강원대학교
AI 소프트웨어학과

머신러닝2
- 다중 회귀분석(Complex)-

KNU
강원대학교

01 다중회귀 회귀(Multiple Linear Regression)

- 종속변수를 설명하기 위해서 두 개 이상의 독립변수가 사용되는 선형회귀모형을 다중선형 회귀모형이라 함

표본지점의 예금유치액

지점번호	홍보비용	직원수	지점성과	고객수	대출정도	유동인구	예금유지액
1	40	15					87
2	50	20					108
3	30	14					69
4	60	22					135
5	70	30					148
6	60	24					132
7	30	16					73
8	60	20					128
9	20	14					50
10	80	32					170

Q) 홍보비용, 직원수 등등이 예금유치액에 어떤 영향을 미치는가?

01 다중회귀 회귀(Multiple Linear Regression)

다중회귀분석

- 다수의 요소를 가지고 y를 예측하고 싶을 때, 이를 다중 선형 회귀분석 이라고 함

$$\hat{y}_i = \beta_0 + \beta_1 x_1 + \beta_1 x_2 + \cdots \beta_n x_n$$

- 가설(Hypothesis) 공부 시간과 시험점수 간의 회귀 모델
- Y=가설(Hypothesis)를 가장 잘 표현 할 수 있는 임의의 선을 그려 가장 적절한 값을 찾음

01 다중회귀 회귀(Multiple Linear Regression)

<단순선형회귀모형>

모형 : $y_i = \beta_0 + \beta_1 x_{i1} + \varepsilon_i$

가정 : $\varepsilon_i \sim iidN(0, \quad \sigma^2)$

<다중 선형회귀모형>

모형 : $y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \varepsilon_i$

가정 : $\varepsilon_i \sim iidN(0, \quad \sigma^2)$

최소제곱법(OLS)은 회귀모형의

오차 제곱의 합 $SSE = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2$ **을 최소로**

하는 회귀계수를 이들의 추정치로 하는 것

01 다중회귀 회귀(Multiple Linear Regression)

모형의 타당성&신뢰성 검정

- 1) 선형성 → 예측하고자 하는 종속변수 y와 독립변수 x간의 선형성을 이루어야 함
  → 비선형인 데이터에서 이러한 문제가 발생함
- 2) 등분산성 → 오차의 분산이 같다는 것을 의미하고 특정한 패턴 없이 고르게 분포하는 것을 의미함
  → 변수의 개수가 많아질수록 일부 구간에서 잔차 분산이 커질 위험이 존재함
- 3) 독립성 → 오차 사이에는 서로 영향을 주지 않으며 오차간의 상관 없이 독립적이어야 함
  → 시간공간적으로 연결되어 있으면 위배 가능
- 4) 정규성 → 잔차가 정규성을 만족하는지 여부 (잔차가 평균이 0인 정규분포)
  → 변수 많아질수록 이상치 영향 커짐 → 잔차 정규성 위배 가능성 ↑
- 5) 다중공선성 → 회귀 모델에서 두 개 이상의 독립변수가 서로 높은 상관관계가 있는 상황
  → 변수 많아질수록 상관관계 높은 변수 포함 가능성 ↑

01 다중회귀 회귀(Multiple Linear Regression)

<단순선형회귀모형>

모형 : $y_i = \beta_0 + \beta_1 x_{i1} + \varepsilon_i$

가정 : $\varepsilon_i \sim iidN(0, \quad \sigma^2)$

<다중 선형회귀모형>

모형 : $y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \varepsilon_i$

가정 : $\varepsilon_i \sim iidN(0, \quad \sigma^2)$

경사하강법은 MSE(또는 SEE)를 최소화하기 위한 일반적 최적화 방법 - 계산 자체에는 특별한 분포 가정이 필요 없음

비용 함수 $MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$ **최소화로**

하는 반복적 접근 방식으로 회귀계수를 이들의 추정치로 하는 것

SSE는 데이터 세트의 모든 관측치에 대한 전체 오류를 제공

MSE는 관측치 수에 걸쳐 총 오류를 평균화

배치크기나 데이터크기가 바뀌어도 스케일이 일정해서 학습률 튜닝이 쉬운 장점

01 다중회귀 회귀(Multiple Linear Regression)

최적화 방법 : 추정량(OLS) VS 최적화 알고리즘(GD)

- **같은 목표 → 최소의 잔차값을 가지는 값을 구하는 것**

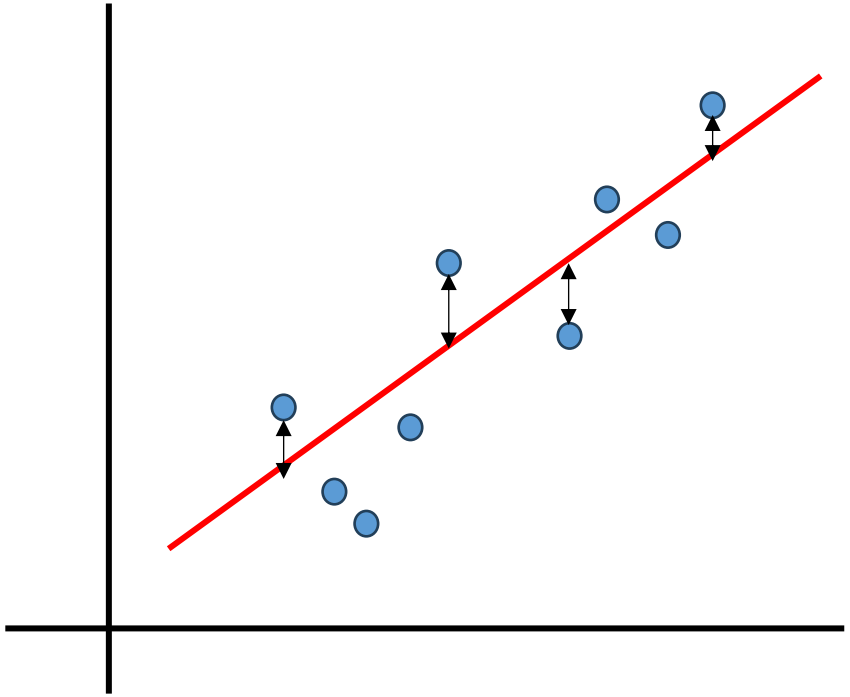
- **OLS : 한 번에 전체 데이터를 써서 정확한 해를 구함**
- **지도를 완벽히 측량해 모든 방향쌍을 다 따져 봄 → 지도가 완성되면 곧장 정상으로 가는 방법**
  → 지도가 단순하고, 범위가 작은 경우

- **GD : 손실의 기울기만 계산해 조금씩 내려감(근사해)**
- **지도를 이동하며 천천히 나아가는 방법**
  → 지도가 복잡하고, 범위가 경우

01 다중회귀 회귀(Multiple Linear Regression)

Cost Function (OLS)

→ 차원이 증가할 수 록 하나의 선으로 데이터를 표현할 수 없고, 최적의 회귀식을 찾기 어려움

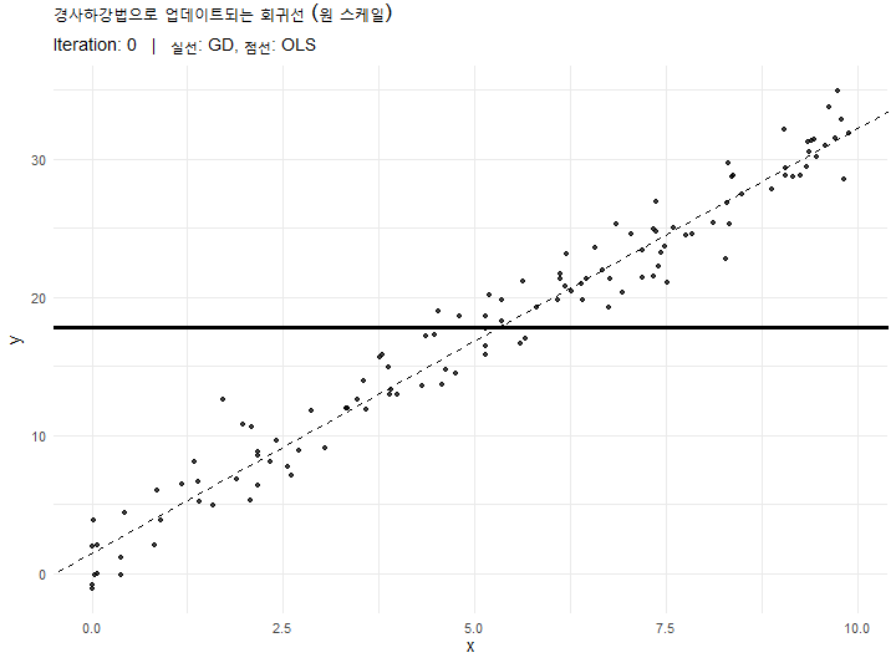


Cost Function (GD)

