

Xue Yu

2020-04-28

Contents

		7
1	{Gaussian elimination}	9
1.1	9
1.2	11
2	Cholesky {Cholesky decomposition}	13
2.1	$A^T A$	13
2.2	14
2.3	14
2.4	Cholesky	15
2.5	15
3	QR	17
3.1	Gram-Schmidt	17
3.2	Householder	19
3.3	21
4	{Eigen decomposition}	23
4.1	23
4.2	26
4.3	28
5	{SVD decomposition}	33
5.1	33
5.2	37
5.3	39
5.4	39
5.5	42
6	{SVD application}	45
6.1	Rigid Alignment / Procrustes Problem	45
6.2	APAR	50
6.3	PCA	50

6.4	50
7	{matirx application}	53
7.1	53
7.2	56
8	{nonlinear equation}	61
8.1	Bisection method	61
8.2	fixed point	62
8.3	Newton's method	62
8.4	Secant method	62
8.5	63
9	{nonlinear equations}	65
9.1	65
9.2	Broyden's method	66
9.3	67
10	{Points Concepts}	69
10.1	critial point	69
10.2	stationary point	69
10.3	inflection point	69
10.4	saddle point	73
10.5	vertex (curve)	75
11	Jacobian Hessian {Gradient Related Concepts}	77
11.1	77
11.2	77
11.3	Jacobian	78
11.4	Hessian	79
12	{Optimization without constraintss}	81
12.1	81
12.2	83
12.3	83
12.4	87
13	KKT {Lagrange multiplier to KKT condition}	89
13.1	89
13.2	95
13.3	KKT	96
13.4	KKT	97
13.5	98
14	{Conjugate gradient}	101
14.1	Gradient descent	101
14.2	Conjugate gradient	102

14.3	106
15	{Interpolate}	109
15.1	109
15.2	110
15.3	110
15.4	111
16	{Numerical Intergration}	113
16.1	113
16.2	117
17	{ODE}	121
17.1	121
17.2	ODE	122
17.3	123
17.4	125
17.5	ODE	126
17.6	127
18	{PDE}	129
18.1	129
18.2	129
18.3	- Navier-Stokes equations	130
18.4	Maxwell's equations	130
18.5	Laplace's equation	131
18.6	Harmonic analysis	133
18.7	Boundary Value Problems	133
18.8	PDE	134
18.9	PDE	134
18.10	PDE	134
18.11	PDE	134
18.12	134

R bookdown

Chapter 1

{Gaussian elimination}

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

$$A \in R^{m \times n}, x \in R^n, b \in R^m, \quad A$$

1.1

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \dots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = b_m$$

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} & b_m \end{bmatrix}$$

- row_1 / a_{11}
- $row_2 - k * row_1, k = \frac{a_{21}}{a_{11}}$
- ...

$$\begin{bmatrix} 1 \\ \vdots \\ 0 \end{bmatrix}$$

- $row_2/(current)a_{22}$
- ...

$$\left[\begin{array}{cccc|c} 1 & \blacksquare & \dots & \blacksquare & \blacksquare \\ 0 & \blacksquare & \dots & \blacksquare & \blacksquare \\ 0 & \blacksquare & \dots & \blacksquare & \blacksquare \end{array} \right] \rightarrow \left[\begin{array}{cccc|c} 1 & \blacksquare & \dots & \blacksquare & \blacksquare \\ 0 & 1 & \dots & \blacksquare & \blacksquare \\ 0 & 0 & \dots & \blacksquare & \blacksquare \end{array} \right] \rightarrow \left[\begin{array}{cccc|c} 1 & \blacksquare & \dots & \blacksquare & \blacksquare \\ 0 & 1 & \dots & \blacksquare & \blacksquare \\ 0 & 0 & \dots & 1 & \blacksquare \end{array} \right]$$

A x_m , $x_1 \dots x_m$

$$\left[\begin{array}{cccc|c} 1 & \blacksquare & \dots & \blacksquare & \blacksquare \\ 0 & 1 & \dots & \blacksquare & \blacksquare \\ & & \dots & & \\ 0 & 0 & & 1 & \blacksquare \end{array} \right] \rightarrow \left[\begin{array}{cccc|c} 1 & & & & \blacksquare \\ & 1 & & & \blacksquare \\ & & & & \blacksquare \\ 0 & & & 1 & \blacksquare \end{array} \right]$$

- x_m get
- $row_{m-1} - k * row$
- ...

- $row * k$
- $row_j + row_i * k$

- LU decomposition: $A = LU$;
- Forward substitution: solve $Ly = b$;
- Backward substitution: solve $Ux = y$.

- $a_{11} \quad 0 \quad 0$
- $a_{11} \ll 1$ overflow

pivot $A\mathbf{x} = \mathbf{b}$ Gaussian elimination with partial
pivoting (GEPP).

[https://web.mit.edu/10.001/Web/Course_Notes/GaussElimPivoting.h
tml](https://web.mit.edu/10.001/Web/Course_Notes/GaussElimPivoting.html)

partial pivoting total pivoting a_{11} ,

$$A = PLU$$

$$\begin{bmatrix} \blacksquare & \blacksquare & \dots & \blacksquare \\ \blacksquare & \blacksquare & \dots & \blacksquare \\ \blacksquare & \blacksquare & \dots & \blacksquare \end{bmatrix} = \mathbf{P} \begin{bmatrix} \blacksquare & & & \\ & \blacksquare & & \\ & & \blacksquare & \\ & & & \blacksquare \end{bmatrix} \begin{bmatrix} \blacksquare & & & \\ & \blacksquare & & \\ & & \blacksquare & \\ & & & \blacksquare \end{bmatrix}$$

A **L** **U**

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{b} \end{array} \right] \rightarrow \left[\begin{array}{c|c} \mathbf{I} & \mathbf{A}^{-1}\mathbf{b} \end{array} \right]$$

$$A^{-1}$$

1.2

`scipy.linalg.solve` , A^{-1} $A^{-1}\mathbf{b}$

$$\begin{cases} 2x_1 + 4x_2 - 2x_3 = 2 \\ x_1 - 3x_2 - 3x_3 = -1 \\ 4x_1 + 2x_2 + 2x_3 = 3 \end{cases}$$

```
import numpy as np
from scipy import linalg

a = np.array([[2, 4, -2],
              [1, -3, -3],
              [4, 2, 2]])
b = np.array([2, -1, 3])

x = linalg.solve(a, b)
np.dot(a, x) # array([ 2., -1.,  3.])
x # array([0.5, 0.33333333, 0.16666667])

array([0.5, 0.33333333, 0.16666667])
```

$$\begin{cases} x_1 = 1/2 \\ x_2 = 1/3 \\ x_3 = 1/6 \end{cases}$$

lu scipy.linalg.lu

Chapter 2

Cholesky {Cholesky decomposition}

Cholesky $\mathcal{O}(n^3)$ operations

2.1 $A^T A$

$$Ax = b$$

$$x = A^{-1}b \quad A \quad A^{-1}.$$

$$\|Ax - b\|_2$$

$$\begin{aligned} \|Ax - b\|^2 &= (Ax - b) \cdot (Ax - b) \\ &= (Ax - b)^T \cdot (Ax - b) \\ &= (x^T A^T - b^T) \cdot (Ax - b) \\ &= (x^T A^T Ax - 2b^T Ax + b^T b) \end{aligned}$$

Error $\|x\|$

$$\frac{\partial E}{\partial \mathbf{x}} = 2A^T A \mathbf{x} - 2A^T \mathbf{b} = 0$$

$$A^T A \mathbf{x} = A^T \mathbf{b} \quad (1)$$

(1) least_squares_SP

(1)

$$\mathbf{x} = (A^T A)^{-1} A^T \mathbf{b} \quad (2)$$

(1) (2) $(A^T A)^{-1}$, A $A^T A$ Tikhonov regularization

2.2

$$A^T A \quad (AB)^T = B^T A^T$$

$$(A^T A)^T = A^T (A^T)^T = A^T A$$

2.3

$$M \quad \Longleftrightarrow \quad x^T M x > 0 \text{ for all } x \in \mathbb{R}^n \quad \mathbf{0}$$

$$M \quad \Longleftrightarrow \quad x^T M x \geq 0 \text{ for all } x \in \mathbb{R}^n$$

$$A \quad A \quad A^T A \quad :$$

$$x^T A^T A x = (Ax)^T \cdot Ax = \|Ax\|^2$$

Cholesky

2.4 Cholesky

A SPD (real Symmetric positive definite matrix) A A
lower triangle L upper triangle L^T.

$$\begin{bmatrix} \blacksquare & \blacksquare & \dots & \blacksquare \\ \blacksquare & \blacksquare & \dots & \blacksquare \\ \blacksquare & \blacksquare & \dots & \blacksquare \end{bmatrix} = \begin{bmatrix} \blacksquare & & \\ & \blacksquare & \\ & & \blacksquare \end{bmatrix} \begin{bmatrix} \blacksquare & & \\ & \blacksquare & \\ & & \blacksquare \end{bmatrix}$$

A

L

L^T

$$\mathbf{A} = \mathbf{A}^T$$

$$\mathbf{x}^T \mathbf{A} \mathbf{x} > 0 \ (\mathbf{x} \neq \mathbf{0})$$

How to prove the existence and uniqueness of Cholesky decomposition?

$A = PLU$

A

$A = LL^T$

Cholesky LU

2.5

$$\begin{pmatrix} 4 & 12 & -16 \\ 12 & 37 & -43 \\ -16 & -43 & 98 \end{pmatrix} = \begin{pmatrix} 2 & 0 & 0 \\ 6 & 1 & 0 \\ -8 & 5 & 3 \end{pmatrix} \begin{pmatrix} 2 & 6 & -8 \\ 0 & 1 & 5 \\ 0 & 0 & 3 \end{pmatrix}$$

```
scipy.linalg.cholesky
import numpy as np
from scipy import linalg
```

```
a = np.array([[4, 12, -16],
              [12, 37, -43],
              [-16, -43, 98]])

L = linalg.cholesky(a, lower=True) #    upper    lower = True

# array([[ 2.,  0.,  0.],
#        [ 6.,  1.,  0.],
#        [-8.,  5.,  3.]])

np.allclose(np.dot(L, L.T) , a) #
```


Chapter 3

QR

QR

$$A = QR$$

Q R upper triangle .

Q

- $QQ^T = I$
- $\det(Q) = \pm 1, \quad \det(Q) = 1 \qquad \text{SO}(n)$
-

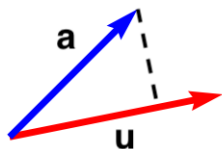
3.1 Gram-Schmidt

Gram-Schmidt Q:

projection

$$\text{proj}_{\mathbf{u}} \mathbf{a} = \frac{\langle \mathbf{u}, \mathbf{a} \rangle}{\langle \mathbf{u}, \mathbf{u} \rangle} \mathbf{u}$$

$$\mathbf{a} \quad \mathbf{u}$$



$$A = [\mathbf{a}_1, \dots, \mathbf{a}_n],$$

$$\begin{aligned} \mathbf{u}_1 &= \mathbf{a}_1, & \mathbf{e}_1 &= \frac{\mathbf{u}_1}{\|\mathbf{u}_1\|} \\ \mathbf{u}_2 &= \mathbf{a}_2 - \text{proj}_{\mathbf{u}_1} \mathbf{a}_2, & \mathbf{e}_2 &= \frac{\mathbf{u}_2}{\|\mathbf{u}_2\|} \\ \mathbf{u}_3 &= \mathbf{a}_3 - \text{proj}_{\mathbf{u}_1} \mathbf{a}_3 - \text{proj}_{\mathbf{u}_2} \mathbf{a}_3, & \mathbf{e}_3 &= \frac{\mathbf{u}_3}{\|\mathbf{u}_3\|} \\ &\vdots & &\vdots \\ \mathbf{u}_k &= \mathbf{a}_k - \sum_{j=1}^{k-1} \text{proj}_{\mathbf{u}_j} \mathbf{a}_k, & \mathbf{e}_k &= \frac{\mathbf{u}_k}{\|\mathbf{u}_k\|} \end{aligned}$$

A

$$\begin{aligned} \mathbf{a}_1 &= \langle \mathbf{e}_1, \mathbf{a}_1 \rangle \mathbf{e}_1 \\ \mathbf{a}_2 &= \langle \mathbf{e}_1, \mathbf{a}_2 \rangle \mathbf{e}_1 + \langle \mathbf{e}_2, \mathbf{a}_2 \rangle \mathbf{e}_2 \\ \mathbf{a}_3 &= \langle \mathbf{e}_1, \mathbf{a}_3 \rangle \mathbf{e}_1 + \langle \mathbf{e}_2, \mathbf{a}_3 \rangle \mathbf{e}_2 + \langle \mathbf{e}_3, \mathbf{a}_3 \rangle \mathbf{e}_3 \\ &\vdots \\ \mathbf{a}_k &= \sum_{j=1}^k \langle \mathbf{e}_j, \mathbf{a}_k \rangle \mathbf{e}_j \end{aligned}$$

$$Q = [\mathbf{e}_1, \dots, \mathbf{e}_n]$$

$$R = \begin{pmatrix} \langle \mathbf{e}_1, \mathbf{a}_1 \rangle & \langle \mathbf{e}_1, \mathbf{a}_2 \rangle & \langle \mathbf{e}_1, \mathbf{a}_3 \rangle & \dots \\ 0 & \langle \mathbf{e}_2, \mathbf{a}_2 \rangle & \langle \mathbf{e}_2, \mathbf{a}_3 \rangle & \dots \\ 0 & 0 & \langle \mathbf{e}_3, \mathbf{a}_3 \rangle & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

R

$$\mathbf{a}_1 \quad \mathbf{e}_1 \quad \mathbf{a}_2 \quad \mathbf{e}_1, \mathbf{e}_2 \quad \mathbf{a}_k \quad \mathbf{e}_1 \cdots \mathbf{e}_k$$

Gram-Schmidt $\begin{pmatrix} 1 & 1+\varepsilon \\ 1 & 1 \end{pmatrix}, \varepsilon \ll 1,$

3.2 Householder

projection

$$\text{proj}_{\mathbf{u}} \mathbf{a} = \frac{\langle \mathbf{u}, \mathbf{a} \rangle}{\langle \mathbf{u}, \mathbf{u} \rangle} \mathbf{u}$$

$$\text{proj}_{\mathbf{u}} \mathbf{a} = \frac{\mathbf{u}^T \cdot \mathbf{a}}{\mathbf{u}^T \cdot \mathbf{u}} \cdot \mathbf{u} = \frac{(\mathbf{u}^T \cdot \mathbf{a}) \cdot \mathbf{u}}{\mathbf{u}^T \cdot \mathbf{u}}$$

$$\text{proj}_{\mathbf{u}} \mathbf{a} = \frac{\mathbf{u} \cdot \mathbf{u}^T \cdot \mathbf{a}}{\mathbf{u}^T \cdot \mathbf{u}}$$

$$\mathbf{u} \cdot \mathbf{u}^T \quad 3 \times 3$$

$$P = \frac{\mathbf{u} \cdot \mathbf{u}^T}{\mathbf{u}^T \cdot \mathbf{u}}$$

$$P^2 = P \dots$$

$$\begin{aligned} \mathbf{b} - 2 \text{proj}_{\mathbf{u}} \mathbf{b} &= \mathbf{b} - 2 \frac{\langle \mathbf{u}, \mathbf{b} \rangle}{\langle \mathbf{u}, \mathbf{u} \rangle} \mathbf{u} \\ &= \mathbf{b} - 2 \frac{\mathbf{u} \cdot \mathbf{u}^T}{\mathbf{u}^T \cdot \mathbf{u}} \mathbf{b} \\ &= (I - 2 \frac{\mathbf{u} \cdot \mathbf{u}^T}{\mathbf{u}^T \cdot \mathbf{u}}) \mathbf{b} \\ &= H_u \mathbf{b} \end{aligned}$$

$$H_u = I - 2 \frac{\mathbf{u} \cdot \mathbf{u}^T}{\mathbf{u}^T \cdot \mathbf{u}}$$

$$\begin{pmatrix} x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \end{pmatrix} \rightarrow \begin{pmatrix} 1 & x & x & x \\ 0 & x & x & x \\ 0 & x & x & x \\ 0 & x & x & x \end{pmatrix}$$

$$R = H_{u_n} \cdots H_{u_1} A Q = H_{u_1}^T \cdots H_{u_n}^T$$

Gram-Schmidt Householder $A^{m \times n}$

- Gram-Schmidt : $Q \in R^{m \times n}, R \in R^{n \times n}$
- Householder: $Q \in R^{m \times m}, R \in R^{m \times n}$

Householder Gram-Schmidt.

3.3

$$A = \begin{pmatrix} 12 & -51 & 4 \\ 6 & 167 & -68 \\ -4 & 24 & -41 \end{pmatrix}$$

$$Q = \begin{pmatrix} 0.8571 & -0.3943 & 0.3314 \\ 0.4286 & 0.9029 & -0.0343 \\ -0.2857 & 0.1714 & 0.9429 \end{pmatrix}$$

$$R = \begin{pmatrix} 14 & 21 & -14 \\ 0 & 175 & -70 \\ 0 & 0 & -35 \end{pmatrix}$$

```
QR    scipy.linalg.qr:
import numpy as np
from scipy import linalg

a = np.array([[12, -51, 4],
              [6, 167, -68],
              [-4, 24, -41]])

q, r = linalg.qr(a)

q
# array([[ -0.85714286,  0.39428571,  0.33142857],
#        [ -0.42857143,  0.90285714, -0.03428571],
#        [ 0.28571429, -0.17142857,  0.94285714]])
```

```
r
# array([[ -14.,  -21.,   14.],
#        [   0., -175.,   70.],
#        [   0.,    0.,  -35.]])

np.allclose(np.dot(q, r), a) # True
```

$$A = QR \quad \begin{matrix} \mathbf{e}_1 \cdots \mathbf{e}_n \\ \vdots \end{matrix}$$

Chapter 4

{Eigen decomposition}

4.1

4.1.1

eigen

$$\mathbf{x}_i,$$

$$\mathbf{v}_i$$

$$\text{minimize } \sum_i ||\mathbf{x}_i - \text{proj}_{\mathbf{v}} \mathbf{x}_i||^2 ||\mathbf{v}|| = 1$$

$$||\mathbf{v}|| = 1 \quad \mathbf{v}$$

$$\begin{aligned} \sum ||\mathbf{x}_i - \text{proj}_{\mathbf{v}} \mathbf{x}_i||^2 &= ||\mathbf{x}_i - (\mathbf{x}_i \cdot \mathbf{v}) \mathbf{v}||^2 \\ &= ||\mathbf{x}_i||^2 - 2(\mathbf{x}_i \cdot \mathbf{v})^2 + (\mathbf{x}_i \cdot \mathbf{v})^2 \\ &= ||\mathbf{x}_i||^2 - (\mathbf{x}_i \cdot \mathbf{v})^2 \end{aligned}$$

$$\mathbf{x}_i \quad \mathbf{v}$$

$$\sum_i (\mathbf{x}_i \cdot \mathbf{v})^2$$

$$||X^T \mathbf{v}||^2 ||\mathbf{v}|| = 1$$

$$\frac{\mathbf{x}_i \cdot \mathbf{v}}{\mathbf{v}} \quad \mathbf{v} \quad X^T \mathbf{v} \quad ||\mathbf{v}|| = 1,$$

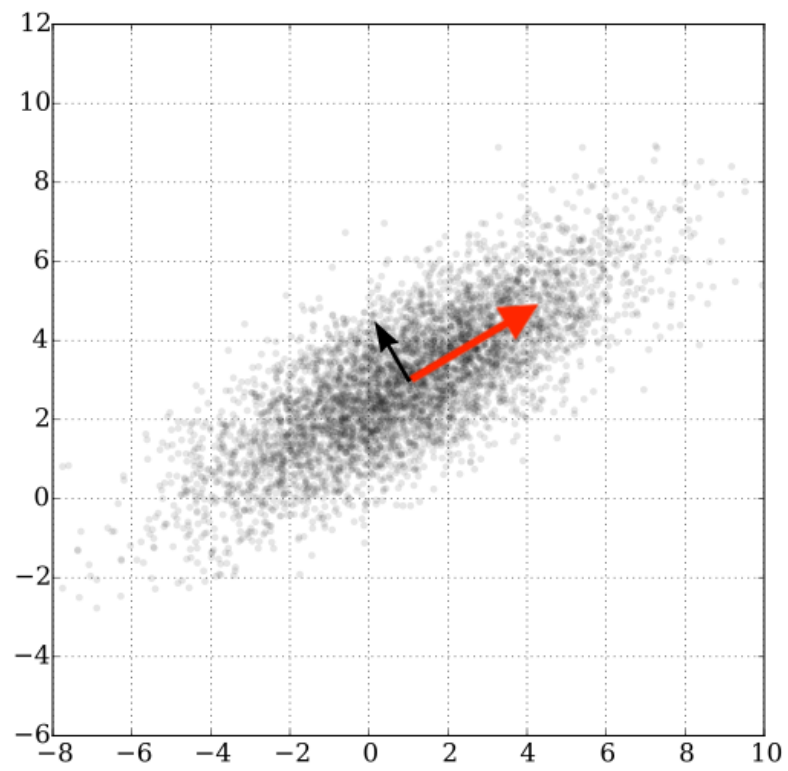


Figure 4.1: wikipedia

$$\|X^T \mathbf{v}\|^2 = \mathbf{v}^T X X^T \mathbf{v}$$

$$X X^T \mathbf{v} = \lambda \mathbf{v}$$

$X X^T$ eigenvalue eigenvector.

