4 Linear Algebra Review

For this topic we quickly review many key aspects of linear algebra that will be necessary for the remainder of the course.

4.1 Vectors and Matrices

For the context of data analysis, the critical part of linear algebra deals with vectors and matrices of real numbers.

In this context, a *vector* $v = (v_1, v_2, \ldots, v_d)$ is equivalent to a point in $\mathbb{R}^d$. By default a vector will be a column of d numbers (where d is context specific)

$$v = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}.$$

but in some cases we will assume the vector is a row

$$v^T = [v_1 \; v_2 \; \ldots \; v_n].$$

An $n \times d$ *matrix* A is then an ordered set of n row vectors $a_1, a_2, \ldots a_n$

$$A = [a_1 \; a_2 \; \ldots \; a_n] = \begin{bmatrix} - & a_1 & - \\ - & a_2 & - \\ & \vdots & \\ - & a_n & - \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{1,2} & \ldots & A_{1,d} \\ A_{2,1} & A_{2,2} & \ldots & A_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n,1} & A_{n,2} & \ldots & A_{n,d} \end{bmatrix},$$

where vector $a_i = [A_{i,1}, A_{i,2}, \ldots, A_{i,d}]$, and $A_{i,j}$ is the element of the matrix in the ith row and jth column. We can write $A \in \mathbb{R}^{n \times d}$ when it is defined on the reals.

A *transpose* operation $(\cdot)^T$ reverses the roles of the rows and columns, as seen above with vector v. For a matrix, we can write:

$$A^T = \begin{bmatrix} | & | & & | \\ a_1 & a_2 & \ldots & a_n \\ | & | & & | \end{bmatrix} = \begin{bmatrix} A_{1,1} & A_{2,1} & \ldots & A_{n,1} \\ A_{1,2} & A_{2,2} & \ldots & A_{n,2} \\ \vdots & \vdots & \ddots & \vdots \\ A_{1,n} & A_{2,d} & \ldots & A_{n,d} \end{bmatrix}.$$

Example: Linear Equations
A simple place these objects arise is in linear equations. For instance

$$\begin{array}{rrrl} 3x_1 & -7x_2 & +2x_3 & = -2 \\ -1x_1 & +2x_2 & -5x_3 & = 6 \end{array}$$

is a system of $n = 2$ linear equations, each with $d = 3$ variables. We can represent this system in matrix-vector notation as

$$Ax = b$$

where

$$b = \begin{bmatrix} -2 \\ 6 \end{bmatrix} \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad \text{and} \quad A = \begin{bmatrix} 3 & -7 & 2 \\ -1 & 2 & -5 \end{bmatrix}.$$

4.2 Addition

We can add together two vectors or two matrices only if they have the same dimensions. For vectors $x = (x_1, x_2, \ldots, x_d) \in \mathbb{R}^d$ and $y = (y_1, y_2, \ldots, y_d) \in \mathbb{R}^d$, then vector

$$z = x + y = (x_1 + y_1, x_2 + y_2, \ldots, x_d + y_d) \in \mathbb{R}^d.$$

Similarly for two matrices $A, B \in \mathbb{R}^{n \times d}$, then $C = A + B$ is defined where $C_{i,j} = A_{i,j} + B_{i,j}$ for all i, j.

4.3 Multiplication

Multiplication only requires alignment along one dimension. For two matrices $A \in \mathbb{R}^{n \times d}$ and $B \in \mathbb{R}^{d \times m}$ we can obtain a new matrix $C = AB \in \mathbb{R}^{n \times m}$ where $C_{i,j}$, the element in the ith row and jth column of C is defined

$$C_{i,j} = \sum_{k=1}^{d} A_{i,k} B_{k,j}.$$

To multiply A times B (where A is to the left of B, the order matters!) then we require the row dimension d of A to match the column dimension d of B. If $n \neq m$, then we *cannot* multiply BA. Keep in mind:

- Matrix multiplication is *associative* $(AB)C = A(BC)$.
- Matrix multiplication is *distributive* $A(B + C) = AB + AC$.
- Matrix multiplication is *not commutative* $AB \neq BA$.

