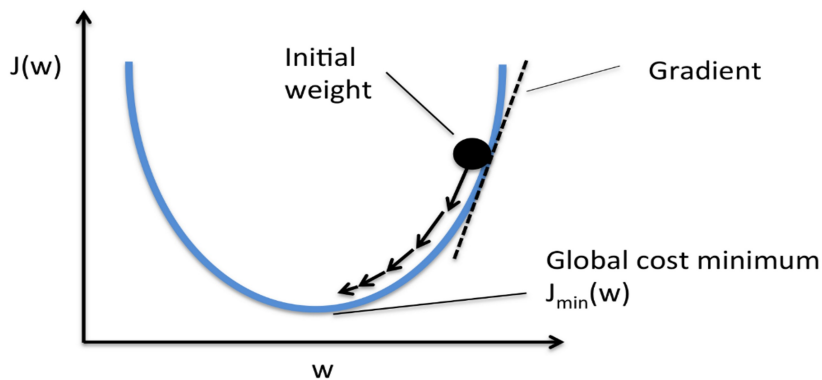


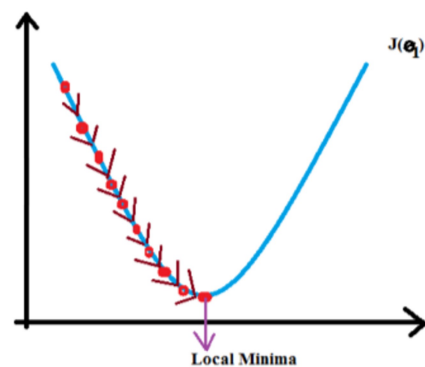
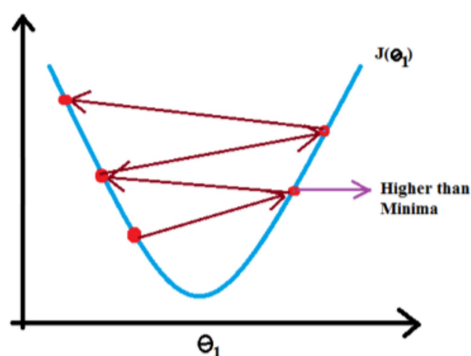
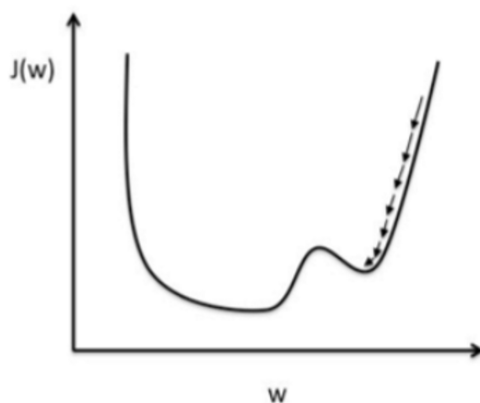
## 6. Rate , overfitting , regularization , Dataset