4.1.2

eigenvalue eigenvector

$$F = m \frac{d^2 \mathbf{x}}{dt^2} = -k \mathbf{x}$$

D:

$$D^2 : f[\mathbf{x}] \rightarrow f[\mathbf{x}] \mathbf{x} \mapsto D^2 \mathbf{x} = \lambda \mathbf{x}$$

4.1.3 Quadratic Energy

setup

Have:

- n items in a dataset
- $w_{ij} \geq 0$ similarity of items i and j
- $w_{ij} = w_{ji}$

Want: - x_i embedding on \mathbb{R}

energy

$$E(\mathbf{x}) = \sum_{ij} w_{ij} (x_i - x_j)^2$$

$$E(\mathbf{x})$$

$$\|\mathbf{x}\|^2 = \mathbf{1}^T \mathbf{x} = 0$$

0.

$$\Lambda = \sum_{ij} w_{ij} (x_i - x_j)^2 - \lambda (\mathbf{x}^T \cdot \mathbf{x} - 1) - \mu (\mathbf{1} \cdot \mathbf{x})$$

$$E(\mathbf{x}) = \mathbf{x}^T(2A - 2W)\mathbf{x}$$

$$2A - 2W \qquad \qquad \qquad 0 \qquad \mathbf{0}.$$

- Spectral Graph Partitioning and the Laplacian with Matlab
- The Smallest Eigenvalues of a Graph Laplacian

4.2

4.2.1

$$A\mathbf{x} = \lambda\mathbf{x} \lambda \in \mathbb{R}, A \in \mathbb{R}^{n \times n}$$

$$A\mathbf{x} = \lambda\mathbf{x} \lambda \in \mathbb{C}, \mathbf{x} \in \mathbb{C}^n$$

$$\lambda \qquad \mathbf{x}$$

4.2.2

$$\begin{matrix} A & n \times n \\ A & A \end{matrix} \qquad \qquad \qquad (A) \quad \rho(A) = \max \{ |\lambda_1|, \cdots, |\lambda_n| \}$$

4.2.3

- $A \in \mathbb{R}^{n \times n}$
- $A \qquad n \qquad \text{rank } n \qquad n \qquad \text{column vector}$

4.2.4

$$\begin{matrix} z = a + ib \in \mathbb{C} & \bar{z} = a - ib \\ A \text{ (m x n)} \end{matrix}$$

$$\left(A^H\right)_{ij} = \overline{A_{ji}}$$

$$A^H = (\overline{A})^T = \overline{A^T}$$

Hermitian

$$A \text{ Hermitian} \iff A = A^H$$

A

$$\begin{bmatrix} 3 & 2+i \\ 2-i & 1 \end{bmatrix}$$

4.2.5

$$A \in \mathbb{C}^{n \times n} \quad A \in \mathbf{R}^{n \times n}, \quad A \text{ n} \quad \mathbf{x}_1, \dots, \mathbf{x}_n \quad (\quad)$$

$$\lambda_1, \dots, \lambda_n.$$

$$\text{span } \mathbb{R}^n$$

$I :$

$$I \text{ n} \quad \text{eigenvalue } 1 \quad I \quad \text{span } \mathbb{R}^n$$

$$\text{Gram-Schmidt} \quad \text{eigenvalue} \quad \text{eigenvector} \quad \text{span} \quad \text{plane}$$

$$o(\quad)o$$

If a real symmetric matrix has repeated eigenvalues, why does it still have n linearly independent eigenvectors?

- $\{1, \sin x, \cos x, \sin 2x, \cos 2x, \dots, \}$
- $f(x) = a_0/2 + \sum_{n=1}^{\infty} a_n \cos nx + b_n \sin nx$

4.2.6

$$A\mathbf{v} = \lambda\mathbf{v}$$

$$p(\lambda) = \det(A - \lambda I) = 0$$

Fundamental theorem of algebra $p(\lambda) = 0$ N " " Spectrum
 n n

$$p(\lambda) = (\lambda - \lambda_1)^{n_1} (\lambda - \lambda_2)^{n_2} \cdots (\lambda - \lambda_k)^{n_k} = 0$$

$$\sum_{i=1}^k n_i = N$$

$$\lambda_i$$

$$(\mathbf{A} - \lambda_i \mathbf{I}) \mathbf{v} = \mathbf{0}$$

$m_i (1 \leq m_i \leq n_i)$ m_i λ_i m_i λ_i geometric mul-
 tiplicity n_i algebraic multiplicity $m_i = n_i =$
 1
 " \mathbf{A} $n \times n$ \mathbf{A} n

4.3

$$\mathbf{A}^k \mathbf{x} = \lambda^k \mathbf{x}$$

$$\mathbf{A}$$

$$\mathbf{A}^{-1} \mathbf{x} = \frac{1}{\lambda} \mathbf{x}$$

setup

$\mathbf{A} \in \mathbb{R}^{n \times n}$ $\mathbf{x}_1, \dots, \mathbf{x}_n \in \mathbb{R}^n$ eigenvector $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n|$ eigenvalues

$$\mathbf{v} \in \mathbb{R}^n$$

$$\mathbf{v} = c_1 \mathbf{x}_1 + \cdots + c_n \mathbf{x}_n$$

$$A^k \mathbf{v} = \lambda_1^k \left(c_1 \mathbf{x}_1 + \left(\frac{\lambda_2}{\lambda_1} \right)^k c_2 \mathbf{x}_2 + \cdots + \left(\frac{\lambda_n}{\lambda_1} \right)^k c_n \mathbf{x}_n \right)$$

4.3.1 eigenvalue eigenvector

$$|\lambda_2| \leq |\lambda_1|$$

$$A^k \mathbf{v} = \lambda_1^k \mathbf{x}$$

$$\mathbf{v}_k = A \mathbf{v}_{k-1}$$

$$\text{eigenvalue} \quad \text{eigenvector} \quad |\lambda_1| \geq 1 \quad \text{normalize}$$

$$\mathbf{w}_k = A \mathbf{v}_{k-1} \mathbf{v}_k = \frac{\mathbf{w}_k}{|\mathbf{w}_k|}$$

norm

4.3.2 eigenvalue eigenvector

eigenvalue eigenvector

$$\mathbf{w}_k = A^{-1} \mathbf{v}_{k-1} \mathbf{v}_k = \frac{\mathbf{w}_k}{|\mathbf{w}_k|}$$

$$A^{-1}$$

$$\left| \frac{1}{\lambda_1} \right| < \left| \frac{1}{\lambda_2} \right| < \cdots < \left| \frac{1}{\lambda_n} \right|$$

4.3.3 σ eigenvalue eigenvector

σ eigenvalue eigenvector :

$$\mathbf{v}_{k+1} = \frac{(A - \sigma I)^{-1} \mathbf{v}_k}{\|(A - \sigma I)^{-1} \mathbf{v}_k\|}$$

$(A - \sigma I)$ eigenvector eigenvalue 0 make sense.

4.3.4 eigenvector eigenvalue

\mathbf{v} eigenvector eigenvalue

$$A\mathbf{v} \approx \lambda\mathbf{v}$$

λ

$$\arg \min_{\lambda} \|A\mathbf{v} - \lambda\mathbf{v}\|^2$$

$\lambda,$

$$\lambda = \frac{\mathbf{v}^T A \mathbf{v}}{\|\mathbf{v}\|^2}$$

Rayleigh quotient)

Rayleigh quotient iteration

$$1. \quad \mathbf{v} \in R^n$$

2.

- $\sigma_k = \frac{\mathbf{v}^T A \mathbf{v}}{\|\mathbf{v}\|^2}$
- $\mathbf{w}_k = (A - \sigma I)^{-1} \mathbf{v}_{k-1}$
- $\mathbf{v}_k = \frac{\mathbf{w}_k}{\|\mathbf{w}_k\|}$

wikipedia Rayleigh quotient iteration

eigenvalue eigenvalue σ

4.3.5 All eigenevalue

eigenvalue

$$\mathbf{v} = c_1 \mathbf{x}_1 + \cdots + c_n \mathbf{x}_n$$

$\mathbf{x} \quad \mathbf{v}_0$

$$\mathbf{v}_0 \cdot \mathbf{x}_1 = 0$$

$\mathbf{v} \quad \mathbf{x}_1 \quad \mathbf{v}_1$

$$\mathbf{v}_1 = c_2 \mathbf{x}_2 + \cdots + c_n \mathbf{x}_n$$

$$A\mathbf{v}_1 = \lambda_2 \left(c_2 \mathbf{x}_2 + \cdots + \left(\frac{\lambda_n}{\lambda_2} \right) c_n \mathbf{x}_n \right) A^k \mathbf{v}_1 = \lambda_2^k \left(c_2 \mathbf{x}_2 + \cdots + \left(\frac{\lambda_n}{\lambda_2} \right)^k c_n \mathbf{x}_n \right)$$

\mathbf{x}_2
eigenvalue.

A

$$\mathbf{x}_1, \cdots, \mathbf{x}_n$$

4.3.6 Householder

A Householder H

$$H\mathbf{x}_1 = \mathbf{e}_1$$

$$\begin{aligned} HAH^T\mathbf{e}_1 &= HAH\mathbf{e}_1 & H &= H^T \\ &= HAH H\mathbf{x}_1 & H^2 &= I \\ &= HA\mathbf{x}_1 \\ &= \lambda_1 H\mathbf{x}_1 \\ &= \lambda_1 \mathbf{e}_1 \end{aligned}$$

:

$$HAH^T = \begin{bmatrix} \lambda_1 & b \\ 0 & B \end{bmatrix}$$

H λ_1 ,

4.3.7 QR

$$A = QRQ^{-1} = Q^TQ^{-1}AQ = Q^TAQ = Q^TQRQ = RQ$$

$$A_1 = AA_k = Q_kR_kA_{k+1} = R_kQ_k$$

$$A_{k+1} = R_k Q_k = Q_k^{-1} Q_k R_k Q_k = Q_k^{-1} A_k Q_k = Q_k^T A_k Q_k$$

A_k eigenvalue

QR algorithm

scipy.linalg

```
>>> import numpy as np
>>> from scipy import linalg
>>> a = np.array([[1, 0], [1, 3]])
>>> linalg.eigvals(a)
array([3.+0.j, 1.+0.j])
```


Chapter 5

{SVD decomposition}

5.1

SVD $A\vec{x} = \vec{y}$

$$R(\vec{x}) = \frac{\|A\vec{x}\|_2}{\|\vec{x}\|_2}$$

$$R(\alpha\vec{x}) = \frac{\|A\alpha\vec{x}\|_2}{\|\alpha\vec{x}\|_2} = \frac{\|\alpha\| \cdot \|A\vec{x}\|_2}{\|\alpha\| \cdot \|\vec{x}\|_2} = \frac{\|A\vec{x}\|_2}{\|\vec{x}\|_2}$$

- $R(\alpha\vec{x}) = R(\vec{x}) \quad \|\vec{x}\|_2 = 1$
- $R(\vec{x}) \geq 0 \quad R^2(\vec{x})$

,

$$L(\vec{x}) = (A\vec{x})^2 - \lambda(\vec{x}^2 - 1)$$

$$(A^T A)\vec{x}_i = \lambda_i \vec{x}_i \tag{1}$$

$$A\vec{x} = \vec{y} \quad A^T A = \text{diag}(\lambda_1, \dots, \lambda_n)$$

- $\lambda_i \geq 0 \forall i, \quad A^T A$
-

$$\begin{aligned}
 & \vec{y}_i = A\vec{x}_i \\
 & \vec{y}_i = \vec{0} \quad \text{if } \vec{x}_i = \vec{0} \\
 & A^T A \vec{x}_i = A^T \vec{y}_i \\
 & A^T A \in \mathbb{R}^{n \times n}, \quad A^T A = A^T A^T, \quad A \in \mathbb{R}^{m \times n}, \quad A A^T \in \mathbb{R}^{m \times m}, \\
 & A A^T A = A, \quad A^T A A^T = A^T
 \end{aligned}$$

```

>>> import numpy as np
>>> a = np.random.rand(3,3)
>>> a
array([[0.73741709, 0.2207241 , 0.60793118],
       [0.00490906, 0.18066958, 0.44795408],
       [0.70657397, 0.5650763 , 0.29043162]])
>>> aat = np.dot(a, a.T)
>>> aat
array([[0.96208341, 0.31582341, 0.82232812],
       [0.31582341, 0.23332846, 0.23566075],
       [0.82232812, 0.23566075, 0.90290852]])
>>> ata = np.dot(a.T, a)
>>> ata
array([[1.04305484, 0.56292085, 0.6557093 ],
       [0.56292085, 0.40067186, 0.37923277],
       [0.6557093 , 0.37923277, 0.6545937 ]])
>>> np.allclose(aat, ata)
False
>>> from scipy import linalg
>>> linalg.eigvals(aat)
array([1.84996505+0.j, 0.0737421 +0.j, 0.17461325+0.j])
>>> linalg.eigvals(ata)
array([1.84996505+0.j, 0.0737421 +0.j, 0.17461325+0.j])

```

$$\begin{aligned}
 & A A^T A = A \\
 & A^T A A^T = A^T
 \end{aligned}$$

$$A, B \in \mathbb{R}^{n \times n}, AB = BA.$$

$$AB\vec{x} = \lambda\vec{x}$$

$$\vec{y} = B\vec{x}, \quad (\lambda \neq 0, \vec{x} \neq \vec{0})$$

$$BA\vec{y} = BAB\vec{x} = B\lambda\vec{x} = \lambda B\vec{x} = \lambda\vec{y}$$

$$\vec{y}_i = A\vec{x}_i \quad \vec{y}_i = \vec{0} \quad \text{if } \vec{x}_i = \vec{0}$$

$$\begin{aligned}
\lambda_i \vec{y}_i &= \lambda_i A \vec{x}_i \\
&= A(\lambda_i \vec{x}_i) \\
&= A(A^T A \vec{x}_i) \quad \text{from (1)} \\
&= (AA^T) \vec{y}_i \quad (2)
\end{aligned}$$

$$\vec{y}_i \quad AA^T$$

$$\begin{aligned}
\| \vec{y}_i \| &= \| A \vec{x}_i \| \\
&= \sqrt{\| \lambda_i A \vec{x}_i \|^2} \\
&= \sqrt{\vec{x}_i^T A^T A \vec{x}_i} \\
&= \sqrt{\vec{x}_i^T A^T A \vec{x}_i} \\
&= \sqrt{\vec{x}_i^T \lambda_i \vec{x}_i} \quad \text{from (1)} \\
&= \sqrt{\lambda_i \vec{x}_i^T \vec{x}_i} \\
&= \sqrt{\lambda_i} \| \vec{x}_i \|
\end{aligned}$$

$$\begin{aligned}
\lambda_i = 0, \vec{y}_i = \vec{0}, \quad \vec{y}_i \quad \vec{0} \quad AA^T \quad \| \vec{y}_i \| &= \sqrt{\lambda_i} \| \vec{x}_i \|. \\
k \quad A^T A \quad 0 \quad \lambda_1, \dots, \lambda_k, \quad \vec{x}_1, \dots, \vec{x}_k \in \mathbb{R}^n, \quad AA^T \quad A^T A
\end{aligned}$$

$$k = \text{number of } \lambda_i > 0 \quad A^T A \vec{x}_i = \lambda_i \vec{x}_i \quad AA^T \vec{y}_i = \lambda_i \vec{y}_i$$

$$\| \vec{x}_i \| = 1,$$

$$\vec{y}_i = \frac{1}{\sqrt{\lambda_i}} A \vec{x}_i \quad (3)$$

$$\| \vec{y}_i \| = \frac{1}{\sqrt{\lambda}} \| A \vec{x}_i \| = \frac{1}{\sqrt{\lambda}} \sqrt{\lambda} \| \vec{x}_i \| = 1$$

$$\vec{x}_i \quad \vec{y}_i$$

$$\bar{V} = (\vec{x}_1 \quad \dots \quad \vec{x}_k) \in \mathbb{R}^{n \times k} \quad \bar{U} = (\vec{y}_1 \quad \dots \quad \vec{y}_k) \in \mathbb{R}^{m \times k}$$

$$\vec{e}_1 \quad \text{i}$$

$$\begin{aligned}
\bar{U}^T A \bar{V} \bar{e}_1 &= \bar{U}^T A \bar{x}_i && \bar{V} \text{ defination} \\
&= \frac{1}{\lambda_i} \bar{U}^T A (\lambda_i \bar{x}_i) \\
&= \frac{1}{\lambda_i} \bar{U}^T A (A^T A \bar{x}_i) && \text{from (1)} \\
&= \frac{1}{\lambda_i} \bar{U}^T (A A^T) A \bar{x}_i \\
&= \frac{1}{\sqrt{\lambda_i}} \bar{U}^T (A A^T) \vec{y}_i && \text{from (3)} \\
&= \frac{1}{\sqrt{\lambda_i}} \bar{U}^T \lambda_i \vec{y}_i && \text{from (2)} \\
&= \sqrt{\lambda_i} \bar{U}^T \vec{y}_i \\
&= \sqrt{\lambda_i} \bar{e}_i
\end{aligned}$$

$$\Sigma = \text{diag}(\sqrt{\lambda_1}, \dots, \sqrt{\lambda_k}),$$

$$\bar{U}^T A \bar{V} = \Sigma$$

$$\bar{U} \in \mathbb{R}^{m \times k}, \bar{V} \in \mathbb{R}^{n \times k}, A \in \mathbb{R}^{m \times n}, \Sigma \in \mathbb{R}^{k \times k}$$

$$\bar{U}, \bar{V} \quad A^T A \bar{x}_i = \vec{0} \quad A A^T \vec{y}_i = \vec{0}, \quad \bar{U}, \bar{V}$$

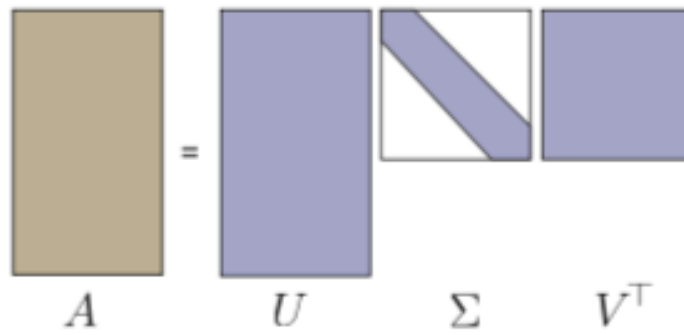
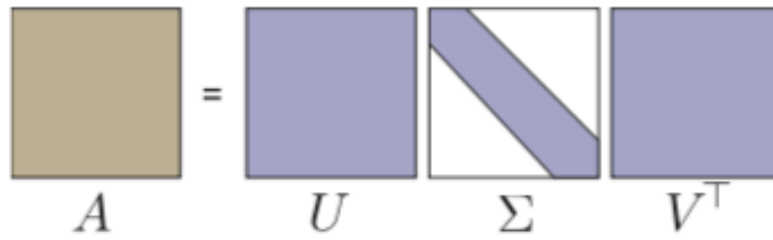
$$\bar{U} \in \mathbb{R}^{m \times k}, \bar{V} \in \mathbb{R}^{n \times k} \mapsto U \in \mathbb{R}^{m \times m}, V \in \mathbb{R}^{n \times n}$$

$$\Sigma$$

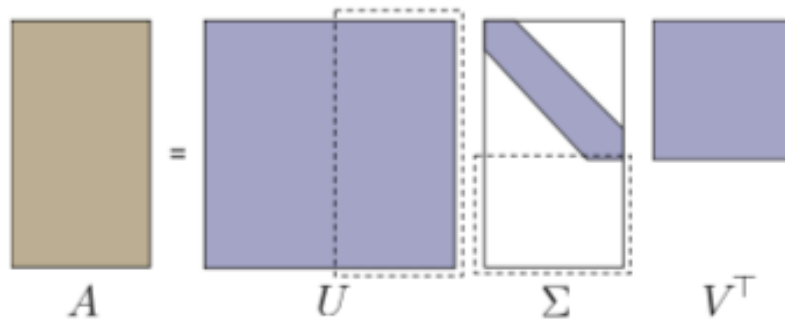
$$\Sigma_{ij} = \begin{cases} \sqrt{\lambda_i} & i = j, i \leq k \\ 0 & \text{otherwise} \end{cases}$$

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

$$A = U \Sigma V^T \quad \mathbf{A}$$



0 U V



5.2

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

- left singular vector) : U span col A
- right singular vector : V span row A (V V^T

- (singular value): $\Sigma \quad \sigma_1 \geq \sigma_2 \cdots \geq 0$

$$\text{SVD} = \mathbf{X} \mathbf{\Lambda} \mathbf{A}$$

singular value decomposition, non-invertible singular
 invertible nonsingular) singular single special /

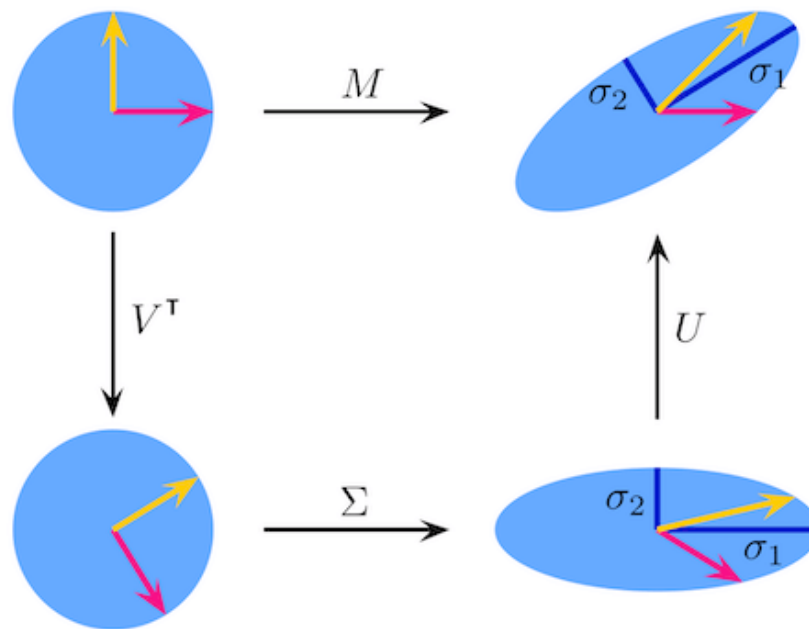
n x n	R	singular	R^{-1} ,	singular value
decompostion		$o(\epsilon)$		

SVD

SVD

- V^T :
- Σ :
- U :

$$\backslash \quad + \quad +$$



$$M = U \cdot \Sigma \cdot V^T$$

5.3

1. $V^T A^T A$
2. $AV = U\Sigma \quad 0 \leq \tilde{u}_i \leq AV$
3. $AA^T \tilde{u}_i = 0$

5.4

5.4.1

A SVD

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

A^{-1} (A nonsingular):

$$\begin{aligned} A^{-1} &= (U\Sigma V^T)^{-1} \\ &= (V^T)^{-1} \Sigma^{-1} U^{-1} \\ &= V \begin{pmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \end{pmatrix}^{-1} U^{-1} \\ &= V \begin{pmatrix} \frac{1}{\sigma_1} & & \\ & \ddots & \\ & & \frac{1}{\sigma_n} \end{pmatrix} U^T \end{aligned}$$

$$A\vec{x} = \vec{b}$$

$$\begin{aligned} A\vec{x} &= \vec{b} \\ U\Sigma V^T \vec{x} &= \vec{b} \\ \vec{x} &= V\Sigma^{-1}U^T \vec{b} \end{aligned}$$

