CS434a/541a: Pattern Recognition
Prof. Olga Veksler

Lecture 2

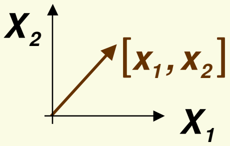
Outline

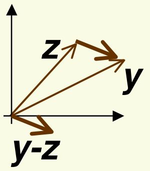
- Review of Linear Algebra
  - vectors and matrices
  - products and norms
  - vector spaces and linear transformations
  - eigenvalues and eigenvectors
- Introduction to Matlab

Why Linear Algebra?

- For each example (e.g. a fish image), we will extract a set of features (e.g. length, width, color)
- This set of features we will represent as a *feature vector*
  - [length, width, color,…]
- All collected examples will be represented as collection of (feature) vectors
  - [l_1, w_1 , c_1 , …], [l_2 , w_2 , c_2 ,…], [l_3 , w_3 , c_3 ,…], …
    example 1 example 2 example 3
- Besides representation, we will often use linear models since they are simple and computationally feasible

Vectors



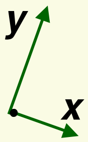


- n-dimensional row vector $\boldsymbol{x} = [x_1 \ x_2 \ \ldots \ x_n]$
- Transpose of row vector is column vector $\boldsymbol{x}^T = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$
- *Vector* product (or *inner* or *dot* product)

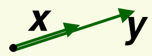
$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \boldsymbol{x} \cdot \boldsymbol{y} = \boldsymbol{x}^T \boldsymbol{y} = x_1 y_1 + x_2 y_2 + \ldots + x_n y_n = \sum_{i=1..k} x_i y_i$$

More on Vectors

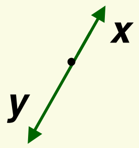
- *Euclidian norm* or *length* $\left|\boldsymbol{x}\right| = \sqrt{\langle \boldsymbol{x}, \boldsymbol{x} \rangle} = \sqrt{\sum_{i=1\ldots n} x_i^2}$
- If $|\boldsymbol{x}|=1$ we say $\boldsymbol{x}$ is *normalized* or *unit* length
- Angle θ between vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ $\cos\,\theta = \dfrac{\boldsymbol{x}^T \boldsymbol{y}}{|\boldsymbol{x}||\boldsymbol{y}|}$



$$\cos\theta = 0$$
$$x^T y = 0$$
x orthogonal to y
$$x \perp y$$



$$\cos\theta = 1$$
$$x^T y = |x||y| > 0$$



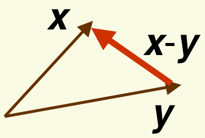
$$\cos\theta = -1$$
$$x^T y = -|x||y| < 0$$

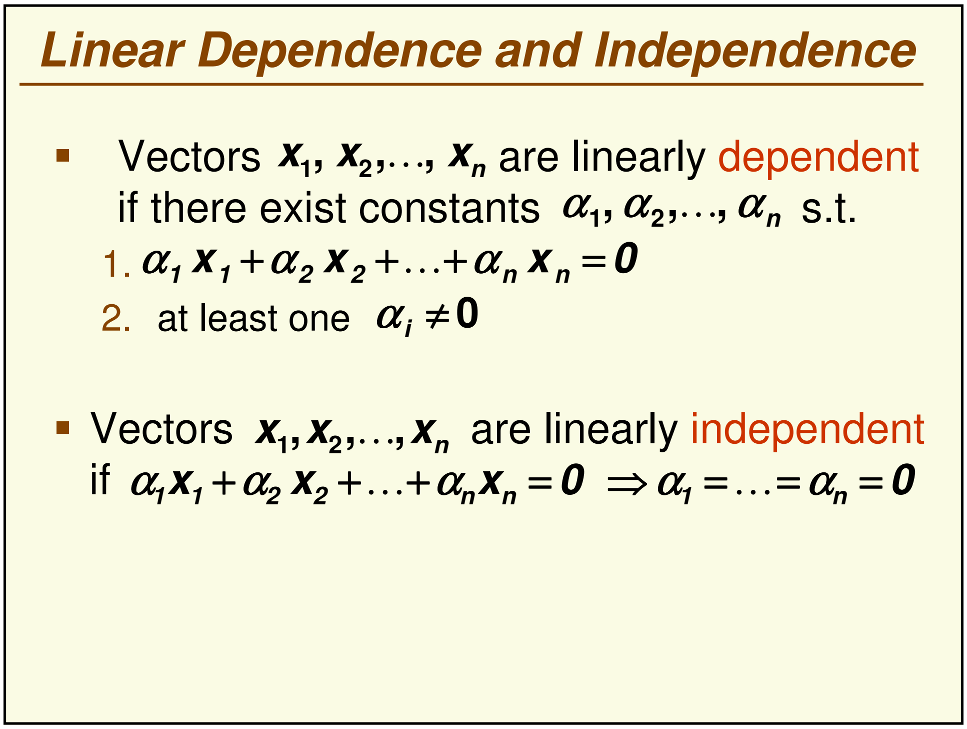
- Thus inner product captures direction relationship between $\boldsymbol{x}$ and $\boldsymbol{y}$

