SUPPORT VECTOR MACHINES & NEURAL NETWORKS:

LECTURE 1 – REVIEW

1. Basics of linear algebra
2. Important concepts of optimization
3. Software platforms and data banks

Commonly used notations

Sets: S, T, U, V (uppercase, italic)

Scalars, constants: $\alpha, \beta, \gamma, s, r, t, a, b, c$ (lowercase)

Vectors: $\mathbf{x}, \mathbf{y}, \mathbf{z}, \mathbf{p}, \mathbf{u}, \mathbf{v}$ (lowercase, bold, column vectors)

Matrices: $A_{m\times n}$, $M_{n\times n}$ (uppercase, dimensionality)

Real spaces: $\mathbb{R}^n$, $\mathbb{R}$, $\mathbb{R}^n_+$, $\mathbb{R}^n_{++}$, $N(A)$, $R(A)$

Matrix spaces: $\boldsymbol{S}^n$, $\boldsymbol{S}^n_+$, $\boldsymbol{S}^n_{++}$, $\boldsymbol{M}_{m\times n}$ (uppercase, bold, dimensionality)

Functions: $f(x)$, $g_j(x)$, $h_i(x)$ (lowercase, italic)

Basics of linear algebra - Vector

- Definition: a vector $\boldsymbol{v}$ (or $\vec{v}$) in $\mathbb{R}^n$ is a geometric object that has magnitude (or length) and direction. It is usually denoted by $\boldsymbol{v}$ (or $\vec{v}$) = $(v_1, v_2, .., v_n)^T$ in its column form.

- Given two points $\boldsymbol{h}$, $\boldsymbol{t}$ $\in \mathbb{R}^n$, then $\boldsymbol{v} = \boldsymbol{h} - \boldsymbol{t} \in \mathbb{R}^n$ is a vector pointing from $\boldsymbol{t}$ (tail) to $\boldsymbol{h}$ (head).

- The length of vector $\boldsymbol{v} \in \mathbb{R}^n$ is given by

$$||\boldsymbol{v}|| = \sqrt{v_1^2 + v_2^2 + .. + v_n^2}.$$

- The vector $\frac{\boldsymbol{v}}{||\boldsymbol{v}||}$ is a unit vector (a vector with length = 1).

Basics of linear algebra - Vector

- (multiplication by a scalar) Given a scalar $\alpha \in \mathbb{R}$ and vector $\boldsymbol{v} \in \mathbb{R}^n$, then

$$\alpha \boldsymbol{v} = (\alpha v_1, \alpha v_2, .., \alpha v_n)^T.$$

- (vector addition) Given two vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$, then

$$\boldsymbol{u} + \boldsymbol{v} = (u_1 + v_1, u_2 + v_2, .., u_n + v_n)^T.$$

- (inner product) Given two vectors $\boldsymbol{u}$, $\boldsymbol{v} \in \mathbb{R}^n$, then

$$<\boldsymbol{u}, \boldsymbol{v}> = \boldsymbol{u} \cdot \boldsymbol{v} = \boldsymbol{u}^T \boldsymbol{v} = \sum_{i=1}^{n} u_i v_i.$$

- (orthogonality) Two vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$ are called orthogonal, if $<\boldsymbol{u}, \boldsymbol{v}> = 0$. We denote $\boldsymbol{u} \perp \boldsymbol{v}$.

Questions

- What's the role of vectors in machine learning?
- How are two given vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$ related? Any implications?
- How to find a vector $\boldsymbol{w}$ orthogonal to a given vector $\boldsymbol{u} \in \mathbb{R}^n$?
- Can you find a third vector $\boldsymbol{s}$ that is orthogonal to both of $\boldsymbol{u}$ and $\boldsymbol{w}$?

*** Questions

- Hint: (geometric meaning of inner product)

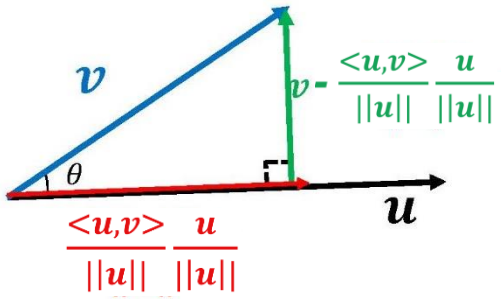
  $$\langle u, v \rangle = \|u\| \, \|v\| \cos\theta$$

  hence $\frac{\langle u,v \rangle}{\|u\|} \frac{u}{\|u\|}$ is the projection of v on u

  (the portion of v related to u !)

  and $w \triangleq v - \frac{\langle u,v \rangle}{\|u\|} \frac{u}{\|u\|}$ is orthogonal to u.

  (the portion of v independent of u !)



- When u and v are unit vectors, $\langle u, v \rangle$ represents the amount of correlated information of u and v.

- Now, s = ?

Basics of linear algebra - Vector

- (norm) A vector norm $||\cdot||$ is a function from $\mathbb{R}^n$ to $\mathbb{R}$ such that (i) $||\boldsymbol{v}|| \geq 0$ for any $\boldsymbol{v} \in \mathbb{R}^n$, and $||\boldsymbol{v}|| = 0$ if and only if $\boldsymbol{v} = 0$; (ii) $||\boldsymbol{u} + \boldsymbol{v}|| \leq ||\boldsymbol{u}|| + ||\boldsymbol{v}||$ for any $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$;
  (iii) $||\alpha\boldsymbol{v}|| = \alpha||\boldsymbol{v}||$ for any $\alpha \in \mathbb{R}$ and $\boldsymbol{v} \in \mathbb{R}^n$.
- (commonly used norm)
  - l_0 norm: $\|\boldsymbol{x}\|_0 = \sum_{i=1}^{n} \mathbb{1}(x_i \neq 0)$;
  - l_1 norm: $\|\boldsymbol{x}\|_1 = \sum_{i=1}^{n} |x_i|$;
  - l_2 norm: $\|\boldsymbol{x}\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$;
  - l_p norm: $\|\boldsymbol{x}\|_p = (\sum_{i=1}^{n} |x_i|^p)^{\frac{1}{p}}$, $p \geq 1$;
  - l_∞ norm: $\|\boldsymbol{x}\|_\infty = \max_{i=1,\ldots,n} |x_i|$.

Questions

- How are these commonly used norms related?

  Example: $\boldsymbol{v}^T = (1\ 0\ 2\ 0)$

  1. Magnitude/size

  2. What's the graph of the set $\{\boldsymbol{x} \in \mathbb{R}^n \mid ||\boldsymbol{x}|| \leq 1\}$? Any implications?

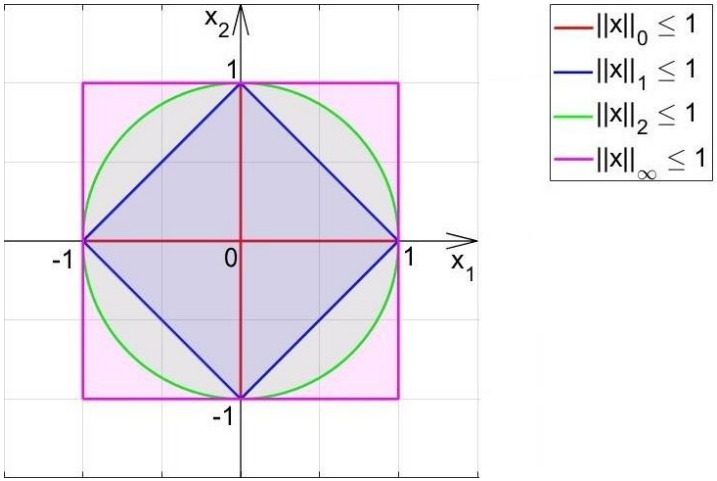
*** Questions

- How are these commonly used norms related?

  Example: $\boldsymbol{v}^T = (1\ 0\ 2\ 0)$

  1. Magnitude/size

  2. What's the graph of the set $\{\boldsymbol{x} \in \mathbb{R}^n \mid ||\boldsymbol{x}|| \leq 1\}$? Any implications?



Basics of linear algebra - Vector

- Definition: A set V of p vectors $\{\boldsymbol{v}_1, \boldsymbol{v}_2, .., \boldsymbol{v}_p\}$ in $\mathbb{R}^n$ are called linearly independent,

  if $\alpha_1\boldsymbol{v}_1 + \alpha_2\boldsymbol{v}_2 + \cdots + \alpha_p\boldsymbol{v}_p = 0$ implies that

  $\alpha_1 = \alpha_2 = \cdots = \alpha_p = 0.$

  Otherwise, they are linearly dependent.

  (span) The set of all vectors generated by a linear combination of the vectors in V is called the span of V.