We can also multiply a matrix A by a scalar α. In this setting $\alpha A = A\alpha$ and is defined by a new matrix B where $B_{i,j} = \alpha A_{i,j}$.

vector-vector products. There are two types of vector-vector products, and their definitions follow directly from that of matrix-matrix multiplication (since a vector is a matrix where one of the dimensions is 1). But it is worth highlighting these.

Given two column vectors $x, y \in \mathbb{R}^d$, the *inner product* or *dot product* is written

$$x^T y = x \cdot y = \langle x, y \rangle = [x_1 \; x_2 \; \ldots \; x_d] \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_d \end{bmatrix} = \sum_{i=1}^{d} x_i y_i,$$

where x_i is the ith element of x and similar for y_i. I prefer the last notation $\langle x, y\rangle$ since the same can be used for row vectors, and there is no confusion with multiplication in using $\cdot$; whether a vector is a row or a column is often arbitrary.

Note that this operation produces a single scalar value. The dot product is a linear operator. So this means for any scalar value α and three vectors $x, y, z \in \mathbb{R}^d$ we have

$$\langle \alpha x, y + z\rangle = \alpha\langle x, y + z\rangle = \alpha\left(\langle x, y\rangle + \langle x, z\rangle\right).$$

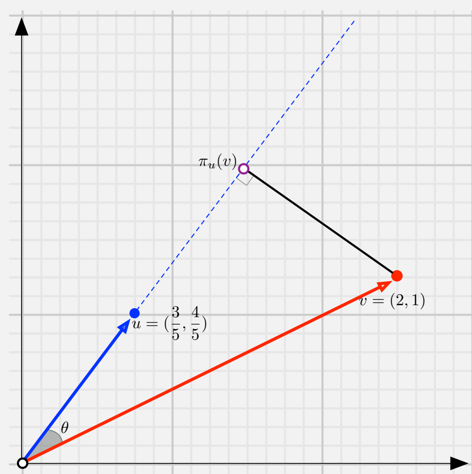
Example: Geometry of Dot Product

A dot product is one of my favorite mathematical operations! It encodes a lot of geometry. Consider two vectors $u = (\frac{3}{5}, \frac{4}{5})$ and $v = (2, 1)$, with an angle θ between them. Then it holds

$$\langle u, v\rangle = \mathrm{length}(u) \cdot \mathrm{length}(v) \cdot \cos(\theta).$$

Here $\mathrm{length}(\cdot)$ measures the distance from the origin. We'll see how to measure length with a "norm" $\|\cdot\|$ soon.

Moreover, since the $\|u\| = \mathrm{length}(u) = 1$, then we can also interpret $\langle u, v\rangle$ as the length of v projected onto the line through u. That is, let $\pi_u(v)$ be the closest point to v on the line through u (the line through u and the line segment from v to $\pi_u(v)$ make a right angle). Then $\langle u, v\rangle = \mathrm{length}(\pi_u(v)) = \|\pi_u(v)\|$.



For two column vectors $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^d$, the *outer product* is written

$$y^T x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} [y_1 \; y_2 \; \dots \; y_d] = \begin{bmatrix} x_1 y_1 & x_1 y_2 & \dots & x_1 y_d \\ x_2 y_1 & x_2 y_2 & \dots & x_2 y_d \\ \vdots & \vdots & \ddots & \vdots \\ x_n y_1 & x_n y_2 & \dots & x_n y_d \end{bmatrix} \in \mathbb{R}^{n \times d}.$$

Note that the result here is a matrix, not a scalar.

matrix-vector products. Another important and common operation is a matrix-vector product. Given a matrix $A \in \mathbb{R}^{n\times d}$ and a vector $x \in \mathbb{R}^d$, their product $y = Ax \in \mathbb{R}^n$.

