

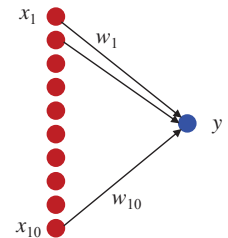
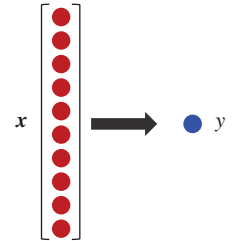
# AI and Speech Homework #3

## 목표

- Task = regression
- System = neural network
  - 가장 간단한 NN 구조 사용
- Supervised learning을 C로 구현
  - Backpropagation에 따라 gradient 계산

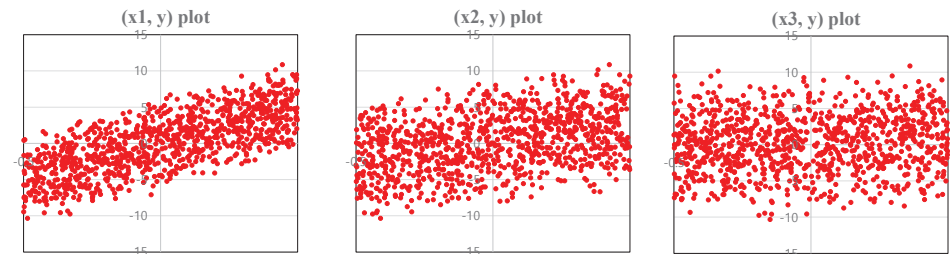
## Problem Setting

- Training dataset
  - Input : 10-D vector of floating-point values
    - $-0.5 \leq x_k \leq 0.5$
  - Output : a floating-point value
    - Example : 10가지 개인 정보 → 신용도 결정
  - 1000 data of  $(x_1, x_2, \dots, x_{10}, y)$  from 1000 people
- Target
  - $(x_1, x_2, \dots, x_{10})$  와  $y$  사이의 관계 learning
- Neural network
  - No hidden layers
  - No bias
  - Linear activation function
  - $y = \sum_{k=1}^{10} w_k x_k$



## Training Data 특성

Scatter diagram



Different input-output dependency

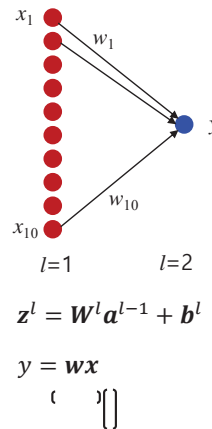
## NN Training

- Training by stochastic gradient descent(SGD) algorithm
  - Weight initialization
    - Uniform between -0.5 and 0.5
    - C code :  $w[k] = ((\text{float})\text{rand}()) / \text{RAND\_MAX} - 0.5$
  - Batch size = 1
  - 1 epoch  $\rightarrow$  1000번 update
  - $\eta = 0.1$

5

## NN Training

- Update process
  - Forward
    - 각 input  $x$  에 대하여 output  $y = wx$  계산
  - Output error
    - $C = \frac{1}{2}(y^* - y)^2$
    - $\delta^{l=2} = \frac{\partial C}{\partial(z=y)} = y - y^*$
  - Gradient
    - $\frac{\partial C}{\partial W^2} = \delta^2 (a^1)^T = (y - y^*) x^T$   
 $\left( \quad \right) \quad \left( \quad \right)$
    - $\frac{\partial C}{\partial w_k} = x_k (y - y^*)$
  - Update
    - $w_k \leftarrow w_k - \eta \frac{\partial C}{\partial w_k} = w_k + \eta x_k (y^* - y)$



6

## C code

- File open, read, write (text file)
 

```
float input[DIM] = {0.0,}, target ;
fopen_s(&fin, "trainingDB.dat", "rt");
fopen_s(&fo, "output.dat", "wt");

for (k = 0; k < DIM; k++)
    fscanf(fin, "%f", input + k);

fscanf(fin, "%f", &target);

for (k = 0; k < DIM; k++)
    fprintf(fo, "%10.7f", weight[k]);
fprintf(fo, "\n");
```

