Review of Linear Algebra

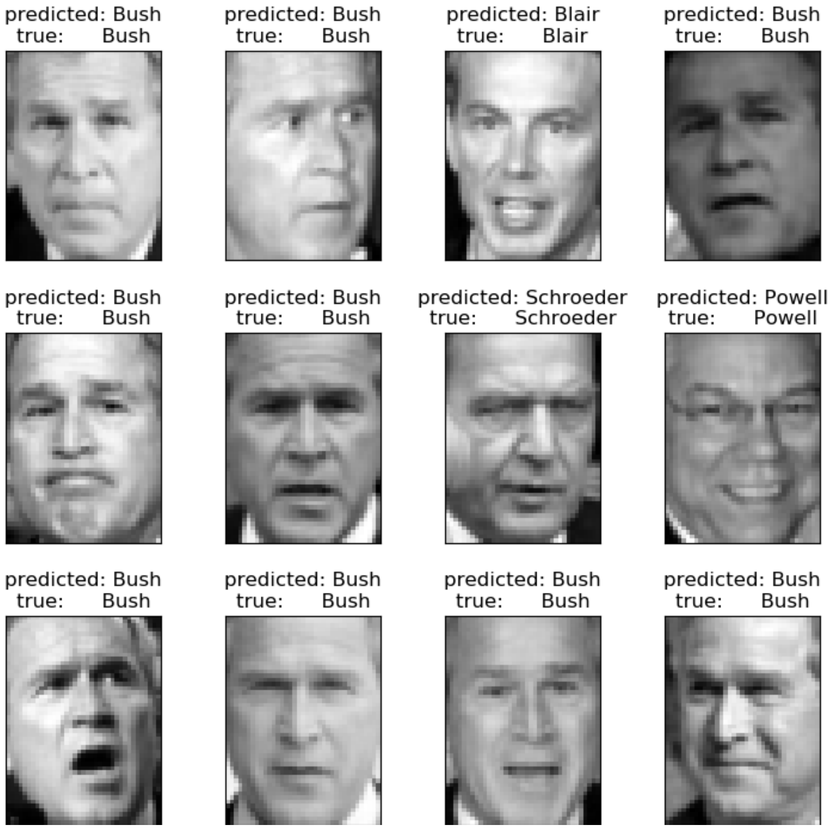
Liang Liang

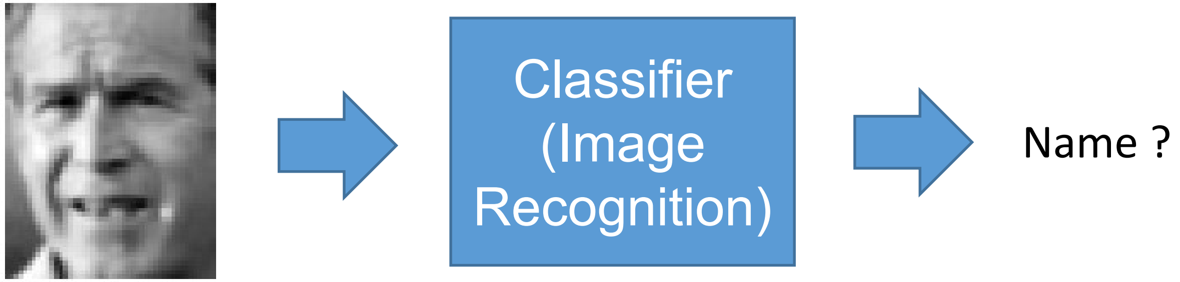
Introduction to Machine Learning with Applications

Outline: Linear Algebra

- Motivating Example - Eigenface
- Basics
- Dot and Vector Products
- Matrices
- Norms
- Rank and linear independence
- Range and Null Space
- Column and Row Space
- Determinant and inverse of a matrix
- Eigenvalues and Eigenvectors
- Singular Value Decomposition
- Matrix Calculus

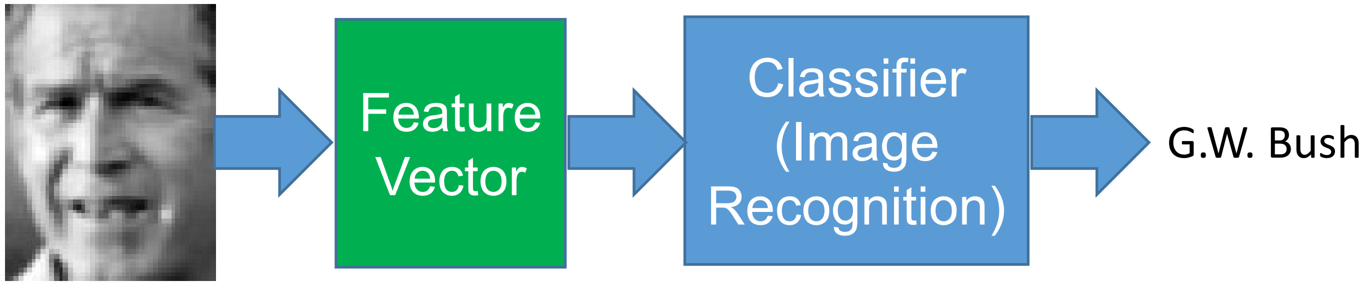
A Motivating Example - Eigenface





This image has 432x288 pixels
Not every pixel is equally important in classifying faces

