Linear Algebra and Numpy

Last updated: September 12, 2016

Motivation for Linear Alegebra

- Linear algebra is the mathematics of vectors
- It is used in many aspects of Machine Learning
  - Dimensionality Reduction
  - Recommender Systems
  - Classification
  - Clustering
- So it's important to be familiar with the concept

Vectors

A vector can be reprsented by an array of real numbers:

$$\mathbf{x} = [x_1, x_2, \ldots, x_n]$$

A vector specifies a point in a (vector) space.
Can think of a vector as an arrow with its tail at the origin
The vector the gives the location of the tip of the arrow
The number of elements in the vector define the dimension of the vector

- $[2, 2]$ is a two dimensional vector
- $[x_1, x_2, \ldots, x_n]$ is an n-dimensional vector
- $[4]$ is a one dimensional vector, called a scalar

Vector Operations

Now that we know what a vector is, what can we do with them?

- We can add a scalar
  - $\mathbf{x} + a = [x_1 + a, x_2 + a, + \ldots x_n + a]$

Vector Operations

Now that we know what a vector is, what can we do with them?

- We can add a scalar
  - $\mathbf{x} + a = [x_1 + a, x_2 + a, + \ldots x_n + a]$
- We can multiply by a scalar
  - $a\mathbf{x} = [ax_1, ax_2, \ldots, ax_n]$

Vector Operations

Now that we know what a vector is, what can we do with them?

- We can add a scalar
  - $\mathbf{x} + a = [x_1 + a, x_2 + a, + \dots x_n + a]$
- We can multiply by a scalar
  - $a\mathbf{x} = [ax_1, ax_2, \dots, ax_n]$
- We can add and subtract them
  - $\mathbf{x} + \mathbf{y} = [x_1 + y_1, x_2 + y_2, \dots, x_n + y_n]$
  - $\mathbf{x} - \mathbf{y} = [x_1 - y_1, x_2 - y_2, \dots, x_n - y_n]$
  - Note: $\mathbf{x}$ and $\mathbf{y}$ must be the same dimension for this to work
  - This is know as a linear combination of $\mathbf{x}$ and $\mathbf{y}$

Dot Product

The dot product of two vectors $\mathbf{x}$ and $\mathbf{y}$ is

$$\mathbf{x} \cdot \mathbf{y} = \sum_i x_i y_i$$

Note that $\mathbf{x}$ and $\mathbf{y}$ must be the same dimension
This operations takes two vectors and returns a scalar
If the dot product of two vectors is zero then the vectors are *orthogonal*, or perpendicular

Norm of a Vector

The norm of a vector **x** is defined as

$$\|x\| = \sqrt{x_1^2 + x_2^2 + \ldots x_n^2}$$

This gives us the length, or magnitude of the vector.
This is called the l_2 norm, in general the l_p norm is

$$\|x\|_p = (|x_1|^p + |x_2|^p + \cdots + |x_n|^p)^{\frac{1}{p}}$$

- When does this come up in machine learning?

Norm of a Vector

The norm of a vector **x** is defined as

$$\|x\| = \sqrt{x_1^2 + x_2^2 + \dots x_n^2}$$

This gives us the length, or magnitude of the vector.
This is called the l_2 norm, in general the l_p norm is

$$\|x\|_p = (|x_1|^p + |x_2|^p + \dots + |x_n|^p)^{\frac{1}{p}}$$

- When does this come up in machine learning?
- Normalization
  - Dividing a vector by its norm $\frac{\mathbf{x}}{\|\mathbf{x}\|}$ makes it a *unit vector* (i.e. $\|\mathbf{u}\| = 1$
  - Puts all variables on scale from [0, 1]

Norm of a Vector

The norm of a vector $\mathbf{x}$ is defined as

$$\|x\| = \sqrt{x_1^2 + x_2^2 + \ldots x_n^2}$$

This gives us the length, or magnitude of the vector.
This is called the l_2 norm, in general the l_p norm is

$$\|x\|_p = (|x_1|^p + |x_2|^p + \cdots + |x_n|^p)^{\frac{1}{p}}$$

- When does this come up in machine learning?
- Normalization
  - Dividing a vector by its norm $\frac{\mathbf{x}}{\|\mathbf{x}\|}$ makes it a *unit vector* (i.e. $\|\mathbf{u}\| = 1$
  - Puts all variables on scale from [0, 1]
- Regularization
  - Helps prevent overfitting

Using the Norm: Distance and Angles

- The norm can help us calculate the difference between two vectors:

$$d\left(\mathbf{x}, \mathbf{y}\right) = \|\mathbf{x} - \mathbf{y}\|$$

- It can also help us calculate the angle θ beteen two vectors (along with the dot product)

$$cos\left(\theta\right) = \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\|\|\mathbf{y}\|}$$

  - This is called the *cosine similarity*
  - Ranges between [-1, 1]

Similarity

- We can use both distance and angle to measure similarity
- Two vectors are similar if the distance between them is "small"
- Two vectors are similar if their cosine similarity is close to 1
- When would this be useful in machine learning?

