Why.N.How: Linear Algebra

Adrian KC Lee ScD
Research Fellow, MGH

Martinos Center
November 6, 2008

Format of tutorial

- Why do we care about Linear Algebra?
- What is a vector?
  - Geometric interpretation.
  - Vector operations, "Norms," Projections.
  - Independence, Basis set.
- What is a matrix?
  - Determinants, Inverse.
  - Under-determined solution / null-space / MEG.
  - Over-determined solution / LMMSE / fMRI.
Why Linear Algebra?

- What is linear algebra?
  - Study of vectors, systems of linear equations...
- Who cares?
  - Extensive applications in neurosciences.
  - Nice geometric interpretation of complex mathematical concepts, e.g., spectral coherence.
- How do we use MATLAB?
  - MATrix LABoratory: fundamental knowledge in linear algebra matters!!

What is a vector?

- Physical interpretation: a geometric entity characterized by a magnitude and a direction.
- Commonly encountered in Euclidean space.
  - In MEG: tangential and normal decomposition.
**Vector Addition**

\[ \mathbf{v} = \begin{pmatrix} \frac{2}{2} \\ \frac{3}{0} \end{pmatrix} \quad \mathbf{w} = \begin{pmatrix} 3 \\ 0 \end{pmatrix} \]

\[ \mathbf{z} = \mathbf{v} + \mathbf{w} = \begin{pmatrix} \frac{2}{2} \\ \frac{3}{0} \end{pmatrix} + \begin{pmatrix} 3 \\ 0 \end{pmatrix} = \begin{pmatrix} 5 \\ 2 \end{pmatrix} \]

**Vector Subtraction**

\[ \mathbf{v} = \begin{pmatrix} \frac{2}{2} \\ \frac{3}{0} \end{pmatrix} \quad \mathbf{w} = \begin{pmatrix} 3 \\ 0 \end{pmatrix} \]

\[ \mathbf{z} = \mathbf{v} - \mathbf{w} = \mathbf{v} + (-\mathbf{w}) = \begin{pmatrix} \frac{2}{2} \\ \frac{3}{0} \end{pmatrix} + \begin{pmatrix} -3 \\ 0 \end{pmatrix} = \begin{pmatrix} -1 \\ -2 \end{pmatrix} \]
“Norm”

\[
\|\mathbf{x}\| = \sqrt{x_1^2 + x_2^2 + x_3^2 + \ldots + x_n^2}
\]

\[
\mathbf{v} = \left(\begin{array}{c}
-3 \\
4
\end{array}\right)
\]

\[
\|\mathbf{v}\| = \sqrt{(-3)^2 + (4)^2} = 5
\]

\[
\mathbf{x} = \left(\begin{array}{c}
x_1 \\
x_2 \\
x_3 \\
\vdots \\
x_n
\end{array}\right)
\]

\[
\|\mathbf{x}\|_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p}
\]

- “Norm” is a measure of distance of a vector.
  - Default norm: L2-norm
  - Other norms: L1-norm; L∞-norm.

Scalar Multiplication / Unit Vector

\[
\mathbf{v} = \left(\begin{array}{c}
-3 \\
4
\end{array}\right)
\]

\[
\frac{1}{2} \mathbf{v} = \left(\begin{array}{c}
-3/2 \\
2
\end{array}\right)
\]

\[
\hat{\mathbf{v}} = \frac{\mathbf{v}}{\|\mathbf{v}\|} = \left(\begin{array}{c}
-3/5 \\
4/5
\end{array}\right)
\]

- Unit vector factors out the magnitude,
  - Watch out for MATLAB vectors (e.g., PCA), they give you unit vectors.
**Dot product / Projection**

*Dot Product:*

\[ a_1 b_1 + ... + a_n b_n = \|a\| \|b\| \cos \theta \]

*Angle between 2 vectors:*

\[ \cos \theta = \frac{a \cdot b}{\|a\| \|b\|} \]

› Two vectors are orthogonal if their dot product equals to 0, i.e., If \( a \perp b \), then \( a \cdot b = 0 \)

› Note: \( a \cdot a = \|a\|^2 \).

---

**Independence / Basis set**

Definition: \( \{v_1, \ldots, v_n\} \) is independent if and only if no vector in this set can be written as a linear combination of the rest:

\[ \beta_1 v_1 + \ldots + \beta_n v_n = 0 \quad \text{iff} \quad \beta_i = 0 \quad \forall \beta_i \]

*Independent set of vectors in \( \mathbb{R}^2 \):

\[ v_1 - v_2 = 0 \]

› Unique weights for decomposition into an independent set.