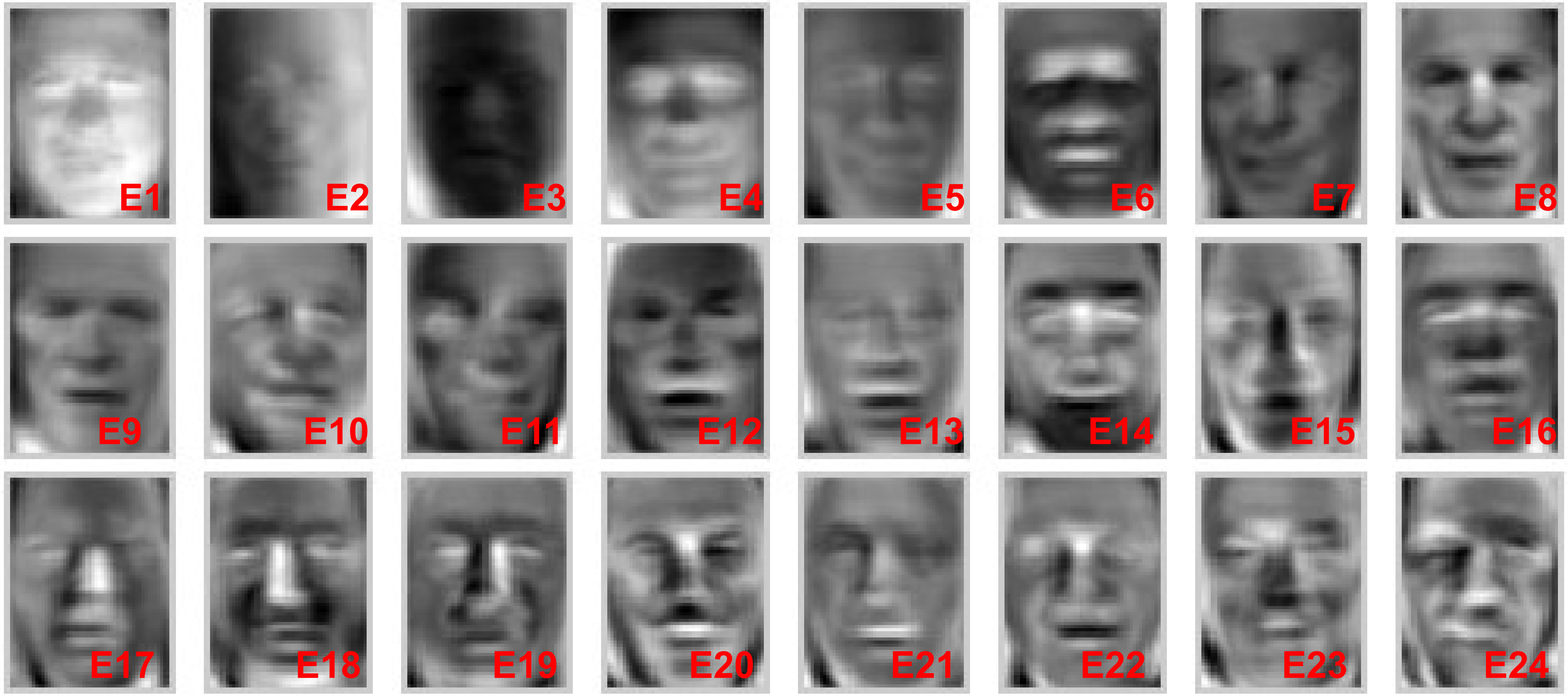
Solution: use ideas from linear algebra to extract features from images - using eigenvectors, eigenvalues



Eigenfaces (using linear algebra)