  We have

  $$span(V) = \{\alpha_1\boldsymbol{v}_1 + \alpha_2\boldsymbol{v}_2 + \cdots + \alpha_p\boldsymbol{v}_p | \ \alpha_1, .., \alpha_p \in \mathbb{R}\},$$

  $$\dim\big(span(V)\big) = \#\text{ of linearly independent vectors in } V.$$

Basics of linear algebra - Matrix

- Definition: a (real) matrix $A \in \boldsymbol{M}_{mxn}(\mathbb{R})$ in is a rectangular array of entries $(a_{ij} \in \mathbb{R})$ $(i = 1,..,m; j = 1,..,n)$ in m rows and n columns.

- Given $$\boldsymbol{A} = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix} = \begin{pmatrix} \boldsymbol{a}_1^{\mathrm{T}} \\ \vdots \\ \boldsymbol{a}_m^{\mathrm{T}} \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}_1 & \dots & \boldsymbol{A}_n \end{pmatrix},$$

  $\boldsymbol{a}_i \in \mathbb{R}^n, i = 1,..,m$, are row vectors of matrix A

  $A_j \in \mathbb{R}^m, j = 1,..,n$, are column vectors of matrix A

- *When $m = n$, A* is a *square matrix* in $\boldsymbol{M}_{n\times n}(\mathbb{R}) \triangleq \boldsymbol{M}_n(\mathbb{R})$.

Basics of linear algebra – Square matrix

- (diagonal & triangular matrix) Given a square matrix $A = (a_{ij}) \in \boldsymbol{M}_n(\mathbb{R})$, $\{a_{ii}\}, i = 1, .., n$, are called the main diagonal entries.

  (i) If all entries of *A* below the main diagonal are zero,

  A is an *upper triangular matrix*;

  (ii) If all entries of *A* above the main diagonal are zero;

  A is a *lower triangular matrix*;

  (iii) If all entries off the main diagonal are zero,

  A is a *diagonal matrix.*

- (identity matrix) The identity matrix I_n is an $n \times n$ diagonal matrix with all main diagonal entries being 1.

- "rectangular diagonal" is an extended idea for non-square matrices.

Questions

- What's the role of matrices in machine leaning?

  Example 1 - record of objects in terms of features/characteristics specifying the objects

  Example 2 – dynamics of a linear system

  Example 3 – Image data

*** Questions

- What's the role of a matrix in machine learning?

Example 1 – record of objects in terms of features/characteristics specifying the objects

Iris flower data set

Iris setosa

Iris versicolor

Iris versginica

		features				class
		sepal length	sepal width	petal length	petal width	Setosa / versicolor / versginica
A =	observation 1	5.1	3.5	1.4	0.2	Iris-setosa
	⋮	⋮	⋮	⋮	⋮	⋮
	observation 92	6.1	3.0	4.6	1.4	Iris-versicolor
	⋮	⋮	⋮	⋮	⋮	⋮
	observation 150	5.9	3.0	5.1	1.8	Iris-virginica

*** Questions

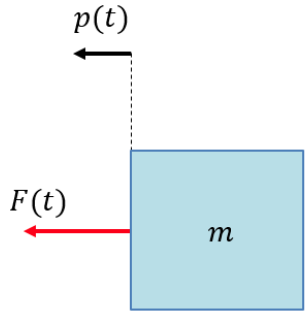
- What's the role of a matrix in machine learning?

Example 2 – dynamics of a linear system

Constant acceleration moving body[1]. The governing equation for a moving body with constant acceleration is Newton's second law:

$$m\ddot{p} = F$$

where p is the body position displacement,
m is the body mass,
F is the external force applied to the body.



Define the state variables x_1 and x_2, and let $\begin{cases} \dot{x_1} = x_2 \\ \dot{x_2} = \ddot{p} \end{cases}$, we have

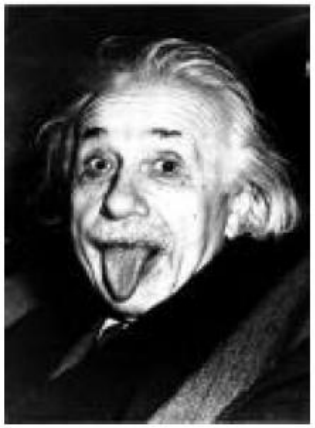
$$\begin{pmatrix} \dot{x_1} \\ \dot{x_2} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 1/m \end{pmatrix} F$$

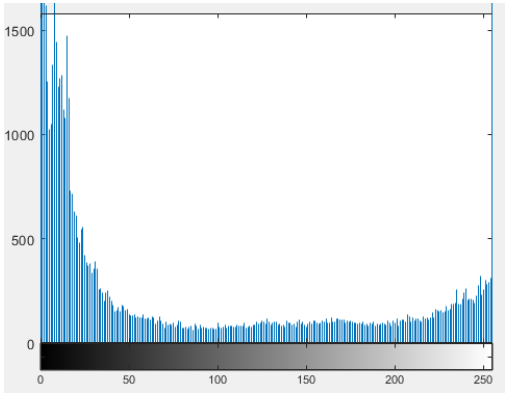
1. https://www.kalmanfilter.net/modeling.html

*** Questions

- What's the role of a matrix in machine learning?

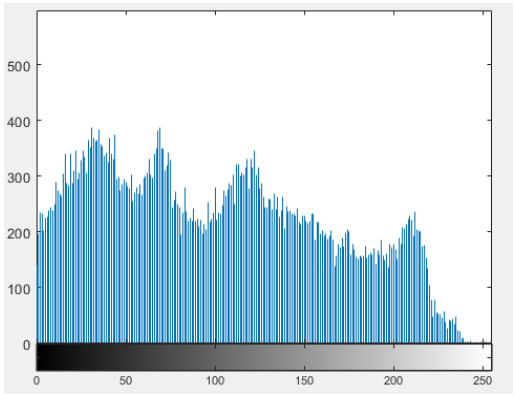
Example 3 – Image data





303x226 uint8

	1	2	3	4	5	6
1	255	255	255	255	255	255
2	255	255	255	251	251	252
3	255	255	255	241	241	241
4	255	255	255	107	107	101
5	255	255	255	81	81	73
6	255	255	255	51	51	44
7	255	255	255	46	46	39
8	255	255	255	45	45	38
9	255	255	255	45	45	38
10	255	255	255	47	47	41
11	255	255	255	44	44	37
12	255	255	255	40	40	33
13	255	255	255	43	43	37
14	255	255	255	43	43	38
15	255	255	255	39	39	34
16	255	255	255	38	38	33
17	255	255	255	39	39	34
18	255	255	255	39	39	34



137x136x3 uint8

```
val(:,:,1) =

  Columns 1 through 9

    37    44    44    41    36    27    17    13    18
    25    40    46    44    39    32    21    15    18
    11    27    44    49    44    36    26    19    23
     2    12    35    50    49    41    32    25    34
     1    10    29    43    50    47    37    35    49
    10    15    19    27    43    51    43    44    57
    16    17    14    19    35    48    46    50    60
    17    12     9    15    28    40    45    51    58
    16    14    15    17    23    35    43    50    53
    12    17    21    18    17    26    39    46    44
    10    19    25    21    17    24    36    36    31
```

Basics of linear algebra - Matrix

- (transposition) Given a matrix $A = (a_{ij}) \in \boldsymbol{M}_{mxn}(\mathbb{R})$, then

  $A^T = (a_{ji}) \in \boldsymbol{M}_{nxm}(\mathbb{R})$

- (multiplication by a scalar) Given a scalar $\alpha \in \mathbb{R}$ and matrix $A \in \boldsymbol{M}_{mxn}(\mathbb{R})$, then

  $\alpha A = (\alpha a_{ij}) \in \boldsymbol{M}_{mxn}(\mathbb{R})$.

- (matrix addition) Given two matrices $A, B \in \boldsymbol{M}_{mxn}(\mathbb{R})$, then

  $A + B = (a_{ij} + b_{ij}) \in \boldsymbol{M}_{mxn}(\mathbb{R})$.

- (matrix multiplication) Given two matrices $A \in \boldsymbol{M}_{mxn}(\mathbb{R})$ and $B \in \boldsymbol{M}_{nxp}(\mathbb{R})$, then

  $AB = (c_{ij}) \in \boldsymbol{M}_{mxp}(\mathbb{R})$ with $c_{ij} = \sum_{k=1}^{n} a_{ik}\, b_{kj}$

** If $A = A_{mx1}$ (a column vector $\boldsymbol{a}$) and $B = B_{1xn}$ (a row vector $\boldsymbol{b}^T$), then $AB = \boldsymbol{a}\boldsymbol{b}^T$ is an $m \times n$ matrix.

