

Ling 5801: Lecture Notes 16

Linear Algebra

Complex equations like HMM filtering can be represented efficiently using linear algebra.

Contents

16.1 Terms	1
16.2 Unary Operations	1
16.3 Binary Operations	2
16.4 Example: Hidden Markov Models	4
16.5 Vector Normalization	4
16.6 Cosine Similarity	5

16.1 Terms

We can define matrices and vectors as arrays of real numbers:

- s is a **scalar** iff $s \in \mathbb{R}$

You will often see scalars written as Greek letters, e.g.: γ

- \mathbf{v} is a **vector** iff $\mathbf{v} \in \mathbb{R}^I$

Scalars in vectors can be identified by one index: say $\mathbf{v} = \begin{bmatrix} 1.8 \\ -3 \end{bmatrix}$ then: $\mathbf{v}_{[2]} = -3$

- \mathbf{M} is a **matrix** iff $\mathbf{M} \in \mathbb{R}^{I \times J}$

Scalars in matrices can be identified by two indices: say $\mathbf{M} = \begin{bmatrix} 1.8 & 12 \\ -3 & 40 \end{bmatrix}$ then: $\mathbf{M}_{[2,1]} = -3$

16.2 Unary Operations

- **transpose:** for all $\mathbf{M} \in \mathbb{R}^{I \times J}$, and all i, j indices to matrix rows and columns,

$$(\mathbf{M}^T)_{[i,j]} = \mathbf{M}_{[j,i]}$$

For example: $\begin{bmatrix} 1.8 & 12 \\ -3 & 40 \end{bmatrix}^T = \begin{bmatrix} 1.8 & -3 \\ 12 & 40 \end{bmatrix}$

- **diagonal:** for all $\mathbf{v} \in \mathbb{R}^I$, and all i indices to matrix rows and columns,

$$\text{diag}(\mathbf{v})_{[i,j]} = \begin{cases} \mathbf{v}_{[i]} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

For example: $\text{diag}\left(\begin{bmatrix} 1.8 \\ -3 \end{bmatrix}\right) = \begin{bmatrix} 1.8 & 0 \\ 0 & -3 \end{bmatrix}$

- **Kronecker delta:** for all i, j indices to matrix rows,

$$(\delta_i)_{[j]} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

For example: $\delta_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$

16.3 Binary Operations

- **scalar sum:** for all $s \in \mathbb{R}$, $\mathbf{M} \in \mathbb{R}^{I \times J}$, and all i, j indices to matrix rows and columns,

$$(s + \mathbf{M})_{[i,j]} = (\mathbf{M} + s)_{[i,j]} = s + \mathbf{M}_{[i,j]}$$

(commutative)

For example: $2 + \begin{bmatrix} 1.8 & 12 \\ -3 & 40 \end{bmatrix} = \begin{bmatrix} 3.8 & 14 \\ -1 & 42 \end{bmatrix}$

- **matrix/vector sum:** for all $\mathbf{M}, \mathbf{N} \in \mathbb{R}^{I \times J}$, with row and column indices i, j ,

$$(\mathbf{M} + \mathbf{N})_{[i,j]} = (\mathbf{N} + \mathbf{M})_{[i,j]} = \mathbf{M}_{[i,j]} + \mathbf{N}_{[i,j]}$$

(commutative)

For example: $\begin{bmatrix} 1.8 & 12 \\ -3 & 40 \end{bmatrix} + \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 2.8 & 14 \\ 0 & 44 \end{bmatrix}$

- **scalar product:** for all $s \in \mathbb{R}$, $\mathbf{M} \in \mathbb{R}^{I \times J}$, with row and column indices i, j ,

$$(s \mathbf{M})_{[i,j]} = (\mathbf{M} s)_{[i,j]} = s \cdot \mathbf{M}_{[i,j]}$$

(commutative)

For example: $2 \begin{bmatrix} 1.8 & 12 \\ -3 & 40 \end{bmatrix} = \begin{bmatrix} 3.6 & 24 \\ -6 & 80 \end{bmatrix}$

- **matrix/vector product:** for all $\mathbf{M} \in \mathbb{R}^{I \times K}$, $\mathbf{N} \in \mathbb{R}^{K \times J}$, with indices i, j, k ,

$$(\mathbf{M} \mathbf{N})_{[i,j]} = \sum_k \mathbf{M}_{[i,k]} \cdot \mathbf{N}_{[k,j]}$$