mean face

E0

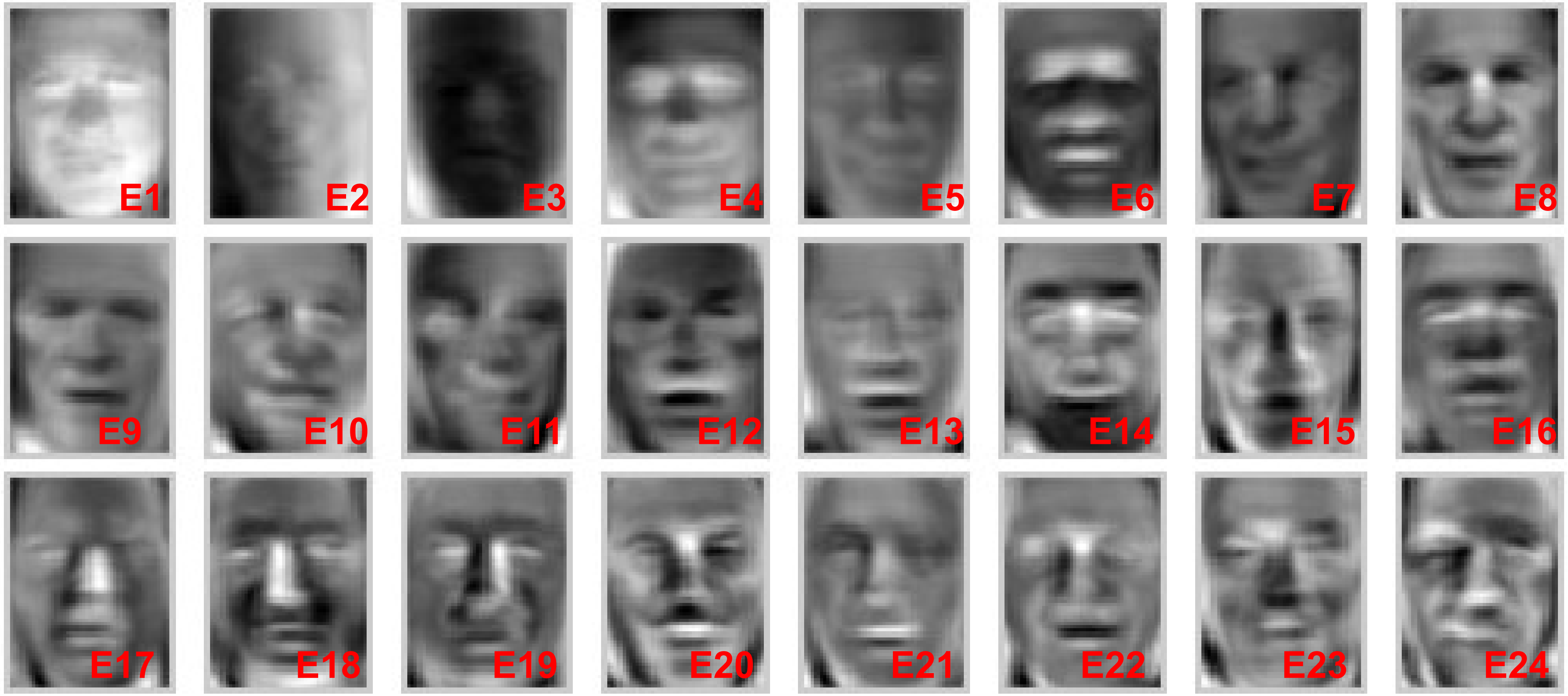


= **E0** + (-0.005)×**E1** + (-0.04)×**E2** + (0.002)×**E3** + …

Eigenfaces (using linear algebra)

mean face

E0



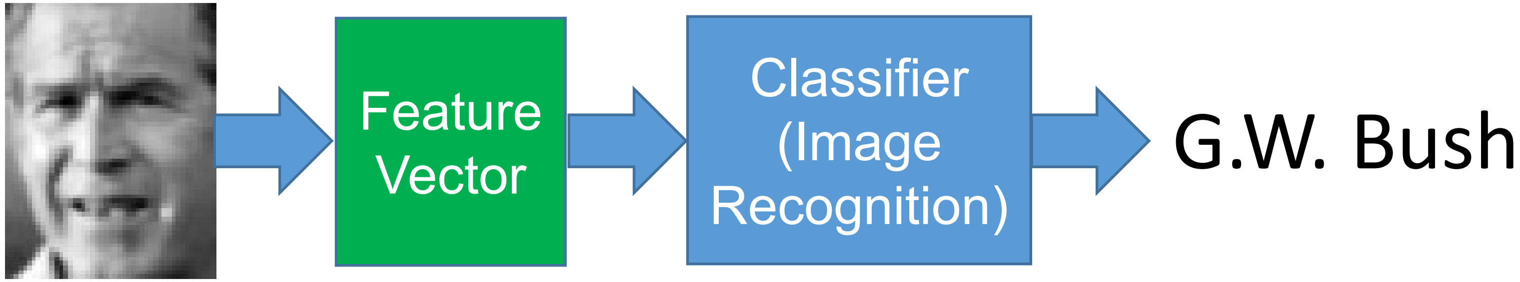
$$= \mathbf{E0} + (-0.015)\times\mathbf{E1} + (0.0037)\times\mathbf{E2} + (-0.01)\times\mathbf{E3} + \ldots$$

$$= E0 + (-0.015)\times E1 + (0.0037)\times E2 + (-0.01)\times E3 + \ldots$$

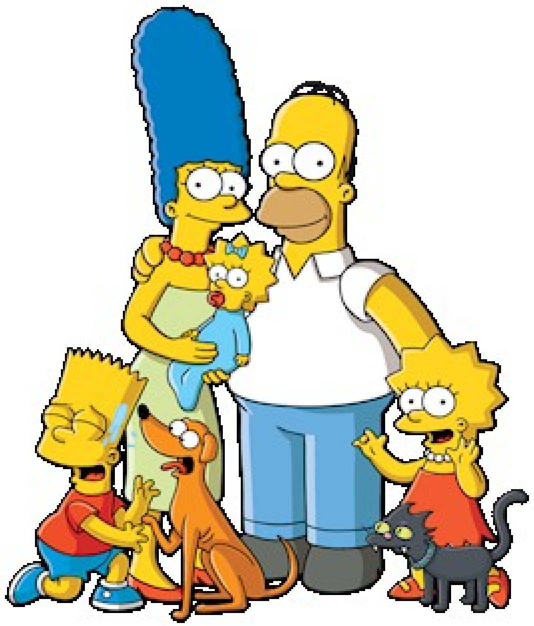
Feature Vector = [-0.015, 0.0037, -0.01, …]

$$= E0 + (-0.005)\times E1 + (-0.04)\times E2 + (0.002)\times E3 + \ldots$$

Feature Vector = [-0.005, -0.04, 0.002, …]