5.4.2

setup

minimize $\|\vec{x}\|_2^2$ such that $A^T A\vec{x} = A^T \vec{b}$

$A^T A$:

$$\begin{aligned}
A^T A &= (U \Sigma V^T)^T (U \Sigma V^T) \\
&= V \Sigma U^T U \Sigma V^T \\
&= V \Sigma^2 V^T
\end{aligned}$$

$$A^T A \vec{x} = A^T \vec{b}$$

$$\begin{aligned}
A^T A \vec{x} = A^T \vec{b} &\iff V \Sigma^2 V^T \vec{x} = (U \Sigma V^T)^T \vec{b} \\
V \Sigma^2 V^T \vec{x} &= V \Sigma U^T \vec{b} \\
\Sigma V^T \vec{x} &= U^T \vec{b}
\end{aligned}$$

$$\begin{aligned}
A^T A \vec{x} = A^T \vec{b} &\iff \Sigma \vec{y} = \vec{d} \\
\vec{y} &= V^T \vec{x} \\
\vec{d} &= U^T \vec{b}
\end{aligned}$$

setup

$$\text{minimize } \|\vec{y}\|_2^2 \text{ such that } \Sigma \vec{y} = \vec{d}$$

$$\vec{y} = V^T \vec{x}$$

$$\text{setup } \vec{x} \quad \vec{y} \quad , \Sigma$$

$$\begin{pmatrix} \sigma_1 & & & \\ & \ddots & & \\ & & \sigma_k & \\ & & & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} d_1 \\ \vdots \\ d_m \end{pmatrix}$$

:

$$\Sigma_{ij}^+ = \begin{cases} \frac{1}{\sigma_i} & i = j, \sigma_i \neq 0, i \leq k \\ 0 & \text{otherwise} \end{cases} \implies \vec{y} = \Sigma_{ij}^+ \vec{d} \implies \vec{x} = V \Sigma_{ij}^+ U^T \vec{b}$$

$$V \Sigma_{ij}^+ U^T \quad \text{Pseudoinverse } ? ,$$

- A square and invertible $A^+ = A^{-1}$
- A overdetermined $A^+ \vec{b}$ gives least-squares solution to $A \vec{x} = \vec{b}$
- A underdetermined $A^+ \vec{b}$ gives least-squares solution to $A \vec{x} = \vec{b}$ with least (Euclidean) norm

5.4.3

A

$$A = U \Sigma V^T \implies A = \sum_{i=1}^l \sigma_i \vec{u}_i \vec{v}_i^T, l = \min\{m, n\}$$

$$\frac{1}{\vec{u} \vec{v}^T} \quad \text{A} \quad \text{x}$$

$$:$$

$$\vec{u} \otimes \vec{v} = \vec{u} \vec{v}^T$$

$$A \vec{x} :$$

$$A \vec{x} = \sum_i \sigma_i (\vec{v}_i \cdot \vec{x}) \vec{u}_i$$

$$A \vec{x} \qquad \sigma_i$$

$$A^+ \vec{x}$$

$$A^+ = \sum_{\sigma_i \neq 0} \frac{\vec{v}_i \cdot \vec{u}^T}{\sigma_i}$$

$$A^+ \qquad \sigma_i$$

Eckart-Young Theorem :

Suppose \tilde{A} is obtained from $A = U \Sigma V^T$ by truncating all but the k largest singular values σ_i of A to zero. Then, \tilde{A} minimizes both $\|A - \tilde{A}\|_{Fro}$ and $\|A - \tilde{A}\|_2$ subject to the constraint that the column space of \tilde{A} has at most dimension k .

$$\begin{matrix} \text{A} & \text{rank} & \tilde{A} & \text{A} & \text{Frobenius norm} & 2\text{-norm,} & \tilde{A} \\ \text{SVD} & \text{A} & A = U \Sigma V^T & \Sigma & k & 0 \end{matrix}$$

Frobenius norm

$$\|A\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2} = \sqrt{\text{trace}(A^* A)} = \sqrt{\sum_{i=1}^{\min\{m,n\}} \sigma_i^2(A)}$$

2-norm

$$\|A\|_2 = \max_{\vec{v} \neq \vec{0}} \frac{\|A\vec{v}\|_2}{\|\vec{v}\|_2} = \max\{\sigma_i\}$$

$$A = U\Sigma V^T \quad \Sigma \quad \sigma_i \geq 0 \quad (\text{The singular values are non-negative real numbers.})$$

$$\Sigma = \text{diag}(\sqrt{\lambda_1}, \dots, \sqrt{\lambda_k})$$

$$A^T A$$

5.5

M:

$$M = \begin{bmatrix} 1 & 0 & 0 & 0 & 2 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \end{bmatrix}$$

$$M = U\Sigma V^*:$$

$$U = \begin{bmatrix} 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & 0 \end{bmatrix} \quad \Sigma = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 \\ 0 & \sqrt{5} & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad V^* = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \\ -\sqrt{0.2} & 0 & 0 & 0 & -\sqrt{0.8} \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ -\sqrt{0.8} & 0 & 0 & 0 & \sqrt{0.2} \end{bmatrix}$$

```
>>> from scipy import linalg
>>> import numpy as np
>>>
>>> a = np.array([[1, 0, 0, 0, 2],
...               [0, 0, 3, 0, 0],
...               [0, 0, 0, 0, 0],
...               [0, 2, 0, 0, 0]])
>>>
>>> u, s, vh = linalg.svd(a)
>>>
>>> u
array([[ 0.,  1.,  0.,  0.],
       [ 1.,  0.,  0.,  0.],
       [ 0.,  0.,  0., -1.],
       [ 0.,  0.,  1.,  0.]])
>>> s
array([3., 2.23606798, 2., 0., 0.])
>>> vh
```

```
array([[ -0.          ,  0.          ,  1.          ,  -0.          ,  0.          ],
       [ 0.4472136 ,  0.          ,  0.          ,  0.          ,  0.89442719],
       [ -0.          ,  1.          ,  0.          ,  -0.          ,  0.          ],
       [ 0.          ,  0.          ,  0.          ,  1.          ,  0.          ],
       [ -0.89442719,  0.          ,  0.          ,  0.          ,  0.4472136 ]])
```

scipy u vh o()o

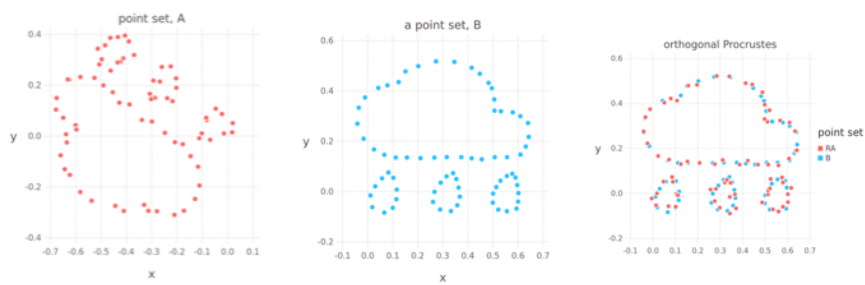
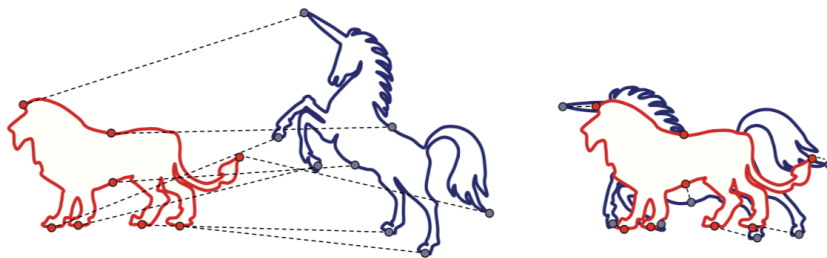
Chapter 6

{SVD application}

SVD

6.1 Rigid Alignment / Procrustes Problem

,



3D 3D point clouds 3D mesh Rigid Alignment.

set up ,

$$E = \sum_{i=1}^n \| p_i - (Rq_i + t) \|^2 \quad p_i \in P, q_i \in Q$$

,

$$p = \frac{1}{n} \sum_{i=1}^n p_i = \frac{1}{n} \sum_{i=1}^n (Rq_i + t) = R \frac{1}{n} \sum_{i=1}^n q_i + t = Rq + t = p - Rq$$

$$\begin{aligned} \sum_{i=1}^n \| p_i - (Rq_i + t) \|^2 &= \sum_{i=1}^n \| p_i - Rq_i - (p - Rq) \|^2 \\ &= \sum_{i=1}^n \| (p_i - p) - R(q_i - q) \|^2 \end{aligned}$$

$$p_i - p \quad q_i - q \quad p \quad q \quad x_i = p_i - p, y_i = q_i - q,$$

$$\begin{aligned} \| x_i - Ry_i \|^2 &= (x_i - Ry_i)^T (x_i - Ry_i) \\ &= (x_i^T - y_i^T R^T) (x_i - Ry_i) \\ &= (x_i^T x_i - x_i^T Ry_i - y_i^T R^T x_i + y_i^T R^T Ry_i) \quad (R^T R = I) \\ &= (x_i^T x_i + y_i^T y_i - x_i^T Ry_i - y_i^T R^T x_i) \end{aligned}$$

$$x_i^T Ry_i \quad x_i^T \quad 1 \times d, R \quad d \times d, y_i \quad d \times 1 \quad a^T = a,$$

$$x_i^T Ry_i = (x_i^T Ry_i)^T = y_i^T Rx_i$$

$$\sum_{i=1}^n \| x_i - Ry_i \|^2 = \sum_{i=1}^n x_i^T x_i + y_i^T y_i - 2y_i^T Rx_i$$

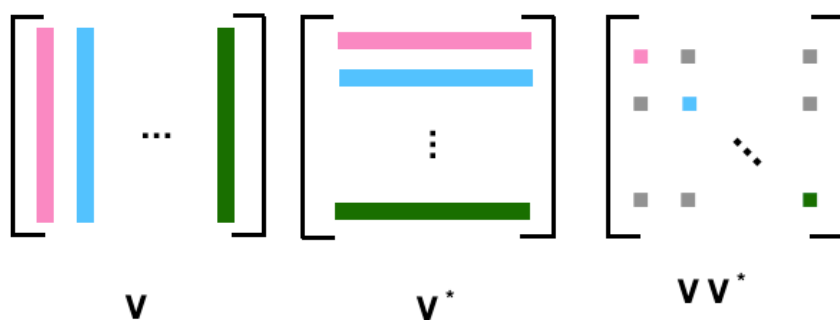
$$x_i^T x_i \quad y_i^T y_i \quad \sum_{i=1}^n y_i^T Rx_i$$

$$\sum_{i=1}^n \| x_i - Ry_i \|^2 = \| X - RY \|_{Fro}^2 = const - 2tr(Y^T RX)$$

- $\text{tr}(A + B) = \text{tr}(A) + \text{tr}(B)$
- $\text{tr}(A^T) = \text{tr}(A)$
- $\|A\|_{Fro}^2 = \sum_{i,j} |a_{ij}|^2$
- $\text{tr}(AB) = \text{tr}(BA)$

$$\text{tr}(A + B) = \text{tr}(A) + \text{tr}(B) \quad \text{tr}(A^T) = \text{tr}(A)$$

Frobenius norm AA^* , AA^*
care



灰色的元素 ■ 我们并不care

$$\text{tr}(AB) = \text{tr}(BA) \quad \text{Tr}(AB) = \text{Tr}(BA) = \sum a_{ij}b_{ji}$$

- $\text{tr}(\mathbf{ABC}) = \text{tr}((\mathbf{ABC})^T) = \text{tr}(\mathbf{CBA}) = \text{tr}(\mathbf{ACB})$,
- $\text{tr}(\mathbf{ABCD}) = \text{tr}(\mathbf{BCDA}) = \text{tr}(\mathbf{CDAB}) = \text{tr}(\mathbf{DABC})$, ABCD

$$-2\text{tr}(Y^T RX) \quad \text{tr}(Y^T RX)$$

$$\begin{aligned}
 \text{tr}(Y^T RX) &= \text{tr}(RXY^T) & \text{tr}(AB) &= \text{tr}(BA) \\
 &= \text{tr}(RU\Sigma V^T) & XY^T &= U\Sigma V^T \\
 &= \text{tr}(\Sigma V^T RU) & \text{tr}(AB) &= \text{tr}(BA) \\
 &= \text{tr}(\Sigma M) & M &= V^T RU, \text{ also orthogonal} \\
 &= \sum_i \sigma_i m_{ii} & \Sigma & \text{ is diagonal}
 \end{aligned}$$

$$M = V^T RU$$

$$AA^T = A^T A = IBB^T = B^T B = I(AB)^T(AB) = B^T A^T AB = B^T B = I$$

$$M = V^T R U = V^T (R U),$$

$$tr(\Sigma M) = \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_n \end{bmatrix} \begin{bmatrix} m_{11} & m_{12} & \dots & m_{1d} \\ m_{21} & m_{22} & \dots & m_{2d} \\ \vdots & \vdots & \vdots & \vdots \\ m_{d1} & m_{d2} & \dots & m_{dd} \end{bmatrix} = \sum_{i=1}^d \sigma_i m_{ii} \leq \sum_{i=1}^d \sigma_i$$

M

$$1 = m_j^T m_j = \sum_{i=1}^d m_{ij}^2 \implies m_{ij}^2 \leq 1 \implies |m_{ij}| \leq 1$$

$$\text{M} \qquad m_i i \leq 1, \qquad \sigma_i \geq 0, \qquad \text{M} \qquad \text{M} = \text{I}$$

$$I = M = V^T R U \implies V = R U \implies R = V U^T$$

Procrustes Problem

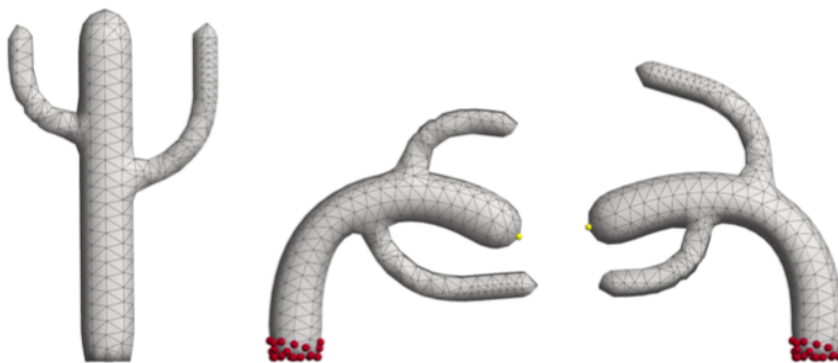
$$\Pi \qquad \Delta$$



$\|X - RY\|^2 \quad \mathbf{R} \quad R = VU^T, \quad XY^T = U\Sigma V^T$

- 1. $\mathbf{R}^T \mathbf{E}$
- 2. $\mathbf{R}^T \mathbf{E} \mathbf{R} \quad R^T R = I$
- 3.

6.2 APAR



paper As-Rigid-As-Possible Surface Modeling, key idea SVD

6.3 PCA

PCA

- : $m = \frac{1}{n} \sum_{i=1}^n x_i$
- : $y_i = x_i - m$
- / : $S = YY^T$ $Y = [y_1 \dots y_n]^T$
- : $S = V\Lambda V^T$
- : $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d$
- : v_1, \dots, v_d
-

6.4

Eckart-Young Theorem ():

$$\text{rank } r \quad \tilde{A} = U \Sigma_r V^T, \quad \| \tilde{A} - A \|_{Fro} \text{ is minimized when } A = U \Sigma V^T, \quad r$$

$$\Sigma_r = \begin{bmatrix} \sigma_1 & & & & \\ & \ddots & & & \\ & & \sigma_r & & \\ & & & 0 & \\ & & & & \ddots \\ & & & & & 0 \end{bmatrix}$$

$x \times n \times 4$ RGBA $m \times n \times 4$, $m \times n$, m
)
 SVD 10 20 50

```

import numpy as np
import matplotlib.pyplot as plt
import matplotlib.image as mpimg

def rgb2gray(rgb):
    return np.dot(rgb[...,:3], [0.299, 0.587, 0.144])

img = mpimg.imread('Mona_Lisa.png')
gray = rgb2gray(img)
plt.imshow(gray, cmap = plt.get_cmap('gray'))

U, s, Vh = np.linalg.svd(gray)

def composite(U, s, Vh, n):
    return np.dot(U[:, :n], np.dot(np.diag(s[:n]), Vh[:n, :]))

for i in [10, 20, 50]:
    new_img = composite(U, s, Vh, i)
    plt.imshow(new_img, cmap='gray')
    title = "new_img_%s" % i
    plt.title(title)
    plt.savefig(title + '.png')
    plt.show()
  
```

original



r = 10



r = 20



r = 50



$$(x,y,z)^T \qquad \qquad \qquad xy \qquad \qquad \qquad (x,y)^T$$

$$f(x) = a_nx^n + a_{n-1}x^{n-a} + \cdots + a_1x + a_0$$

$$a_n \cdots, a_r \qquad \qquad f(x) \qquad \qquad \qquad \mathrm{o}(\quad)\mathrm{o}$$

Chapter 7

{matirx application}

7.1

7.1.1 Conjugate transpose

A

$$A = \begin{bmatrix} 1 & -2-i & 5 \\ 1+i & i & 4-2i \end{bmatrix}$$

A^T

$$A^T = \begin{bmatrix} 1 & 1+i \\ -2-i & i \\ 5 & 4-2i \end{bmatrix}$$

$\overline{A^T}$:

$$\overline{A^T} = \begin{bmatrix} 1 & 1-i \\ -2+i & -i \\ 5 & 4+2i \end{bmatrix}$$

A^*, A^H Hermitian , $\overline{A^T}$

7.1.2 Hermitian

Hermitian matrix $\begin{matrix} i & j \\ j & i \end{matrix}$

$$z = a + ib \in \mathbb{C}, \quad \bar{z} = a - ib$$

$$A \text{ Hermitian} \iff a_{ij} = \overline{a_{ji}} \quad A \text{ Hermitian} \iff A = A^H$$

$A \in \mathbb{R}^{n \times n}$ Hermitian $a_{ij} = a_{ji}$ Hermitian

A

$$A = \begin{bmatrix} 2 & 2+i & 4 \\ 2-i & 3 & i \\ 4 & -i & 1 \end{bmatrix}$$

Hermitian

7.1.3 positive definite

$$n \times n \quad M \quad z \quad z^T M z > 0 \quad z^T \quad z$$

$$M \text{ positive definite} \iff x^T M x > 0 \text{ for all } x \in \mathbb{R}^n \quad \mathbf{0}$$

$$M \quad M = -I :$$

$$\begin{bmatrix} 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} = -2 < 0$$

$$n \times n \quad M \quad z \quad z^* M z > 0 \quad z^* \quad z \quad M \quad z \quad z^* M z \quad 0$$

$$M \text{ positive definite} \iff x^* M x > 0 \text{ for all } x \in \mathbb{C}^n \quad \mathbf{0}$$

Hermitian

- $M \quad \lambda_i$
- ...

7.1.4 orthogonal matrix

$$Q^T = Q^{-1} \Leftrightarrow Q^T Q = Q Q^T = I.$$

$$1 = \det(I) = \det(Q^T Q) = \det(Q^T) \det(Q) = (\det(Q))^2 \Rightarrow \det(Q) = \pm 1$$

-
- $+1$ (special orthogonal group)
- -1
- $n \times n$ $O(n)$
- $SO(n)$

7.1.5 unitary matrix

/

$$U^*U = UU^* = I_n$$

U U^*

yǒu 17:00 19:00

unitary unit take

- $U^{-1} = U^*$,
- $|\lambda_n| = 1$, U λ_n 1
- $|\det(U)| = 1$, U 1
- $(U\vec{x}) \cdot (U\vec{y}) = \vec{x} \cdot \vec{y}$, U \vec{x} \vec{y}
- ...

7.1.6 normal matrix

normal matrix A A

$$\mathbf{A}^* \mathbf{A} = \mathbf{A} \mathbf{A}^*$$

 A^* A

$$A^* = A^T \quad AA^T = A^T A.$$

$$A = U \Lambda U^* \quad \Lambda = \text{diag}(\lambda_1, \lambda_2, \dots) \quad U$$

/ /

7.1.7

-
-
- 1
-
-

7.2

7.2.1 $A = PLU$

- $A = PLU$, L is lower triangular, U is upper triangular, P is permutation matrix.
- PLU decomposition of A .

$$\begin{bmatrix} \blacksquare & \blacksquare & \dots & \blacksquare \\ \blacksquare & \blacksquare & \dots & \blacksquare \\ \blacksquare & \blacksquare & \dots & \blacksquare \end{bmatrix} = P \begin{bmatrix} \blacksquare & & & \\ & \blacksquare & & \\ & & \blacksquare & \\ & & & \blacksquare \end{bmatrix} \begin{bmatrix} \blacksquare & & & \\ & \blacksquare & & \\ & & \blacksquare & \\ & & & \blacksquare \end{bmatrix}$$

$A \qquad \qquad L \qquad \qquad U$

PLU

7.2.2 Cholesky

- A is hermitian positive definite
- $A = LL^*$

$$\begin{bmatrix} \blacksquare & \blacksquare & \dots & \blacksquare \\ \blacksquare & \blacksquare & \dots & \blacksquare \\ \blacksquare & \blacksquare & \dots & \blacksquare \end{bmatrix} = \begin{bmatrix} \blacksquare & & & \\ & \blacksquare & & \\ & & \blacksquare & \\ & & & \blacksquare \end{bmatrix} \begin{bmatrix} \blacksquare & & & \\ & \blacksquare & & \\ & & \blacksquare & \\ & & & \blacksquare \end{bmatrix}$$

$A \qquad \qquad L \qquad \qquad L^*$

$A = A^H$

$x^T A x > 0 \ (x \neq 0)$

A is Hermitian L is lower triangular $L^* L$.

7.2.3 QR

- A is $m \times n$, $m \geq n$
- $A = QR$, Q is $m \times m$ unitary matrix, R is $n \times n$ upper triangular matrix.