When A is composed of row vectors $[a_1; a_2; \dots ; a_n]$, then I imagine this as transposing x (which should be a column vector here, so a row vector after transposing), and taking the dot product with each row of A.

$$y = Ax = \begin{bmatrix} - & a_1 & - \\ - & a_2 & - \\ & \vdots & \\ - & a_n & - \end{bmatrix} x = \begin{bmatrix} \langle a_1, x \rangle \\ \langle a_2, x \rangle \\ \vdots \\ \langle a_n, x \rangle \end{bmatrix}.$$

4.4 Norms

The standard *Euclidean norm* of a vector $v = (v_1, v_2, \ldots, v_d) \in \mathbb{R}^d$ is defined

$$\|v\| = \sqrt{\sum_{i=1}^{d} v_i^2} = \sqrt{\langle v, v \rangle}.$$

This measures the "straight-line" distance from the origin to the point at v. A vector v with norm $\|v\| = 1$ is said to be a *unit vector*; sometimes a vector x with $\|x\| = 1$ is said to be *normalized*.

However, a "norm" is a more generally concept. A class called L_p *norms* are well-defined for any parameter $p \in [1, \infty)$ as

$$\|v\|_p = \left(\sum_{i=1}^{d} |v_i|^p \right)^{1/p}.$$

Thus, when no p is specified, it is assumed to be $p = 2$. It is also common to denote $\|v\|_\infty = \max_{i=1}^{d} |v_i|$.

Because subtraction is well-defined between vectors $v, u \in \mathbb{R}^d$ of the same dimension, then we can also take the norm of $\|v - u\|_p$. While this is technically the norm of the vector resulting from the subtraction of u from v; it also provides a distance between u and v. In the case of $p = 2$, then

$$\|u - v\|_2 = \sqrt{\sum_{i=1}^{d} (u_i - v_i)^2}$$

is precisely the straight-line (Euclidean) distance between u and v.

Moreover, all L_p norms define a distance $D_p(u, v) = \|u - v\|_p$, which satisfies a set of special properties, which a required for a distance to be a *metric*. This include:

- **Symmetry:** For any $u, v \in \mathbb{R}^d$ we have $D(u, v) = D(v, u)$.
- **Non-negativity:** For any $u, v \in \mathbb{R}^d$ we have $D(u, v) \geq 0$, and $D(u, v) = 0$ if and only if $u = v$.
- **Triangle Inequality:** For any $u, v, w \in \mathbb{R}^d$ we have $D(u, w) + D(w, v) \geq D(u, v)$.

We can also define norms for matrices A. These take on slightly different notational conventions. The two most common are the spectral norm $\|A\| = \|A\|_2$ and the Frobenius norm $\|A\|_F$. The *Frobenius norm* is the most natural extension of the $p = 2$ norm for vectors, but uses a subscript F instead. It is defined for matrix $A \in \mathbb{R}^{n \times d}$ as

$$\|A\|_F = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{d} A_{i,j}^2} = \sqrt{\sum_{i=1}^{n} \|a_i\|^2},$$

where $A_{i,j}$ is the element in the ith row and jth column of A, and where a_i is the ith row vector of A. The *spectral norm* is defined for a matrix $A \in \mathbb{R}^{n \times d}$ as

$$\|A\| = \|A\|_2 = \max_{x \in \mathbb{R}^d} \|Ax\| / \|x\| = \max_{y \in \mathbb{R}^n} \|yA\| / \|y\|.$$

Its useful to think of these x and y vectors as being unit vectors, then the denominator can be ignored. Then we see that x and y only contain "directional" information, and the $\arg\max$ vectors point in the directions that maximize the norm.

4.5 Linear Independence

Consider a set of k vectors $x_1, x_2, \ldots, x_k \in \mathbb{R}^d$, and a set of k scalars $\alpha_1, \alpha_2, \ldots, \alpha_k \in \mathbb{R}$. Then because of linearity of vectors, we can write a new vector in $\mathbb{R}^d$ as

$$z = \sum_{i=1}^{k} \alpha_i x_i.$$

For a set of vectors $X = \{x_1, x_2, \ldots, x_k\}$, for any vector z where there exists a set of scalars α where z can be written as the above summation, then we say z is *linearly dependent* on X. If z **cannot** be written with any choice of α_is, the we say z is *linearly independent* of X. All vectors $z \in \mathbb{R}^d$ which are linearly dependent on X are said to be in its *span*.

$$\mathsf{span}(X) = \left\{ z \;\middle|\; z = \sum_{i=1}^{k} \alpha_i x_i, \;\; \alpha_i \in \mathbb{R} \right\}.$$

If $\mathsf{span}(X) = \mathbb{R}^d$ (that is for vectors $X = x_1, x_2, \ldots, x_k \in \mathbb{R}^d$ *all* vectors are in the span), then we say X forms a *basis*.

