Machine Learning for Engineering and Science Applications

Why Linear Algebra?

Scalars, Vectors, Tensors

Why linear algebra is useful

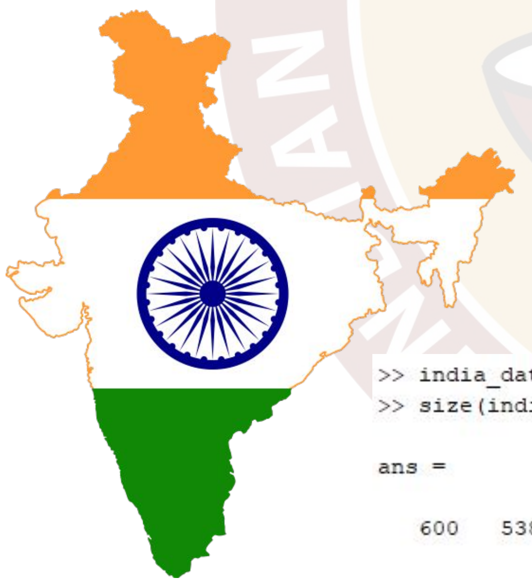
- In many Machine Learning algorithms, the input and the output are both represented as vectors
  - Maps, therefore, require matrices

- By vectors we simply mean a collection of numbers

- Part of the problem is to convert a seemingly qualitative input (such as a picture, sound, colour, etc) into a number

- Let us see an example….

From image to vector



```
>> india_data=imread('india.png');
>> size(india_data)

ans =

   600   538     3
```

255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	240
255	255	255	255	255	255	255	255	255	255	231	80
255	255	255	255	255	255	255	255	255	242	78	51
255	255	255	255	255	255	255	255	255	131	51	51
255	255	255	255	255	255	255	255	225	54	51	51
82	140	209	254	255	255	255	255	108	51	51	51
51	51	51	89	180	227	209	120	51	51	51	51
51	51	51	51	51	51	51	51	51	51	51	51
51	51	51	51	51	51	51	51	51	51	51	51
51	51	51	51	51	51	51	51	51	51	51	51
51	51	51	51	51	51	51	51	51	51	51	51
51	51	51	51	51	51	51	51	51	51	51	51
51	51	51	51	51	51	51	51	51	51	51	51

https://upload.wikimedia.org/wikipedia/commons/2/27/MnistExamples.png

https://upload.wikimedia.org/wikipedia/commons/thumb/0/05/India_geo_stub.svg/538px-India_geo_stub.svg.png

Notation

- Scalar : Single number.

  Example : *Let* $\alpha \in \mathbb{R}$, *be the learning rate*

  Let $n \in \mathbb{N}$, *be the number of hyperparameters*

Vector : In ML, array of numbers.

Example : *Let* $\vec{x} \in \mathbb{R}^n$, *be the input vector.*

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix}$$

Matrix : In ML, 2-D array of numbers.

Example : *Let* $W = \mathbb{R}^{m \times \mathrm{n}}$ *be the matrix of weights*

Tensors : In ML, array of numbers with dimensions greater than 2

Example : $A_{i,j,k}$

Scalars, Vectors, Matrices, Tensors

- Scalar (0th order tensor)

$$\alpha = 3$$

Vector (1st order tensor)

$$\vec{v} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}$$

Dimension of the example vector is?

Matrices, Tensors

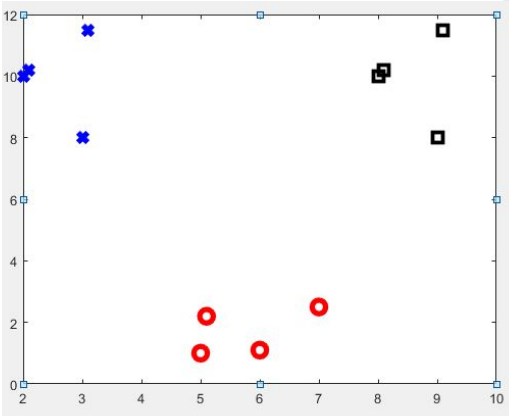
- Matrix (2nd order tensor)

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 5 \end{bmatrix}$$

Tensors (3rd and higher order tensors)

- Example: Colour images, Video data

Implications of tensor representation



- We represent both vectors and transformations as tensors
  - Transformations between vectors -> vectors are naturally represented as matrices
- Could be high dimensional representations
  - Need algorithms that work well in high dimensions
- Lets us go back and forth between images and numbers
  - Very useful for engineering applications