2019년 4월 4일 목요일 오후 5:11

### 1. Rate

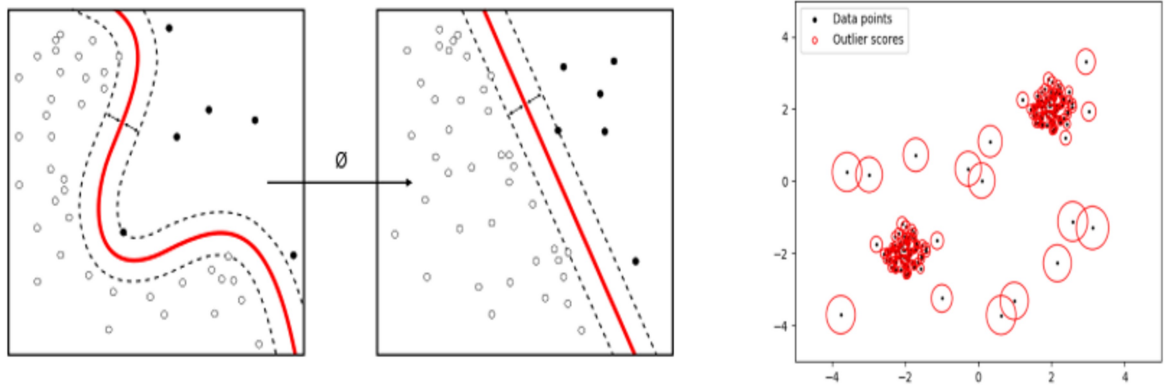
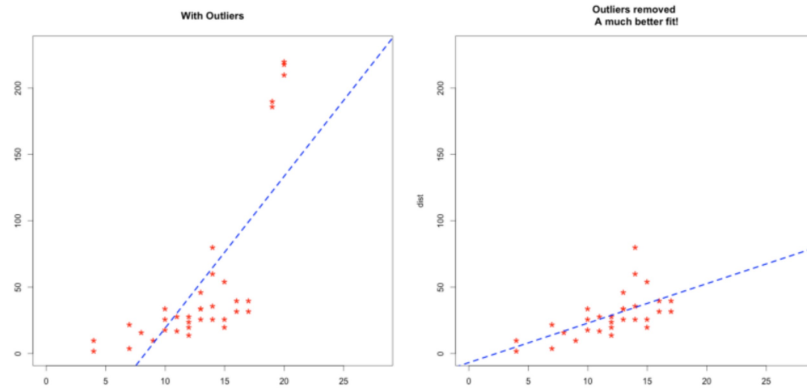


```
# Minimize error using cross entropy
learning_rate = 0.001
cost = tf.reduce_mean(-tf.reduce_sum(Y*tf.log(hypothesis), reduction_indices=1)) # Cross entropy
optimizer = tf.train.GradientDescentOptimizer(learning_rate).minimize(cost) # Gradient Descent
```