7

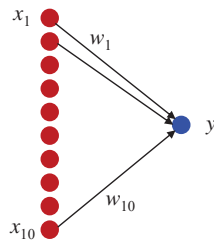
## C code

- 전체 구조
  - for (input index)
    - Data read
    - Output computation
    - Gradient computation
    - Weight update
    - Weight and cost write

8

## Check-Point

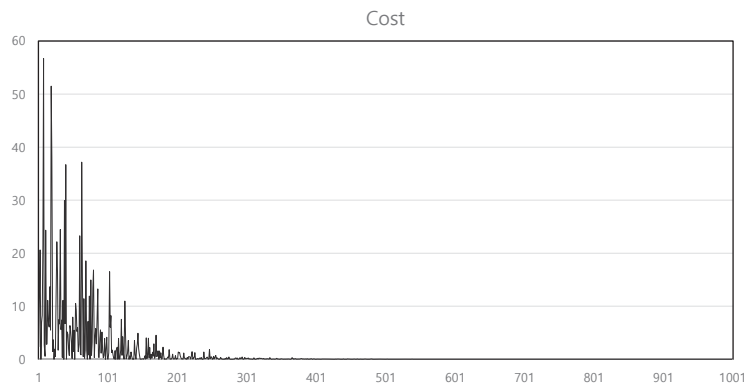
- 500번째  $x$  에 대한  $C = \frac{1}{2}(y^* - y)^2 \cong 0.00057 \dots$
- 1000번째  $x$  에 대한  $C = \frac{1}{2}(y^* - y)^2 \cong 0.0000012 \dots$
- After 1000 updates, weight의 크기 순서
  - $w_1 > w_8 > w_2 > \dots > w_{10}$



9

## 결과 분석

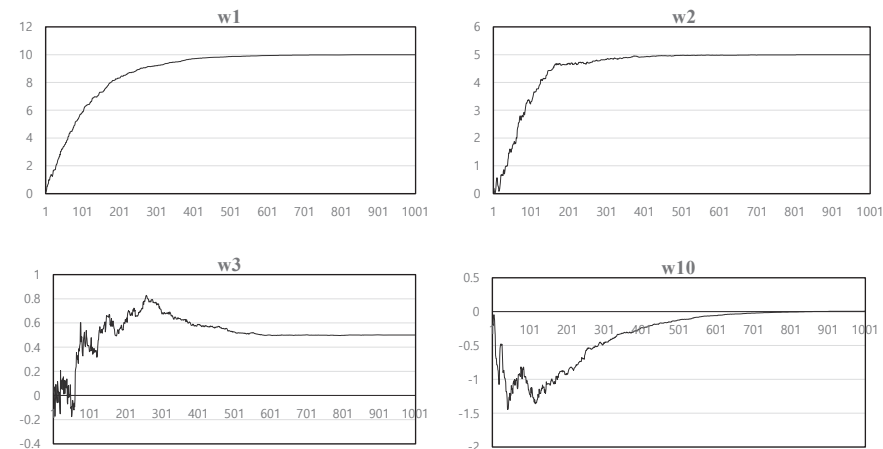
- 각  $x$  에 대한  $C = \frac{1}{2}(y^* - y)^2$



