1 선형 문제와 비선형 문제

1 선형 문제와 비선형 문제

1 두 변수의 상관 관계

- 다수의 데이터에서 예측과 분석을 수행할 때
  - 수학적 모델과 통계 모델을 검토
  - “임의의 2개의 데이터를 비교“
  - 수집한 데이터가 어떻게 변하는지 확인한 후 앞으로 수집할 데이터의 변화를 비교, 예측

1 두 변수의 상관 관계

변수

- 데이터를 다룰 때 데이터를 구성하는 항목

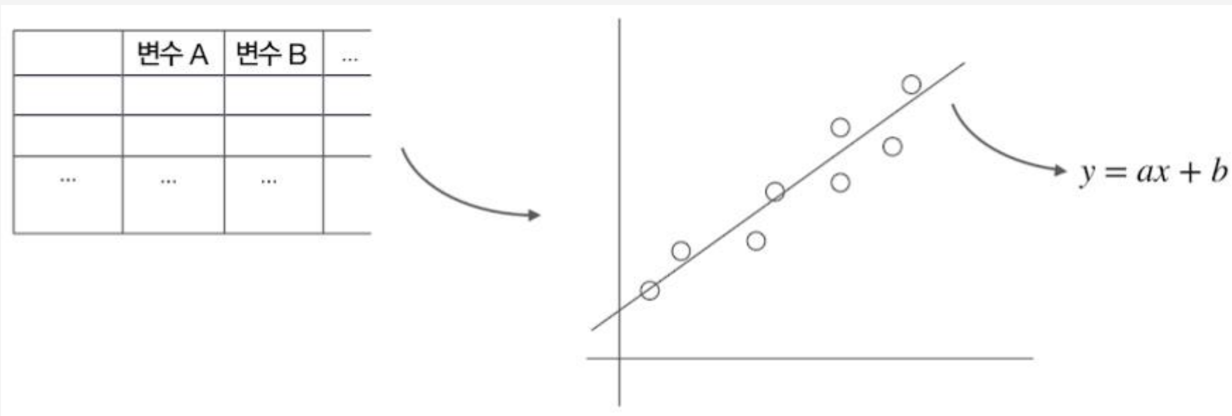
특징량(Feature)

- 데이터에서 변화를 나타내는 한 가지 이상의 변수 쌍, 또는 변수 쌍을 사용한 계산식

1 선형 문제와 비선형 문제

1 두 변수의 상관 관계

[통계표에서 변수 2개(특징량)를 선택해 산포도를 그리고 모델을 도출]



※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

2 선형 계획법과 비선형 계획법

선형 계획법

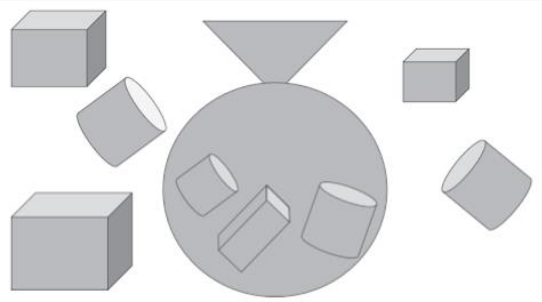
- 선형 함수
  - 변수값 쌍을 그래프에 점으로 표현하였을 때 일직선처럼 보일 경우
- 선형 계획 문제
  - 점의 분포를 선형 함수의 제약과 조건을 이용해 구할 수 있는 문제(예: 배낭 문제)
- 정수 계획 문제
  - 선형 계획 문제 중 정수에 한정해서 문제를 해결할 때

2 선형 계획법과 비선형 계획법

선형 계획법

▶ 선형 함수를 최적화하여 문제를 해결하는 것

[배낭 문제 예]



조건)
여러 개의 짐의 크기와 값이 변수값으로 분포
배낭 안 짐의 값을 최대화 하기 위해 짐을 선택
배낭 용량은 늘릴 수 없음

※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

2 선형 계획법과 비선형 계획법

선형 계획법

- 선형 계획법 실제 사례
  - 금융회사에서 포트폴리오 구성 시
    - 투자 대상과 예상 수익율을 변수로 구성
    - 투자 금액 및 투자 시 필요한 제약 조건 설정
    - 연간 예상 수익율을 최대화 하는 것을 목표로 목적 함수를 구성하여 분석 수행