› Basis set: a set of independent vectors that span \( V \), e.g., \( v_1 \) and \( v_2 \) form a basis set for \( \mathbb{R}^2 \) but not \( \mathbb{R}^3 \).
Matrix (Addition / Scalar Multiplication / Transpose)

- **Matrix:** rectangular table of elements
  - used to describe linear equations.

\[
A = \begin{bmatrix}
  a_{11} & \cdots & a_{1n} \\
  \vdots & \ddots & \vdots \\
  a_{m1} & \cdots & a_{mn}
\end{bmatrix}
\]

- **Matrix Multiplication:** is NOT commutative:
  - i.e., generally, \(AB \neq BA\)

\[
A = \begin{bmatrix}
  1 & 2 \\
  0 & 1
\end{bmatrix};
B = \begin{bmatrix}
  2 & 3 \\
  1 & 0
\end{bmatrix}
\]

\[
A.B = \begin{bmatrix}
  1.2 + 2.1 & 1.3 + 2.0 \\
  0.2 + 1.1 & 0.3 + 1.0
\end{bmatrix} = \begin{bmatrix}
  4 & 3 \\
  1 & 0
\end{bmatrix}
\]

\[
A = \begin{bmatrix}
  2.1 + 3.0 & 2.2 + 3.1 \\
  1.1 + 0.0 & 1.2 + 0.1
\end{bmatrix} = \begin{bmatrix}
  2 & 7 \\
  1 & 2
\end{bmatrix}
\]

\[
A + B = \begin{bmatrix}
  2 + 1 & 4 + 3 \\
  4 + 1 & 11 + 2
\end{bmatrix} = \begin{bmatrix}
  3 & 7 \\
  4 & 13
\end{bmatrix}
\]
Matrix Multiplication

- Dimension of matrices must match

\[ C_{(m,p)} = A_{(m,n)} B_{(n,p)} \]

\[ A = \begin{bmatrix} 1 & 0 & 2 \\ -1 & 3 & 1 \end{bmatrix}; \quad B = \begin{bmatrix} 3 & 1 \\ 2 & 1 \\ 1 & 0 \end{bmatrix} \]

\[ B = \begin{bmatrix} 1.3 + 0.2 + 2.1 \\ -1.3 + 3.2 + 1.1 \\ 1.1 + 0.1 + 2.0 \end{bmatrix} = \begin{bmatrix} 5 \\ 4 \\ 2 \end{bmatrix} \]

Solving Linear Equations

\[ a_{11}x_1 + \cdots + a_{1n}x_n = b_1 \]
\[ a_{m1}x_1 + \cdots + a_{mn}x_n = b_n \]

\[ Ax = b \]

\[ A = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix} \text{ m rows} \]
\[ x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \]
\[ b = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix} \]
Matrix (Geometric interpretation)

\[ \mathbf{Ax} = \mathbf{b} \]

Solve for \( \mathbf{x} \):

\[ \mathbf{x} = \begin{bmatrix} 3 \\ -1 \end{bmatrix} \]

Column space:

\[ x_1 \begin{bmatrix} 2 \\ 4 \\ 1 \end{bmatrix} + x_2 \begin{bmatrix} 4 \\ 11 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix} \]

Row space:

\[ 2x_1 + 4x_2 = 2 \]
\[ 4x_1 + 11x_2 = 1 \]

How to solve for \( \mathbf{x} \) in \( \mathbf{Ax} = \mathbf{b} \)?

(M equations = N unknowns)

- How many equations do I need to solve for 3 unknowns?
  - In general: M equations, N unknowns.
  - Can we guarantee a unique solution when M=N?

Demonstration:

http://demonstrations.wolfram.com/PlanesSolutionsAndGaussianEliminationOfA33LinearSystem/
Solve $Ax = b$ by finding $A^{-1}$
(M equations = N unknowns)

- If $Ax = b$ and we want to solve for $x$:
  - We want $x = A^{-1}b$
- Does $A^{-1}$ always exist?
  - Only if $A$ is a square matrix (i.e., $M=N$).
  - Only if determinant of $A$ is not 0.
- How do we find determinant and $A^{-1}$?
  - Can use Gaussian elimination to find determinant.
  - Can use Cramer’s rule to find $A^{-1}$.
  - Or use MATLAB ☺!!!!

Undetermined case
(M equations < N unknowns)

- $M = N$
- Vectors span 2 dimension
  - Rank = 2.
  - Nullity = $N-R = 0$.
- $M < N$
- Vectors span 2 dimension
  - Rank = 2.
  - Nullity = $N-R = 1$. 
Null-space
(M equations < N unknowns)

- There is no unique solution since we are in an underdetermined case.
  - Any point lying on the red line is a valid solution!
- How do we find $A^{-1}$ when $A$ is not square ($M \neq N$)?
  - Pseudo-inverse.
  - "Do the best we can do, given what we have."