QR

$$\begin{array}{ccc}
 \begin{bmatrix} \blacksquare & \blacksquare & \cdots & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare \end{bmatrix} & = & \begin{bmatrix} | & | & \cdots & | \\ \mathbf{q}_1 & \mathbf{q}_2 & \cdots & \mathbf{q}_n \\ | & | & \cdots & | \end{bmatrix} \begin{bmatrix} \blacksquare & \cdots & \blacksquare \\ & \blacksquare & \cdots & \blacksquare \\ & & \blacksquare & \cdots & \blacksquare \\ & & & \blacksquare & \cdots & \blacksquare \end{bmatrix} \\
 \mathbf{A} & & \mathbf{Q} \quad \mathbf{R}
 \end{array}$$

$\mathbf{A} = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{0} \end{bmatrix} \mathbf{R}$

$$A = QR = Q \begin{bmatrix} R_1 \\ 0 \end{bmatrix} = [Q_1, Q_2] \begin{bmatrix} R_1 \\ 0 \end{bmatrix} = Q_1 R_1$$

where R_1 is an $n \times n$ upper triangular matrix, 0 is an $(m - n) \times n$ zero matrix, Q_1 is $m \times n$, Q_2 is $m \times (m - n)$, and Q_1 and Q_2 both have orthogonal columns.

QR Gram-Schmidt Householder reflections.

7.2.4 / Eigendecomposition / spectral decomposition

$$\begin{array}{l}
 \bullet \\
 \bullet \quad \mathbf{A} = \mathbf{Q} \mathbf{\Lambda} \mathbf{Q}^{-1}
 \end{array}$$

$$\begin{array}{cccc}
 \begin{bmatrix} \blacksquare & \cdots & \blacksquare \\ \blacksquare & \cdots & \blacksquare \\ \blacksquare & \cdots & \blacksquare \end{bmatrix} & = & \begin{bmatrix} \blacksquare & \cdots & \blacksquare \\ \blacksquare & \cdots & \blacksquare \\ \blacksquare & \cdots & \blacksquare \end{bmatrix} \begin{bmatrix} \blacksquare & & \\ & \blacksquare & \\ & & \blacksquare \end{bmatrix} \begin{bmatrix} \blacksquare & \cdots & \blacksquare \\ \blacksquare & \cdots & \blacksquare \\ \blacksquare & \cdots & \blacksquare \end{bmatrix} \\
 \mathbf{A} & & \mathbf{Q} & \mathbf{\Lambda} & \mathbf{Q}^{-1}
 \end{array}$$

$$\begin{array}{l}
 \mathbf{Q} \quad n \times n \quad \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \quad \mathbf{A} \quad \vec{q}_i, \mathbf{\Lambda} \quad i \quad \Lambda_{ii} = \lambda_i \quad \vec{q}_i \quad \lambda_i. \\
 \vec{q}_i, (i = 1, \dots, N) \quad \vec{q}_i, (i = 1, \dots, N) \quad \mathbf{Q} \quad \mathbf{Q} \\
 \mathbf{Q}^{-1}
 \end{array}$$

Q

$$\begin{aligned} A\mathbf{v} &= \lambda\mathbf{v} \\ A\mathbf{Q} &= \mathbf{Q}\Lambda \\ A &= \mathbf{Q}\mathbf{Q}^{-1}. \end{aligned}$$

•

$n \times n$ n

1

$$A = \mathbf{Q}\mathbf{Q}^T$$

Q Λ

$$\begin{bmatrix} \blacksquare & \blacksquare & \cdots & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare \end{bmatrix} = \begin{bmatrix} | & | & & | \\ \mathbf{q}_1 & \mathbf{q}_2 & \cdots & \mathbf{q}_n \\ | & | & & | \end{bmatrix} \begin{bmatrix} \diagdown & & & \\ & \diagdown & & \\ & & \diagdown & \\ & & & \diagdown \end{bmatrix} \begin{bmatrix} - & \mathbf{q}_1^T & - \\ - & \mathbf{q}_2^T & - \\ & \cdots & \\ - & \mathbf{q}_n^T & - \end{bmatrix}$$

A Q Λ Q^T
 $A = A^T$ $QQ^T = I$

•

$$A = \mathbf{U}\mathbf{U}^*$$

U

$$\begin{bmatrix} \blacksquare & \blacksquare & \cdots & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare \end{bmatrix} = \begin{bmatrix} \blacksquare & \blacksquare & \cdots & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare \end{bmatrix} \begin{bmatrix} \diagdown & & & \\ & \diagdown & & \\ & & \diagdown & \\ & & & \diagdown \end{bmatrix} \begin{bmatrix} \blacksquare & \blacksquare & \cdots & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare \end{bmatrix}$$

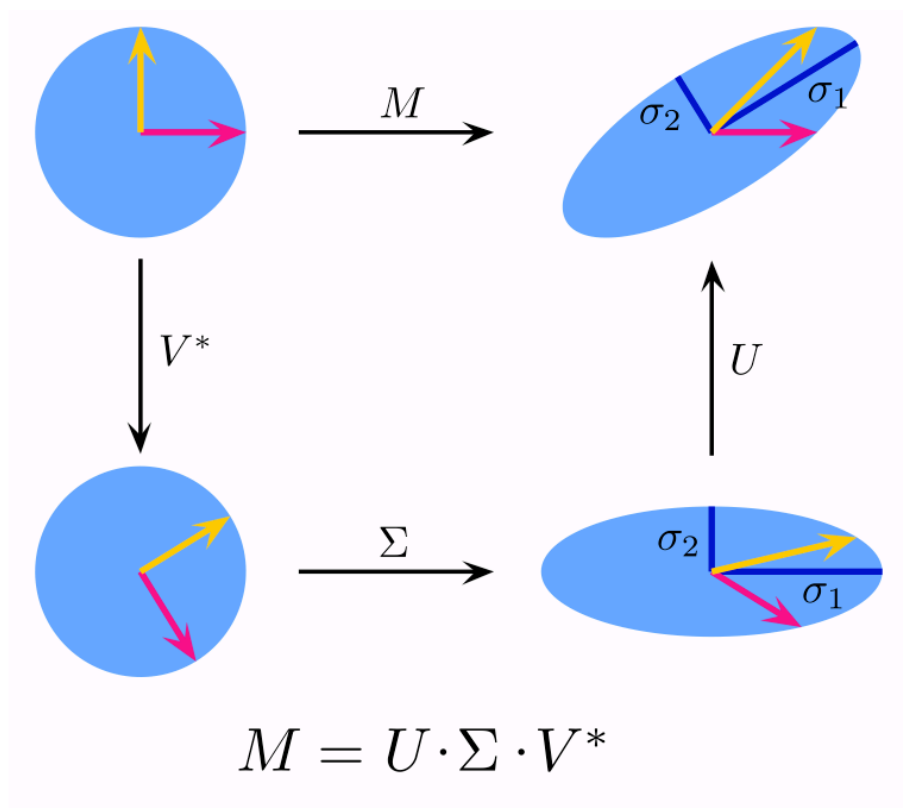
A U Λ U^*
 $A^*A = AA^*$ $U^*U = UU^* = I$

$$x_{t+1} = Ax_t \quad x_0 = c \quad x_t = A^t c$$

$$x_t = V D^t V^{-1} c \quad V D A \quad D \quad t \quad D^t$$

7.2.5

- $m \times n$ A
- $A = U \Sigma V^*$, $U^* U = V^* V = I$, Σ A ,
- U V



/

Chapter 8

{nonlinear equation}

set up

$$f : \mathbb{R}^n \rightarrow \mathbb{R}^m \vec{x}^* : f(\vec{x}^*) = \vec{0}$$

$$f : \mathbb{R} \rightarrow \mathbb{R}$$

- $x \rightarrow y, f(x) \rightarrow f(y)$
- Lipschitz $|f(x) - f(y)| \leq C|x - y|$
- $\forall x, \exists f'(x)$
- $C^k : k \quad C^\infty$

8.1 Bisection method

$$f : [a, b] \mapsto \mathbb{R} \quad f(a) < f(b) \quad u \quad f(a) < u < f(b) \quad c, a < c < b \quad f(c)=u \quad f(a) > f(b)$$

$$[a, b]$$

$$f(x) = 0 \quad (x) :$$

1. $[a, b] \quad f(a) f(b)$
2. $m = \frac{a+b}{2} \quad f(m)$
3. $f(m) = 0 \quad m$
4. $f(m) f(a) \quad [m, b] \quad , \quad [a, m].$
5. $2 \quad 3$

work

8.2 fixed point

fixed point

$$g(x) = x$$

$$x_0 x_{k+1} = g(x_k)$$

$$g(x) = f(x) + x$$

$$g(x) \quad f(x)$$

8.3 Newton's method

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

$$f'(x) \quad o(\quad)o$$

8.4 Secant method

$$f'(x)$$

$$x_{k+1} = x_k - \frac{f(x_k)(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})}$$

$$f'(x) \quad x_0, x_1, \quad \alpha = \frac{1 + \sqrt{5}}{2} \approx 1.618$$

8.5

$$\cos(x) - x^3 = 0$$

```
numpy
import numpy as np
from scipy.optimize import fsolve

def func(x):
    return np.cos(x) - x**3

result = fsolve(func, 1)
print(result)
# 0.86547403
print( func(result) )
#2.22044605e-16

x = 0.86547    ok
```


Chapter 9

{nonlinear equations}

$f : \mathbb{R}^n \rightarrow \mathbb{R}^m \vec{x}^* : f(\vec{x}^*) = \vec{0}$

n m. $f(\vec{x}) = A\vec{x} - \vec{b}$

9.1

9.1.1 Jacobian matrix

$$(Df)_{ij} = \frac{\partial f_i}{\partial x_j} = \begin{bmatrix} \frac{\partial \mathbf{f}}{\partial x_1} & \cdots & \frac{\partial \mathbf{f}}{\partial x_n} \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}$$

$f : \mathbb{R}^n \rightarrow \mathbb{R}^m \quad \mathbf{x} \quad \mathbf{x}$
f x

$f(\vec{x}) \approx f(\vec{x}_k) + Df(\vec{x}_k)(\vec{x} - \vec{x}_k)$

$\vec{x}_{k+1} = \vec{x}_k - [Df(\vec{x}_k)]^{-1}f(\vec{x}_k)$

$[Df(\vec{x}_k)]^{-1}f(\vec{x}_k) = \vec{y}_k$

$$[Df(\vec{x}_k)]\vec{y}_k = f(\vec{x}_k)$$

$$[Df(\vec{x}_k)]^{-1}, \quad \vec{y}_k.$$

$$1$$

$$\bullet$$

$$\bullet \quad Df(\vec{x}_k)$$

9.2 Broyden's method

Broyden's method

$f'(x)$:

$$f'(x_k) \approx \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}} x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

$$J \cdot (\vec{x}_k - \vec{x}_{k-1}) \approx f(\vec{x}_k) - f(\vec{x}_{k-1}) J \approx Df(\vec{x}_k)$$

$$\bullet \quad \vec{x}_k \quad J_k$$

$$\bullet \quad \vec{x}_k$$

$$\bullet \quad J_k$$

$$\text{minimize}_{J_k} \|J_k - J_{k-1}\|_{Fro}^2 \text{ such that } J \cdot (\vec{x}_k - \vec{x}_{k-1}) \approx f(\vec{x}_k) - f(\vec{x}_{k-1})$$

$$J_k = J_{k-1} + \frac{(f(\vec{x}_k) - f(\vec{x}_{k-1}) - J_{k-1}\Delta\vec{x})}{\|\vec{x}_k - \vec{x}_{k-1}\|^2} (\Delta\vec{x}^T) \vec{x}_{k+1} = \vec{x}_k - J_k^{-1} f(\vec{x}_k)$$

$$J_0 = I, \quad J_k^{-1},$$

$$J_k = J_{k-1} + \frac{(f(\vec{x}_k) - f(\vec{x}_{k-1}) - J_{k-1}\Delta\vec{x})}{\|\vec{x}_k - \vec{x}_{k-1}\|^2} (\Delta\vec{x}^T)$$

$$J_k = J_{k-1} + \vec{u}_k \vec{v}_k^T$$

Sherman-Morrison Formula:

$$(A + uv^T)^{-1} = A^{-1} - \frac{A^{-1}uv^TA^{-1}}{1 + v^TA^{-1}u}.$$

$$J_k^{-1} = J_{k-1}^{-1} - \frac{J_{k-1}^{-1}\vec{u}_k\vec{v}_k^T J_{k-1}^{-1}}{1 + \vec{v}_k^T J_{k-1}^{-1} \vec{u}_k}$$

$$J_k^{-1}.$$

9.3

scipy fsolve

$$\begin{cases} x_0 + x_1^2 = 4 \\ e^{x_0} + x_0 x_1 = 3 \end{cases}$$

```
from scipy.optimize import fsolve
import math

def equations(p):
    x0, x1 = p
    return ( x0 + x1**2 - 4, math.exp(x0) + x0 * x1 - 3 )

x0, x1 = fsolve(equations, (1, 1))

print(x0, x1)
# 0.6203445234801195 1.8383839306750887
print(equations((x0, x1)))
# (4.4508396968012676e-11, -1.0512035686360832e-11)
```


Chapter 10

{Points Concepts}

10.1 critial point

- $f : \mathbb{R} \rightarrow \mathbb{R}$ 0
- $f : \mathbb{C} \rightarrow \mathbb{C}$: 0
- $f : \mathbb{R}^n \rightarrow \mathbb{R}$: 0
- $f : \mathbb{R}^m \rightarrow \mathbb{R}^n$: Jacobian

10.2 stationary point

(stationary point) $f : \mathbb{R} \rightarrow \mathbb{R}$

$$\left. \frac{dy}{dx} \right|_p = 0$$

(stationary point) 0 x y stationary point

10.3 inflection point

inflect

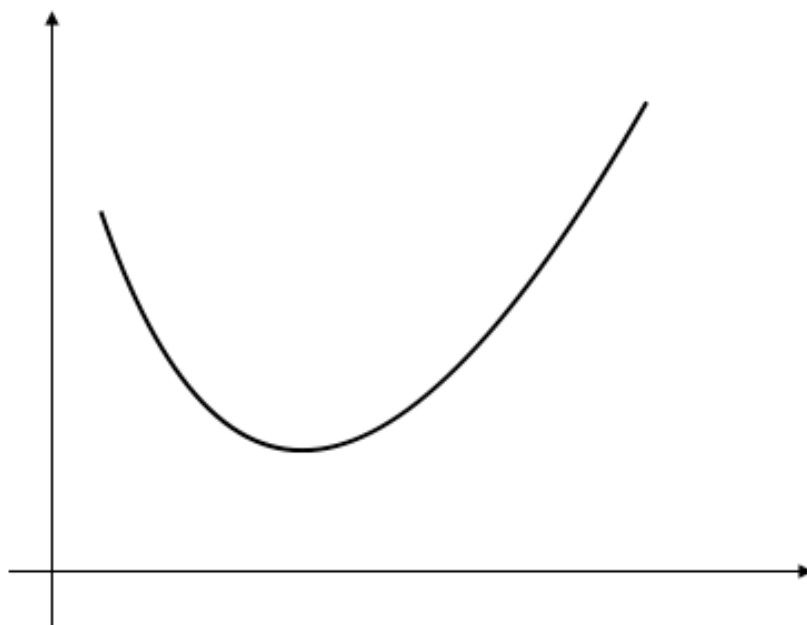
‘ ’ inflection flex

Inflection point

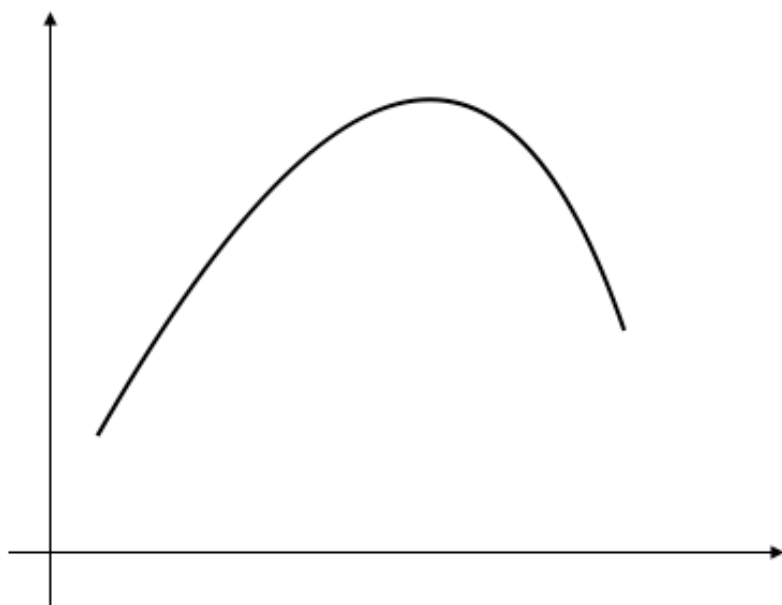
convex concave

10.3.1 convex

$$f((1 - \alpha)x + \alpha y) \leq (1 - \alpha)f(x) + \alpha f(y)$$

**10.3.2 concave**

$$f((1 - \alpha)x + \alpha y) \geq (1 - \alpha)f(x) + \alpha f(y)$$



convex v , concave/cave

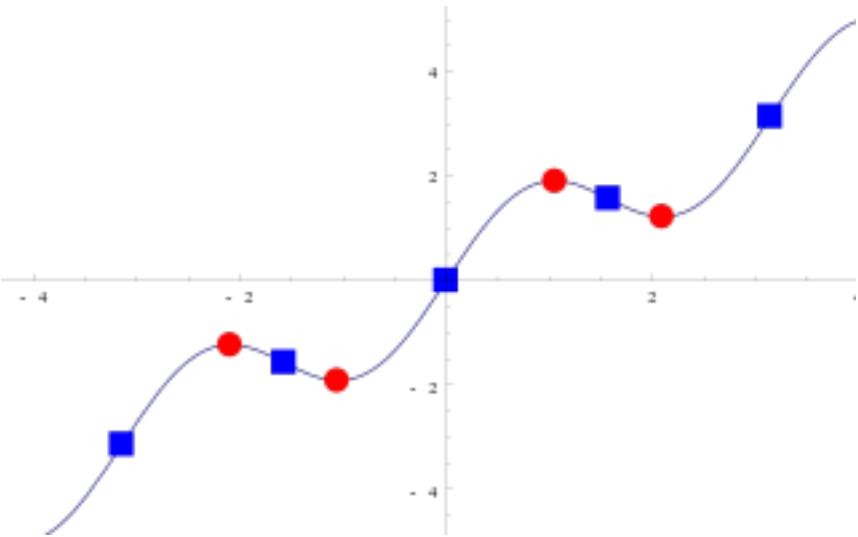
con  ex

conCAVE:

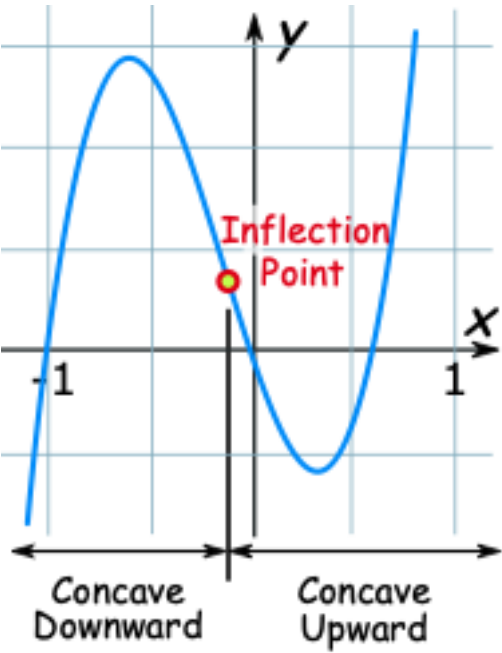


convex concave

/ stationary points/critical points, inflection points.



inflection point

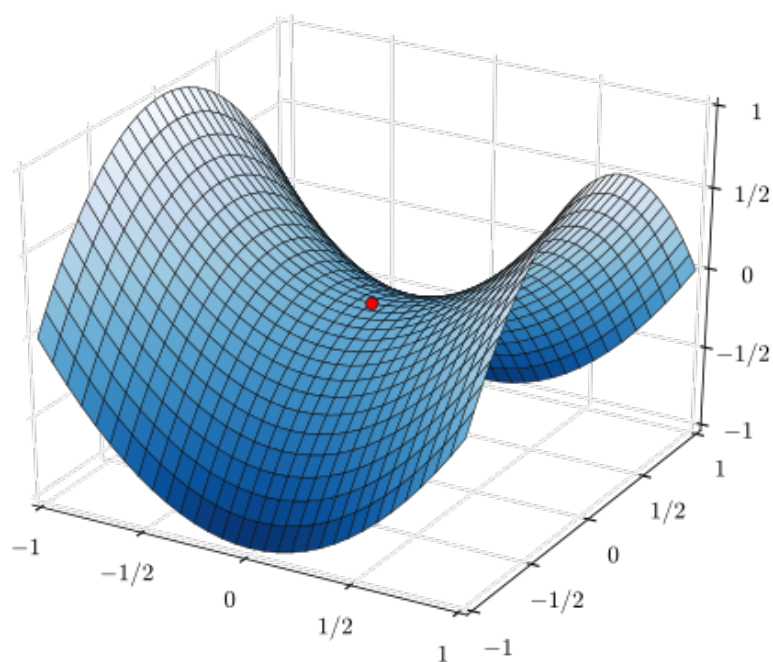


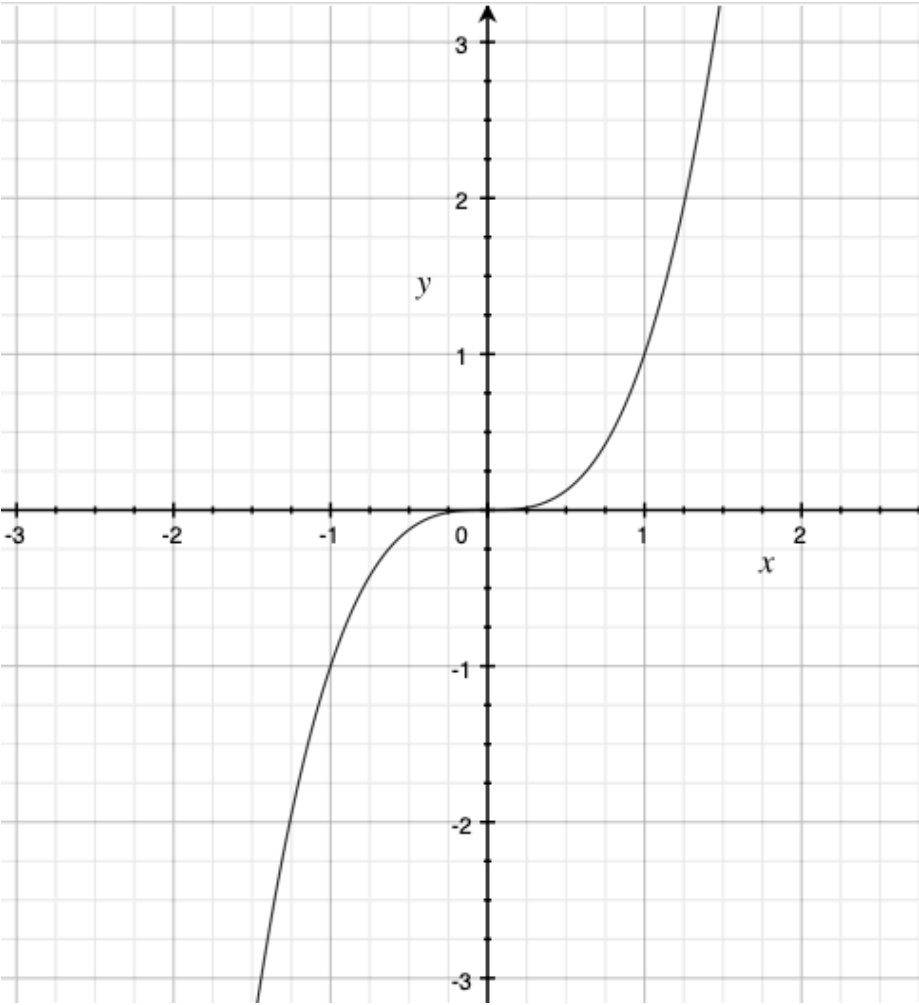
C^0 C^∞ convex/concave

10.4 saddle point

saddle point minmax point.