Basics of linear algebra – Square matrix

- Given a square matrix $A = (a_{ij}) \in \boldsymbol{M}_n(\mathbb{R})$,

  (symmetric) If $A = A^T$, i.e., $a_{\mathrm{ij}} = a_{ji}$, $\forall\ i, j = 1, .., n$,

  then *A* is a *symmetric matrix*.

  (skew symmetric) If $A = -A^T$, then *A* is a skew symmetric matrix.

  (inverse matrix) If there exists a matrix $B \in \boldsymbol{M}_n(\mathbb{R})$, such that $AB = BA = I_n$, then matrix is invertible (or nonsingular), and we denote its inverse matrix $A^{-1} = B$ (and $B^{-1} = A$, equivalently).

Basics of linear algebra – Square matrix

- Let $A = (a_{ij}) \in \boldsymbol{M}_n(\mathbb{R})$ be a square matrix.

  (determinant) The determinant of A (symmetric or not) is a real number using the Leibniz formula or Laplace expansion to indicate if A is invertible and the volume of the parallelotope *P(A)* formed by the column vectors of matrix A.

(invertible) Matrix A^{-1} exists if and only if $\det(A) \neq 0$.

(volume) $vol\big(P(A)\big) = |\det(A)|$

(degeneracy) If $\det(A) = 0$, then $P(A)$ is degenerated and column vectors $\{A_1, A_2, .., A_n\}$ are linearly dependent. Otherwise, *P(A)* is solid and $\{A_i\}$ are linearly indpendent.

Basics of linear algebra – Square matrix

- Let $A, B \in \boldsymbol{M}_n(\mathbb{R})$ be two square matrices.

- Properties of the determinant function:

  (i) $\det(A) = \det(A^T)$;

  (ii) $\det(I_n) = 1$;

  (iii) $\det(AB) = \det(A)\det(B)$;

  (iv) $\det(A^{-1}) = \frac{1}{\det(A)}$.

  (v) $\det(A)$ = product of eigenvalues of matrix *A*.

Basics of linear algebra – Square matrix

- Let $A = (a_{ij}) \in \boldsymbol{M}_n(\mathbb{R})$ be a square matrix.

  (trace) The trace of A (symmetric or not) is the sum of diagonal entries, $\mathrm{tr}(A) = \sum_{i=1}^{n} a_{ii}$.

- Properties of trace of square matrices: for $A, B \in \boldsymbol{M}_n(\mathbb{R})$:

  (i) $tr(A + B) = tr(A) + tr(B)$

  (ii) $tr(\alpha A) = \alpha\, tr(A)$

  (iii) $tr(A) = tr\,(A^T)$

- Properties of trace of matrices: for $A, B \in \boldsymbol{M}_{m\times n}(\mathbb{R})$:

  $tr(A^T B) = tr(AB^T)$

  $= tr(B^T A) = tr(BA^T) = \sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij} b_{ij}$

Questions

- What is the geometric meaning of the trace of a matrix?

  Hint:

  1. the trace of a square matrix $A \in \boldsymbol{M}_n$ is the sum of its eigenvalues.
  2. square root of the trace of $A^T A$ is the Frobenius norm of a rectangular matrix $A \in \boldsymbol{M}_{m \times n.}$

Basics of linear algebra – Transformation

- (linear transformation) A matrix $A \in \boldsymbol{M}_{m\times n}(\mathbb{R})$ is a linear transformation that maps from $\mathbb{R}^n$ to $\mathbb{R}^m$ such that

$$A\colon \boldsymbol{x} \in \mathbb{R}^n \ \rightarrow \ \boldsymbol{y} = A\boldsymbol{x} \in \mathbb{R}^m.$$

Properties:

(i) For $\alpha, \beta \in \mathbb{R}$ and $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$,

$$A(\alpha\boldsymbol{x} + \beta\boldsymbol{y}) = \alpha A(\boldsymbol{x}) + \beta A(\boldsymbol{y}).$$

(ii) If $\det(A) \neq 0$, linear transformation A is an isomorphism.

(iii) For a square matrix $A \in M_n$, $\boldsymbol{y} = A\boldsymbol{x}$ usually means vector $\boldsymbol{y}$ is a rotation of $\boldsymbol{x}$ followed by a scaling/stretch and another rotation in $\mathbb{R}^n$.

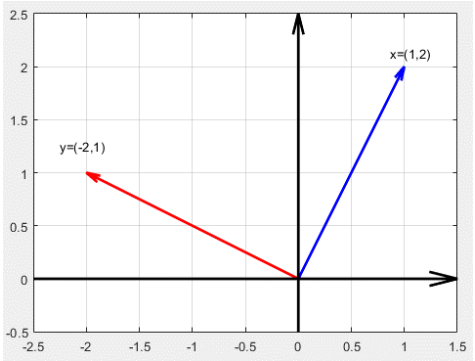
***Basics of linear algebra – Transformation

For example, $\boldsymbol{x} = (1, 2)^T$

- Rotation

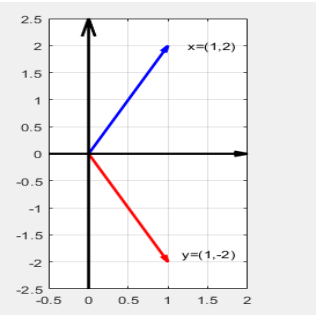
  e.g., $A = \begin{pmatrix} \cos\frac{\pi}{2} & -\sin\frac{\pi}{2} \\ \sin\frac{\pi}{2} & \cos\frac{\pi}{2} \end{pmatrix}$, $\boldsymbol{y} = \begin{pmatrix} -2 \\ 1 \end{pmatrix}$,

  an angle $\frac{\pi}{2}$ counterclockwise



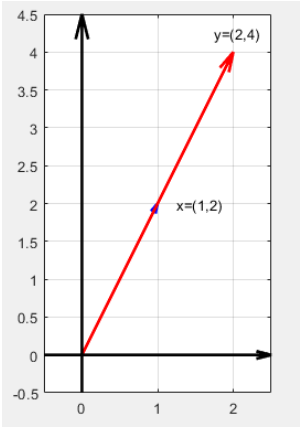
- Reflection

  e.g., $A = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$, $\boldsymbol{y} = \begin{pmatrix} 1 \\ -2 \end{pmatrix}$, reflection through the x axis



- Scaling

  e.g., $A = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$, $\boldsymbol{y} = \begin{pmatrix} 2 \\ 4 \end{pmatrix}$, by 2 in all directions.



Questions:

- For $\boldsymbol{x} \in \mathbb{R}^n$, $A \in \boldsymbol{M}_{m\times n}(\mathbb{R})$, $B \in \boldsymbol{M}_{n\times p}(\mathbb{R})$, $C \in \boldsymbol{M}_{m\times p}(\mathbb{R})$

(i) (matrix-vector product)

What's the operational meaning of $\mathrm{y} = A\boldsymbol{x} \in \mathbb{R}^m$?

(ii) (matrix-matrix product)

What's the operational meaning of $\mathrm{C} = AB$?

Basics of linear algebra – Transformation

- $\boldsymbol{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$, $A_{mxn} = \begin{pmatrix} \boldsymbol{a}_1^T \\ \vdots \\ \boldsymbol{a}_m^T \end{pmatrix} = (\boldsymbol{A}_1 \cdots \boldsymbol{A}_n)$, $B_{nxp} = \begin{pmatrix} \boldsymbol{b}_1^T \\ \vdots \\ \boldsymbol{b}_n^T \end{pmatrix} = (\boldsymbol{B}_1 \cdots \boldsymbol{B}_p)$,

  where $\boldsymbol{a}_i^T$ is $1 \times n$, $i = 1, \dots, m$; $\boldsymbol{b}_j^T$ is $1 \times p$, $j = 1, \dots, n$;

  $\boldsymbol{A}_j$ is $m \times 1$, $j = 1, \dots, n$; $\boldsymbol{B}_k$ is $n \times 1$, $k = 1, \dots, p$.