More on Vectors

- Vectors x and y are orthonormal if they are orthogonal and |x|=|y|=1
- Euclidian distance between vectors x and y

$$\left|\boldsymbol{x}-\boldsymbol{y}\right| = \sqrt{\sum_{i=1\ldots n} (x_i - y_i)^2}$$

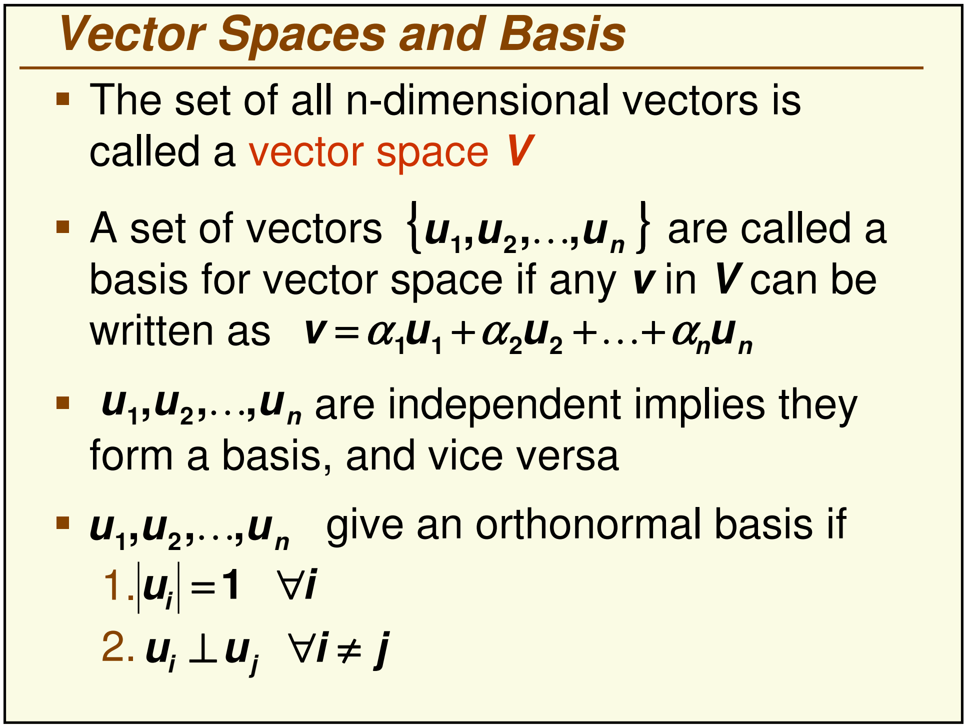




Linear Dependence and Independence

- Vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$ are linearly dependent if there exist constants $\alpha_1, \alpha_2, \ldots, \alpha_n$ s.t.
  1. $\alpha_1 \boldsymbol{x}_1 + \alpha_2 \boldsymbol{x}_2 + \ldots + \alpha_n \boldsymbol{x}_n = \boldsymbol{0}$
  2. at least one $\alpha_i \neq \mathbf{0}$

- Vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$ are linearly independent if $\alpha_1 \boldsymbol{x}_1 + \alpha_2 \boldsymbol{x}_2 + \ldots + \alpha_n \boldsymbol{x}_n = \boldsymbol{0} \Rightarrow \alpha_1 = \ldots = \alpha_n = \boldsymbol{0}$



Vector Spaces and Basis

- The set of all n-dimensional vectors is called a vector space $\boldsymbol{V}$

- A set of vectors $\{\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n\}$ are called a basis for vector space if any $\boldsymbol{v}$ in $\boldsymbol{V}$ can be written as $\boldsymbol{v} = \alpha_1 \boldsymbol{u}_1 + \alpha_2 \boldsymbol{u}_2 + \ldots + \alpha_n \boldsymbol{u}_n$