Similarity

- We can use both distance and angle to measure similarity
- Two vectors are similar if the distance between them is "small"
- Two vectors are similar if their cosine similarity is close to 1
- When would this be useful in machine learning?
  - Classification
  - Clustering
  - Recommender systems

Linear Combination

- For some vectors $(\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_k)$ if one vector $\mathbf{x_i}$ can be written as

$$\mathbf{x}_i = a_1\mathbf{x}_1 + a_2\mathbf{x}_2 + \cdots + a_{i-1}\mathbf{x}_{i-1} + a_{i+1}\mathbf{x}_{i+i} + \cdots + a_k\mathbf{x}_k$$

  Then $\mathbf{x}_i$ is a linear combination of the vectors $(\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_{i-1}, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_k)$
- We say that the vectors are *linearly dependent*

Definition

A matrix is an array of numbers with n rows and p columns:

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix}$$

The dimension of X is $n \times p$
x_{ij} refers to the element in the i^{th} row and j^{th} column

Basic Properties

If X and Y are both $n \times p$ matrices then

- $X + Y$ is a matrix whose $(i, j)^{th}$ entry is $x_{ij} + y_{ij}$
- $X - Y$ is a matrix whose $(i, j)^{th}$ entry is $x_{ij} - y_{ij}$
- aX is a matrix whose $(i, j)^{th}$ entry is ax_{ij}

Matrix Multiplication

- In order to multiply two matrices they must be *conformable*
- This means the number of columns of the first matrix must be the same as the number of rows of the second
- If two matrices are conformable then the product of XY is a matrix M whose $(i, j)^{th}$ element is the dot product of the i^{th} row of X with the j^{th} column of Y

$$m_{ij} = \sum_{s=1}^{k} x_{is} y_{sj} = x_{i1} y_{1j} + \cdots + x_{ik} y_{kj}$$

- Note that even if both XY and YX exist, in general $XY \neq YX$

Additonal Properties

- If X and Y are both $n \times p$ matrices, then $X + Y = Y + X$
- If X, Y, and Z are all $n \times p$ matrices, then $X + (Y + Z) = (X + Y) + Z$
- If X, Y, and Z are all conformable, then $X(YZ) = (XY)Z$
- If X is of dimension $n \times k$ and Y and Z are of dimension $k \times p$, then $X(Y + Z) = XY + XZ$
- If X is of dimension $p \times n$ and Y and Z are of dimension $k \times p$, then $(Y + Z)X = YX + ZX$
- If a and b are real numbers, and X is an $n \times p$ matrix, then $(a + b)X = aX + bX$
- If a is a real number, and X and Y are both $n \times p$ matrices, then $a(X + Y) = aX + aY$
- If z is a real number, and X and Y are conformable, then $X(aY) = a(XY)$

Transpose

If X is an $n \times p$ matrix

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix}$$

Then the transpose of X is a $p \times n$ matrix with the rows and columns of X interchanged

$$X^T = \begin{bmatrix} x_{11} & x_{21} & \cdots & x_{n1} \\ x_{12} & x_{22} & \cdots & x_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1p} & x_{2p} & \cdots & x_{np} \end{bmatrix}$$

Properites of Transpose

- Let X be an $n \times p$ matrix and a a real number, then

$$(cX)^T = cX^T$$

- Let X and Y be $n \times p$ matrices, then

$$(X \pm Y)^T = X^T \pm Y^T$$

- Let X be an $n \times k$ matrix and Y be a $k \times p$ matrix, then

$$(XY)^T = Y^T X^T$$

Vectors in Matrix Form

It can be useful to think of vectors as matrices

- A column vector is an $n \times 1$ matrix

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

- A row vector is written as the transpose

$$\mathbf{x}^T = \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix}$$

- If two vectors $\mathbf{x}$ and $\mathbf{y}$ have the same length then the dot product is matrix multiplication

$$\mathbf{x}^T\mathbf{y} = x_1 y_1 + \cdots + x_n y_n$$

Inverse of a Matrix

- The inverse of an $n \times n$ matrix X is a matrix X^{-1} of the same dimension where
$$X^{-1}X = XX^{-1} = I$$
- I is the identity matrix
- if X^{-1} exists then X is said to be *invertible* or *nonsingular*, otherwise X is *noninvertible* or *singular*
- If any row (column) of X can be represented as a linear combination of the other rows (columns) of X, then X is singular
    - The number of linearly independent columns or rows of X is called the rank of X
    - If $rank(X) < n$ then X is singular
    - If $rank(X) = n$ the X is nonsingular

Properties of the Inverse

- If X is invertible, then X^{-1} is invertible and
$$(X^{-1})^{-1} = X$$
- If X and Y are both $n \times n$ invertible matrices, then XY is invertible and
$$(XY)^{-1} = Y^{-1}X^{-1}$$
- If X is invertible, then X^T is invertible and
$$\left(X^T\right)^{-1} = \left(X^{-1}\right)^T$$

Matrix Equations

A system of equations of the form:

$$\begin{aligned} a_{11}x_1 + \cdots + a_{1n}x_n &= b_1 \\ \vdots \qquad\qquad & \quad \vdots \\ a_{m1}x_1 + \cdots + a_{mn}x_n &= b_m \end{aligned}$$

can be written as a matrix equation:

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

and hence, has solution

$$\mathbf{x} = A^{-1}\mathbf{b}$$

Eigenvectors and Eigenvalues

Let A be an $n \times n$ matrix and $\mathbf{x}$ be an $n \times 1$ nonzero vector. An *eigenvalue* of A is a number λ such that

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

A vector $\mathbf{x}$ satisfying this equation is called an *eigenvector* associated with λ

Eigenvectors and eigenvalues will play a huge role in matrix methods later in the course (PCA, SVD, NMF).