$$f(x, y) = x^2 - y^2, \quad (0, 0)$$





$y = x^3$ (0, 0)

10.5 vertex (curve)

$$y = ax^2 + bx + c$$

$$k(x) = \frac{2a}{(1 + (2ax + b)^2)^{\frac{3}{2}}}.$$

$$x = -b/2a$$

$$0$$

Chapter 11

Jacobian Hessian {Gradient Related Concepts}

- :
- $f : \mathbb{R} \rightarrow \mathbb{R}$
 - $f : \mathbb{R}^n \rightarrow \mathbb{R}$
 - $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$
- f

11.1

$$f : \mathbb{R} \rightarrow \mathbb{R}$$

$$f(x) \approx f(x_0) + f'(x_0)(x - x_0)$$

11.2

$$f : \mathbb{R}^n \rightarrow \mathbb{R}$$

$$\nabla f = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{pmatrix}$$

$n \times 1 \quad \vec{x} \quad \nabla_x:$

$$\nabla_x \stackrel{\text{def}}{=} \left[\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \dots, \frac{\partial}{\partial x_n} \right]^T = \frac{\partial}{\partial \mathbf{x}}$$

$$f(\vec{x}) \approx f(\vec{x}_0) + \nabla f(\vec{x}_0) \cdot (\vec{x} - \vec{x}_0)$$

11.3 Jacobian

Jacobian

$f : \mathbb{R}^n \rightarrow \mathbb{R}^m$

$f : \mathbb{R}^n \rightarrow \mathbb{R}^m \quad \mathbf{x} \quad \mathbf{x}$

$\mathbf{f} \quad \mathbf{x}$

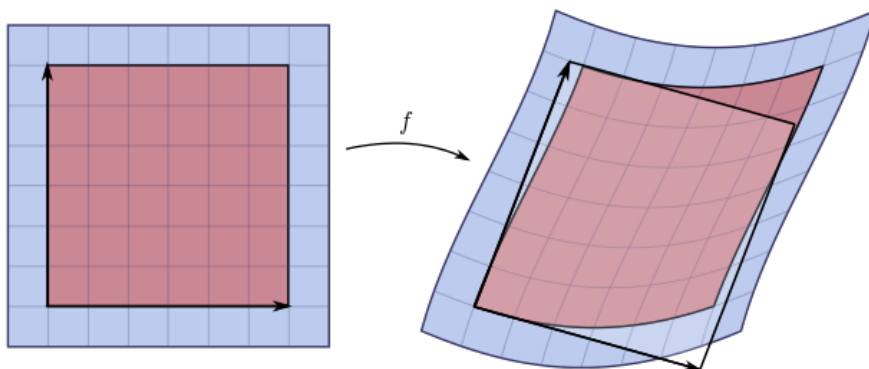
$$\mathbf{J} = \begin{bmatrix} \frac{\partial \mathbf{f}}{\partial x_1} & \cdots & \frac{\partial \mathbf{f}}{\partial x_n} \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}$$

$$\mathbf{J}_{ij} = \frac{\partial f_i}{\partial x_j}.$$

$$Df \text{ D}\mathbf{f} \mathbf{J}_{\mathbf{f}}(x_1, \dots, x_n) \quad \frac{\partial (f_1, \dots, f_m)}{\partial (x_1, \dots, x_n)}.$$

$$f(\vec{x}) \approx f(\vec{x}_k) + J(\vec{x}_k)(\vec{x} - \vec{x}_k)$$

$m = n$ Jacobian



Jacobian Matrix

11.4 Hessian

$$f : \mathbb{R}^n \rightarrow \mathbb{R}$$

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

$n \times n$

$$\mathbf{H}_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}$$

- $f : \mathbb{R} \rightarrow \mathbb{R}$

$$f(x) \approx f(x_0) + f'(x_0)(x - x_0) + \frac{1}{2}f''(x_0)(x - x_0)^2$$

- $f : \mathbb{R}^n \rightarrow \mathbb{R}$:

$$f(x_1, x_2) = f(x_{10}, x_{20}) + f_{x_1}(x_{10}, x_{20})\Delta x_1 + f_{x_2}(x_{10}, x_{20})\Delta x_2 + \frac{1}{2}[f_{x_1 x_1}(x_{10}, x_{20})\Delta x_1^2 + 2f_{x_1 x_2}(x_{10}, x_{20})\Delta x_1 \Delta x_2 + f_{x_2 x_2}(x_{10}, x_{20})\Delta x_2^2]$$

$$\Delta x_1 = x_1 - x_{10} \quad \Delta x_2 = x_2 - x_{20} \quad f_{x_1} = \frac{\partial f}{\partial x_1} \quad f_{x_2} = \frac{\partial f}{\partial x_2} \quad f_{x_1 x_1} = \frac{\partial^2 f}{\partial x_1^2} \quad f_{x_2 x_2} = \frac{\partial^2 f}{\partial x_2^2} \quad f_{x_1 x_2} = \frac{\partial^2 f}{\partial x_1 \partial x_2} =$$

$$f(\vec{x}) \approx f(\vec{x}_0) + \nabla f(\vec{x}_0) \cdot (\vec{x} - \vec{x}_0) + \frac{1}{2}(\vec{x} - \vec{x}_0)^T H(\vec{x}_0)(\vec{x} - \vec{x}_0)$$

$$H(x_0) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} \end{bmatrix}_{x_0}$$

Hessian

$$f : \mathbb{R} \rightarrow \mathbb{R}, \quad x = x_0 \quad f'(x_0) = 0, f''(x_0) \neq 0,$$

- $f''(x) < 0$,
- $f''(x) > 0$,
- $f''(x) = 0$,
- $f''(x)$

$$f : \mathbb{R}^n \rightarrow \mathbb{R} \quad \vec{x}_0 \quad \vec{0} \quad H(\vec{x}_0)$$

- H ,
- H ,
- H
- H

- Cholesky
-

Chapter 12

{Optimization without constraintss}

12.1

		$E(\vec{x}) = \ A\vec{x} - \vec{b} \ ^2$	
\vec{b}	\vec{a}	$E(c) = \ c\vec{a} - \vec{b} \ ^2$	
		$E(\vec{x}) = \vec{x}^T A\vec{x}$	$\ \vec{x} \ = 1$
Pseudoinverse		$E(\vec{x}) = \ \vec{x} \ ^2$	$A^T A\vec{x} = A^T \vec{b}$
		$E(C) = \ $	$C^T C = I_{d \times d}$
		$X - CC^T X \ _{Fro}$	
Broyden step		$E(J_k) = \ $	$J_k \cdot (\vec{x}_k - \vec{x}_{k-1}) =$
		$J_k - J_{k-1} \ ^2_{Fro}$	$f(\vec{x}_k) - f(\vec{x}_{k-1})$

set up

$$min_{\vec{x}} f(\vec{x})$$

•

$$E(a,c) = \sum_i (y_i - ce^{ax_i})^2$$

•

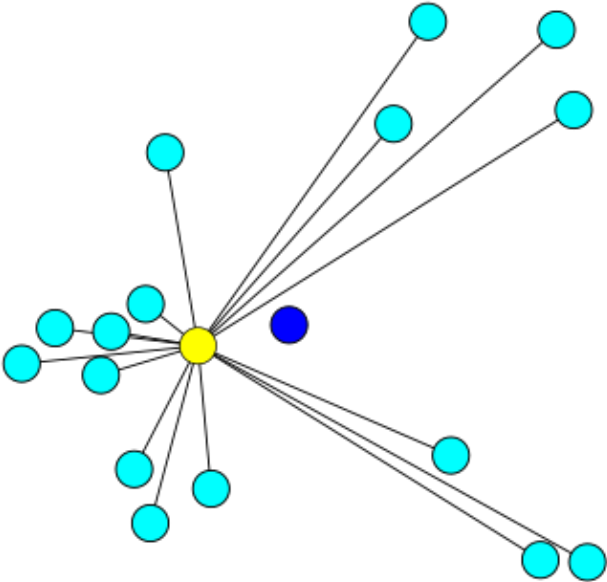
$$g(h;\mu,\sigma)=\frac{1}{\sigma\sqrt{2\pi}}e^{-(h-\mu)^2/2\sigma^2}$$

$$h_1,\cdots,h_n$$

$$P(h_1,\cdots,h_n;\mu,\sigma)=\prod_i g(h_i,\mu,\sigma)$$

$$\mu,\sigma \qquad \qquad \qquad /NLP$$

- geometric median



$$E(\vec{x}) = \sum_i \| \vec{x} - \vec{x}_i \|_2$$

ℓ_2 norm,

12.2

$$\vec{x}^* \in \mathbb{R}^n f : \mathbb{R}^n \rightarrow \mathbb{R} \forall \vec{x} \in \mathbb{R}^n f(\vec{x})^* \leq f(\vec{x})$$

$$\vec{x}^* \in \mathbb{R}^n f : \mathbb{R}^n \rightarrow \mathbb{R} \forall \| \vec{x} - \vec{x}^* \| < \varepsilon, f(\vec{x})^* \leq f(\vec{x})$$

12.3

12.3.1

$$f : \mathbb{R} \rightarrow \mathbb{R}$$

$$0$$

$$f'(x) = 0$$

$$f'(x) = 0$$

$$f(x) =$$

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}.$$

12.3.2 Golden-section search

Unimodal function

unimodular

$$f : [a, b] \rightarrow \mathbb{R} \quad x^* \in [a, b] \quad \text{f} \quad x \in [a, x^*] \quad x \in [x^*, b]$$

$$a < x_0 < x_1 < b,$$

$$\begin{aligned} & \bullet f(x_0) \leq f(x_1), \quad x \in [x_1, b], \quad f(x) \geq f(x_0), \quad x^* \in [a, x_1], \quad [x_1, b] \\ & \bullet f(x_0) \geq f(x_1), \quad x \in [a, x_0], \quad f(x) \geq f(x_1), \quad x^* \in [x_0, b], \quad [a, x_0] \end{aligned}$$

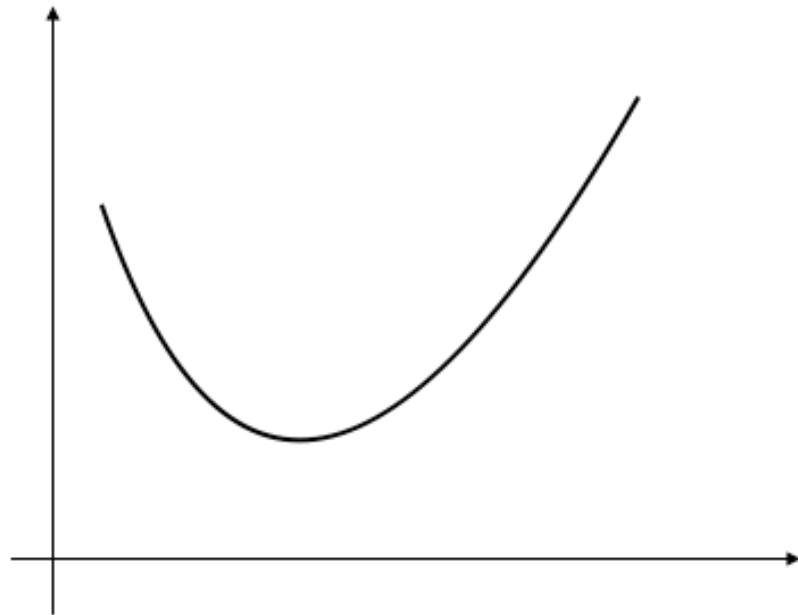


Figure 12.1: unimodular.png

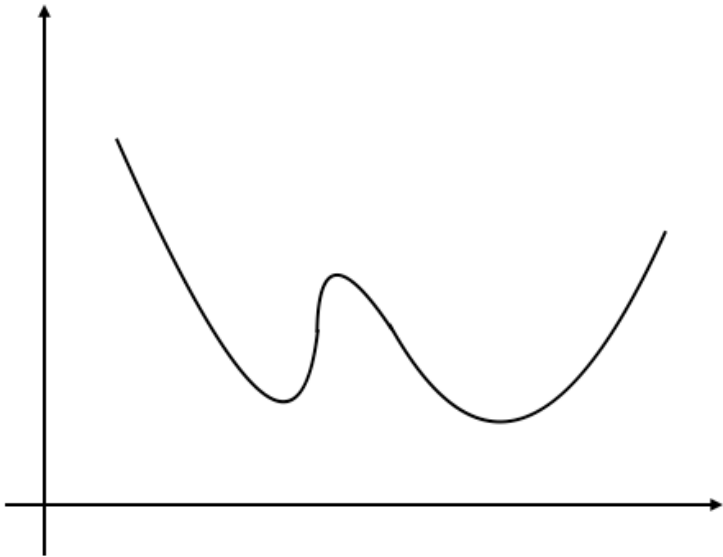


Figure 12.2: Bimodal.png

$f(x_0) \leq f(x_1), \qquad \qquad \qquad a = 0, \, b = 1,$

$x_0 = \alpha, x_1 = 1 - \alpha, \alpha \in (0, \frac{1}{2})$

• $f(x_0) \leq f(x_1)$

$[a, x_1] = [0, 1 - \alpha]$

$\alpha(1 - \alpha), (1 - \alpha)^2$

$x_0 = \alpha = (1 - \alpha)^2$

$\alpha^2 - 3\alpha + 1 = 0 \alpha = \frac{1}{2}(3 - \sqrt{5}) 1 - \alpha = \frac{1}{2}(\sqrt{5} - 1)$

$1 - \alpha = \tau$

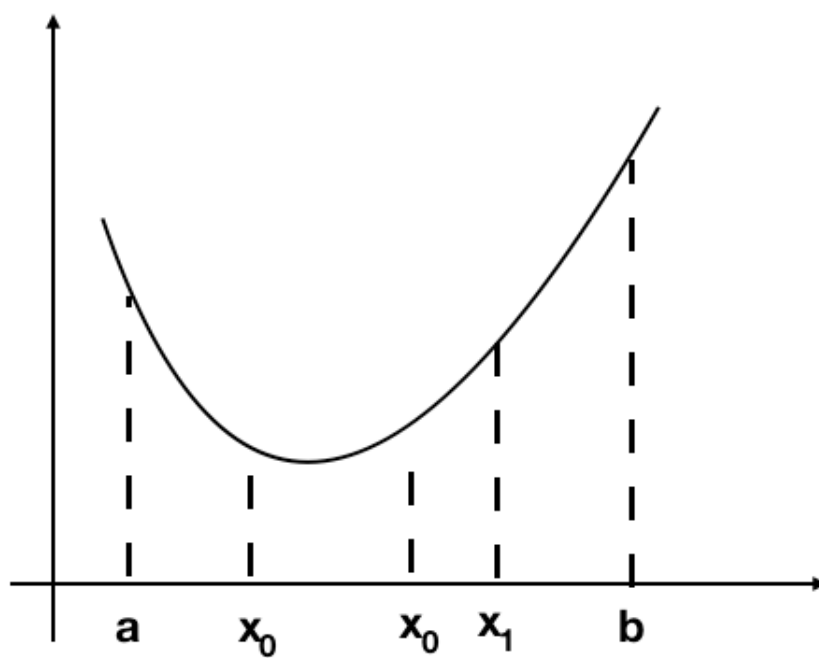


Figure 12.3: unimodal_01.png

$$\bullet \quad f(x_0) \geq f(x_1)$$

$$[x_0, 1] = [\alpha, 1]$$

$$\alpha + \alpha(1 - \alpha), \alpha + (1 - \alpha)^2$$

$$x_0 = 1 - \alpha = \alpha + \alpha(1 - \alpha)$$

1. $a, b \in [a, b]$ unimodular
2. $x_0 = a + (1 - \tau)(b - a), x_1 = a + \tau(b - a), f_0 = f(x_0), f_1 = f(x_1)$
3. $b - a$
 - $f_0 \geq f_1, [a, x_0]$
 - $a \leftarrow x_0$
 - $x_0 \leftarrow x_1, f_0 \leftarrow f_1$
 - $x_1 \leftarrow a + \tau(b - a), f_1 \leftarrow f(x_1)$
 - $f_1 > f_0, [x_1, b]$
 - $b \leftarrow x_1$
 - $x_1 \leftarrow x_0, f_1 \leftarrow f_0$
 - $x_0 \leftarrow a + (1 - \tau)(b - a), f_0 \leftarrow f(x_0)$

12.4

$$f : \mathbb{R}^n \rightarrow \mathbb{R},$$

12.4.1

$$\begin{aligned} F(\vec{x}) &= F(\vec{a}) - \nabla F(\vec{a}) \cdot (\vec{x} - \vec{a}) \\ \vec{b} = \vec{a} - \gamma \nabla F(\vec{a}) &\quad \gamma > 0 \quad F(\vec{a}) \geq F(\vec{b}) \end{aligned}$$

- x_0
- $g_k(t) = f(\vec{x}_k - t \nabla f(\vec{x}_k))$
- $t^* \geq 0$
- $\vec{x}_{k+1} = \vec{x}_k - t^* \nabla f(\vec{x}_k)$

12.4.2

$$\vec{x}_{k+1} = \vec{x}_k - [H_f(\vec{x}_k)]^{-1} \nabla f(\vec{x}_k)$$

$\nabla f(\vec{x})$ Hessian +n Quasi-Newton method

12.4.3 BFGS

BFGS Broyden–Fletcher–Goldfarb–Shanno algorithm Shanno
Shanno Broyden’s method Hessian :

$$\vec{x}_{k+1} = \vec{x}_k - \alpha_k B_k^{-1} \nabla f(\vec{x}_k) B_k \approx H_f(\vec{x}_k)$$

$$B_{k+1}(\vec{x}_{k+1} - \vec{x}_k) = \nabla f(\vec{x}_{k+1}) - \nabla f(\vec{x}_k)$$

B

•
•

$$\min_{B_{k+1}} \| B_{k+1} - B_k \| \text{ s.t. } B_{k+1}^T = B_{k+1} B_{k+1} (\vec{x}_{k+1} - \vec{x}_k) = \nabla f(\vec{x}_{k+1}) - \nabla f(\vec{x}_k)$$

$$\$ B_{\{k+1\}} - B_k \$ \quad \$ B_{\{k+1\}^{\wedge\{-1\}}} - B_k^{\wedge\{-1\}} \$$$

$$\min_{H_{k+1}} \| H_{k+1} - H_k \| \text{ s.t. } H_{k+1}^T = H_{k+1} \vec{x}_{k+1} - \vec{x}_k = H_{k+1} (\nabla f(\vec{x}_{k+1}) - \nabla f(\vec{x}_k))$$