MSE를 사용해 기울기(Gradient)를 계산해 가중치를 반복적으로 갱신

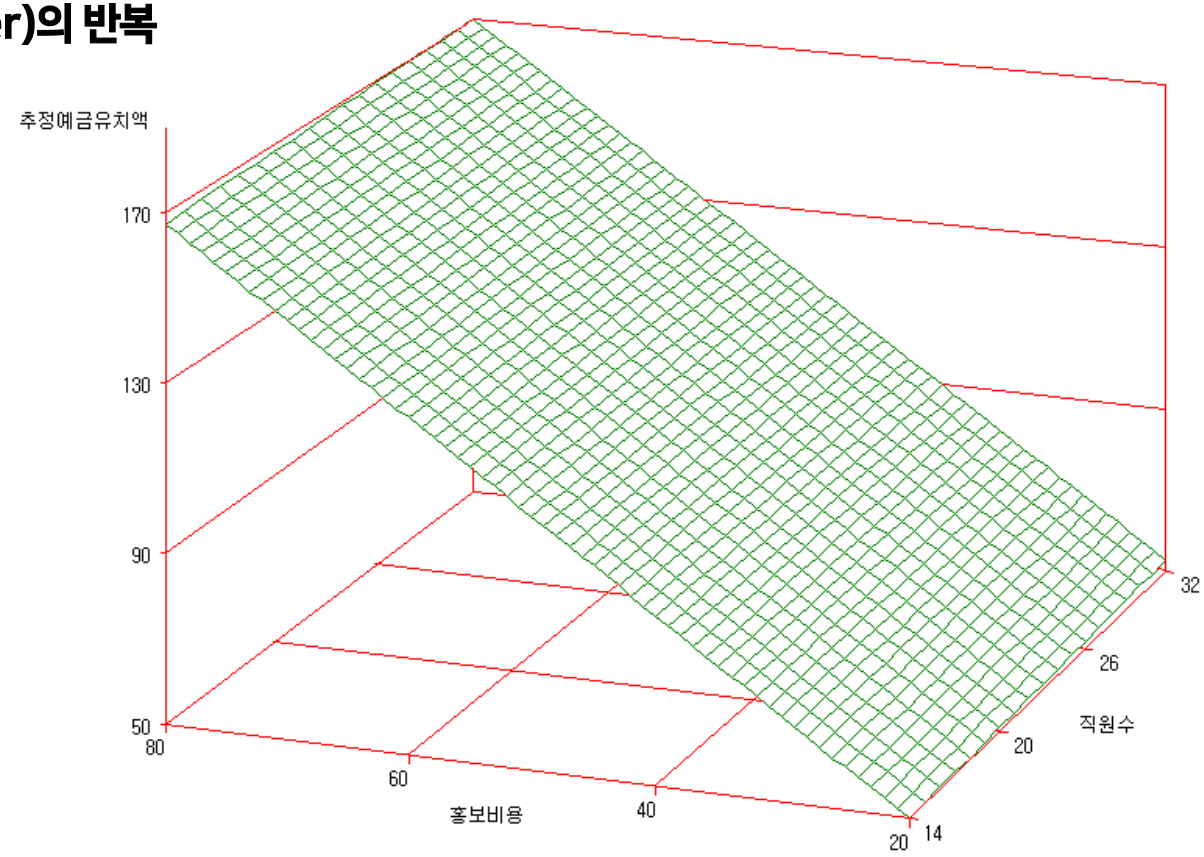
→ 대규모 데이터, 다변량 회귀분석에 사용

→ 초기값과 학습률 설정에 따라 성능과 속도가 달라짐



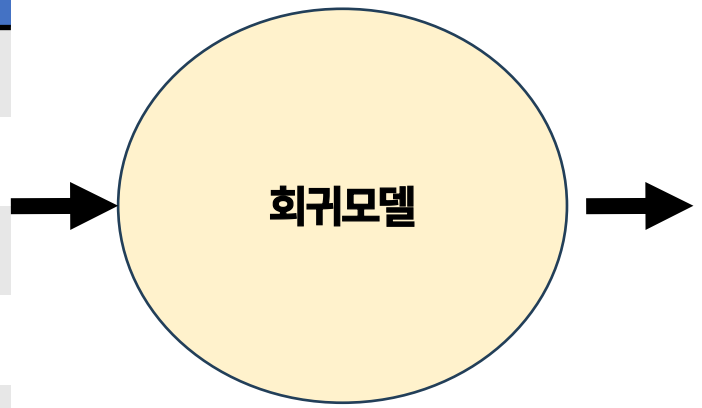
01 다중회귀 회귀(Multiple Linear Regression)

- 복잡한 문제에 있어 단순히 OLS로 판단할 수 없음
- GD를 활용해 최적의 초평면을 찾아주는 것이 필요
- 학습 프로세스는 (Forward → Loss → Backward → Optimizer)의 반복



01 다중회귀 회귀(Multiple Linear Regression)

독립1	독립2	독립3	종속
…	…	…	…
…	…	…	…
…	…	…	…
…	…	…	…
…	…	…	…



통계(회귀)에서 Parameter 표기법

$$y_i = \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_0$$

회귀 계수(Regression Coefficient)

→ 잔차를 최소화

머신러닝/딥러닝에서 Parameter 표기법

$$y_i = W_1^T x_1 + W_2^T x_2 + W_3^T x_3 + b$$

가중치(Weight) **바이어스(Bias)**

→ 손실(loss)를 최소화

01 다중회귀 회귀(Multiple Linear Regression)

- 내적 : 같은 위치끼리 곱해서 모두 더한 값
- 두 벡터의 곱을 활용해 방향성과 크기를 하나의 숫자로 알려주는 연산

$$\hat{y}_i = W_1^T x_1 + W_2^T x_2 + W_3^T x_3 + b$$

$$W_1 = \begin{bmatrix} 0.6 \\ 0.8 \\ 0.1 \\ 0.2 \\ -1 \\ 0.7 \end{bmatrix} \qquad x_1 = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{bmatrix} \qquad W_1^T x_1 = [0.6, 0.8, 0.1, 0.2, -1, 0.7] \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{bmatrix} = 2.5$$

다중회귀 회귀(Multiple Linear Regression)

Forward

- 입력값으로 하나의 y값을 만드는 단계
- W는 방향(법선), b는 원점으로 부터의 평행 이동을 정의함 → $w^T x_n + b + \varepsilon_n = 0$

$$H(W,b) = W_1 x_1 + b + \varepsilon_n \longrightarrow score = W(hours) + b + \varepsilon_n$$

$$y_i = \beta_1 x_1 + \beta_0 + \varepsilon_n \longrightarrow score = \beta_1(hours) + \beta_0 + \varepsilon_n$$

다중회귀 회귀(Multiple Linear Regression)

비용 함수(Cost function) - Loss값 도출 단계

- 가설(Hypothesis)에서 세워진 식을 통해 예측값을 도출함
- 오차 = 실제값 - 예측값 → 음수, 양수 모두를 포함하고 있으므로 제곱해 더함
- MSE를 최소로 만드는 W와 b를 찾아서 회귀분석 식을 도출함

hours(x)	2	3	4	5
실제값	25	50	42	61
예측값	27	40	53	66
오차	-2	10	-7	-5

$$\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = (-2)^2 + 10^2 + (-7^2) + (-5^2) = 178$$

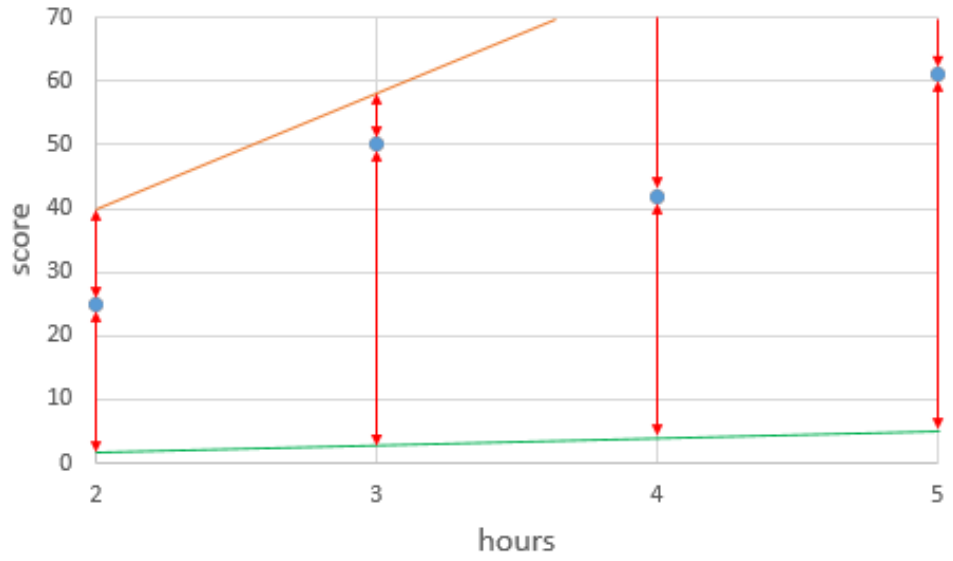
$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \frac{178}{4} = 44.5$$

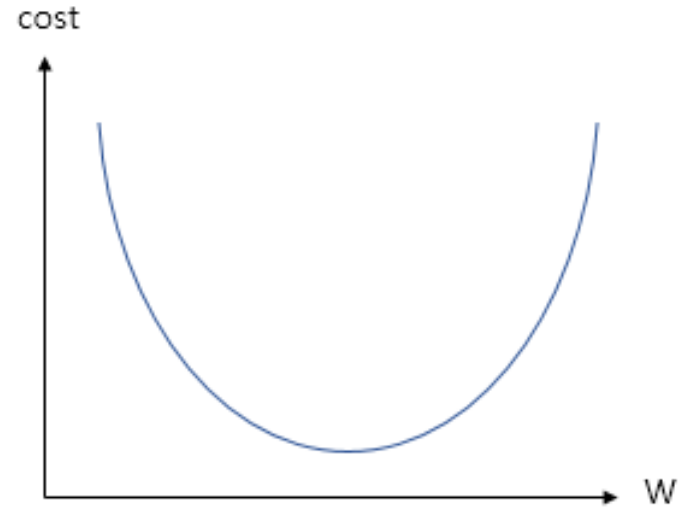
02 지도학습(Supervised Learning)

다중회귀 회귀(Multiple Linear Regression)

Backward(기울기 계산) – 어떻게 손실함수를 최소화 할지

- 학습 프로세스는 Forward → loss → Backward → Optimizer 업데이트
- 이를 머신러닝/딥러닝에서 학습이라고 부름
- Y=Wx+b에서 W의 크기가 지나치게 높거나 낮을 때, 오차가 커지는 것을 알 수 있음

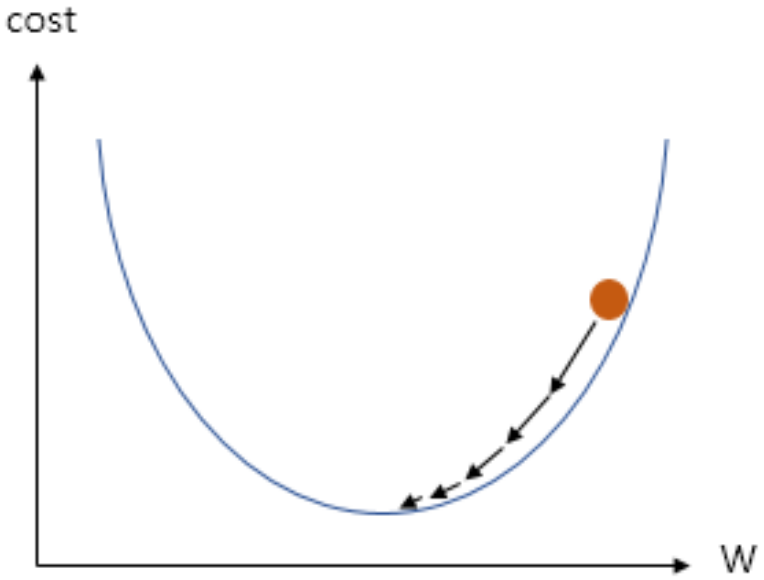


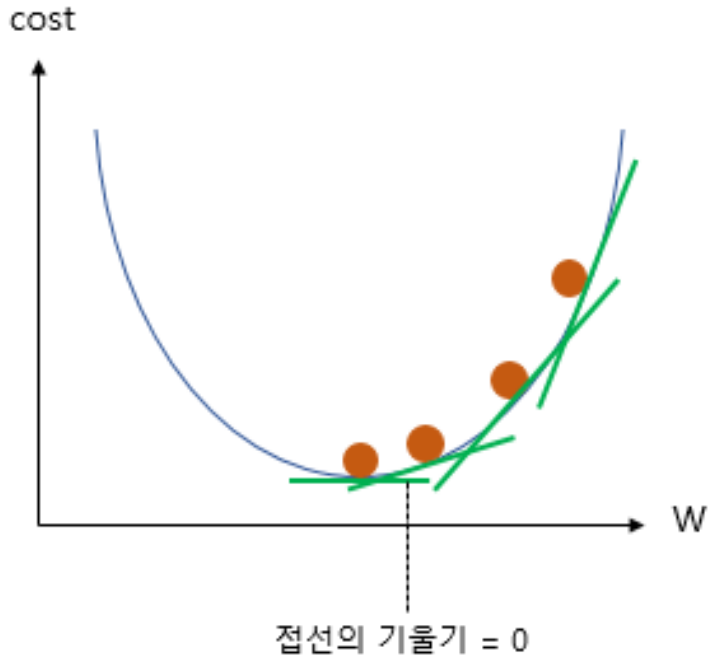


다중회귀 회귀(Multiple Linear Regression)