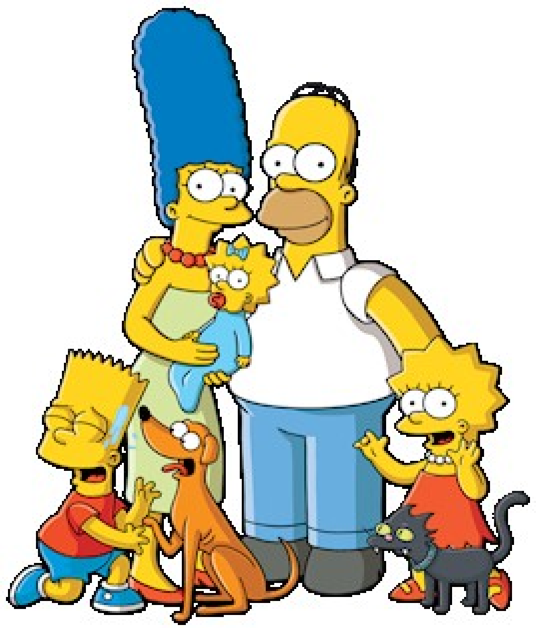
Use a Numerical Vector to represent an Object



$$\boldsymbol{x} = \begin{bmatrix} name \\ age \\ gender \\ height \\ weight \\ occupation \end{bmatrix}$$

"Feature Vector"

Use a **Numerical Vector** to represent an Object



$$\boldsymbol{x} = \begin{bmatrix} name \\ age \\ gender \\ height \\ weight \\ occupation \end{bmatrix}$$

$$\boldsymbol{x_1} = \begin{bmatrix} Homer\ J.Simpson \\ 35\ years \\ male \\ 175\ cm \\ 200\ Kg \\ safety\ inspector \end{bmatrix}$$

How can we convert this vector to a numerical vector ?
(e.g. represent the name using some number(s))

A challenging problem in Machine Learning:
How can we build a good representation of data/object

Representation Learning: A Review and New Perspectives

Yoshua Bengio[†], Aaron Courville, and Pascal Vincent[†]
Department of computer science and operations research, U. Montreal
† also, Canadian Institute for Advanced Research (CIFAR)

https://arxiv.org/pdf/1206.5538.pdf

Note: read this paper in the middle of this semester

Linear Algebra Basics

- Linear algebra provides a way of compactly representing and operating on sets of linear equations

  $4x_1 + 5x_2 = 1$

  $6x_1 + 7x_2 = 2$

  the above equations can be written in the form of $A\boldsymbol{x} = b$

  $A = \begin{bmatrix} 4 & 5 \\ 6 & 7 \end{bmatrix}$, $b = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$, $\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

- $A \in \mathbb{R}^{M \times N}$ denotes a matrix with M rows and N columns, and every element of the matrix is a real number.
- $\mathbb{R}^{M \times N}$ denotes the space of M-by-N matrices (another notation $\mathcal{R}^{M \times N}$)
- $\boldsymbol{x} \in \mathbb{R}^N$ denotes a vector with N elements which are real numbers
- $\mathbb{R}^N$ denotes the space of N-dimensional vectors (another notation $\mathcal{R}^N$)

- $\boldsymbol{x} \in \mathbb{R}^N$ denotes an *N*-dimensional vector

a column vector $\boldsymbol{x} = \begin{bmatrix} x_{[1]} \\ x_{[2]} \end{bmatrix}$ or a row vector $\boldsymbol{x} = \begin{bmatrix} x_{[1]} & x_{[2]} \end{bmatrix}$

- In the lecture notes, the statement "a vector $\boldsymbol{x}$" refers to a column vector unless we explicitly define "$\boldsymbol{x}$ is a row vector"

Linear Algebra Basics

- transpose of a matrix by flipping the rows and columns.

  $A \in \mathbb{R}^{M \times N}$ then, $A^T \in \mathbb{R}^{N \times M}$

  (A' (A-prime) is also used to denote matrix transpose)

  $$A = \begin{bmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{bmatrix}, \text{ then } A^T = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

- For each element of the matrix A, the transpose can be written as

  $$A^T_{i,j} = A_{j,i}$$

Linear Algebra Basics

- Matrix transpose has the following properties
  - $(A^T)^T = A$
  - $(AB)^T = B^T A^T$
  - $(A + B)^T = A^T + B^T$
- A square matrix $A \in \mathbb{R}^{N \times N}$ is symmetric if $A = A^T$

Vector and Matrix Multiplication

- Product of two matrices $A \in \mathbb{R}^{M \times N}$ and $B \in \mathbb{R}^{N \times P}$ is a matrix $C \in \mathbb{R}^{M \times P}$

$$C_{m,p} = \sum_{n=1}^{N} A_{m,n} B_{n,p}$$

Vector and Matrix Multiplication

- Given two vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^N$, the expression $\boldsymbol{x}^T\mathbf{y}$ (a.k.a $\boldsymbol{x} \cdot \boldsymbol{y}$) is called **inner product** or **dot product** of the two vectors, and it is a real number

$$\boldsymbol{x}^T\mathbf{y} = \begin{bmatrix} x_{[1]} & x_{[2]} & x_{[3]} \end{bmatrix} \begin{bmatrix} y_{[1]} \\ y_{[2]} \\ y_{[3]} \end{bmatrix} = \sum_{n=1}^{3} x_{[n]} y_{[n]},$$

where $\boldsymbol{x}, \mathbf{y} \in \mathbb{R}^3$

vector space $\mathbb{R}^N$: a collection/set of all vectors that have N elements