(not commutative)

For example:

$$\begin{bmatrix} 1.8 & 12 \\ -3 & 40 \\ 15 & -6 \\ 7 & 18 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} = \begin{bmatrix} (1.8 \cdot 1) + (12 \cdot 4) & (1.8 \cdot 2) + (12 \cdot 5) & (1.8 \cdot 3) + (12 \cdot 6) \\ (-3 \cdot 1) + (40 \cdot 4) & (-3 \cdot 2) + (40 \cdot 5) & (-3 \cdot 3) + (40 \cdot 6) \\ (15 \cdot 1) + (-6 \cdot 4) & (15 \cdot 2) + (-6 \cdot 5) & (15 \cdot 3) + (-6 \cdot 6) \\ (7 \cdot 1) + (18 \cdot 4) & (7 \cdot 2) + (18 \cdot 5) & (7 \cdot 3) + (18 \cdot 6) \end{bmatrix}$$

$$= \begin{bmatrix} 49.8 & 63.6 & 77.4 \\ 157 & 194 & 231 \\ -9 & 0 & 9 \\ 79 & 104 & 129 \end{bmatrix}$$

Practice: Complete the following:

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} = \begin{bmatrix} (_ \cdot _) + (_ \cdot _) & (_ \cdot _) + (_ \cdot _) \\ (_ \cdot _) + (_ \cdot _) & (_ \cdot _) + (_ \cdot _) \end{bmatrix}$$

There are two special cases of matrix multiplication for vectors:

1. **inner ('dot') product:** for vectors $\mathbf{v}, \mathbf{u} \in \mathbb{R}^I$,

$$\mathbf{v}^\top \mathbf{u} = \sum_i \mathbf{v}_{[i]} \cdot \mathbf{u}_{[i]}$$

For example:

$$\begin{bmatrix} 1.8 & -3 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} = (1.8 \cdot 1) + (-3 \cdot 2)$$

$$= -4.2$$

2. **outer product:** for vectors $\mathbf{v} \in \mathbb{R}^I, \mathbf{u} \in \mathbb{R}^J$,

$$(\mathbf{v} \mathbf{u}^\top)_{[i,j]} = \mathbf{v}_{[i]} \cdot \mathbf{u}_{[j]}$$

For example:

$$\begin{bmatrix} 1.8 \\ -3 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} = \begin{bmatrix} 1.8 \cdot 1 & 1.8 \cdot 2 & 1.8 \cdot 3 \\ -3 \cdot 1 & -3 \cdot 2 & -3 \cdot 3 \end{bmatrix}$$

$$= \begin{bmatrix} 1.8 & 3.6 & 5.4 \\ -3 & -6 & -9 \end{bmatrix}$$

- **Kronecker product:** for all $\mathbf{M} \in \mathbb{R}^{I \times J}, \mathbf{N} \in \mathbb{R}^{K \times L}$,

$$\mathbf{M} \otimes \mathbf{N} = \begin{bmatrix} \mathbf{M}_{[1,1]} \mathbf{N} & \cdots & \mathbf{M}_{[1,J]} \mathbf{N} \\ \vdots & \ddots & \vdots \\ \mathbf{M}_{[I,1]} \mathbf{N} & \cdots & \mathbf{M}_{[I,J]} \mathbf{N} \end{bmatrix}$$

For example:

$$\begin{aligned} \begin{bmatrix} 1.1 & 4 \\ -3 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} &= \begin{bmatrix} 1.1 \cdot 1 & 1.1 \cdot 2 & 1.1 \cdot 3 & 4 \cdot 1 & 4 \cdot 2 & 4 \cdot 3 \\ 1.1 \cdot 4 & 1.1 \cdot 5 & 1.1 \cdot 6 & 4 \cdot 4 & 4 \cdot 5 & 4 \cdot 6 \\ -3 \cdot 1 & -3 \cdot 2 & -3 \cdot 3 & 1 \cdot 1 & 1 \cdot 2 & 1 \cdot 3 \\ -3 \cdot 4 & -3 \cdot 5 & -3 \cdot 6 & 1 \cdot 4 & 1 \cdot 5 & 1 \cdot 6 \end{bmatrix} \\ &= \begin{bmatrix} 1.1 & 2.2 & 3.3 & 4 & 8 & 12 \\ 4.4 & 5.5 & 6.6 & 16 & 20 & 24 \\ -3 & -6 & -9 & 1 & 2 & 3 \\ -12 & -15 & -18 & 4 & 5 & 6 \end{bmatrix} \end{aligned}$$