Backward(기울기 계산)

- 기울기 W가 무한대로 커지면 커질수록 cost값 또한 무한대로 커지고, W가 무한대로 작아져도 cost값이 무한대로 커짐
- Cost가 가장 최소값을 가지게 하는 W를 찾아야 하는 것이 목적이므로 볼록한 맨 아래 부분의 W값을 찾아야 함
- 임의의 랜덤값 W를 정하고 볼록한 부분으로 향해 점차 W 값을 수정해감
- 접선의 기울기가 0이 되는 지점이 Cost가 최소화가 되는 지점





다중회귀 회귀(Multiple Linear Regression)

Backward(기울기 계산)

- 기울기 W가 무한대로 커지면 커질수록 cost값 또한 무한대로 커지고, W가 무한대로 작아져도 cost값이 무한대로 커짐
- Cost가 가장 최소값을 가지게 하는 W를 찾아야 하는 것이 목적이므로 볼록한 맨 아래 부분의 W값을 찾아야 함
- 임의의 랜덤값 W를 정하고 볼록한 부분으로 향해 점차 W 값을 수정해감
- 접선의 기울기가 0이 되는 지점이 Cost가 최소화가 되는 지점

$$Cost = L(w,b) = \frac{1}{n}\sum_{i=1}^{n}(y_i - (b + w_1 x_i))^2$$

$$y_i = w_0 + w_1 x_{i1} + w_2 x_{i2} + w_3 x_{i3} + b$$

$$\frac{\partial MSE}{\partial w_1} = -\frac{2}{n}(y_i - \hat{y}_i) \cdot x_{i1}$$

$$\frac{\partial MSE}{\partial w_2} = -\frac{2}{n}(y_i - \hat{y}_i) \cdot x_{i2}$$

$$\frac{\partial MSE}{\partial w_3} = -\frac{2}{n}(y_i - \hat{y}_i) \cdot x_{i3}$$

$$\frac{\partial MSE}{\partial b} = -\frac{2}{n}(y_i - \hat{y}_i)$$

다중회귀 회귀(Multiple Linear Regression)

Optimizer - 경사하강법의 프로세스(Gradient Descent)

Optimizer는 손실 L(θ)를 최소화하도록 파라미터 θ=(W,b)를 갱신하는 알고리즘 → 그 중에 하나인 경사하강법(Gradient Descent)

1) 초기화: 최적화해야 하는 매개변수의 초기 값을 선택하는 것부터 시작함

   → 무작위로 설정되거나 일부 경험적 방법을 기반으로 설정함

2) 기울기 계산: 각 매개변수에 대한 비용 함수의 기울기를 계산

3) 매개변수 업데이트: 기울기 반대 방향으로 매개변수를 조정

4) 비용 함수가 최소값에 수렴할 때까지 2단계와 3단계를 반복하며 업데이트

$$w_1 \leftarrow w_1 - \eta \frac{\partial MSE}{\partial w_1} \quad w_2 \leftarrow w_1 - \eta \frac{\partial MSE}{\partial w_2}$$

$$w_3 \leftarrow w_1 - \eta \frac{\partial MSE}{\partial w_3} \quad b \leftarrow b - \eta \frac{\partial MSE}{\partial b}$$

다중회귀 회귀(Multiple Linear Regression)

Optimizer - 경사하강법의 프로세스(Gradient Descent)

비용함수 : 예측 오류 또는 손실에 대한 정량적 측정을 제공하여 모델 성능을 수치적으로 평가

→ 최적의 매개변수를 찾는 것을 목표

각 매개변수 β_j 가 비용 함수 J에 어떻게 영향을 미치는지 이해하는 것이 중요함

$$L(w_0, w_1, \dots w_k) = \frac{1}{n}\sum_{i=1}^{n}(y_i - (w_0 + w_1 x_{i1} + \cdots + w_k x_{ik}))^2$$

Step 1) 비용함수 정의- loss 단계

$$\frac{\partial L}{\partial w_0} = -\frac{2}{n}\sum_{i=1}^{n}(y_i - (w_0 + w_1 x_{i1} + \cdots + w_k x_{ik}))$$

Step 2) 매개변수 편미분 – Backward

$$\frac{\partial L}{\partial w_j} = -\frac{2}{n}\sum_{i=1}^{n}(y_i - (w_0 + w_1 x_{i1} + \cdots + w_k x_{ik})) \cdot x_{ij}$$

$$\frac{\partial MSE}{\partial w_1} = -\frac{2}{n}(y_i - \hat{y}_i) \cdot x_{i1}$$

다중회귀 회귀(Multiple Linear Regression)

Optimizer - 경사하강법의 프로세스(Gradient Descent)

비용함수 : 예측 오류 또는 손실에 대한 정량적 측정을 제공하여 모델 성능을 수치적으로 평가

→ 최적의 매개변수를 찾는 것을 목표

각 매개변수 β_j 가 비용 함수 J에 어떻게 영향을 미치는지 이해하는 것이 중요함

$$w_j^{(new)} = w_j^{(old)} - \eta\left(\frac{\partial MSE}{\partial w_j}\right)$$

Step 3) 매개변수 업데이트 - Optimizer

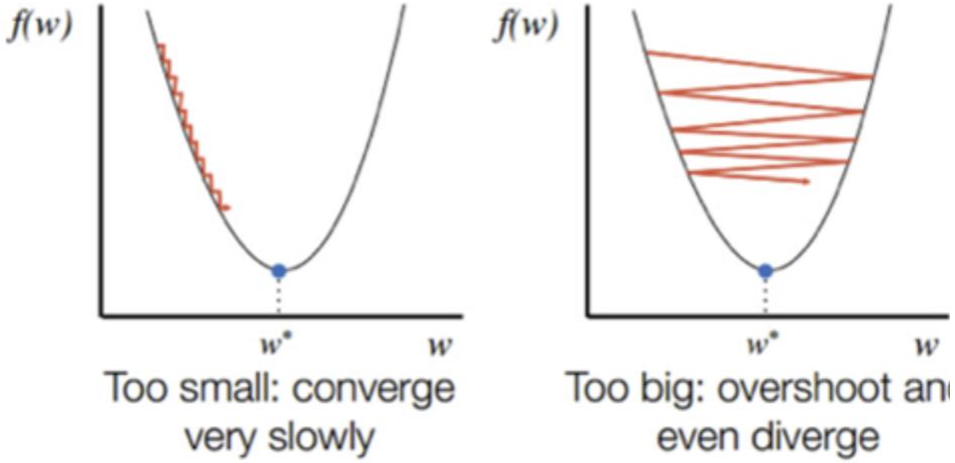
$$\frac{\partial MSE}{\partial w_1} = -\frac{2}{n}(y_i - \hat{y}_i) \cdot x_{i1}$$

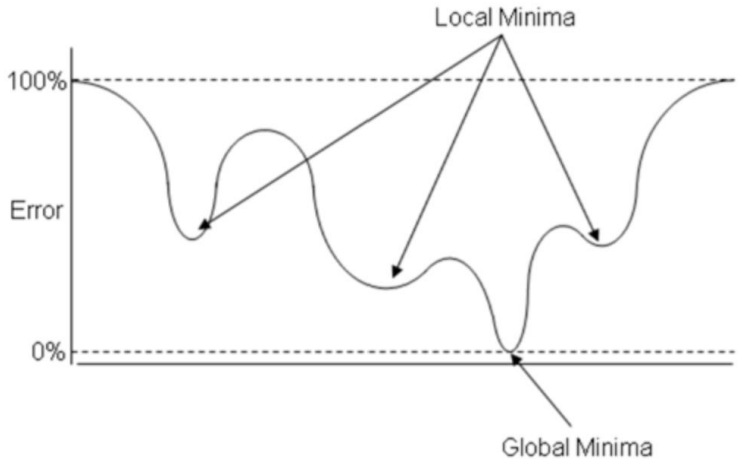
Step 4) Step 2~3반복

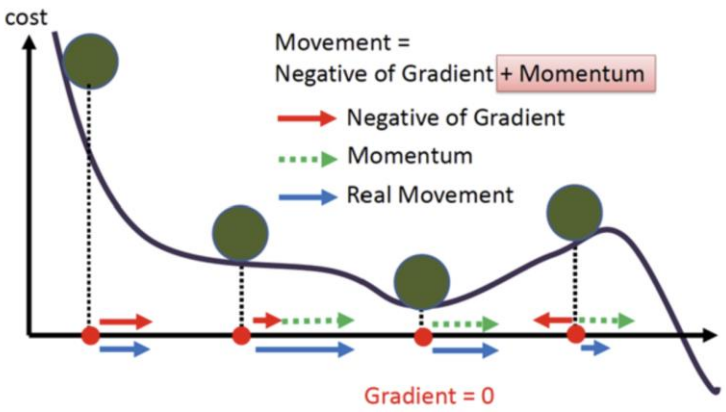
다중회귀 회귀(Multiple Linear Regression)

Optimizer - 경사하강법(Gradient descent)

- 미분을 이용하여 기울기가 최소인 지점을 찾음
- 학습률(Learning Rate)
  - 새로운 정보를 얼마나 반영할지를 조절
  - 학습률(Learning Rate)을 이용해 점차적으로 변화량을 조절함
- 모멘텀(Momentum) : 어느 정도 기존의 방향을 유지할 것인지 조정







다중회귀 회귀(Multiple Linear Regression)

Optimizer(Hyper parameter) - 학습률(learning rate)