10

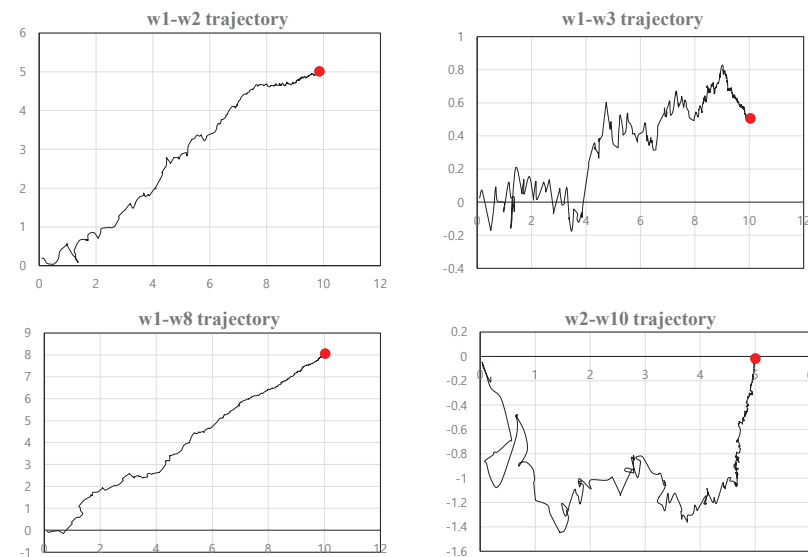
## 결과 분석

- Weight trajectory



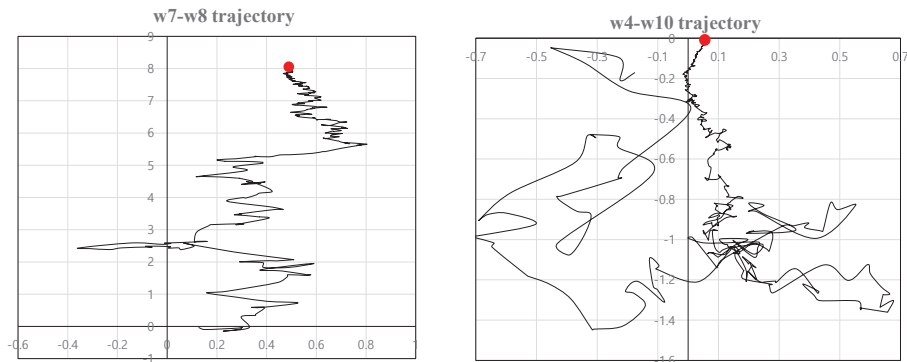
11

## 결과 분석



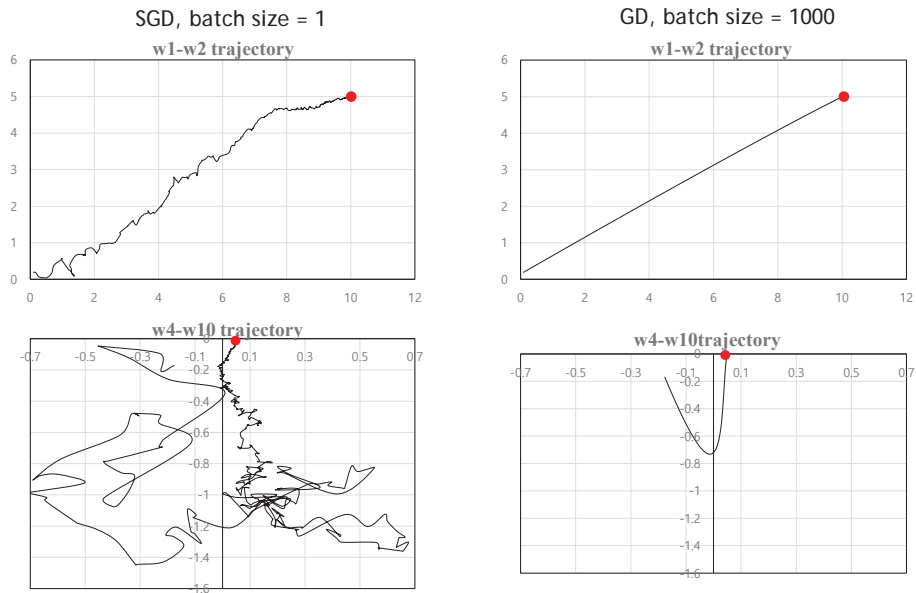
12

## 결과 분석



13

## 결과 분석 : Batch Size



## 과제 제출

- Due : 5월 31일(월) 23:59 13주차
- File name : AI\_Speech\_HW3\_이름\_학번.hwp, docx, pdf
- 제출 내용
  - C code
  - Learning output : weight, cost
  - 다양한 실험 결과와 설명

15