Broyden–Fletcher–Goldfarb–Shanno algorithm

Chapter 13

KKT {Lagrange multiplier to KKT condition}

/ level set/contour

min/max $f(x, y)$ subject to $g(x, y) = c$

$$\mathcal{L}(x, y, \lambda) = f(x, y) - \lambda \cdot (g(x, y) - c) \quad \mathcal{L}(x, y, \lambda) = f(x, y) + \lambda \cdot (g(x, y) - c)$$

min/max $f(x, y)$ subject to $g(x, y) = 0$

$$\mathcal{L}(x, y, \lambda) = f(x, y) - \lambda \cdot g(x, y) \quad \mathcal{L}(x, y, \lambda) = f(x, y) + \lambda \cdot g(x, y)$$

$$g(\mathbf{x}) = \mathbf{0}$$

13.1

•

$$f(x,y)=x+y\text{s.t. }x^2+y^2=1$$

$$f(x,y) \geq g(x,y), \qquad f(x,y) = g(x,y) \qquad \qquad \qquad :$$

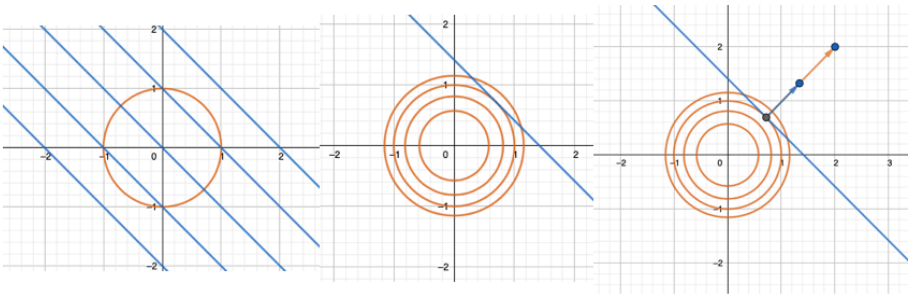


Figure 13.1: level_set_0.png

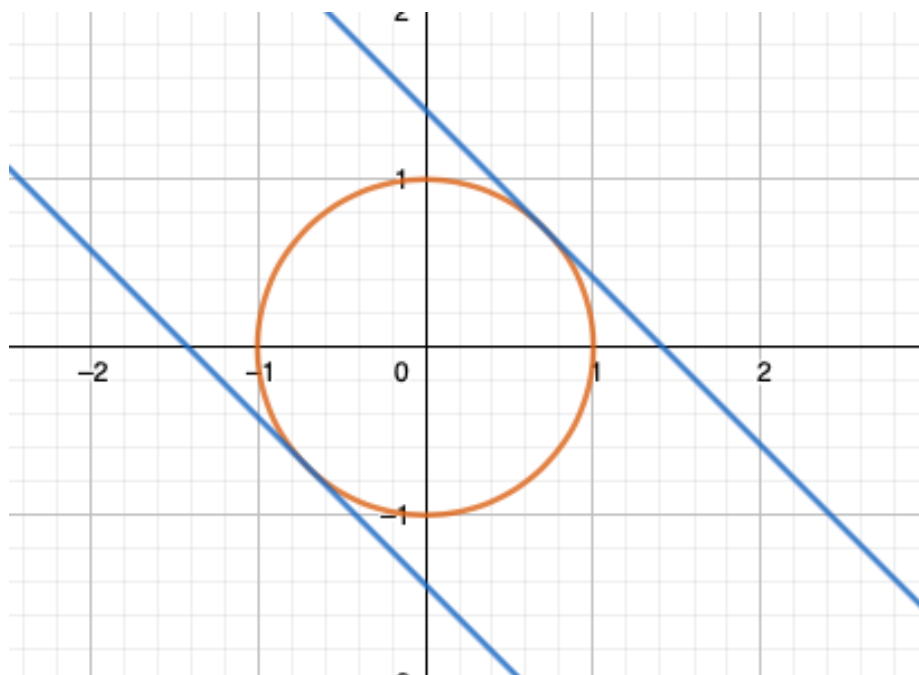
$$\nabla f = \lambda \nabla g$$

$$\text{pack} \quad :$$

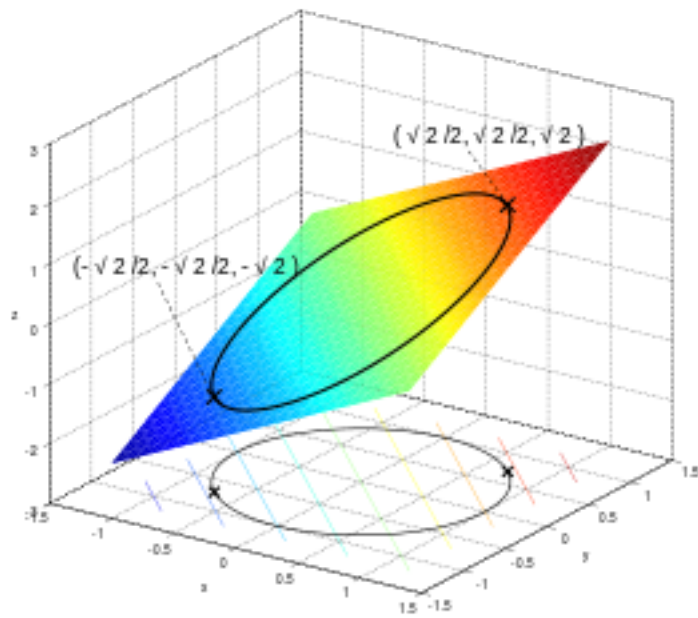
$$\begin{aligned} \mathcal{L}(x,y,\lambda) &= f(x,y) - \lambda \cdot g(x,y) \\ &= x + y - \lambda(x^2 + y^2 - 1). \end{aligned}$$

$$\frac{\partial \mathcal{L}}{\partial x} = 1 - 2\lambda x = 0 \frac{\partial \mathcal{L}}{\partial x} = 1 - 2\lambda y = 0 \frac{\partial \mathcal{L}}{\partial \lambda} = x^2 + y^2 - 1 = 0$$

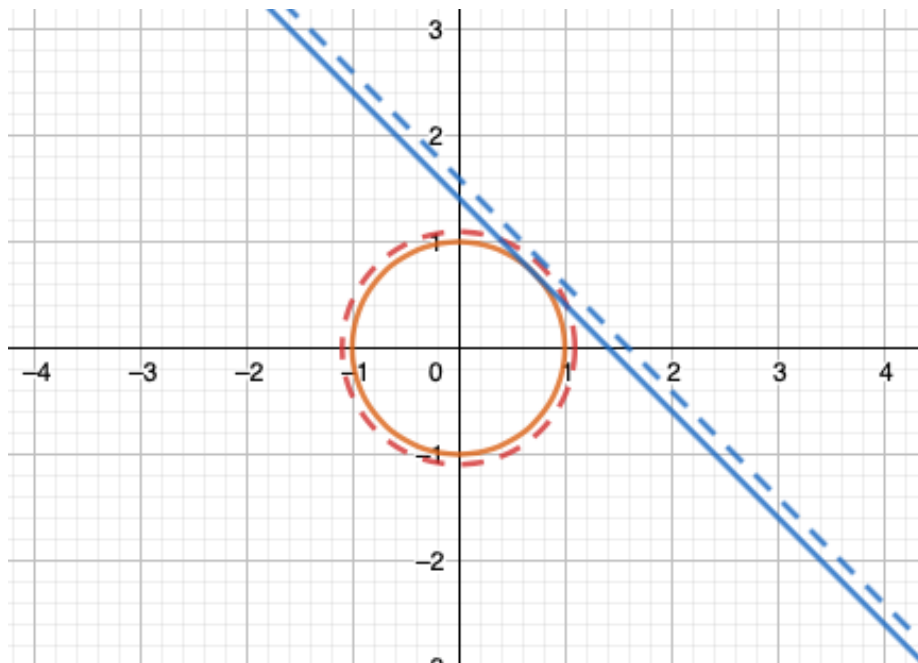
$$\lambda = \pm \frac{1}{\sqrt{2}}$$



$$f\left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right) = \sqrt{2}f\left(-\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right) = -\sqrt{2}$$

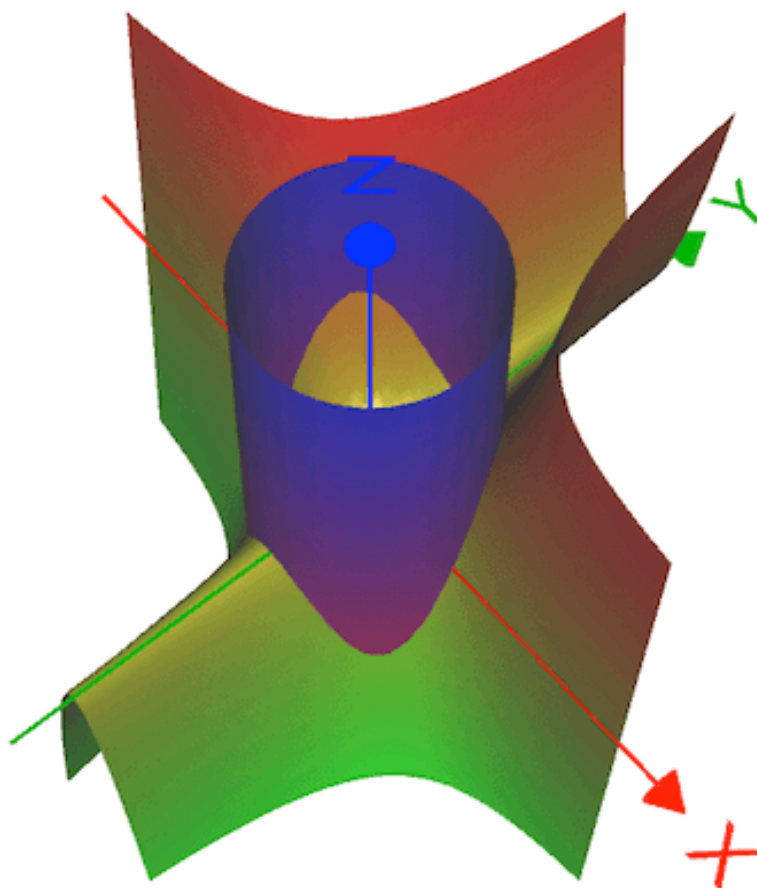


$$\lambda = \frac{1}{\sqrt{2}} \quad f(x,y) = \dots$$



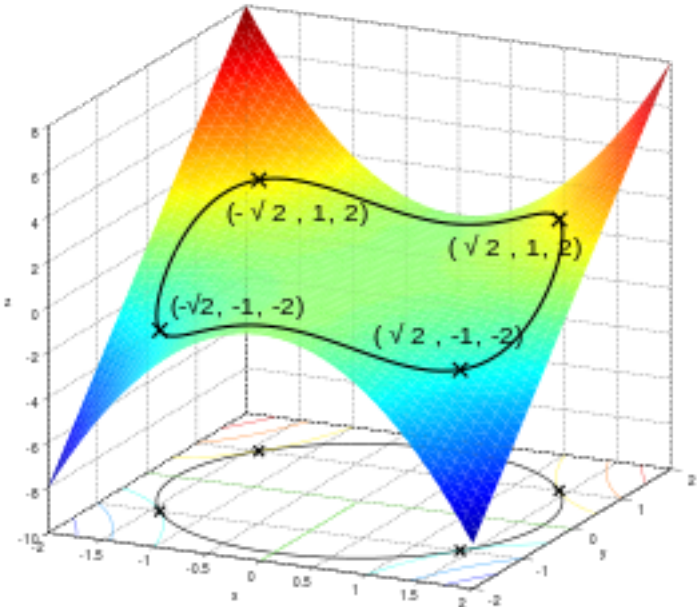
•

$$f(x, y) = x^2y \text{ s.t. } x^2 + y^2 = 3$$



$$\begin{aligned}\mathcal{L}(x, y, \lambda) &= f(x, y) - \lambda \cdot g(x, y) \\ &= x^2y - \lambda(x^2 + y^2 - 3).\end{aligned}$$

$$\frac{\partial \mathcal{L}}{\partial x} = 2xy - 2\lambda x = 0 \quad \frac{\partial \mathcal{L}}{\partial x} = x^2 - 2\lambda y = 0 \quad \frac{\partial \mathcal{L}}{\partial \lambda} = x^2 + y^2 - 3 = 0$$



•

$\{p_1, p_2, \dots, p_n\}, \quad :$

$f(p_1, p_2, \dots, p_n) = - \sum_{j=1}^n p_j \log_2 p_j \text{s.t. } g(p_1, p_2, \dots, p_n) = \sum_{j=1}^n p_j = 1$

$$\begin{aligned} \mathcal{L}(x, y, \lambda) &= f(x, y) - \lambda \cdot g(x, y) \\ &= - \sum_{j=1}^n p_j \log_2 p_j - \lambda (\sum_{j=1}^n p_j - 1) \end{aligned}$$

p_k

$$- \left(\frac{1}{\ln 2} + \log_2 p_k^* \right) - \lambda = 0.$$

$$p_k^* = \frac{1}{n}.$$

13.2

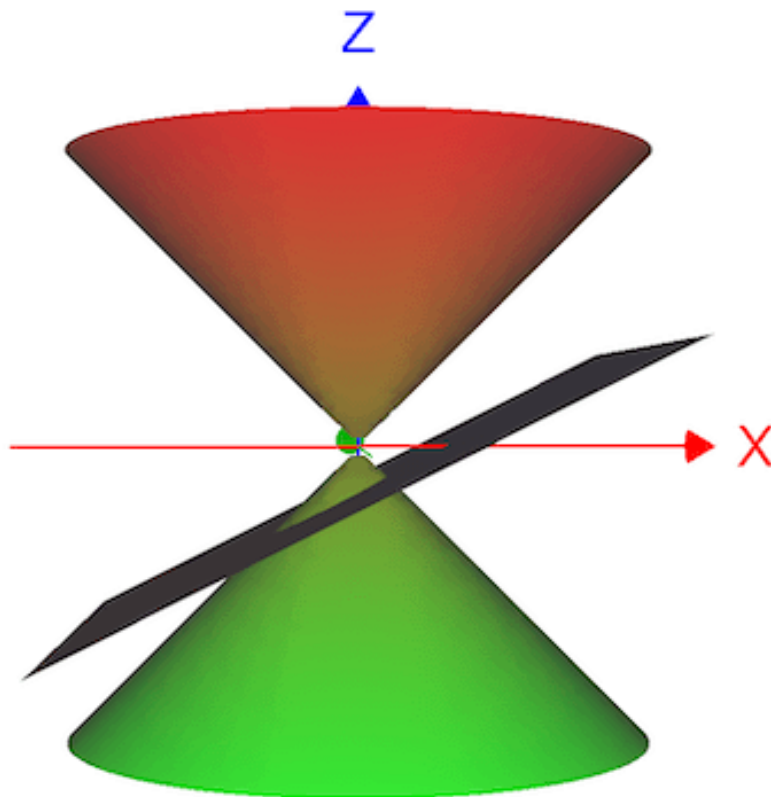
$$f(x, y) \text{ s.t. } g_i(x) = 0, i = 1, \dots, M$$

$$\mathcal{L}(x_1, \dots, x_n, \lambda_1, \dots, \lambda_M) = f(x_1, \dots, x_n) - \sum_{k=1}^M \lambda_k g_k(x_1, \dots, x_n)$$

$$\nabla_{x_1, \dots, x_n, \lambda_1, \dots, \lambda_M} \mathcal{L}(x_1, \dots, x_n, \lambda_1, \dots, \lambda_M) = 0 \iff \begin{cases} \nabla f(\mathbf{x}) - \sum_{k=1}^M \lambda_k \nabla g_k(\mathbf{x}) = 0 \\ g_1(\mathbf{x}) = \dots = g_M(\mathbf{x}) = 0 \end{cases}$$

13.2.1

$$z^2 = x^2 + y^2 \quad x - 2z = 3$$



$$d = \sqrt{x^2 + y^2 + z^2}, \quad f(x, y, z) = d^2$$

$$f(x, y, z) = x^2 + y^2 + z^2 x^2 + y^2 = z^2 x - 2z = 3$$

$$\begin{aligned} \mathcal{L}(x, y, z, \lambda, \mu) &= f(x, y, z) - \lambda \cdot g(x, y, z) - \mu \cdot h(x, y, z) \\ &= x^2 + y^2 + z^2 - \lambda(x^2 + y^2 - z^2) - \mu(x - 2z - 3) \end{aligned}$$

$$\frac{\partial \mathcal{L}}{\partial x} = 2x - 2\lambda x - \mu = 0 \quad \frac{\partial \mathcal{L}}{\partial y} = 2y - 2\lambda y = 0 \quad \frac{\partial \mathcal{L}}{\partial y} = 2z + 2\lambda z + 2\mu = 0 \quad \frac{\partial \mathcal{L}}{\partial \lambda} = x^2 + y^2 - z^2 = 0 \quad \frac{\partial \mathcal{L}}{\partial \mu} = x - 2z - 3 = 0$$

:

$$(-3, 0, 3) \rightarrow d_{max} = 3\sqrt{2}(1, 0, -1) \rightarrow d_{min} = \sqrt{2}$$

13.3 KKT

13.3.1

KKT Karush–Kuhn–Tucker conditions

$$f(x, y) \text{ s.t } g(\mathbf{x}) \leq 0$$

primal feasibility , feasible region

$$K = \{\mathbf{x} \in \mathbb{R}^n | g(\mathbf{x}) \leq 0\}$$

- $g(\mathbf{x}) = 0$, (boundary solution) (active)
- $g(\mathbf{x}) < 0$, (interior solution) (inactive)

$$\mathcal{L}(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda \cdot g(\mathbf{x})$$

•

$$\nabla f = \mathbf{0}, \lambda = 0$$

•

$$\nabla f(\mathbf{x}) + \sum_{i=1}^m \mu_i \nabla g_i(\mathbf{x}) + \sum_{j=1}^{\ell} \lambda_j \nabla h_j(\mathbf{x}) = \mathbf{0}$$

$$g_i(\mathbf{x}) \leq 0, \mu_i \geq 0, \mu_i g_i(\mathbf{x}) = 0 \quad (i = 1, \dots, m)$$

$$\lambda_j \geq 0$$

$\lambda_j \geq 0$ dual feasibility.

$$\lambda_j h_j(\mathbf{x}) = 0$$

$\lambda_j h_j(\mathbf{x}) = 0$ complementary slackness.

13.4 KKT

Optimize

$$f(\mathbf{x})$$

subject to

$$g_i(\mathbf{x}) \leq 0, h_i(\mathbf{x}) = 0.$$

$$g_i \quad (i = 1, \dots, m) \quad h_i \quad (i = 1, \dots, \ell)$$

- Stationarity

$$\text{For maximizing : } \nabla f(x^*) - \sum_{i=1}^m \mu_i \nabla g_i(x^*) - \sum_{j=1}^{\ell} \lambda_j \nabla h_j(x^*) = \mathbf{0},$$

$$\text{For minimizing : } \nabla f(x^*) + \sum_{i=1}^m \mu_i \nabla g_i(x^*) + \sum_{j=1}^{\ell} \lambda_j \nabla h_j(x^*) = \mathbf{0},$$

- Primal feasibility

$$g_i(x^*) \leq 0, \text{ for } i = 1, \dots, m, h_j(x^*) = 0, \text{ for } j = 1, \dots, \ell$$

- Dual feasibility

$$\mu_i \geq 0, \text{ for } i = 1, \dots, m$$

- Complementary slackness

$$\mu_i g_i(x^*) = 0, \text{ for } i = 1, \dots, m.$$

13.5

$$\text{minimize } x_1^2 + x_2^2 - 4x_1 - 4x_2 \text{ s.t. } x_1^2 \leq x_2, x_1 + x_2 \leq 2$$

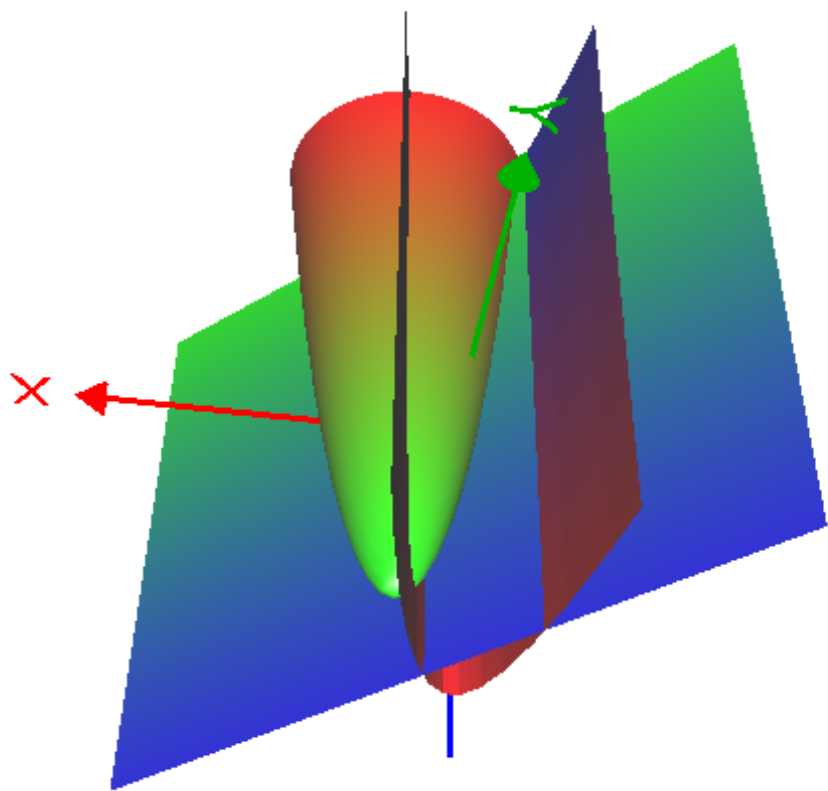
$$L(x_1, x_2, \mu_1, \mu_2) = x_1^2 + x_2^2 - 4x_1 - 4x_2 + \mu_1(x_1^2 - x_2) + \mu_2(x_1 + x_2 - 2)$$

$$2x_1 + 2\mu_1 x_1 + \mu_2 - 4 = 0, 2x_2 - \mu_1 + \mu_2 - 4 = 0, \mu_1(x_1^2 - x_2) = 0, \mu_2(x_1 + x_2 - 2) = 0, \mu_1, \mu_2 \geq 0$$

- $\mu_1 = 0, x_1 + x_2 - 2 = 0 \rightarrow x_2 = 1, x_1 = 1, \mu_2 = 2$
- $x_1^2 - x_2 = 0, x_1^2 = x_2 \rightarrow x_1 = -2, x_2 = 4, \mu_1 = 4, \mu_2 = 0$
- $\mu_1 = \mu_2 = 0 \rightarrow x_1 = 2, x_2 = 2, x_1^2 = x_2$
- $x_1^2 - x_2 = 0, x_1 + x_2 - 2 = 0 \rightarrow x_1 = 1, x_2 = -2 \dots$

$$f(1, 1) = -6$$

$$f(x_1, x_2) = x_1^2 + x_2^2 - 4x_1 - 4x_2 = (x_1 - 2)^2 + (x_2 - 2)^2 - 8 \quad x_1, x_2$$