- $A\boldsymbol{x} = \begin{pmatrix} \boldsymbol{a}_1^T \\ \vdots \\ \boldsymbol{a}_m^T \end{pmatrix} \boldsymbol{x} = \begin{pmatrix} \boldsymbol{a}_1^T \boldsymbol{x} \\ \vdots \\ \boldsymbol{a}_m^T \boldsymbol{x} \end{pmatrix}$ *(algebraic)*

  $A\boldsymbol{x} = (\boldsymbol{A}_1 \cdots \boldsymbol{A}_n) \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \sum_{j=1}^{n} x_j \boldsymbol{A}_j$ *(sum of column vectors)*

- $AB = \begin{pmatrix} \boldsymbol{a}_1^T \\ \vdots \\ \boldsymbol{a}_m^T \end{pmatrix} (\boldsymbol{B}_1 \cdots \boldsymbol{B}_p) = \begin{pmatrix} \boldsymbol{a}_1^T \boldsymbol{B}_1 & \cdots & \boldsymbol{a}_1^T \boldsymbol{B}_p \\ \vdots & \ddots & \vdots \\ \boldsymbol{a}_m^T \boldsymbol{B}_1 & \cdots & \boldsymbol{a}_m^T \boldsymbol{B}_p \end{pmatrix} = \left(\boldsymbol{a}_i^T \boldsymbol{B}_k\right)_{\substack{i=1,\dots,m \\ k=1,\dots,p}}$ *(algebraic)*

  $AB = \begin{pmatrix} \boldsymbol{a}_1^T \\ \vdots \\ \boldsymbol{a}_m^T \end{pmatrix} B = \begin{pmatrix} \boldsymbol{a}_1^T B \\ \vdots \\ \boldsymbol{a}_m^T B \end{pmatrix}$ *(row by row)*

  $AB = A(\boldsymbol{B}_1 \cdots \boldsymbol{B}_p) = (A\boldsymbol{B}_1 \cdots A\boldsymbol{B}_p)$ *(column by column)*

  $AB = (\boldsymbol{A}_1 \cdots \boldsymbol{A}_n) \begin{pmatrix} \boldsymbol{b}_1^T \\ \vdots \\ \boldsymbol{b}_n^T \end{pmatrix} = \sum_{j=1}^{n} \boldsymbol{A}_j \boldsymbol{b}_j^T$ *(sum of rank-1 matrices)*

Covariance matrix

- Data records in "object-feature" format are stored as a matrix A_{mxn} (m objects with n features).
- Matrix AA^T is viewed as the "object" covariance matrix that describes the correlations of each pair of objects.
- Matrix A^TA is viewed as the "feature" covariance matrix that describes the correlations of any two features.
- Sum of rank-1 matrix operations shows a way to construct a covariance matrix.
- Eigenvectors of AA^T represents m eigen-objects.
- Eigenvectors of A^TA represents n eigen-features.

Basics of linear algebra – Matrix norm

- For $A \in \boldsymbol{M}_{m\times n}(\mathbb{R})$, some commonly used matrix norms:
- (i) Frobenius norm

$$||A||_F = \sqrt{\Sigma_{i=1}^{m}\Sigma_{j=1}^{n} a_{ij}^2} = \sqrt{tr(A^T A)}$$

- (ii) matrix p-norm $(p \geq 1)$

$$||A||_p = sup_{\boldsymbol{x}\neq 0} \frac{||A\boldsymbol{x}||_p}{||\boldsymbol{x}||_p} = max_{||\boldsymbol{x}||_p=1} ||A\boldsymbol{x}||_p$$

In particular,

$||A||_1 = max_{j=1,..,n} \sum_{i=1}^{m} |a_{ij}|$ = *maximum absolute column sum of* A;

$||A||_\infty = max_{i=1,..,m} \sum_{j=1,..,n} |a_{ij}|$ = *maximum absolute row sum of* A.

Basics of linear algebra – Eigenvalue and eigenvector

- Definion: For a square matrix $A \in \boldsymbol{M}_n(\mathbb{R})$, an eigenvector $\boldsymbol{u} \in R^n$ is a directional vector such that $A\boldsymbol{u} = \lambda\boldsymbol{u}$ for some $\lambda \in \mathbb{R}$. The corresponding λ is an eigenvalue.

- (geometric meaning) an eigenvector, corresponding to a nonzero eigenvalue, points in a direction in which it is stretched by the transformation A and the eigenvalue is the factor by which it is stretched. If the eigenvalue is negative, the direction is reversed.
- (characteristic equation)

$$\det(A - \lambda I_n) = 0.$$

matrix A has a non-zero eigenvector if and only if the determinant of matrix $A - \lambda I_n$ is zero.

Basics of linear algebra – Eigenvalue and eigenvector

- $\det(A - \lambda I_n)$ is the characteristic polynomial that has n roots $\{\lambda_1, .., \lambda_n\}$ (some may repeat) as eigenvalues.
- (real eigenvalue) All eigenvalues of a symmetric matrix $A \in \boldsymbol{S}_n$ ($\triangleq$ set of all symmetric $n \times n$ matrices) are real.

Properties: For a square matrix $A \in \boldsymbol{M}_n(\mathbb{R})$ with eigenvalues $\{\lambda_i\}$,

(a) The eigenvalues of A^k are $\{\lambda_i^k\}$.

(b) If A^{-1} exists, then the eigenvalues of A^{-1} are $\{\lambda_i^{-1}\}$.

(c) $tr(A) = \Sigma_i \lambda_i = \lambda_1 + \lambda_2 + \cdots + \lambda_n$.

(d) $\det(A) = \Pi_i \lambda_i = \lambda_1 \lambda_2 \cdots \lambda_n$.

Basics of linear algebra- Definite matrix

- (definiteness): A symmetric matrix $M \in \boldsymbol{M}_n(\mathbb{R})$ is
  (i) positive semidefinite (psd) if $\boldsymbol{x}^T M\boldsymbol{x} \geq 0, \forall\, \boldsymbol{x} \in \mathbb{R}^n$;
  (ii) positive definite (pd) if $\boldsymbol{x}^T M\boldsymbol{x} > 0, \forall\, \boldsymbol{x} \in \mathbb{R}^n \setminus \{\boldsymbol{0}\}$;
  (iii) negative semidefinite (nsd) if $\boldsymbol{x}^T M\boldsymbol{x} \leq 0, \forall\, \boldsymbol{x} \in \mathbb{R}^n$;
  (iv) negative definite (nd) if $\boldsymbol{x}^T M\boldsymbol{x} < 0, \forall\, \boldsymbol{x} \in \mathbb{R}^n \setminus \{\boldsymbol{0}\}$.
  Otherwise M is an indefinite matrix.

- Properties:
- (i) M is psd iff all eigenvalues of M are non-negative.
- (ii) M is pd iff all eigenvalues of M are positive.
- (iii) M is nsd iff all eigenvalues of M are non-positive.
- (iv) M is nd iff all eigenvalues of M are negative.

Basics of linear algebra- Definite matrix

Properties:

(v) (square root matrix) M is psd iff $M = B^T B$ for some $B \in \boldsymbol{M}_n(\mathbb{R})$. Moreover, B is invertible when M is pd.

We may write $B = M^{\frac{1}{2}} = \sqrt{M}$.

(vi) For $A \in \boldsymbol{M}_{m\times n}(\mathbb{R})$, $A^T A \in \boldsymbol{M}_n(\mathbb{R})$ and $AA^T \in \boldsymbol{M}_m(\mathbb{R})$ are psd. Moreover, $A^T A$ and AA^T are pd, if A has full rank.

Questions

- How to find the eigenvalues and eigenvectors of a matrix $A \in \boldsymbol{M}_n(\mathbb{R})$ effiently?

Basics of linear algebra - Subspace

- For a given matrix $A = (a_{ij}) \in \boldsymbol{M}_{mxn}(\mathbb{R})$, we have

(Column space of A)

$C(A) = \{\boldsymbol{u} \in R^m \mid \boldsymbol{u} = A\boldsymbol{x}$ for some $\boldsymbol{x} \in \mathbb{R}^n\} \subset \mathbb{R}^m$

(Row space of A)

$R(A) = C(A^T) = \{\boldsymbol{v} \in R^n \mid \boldsymbol{v} = A^T\boldsymbol{y}$ for some $\boldsymbol{y} \in \mathbb{R}^m\} \subset \mathbb{R}^n$

(Null space of A)

$N(A) = \{\boldsymbol{w} \in R^n \mid A\boldsymbol{w} = 0\} \subset \mathbb{R}^n$

(Null space of A^T)

$N(A^T) = \{\boldsymbol{s} \in R^m \mid A^T\boldsymbol{s} = 0\} \subset \mathbb{R}^m$

Basics of linear algebra - Subspaces

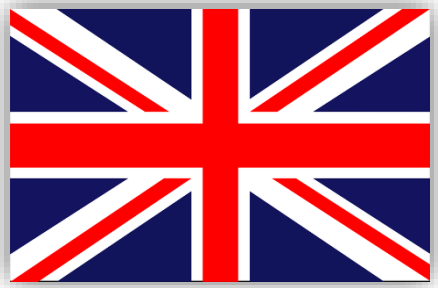
- For a given matrix $A = (a_{ij}) \in \boldsymbol{M}_{mxn}(\mathbb{R})$,
- (Orthogonality) $R(A) \perp N(\mathrm{A})$ and $C(A) \perp N(A^T)$
- (Dimensionality) $\dim(R(A)) = \dim(C(A))$
- (Rank) $rank(A) \triangleq \dim\big(R(A)\big) = \dim(C(A)) \leq min\{m, n\}$
- (Complementarity) Assume $rank(A) = r$, then $\dim\big(N(A)\big) = n - r$ and $\dim(N(A^T)) = m - r$.