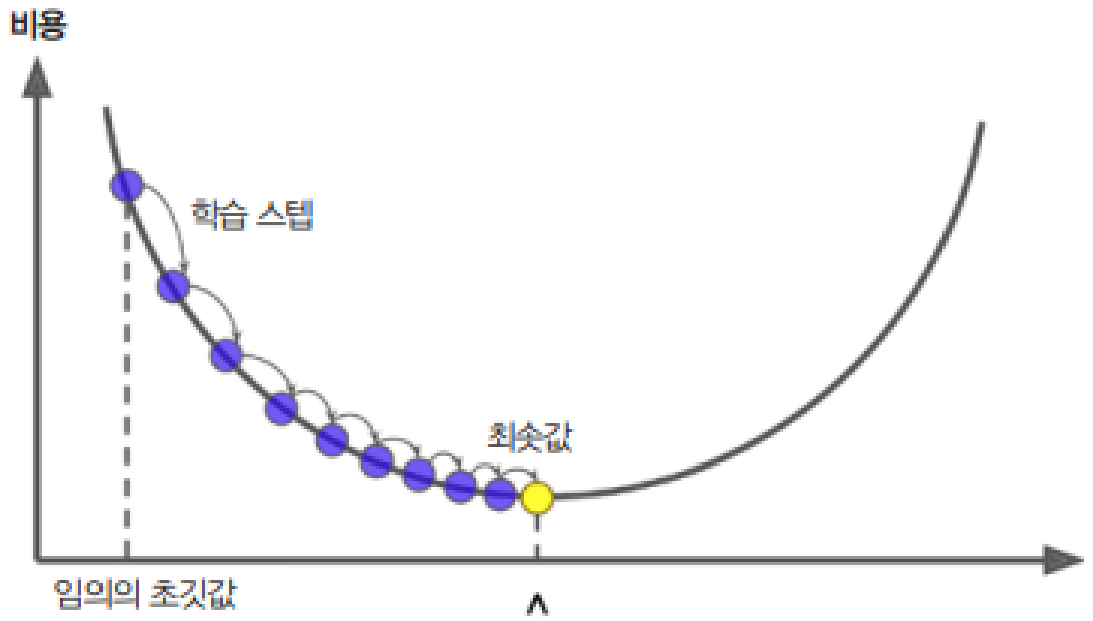
- W의 값을 변경할 때, 얼마나 크게 변경할지를 결정하는 값 (η)
- 접점의 기울기가 0인 지점을 찾는 것에 있어 어떤 크기의 폭으로 이동할지를 결정함

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

$$\hat{y}_i = w_0 + w_1 x_{i1} + w_2 x_{i2} + \cdots + w_p x_{ip}$$

$$\frac{\partial MSE}{\partial w_j} = \frac{2}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)x_{ij}$$

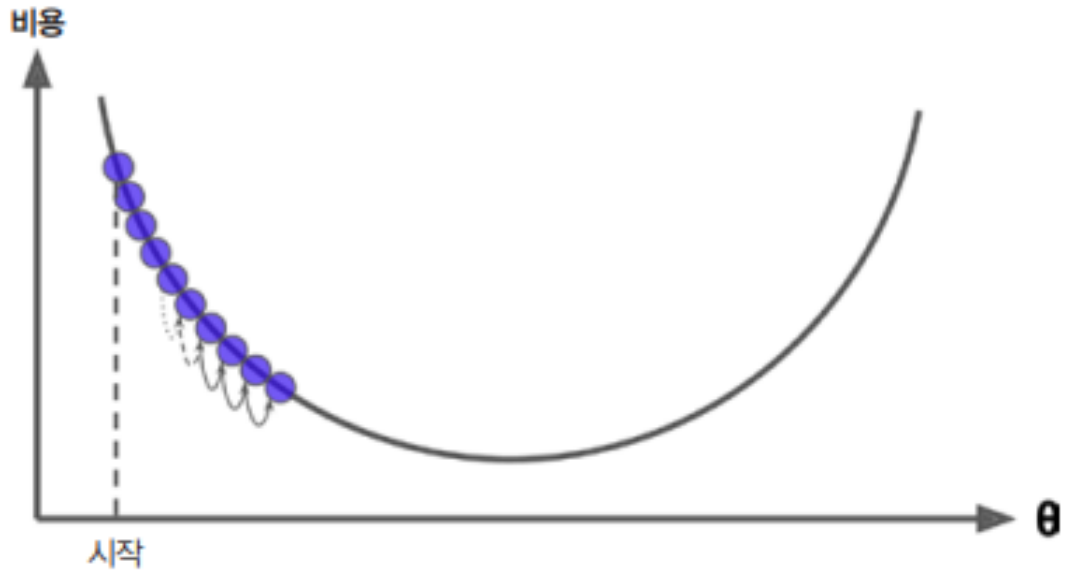
$$w_j \leftarrow w_j - \eta \frac{\partial MSE}{\partial w_j}$$

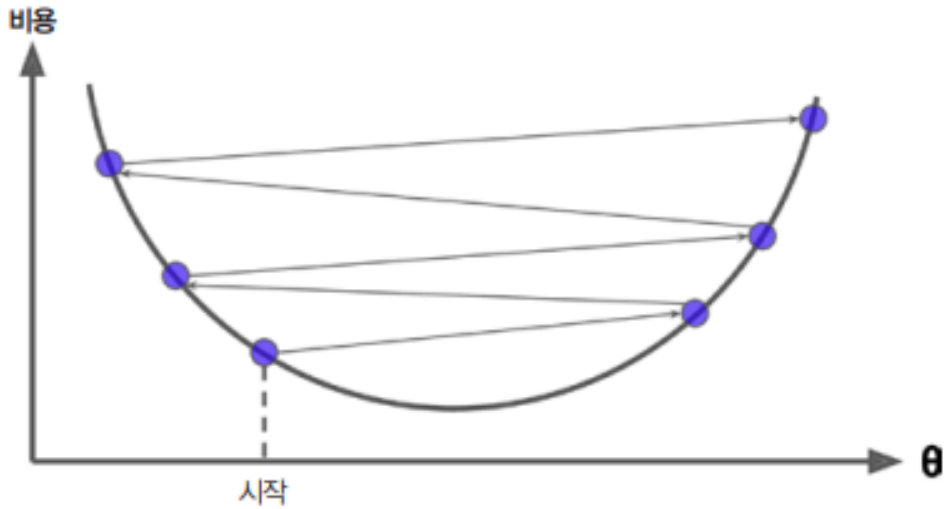


다중회귀 회귀(Multiple Linear Regression)

Optimizer(Hyper parameter) - 학습률(learning rate)

- 학습률이 너무 작으면 알고리즘이 수렴하기 위해 반복을 많이 진행하므로 시간이 오래걸림
- 학습률이 너무 크면 골짜기를 가로질러 반대편으로 건너뛰게 되어 접선이 0이 되는 지점을 찾지 못함
- 학습 횟수(Epoch)를 통해 최적의 Low Cost를 찾는 것이 중요함

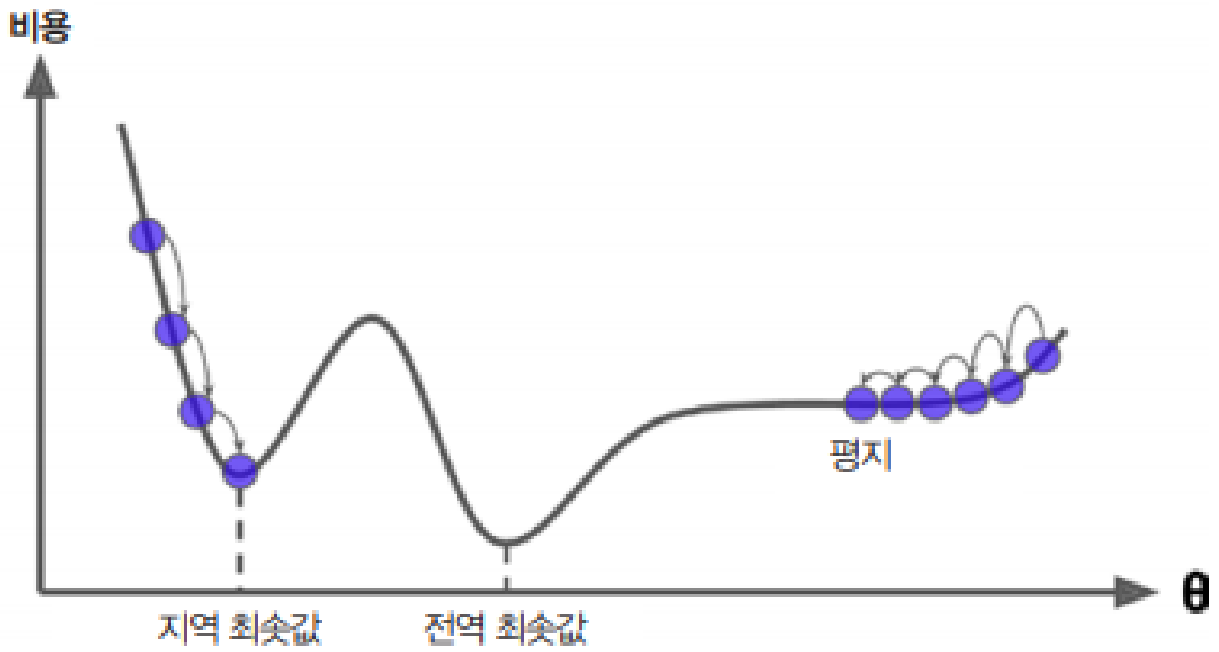




다중회귀 회귀(Multiple Linear Regression)

Optimizer(Hyper parameter) - 최적의 Cost 결정의 문제

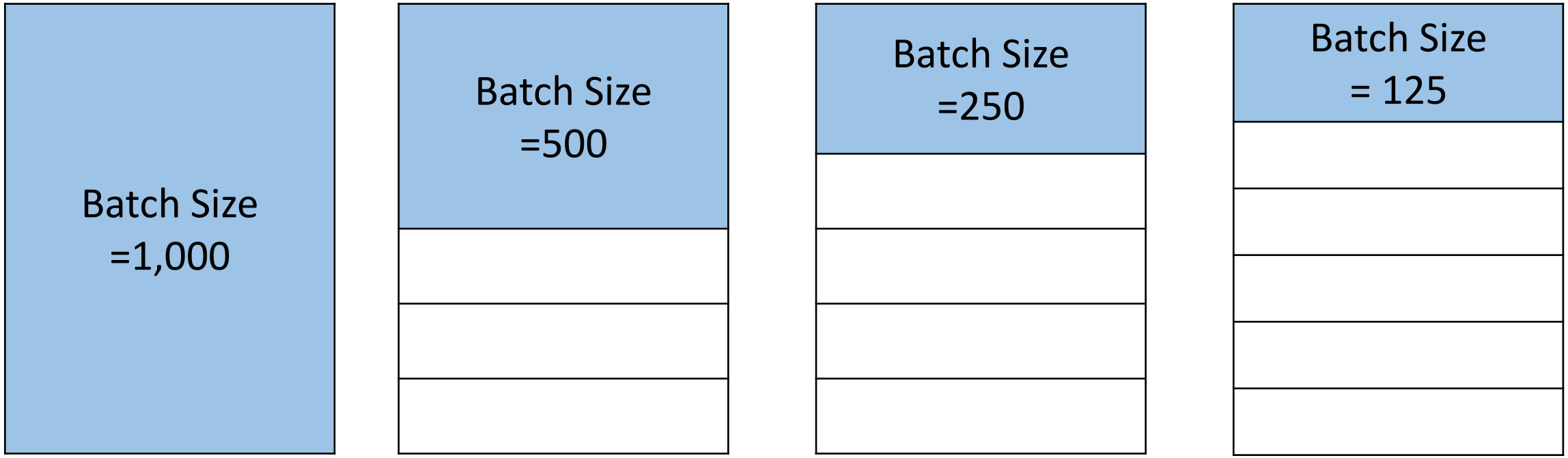
- 알고리즘이 왼쪽에서 시작하면, 전역 최소값보다 덜 좋은 지역 최소값에 수렴
- 오른쪽에 시작하면, 평탄한 지역을 지나기 위해 오랜 시간이 걸리고 일찍 멈추게 되어 전역 최솟값에 도달하지 못함