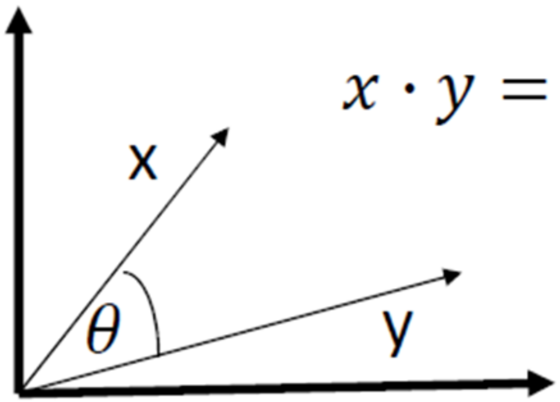
Vector and Matrix Multiplication

- Given two vectors $\boldsymbol{x} \in \mathbb{R}^N$ and $\boldsymbol{y} \in \mathbb{R}^M$, then $\boldsymbol{x}\boldsymbol{y}^T$ is called the **outer product** of the two vectors, and it is a matrix $[\boldsymbol{x}\boldsymbol{y}^T]_{n,m} = x_{[n]}y_{[m]}$

$$\boldsymbol{x}\boldsymbol{y}^T = \begin{bmatrix} x_{[1]} \\ x_{[2]} \\ x_{[3]} \end{bmatrix} \begin{bmatrix} y_{[1]} & y_{[2]} \end{bmatrix} = \begin{bmatrix} x_{[1]}y_{[1]} & x_{[1]}y_{[2]} \\ x_{[2]}y_{[1]} & x_{[2]}y_{[2]} \\ x_{[3]}y_{[1]} & x_{[3]}y_{[2]} \end{bmatrix}$$

Vector and Matrix Multiplication

- The dot product has a geometrical interpretation, for vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^2$ with angle θ between them, then



$$x \cdot y = |x||y| \cos\theta$$

note: $\boldsymbol{x} \cdot \boldsymbol{y} = \boldsymbol{x}^T \mathbf{y}$

$$|\boldsymbol{x}| = \sqrt{x_{[1]}^2 + x_{[2]}^2}$$

'length' of the vector

- In a high-dimensional vector space $\mathbb{R}^N$, we can also think $\cos\theta$ is the 'angle', which measures the similarity between two feature vectors, $\boldsymbol{x}$ and $\boldsymbol{y}$

$$\cos\theta = \frac{\boldsymbol{x} \cdot \boldsymbol{y}}{|\boldsymbol{x}||\boldsymbol{y}|}$$

Trace of a Square Matrix

- The trace of a square matrix $A \in \mathbb{R}^{N\times N}$, denoted as trace(A) or tr(A), is the sum of the diagonal elements in the matrix

  $$\mathrm{tr}(A) = \sum_{n=1}^{N} A_{n,n}$$

- Matrix trace has the following properties
  - $\mathrm{tr}(A) = \mathrm{tr}(A^T)$ for $A \in \mathbb{R}^{N\times N}$
  - $\mathrm{tr}(A+B) = \mathrm{tr}(A) + \mathrm{tr}(B)$, for $A, B \in \mathbb{R}^{N\times N}$
  - $\mathrm{tr}(t \times A) = t \times \mathrm{tr}(A)$ where $t \in \mathbb{R}$
  - If ABC, CBA *and* CAB are square matrices, then $\mathrm{tr}(ABC) = \mathrm{tr}(CBA) = \mathrm{tr}(CAB)$

Linear Independence and Rank

- A set of vectors $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_N\} \subset \mathbb{R}^M$ are linearly independent if no vector can be represented as a linear combination of the remaining vectors

$$\boldsymbol{x}_n = \sum_{k \neq n} \alpha_k \boldsymbol{x}_k$$

linearly dependent if some α_k is not zero

Linear Independence and Rank

- The **column rank** of a matrix $A \in \mathbb{R}^{M \times N}$ is the size of the largest set of linearly independent columns of A
- e.g., the column rank of A is 2 means 2 columns in A are linearly independent.

- The **row rank** of a matrix $A \in \mathbb{R}^{M \times N}$ is the size of the largest set of linearly independent rows of A

  e.g., the row rank of A is 3 means 3 rows in A are linearly independent.

