

First, why is this important?

By convention, we typically represent data as a matrix of values

- Columns correspond to features
- Rows correspond to observations of those features

$$D = \begin{bmatrix} x_{1,1} & \cdots & x_{1,N} \\ \vdots & \ddots & \vdots \\ x_{M,1} & \cdots & x_{M,N} \end{bmatrix}$$

where $x_{i,j}$ represents j^{th} feature value of i^{th} observation

N = number of features

M = number of observations

- ▶ Linear algebra underpins much of language modeling with neural networks, topic modeling, LSA, etc...

Vector spaces, inner & outer products

- In this class we represent data in a vector space
 - This means a data point lies in a grid (2D), cube (3D), or hypercube (4D+) ... in the general case it's N -dimensional
 - A point in space is represented by a vector
 - vector: $\mathbf{x} \in \mathbb{R}^N$
 - its scalar components: $x_i \in \mathbb{R} \quad \text{where} \quad 0 \leq i < N$
 - concretely: $\mathbf{x} = [x_0, x_1, \dots, x_{N-1}]^T$
- Inner product (dot product in this class): $\langle \mathbf{a}, \mathbf{b} \rangle = \sum_{i=0}^{N-1} a_i b_i \quad \text{where} \quad \mathbf{a}, \mathbf{b} \in \mathbb{R}^N$
- Outer product: $\mathbf{a} \otimes \mathbf{b} = \mathbf{a}\mathbf{b}^T = \begin{bmatrix} a_1 b_1 & \dots & a_1 b_N \\ \vdots & \ddots & \vdots \\ a_N b_1 & \dots & a_N b_N \end{bmatrix}$