다중회귀 회귀(Multiple Linear Regression)

Optimizer(Hyper parameter) – Batch Size

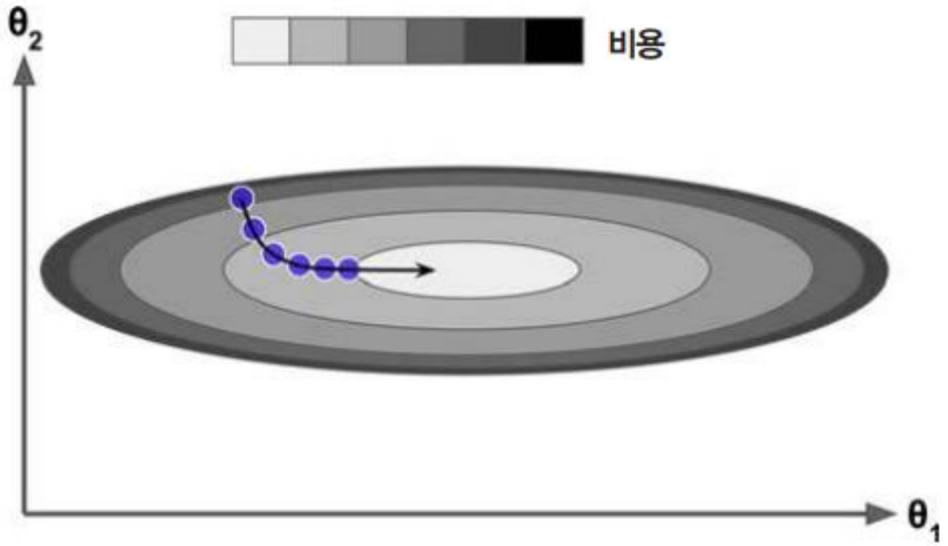
- 한 번의 기울기 계산에 사용할 샘플 수를 정해 기울기 노이즈·메모리 사용량·처리량
- 작은 배치 : 기울기의 노이즈를 증가 시키고, 계산이 많아 속도가 느릴 수 있음
- 큰 배치 : 처리량이 많아지고, 일반화가 떨어질 수 있음



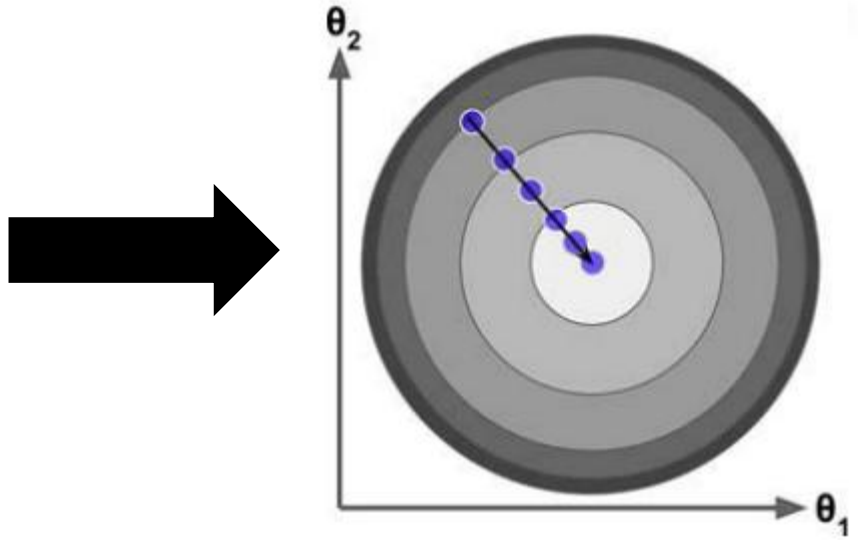
다중회귀 회귀(Multiple Linear Regression)

Optimizer - 변수 스케일의 크기의 문제

- 변수 1이 변수 2보다 스케일이 작은 훈련 데이터인 경우
- 표준화 진행을 통해 변수의 수준을 같게 변환해 학습의 시간을 줄여 줌



표준화를 적용하지 않은 경사 하강법

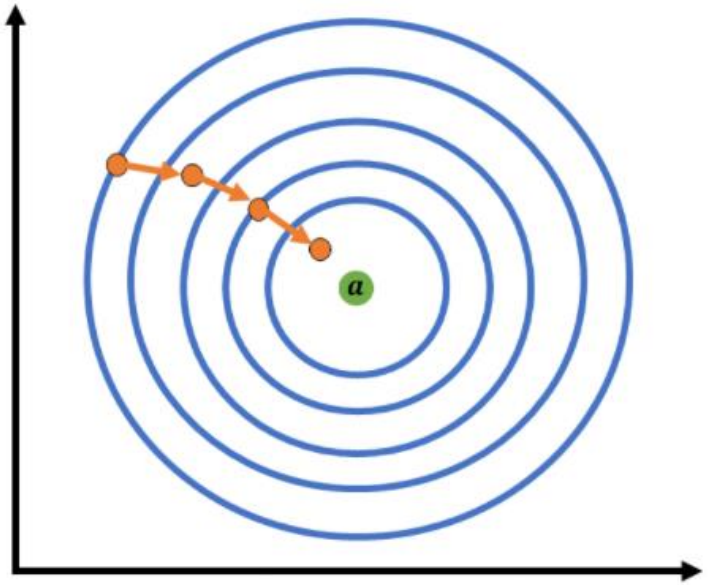


표준화를 적용한 경사 하강법

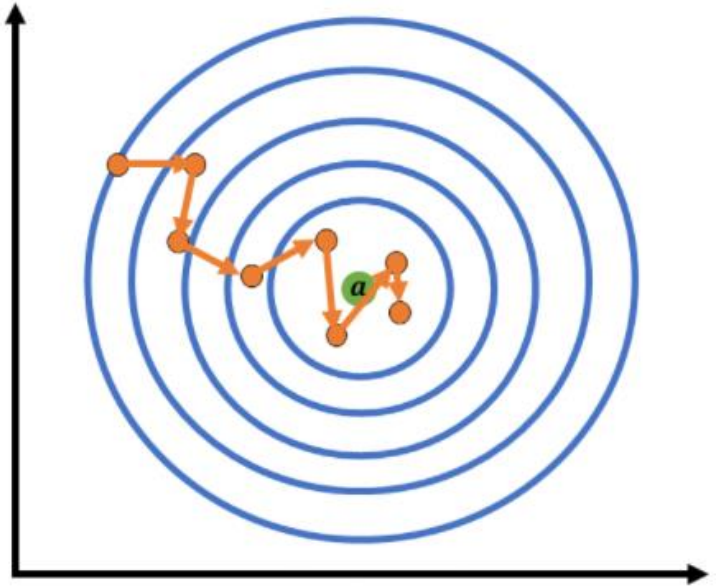
다중회귀 회귀(Multiple Linear Regression)

Optimizer - 변수 스케일의 크기의 문제

- 미니배치 경사 하강법(Batch Gradient Descent) : 전체 데이터셋을 통해 W값을 업데이트 하는 방법
- 확률적 경사 하강법(Stochastic Gradient Descent) : 하나의 데이터를 통해 W값을 구한 뒤 다음 데이터셋을 보며 W를 업데이트



미니배치 경사 하강법
(Batch Gradient Descent, BGD)

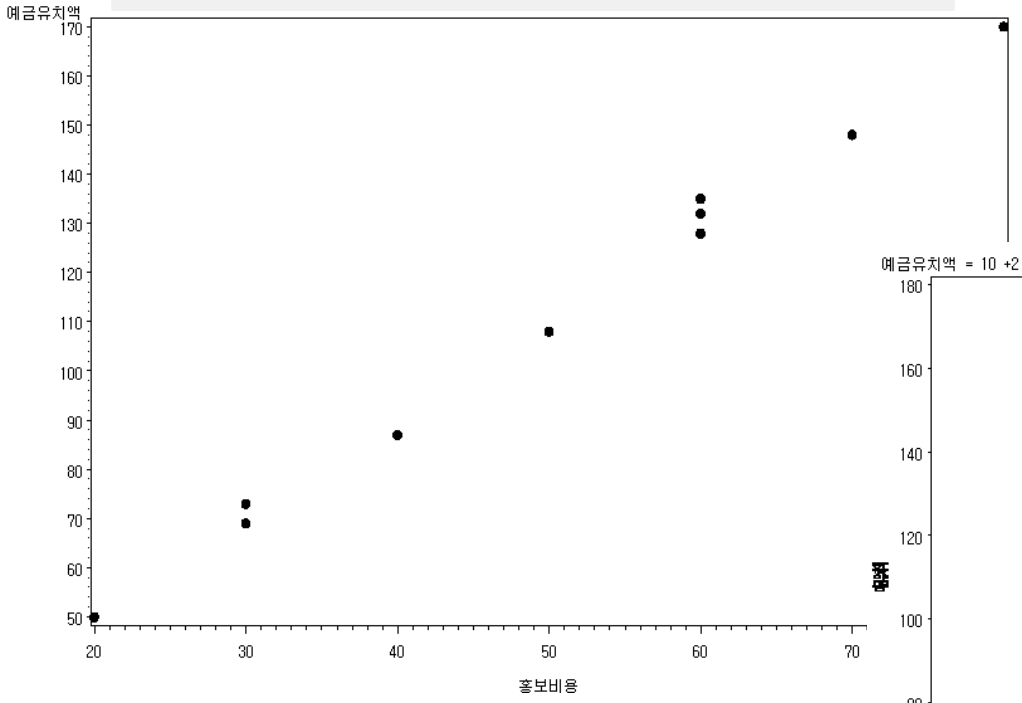


확률적 경사 하강법
(Stochastic Gradient Descent, SGD)

