CSE 5523: Lecture Notes 8
Linear Algebra Notation

We'll use linear algebra notation to simplify our equations.

Contents

8.1 Terms

We define matrices and vectors as arrays of real numbers:

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

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

2. $\mathbf{v}$ is a **vector** iff $\mathbf{v} \in \mathbb{R}^I$. It can define a **point** in some I-dimensional space.

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

3. $\mathbf{M}$ is a **matrix** iff $\mathbf{M} \in \mathbb{R}^{I \times J}$. It can be a **linear transform** to project points between spaces.

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

8.2 Unary Operations

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

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

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

2. **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: $\text{diag}(\begin{bmatrix} 1.8 \\ -3 \end{bmatrix}) = \begin{bmatrix} 1.8 & 0 \\ 0 & -3 \end{bmatrix}$.

3. **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: $\delta_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$.

8.3 Binary Operations

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

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

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

4. **pointwise or Hadamard product**: for all $\mathbf{M}, \mathbf{N} \in \mathbb{R}^{I \times J}$, with row and column indices i, j,

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

   (commutative)

   For example: $\begin{bmatrix} 1.8 & 12 \\ -3 & 40 \end{bmatrix} \odot \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 1.8 & 24 \\ -9 & 160 \end{bmatrix}$

5. **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)

   There are two special cases of matrix multiplication for vectors:

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

$\mathbf{v}^\top \mathbf{u} = \sum_k \mathbf{v}_{[k]} \cdot \mathbf{u}_{[k]}$

For example:

$$\begin{bmatrix} 1.8 & -3 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} = (1.8 \cdot 1) + (-3 \cdot 2) \\ = -4.2$$

This **projects** a point onto a line (gives fraction of distance of nearest point from origin).

(b) **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:

$$\begin{bmatrix} 1.8 \\ -3 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} = \begin{bmatrix} 1.8 \cdot 1 & 1.8 \cdot 2 & 1.8 \cdot 3 \\ -3 \cdot 1 & -3 \cdot 2 & -3 \cdot 3 \end{bmatrix} \\ = \begin{bmatrix} 1.8 & 3.6 & 5.4 \\ -3 & -6 & -9 \end{bmatrix}$$

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

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

This is just a repeated inner or outer product.

(*not* commutative)

For example:

$$\begin{bmatrix} 1.8 & 12 \\ -3 & 40 \\ 15 & -6 \\ 7 & 18 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} = \begin{bmatrix} (1.8 \cdot 1) + (12 \cdot 4) & (1.8 \cdot 2) + (12 \cdot 5) & (1.8 \cdot 3) + (12 \cdot 6) \\ (-3 \cdot 1) + (40 \cdot 4) & (-3 \cdot 2) + (40 \cdot 5) & (-3 \cdot 3) + (40 \cdot 6) \\ (15 \cdot 1) + (-6 \cdot 4) & (15 \cdot 2) + (-6 \cdot 5) & (15 \cdot 3) + (-6 \cdot 6) \\ (7 \cdot 1) + (18 \cdot 4) & (7 \cdot 2) + (18 \cdot 5) & (7 \cdot 3) + (18 \cdot 6) \end{bmatrix} \\ = \begin{bmatrix} 49.8 & 63.6 & 77.4 \\ 157 & 194 & 231 \\ -9 & 0 & 9 \\ 79 & 104 & 129 \end{bmatrix}$$

It applies projections to points or other projections.

7. **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:

$$\begin{bmatrix}1.1 & 4\\ -3 & 1\end{bmatrix} \otimes \begin{bmatrix}1 & 2 & 3\\ 4 & 5 & 6\end{bmatrix} = \begin{bmatrix}1.1\cdot 1 & 1.1\cdot 2 & 1.1\cdot 3 & 4\cdot 1 & 4\cdot 2 & 4\cdot 3\\ 1.1\cdot 4 & 1.1\cdot 5 & 1.1\cdot 6 & 4\cdot 4 & 4\cdot 5 & 4\cdot 6\\ -3\cdot 1 & -3\cdot 2 & -3\cdot 3 & 1\cdot 1 & 1\cdot 2 & 1\cdot 3\\ -3\cdot 4 & -3\cdot 5 & -3\cdot 6 & 1\cdot 4 & 1\cdot 5 & 1\cdot 6\end{bmatrix}$$

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

This builds points or transforms as Cartesian products of other points or transforms.

8.4 Representing factored probability models as matrix equations

Factored probability estimation can be represented as a matrix equation.

