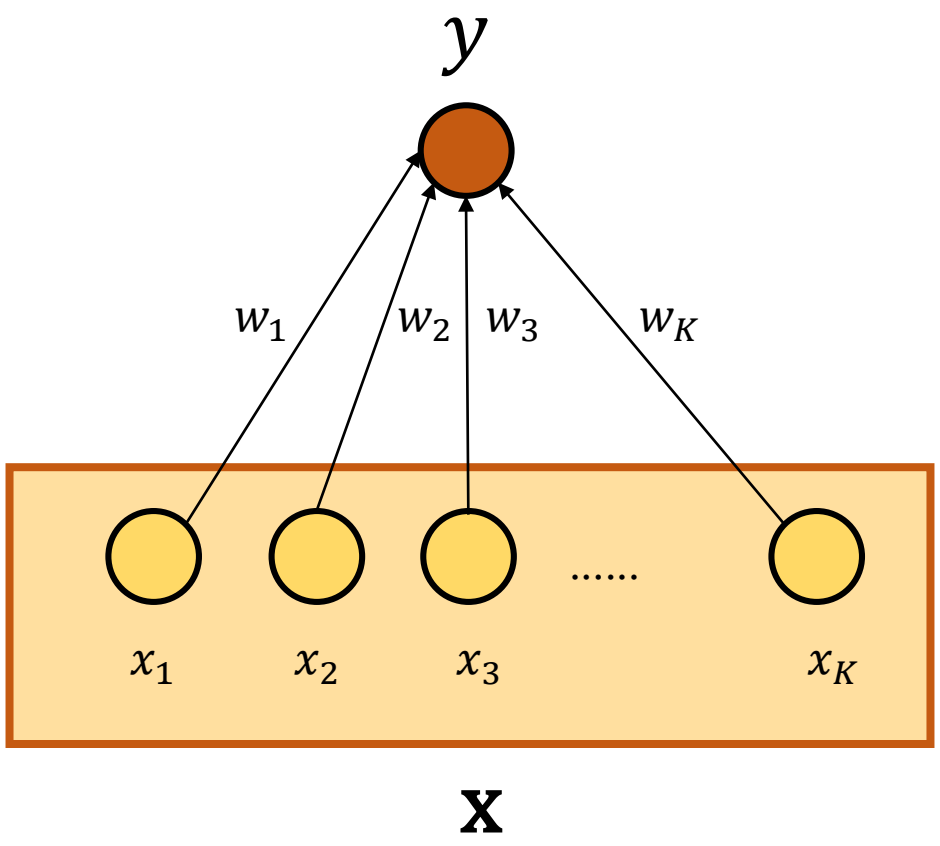
MATLAB Practice

Artificial Intelligence

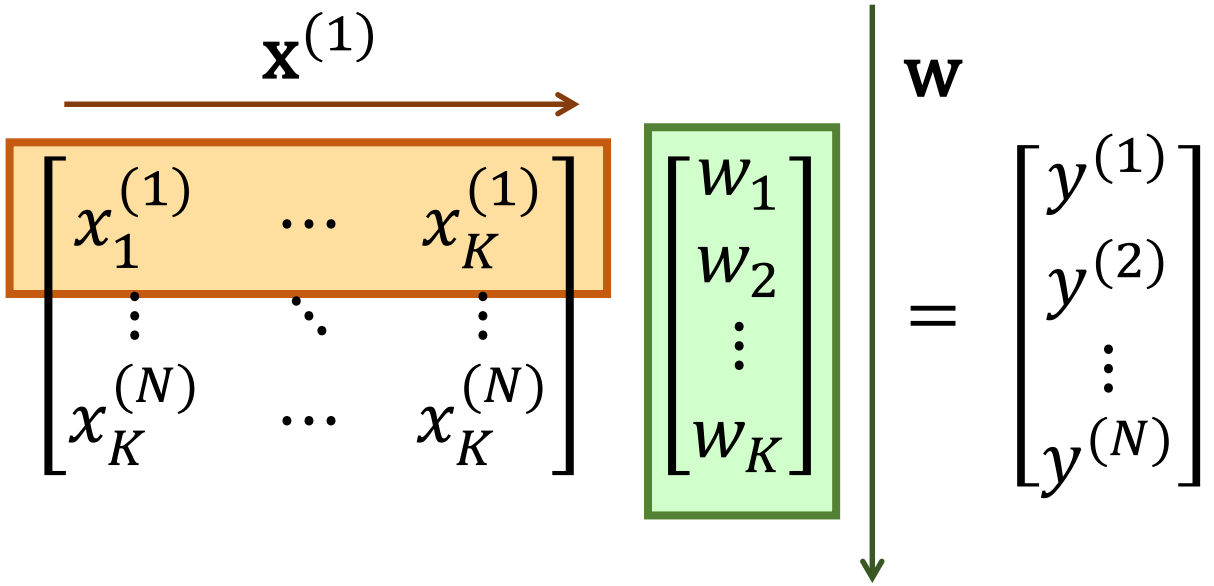
2014.04.24 Practice 2

Simple Neural Net



- Input : K dimension vector $\mathbf{x} \in \mathbb{R}^K$
- Output : Scalar value y
- Weight : K dimension vector $\mathbf{w} \in \mathbb{R}^K$
- Relation : $y = w_1 x_1 + w_2 x_2 + \cdots + w_K x_K = \sum_{k=1}^{K} w_k x_k$
- Objective : Find $\mathbf{w}$ for given data set $S = \{(\mathbf{x}^{(n)}, y^{(n)})\}_{n=1}^{N}$

Simple Neural Net



$$\mathbf{X} \cdot \mathbf{w} = \mathbf{y}$$

Simple Neural Net

- But in general, $\mathbf{X}$ is not square matrix
  - ✓ Use psuedoinverse instead : $w = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$

- What if $\mathbf{X}^T\mathbf{X}$ is not invertible?
  - ✓ Approximate : $w = (\mathbf{X}^T\mathbf{X} + \varepsilon\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$
  - ✓ Try various value for ε !

Simple Neural Net

- Use custom data for practice

  ✓ Ex) $\mathbf{X} = \begin{bmatrix} 1 & 2 & 3 \\ 6 & 5 & 4 \\ 7 & 8 & 9 \end{bmatrix}$, $\mathbf{y} = \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix}$

- No submission required
  ✓ But might be very useful warm-up for final project

Final Project Introduction

- Formal description will be announced at next class (4/29)
- Implement model based on neural network to solve real data real data (ex : MNIST)
- MATLAB, C++, Python etc.