Example: Linear Independence
Consider input vectors in a set X as

$$x_1 = \begin{bmatrix} 1 \\ 3 \\ 4 \end{bmatrix} \qquad x_2 = \begin{bmatrix} 2 \\ 4 \\ 1 \end{bmatrix}$$

And two other vectors

$$z_1 = \begin{bmatrix} -3 \\ -5 \\ 2 \end{bmatrix} \qquad z_2 = \begin{bmatrix} 3 \\ 7 \\ 1 \end{bmatrix}$$

Note that z_1 is linearly dependent on X since it can be written as $z_1 = x_1 - 2x_2$ (here $\alpha_1 = 1$ and $\alpha_2 = -2$). However z_2 is linearly independent from X since there are no scalars α_1 and α_2 so that $z_2 = \alpha_1 x_1 + \alpha_2 x_2$ (we need $\alpha_1 = \alpha_2 = 1$ so the first two coordinates align, but then the third coordinate cannot).
Also the set X is linearly independent, since there is no way to write $x_2 = \alpha_1 x_1$.

A set of vectors $X = \{x_1, x_2, \ldots, x_n\}$ is *linearly independent* if there is no way to write any vector $x_i \in X$ in the set with scalars $\{\alpha_1, \ldots, \alpha_{i-1}, \alpha_{i+1}, \ldots, \alpha_n\}$ as the sum

$$x_i = \sum_{\substack{j=1 \\ j \neq i}}^{n} \alpha_j x_j$$

of the other vectors in the set.

4.6 Rank

The *rank* of a set of vectors $X = \{x_1, \ldots, x_n\}$ is the size of the largest subset $X' \subset X$ which are linearly independent. Usually we report $\textsf{rank}(A)$ as the rank of a matrix A. It is defined as the rank of the rows of the matrix, or the rank of its columns; it turns out these quantities are always the same.

If $A \in \mathbb{R}^{n \times d}$, then $\textsf{rank}(A) \leq \min\{n, d\}$. If $\textsf{rank}(A) = \min\{n, d\}$, then A is said to be *full rank*. For instance, if $d < n$, then using the rows of $A = [a_1; a_2; \ldots; a_n]$, we can describe *any* vector $z \in \mathbb{R}^d$ as the linear combination of these rows: $z = \sum_{i=1}^n \alpha_i a_i$ for some set $\{\alpha_1, \ldots, \alpha_n\}$; in fact, we can set all but d of these scalars to 0.

4.7 Inverse

A matrix A is said to be square if it has the same number of column as it has rows. A square matrix $A \in \mathbb{R}^{n \times n}$ may have an *inverse* denoted A^{-1}. If it exists, it is a unique matrix which satisfies:

$$A^{-1}A = I = AA^{-1}$$

where I is the $n \times n$ identity matrix

$$I = \begin{bmatrix} 1 & 0 & \ldots 0 & 0 & \\ 0 & 1 & \ldots 0 & 0 & \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & 0 \\ 0 & 0 & \ldots & 0 & 1 \end{bmatrix} = \textsf{diag}(1, 1, \ldots, 1).$$

Note that I serves the purpose of 1 in scalar algebra, so for a scalar a then using $a^{-1} = \frac{1}{a}$ we have $aa^{-1} = 1 = a^{-1}a$.

A matrix is said to be *invertable* if it has an inverse. Only square, full-rank matrices are invertable; and a matrix is always invertable if it is square and full rank. If a matrix is not square, the inverse is not defined. If a matrix is not full rank, then it does not have an inverse.