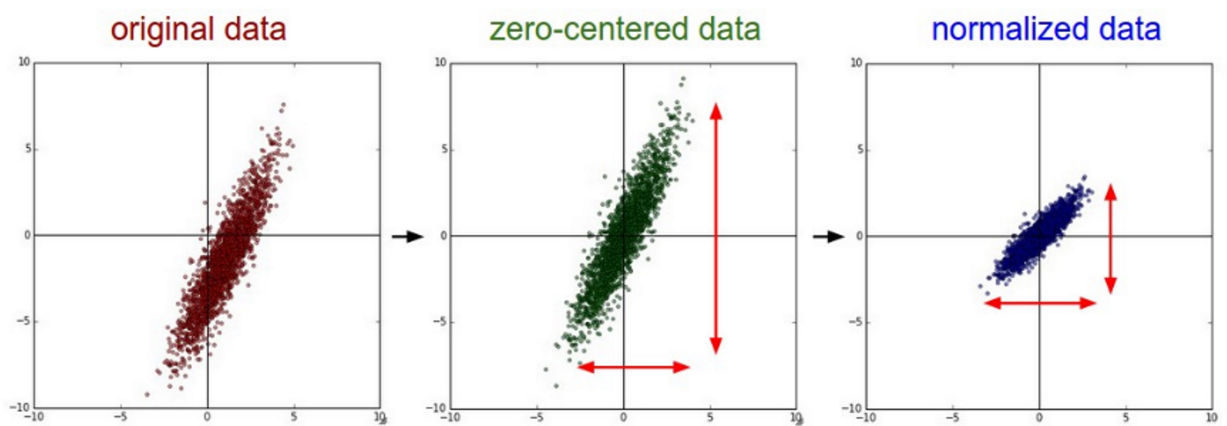


### 2. Data preprocessing

#### a. Outlier



#### a. Normalized (정규화)



##### ○ Zero-centeren

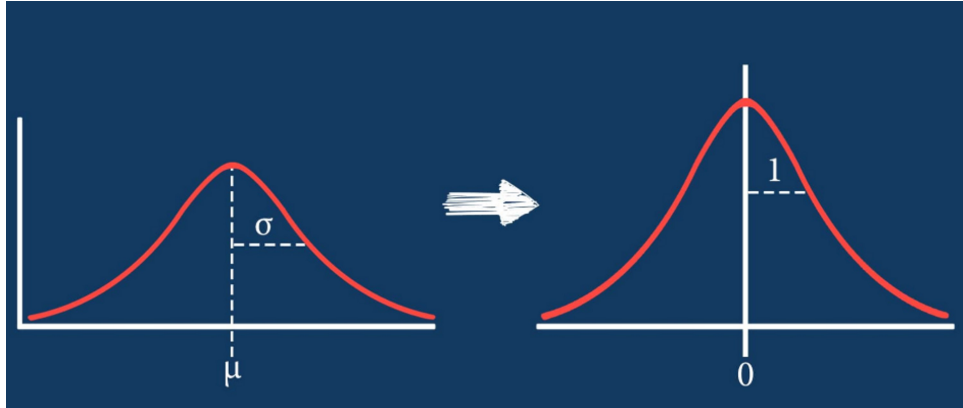
: 각 데이터 포인트에서 평균을 제로 중심으로 만들기 위해 평균을 뺀 프로세스

##### ○ Normalized (정규화)

- 전체 구간을 **표준값의 범위로 변환** ( -1~1, 0~1, 0~100 ... )하여 데이터를 관찰하는 방법,
- 데이터 군 내에서 특정 **데이터가 가지는 위치**를 확인

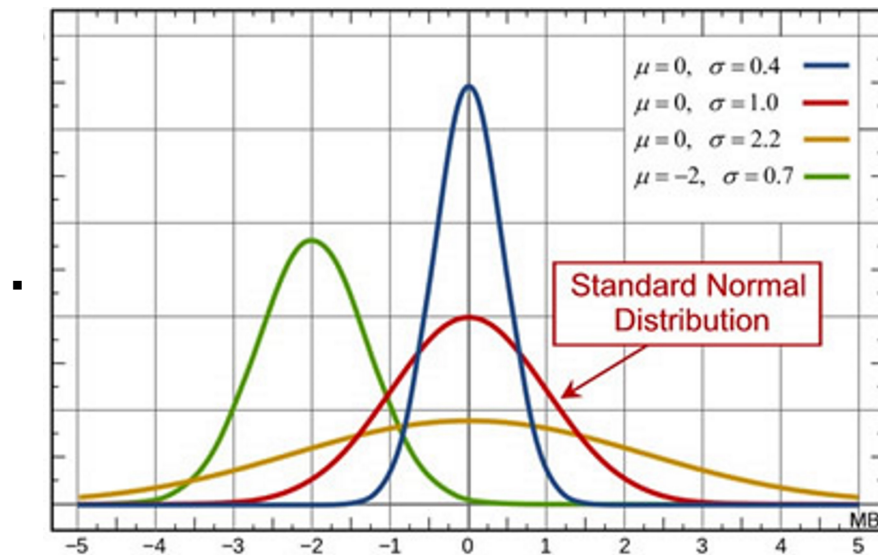
- $$x_{new} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