- **matrix rank = row rank = column rank**

Span

- The **span** of a set of vectors $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_N\}$ is the set of all vectors that can be expressed as a linear combination

$$\text{span}(\{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_N\}) = \left\{\boldsymbol{v} \mid \boldsymbol{v} = \sum_{n=1}^{N} \alpha_n \boldsymbol{x}_n , \alpha_n \in \mathbb{R}\right\}$$

- If $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_N\} \subset \mathbb{R}^N$ is a set of linearly independent vectors, then $\text{span}(\{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_N\}) = \mathbb{R}^N$

Range (column space)

- The range of a matrix $A \in \mathbb{R}^{M \times N}$, denoted as $\mathcal{R}(A)$, is the span of the columns of A

  it is also called column space

Column Space and Row Space

- row space and column space are the linear subspaces generated by row and column vectors of a matrix

- For a matrix $A \in \mathbb{R}^{M \times N}$,
  - col space of A = span (cols of A)
  - row space of A = span (rows of A)
  - rank(A) = dim (row space of A) = dim (col space of A)

  note: column is col, dimension is dim

  dim of a space is the size of the largest set of linearly independent vectors in the space

Null Space

- The nullspace of a matrix $A \in \mathbb{R}^{M \times N}$, denoted as $\mathcal{N}(A)$, is the set of all vectors defined in

$$\mathcal{N}(A) = \left\{ \boldsymbol{x} \mid A\boldsymbol{x} = 0, \boldsymbol{x} \in \mathbb{R}^N \right\}$$

Vector Norm

- Norm of a vector $||\boldsymbol{x}||$ is a measure of the 'length' of the vector $\boldsymbol{x}$, and $\boldsymbol{x} \in \mathbb{R}^N$

- Common norms used in machine learning are
  - ℓ_2 norm $||\boldsymbol{x}||_2 = \sqrt{\sum_{n=1}^{N} x_{[n]}^2} = \sqrt{\boldsymbol{x}^T \boldsymbol{x}}$ (Euclidean norm)

Vector Norm of $\boldsymbol{x} \in \mathbb{R}^N$

- Common norms used in machine learning are
  - ℓ_1 norm $\|\boldsymbol{x}\|_1 = \sum_{n=1}^{N} |x_{[n]}|$
    (sum of the absolute values of the elements)
  - ℓ_2 norm $\|\boldsymbol{x}\|_2 = \sqrt{\sum_{n=1}^{N} x_{[n]}^2} = \sqrt{\boldsymbol{x}^T \boldsymbol{x}}$
    (Euclidean norm)
  - ℓ_∞ norm $\|\boldsymbol{x}\|_\infty = \max\{|x_{[1]}|, \dots, |x_{[N]}|\}$
    (max of the absolute values)

Vector Norm

- In general, we can define a ℓ_p norm with $p \geq 1$

$$\|\boldsymbol{x}\|_p = \left(\sum_{n=1}^{N} \left|x_{[n]}\right|^p\right)^{\frac{1}{p}}$$

- $\|\boldsymbol{x}\|_p$ is not a norm when $0 < p < 1$ because triangle inequality may not hold. an example:

$$\|[1,0]\|_p = 1$$

$$\|[0,1]\|_p = 1$$

$$\|[1,0]+[0,1]\|_p = \|[1,1]\|_p = 2^{\frac{1}{p}} > 2 = \|[1,0]\|_p + \|[0,1]\|_p$$

Vector Norm

- ℓ_0 norm: a special norm of $\boldsymbol{x} \in \mathbb{R}^N$

$$\|\boldsymbol{x}\|_0 = \sum_{n=1}^{N} \left|x_{[n]}\right|^0$$

$$\left|x_{[n]}\right|^0 = 1 \text{ if } \left|x_{[n]}\right| > 0$$
$$\left|x_{[n]}\right|^0 = 0 \text{ if } \left|x_{[n]}\right| = 0$$

Thus, $\|\boldsymbol{x}\|_0$ is the number of non-zero elements in $\boldsymbol{x}$

$$\sum_{n=1}^{N} \left|x_{[n]}\right|^p \rightarrow \|\boldsymbol{x}\|_0 \; when \; p \rightarrow 0$$

Vector Norm

$||\boldsymbol{x}||$ is the same as $||\boldsymbol{x}||_p$

$p \geq 1$ or $p = 0$

Vector Norm

- Norm of a vector $||\boldsymbol{x}||$ is a measure of the 'length' of vector $\boldsymbol{x}$
- $\boldsymbol{x} \in \mathbb{R}^N$
- norm $||\boldsymbol{x}||$ is a function $f: \mathbb{R}^N \rightarrow \mathbb{R}$ that satisfies
  - For every $\boldsymbol{x} \in \mathbb{R}^N$, $f(\boldsymbol{x}) \geq 0$ (non-negativity)
  - $f(\boldsymbol{x}) = 0$ if and only if (iff) $\boldsymbol{x} = \mathbf{0}$ (definiteness)
  - For every $\boldsymbol{x} \in \mathbb{R}^N$, $t \in \mathbb{R}$ $f(t\boldsymbol{x}) = |t| f(\boldsymbol{x})$ (homogeneity)
  - For every $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^N$, $f(\boldsymbol{x} + \boldsymbol{y}) \leq f(\boldsymbol{x}) + f(\boldsymbol{y})$ (triangle inequality)

Matrix Norm

- Norms can be defined for matrices

  - Frobenius Norm (F Norm) for $A \in \mathbb{R}^{M \times N}$

$$\|A\|_F = \sqrt{\sum_{m=1}^{M} \sum_{n=1}^{N} A_{m,n}^2} = \sqrt{tr(A^T A)}$$

other norms:
http://mathworld.wolfram.com/MatrixNorm.html

Identity Matrix

- The identity matrix, denoted by $I \in \mathbb{R}^{N \times N}$ is a square matrix with ones on the diagonal and zeros everywhere else.

