline{A}\,\underline{x} = \underline{b}$

$\underline{L}\,\underline{U}\,\underline{x} = \underline{b}$ Obtain LU decomposition

$\underline{U}\,\underline{x} = \underline{L}^{-1}\underline{b}$ Forward substitution

$\underline{x} = \underline{U}^{-1}(\underline{L}^{-1}\underline{b})$ Backward substitution

② QR Decomposition $A = QR$

② QR Decomposition $\underline{A} = \underline{Q}\,\underline{R}$

$$\underline{Q}^T \underline{Q} = \underline{I}$$

$\underline{R}$: upper triangular

To decompose $\underline{A} = \underline{Q}\,\underline{R}$

Classical Gram-Schmidt

Modified Gram-Schmidt

Householder

$\downarrow$ More stable, more expensive

$$\underline{A}\,\underline{x} = \underline{b}$$

$\underline{Q}\,\underline{R}\,x = \underline{b}$ Obtain QR decomposition

$\underline{R}\,\underline{x} = \underline{Q}^T \underline{b}$ Orthogonalize

$\underline{x} = \underline{R}^{-1} \underline{Q}^T \underline{b}$ Forward substitution

③ Singular Value Decomposition

$$\underline{A} = \underline{U}\,\underline{\Sigma}\,\underline{V}^T \qquad \underline{U}^T\underline{U} = \underline{U}\,\underline{U}^T = \underline{I}$$

$$\underline{V}^T\underline{V} = \underline{V}\,\underline{V}^T = \underline{I}$$

$\underline{\Sigma}$: diagonal matrix

Methods: Golub-Kahan (-Lanczos)

Lawson-Hanson-Chan

$$\underline{A}\,\underline{x} = \underline{b}$$

$$\underline{U}\,\underline{\Sigma}\,\underline{V}^T\underline{x} = \underline{b} \qquad \text{Obtain SVD}$$

$$\underline{\Sigma}\,\underline{V}^T\underline{x} = \underline{U}^T\underline{b} \qquad \text{Unitary } \underline{U}$$

$$\underline{V}^T\underline{x} = \underline{\Sigma}^{-1}\underline{U}^T\underline{b} \qquad \text{Diagonal } \underline{\Sigma}$$

$$\underline{x} = \underbrace{\underline{V}\,\underline{\Sigma}^{-1}\underline{U}^T}_{\text{Pseudo-inverse of } \underline{A}}\underline{b} \qquad \text{Unitary } \underline{V}$$

Methods

Increasing stability + Increasing cost

LU – Gaussian elim.

QR – Classical G-S

QR – Modified G-S

QR – Householder

SVD – G-K/LHC

Note: Some automatically solve the normal equations

$$\underline{A}^T \underline{A}\, \underline{x} = \underline{A}^T \underline{b}$$

Scalar Differential Equations

Monday, November 4, 2019 3:25 PM

A differential equation is simply one which involves one or more derivatives

Independent variables → "input"

Dependent variables → "output"

Examples:

(1) $\frac{dy}{dt} = t^2$ → $y(t)$ is the unknown function

Dependent variable

Independent variable

(2) $\frac{\partial^2 u}{\partial x^2} = \frac{\partial u}{\partial t}$ → $u(x,t)$ is the unknown function

Dependent variable

Independent variables

Focus on scalar equations: Dependent variable is a scalar function

Let $\mathcal{D}$ represent the differential operator and f denote the non-homogeneous term.

Then a scalar differential equation can be written

$$\mathcal{D}u = f$$

Example (1) above

$$u \to y(t), \quad \mathcal{D} = \frac{d}{dt}, \quad f = t^2$$

Example (2) above

$$u \to u(x,t), \quad \mathcal{D} = \frac{\partial^2}{\partial x^2} - \frac{\partial}{\partial t}, \quad f = 0$$

Classification: Always an important step

Ordinary Differential Equation: If there is only one independent variable

Example (1) above

Partial Differential Equation: If there are two or more independent variables

Example (2) above

Order: Highest order derivative in the ODE or PDE

Example (1) is first order

Example (2) is Second order

Homogeneous Equation: If the only terms which do not include the dependent variable are zero ($f = 0$)

If not homogeneous, then equation is called non-homogeneous ($f \neq 0$)

Linear: Any differential equation in which multiple solutions obey the principle of superposition

If $y_1(t)$ and $y_2(t)$ are both solutions of a homogeneous differential equation, then $c_1 y_1(t) + c_2 y_2(t)$ is also a solution

Assume two solutions to the non-homogenous equations $\mathcal{D} u_1 = f_1$ and $\mathcal{D} u_2 = f_2$

If $\mathcal{D}(c_1 u_1 + c_2 u_2) = c_1 f_1 + c_2 f_2$, then equation is linear.

Notice that for $f_1 = f_2 = 0$, the equation is homogeneous and would obey the first form of the superposition principle.