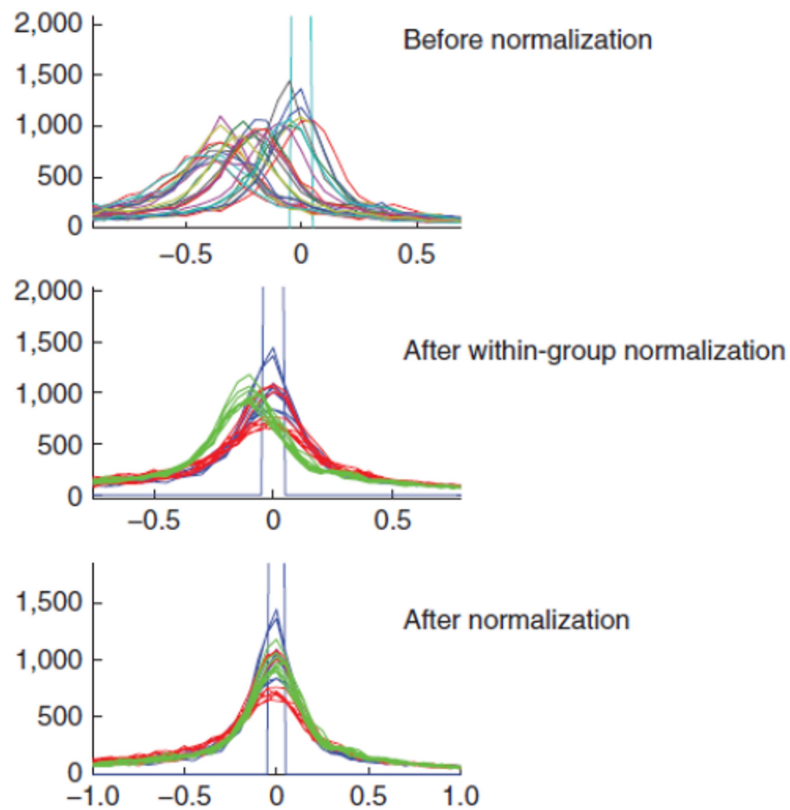
- standardization (표준화)



- 평균을 기준으로 얼마나 떨어져 있는지를 나타내는 값,
    - 2개 이상의 단위가 다른 대상을 비교가능하게 한다.

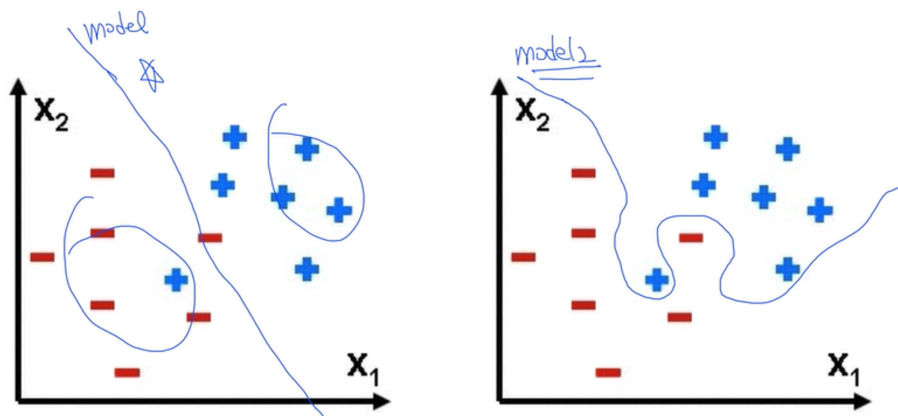
- $$x'_j = \frac{x_j - \mu_j}{\sigma_j}$$





## 1. **Overfitting**

### Overfitting



### Solutions for overfitting

- More training data!
- Reduce the number of features
- Regularization

## 2. **Regularization (일반화, 규제)**

## Example: Linear regression (housing prices)



$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$  가 있고 **overfitting** 문제가 있다고 가정하자.

그리고 우리는  $\theta_3$ 과  $\theta_4$ 의 영향을 줄여 overfitting 문제를 개선하고자 한다 어떻게 하면 될까?  
가설함수의 표현을 바꾸거나 몇몇 특징을 포기하는 대신 **비용함수를 수정**함으로써 과적합을 개선할 수 있다.

$$\min_{\theta} \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 + 1000 \cdot \theta_3^2 + 1000 \cdot \theta_4^2$$

2차원의 평면에 그려지는 그래프에서  **$\theta$ 항의 차수가 크면 클수록 구불구불**해지는 성질이 커진다. 따라서 이런 성질을 최소화하기 위해서 각각 1000을 곱해줬다. 이렇게 비용함수를 통해서 나오게 되는  $\theta_3$ 과  $\theta_4$ 의 계수  $\theta$ 는 0에 가까워 지게 되고 구불구불하게 나타나는 성질은 줄어들게 된다.