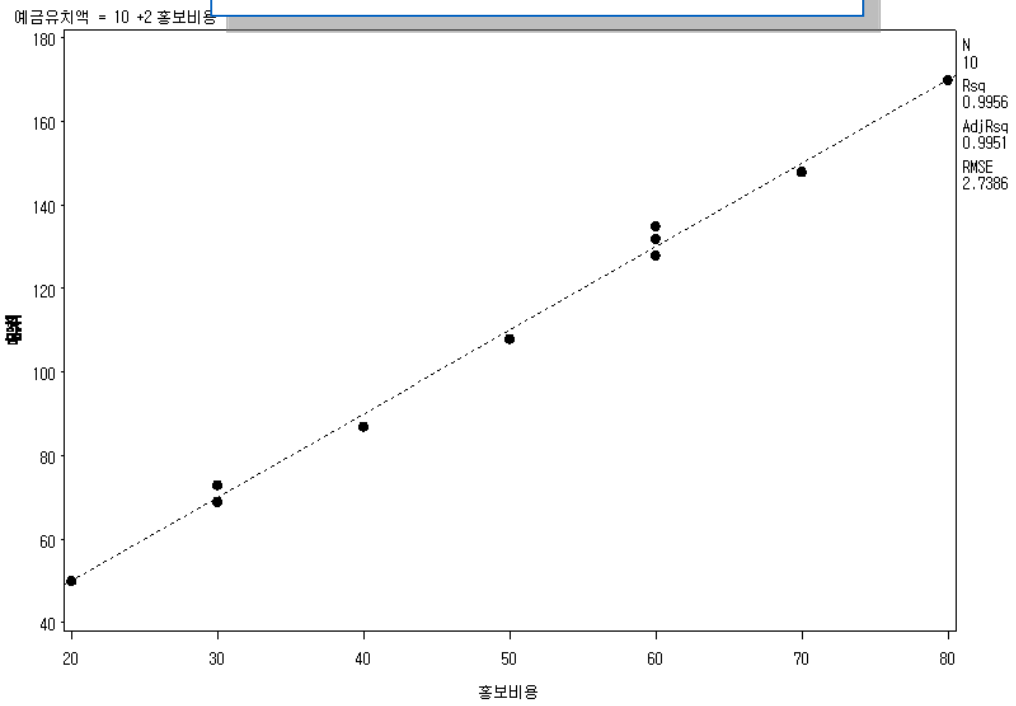
다중회귀 회귀(Multiple Linear Regression)

$$y = \beta_0 + \beta_1 x_1 + \varepsilon, \quad y = 예금유치액, x_1 = 홍보비용$$

홍보비용과 예금유치액에 대한 산점도



$$\hat{y} = 10 + 2 \times 홍보비용$$



다중회귀 회귀(Multiple Linear Regression)

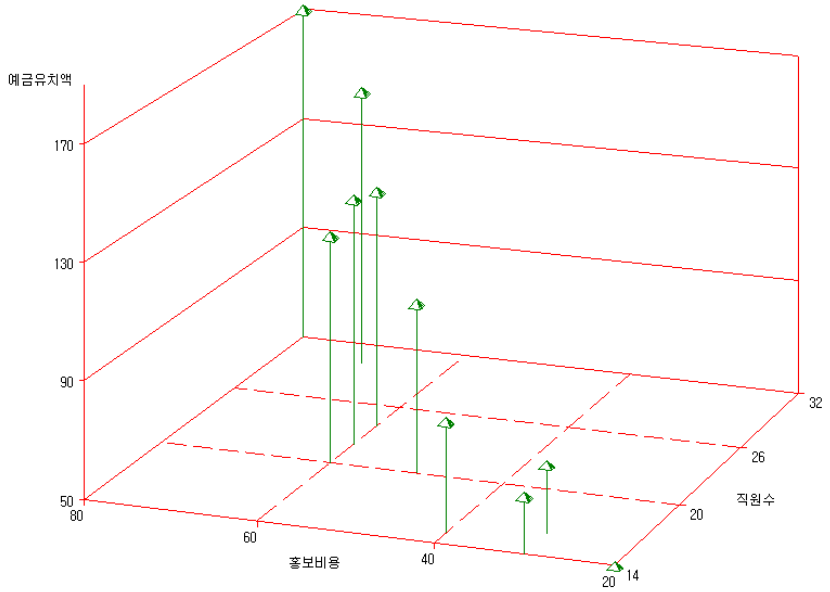
$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \varepsilon$$

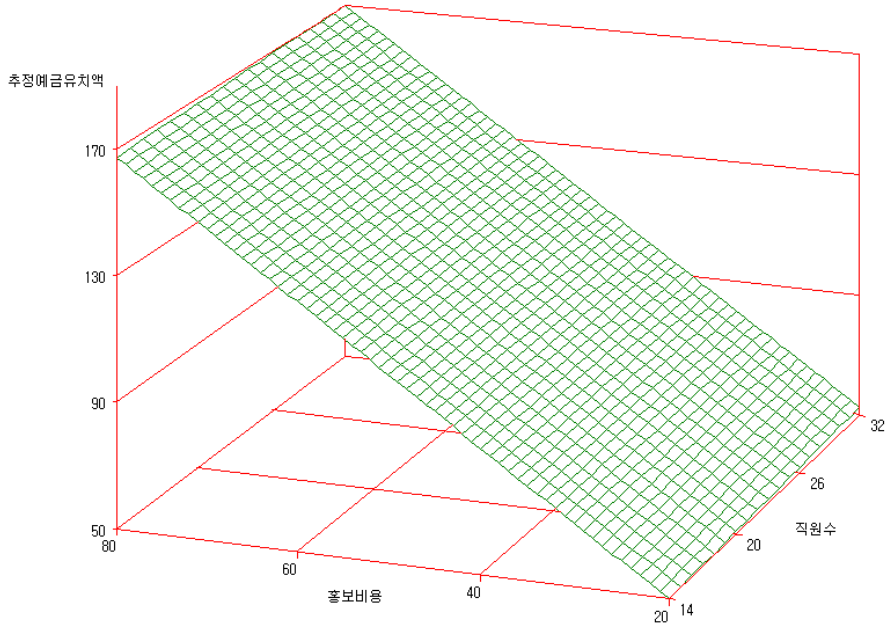
y = 예금유치액, x_1=홍보비용, x_2=직원수

홍보비용, 직원수 그리고 예금유치액에 대한 산점도

추정된 다중회귀모형

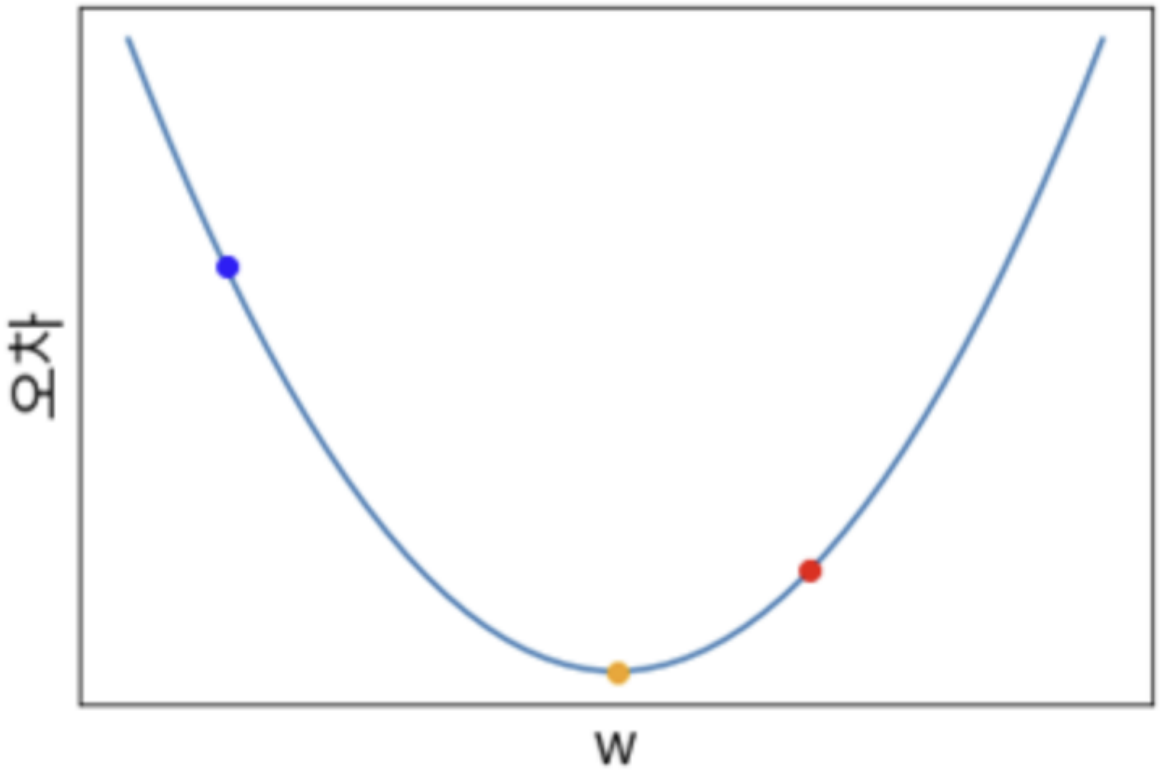
$$\hat{y} = 9.23 + 1.95 \times \text{홍보비용} + 0.15 \times \text{직원수}$$

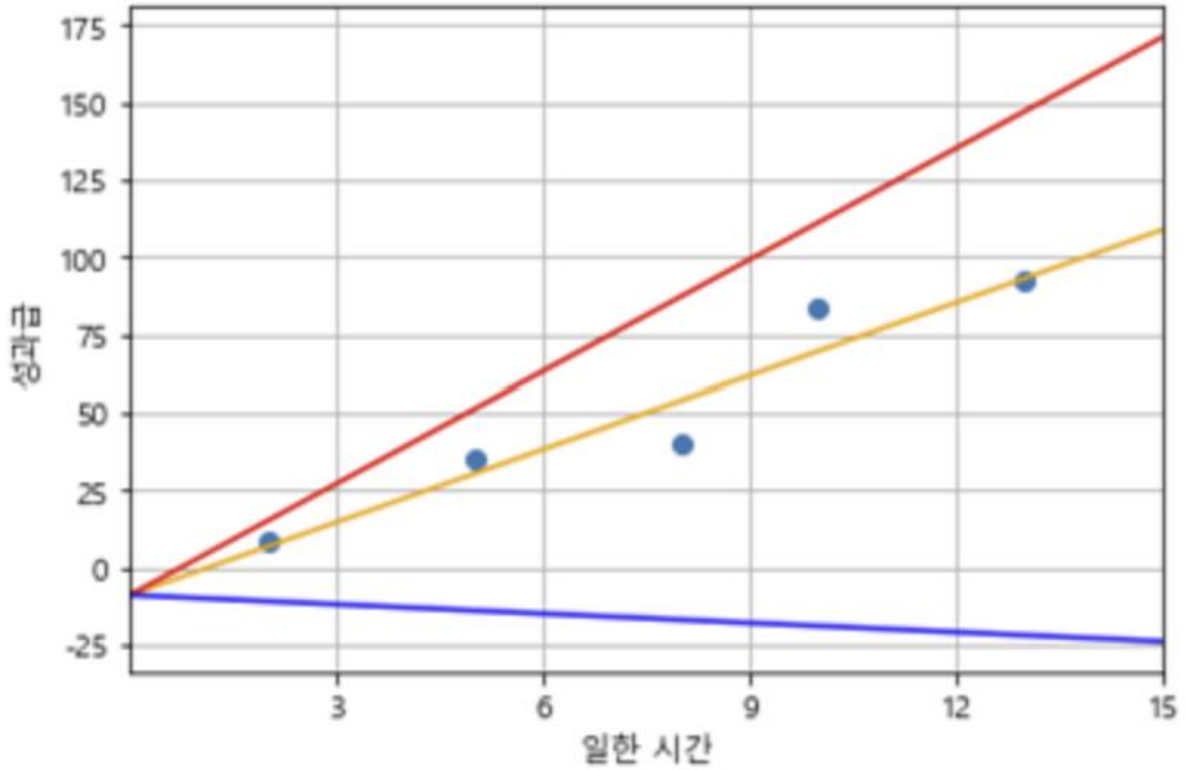




02 지도학습(Supervised Learning)