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Diagonal Matrix

- A diagonal matrix, $D = diag(d_1, d_2, \ldots, d_N) \in \mathbb{R}^{N \times N}$, is a square matrix where all non-diagonal elements are zeros

$$D = \begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix} = diag(d_1, d_2)$$

Orthogonal Matrix

- Two vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^N$ are orthogonal (e.g., perpendicular) to each other if $\boldsymbol{x} \cdot \boldsymbol{y} = 0$.
- two vectors $\boldsymbol{x}, \boldsymbol{y}$ are orthonormal to each other if they are orthogonal and $||\boldsymbol{x}||_2 = ||\boldsymbol{y}||_2$=1
- A matrix $U \in \mathbb{R}^{N \times N}$ is called orthogonal if $U^T U = I = U U^T$
  - the columns are orthonormal to each other
  - the rows are orthonormal to each other
  - a property:$||\boldsymbol{U}\boldsymbol{x}||_2 = ||\boldsymbol{x}||_2$

Determinant of a Matrix

- The determinant of a square matrix $A \in \mathbb{R}^{N \times N}$ us a function $f: \mathbb{R}^N \rightarrow \mathbb{R}$, denoted by $|A|$ or **det**(A), and is calculated as

$$|A| = \sum_{i=1}^{N} (-1)^{i+j} A_{i,j} \, |B| \, , \quad \text{for any } j \in \{1, 2, \ldots, N\}$$

B is obtained by eliminating the *i*-th row and the *j*-th col of A

Inverse of a Matrix

- The inverse of a square matrix $A \in \mathbb{R}^{N \times N}$ is denoted A^{-1} such that $A^{-1}A = I = AA^{-1}$
- For some square matrices, A^{-1} may not exist, those matrices are called singular or non-invertible. A^{-1}exists if and only if A is full rank (rank(A) = N)

Inverse of a Matrix

- For non-square matrices, $A \in \mathbb{R}^{M \times N}$, the inverse, denoted by A^{+}, is given by

$$A^{+} = \left(A^{T}A\right)^{-1}A^{T} \text{ which is called pseudo inverse}$$

assuming the square matrix $A^{T}A$ is invertible (usually $N < M$)

$A^{+}A = \left(A^{T}A\right)^{-1}A^{T}A = I$, A^{+} is the left-inverse of A

Note: $AA^{+} = A\left(A^{T}A\right)^{-1}A^{T} \neq I$, $A^{+}$is NOT the right-inverse of A

Eigenvalues and Eigenvectors

- Given a square matrix $A \in \mathbb{R}^{N \times N}$, the scalar λ is an eigenvalue and $\boldsymbol{x}$ is an eigenvector of A if

$$A\boldsymbol{x} = \lambda\boldsymbol{x}, \text{ and } \boldsymbol{x} \neq \mathbf{0}$$

$\lambda \in \mathbb{C}$ (it could be a complex number)

Eigenvalues and Eigenvectors

- Given a square matrix $A \in \mathbb{R}^{N\times N}$, $\lambda \in \mathbb{C}$ (could be a complex number) is an eigenvalue and $\boldsymbol{x} \in \mathbb{C}$ is an eigenvector of A if

$$A\boldsymbol{x} = \lambda\boldsymbol{x}, \text{ and } \boldsymbol{x} \neq \boldsymbol{0}$$

  obtain $\boldsymbol{x}$ and λ by solving the linear equations:

$$(A - \lambda I)\boldsymbol{x} = \boldsymbol{0}, \text{ and } |A - \lambda I| = 0$$

  Eigenvalues may not be unique

  Eigenvalues could be complex numbers

- Eigen Decomposition Theorem

  http://mathworld.wolfram.com/EigenDecompositionTheorem.html

  http://mathworld.wolfram.com/Eigenvector.html

Eigenvalues and Eigenvectors

- Given a square matrix $A \in \mathbb{R}^{N\times N}$, $\lambda \in \mathbb{C}$ (could be a complex number) is an eigenvalue and $\boldsymbol{x} \in \mathbb{C}$ is an eigenvector of A if

$$A\boldsymbol{x} = \lambda\boldsymbol{x}, \text{ and } \boldsymbol{x} \neq \mathbf{0}$$

- **Eigen Decomposition Theorem**

if we can obtain N linearly independent eigenvectors, then

$$A = PDP^{-1}$$

every column of $P \in \mathbb{R}^{N\times N}$ is an eigenvector of A

D=diag($\{\lambda_1, \ldots, \lambda_N\}$)

if the number of linearly independent eigenvectors $< N$

A can be written using a so-called singular value decomposition.