If the equation does not obey the principle of superposition, then the equation is non-linear

Notation:

For simplicity $\frac{du}{dt} \rightarrow u_t$, $\frac{d^2u}{dt^2} \rightarrow u_{tt}$

13

$$\frac{\partial^3 u}{\partial x \partial t^2} \to u_{xtt}$$

ODEs $\frac{du}{dt} \to u_t = \dot{u}$, $\frac{d^2u}{dt^2} \to u_{tt} = \ddot{u}$

$\frac{du}{dx} \to u_x = u'$, $\frac{d^2u}{dx^2} \to u_{xx} = u''$

Examples:

	Type	Ind Var	Dep Var	Order	Homo	Linear
$y_{xxx} + y\, y_{xx} + x = 0$	ODE	x	y	3	N	N
$x_{yyy} + x\, x_{yy} + x = 0$	ODE	y	x	3	Y	N
$u_{xx} + u_{yy} = u_{tt}$	PDE	x, y, t	u	2	Y	Y

Another check on homogeneity

If one sets the dependent variable to zero, is the equation satisfied? If yes, then equation is homogenous

Differential Operators

Let $\phi(x,y,z)$ be a scalar field and

$$\underline{u}(x,y,z) = \big(u(x,y,z) \quad v(x,y,z) \quad w(x,y,z) \big)$$

be a vector field

① Gradient ∇ : Derivative wrt $x, y + z$ directions

$$\nabla\phi = \begin{bmatrix} \phi_x \\ \phi_y \\ \phi_z \end{bmatrix} \qquad \nabla\underline{u} = \begin{bmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{bmatrix}$$

Gradient increases the dimension of the object

Aside: These objects are all tensor fields

ϕ is a tensor of zeroth order (scalar)

$\underset{\sim}{u}$ is a tensor of first order (vector)

(as is $\nabla \phi$)

$\nabla \underset{\sim}{u}$ is a tensor of second order

(appears as a matrix, but has a connection to the underlying x, y, z coordinate system)

In 2d,

$$\nabla \phi = \begin{bmatrix} \phi_x \\ \phi_y \end{bmatrix}, \quad \nabla \underset{\sim}{u} = \begin{bmatrix} u_x & v_x \\ u_y & v_y \end{bmatrix}$$

② Divergence $\nabla \cdot (\ )$ Not applicable to Scalar fields

$\nabla \cdot \phi$ does not exist

$$\nabla \cdot \underset{\sim}{u} = \begin{bmatrix} \partial_x & \partial_y & \partial_z \end{bmatrix} \begin{bmatrix} u \\ v \\ \end{bmatrix}$$

$$\nabla \cdot \underline{u} = \begin{bmatrix} \partial_x & \partial_y & \partial_z \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

$$= u_x + v_y + w_z$$

scalar!

In 2d, $\nabla \cdot \underline{u} = \begin{bmatrix} \partial_x & \partial_y \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$

$$= u_x + v_y$$

Note: For consistency ∇ should be written $\underline{\nabla}$, Then,

$\underline{\nabla} \cdot \underline{u} = \underline{\nabla}^T \underline{u}$ is an inner product

③ Laplacian $\nabla \cdot \nabla = \nabla^2 \; (= \Delta)$

$$\nabla \cdot \nabla \phi = (\partial_x \;\; \partial_y \;\; \partial_z) \cdot (\phi_x \;\; \phi_y \;\; \phi_z)$$

$$= \phi_{xx} + \phi_{yy} + \phi_{zz}$$

$$\nabla \cdot \nabla \underline{u} = \begin{bmatrix} u_{xx} + u_{yy} + u_{zz} \\ v_{xx} + v_{yy} + v_{zz} \\ w_{xx} + w_{yy} + w_{zz} \end{bmatrix}$$

$$\begin{bmatrix} \vdots \\ [\text{illegible}] \\ w_{xx} + w_{yy} + w_{zz} \end{bmatrix}$$

In 2d, $\nabla \cdot \nabla \underline{u} = \begin{bmatrix} u_{xx} + u_{yy} \\ v_{xx} + v_{yy} \end{bmatrix}$

(4) Curl $\quad \nabla \times (\quad)$ $\qquad$ Only operates on vectors & tensors of higher order

$$\nabla \times \underline{u} = \det\left(\begin{bmatrix} \underline{e}_x & \underline{e}_y & \underline{e}_z \\ \partial_x & \partial_y & \partial_z \\ u & v & w \end{bmatrix} \right)$$

$$= \begin{bmatrix} w_y - v_z \\ u_z - w_x \\ v_x - u_y \end{bmatrix}$$

Alternatively, curl can be written is terms of Levi-Civita symbol

In 2d: $\nabla \times \underline{u} = (v_x - u_y)\, \underline{e}_z$

Operators (1) - (4) are used primarily to build PDEs