다중회귀 회귀(Multiple Linear Regression)





다중회귀 회귀(Multiple Linear Regression)-단순모델

```
library(MASS) #Boston 데이터셋
library(car)  #vif, durbinWatsonTest
library(skimr) #데이터 요약
library(rstatix) #shapiro_test
library(caret) #데이터 분할

set.seed(123)

data("Boston")
df <- Boston
skim(df)

initial_predictors <- c("lstat", "rm", "age", "dis", "rad", "tax", "ptratio", "indus", "nox",
"crim")

train_indices <- createDataPartition(df$medv, p = 0.8, list = FALSE)

train_df <- df[train_indices, ]
test_df  <- df[-train_indices, ]

result <- lm(train_df$medv ~ ., data = train_df[,initial_predictors])
summary(result)
```

다중회귀 회귀(Multiple Linear Regression)-GD기반 모델

```
source("Gradient Descent.R")   #R이 저장된 경로

train_indices <- createDataPartition(df$medv, p=0.8, list=FALSE)

df_train <- df[train_indices, ]
df_test <- df[-train_indices, ]

# 학습
gd_model <- train_bb_lm(df_train, target = "medv", predictors = preds, l2 = 1e-4)

# 결과요약
train_metrics.bb_lm(gd_model, df_train)

# 저장
saveRDS(gd_model, file = "bb_lm_boston.rds")

# 불러오기
loaded <- readRDS("bb_lm_boston.rds")
```

다중회귀 회귀(Multiple Linear Regression)-GD기반 모델

```
# 예측 & 평가
pred <- predict(loaded, newdata = df_test)
rmse <- sqrt(mean((df_test$medv - pred)^2))
r2   <- 1 - sum((df_test$medv - pred)^2) / sum((df_test$medv - mean(df_test$medv))^2)

print(round(pred, 3))
print(paste("RMSE:", round(rmse, 3)))
print(paste("R-Squared:", round(r2, 3)))

# 원본 스케일의 최종 계수 확인
coef(loaded)
```

03 다항 회귀분석(Polynomial Regression)

모형의 타당성&신뢰성 검정

- 1) 선형성 → 예측하고자 하는 종속변수 y와 독립변수 x간의 선형성을 이루어야 함
  → 비선형인 데이터에서 이러한 문제가 발생함
- 2) 등분산성 → 분산이 같다는 것을 의미하고 특정한 패턴 없이 고르게 분포하는 것을 의미함
  → 차원이 증가할수록 예측값의 범위가 넓어져, 잔차 분산이 커지거나 작아질 수 있음
- 3) 독립성 → 잔차 사이에는 상관관계가 없이 독립적이어야 함
  → 시간공간적으로 연결되어 있으면 위배 가능
- 4) 정규성 → 잔차가 정규성을 만족하는지 여부 (오차가 평균이 0인 정규분포)
  → 차원이 증가하면 일부 구간에서 모델이 데이터에 과도하게 맞춰지고, 잔차가 극단값을 포함할 가능성이 높아짐
- 5) 다중공선성 → 회귀 모델에서 두 개 이상의 독립변수가 서로 높은 상관관계가 있는 상황
  → 고차항과 원변수는 수학적으로 강한 상관성을 가짐

03 다중회귀 회귀(Multiple Linear Regression)

- 4) 정규성 → 종속변수가 정규성을 만족하는지 여부 (종속변수가 정규성을 만족하면 잔차가 평균이 0)

  샤피로 윌크 검정(Shaprio-Wilk test) : 표본수가 2,000미만인 데이터셋에 적합한 정규성 검정

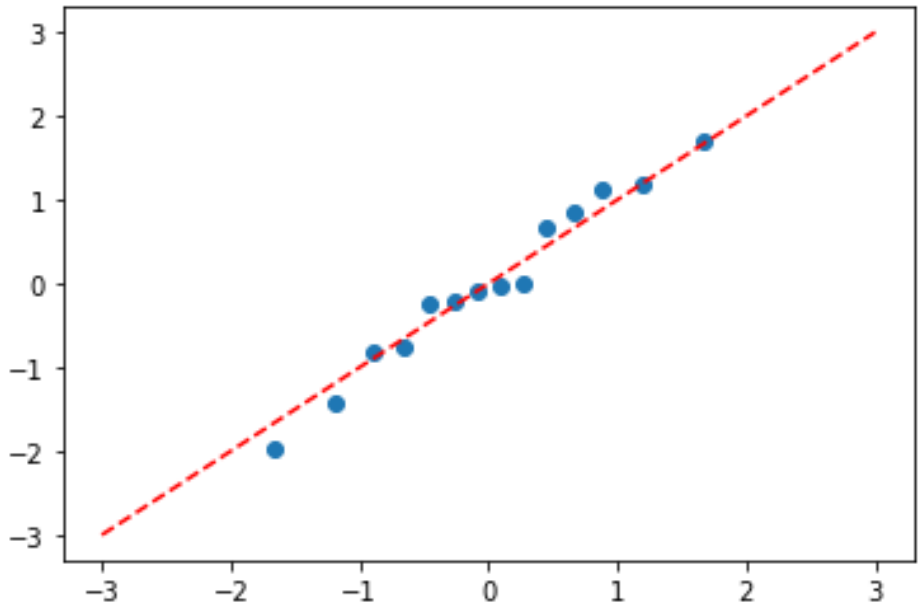
  스미르노프 검정(Kolmogorov-Smirnov test) : 표본수가 2,000초과인 데이터셋에 사용

  Qantile-Quantile plot(Graphic test) : 데이터셋이 정규분포를 따르는지 판단하는 시각적 분석방법 (Qqplot)
  →Z-Score : 잔차/표준편차

- **4) 정규성 가설검정**

정규성 검정에 대한 p-value : 0.05보다 크면 귀무가설 채택

Null hypothesis(H0; 귀무가설) : 데이터가 정규분포를 따름
Alternative hypothesis(H1; 대립가설) : 데이터가 정규분포를 따르지 않음



ShapiroResult(statistic=0.9721834659576416, pvalue=0.9044272899627686)

03 다중회귀 회귀(Multiple Linear Regression)

- **가정 검정**

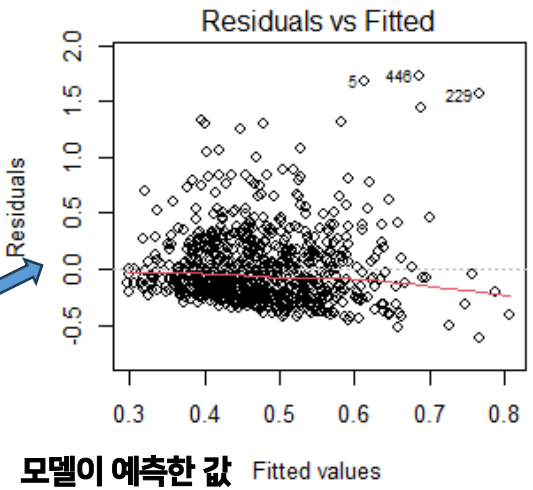
```
#잔차
opar <- par(no.readonly = TRUE)  # 그래프가 들어갈 window 생성
par(mfrow = c(2, 2))  # 그래프가 들어갈 공간 생성 2X2
plot(result)  # 그래프 결과 도출
par(opar)  # 그래픽 매개변수 복원

#정규성 : p>0.05보다 크면 귀무가설 채택(정규성이 있음)
shapiro_test(result$residuals)

#등분산성 : p>0.05보다 크면 귀무가설 채택(등분산성임)
ncvTest(result)
```

03 다중회귀 회귀(Multiple Linear Regression)

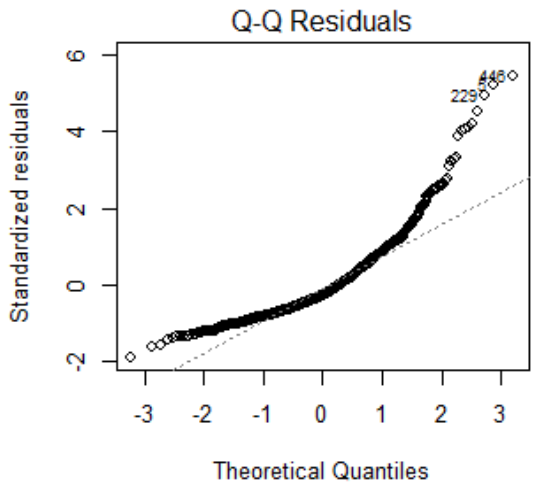
- 가정 검정

등분산성

잔차의 평균이 0 부근에서 평평한지



모델이 예측한 값

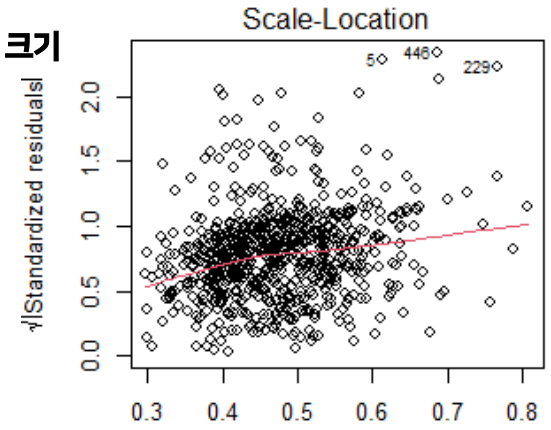


잔차의 정규성

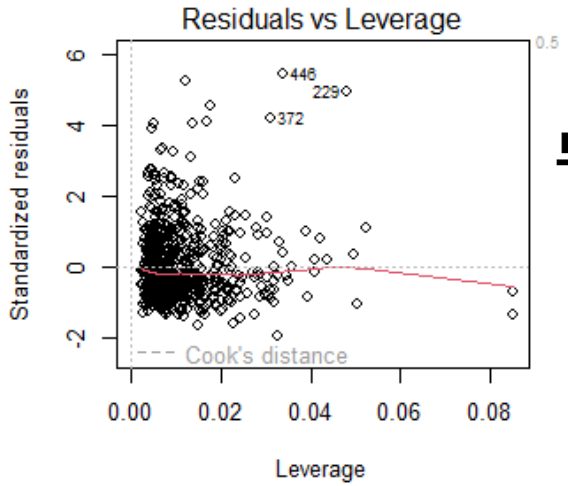
기대위치에 대한 표준화 잔차 값

루트 표준화잔차의 크기

등분산성



모델이 예측한 값



모델을 크게 흔드는 이상치를 판단

모형의 타당성&신뢰성 검정

- 5) 다중공선성 → 회귀 모델에서 두 개 이상의 독립변수가 서로 높은 상관관계가 있는 상황
  → 고차항과 원변수는 수학적으로 강한 상관성을 가짐