$$\theta_0 + \theta_1 x + \theta_2 x^2$$

위 예에서는  $x$ 의 3,4차 항의 계수인  $\theta$ 들만 regularize를 했지만 사실은 모든 특징 즉 **모든 계수**에 대해서 regularize를 해야한다고 한다.

$$\min_{\theta} \frac{1}{2m} \left[ \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 + \lambda \sum_{j=1}^n \theta_j^2 \right]$$

**$\lambda$ 가 너무 클 경우** 어떻게 될까? 아마 상수항을 제외한 모든 항의 계수는 0에 가까워 질 것이고 이로 인해 도출되는 비용함수는 거의 **상수항**에 가까워 지게 되어서 **underfit**이 될 것이다. 따라서 Regularization을 잘 하기 위해서는 **적절한  $\lambda$ 선택**이 중요하다.

## Regularization

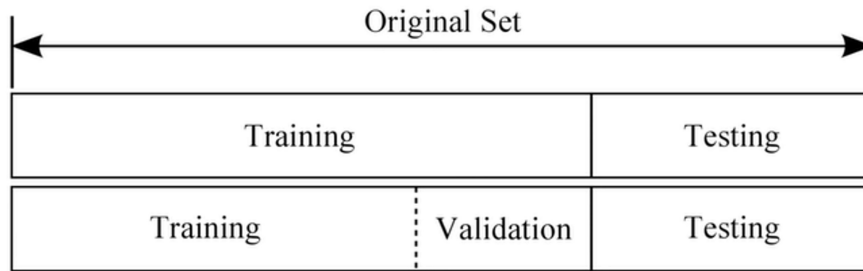
- Let's not have too big numbers in the weight

$$\mathcal{L} = \frac{1}{N} \sum_i \mathcal{D}(s(w x_i + b), L_i) + \lambda \sum W^2$$

Annotations:   
 - **LOSS** points to the first term.   
 - **TRAINING SET** points to the data  $(x_i, L_i)$ .   
 - **regularization strength** points to the coefficient  $\lambda$ .

### 3. Dataset

# Training, validation and test sets

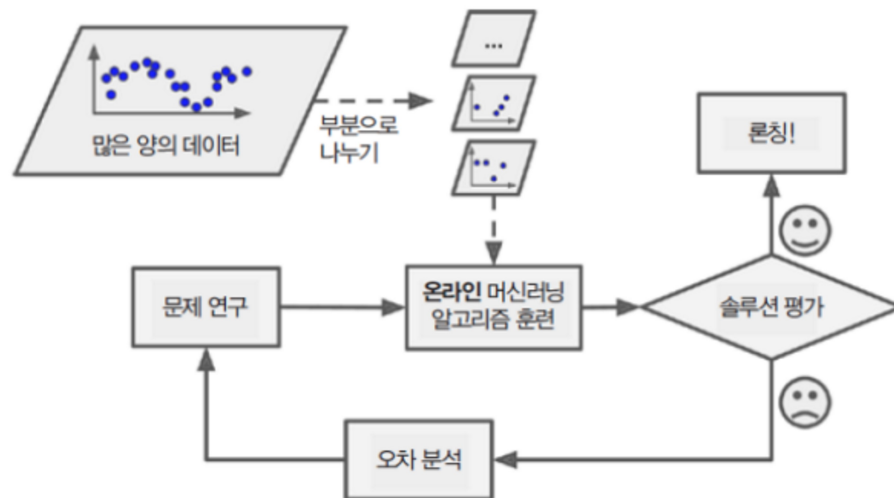


## - Training(training, validation), Testing

- Training : 교과서
- Validation : 모의고사
- Testing : 실전

## - **Online learning :**

- 있는 데이터에 추가해서 training
- 온라인 학습 online learning 에서는 데이터를 순차적으로 한 개씩 또는 미니배치 mini-batch 라 부르는 작은 묶음 단위로 주입하여 시스템을 훈련
- 예 : 주식가격, 시계열 데이터



## - **Accuracy (정확도) :**