16.4 Example: Hidden Markov Models

Hidden Markov Model filtering can be represented as a matrix chains:

$$\mathbf{p}^\top = \begin{matrix} \text{front} \\ \text{back} \end{matrix} \begin{pmatrix} .5 & .5 \end{pmatrix}$$

$$\mathbf{A} = \begin{matrix} & \begin{matrix} \text{front} \\ \text{back} \end{matrix} \end{matrix} \begin{pmatrix} .8 & .2 \\ .2 & .8 \end{pmatrix}$$

$$\mathbf{B} = \begin{matrix} & \begin{matrix} .5\text{kHz} & .75\text{kHz} & 1\text{kHz} & 1.25\text{kHz} & 1.5\text{kHz} & 1.75\text{kHz} & 2\text{kHz} \end{matrix} \\ \begin{matrix} \text{front} \\ \text{back} \end{matrix} & \begin{pmatrix} 0 & 0 & .1 & .2 & .4 & .2 & .1 \\ .1 & .2 & .4 & .2 & .1 & 0 & 0 \end{pmatrix} \end{matrix}$$

$$P(Y_2, x_{0..2}) = \mathbf{p}^\top \text{diag}(\mathbf{B} \delta_{x_0}) \mathbf{A} \text{diag}(\mathbf{B} \delta_{x_1}) \mathbf{A} \text{diag}(\mathbf{B} \delta_{x_2})$$

16.5 Vector Normalization

We can normalize these vectors using an n -norm of a vector v :

$$\|v\|_n = \left(\sum_j (v_j)^n \right)^{\frac{1}{n}} \quad (1)$$

There are several useful instantiations of this:

- The two-norm calculates the length of vector v as Euclidean coordinates:

$$\|v\|_2 = \left(\sum_j (v_j)^2 \right)^{\frac{1}{2}} \quad (2)$$

$$= \left(\sum_j v_j \cdot v_j \right)^{\frac{1}{2}} \quad (3)$$

$$= \sqrt{\sum_j v_j \cdot v_j} \quad (4)$$

For example: $\left\| \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\|_2 = \sqrt{2}$ $\left\| \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\|_2 = \sqrt{3}$

- The one-norm calculates ‘Manhattan distance’ (a sum over vector cells):

$$\|v\|_1 = \left(\sum_j (v_j)^1 \right)^{\frac{1}{1}} \quad (5)$$

$$= \sum_j v_j \quad (6)$$

For example: $\left\| \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\|_1 = 2$ $\left\| \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\|_1 = 3$

- The infinity-(‘inf’-)-norm calculates the maximum over vector cells (largest cell dominates):

$$\|v\|_\infty = \left(\sum_j (v_j)^\infty \right)^{\frac{1}{\infty}} \quad (7)$$

$$= \max_j v_j \quad (8)$$

For example: $\left\| \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\|_\infty = 1$ $\left\| \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\|_\infty = 1$

Norms are useful, as the name suggests, for **normalizing** vectors (resizing them to unit length):

$$\frac{\begin{bmatrix} 1 \\ 1 \end{bmatrix}}{\left\| \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\|_2} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}$$

16.6 Cosine Similarity

The dot product of two vectors, after being normalized, is the coordinate of one projected orthogonally onto a (basis) axis defined by the other. The cosine is then the length of this projection (the

‘adjacent edge’) over one (the ‘hypotenuse’):

$$\cos(v, u) = \frac{v^T u}{\|v\|_2 \|u\|_2}$$

This makes a good similarity metric: it’s one if v and u are aligned, zero if orthogonal:

$$\cos\left(\begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix}\right) = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} = 1$$

$$\cos\left(\begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ -1 \end{bmatrix}\right) = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix} = 0$$

Practice:

Recalling that 3,4,5 and 5,12,13 are right triangles,

1. what is the cosine similarity of vectors $\begin{bmatrix} 3 \\ 4 \end{bmatrix}$ and $\begin{bmatrix} 5 \\ 12 \end{bmatrix}$, and
2. what is the cosine similarity of vectors $\begin{bmatrix} 3 \\ 4 \end{bmatrix}$ and $\begin{bmatrix} 12 \\ 5 \end{bmatrix}$?