- $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n$ are independent implies they form a basis, and vice versa

- $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n$ give an orthonormal basis if
  1. $|\boldsymbol{u}_i| = \mathbf{1} \quad \forall \boldsymbol{i}$
  2. $\boldsymbol{u}_i \perp \boldsymbol{u}_j \quad \forall \boldsymbol{i} \neq \boldsymbol{j}$

Matrices

- n by m matrix A and its m by n transpose A^T

$$A=\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & \cdots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix} \qquad A^T=\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{n1} \\ x_{12} & x_{22} & \cdots & x_{n2} \\ \vdots & \vdots & \cdots & \vdots \\ x_{1m} & x_{2m} & \cdots & x_{nm} \end{bmatrix}$$

Matrix Product

$$AB=\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1d} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nd} \end{bmatrix}\begin{bmatrix} b_{11} & \cdots & b_{1m} \\ b_{21} & \cdots & b_{2m} \\ b_{31} & \cdots & b_{3m} \\ \vdots & \cdots & \vdots \\ b_{d1} & \cdots & b_{dm} \end{bmatrix}=\begin{bmatrix} & & \\ & c_{ij} & \\ & & \end{bmatrix}=C$$

$$c_{ij}=\left\langle a^i, b_j \right\rangle$$

a^i is row i of A
b_j is column j of B

- # of columns of A = # of rows of B
- even if defined, in general $AB \neq BA$

Matrices

- **Rank** of a matrix is the number of linearly independent rows (or equivalently columns)
- A square matrix is **non-singular** if its rank equal to the number of rows. If its rank is less than number of rows it is **singular**.
- **Identity matrix** $I = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$
  $AI=IA=A$
- Matrix A is **symmetric** if $A=A^T$ $\begin{bmatrix} 1 & 2 & 9 & 5 \\ 2 & 7 & 4 & 8 \\ 9 & 4 & 3 & 6 \\ 5 & 8 & 6 & 4 \end{bmatrix}$

Matrices

- Matrix A is **positive definite** if
$$x^T A x = \sum_{i,j} A_{i,j} x_i x_j > 0$$
- Matrix A is **positive semi-definite** if
$$x^T A x = \sum_{i,j} A_{i,j} x_i x_j \geq 0$$
- Trace of a square matrix A is sum on the elements on the diagonal
$$tr[A] = \sum_{i=1}^{n} a_{ii}$$

Matrices

- **Inverse** of a square matrix $\boldsymbol{A}$ is matrix $\boldsymbol{A}^{-1}$ s.t. $\boldsymbol{AA}^{-1} = \boldsymbol{I}$

- If $\boldsymbol{A}$ is singular or not square, inverse does not exist. Pseudo-inverse $\boldsymbol{A}^{\dagger}$ is defined whenever $\boldsymbol{A}^{T}\boldsymbol{A}$ is not singular (it is square)
  - $\boldsymbol{A}^{\dagger} = (\boldsymbol{A}^{T}\boldsymbol{A})^{-1}\boldsymbol{A}^{T}$
  - $\boldsymbol{A}^{\dagger}\boldsymbol{A} = (\boldsymbol{A}^{T}\boldsymbol{A})^{-1}\boldsymbol{A}^{T}\boldsymbol{A} = \boldsymbol{I}$

Matrices

- Determinant of n by n matrix $\boldsymbol{A}$ is

$$\det(A) = \sum_{k=1}^{n} (-1)^{k+i} a_{ik} \det(A_{ik})$$

  - Where $\boldsymbol{A}_{ik}$ obtained from $\boldsymbol{A}$ by removing the ith row and kth column

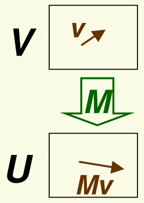
- Absolute value of determinant gives the volume of parallelepiped spanned by the matrix rows

$$\left\{\beta_1 a^1 + \beta_2 a^2 + \ldots + \beta_n a^n\right\}$$

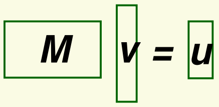
$$\beta_i \in [0,1] \ \forall i$$

Linear Transformations

- A linear transformation from vector space ***V*** to vector space ***U*** is a mapping which can be represented by a matrix ***M***:
  - ***u*** = ***Mv***
- If U and V have the same dimension, ***M*** is a square matrix



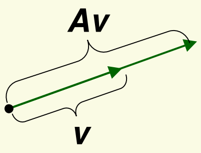
- In pattern recognition, often ***U*** has smaller dimensionality than ***V***, i.e. transformation ***M*** is used to reduce the number of features.