For example, our Naive Bayes model:

$$\mathbf{f}^\top = \begin{matrix} \text{apple} & \text{pear} \\ (.6 & .4) \end{matrix} \qquad \mathbf{C} = \begin{matrix} & \text{red} & \text{green} \\ \text{apple} & .67 & .33 \\ \text{pear} & 0 & 1 \end{matrix} \qquad \mathbf{S} = \begin{matrix} & \text{round} & \text{long} \\ \text{apple} & .8 & .2 \\ \text{pear} & .2 & .8 \end{matrix}$$

$$\mathsf{P}(F, c, s) = \mathbf{f}^\top (\mathbf{C}\, \delta_c \odot \mathbf{S}\, \delta_s)$$

Sample code in pandas:

```
import pandas

fT = pandas.read_csv('fT.csv')
C  = pandas.read_csv('C.csv',index_col=0)
S  = pandas.read_csv('S.csv',index_col=0)

print( fT @ (C['green'] * S['round']) )
```

Sample input file ‘fT.csv’:

```
apple,pear
.6,.4
```

Sample input file ‘C.csv’:

```
,red,green
apple,.67,.33
pear,0,1
```

Sample input file ‘S.csv’:

```
,round,long
apple,.8,.2
pear,.2,.8
```

Sample output:

```
0    0.2384
dtype: float64
```

8.5 Determinants and inversion

1. **determinant**: for all $\mathbf{M} \in \mathbb{R}^{I \times I}$, (Laplace expansion)

   $$|\mathbf{M}| = \sum_{i=1}^{I} (-1)^{i+1} \mathbf{M}_{[1,i]} \left| \begin{bmatrix} \mathbf{M}_{[2,1]} & \cdots & \mathbf{M}_{[2,i-1]} & \mathbf{M}_{[2,i+1]} & \cdots & \mathbf{M}_{[2,J]} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}_{[I,1]} & \cdots & \mathbf{M}_{[I,i-1]} & \mathbf{M}_{[I,i+1]} & \cdots & \mathbf{M}_{[I,J]} \end{bmatrix} \right|$$

   It is the volume (scaling factor) of the linear transformation defined by $\mathbf{M}$.

   It is also the volume of the parallelepiped spanned by the points in the matrix.

   For example: $\left| \begin{bmatrix} 1.8 & 12 \\ -3 & 40 \end{bmatrix} \right| = (1.8 \cdot 40) - (12 \cdot -3) = 108$.

2. **matrix inverse**: for all $\mathbf{M} \in \mathbb{R}^{I \times I}$ and $\mathbf{v} \in \mathbb{R}^{I}$,

   $\mathbf{v} = \mathbf{M}^{-1}\, \mathbf{M}\, \mathbf{v}$.

   For example: $\begin{bmatrix} 1 \\ 3 \end{bmatrix} = \begin{bmatrix} .6 & -.7 \\ -.2 & .4 \end{bmatrix} \begin{bmatrix} 4 & 7 \\ 2 & 6 \end{bmatrix} \begin{bmatrix} 1 \\ 3 \end{bmatrix}$.

   However, there is no inverse if the matrix is **singular**: $|\mathbf{M}| = 0$.

   For example: $\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} ? & ? \\ ? & ? \end{bmatrix} \left(\begin{bmatrix} a+b \\ a+b \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} \right)$.

8.6 Vector Normalization

We can normalize these vectors using an n**-norm** of a vector $\mathbf{v}$:

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

There are several useful instantiations of this:

1. The two-norm calculates the length of vector $\mathbf{v}$ as Euclidean coordinates:

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

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

$$= \sqrt{\textstyle\sum_j \mathbf{v}_j \cdot \mathbf{v}_j} \quad (4)$$

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

2. The one-norm calculates 'Manhattan distance' (a sum over vector cells):

$$\|\mathbf{v}\|_1 = \left(\sum_j (\mathbf{v}_j)^1 \right)^{\frac{1}{1}} \tag{5}$$

$$= \sum_j \mathbf{v}_j \tag{6}$$

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

3. The infinity-('inf'-)norm calculates the maximum over vector cells (largest cell dominates):

$$\|\mathbf{v}\|_\infty = \left(\sum_j (\mathbf{v}_j)^\infty \right)^{\frac{1}{\infty}} \tag{7}$$

$$= \max_j \mathbf{v}_j \tag{8}$$

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

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

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

8.7 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'):

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

This makes a good similarity metric: it's one if $\mathbf{v}$ and $\mathbf{u}$ are aligned, zero if orthogonal:

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

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

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