Questions:

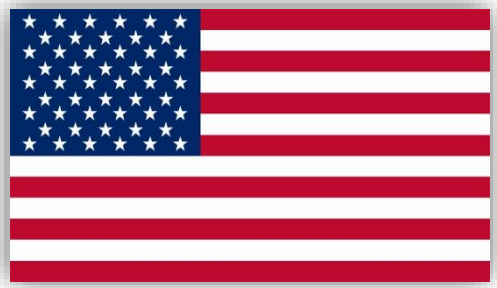
- How to calculate $rank(A)$?
- What's the geometric meaning of $rank(A)$?
- Any implications (complexity of image)?
- What's about "$rank$-*one*" matrix and "$rank$-k" matrix?

*** Questions:

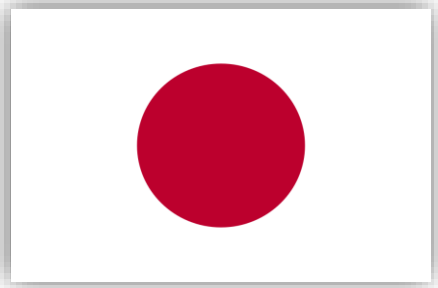
Finland



United Kingdom



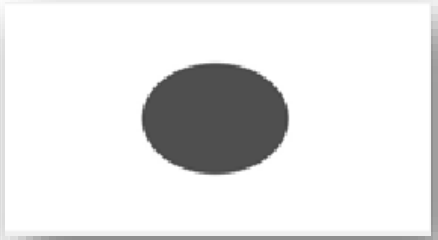
United States



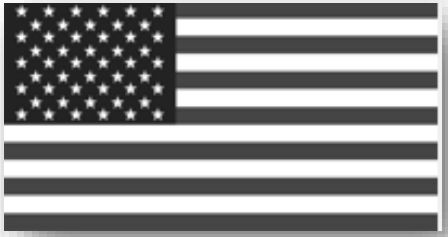
Japan

Convert the truecolor images RGB to the grayscale images and store them into 100×200 matrices. Compare the rank:

Rank = 9



Rank = 37



Rank = 60

Rank = 85

Basics of linear algebra – Matrix decomposition

- (orthonormality) A set of m n-vectors form an orthonormal list if all vectors in the set are mutually orthogonal and all of unit length.

  1. All vectors in an orthonormal list are linearly independent.
  2. (Gram-Schmidt theorem) For a set of linearly independent vectors, there exists an orthonormal list of the same number of vectors sharing the same span.
  3. Orthonormal basis is an orthonormal list of basis vectors of a space.

- For an $m \times n$ matrix A, the following statements are equivalent:

  (a) $A^T A = I_n$; (b) $||A\boldsymbol{x}|| = ||\boldsymbol{x}|| \; \forall \; \boldsymbol{x} \in \mathbb{R}^n$;

  (c) $<A\boldsymbol{x}, A\boldsymbol{y}> = <\boldsymbol{x}, \boldsymbol{y}> \; \forall \; \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$;

  (d) The column vectors of A are orthonormal.

Basics of linear algebra – Matrix decomposition

(orthogonal matrix) a square matrix Q is an orthogonal matrix if $Q^T = Q^{-1}$.

1. The transpose of an orthogonal matrix is orthogonal.
2. The inverse of an orthogonal matrix is orthogonal.
3. A product of orthogonal matrices is orthogonal.
4. $\det(Q) = 1\ or\ -1$ for an orthogonal matrix Q.

- For a square matrix *A*, the following statements are equivalent:

  (a) *A* is orthogonal; (b) $||A\boldsymbol{x}|| = ||\boldsymbol{x}||, \forall\ \boldsymbol{x} \in \mathbb{R}^n$;

  (c) $<A\boldsymbol{x}, A\boldsymbol{y}> = <\boldsymbol{x}, \boldsymbol{y}>, \forall\ \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$;

  (d) The column vectors of *A* are orthonormal;

  (e) The row vectors of *A* are orthonormal.

Basics of linear algebra - QR decomposition

- (Square matrix) Any square matrix $A \in \boldsymbol{M}_n(\mathbb{R})$ may be decomposed as $A = QR$, where Q is $n \times n$ orthogonal and R is $n \times n$ upper triangular. If A is invertible, then the factorization is unique if we require all diagonal entries of R to be positive.

- (Rectangular matrix) If $A \in \boldsymbol{M}_{m \times n}(\mathbb{R})$, then $A = QR$, where Q is an $m \times n$ matrix with column vectors forming an orthonormal list and R is an $n \times n$ upper-triangular matrix. If A is of full rank and we require the diagonal entries of R to be positive, then the factorization is unique and R is the same as that in the Cholesky factorization of $A^T \mathrm{A}$.

Questions

- How to perform QR decomposition?
- What's the use of QR factorization?

Basics of linear algebra – Eigen-decomposition

- Given a square matrix $A \in \boldsymbol{M}_n(\mathbb{R})$ with n linearly independent eigenvectors $\{\boldsymbol{q}_i\}$ corresponding to eigenvalues $\{\lambda_i\}$, $i = 1,..,n$, it can be factorized as

$$A = Q\Lambda Q^{-1},$$

where $\mathrm{Q} \in \boldsymbol{M}_n(\mathbb{R})$ with $\boldsymbol{q}_i$ being the i^{th} column, $\Lambda \in \boldsymbol{M}_n(\mathbb{R})$ is diagonal with $\Lambda_{ii} = \lambda_i$. The eigenvectors $\boldsymbol{q}_i$ are usually normalized, but they need not be.

- If $A \in \boldsymbol{S}_n(\mathbb{R})$ is a symmetric square real matrix, then all $\{\lambda_i\}$ are real and $\{\boldsymbol{q}_i\}$ can be chosen orthonormal. In this case, $A = Q\Lambda Q^T$ where Q is an orthogonal matrix.

Questions

- How to perform eigen-decomposition?
- What’s the use of eigen-decomposition?

Basics of linear algebra – Singular value & SVD

- Given a matrix $A \in \boldsymbol{M}_{m\times n}(\mathbb{R})$, the singular value decomposition (SVD) of matrix A is a factorization of the form $A = U\Sigma V^T$, where $U \in \boldsymbol{M}_m(\mathbb{R})$ is orthogonal, $\Sigma \in M_{m\times n}$ is a "rectangular diagonal" matrix with nonnegative real numbers $\{\sigma_i\}$ (called singular values) on the diagonal, and $V \in \boldsymbol{M}_n(\mathbb{R})$ is orthogonal.

- Since $AA^T \in \boldsymbol{S}_m(\mathbb{R})$ and $A^TA \in \boldsymbol{S}_n(\mathbb{R})$, we use (normalized) eigenvectors $\{\boldsymbol{u}_i\}$ of AA^T as the columns of U and (normalized) eigenvectors $\{\boldsymbol{v}_i\}$ of A^TA as columns of V.

Basics of linear algebra – Singular value & SVD

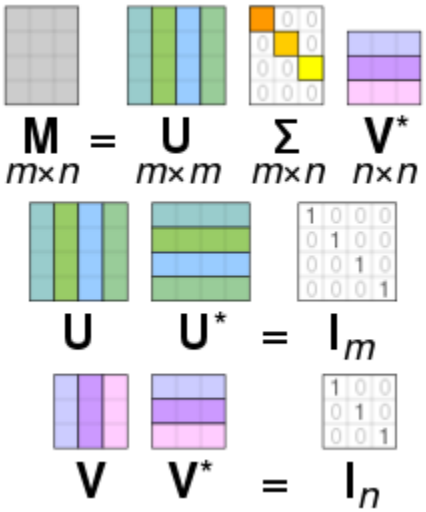
- When $m = n$ and A is symmetric, we know $U = V$ and

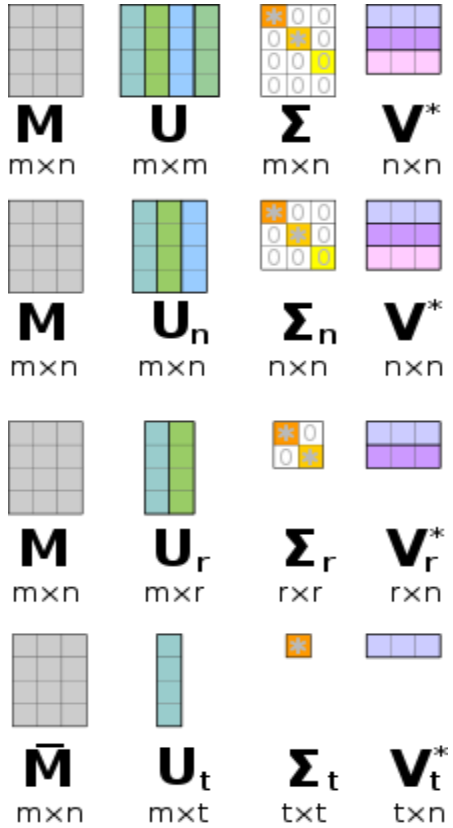
$$\mathrm{A}^2 = AA^T = (U\Sigma V^T)(V\Sigma U^T) = \mathrm{U}\Sigma^2 U^T$$

Eigen-decomposition says $A^2 = \mathrm{Q}\Lambda Q^T$. Knowing that $Q = U$ in this case, we have $\Sigma^2 = \Lambda$. Hence $\sigma_i^2 = \lambda_i$, where $\{\lambda_i\}$ are non-negative eigenvalues of matrix $A^{\mathrm{T}}A$.