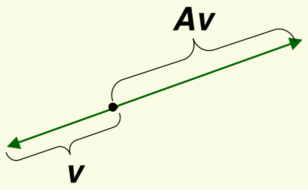


Eigenvectors and Eigenvalues

Note: A0=λ0 for any λ, not interesting

- Given n by n matrix ***A***, and nonzero vector ***x***. Suppose there is λ which satisfies $\boxed{\boldsymbol{Ax} = \lambda \boldsymbol{x}}$
  - ***x*** is called an eigenvector of ***A***
  - λ is called an eigenvalue of ***A***
- Linear transformation ***A*** maps an eigenvector ***v*** in a simple way. Magnitude changes by λ, direction
  - If $\lambda > 0$
  - If $\lambda < 0$

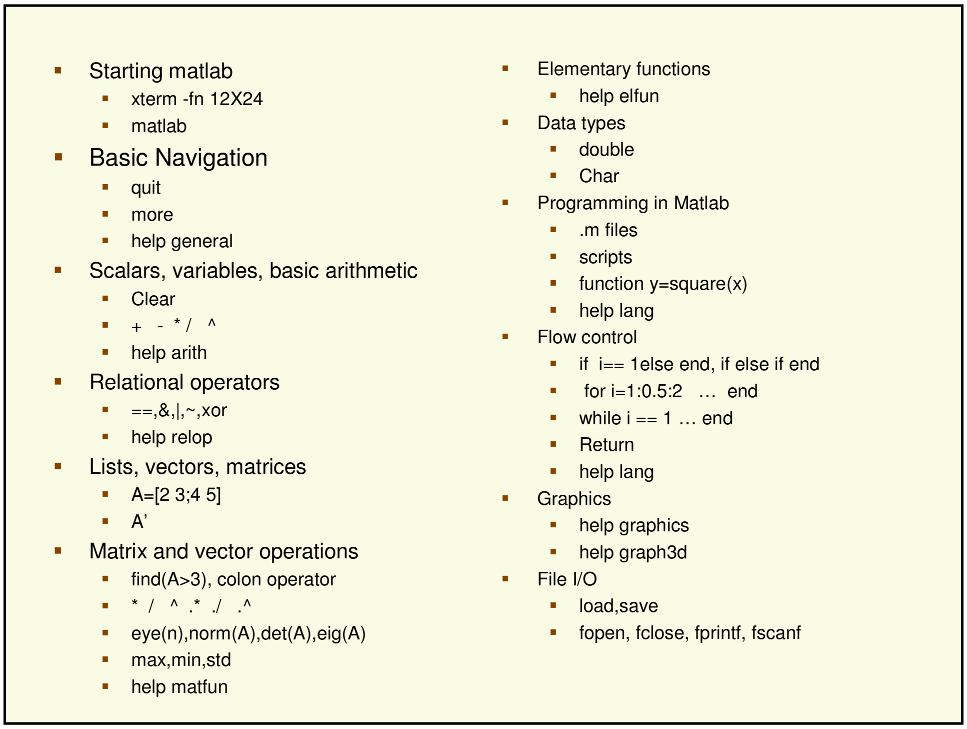




Eigenvectors and Eigenvalues

- If ***A*** is real and symmetric, then all eigenvalues are real (not complex)
- If ***A*** is non singular, all eigenvalues are non zero
- If ***A*** is positive definite, all eigenvalues are positive

MATLAB



- Starting matlab
  - xterm -fn 12X24
  - matlab
- Basic Navigation
  - quit
  - more
  - help general
- Scalars, variables, basic arithmetic
  - Clear
  - + - * / ^
  - help arith
- Relational operators
  - ==,&,|,~,xor
  - help relop
- Lists, vectors, matrices
  - A=[2 3;4 5]
  - A’
- Matrix and vector operations
  - find(A>3), colon operator
  - * / ^ .* ./ .^
  - eye(n),norm(A),det(A),eig(A)
  - max,min,std
  - help matfun
- Elementary functions
  - help elfun
- Data types
  - double
  - Char
- Programming in Matlab
  - .m files
  - scripts
  - function y=square(x)
  - help lang
- Flow control
  - if i== 1else end, if else if end
  - for i=1:0.5:2 … end
  - while i == 1 … end
  - Return
  - help lang
- Graphics
  - help graphics
  - help graph3d
- File I/O
  - load,save
  - fopen, fclose, fprintf, fscanf