- $M < N$
  - Rank = 2
  - Nullity = 1

Undetermined soln in MEG

- In MEG:
  - 306 sensors ($M = 306$ equations).
  - 6000 dipole estimates ($N = 6000$ unknowns).
  - $M << N$.
  - MNE: uniqueness comes from other constraints.

- Side topics:
  - How to obtain pseudo-inverse?
  - Singular Value Decomposition.
  - Eigenvalues, eigenvectors.
  - $Ax = \lambda x$
Overdetermined case
(M equations > N unknowns)

Experiment:

- Relationship between number of people wanting me to stop talking ($y$) as a function of time ($t$).

Model: $\hat{y} = C_0 + C_1 t$
  
  » $C_0$: baseline Care-Factor; $C_1$: drift in Attention.
  
  » Data collected ($y$: reported distractions) at time $t_1=0$ ($y=y_1$); $t_2=10$ ($y=y_2$); $t_3=30$ ($y=y_3$); $t_4=40$ ($y=y_4$).

\[ \text{Ax} = \text{b} \]

\[
\begin{align*}
C_0 + t_1 C_1 &= y_1 \\
C_0 + t_2 C_1 &= y_2 \\
C_0 + t_3 C_1 &= y_3 \\
C_0 + t_4 C_1 &= y_4
\end{align*}
\]

\[ A = \begin{bmatrix}
1 & 0 \\
1 & 10 \\
1 & 30 \\
1 & 40
\end{bmatrix}, \quad x = \begin{bmatrix}
C_0 \\
C_1
\end{bmatrix}, \quad b = \begin{bmatrix}
y_1 \\
y_2 \\
y_3 \\
y_4
\end{bmatrix} \]

Results (Linear regression)

Model: $\hat{y} = C_0 + C_1 t$

\[ \hat{y} = 1.1 + 0.4t \]

Observations:

- $t = 0, y = 0$
- $t = 10, y = 8$
- $t = 30, y = 8$
- $t = 40, y = 20$

Minimize $e = y - \hat{y}$

\[ e = b - Ax \]
**LMMSE**

- Framework: estimate a set of weights \((C_0, C_1)\) such that the error \((e)\) between our observed data \((b)\) and our weighted design matrix \((A)\) is minimum,
  - i.e., “minimize” \(e = b - Ax = y - (C_0 \cdot 1 + C_1 \cdot t)\)
  - Linearly Minimize by Mean Square Error: \(\| e \|^2\)

**LMMSE / GLM**

- Closed form solution (LMMSE)
  - Find \(x\) that minimizes \(\|Ax - b\|^2\)
  - Least square solution to \(Ax = b\): \(x = (A^tA)^{-1}A^tb\)

- Now consider Random variables as vectors (GLM):
  - Least square solution to \(X\beta = Y\): \(\beta = (X^tX)^{-1}X^tY\)
  - where \(Y\) = Observed data; \(X\) = Design matrix; \(\beta\) = Parameter estimation
GLM in a nut-shell (fMRI context)

\[ Y = X\beta + \varepsilon \]

- **Y**: BOLD signal at various time at a single voxel
- **X**: Designed matrix
- **\( \beta \)**: Estimation of each component in **X**  
  (Parameters estimation)
- **\( \varepsilon \)**: Noise (also known as residuals, i.e., anything that is not modeled by **X**)

\( \varepsilon \) is orthogonal to all our modeled parameters:

Error (\( \varepsilon \)) is orthogonal to all our modeled parameters:

Angle between 2 vectors:

\[ \cos \theta = \frac{a \cdot b}{|a| |b|} \]

- \( \sigma_{yx} = E[\tilde{Y}\tilde{X}] = \sigma_y \sigma_x \cos \theta \)
- \( \cos \theta = \frac{\sigma_{yx}}{\sigma_y \sigma_x} = \rho \)
- \( MMSE = \sigma^2_y (1 - \rho^2) \)
- \( \rho = \text{correlation coefficient} \)

\( Y \): Data, \( X \): estimate

Coherence measure

(Gross et al., 2001):

\[ \frac{|C_{xy}(f)|^2}{C_{xx}(f)C_{yy}(f)} \]
Useful Resources

- MIT OpenCourseWare:
  - 18.06 Linear Algebra (Strang’s Lecture series)

- Good link between Linear Algebra and MATLAB:
  - [http://www.ling.upenn.edu/courses/ling525/linear_algebra_review.html](http://www.ling.upenn.edu/courses/ling525/linear_algebra_review.html)

- Why.N.How Wiki:
  - [https://gate.nmr.mgh.harvard.edu/wiki/whynhow/index.php/Main_Page](https://gate.nmr.mgh.harvard.edu/wiki/whynhow/index.php/Main_Page)