(2,2) (1, 1) (2,2),

Chapter 14

{Conjugate gradient}

$Ax = b$, Jacobi Method Gauss–Seidel method
A $(A^T = A), \quad \forall \vec{x} \neq 0, \vec{x}^T A \vec{x} > 0$

$$Ax = b$$

14.1 Gradient descent

Gradient descent Machine Learning

‘ ‘ , $f : \mathbb{R}^n \rightarrow \mathbb{R}$
P P

$$f(\vec{x}) = \frac{1}{2} \vec{x}^T A \vec{x} - \vec{b}^T \vec{x} + c$$

$$f(\vec{x})$$

$$\nabla f(\vec{x}) = A\vec{x} - \vec{b}$$

- $\vec{d}_k = -\nabla f(\vec{x}_{k-1}) = \vec{b} - A\vec{x}_{k-1}$
- $\vec{x}_k = \vec{x}_{k-1} + \alpha_k \vec{d}_k, \quad \alpha_k \quad f(\vec{x}_k) < f(\vec{x}_{k-1})$

$$\alpha_k:$$

$$\begin{aligned} g(\alpha) &= f(\vec{x} + \alpha \vec{d}) \\ &= \frac{1}{2}(\vec{x} + \alpha \vec{d})^T A(\vec{x} + \alpha \vec{d}) - \vec{b}^T(\vec{x} + \alpha \vec{d}) + c \\ &= \frac{1}{2}(\vec{x}^T A \vec{x} + 2\alpha \vec{x}^T A \vec{d} + \alpha^2 \vec{d}^T A \vec{d}) - \vec{b}^T \vec{x} - \alpha \vec{b}^T \vec{d} + c \\ &= \frac{1}{2}\alpha^2 \vec{d}^T A \vec{d} + \alpha(\vec{x}^T A \vec{d} - \vec{b}^T \vec{d}) + \text{const} \end{aligned}$$

$$\alpha$$

$$\begin{aligned} \frac{dg(\alpha)}{d\alpha} &= \alpha \vec{d}^T A \vec{d} + (\vec{x}^T A \vec{d} - \vec{b}^T \vec{d}) \\ &= \alpha \vec{d}^T A \vec{d} + \vec{d}^T A \vec{x} - \vec{d}^T \vec{b} \\ &= \alpha \vec{d}^T A \vec{d} + \vec{d}^T (A \vec{x} - \vec{b}) \end{aligned}$$

$$0$$

$$\alpha = \frac{\vec{d}^T (\vec{b} - A \vec{x})}{\vec{d}^T A \vec{d}}$$

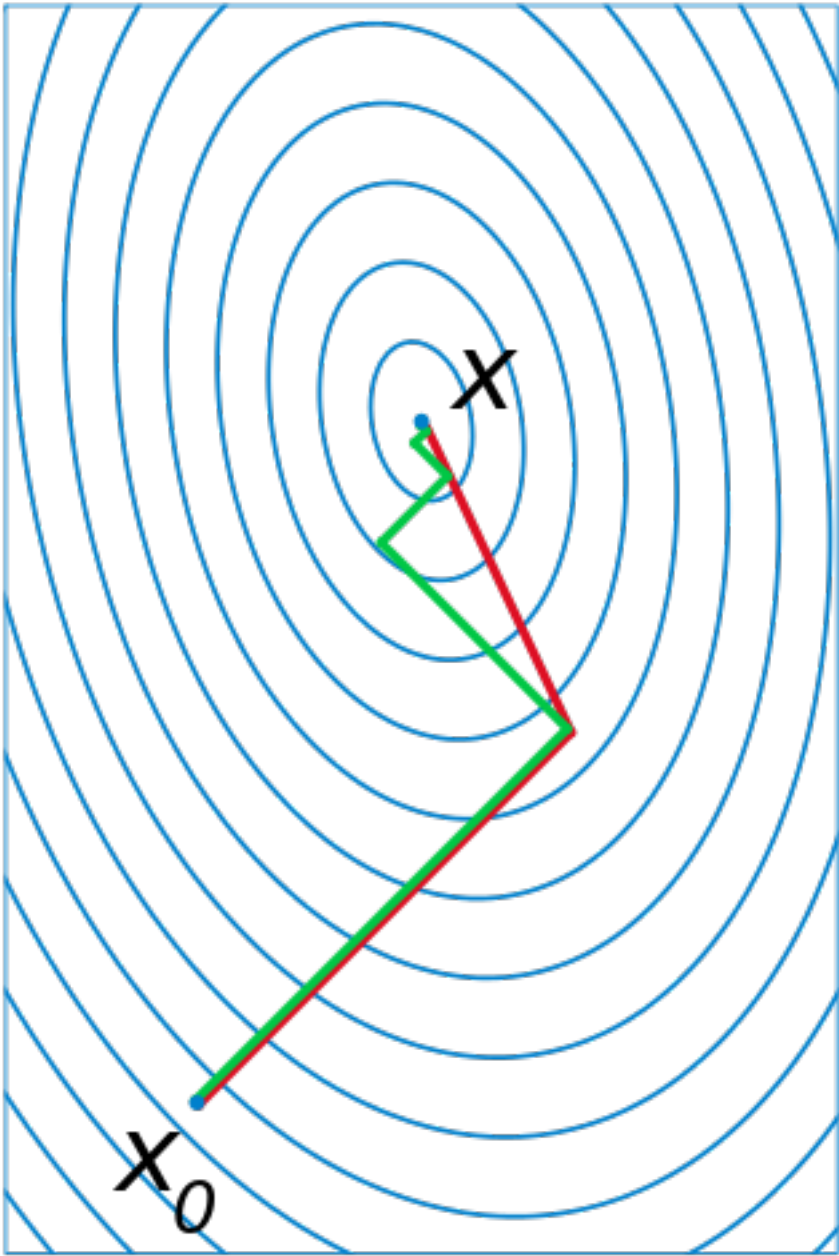
$$\vec{d}_k = \vec{b} - A \vec{x}_{k-1}$$

$$\alpha_k = \frac{\vec{d}_k^T \vec{d}_k}{\vec{d}_k^T A \vec{d}_k}$$

$$\vec{d}_k = \vec{b} - A \vec{x}_{k-1} \alpha_k = \frac{\vec{d}_k^T \vec{d}_k}{\vec{d}_k^T A \vec{d}_k} \vec{x}_k = \vec{x}_{k-1} + \alpha_k \vec{d}_k$$

$$AA^T$$

14.2 Conjugate gradient



/

z

n = 2,

call API

‘ (conjugate)’ \vec{u} \vec{v} \mathbf{A}

$$\vec{u}^T \mathbf{A} \vec{v} = 0.$$

‘ ,

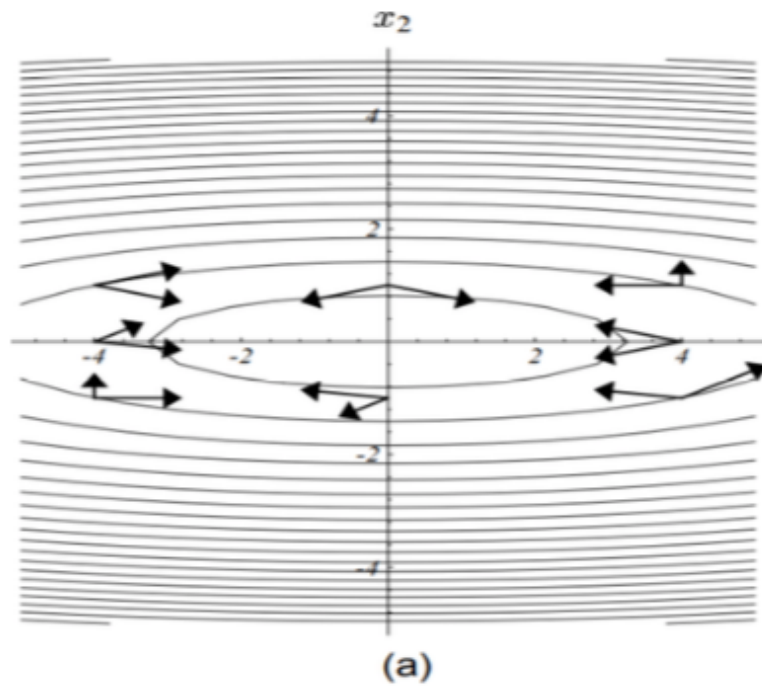
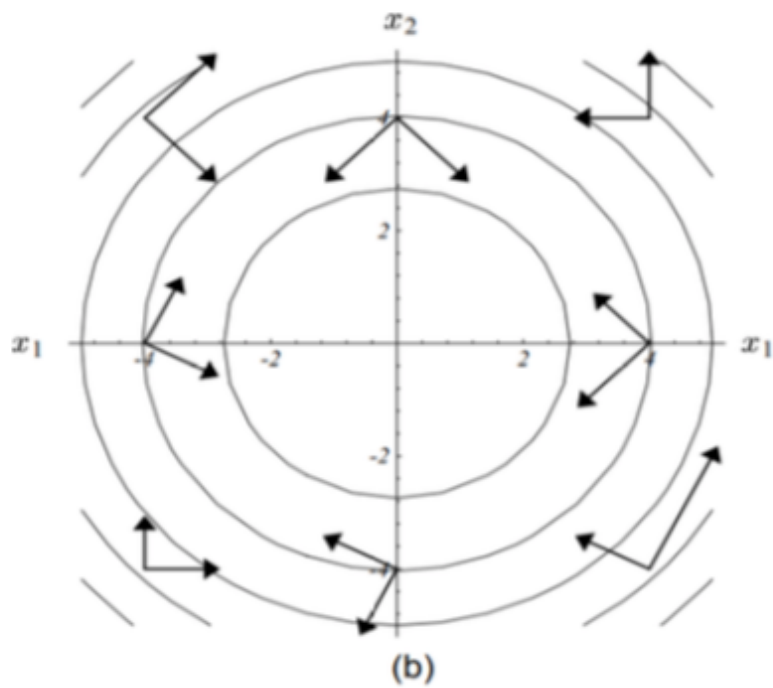


Figure 14.1: conjugate_02.png

‘ ,



-from wikipedia

```

 $\mathbf{r}_0 := \mathbf{b} - \mathbf{A}\mathbf{x}_0$ 
if  $\mathbf{r}_0$  is sufficiently small, then return  $\mathbf{x}_0$  as the result
 $\mathbf{p}_0 := \mathbf{r}_0$ 
 $k := 0$ 
repeat
     $\alpha_k := \frac{\mathbf{r}_k^\top \mathbf{r}_k}{\mathbf{p}_k^\top \mathbf{A} \mathbf{p}_k}$ 
     $\mathbf{x}_{k+1} := \mathbf{x}_k + \alpha_k \mathbf{p}_k$ 
     $\mathbf{r}_{k+1} := \mathbf{r}_k - \alpha_k \mathbf{A} \mathbf{p}_k$ 
    if  $\mathbf{r}_{k+1}$  is sufficiently small, then exit loop
     $\beta_k := \frac{\mathbf{r}_{k+1}^\top \mathbf{r}_{k+1}}{\mathbf{r}_k^\top \mathbf{r}_k}$ 
     $\mathbf{p}_{k+1} := \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k$ 
     $k := k + 1$ 
end repeat
return  $\mathbf{x}_{k+1}$  as the result

```

wikipedia

MATLAB

<https://www.zhihu.com/question/27157047/answer/121950241>

14.3

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

A

- dense and/or small
- large and sparse, or not available explicitly
- narrow-banded

Tridiagonal matrix:

$$\begin{pmatrix} a_1 & b_1 & & & \\ c_1 & a_2 & b_2 & & \\ & c_2 & \ddots & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ & & & c_{n-1} & a_n \end{pmatrix}$$

Pentadiagonal matrix

$$\begin{pmatrix} c_1 & d_1 & e_1 & 0 & \cdots & \cdots & 0 \\ b_1 & c_2 & d_2 & e_2 & \ddots & & \vdots \\ a_1 & b_2 & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & a_2 & \ddots & \ddots & \ddots & e_{n-3} & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & d_{n-2} & e_{n-2} \\ \vdots & & \ddots & a_{n-3} & b_{n-2} & c_{n-1} & d_{n-1} \\ 0 & \cdots & \cdots & 0 & a_{n-2} & b_{n-1} & c_n \end{pmatrix}.$$

, 0

- symmetric positive definite dense and/or small Cholesky
 - symmetric positive definite large and sparse
 - symmetric indefinite, dense and/or small Bunch–Kaufman
 - symmetric indefinite, large and sparse MINRES
 - nonsymmetric, large and sparse GMRES BiCGSTAB or IDR
-
- https://en.wikipedia.org/wiki/Gradient_descent
 - https://en.wikipedia.org/wiki/Conjugate_gradient_method
 - Solution of Linear Systems via Chen Greif

Chapter 15

{Interpolate}

B

/ ϕ_1, ϕ_2, \cdots

$$f(x) = \sum_i a_i \phi_i(x)$$

a

k $(x_1, y_1), \cdots, (x_k, y_k)$ $x_1 < x_2 < \cdots < x_k$

15.1

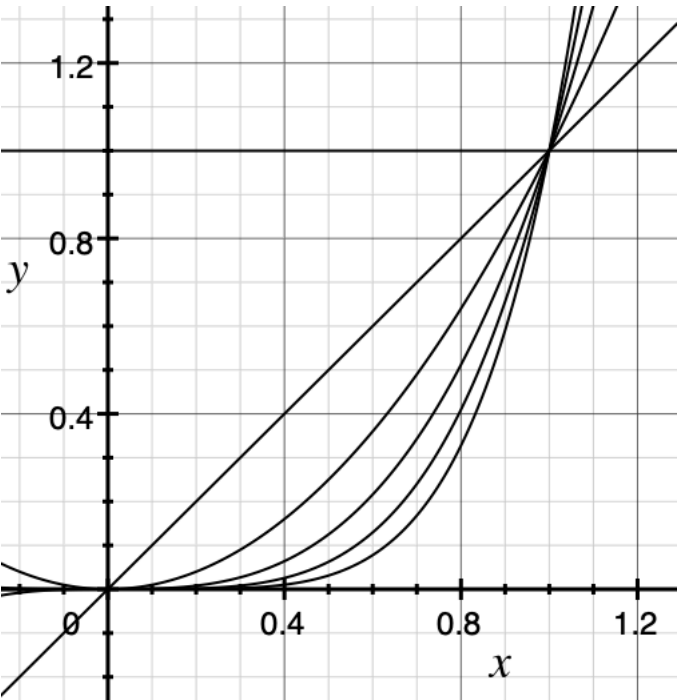
:

$$f(x) = a_0 + a_1x + a_2x^2 + \cdots + a_{k-1}x^{k-1}$$

$$\begin{bmatrix} 1 & x_1 & \cdots & x_1^{k-1} \\ 1 & x_2 & \cdots & x_2^{k-1} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{k-1} & \cdots & x_{k-1}^{k-1} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{k-1} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_k \end{bmatrix}$$

x,y a,

$$\{1, x, x^2, \cdots, x^{k-1}\}$$



B

15.2

$$\phi_i(x) = \frac{\prod_{j \neq i} (x - x_j)}{\prod_{j \neq i} (x_i - x_j)}$$

$$\phi_i(x_l) = \begin{cases} 1, l = i \\ 0, otherwise \end{cases}$$

*

$$x_i \approx x_j,$$

a

15.3

Chapter 16

{Numerical Intergration}

16.1

$f : \mathbb{R} \rightarrow \mathbb{R}$

16.1.1

/

$$\int_a^b f(x) dx \approx \sum_{i=0}^{n-1} f(t_i)(x_{i+1} - x_i)$$

$$\sum_{i=0}^{n-1} f(t_i)(x_{i+1} - x_i)$$

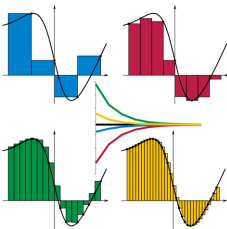
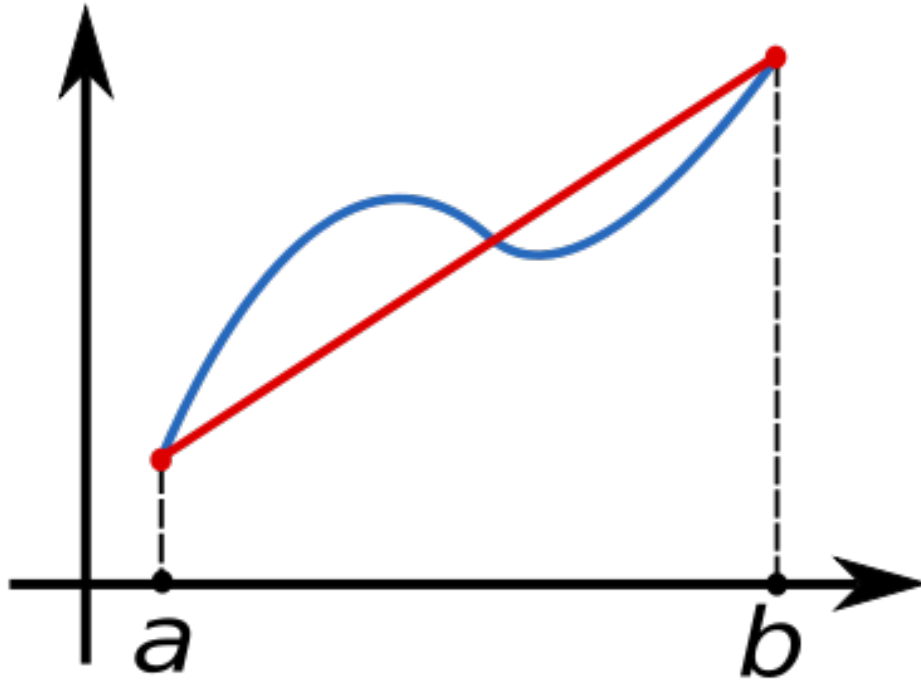


Figure 16.1: wikipedia

$$f(t_i)$$

16.1.2

‘ , ’

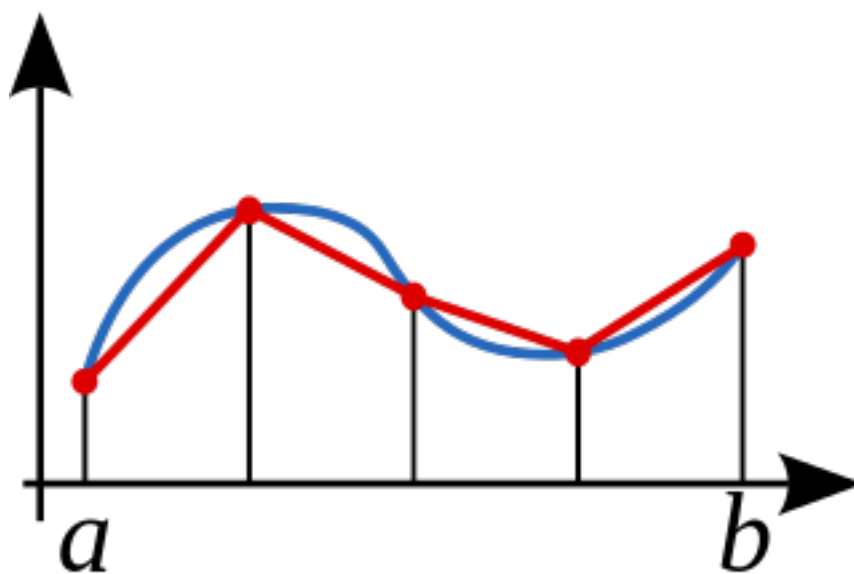


$$\int_a^b f(x) dx \approx \frac{b-a}{2} [f(a) + f(b)]$$

$$\int_a^b f(x) dx \approx (b-a) f\left(\frac{a+b}{2}\right)$$

Trapezoidal rule

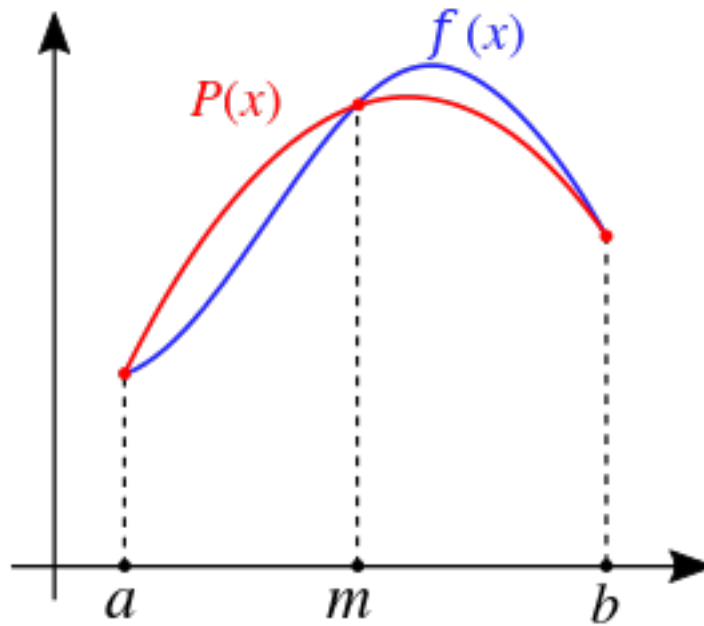
- 1.
- 2.



16.1.3

Simpson's rule -

$$\int_a^b f(x) dx \approx \frac{b-a}{6} \left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right]$$



$$f(x) = Ax^2 + Bx + C$$

Cotes formula

- Newton-Cotes rule / Newton-

16.1.4 -

n -

$$\int_a^b f(x) dx \approx \sum_{i=0}^n w_i f(x_i)$$

- $f(x_0), f(x_1), \dots, f(x_n)$
- $n+1$ $f(x)$
- n

$$\int_a^b f(x) dx \approx \sum_{i=0}^n w_i f(x_i)$$

$$\begin{aligned}\int_a^b f(x)dx &= \int_a^b \left[\sum_i a_i \phi_i(x) \right] dx \\ &= \sum_i a_i \left[\int_a^b \phi_i(x) \right] dx \\ &= \sum_i c_i a_i, \text{ for } c_i \equiv \int_a^b \phi_i(x) dx\end{aligned}$$

$$w_1\cdot 1+w_2\cdot 2+\cdots+w_n\cdot 1=\int_a^b 1dx=b-a w_1\cdot x_1+w_2\cdot x_2+\cdots+w_n\cdot x_n=\int_a^b xdx=(b^2-a^2)/2:w_1\cdot x_1^{n-1}+w_2\cdot x_2^{n-1}+\cdots+w_n\cdot x$$

16.1.5

$$f:\mathbb{R}^k\rightarrow\mathbb{R}\qquad\qquad\qquad\begin{matrix}O(\Delta x^3),&&O(\Delta x^4)\\&&/\end{matrix}$$

16.2

$$f'(x)=\sum_i a_i \phi_i'(x)$$

$$f'(x)=\lim_{h\rightarrow 0}\frac{f(x+h)-f(x)}{h}$$

h

$$f'(x)\approx \frac{f(x+h)-f(x)}{h}$$

x

$$f'(x)\approx \frac{f(x)-f(x-h)}{h}$$

$$f(x+h) = f(x) + f'(x)h + \frac{1}{2}f''(x)h^2 + \dots$$

$O(h)$:

$$f'(x) = \frac{f(x+h) - f(x)}{h} + O(h)$$

$$f(x+h) = f(x) + f'(x)h + \frac{1}{2}f''(x)h^2 + \frac{1}{6}f'''(x)h^3 + \dots \quad (1)$$

$$f(x-h) = f(x) - f'(x)h + \frac{1}{2}f''(x)h^2 - \frac{1}{6}f'''(x)h^3 + \dots \quad (2)$$

$$f(x+h) - f(x-h) = 2f'(x) + \frac{1}{3}f'''(x)h^3 + \dots$$

$O(h^2)$

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}$$

1 2 $O(h)$

$$f''(x) \approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$

$$D(h) = \frac{f(x+h) - f(x)}{h} = f'(x) + \frac{1}{2}f''(x)h + O(h^2)$$

α $D(h), D(\alpha h)$:

$$D(h) = f'(x) + \frac{1}{2}f''(x)h + O(h^2) \quad D(\alpha h) = f'(x) + \frac{1}{2}f''(x)\alpha h + O(h^2)$$

$$\begin{pmatrix} D(h) \\ D(\alpha h) \end{pmatrix} = \begin{pmatrix} 1 & \frac{1}{2}h \\ 1 & \frac{1}{2}\alpha h \end{pmatrix} \begin{pmatrix} f'(x) \\ f''(x) \end{pmatrix} + O(h^2)$$

$$\begin{pmatrix} f'(x) \\ f''(x) \end{pmatrix} = \begin{pmatrix} 1 & \frac{1}{2}h \\ 1 & \frac{1}{2}\alpha h \end{pmatrix}^{-1} \begin{pmatrix} D(h) \\ D(\alpha h) \end{pmatrix} + O(h^2)$$

$O(h^2)$ $f'(x)$:

$$f'(x) = \frac{1}{1-\alpha}(-\alpha D(h) + D(\alpha h)) + O(h^2)$$

-
-
- -
- Taylor series
- Riemann_sum
- Trapezoidal rule
- Simpson's rule

/

Scipy integrate SciPy

Chapter 17

{ODE}

- ordinary differential equation (ODE)

$$m \frac{d^2x}{dt^2} = F(x)$$

logistic function:

$$\frac{dP}{dt} = P$$

17.1

$F(t) : \mathbb{R} \rightarrow \mathbb{R}^n$ satisfying : $F[t, f(t), f'(t), f''(t), \dots, f^{(k)}(t)] = 0$ Given $f(0), f'(0), f''(0), \dots, f^{(k-1)}(0)$

$$f(t), f'(t), \dots, f^{(k)}(t) \quad t$$

- vs
n implicit explicit

$$f^{(k)}(t) = F[t, f(t), f'(t), f''(t), \dots, f^{(k-1)}(t)]$$

- vs

autonomous f

$$\frac{d}{dt}x(t) = f(x(t))$$

$$\frac{d}{dt}x(t) = g(x(t),t)$$

•

.