Basics of linear algebra – SVD forms

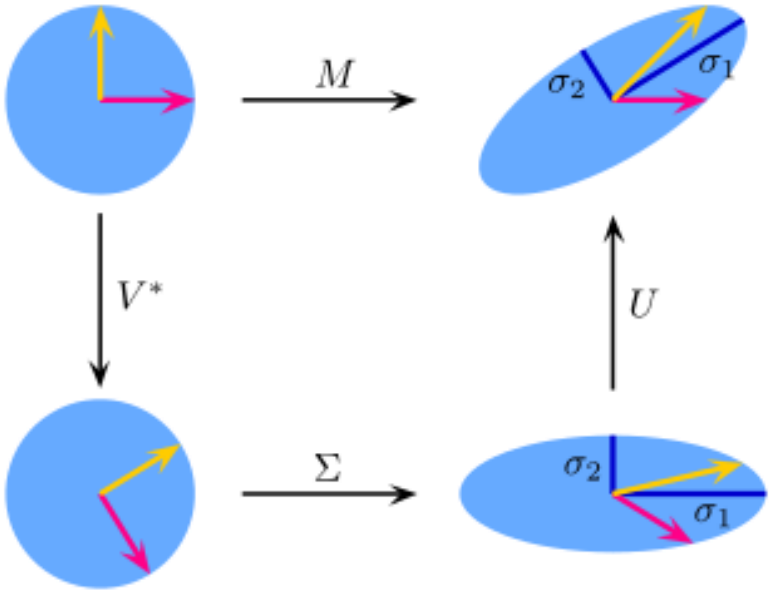
- (From https://en.wikipedia.org/wiki/Singular_value_decomposition)
- Various compact forms of SVD:





Basics of linear algebra – Geometric meaning of SVD

- (From https://en.wikipedia.org/wiki/Singular_value_decomposition)
- Meaning of SVD



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

Questions

1. How to compute SVD?

2. What's the use of SVD?
   - Low rank matrix approximation
   - New meaning of matrix norms
   - Principal component analysis

Basics of linear algebra – Low-rank approximation

(Eckart-Young-Mirsky Theorem)

- Given a matrix $A \in \boldsymbol{M}_{m\times n}(\mathbb{R}), \text{assuming } m \geq n$, SVD leads to $A = U\Sigma V^T$, where $U \in \boldsymbol{M}_m(\mathbb{R}), V \in \boldsymbol{M}_n(\mathbb{R})$ are orthogonal, $\Sigma \in M_{m\times n}(\mathbb{R})$ is rectangular diagonal with $\sigma_1 \geq \sigma_2 \geq \cdots \sigma_n \geq 0\ (= \sigma_{n+1})$.

  Define a matrix

  $$A_k = \sum_{i=1}^{k} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T \text{ , for } k = 1,\ldots,n.$$

  Then A_k is the closest "$rank$-k" approximation of matrix A in terms of the $||\cdot||_2$ norm. In fact,

  $$||A - A_k||_2 = \sigma_{k+1}, \text{ for } k = 1, \ldots, n.$$

Basics of linear algebra – Matrix norm

Matrix norm in terms of singular values:

- Let matrix $A \in \boldsymbol{M}_{m\times n}(\mathbb{R})$. Assuming $rank(A) = r \leq \min(m,n)$ with singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r \geq 0$.

  Recall that some commonly used matrix norms:

  (i) Frobenius norm

$$||A||_F = \sqrt{\Sigma_{i=1}^{m}\Sigma_{j=1}^{n} a_{ij}^2} = \sqrt{tr(A^T A)}$$

  (ii) matrix p-norm $(p \geq 1)$

$$||A||_p = sup_{\boldsymbol{x}\neq 0} \frac{||A\boldsymbol{x}||_p}{||\boldsymbol{x}||_p} = max_{||\boldsymbol{x}||_p=1} ||A\boldsymbol{x}||_p$$

Basics of linear algebra – Matrix norm

Now we have

- (i) Frobenius norm

$$||A||_F = \sqrt{\Sigma_{i=1}^{m}\Sigma_{j=1}^{n} a_{ij}^2} = \sqrt{tr(A^T A)} = \sqrt{\Sigma_{i=1}^{r}\sigma_i^2}$$

- (ii) matrix p-norm ($p \geq 1$)

$$||A||_p = max_{\boldsymbol{x}\neq 0} \frac{||A\boldsymbol{x}||_p}{||\boldsymbol{x}||_p} = max_{||\boldsymbol{x}||_p=1} ||A\boldsymbol{x}||_p$$

In particular,

$$||A||_2 = max_{\boldsymbol{x}\neq 0} \frac{||A\boldsymbol{x}||_2}{||\boldsymbol{x}||_2} = \sigma_1 \text{ (similarly } min_{\boldsymbol{x}\neq 0} \frac{||A\boldsymbol{x}||_2}{||\boldsymbol{x}||_2} = \sigma_r)$$

(iii) Nuclear norm

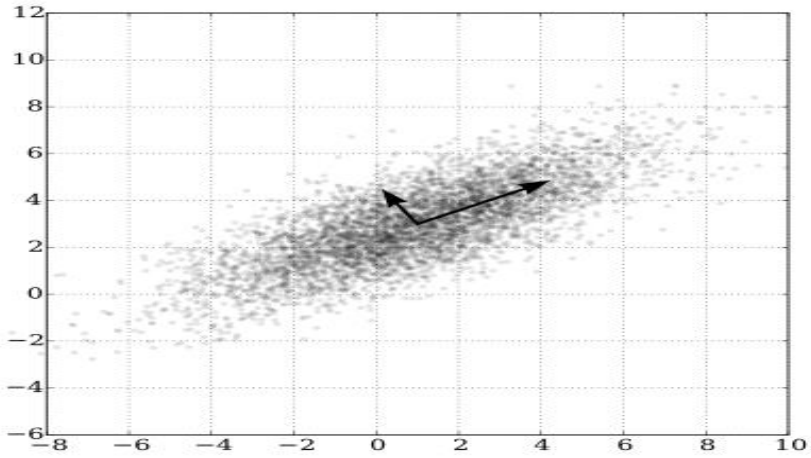
$$||A||_* = \Sigma_{i=1}^{r}\sigma_i$$

Basics of linear algebra - PCA

(https://en.wikipedia.org/wiki/Principal_component_analysis#:~:text=Principal%20component%20analysis%20(PCA)%20is,components%20and%20ignoring%20the%20rest.)

Principal component analysis (**PCA**) is the process of computing the principal components and using them to perform a change of basis on the data, sometimes using only the first few principal components and ignoring the rest. The 1st principal component intends to explain the most variance.

The 2nd principal component explains the most variance in what is left once the effect of the first component is removed



Basics of linear algebra - PCA

- (basic ideas)

  $A =$ data matrix (shift sample mean of each column to 0)

  $A^T A$ = (recognized as/proportional to) covariance matrix

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

  $A^T A = V\Sigma^T U^T U\Sigma^T V^T = V\Sigma^T \Sigma V^T = V\Sigma^2 V^T$

  T = score matrix = $AV = U\Sigma V^T V = U\Sigma$

  Hence each column of T is given by one of the left singular vectors of A multiplied by the corresponding singular value.

  This form is also the polar decomposition of T.

  * *A projected to V represented by U and Σ*

  $(principal\ components)$.

Questions

- 1. How to conduct PCA for practical application?
- 2. Properties and limitations of PCA?

*** Questions

- How to conduct PCA?