4.8 Orthogonality

Two vectors $x, y \in \mathbb{R}^d$ are *orthogonal* if $\langle x, y \rangle = 0$. This means those vectors are at a right angle to each other.

Example: Orthongonality
Consider two vectors $x = (2, -3, 4, -1, 6)$ and $y = (4, 5, 3, -7, -2)$. They are orthogonal since

$$\langle x, y \rangle = (2 \cdot 4) + (-3 \cdot 5) + (4 \cdot 3) + (-1 \cdot -7) + (6 \cdot -2) = 8 - 15 + 12 + 7 - 12 = 0.$$

A square matrix $U \in \mathbb{R}^{n \times n}$ is *orthogonal* if all of its columns $[u_1, u_2, \ldots, u_n]$ are normalized and are all orthogonal with each other. It follows that

$$U^T U = I = UU^T$$

since for any normalized vector u that $\langle u, u \rangle = \|u\| = 1$.

A set of columns (for instance those of an orthogonal U) which are normalized and all orthogonal to each other are said to be *orthonormal*. If $U \in \mathbb{R}^{n \times d}$ and has orthonormal columns, then $U^T U = I$ (here I is $d \times d$) but $UU^T \neq I$.

Orthogonal matrices are norm preserving. That means for an orthogonal matrix $U \in \mathbb{R}^{n \times n}$ and any vector $x \in \mathbb{R}^n$, then $\|Ux\| = \|x\|$.

Moreover, the columns $[u_1, u_2, \ldots, u_n]$ of an orthogonal matrix $U \in \mathbb{R}^{n \times n}$ form an *basis* for $\mathbb{R}^n$. This means that for any vector $x \in \mathbb{R}^n$, there exists a set of scalars $\alpha_1, \ldots, \alpha_n$ such that $x = \sum_{i=1}^n \alpha_i u_i$. More interestingly, we also have $\|x\|^2 = \sum_{i=1}^n \alpha_i^2$.

This can be interpreted as U describing a *rotation* (with possible mirror flips) to a new set of coordinates. That is the old coordinates of x are $(x_1, x_2, \ldots, x_n)$ and the coordinates in the new orthogonal basis $[u_1, u_2, \ldots, u_n]$ are $(\alpha_1, \alpha_2, \ldots, \alpha_n)$.

4.9 Python numpy Example

Python provides an excellent library called `numpy` (pronounced 'num-pie') for handling arrays and matrices, and performing linear basic algebra.

```
import numpy as np
from numpy import linalg as LA

#create an array, a row vector
v = np.array([1,2,7,5])
print v
#[1 2 7 5]
print v[2]
#7

#create a n=2 x d=3 matrix
A = np.array([[3,4,3],[1,6,7]])
print A
#[[3 4 3]
# [1 6 7]]
print A[1,2]
#7
print A[:, 1:3]
#[[4 3]
# [6 7]]

#adding and multiplying vectors
u = np.array([3,4,2,2])
#elementwise add
print v+u
#[4 6 9 7]
#elementwise multiply
print v*u
#[ 3  8  14 10]
# dot product
print v.dot(u)
# 35
print np.dot(u,v)
# 35
```

```
#matrix multiplication
B = np.array([[1,2],[6,5],[3,4]])
print A.dot(B)
#[[36 38]
# [58 60]]
x = np.array([3,4])
print B.dot(x)
#[11 38 25]

#norms
print LA.norm(v)
#8.88819441732
print LA.norm(v,1)
#15.0
print LA.norm(v,np.inf)
#7.0
print LA.norm(A, 'fro')
#10.9544511501
print LA.norm(A,2)
#10.704642743

#transpose
print A.T
#[[3 1]
# [4 6]
# [3 7]]
print x.T
#[3 4]   (always prints in row format)

print LA.matrix_rank(A)
#2
C = np.array([[1,2],[3,5]])
print LA.inv(C)
#[[-5.  2.]
# [ 3. -1.]]
print C.dot(LA.inv(C))
#[[  1.00000000e+00   2.22044605e-16]       (nearly  [[1 0]
# [  0.00000000e+00   1.00000000e+00]]                [0 1]] )
```