Eigenvalues and Eigenvectors

- **Symmetric matrix** $A \in \mathbb{R}^{N \times N}$
  - eigenvalues are **real** numbers, $\{\lambda_1, \dots, \lambda_N\}$, D=diag($\{\lambda_1, \dots, \lambda_N\}$)
  - eigenvectors are **orthogonal**, $\{\boldsymbol{x}_1, \dots, \boldsymbol{x}_M, M \le N\}$, $\boldsymbol{x_n}^T \boldsymbol{x_k} = 0$ if $k \neq n$
  (usually set $||\boldsymbol{x_n}||_2 = 1$ to achieve orthonormal)

Singular Value Decomposition (full)

- A matrix $A \in \mathbb{R}^{\boldsymbol{M \times N}}$ $(M \geq N)$ can always be decomposed as

$$A = UDV^T$$

$$A = [\boldsymbol{u}_1, \boldsymbol{u}_2, \dots, \boldsymbol{u}_M] D [\boldsymbol{v}_1, \boldsymbol{v}_2, \dots, \boldsymbol{v}_N]^T$$

  - $D \in \mathbb{R}^{M \times N}$, and main diagonal elements of D are $[\lambda_1, \dots, \lambda_N, 0, \dots, 0]$
  - the columns of $U \in \mathbb{R}^{M \times M}$ are orthonormal: $U^T U = I$
  - the columns of $V \in \mathbb{R}^{N \times N}$ are orthonormal: $V^T V = I$
  - singular values $\{\lambda_1, \dots, \lambda_N\}$ are real numbers
  - every column of U, denoted by $\boldsymbol{u}_m$, is called a left-singular vector of A
  - every column of V, denoted by $\boldsymbol{v}_n$, is called a right-singular vector of A

Matrix Calculus (basic concepts)

- scalar by vector

For vectors $\boldsymbol{x}, \boldsymbol{b} \in \mathbb{R}^N$, let $f(\boldsymbol{x}) = \boldsymbol{b}^T \boldsymbol{x}$, then $\frac{\partial f}{\partial \boldsymbol{x}} = \nabla_{\boldsymbol{x}} f = \boldsymbol{b}$

$$\nabla_{\boldsymbol{x}} f = \begin{bmatrix} \frac{\partial f}{\partial x_{[1]}} \\ \cdots \\ \frac{\partial f}{\partial x_{[N]}} \end{bmatrix} \text{ where } \boldsymbol{x} = \begin{bmatrix} x_{[1]} \\ \cdots \\ x_{[N]} \end{bmatrix}$$

We need this to understand back-propagation in neural networks

Matrix Calculus (basic concepts)

- scalar by vector

Quadratic function $f(\boldsymbol{x}) = \boldsymbol{x}^T A\boldsymbol{x}$ with $A \in \mathbb{R}^{N\times N}$ and A is symmetric

then: $\nabla_{\boldsymbol{x}} f = 2A\boldsymbol{x}$

References for self study

http://cs229.stanford.edu/section/cs229-linalg.pdf

https://atmos.washington.edu/~dennis/MatrixCalculus.pdf

If you want to be a machine learning researcher, then read:

Mathematics for Econometrics (chapter 1 to 5)
https://www.springer.com/us/book/9781461481447

Matrix Cookbook by KB Peterson

Singular Value Decomposition (Economy-Size)

- A matrix $A \in \mathbb{R}^{M \times N}$ $(M \geq N)$ can always be decomposed as

$$A = UDV^T$$

$$A = [\boldsymbol{u}_1, \boldsymbol{u}_2, \dots, \boldsymbol{u}_N] \begin{bmatrix} \lambda_1 & & \\ & \dots & \\ & & \lambda_N \end{bmatrix} [\boldsymbol{v}_1, \boldsymbol{v}_2, \dots, \boldsymbol{v}_N]^T$$

  - the columns of $U \in \mathbb{R}^{M \times N}$ are orthonormal: $U^T U = I$
  - the columns of $V \in \mathbb{R}^{N \times N}$ are orthonormal: $V^T V = I$
  - $D = diag(\lambda_1, \dots, \lambda_N)$
  - singular values $\{\lambda_1, \dots, \lambda_N\}$ are real numbers
  - every column of U, denoted by $\boldsymbol{u}_n$, is called a left-singular vector of A
  - every column of V, denoted by $\boldsymbol{v}_n$, is called a right-singular vector of A
  - $A\boldsymbol{v}_n = \lambda_n \boldsymbol{u}_n$ and $A^T \boldsymbol{u}_n = \lambda_n \boldsymbol{v}_n$, for n=1 to N

Singular Value Decomposition (SVD) and Eigen Decomposition

- A matrix $A \in \mathbb{R}^{M \times N}$ $(M \geq N)$ can always be decomposed as (**SVD**)

$$A = UDV^T$$

$$A = [\boldsymbol{u}_1, \boldsymbol{u}_2, \dots, \boldsymbol{u}_N] \begin{bmatrix} \lambda_1 & & \\ & \dots & \\ & & \lambda_N \end{bmatrix} [\boldsymbol{v}_1, \boldsymbol{v}_2, \dots, \boldsymbol{v}_N]^T$$

- $AA^T = UDV^T(UDV^T)^T = UDV^TVDU^T = U(DD)U^T$

  U contains N linearly independent eigenvectors of AA^T

  λ_n^2 is an eigenvalue of AA^T

- $A^TA = (UDV^T)^TUDV^T = VDU^TUDV^T = V(DD)V^T$

  V contains N linearly independent eigenvectors of A^TA

  λ_n^2 is an eigenvalue of A^TA

Pseudo Inverse and SVD

$$A = UDV^T$$

$D = diag(\lambda_1, \lambda_2, \dots \lambda_r, 0,0,0)$, where $\lambda_1 > 0 \dots$ and $\lambda_r > 0$

$$D^+ = diag\left(\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \dots, \frac{1}{\lambda_r}, 0, \dots, 0\right)$$

$$A^+ = VD^+U^T$$