Data matrix $A = \begin{pmatrix} \boldsymbol{x}_1^T \\ \vdots \\ \boldsymbol{x}_m^T \end{pmatrix} = \begin{pmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mn} \end{pmatrix}$.

1. Centralization, $\boldsymbol{\mu} = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{x}_i$, $\bar{A} = \begin{pmatrix} \boldsymbol{x}_1^T - \boldsymbol{\mu}^T \\ \vdots \\ \boldsymbol{x}_m^T - \boldsymbol{\mu}^T \end{pmatrix}$.
2. Computing eigenvalues and eigenvectors

   Method 1: Eigen decomposition on the covariance matrix $\bar{A}^T\bar{A}$.

   Method 2: SVD on $\bar{A}$.

   $$V \in \mathbb{R}^{n\times n},\ \Sigma_{nxn} = diag(\sigma_1, \dots, \sigma_n)$$
3. Select the number of principal components (e.g., Fraction-of-variance-explained, FVE)

   $$FVE_k = \frac{\sum_{i=1}^{k}\sigma_i^2}{\sum_{i=1}^{n}\sigma_i^2} = \frac{\sum_{i=1}^{k}\sigma_i^2}{\|\bar{A}\|_F^2} > FVE_{threshold}$$
4. Compute the PC-scores

$$V^{(k)} \in \mathbb{R}^{n\times k}, \text{first } k \text{ eigenvectors, and } S = \bar{A}V^{(k)}$$

Prof. Gilbert Strang's comment

A number of important matrix decompositions (Golub & Van Loan 1996) involve orthogonal matrices, including especially:

QR decomposition

$M = QR$, Q orthogonal, R upper triangular

Singular value decomposition

$M = U\Sigma V^{\mathrm{T}}$, U and V orthogonal, Σ diagonal matrix

Eigendecomposition of a symmetric matrix (decomposition according to the spectral theorem)

$S = Q\Lambda Q^{\mathrm{T}}$, S symmetric, Q orthogonal, Λ diagonal

Polar decomposition

$M = QS$, Q orthogonal, S symmetric positive-semidefinite

Reference Books

1. Gilbert Strang, Introduction to Linear Algebra, 5th Edition, 2016, Wellesley-Cambridge Press, ISBN-13: 978-0980232776

2. Gilbert Strang, Linear Algebra and Learning from Data, First Edition, 2019, Wellesley Cambridge Press, ISBN-13: 978-0692196380

3. A. Messac, Optimization in Practice with Matlab, Cambridge Univ. Press, ISBN: 9781107109186, 2015.

Software platform for linear algebra

- MATLAB functions:
  - $rank(A)$, $\det(A)$, $tr(A)$, $norm(\boldsymbol{x})$, $norm(A)$,
    QR, Eigen-decomposition, SVD, PCA, …

- Python functions:
  - Libraries: $pandas,\ numpy,\ numpy.linalg,\ \ldots$

Software platform for linear algebra

- MATLAB
  - Generate a random matrix A.

```
rng('default')
% A = magic(3);              % A is 3x3 square
m = 3; n= 3;
A = randi([1 10],m,n);       % A is mxn, and a_ij \in [1, 10]
```

```
A =

     9    10     3
    10     7     6
     2     1    10
```

  - How to calculate $\mathrm{rank}(A)$?
    - Gaussian elimination

      LU factorization: $PA = LU$, an upper triangular matrix U, a lower triangular matrix L, and a permutation matrix P.

      These matrices describe the steps needed to perform Gaussian elimination on the matrix until it is in reduced row echelon form. The L matrix contains all of the multipliers, and the permutation matrix P accounts for row interchanges.

```
%% Gaussian elimination
[L,U,P] = lu(A);
rank_A_LU = sum(diag(U)~=0);          % alternative: nnz(diag(U));
fprintf('By LU factorization, the rank of A is: %d\n', rank_A_LU);
```

```
By LU factorization, the rank of A is: 3
```

```
P =

     0     1     0
     1     0     0
     0     0     1
```

```
L =

    1.0000         0         0
    0.9000    1.0000         0
    0.2000   -0.1081    1.0000
```

```
U =

   10.0000    7.0000    6.0000
         0    3.7000   -2.4000
         0         0    8.5405
```

Software platform for linear algebra

- MATLAB
  - How to calculate $\text{rank}(A)$?
    - QR factorization

      $A = QR$, an orthogonal matrix Q and an upper triangular matrix R.

      $\text{rank}(A) = \sum_i \mathbb{I}(r_{ii} \neq 0)$

```
A =

     9    10     3
    10     7     6
     2     1    10
```

```
%% QR factorization
[Q,R] = qr(A);
rank_A_QR = nnz(diag(R));
fprintf('By QR factorization, the rank of A is: %d\n', rank_A_QR);
```

```
By QR factorization, the rank of A is: 3
```

```
Q =

   -0.6617    0.7427    0.1031
   -0.7352   -0.6157   -0.2835
   -0.1470   -0.2633    0.9534
```

```
R =

  -13.6015  -11.9105   -7.8668
         0    2.8532   -4.0998
         0         0    8.1428
```

Software platform for linear algebra

```
A =

     9    10     3
    10     7     6
     2     1    10
```

- MATLAB
  - How to calculate $\mathrm{rank}(A)$?
    - Eigen decomposition (if A is square)

      $AV = VD$, diagonal matrix D of eigenvalues, and matrix V whose columns are the corresponding right eigenvectors.

$$D = \begin{bmatrix} \lambda_1 & & \\ & \lambda_2 & \\ & & \lambda_3 \end{bmatrix}, A\boldsymbol{V}_i = \lambda_i \boldsymbol{V}_i, i = 1,2,3.\ rank(A) = \sum_i \mathbb{I}(\lambda_i \neq 0)$$

```
%% Eigen decomposition
[V,D] = eig(A);
rank_A_Eig = nnz(diag(D));
fprintf('By eigendecomposition, the rank of A is: %d\n', rank_A_Eig);
```

```
By eigendecomposition, the rank of A is: 3
```

```
V =

    0.6823   -0.7062   -0.5253
   -0.7291   -0.6728   -0.2182
   -0.0533   -0.2204    0.8225
```

```
D =

   -1.9198         0         0
         0   19.4625         0
         0         0    8.4573
```

Software platform for linear algebra

- MATLAB
  - How to calculate $\text{rank}(A)$?
    - MATLAB function: 'rank',

```
>> rank(A)

ans =

     3
```

rank uses a method based on the singular value decomposition, or SVD. The SVD algorithm is more time consuming than some alternatives, but it is also the most reliable.

- SVD

$A = USV^T$, a diagonal matrix S of the same dimension as A and with singular value as diagonal elements in decreasing order, and unitary matrices U and V.

$$rank(A) = \sum_i \mathbb{I}(\sigma_i \neq 0)$$

```
%% SVD
[U,S,V] = svd(A);
rank_A_SVD = nnz(diag(S));
fprintf('By SVD, the rank of A is: %d\n', rank_A_SVD);
```

```
By SVD, the rank of A is: 3
```

```
U =

   -0.6601   -0.4303   -0.6158
   -0.6758   -0.0177    0.7368
   -0.3279    0.9025   -0.2791
```

```
S =

   20.0265         0         0
         0    8.6975         0
         0         0    1.8142
```

```
V =

   -0.6669   -0.2580    0.6991
   -0.5822   -0.4052   -0.7049
   -0.4651    0.8771   -0.1200
```

Software platform for linear algebra

- MATLAB
  - How to calculate $\det(A)$?
    - MATLAB function:

      'det' uses the LU decomposition to calculate the determinant

```
A =

     9    10     3
    10     7     6
     2     1    10
```

```
det_A = det(A);  % returns the determinant of square matrix A.
```

```
det_A =

  -316
```

    - QR factorization: $\det(A) = \prod_i r_{ii}$

```
[Q,R] = qr(A);
det_A_QR = prod(diag(R));
fprintf('By QR factorization, the det of A is: %f\n', det_A_QR);
```

```
By QR factorization, the det of A is: -316.000000
```

    - Eigen decomposition (if A is square) : $\det(A) = \prod_i \lambda_i$

```
[V,D] = eig(A);
det_A_Eig = prod(diag(D));
fprintf('By eigendecomposition, the det of A is: %f\n', det_A_Eig);
```

```
By eigendecomposition, the det of A is: -316.000000
```

Software platform for linear algebra

- MATLAB
  - How to calculate $\mathrm{trace}(A)$?
    - MATLAB function:

      'trace' calculates the sum of the diagonal elements of matrix A.