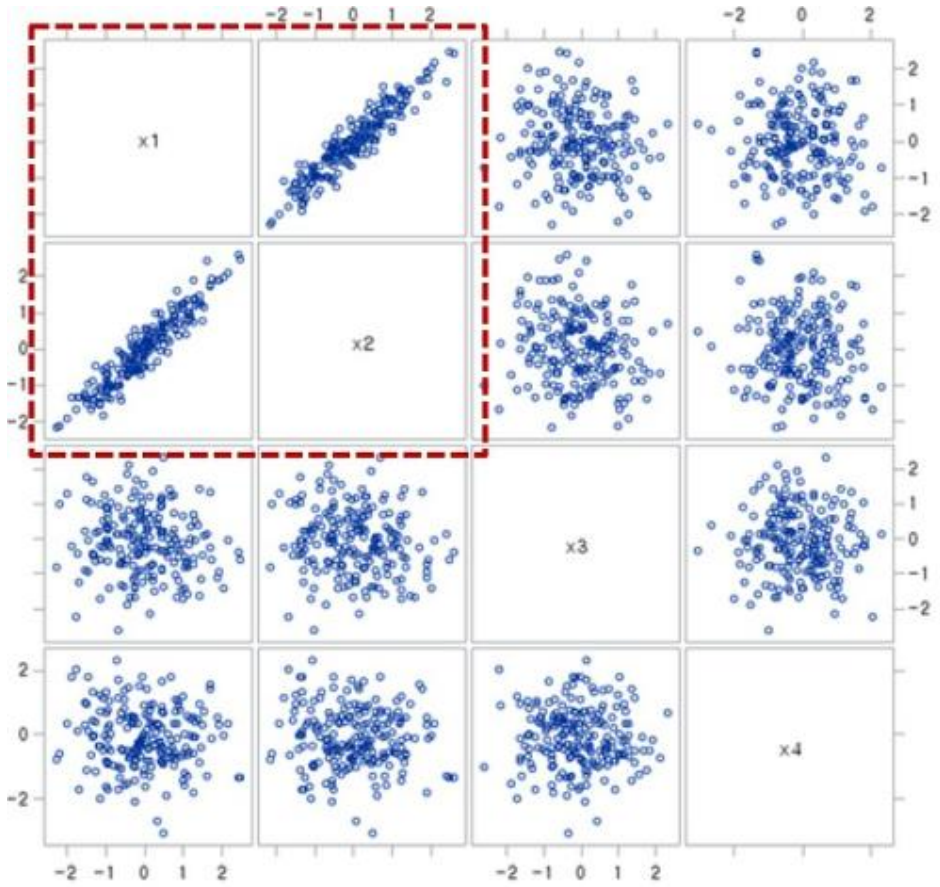
- 독립변수들간의 상관관계가 높을 경우, 회귀계수의 값이 매우커지는 문제

- 회귀계수의 변동성이 커져서 통계량과 모수가 서로 반대 부호를 가질 수 있음

- 분산팽창계수 : Variance Inflation Factor
  - VIF = $\frac{1}{1-R^2}$

03 다중회귀 회귀(Multiple Linear Regression)

독립 변수의 일부가 다른 독립 변수의 조합으로 표현될 수 있는 경우
VIF 지수가 10이상일 때 연관성이 있다고 판단 → R-Squred값이 유지되거나 높아지는 모델을 찾아야 함



03 다중회귀 회귀(Multiple Linear Regression)

- **가정 검정**

```
#다중공선성(변수의 중복성) : 10이상 GVIF^(1/(2*DF))
vif_values <- vif(result)

print(vif_values)

selected_predictors <- names(vif_values[vif_values < 7])
print(selected_predictors)

#이상치검정 : cook's D 값이 높으면 문제(y값이 3~4보다 크면)
influencePlot(result, id.method="identify")

#이상치 제거
df=df[-c(121),] #하나의 값 제거
df=df[-c(121, 130, 145, 160),] #여러 개의 값 제거
```

03 다중회귀 회귀(Multiple Linear Regression)

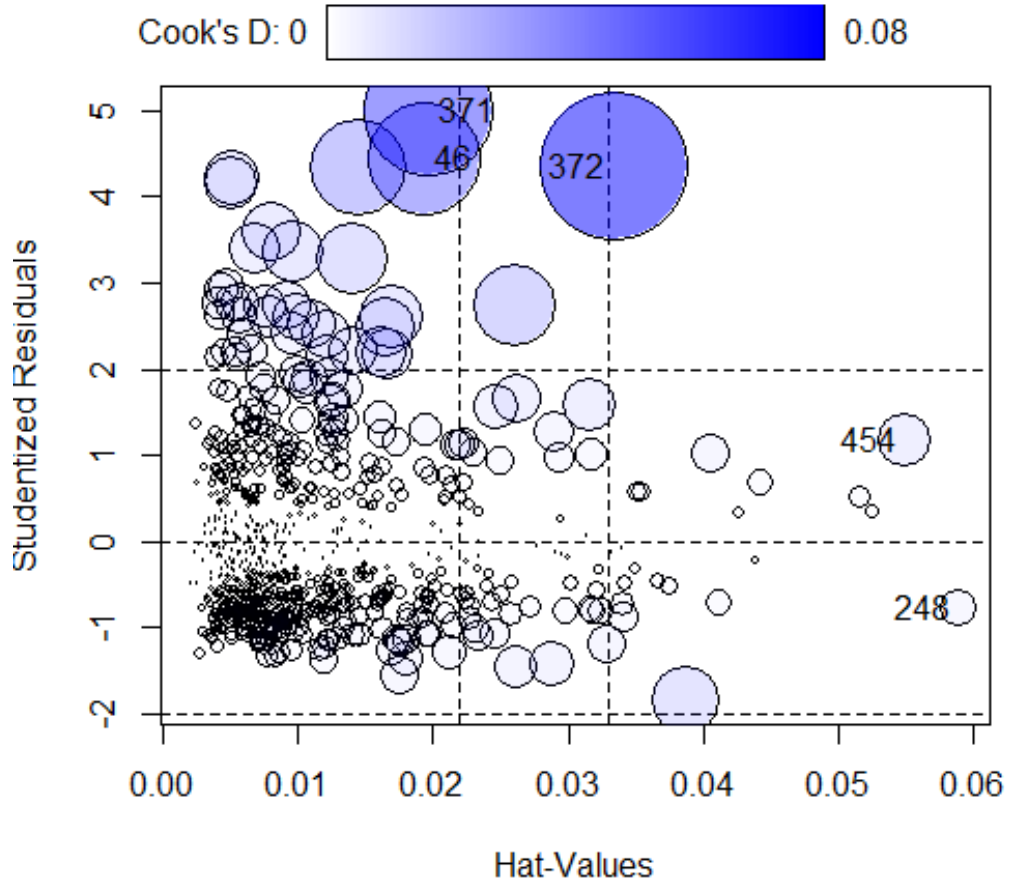
- 가정 검정

X축 : 독립 변수의 값이 모든 값의 평균에서 얼마나 떨어져 있는가?

Y축 : 관찰값과 예측된 값 상의 떨어져 있는 표준편차

데이터의 크기 : 회귀 모델에 영향을 크게 주는 것(하나의 데이터가 너무 크면 문제)

표준화 잔차(얼마나 틀렸나)와 레버리지(얼마나 특이한 위치인가)



03 다중회귀 회귀(Multiple Linear Regression)

- **AIC(Akaike information criterion)를 활용한 변수선택 →** $AIC = -2 \times log - likelihood + 2 \times k$

  k = 추정한 파라미터 개수

- **모형 적합도(log-likelihood) 와 모형 복잡도(변수 개수) 를 동시에 고려하는 지표 → 절대적인 임계값이 없음**

- **AIC는 로그우도(log-likelihood)와 모형 복잡도를 합친 지표 → 값이 절대적으로 몇 이상이면 나쁘다/좋다라고 정할 수 없음**

- **단순히 AIC값이 작을수록 좋은 모델이라는 상대적 기준만 존재**

```
df <- read.csv("diabetes.csv", header = TRUE, na = ".")

result <- lm(df$Diabetes ~ ., data = df)
```

03 다중회귀 회귀(Multiple Linear Regression)

- **변수선택**

```
AIC(Akaike information criterion) AIC값이 줄어들면 그 변수는 의미가 있음

library(MASS)

#후진소거법
result_bk = lm(df$Diabetes ~., data=df)
Df_fit_bk =stepAIC(result_bk, direction= "backward", trace = T) #trace 하나씩 빼는 과정을 보여줌

#단계적선택법
Df_fit_st = stepAIC(result_bk, direction = "both", trace = T)

#전진선택법
result_fw = lm(df$Diabetes ~ 1, data=df)
Df_fit_fk =stepAIC(result_fw, direction= “forward”,scope = (backward 변수 카피) ,trace = T)
```

03 다중회귀 회귀(Multiple Linear Regression)

- p-값 모두 회귀 모델이 선형성, 독립성, 등분산성, 정규성 및 다중 공선성이 없다는 가정을 충족
- 전체 모형의 유의성(모델에 대한 p-value)
  → 전체 모형이 유의한지는 p-value가 0.05 미만인지를 통해 알 수 있음(0.05 미만이라면 모델이 유의함)
- 회귀 모형의 결정계수(R-Squred)
  → 1에 가까울수록 유의함
- 설명력 RMSE/MSE
  → 수치가 낮을수록 모델이 좋은 설명력을 가진다고 판단함
- 회귀식에서 유의한 변수 (끝에 별이 달린 것)과 회귀 계수(Estimate)의 부호 및 크기