17.2 ODE

ODE

$$y''' = 3y'' - 2y' + y$$

$$\frac{d^2y}{dt^2} = \frac{d}{dt}\left(\frac{dy}{dt}\right)$$

$$z = \frac{dy}{dt}w = \frac{d^2y}{dt^2}$$

$$\frac{d}{dt}\begin{pmatrix} y \\ z \\ w \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & -2 & 3 \end{pmatrix} \begin{pmatrix} y \\ z \\ w \end{pmatrix}$$

ODE ODE ODE:

$$f^{(k)}(t) = F[t, f(t), f'(t), f''(t), \dots, f^{(k-1)}(t)]$$

ODE:

$$\frac{d}{dt} \begin{pmatrix} g_1(t) \\ g_2(t) \\ \vdots \\ g_{k-1}(t) \\ g_k(t) \end{pmatrix} = \begin{pmatrix} g_2(t) \\ g_3(t) \\ \vdots \\ g_k(t) \\ F[t, g_1(t), g_2(t), \dots, g_{(k-1)}(t)] \end{pmatrix}$$

$$g_2(t) = g_1'(t), g_3(t) = g_2'(t) = g_1''(t).$$

$$f'(t) = F[t, f(t)], \quad g(t) = t, \quad f'(t)$$

$$\frac{d}{dt} \begin{pmatrix} g(t) \\ f(t) \end{pmatrix} = \begin{pmatrix} 1 \\ F[g(t), f(t)] \end{pmatrix}$$

ODE

$$f'(t) = F[f(t)]$$

17.3

ODE

- slope field

A slope field is a collection of short line segments, whose slopes match that of a solution of a first-order differential equation passing through the segment's midpoint. The pattern produced by the slope field aids in visualizing the shape of the curve of the solution. This is especially useful when the solution to a differential equation is difficult to obtain analytically.

$$\frac{dy}{dx} = x^2 - x - 2$$

$$x^3/3 - x^2/2 - 2x + 4x^3/3 - x^2/2 - 2xx^3/3 - x^2/2 - 2x - 4$$

ODE

- phase diagram

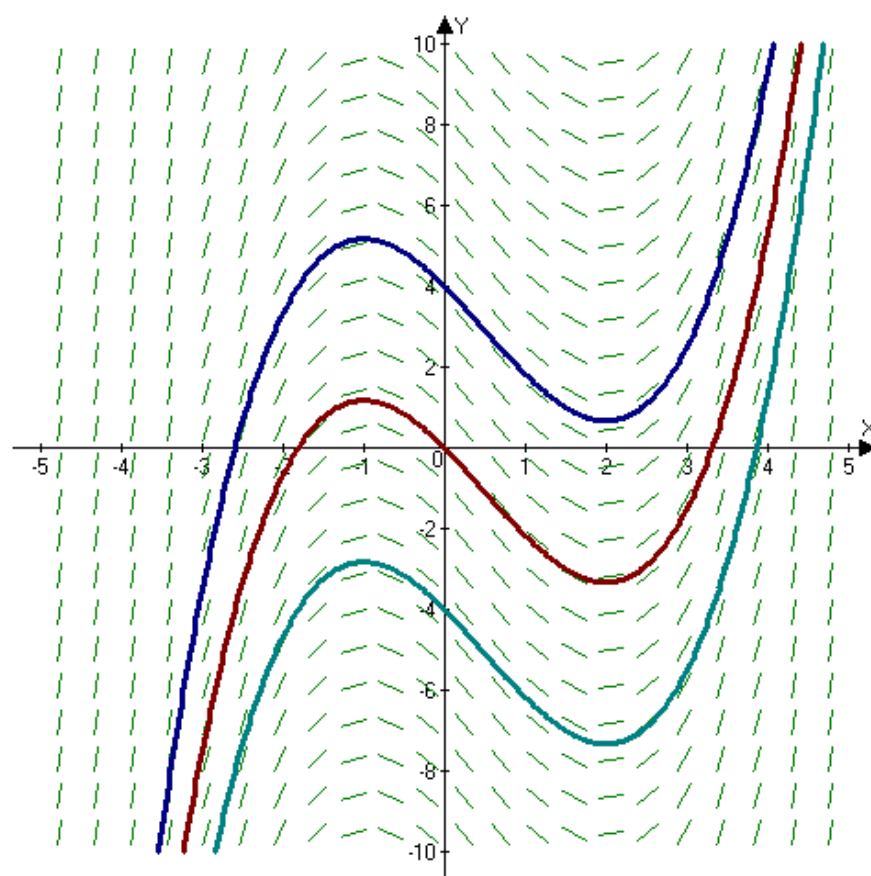
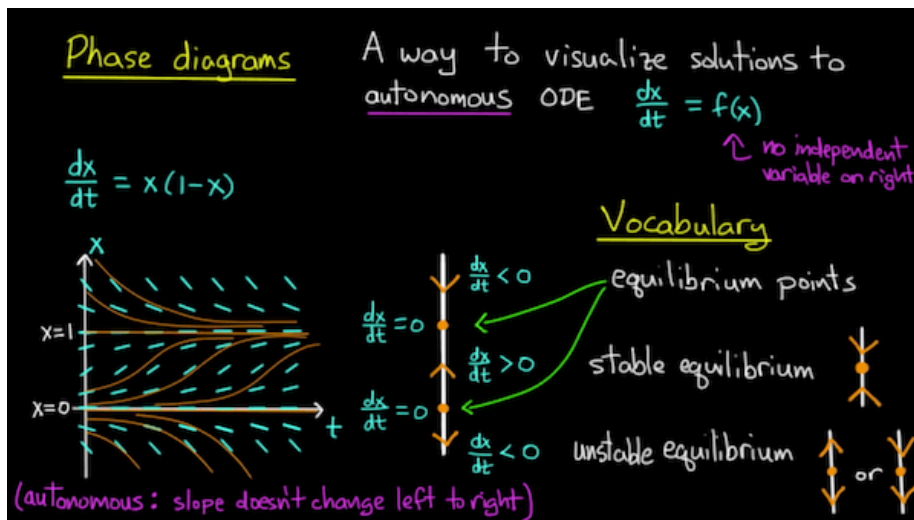


Figure 17.1: Slope_Field.png



$$dx/dt = f(x):$$

$$\frac{dx}{dt} = x(1-x)$$

$$x = 0 \quad x = 1 \quad dx/dt = 0, \quad \text{phase line}$$

17.4

$$y' = 2y/t:$$

$$\frac{dy}{dt} = \frac{2y}{t}$$

$$\ln|y| = 2\ln t + c$$

$$y = Ct^2$$

•

$$y(0) \neq 0$$

•

$$y(0) = 0, \quad C \in \mathbb{R}$$

•

F Lipschitz continuity $|F[\vec{y}] - F[\vec{x}]|_2 \leq L|\vec{y} - \vec{x}|_2$ $f'(t) = F[f(t)]$

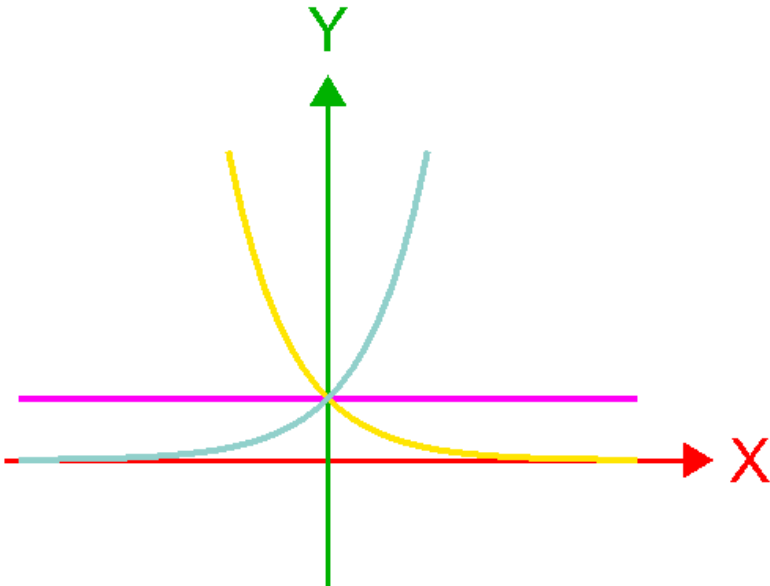
17.5 ODE

ODE

$$y' = ay$$

$$y(t) = Ce^{at}$$

a



- a = 0: y(t) = t
- a < 0: t > y(t) > 0
- a > 0: t < y(t)

ODE

$$\vec{y}' = A\vec{y}$$

$$\vec{y}_1, \dots, \vec{y}_k \text{ A } \lambda_1, \dots, \lambda_k \qquad \vec{y}(0) = c_1\vec{y}_1 + \dots + c_k\vec{y}_k$$

$$\vec{y}(t) = c_1e^{\lambda_1t}\vec{y}_1 + \dots + c_ke^{\lambda_kt}\vec{y}_k$$

$$\text{ODE } \vec{y}' = F[\vec{y}], F$$

$$F[\vec{y}] = F[\vec{y}_0] + J_F(\vec{y}_0)(\vec{y} - \vec{y}_0)$$

$$t_k \quad \vec{y}_k, \quad \vec{y}' = F[\vec{y}] \quad \vec{y}_{k+1}$$

17.6

17.6.1

$$y_{k+1} = y_k + hF[y_k]$$

$$y' = ay \qquad a < 0, 0 \leq h \leq \frac{2}{|a|} \qquad h$$

17.6.2

$$y_k = y_{k+1} - hF[y_{k+1}]$$

$$y_{k+1}.$$

Chapter 18

{PDE}

ODE partial differential equation)

18.1

ODE Lipschitz continuity PDE finite
element method o()o

18.2

:

$f : \mathbb{R}^3 \rightarrow \mathbb{R}, \vec{v} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ Gradient: $\nabla f = (\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \frac{\partial f}{\partial x_3})$ Divergence: $\nabla \cdot \vec{v} = \frac{\partial v_1}{\partial x_1} + \frac{\partial v_2}{\partial x_2} + \frac{\partial v_3}{\partial x_3}$ Curl: $\nabla \times \vec{v} = (\frac{\partial v_3}{\partial x_2} - \frac{\partial v_2}{\partial x_3}, \frac{\partial v_1}{\partial x_3} - \frac{\partial v_3}{\partial x_1}, \frac{\partial v_2}{\partial x_1} - \frac{\partial v_1}{\partial x_2})$

operator	operand	result
Gradient	Multivariate function $f : \mathbb{R}^3 \rightarrow \mathbb{R}$	Vector $\mathbb{R} \rightarrow \mathbb{R}^3$
Divergence	Vector Field $\vec{v} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$	scalar $\mathbb{R}^3 \rightarrow \mathbb{R}$
Curl	Vector Field $\vec{v} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$	Vector Field $\mathbb{R}^3 \rightarrow \mathbb{R}^3$
Laplacian	Multivariate function $f : \mathbb{R}^3 \rightarrow \mathbb{R}$	scalar $\mathbb{R}^3 \rightarrow \mathbb{R}$

•
•

$f(t;x,y,z)$ nabla $\nabla = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z})$, t

18.3 - Navier-Stokes equations

Navier-Stokes equations PDE

$\rho\left(\frac{\partial \vec{v}}{\partial t}+\vec{v}\cdot\nabla\vec{v}\right)=-\nabla p+\mu\nabla^2\vec{v}+\vec{f}$ $t\in[0,\infty)$: time $\vec{v}(t):\Omega\rightarrow\mathbb{R}^3$: velocity $\rho(t):\Omega\rightarrow\mathbb{R}$: density $p(t)$:

wikipedia

$$\overbrace{\rho\left(\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v}\right)}^{\text{Inertia}} = \underbrace{-\nabla p}_{\text{Pressure gradient}} + \underbrace{\mu \nabla^2 \mathbf{v}}_{\text{Viscosity}} + \underbrace{\mathbf{f}}_{\text{Other forces}}$$

Unsteady acceleration Convective acceleration

Navier-Stokes equations

Prove or give a counter-example of the following statement:

In three space dimensions and time, given an initial velocity field, there exists a vector velocity and a scalar pressure field, which are both smooth and globally defined, that solve the Navier–Stokes equations.

\$1,000,000

- P vs NP
-
-
-
- ...

18.4 Maxwell’s equations

PDE -

Gauss’s law: $\nabla\cdot\mathbf{E}=\frac{\rho}{\varepsilon_0}$ Gauss’s law for magnetism: $\nabla\cdot\mathbf{B}=0$ Maxwell–Faraday equation: $\nabla\times\mathbf{E}=-\frac{\partial\mathbf{B}}{\partial t}$ A

18.5 Laplace's equation

$\nabla^2 f(\vec{x}) = 0$

harmonic function

harmonic (overtone)

- $f : \mathbb{R} \rightarrow \mathbb{R} :$,

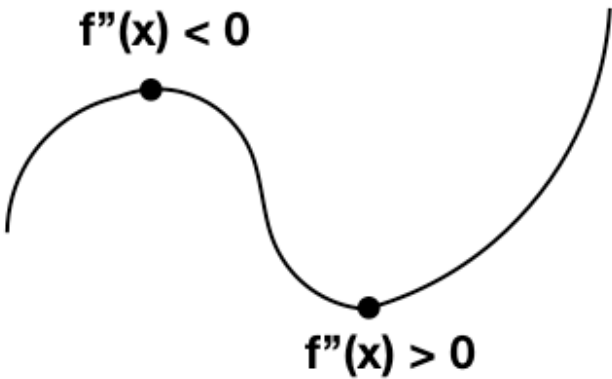


Figure 18.1: 2nd_derivative.png

- $f : \mathbb{R}^n \rightarrow \mathbb{R} :$
harmonic function: $f(x,y) = e^x \sin y$
,

Hodge

.....

, setup

minimize_f $\int_{\Omega} \| \nabla f(\vec{x}) \|_2^2 \, d\vec{x}$ such that $f(\vec{x}) = g(\vec{x}), \forall \vec{x} \in \partial\Omega$

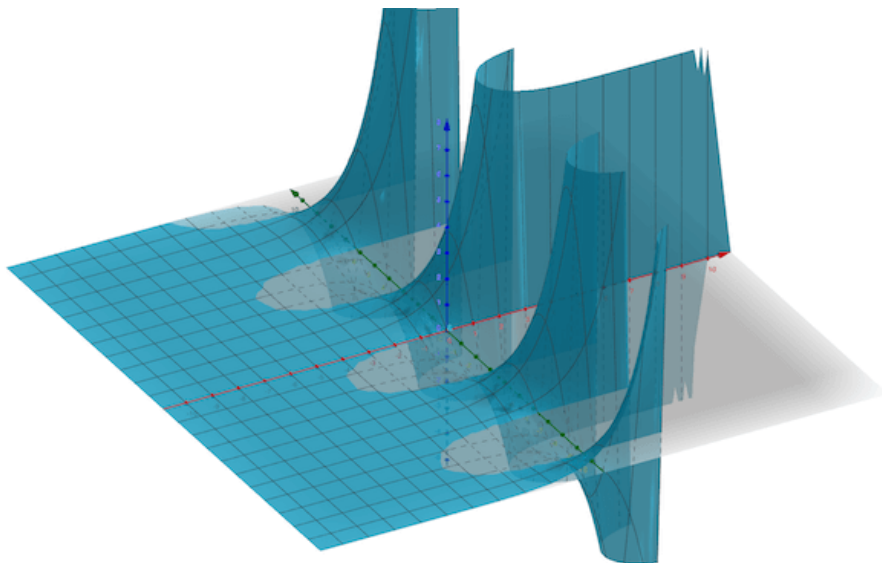


Figure 18.2: harmonic_f2.png

$$\begin{array}{ll} \Omega & \partial\Omega \\ \text{minimize}_f \int_{\Omega} \|\nabla f(\vec{x})\|_2^2 d\vec{x} & \text{Dirichlet's energy} \end{array}$$

$$E[f] = \int_{\Omega} \|\nabla f(\vec{x})\|_2^2 d\vec{x}$$

energy function

L^2 norm

- $f = g$
- f

f 'as smooth as possible' 'as rigid as possible'

f

h

$$E[f + h] \geq E[f]$$

$E[f + \epsilon h] :$

$$\begin{aligned} E[f + \epsilon h] &= \int_{\Omega} \|\nabla f(\vec{x}) + \epsilon \nabla h(\vec{x})\|_2^2 d\vec{x} \\ &= \int_{\Omega} (\|\nabla f(\vec{x})\|_2^2 + 2\epsilon \nabla f(\vec{x}) \cdot \nabla h(\vec{x}) + \epsilon^2 \|\nabla h(\vec{x})\|_2^2) d\vec{x} \end{aligned}$$

ϵ

$$\begin{aligned}\frac{d}{d\epsilon}E[f + \epsilon h] &= \int_{\Omega} (2f(\vec{x}) \cdot \nabla h(\vec{x}) + 2\epsilon \|\nabla h(\vec{x})\|_2^2) d\vec{x} \\ \frac{d}{d\epsilon}E[f + \epsilon h]|_{\epsilon=0} &= 2 \int_{\Omega} (f(\vec{x}) \cdot \nabla h(\vec{x})) d\vec{x}\end{aligned}$$

$$h(\vec{x}) = 0, \vec{x} \in \partial\Omega$$

$$\int_{\Omega} \nabla u \cdot \nabla v \, d\Omega = \int_{\Gamma} v \nabla u \cdot \hat{\mathbf{n}} \, d\Gamma - \int_{\Omega} v \nabla^2 u \, d\Omega$$

$$\frac{d}{d\epsilon}E[f + \epsilon h]|_{\epsilon=0} = -2 \int_{\Omega} (h(\vec{x}) \nabla^2 f(\vec{x})) d\vec{x}$$

0

$$\nabla^2 f(\vec{x}) = 0, x \in \Omega \quad \partial\Omega$$

PDE

$$\begin{aligned}\nabla^2 f(\vec{x}) &= 0 \\ f(\vec{x}) &= g(\vec{x}), \forall \vec{x} \in \partial\Omega\end{aligned}$$

Dirichlet problem

$$\mathbb{R}^n \quad \quad \quad \mathbf{g} \quad \quad \quad \mathbf{f} \quad \quad \quad \mathbf{f} \quad \quad \quad \mathbf{f} = \mathbf{g}$$

18.6 Harmonic analysis

$$\nabla^2 f = \lambda f$$

PDE

18.7 Boundary Value Problems

Dirichlet problem

- Dirichlet conditions: $f(\vec{x}) = g(\vec{x})$ on $\partial\Omega$
- Neumann conditions: $\nabla f(\vec{x}) = g(\vec{x})$ on $\partial\Omega$
- Robin boundary condition: $a f(\vec{x}) + b \nabla f(\vec{x}) = g(\vec{x})$ on $\partial\Omega$

18.8 PDE

PDE

$$\sum_{ij} a_{ij} \frac{\partial f}{\partial x_i \partial x_j} + \sum_i b_i \frac{\partial f}{\partial x_i} + cf = 0$$

$$(\nabla^T A \nabla + \nabla \cdot \vec{b} + c)f = 0$$

- A elliptic
- A 0 parabolic
- A hyperbolic
- ultrahyperbolic

18.9 PDE

- &
- C^∞
- / $\nabla^2 f = g$

18.10 PDE

-
- $\frac{\partial f}{\partial t} = \alpha \nabla^2 f$
-

18.11 PDE

- : $\frac{\partial^2 f}{\partial t^2} = c^2 \nabla^2 f$
- :

18.12

$$y_k'' :$$

$$y_k'' = \frac{y_{k+1} - 2y_k + y_{k-1}}{h^2}$$

$$f(x) \in [0,1]$$

$$y_0 = f(0), y_1 = f(h), y_2 = f(2h), \dots, y_n = f(nh)$$

‘ ’ :

$$y_k'' = \frac{y_{k+1} - 2y_k + y_{k-1}}{h^2}$$

$$y_k'' h^2 = y_{k+1} - 2y_k + y_{k-1}$$

$$y_k \quad \vec{y} \in \mathbb{R}^{n+1} \quad y_k'' \quad \vec{w} \in \mathbb{R}^{n+1} \quad :$$

$$h^2 \vec{w} = L_1 \vec{y}$$

$$L_1$$

- Dirichlet

$$\begin{pmatrix} -2 & 1 & & & & \\ 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & -2 & 1 \\ & & & & 1 & -2 \end{pmatrix}$$

- Neumann

$$\begin{pmatrix} -1 & 1 & & & & \\ 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & -2 & 1 \\ & & & & 1 & -1 \end{pmatrix}$$

- $f(0) = f(1)$

$$\begin{pmatrix} -2 & 1 & & & & 1 \\ 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & -2 & 1 \\ 1 & & & & 1 & -2 \end{pmatrix}$$

2D

$$(\nabla^2 y)_{k,l} = \frac{1}{h^2} (y_{(k-1),l} + y_{k,(l-1)} + y_{(k+1),l} + y_{k,(l+1)} - 4y_{k,l})$$