$$\mathrm{trace}(A) = \sum_i a_{ii}$$

    - Eigen decomposition (if A is square) :

$$\mathrm{trace}(A) = \sum_i \lambda_i$$

```
A =

     9    10     3
    10     7     6
     2     1    10
```

```
%% Trace
tr_Def = sum(diag(A));
tr_Mat = trace(A);
tr_Eig = sum(diag(D));
```

```
tr_Def =

    26


tr_Mat =

    26


tr_Eig =

   26.0000
```

Software platform for linear algebra

- MATLAB
  - How to calculate norm of a vector, i.e., $\mathrm{norm}(\boldsymbol{x}), \boldsymbol{x} \in \mathbb{R}^n$?

```
n = 3;                    % dimension of vectors
rng(200)                  % Get the current generator settings
x = randi([-5,5],n,1);    % RANDI generates some values [-5,5]
```

```
x =

     5
    -3
     1
```

```
% Define p-norm function
p = 1; % P>=1
norm_def = sum(abs(x).^p).^(1/p);
fprintf('The %d-norm of x is %f. \n', p,norm_def);
fprintf('The inf-norm of x is %f. \n', max(abs(x)));

% Matlab function: norm
p = 1; % P>=1
norm_Mat = norm(x,p);
fprintf('The %d-norm of x is %f. \n', p,norm_Mat);
fprintf('The inf-norm of x is %f. \n', norm(x,inf));
```

```
The 1-norm of x is 9.000000.
The 2-norm of x is 5.916080.
The inf-norm of x is 5.000000.
```

Software platform for linear algebra

- MATLAB
  - How to calculate norm of a matrix, i.e., $\mathrm{norm}(A), A \in \mathbb{R}^{n \times n}$?

```
A =

     9    10     3
    10     7     6
     2     1    10
```

```
% 1-norm
fprintf('The 1-norm of A is %f. \n', max(vecnorm(A,1,1)));    % maximum absolute column sum of A
fprintf('The 1-norm of A is %f. \n', norm(A,1));              % Matlab function 'norm'
```

```
The 1-norm of A is 21.000000.
The 1-norm of A is 21.000000.
```

```
% inf-norm
sprintf('The inf-norm of A is %f. \n', max(vecnorm(A,1,2)));% maximum absolute row sum of A
fprintf('The inf-norm of A is %f. \n', norm(A,inf));         % Matlab function 'norm'
```

```
The inf-norm of A is 23.000000.
The inf-norm of A is 23.000000.
```

```
% 2-norm
fprintf('The 2-norm of A is %f. \n', norm(A,2));     % Matlab function 'norm'
fprintf('The 2-norm of A is %f. \n', max(svd(A)));   % SVD singular value, ||A||_2 = sigma_max
```

```
The 2-norm of A is 20.026533.
The 2-norm of A is 20.026533.
```

```
% Frobenious-norm
fprintf('The F-norm of A is %f. \n', sqrt(sum(sum(A.^2))));% sum of all elements square
fprintf('The F-norm of A is %f. \n', norm(A,'fro'));       % Matlab function 'norm'
fprintf('The F-norm of A is %f. \n', sqrt(trace(A'*A)));   % trace(A'A)
fprintf('The F-norm of A is %f. \n', norm(svd(A)));        % SVD singular value,||A||_F = ||sigma||_2
```

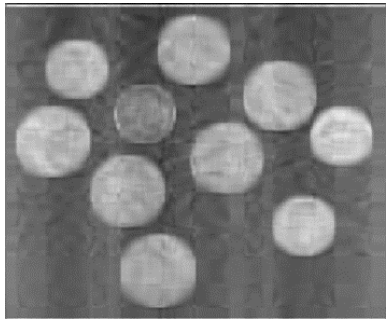
```
The F-norm of A is 21.908902.
The F-norm of A is 21.908902.
The F-norm of A is 21.908902.
The F-norm of A is 21.908902.
```

Software platform for linear algebra

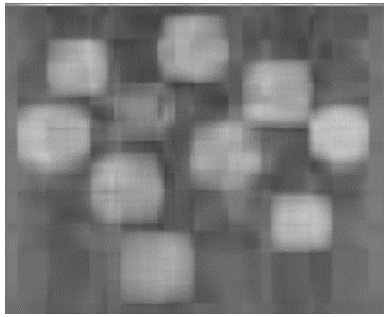
- MATLAB
  - How to conduct PCA? An example.

```
%% Read the orifinal picture
I = imread('coin.png');
X = rgb2gray(I);
figure
imshow(X)
%% 1. Centralization
A = double(X');
Abar = A-mean(A,2);
figure
G = uint8(255 * mat2gray(Abar));
imshow(G')
%% 2. Computing eigenvalues and eigenvectors
[U,S,V] = svd(Abar);
%% 3. Select the number of principal components
normF = norm(Abar,'fro');
D = S.^2;
FEV = cumsum(diag(D))/(normF ^2);
%% 4. Compute the PC-scores
k = find( FEV >= 0.85-(1e-6),1);
FEV_k = FEV(k);
Ak = U*S(:,1:k)*V(:,1:k)';
figure
G = uint8(255 * mat2gray(Ak));
imshow(G');
```

Centralized image
$n = 293$



$FVE_{threshold} = 0.95$
$k = 17$
$FVE_k = 0.9503$



$FVE_{threshold} = 0.85$
$k = 7$
$FVE_k = 0.8617$

Software platform for linear algebra

- Python:

```
import pandas as pd
import numpy as np
import scipy as sc
import numpy.linalg as linalg # SciPy Linear Algebra Library
import pprint
```

```
A = np.array([[9,10,3],[10,7,6],[2,1,10]])
print("A = \n", A)
```

```
A =
 [[ 9 10  3]
 [10  7  6]
 [ 2  1 10]]
```

  - Compute $\text{rank}(A)$ and $determinant(A)$

```
# %% Python function to compute Rank and Determinant
print('The rank of A is', linalg.matrix_rank(A))
print('The determinant of A is', linalg.det(A))
```

```
The rank of A is 3
The determinant of A is -316.00000000000006
```

Software platform for linear algebra

- Python:
  - Compute $\text{rank}(A)$ and $determinant(A)$
    - LU factorization

```
# %% LU factorization
P, L, U = linalg.lu(A)
print('P = \n', P)
print('L = \n', L)
print('U = \n', U)
print('The rank of A is', np.count_nonzero(np.diag(U)))
```

```
P =
 [[0. 1. 0.]
 [1. 0. 0.]
 [0. 0. 1.]]
L =
 [[ 1.          0.          0.        ]
 [ 0.9         1.          0.        ]
 [ 0.2        -0.10810811  1.        ]]
U =
 [[10.          7.          6.        ]
 [ 0.          3.7        -2.4       ]
 [ 0.          0.          8.54054054]]
The rank of A is 3
```

    - Eigen decomposition

```
# %% Eigen decomposition
D, V = linalg.eig(A)
print('D = \n', D)
print('V = \n', V)
print('The rank of A is', np.count_nonzero(D))
print('The determinant of A is', np.prod(D))
```

```
D =
 [-1.91980041 19.46247864  8.45732176]
V =
 [[ 0.68233125 -0.70624501 -0.52528359]
 [-0.72909603 -0.6727951  -0.21823394]
 [-0.05332023 -0.22037409  0.82246647]]
The rank of A is 3
The determinant of A is -316.00000000000057
```

Software platform for linear algebra

- Python:
  - SVD

```
# %% SVD
U, S, Vh = linalg.svd(A)
print('U = \n', U)
print('S = \n', S)
print('Vh = \n', Vh)
print('The rank of A is', np.count_nonzero(S))
```

```
U =
 [[-0.66008422 -0.4302736  -0.61575438]
 [-0.67584088 -0.01765875  0.73683599]
 [-0.32791452  0.90252579 -0.27914022]]
S =
 [20.02653326  8.69750672  1.81420573]
Vh =
 [[-0.66686508 -0.58220975 -0.46510512]
 [-0.25800477 -0.40515305  0.87708867]
 [ 0.69908833 -0.70489915 -0.11996953]]
The rank of A is 3
The determinant of A is -316.00000000000057
```

  - Matrix norm

```
# Matrix norm
print('The 1-norm of A is', linalg.norm(A,1))

print('\nThe 2-norm of A is', linalg.norm(A,2))
print('The 2-norm of A is', np.max(S))

print('\nThe inf-norm of A is', linalg.norm(A,np.inf))

print('\nThe F-norm of A is', linalg.norm(A,'fro'))
print('The F-norm of A is', np.sqrt(np.matrix.trace(A.T@A)))
print('The F-norm of A is', linalg.norm(S))
```

```
The 1-norm of A is 21.0

The 2-norm of A is 20.02653326077009
The 2-norm of A is 20.02653326077009

The inf-norm of A is 23.0

The F-norm of A is 21.908902300206645
The F-norm of A is 21.908902300206645
The F-norm of A is 21.908902300206638
```