2 선형 계획법과 비선형 계획법

선형 계획법

- 선형 계획법 실제 사례
  - 식당의 종업원 수를 시간별로 조정하는 문제 시
    - 시간대 별 종업원 수를 변수로 구성
    - 시간대에 따른 종업원 수를 제약 조건으로 설정
    - 시간대 별 최적 종업원 수를 목표로 목적 함수로 구성하여 분석 수행

2 선형 계획법과 비선형 계획법

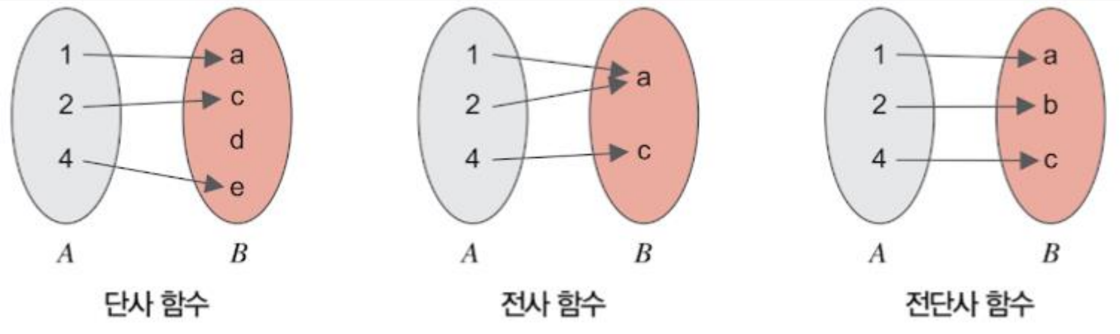
선형 계획법

- 사상
  - 변수값 쌍을 점 형태로 표현하여
    함수 형태로 변환한 것
  - 단사 함수(일대 일 함수): 점의 집합을
    각각 A와 B라고 할 경우 A의 원소와 B의 원소가
    1:1로 대응하는 경우
  - 전사 함수: A가 B의 모든 원소에 대응하는 함수
  - 전단사 함수: A와 B의 원소가 중복 없이
    모두 1:1로 대응하는 함수

2 선형 계획법과 비선형 계획법

선형 계획법

▶ 사상



※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

2 선형 계획법과 비선형 계획법

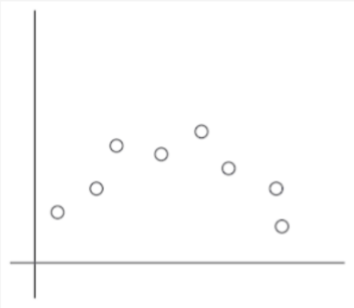
비선형 계획법

- 변수값 쌍으로 구성하는 점의 분포가 선형으로 표현되지 못할 경우
- 선형 분리 불가능한 분포도

2 선형 계획법과 비선형 계획법

비선형 계획법

- 비선형 분포 : 사상 개념으로 대응할 수 없을 경우



- 비선형 문제: 비선형 분포를 다루는 문제

※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

2 선형 계획법과 비선형 계획법

비선형 계획법

- 비선형 계획법 실제 적용 예
  - 신제품 출시에 따른 가격 책정 문제
    - 신제품에 대한 가격 Price를 결정하기 위해 수요량을 예측해 볼 경우
    - 수요량 Demands는 Price에 따라 증가 또는 감소할 수 있는 종속 변수
    - 따라서 최종 목적 함수인 가격 결정은 신제품의 판매 이익과 원가에 의해 결정되며 판매 이익 및 원가는 수요에 따라 변동될 수 있음

2 선형 계획법과 비선형 계획법

비선형 계획법

- 비선형 계획법 실제 적용 예
  - 신제품 출시에 따른 가격 책정 문제
    - 결국 최종 목적 함수가 2차 함수가 되어 선형 계획법으로 해결이 어려움
    - 2차 함수의 최대 또는 최소값 산출을 위해 각 변수에 따른 편미분을 이용하여 조건을 만족하는 결과 값 산출

2 선형 계획법과 비선형 계획법

비선형 계획법

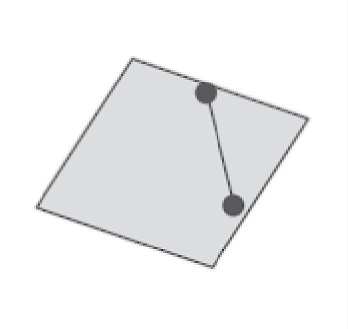
- 볼록 계획 문제
  - 볼록(凸)함수 & 오목(凹)함수로 점의 분포를 나타낼 수 있는 경우에 사용
  - 볼록 최적화 방법을 이용해 해결

2 선형 계획법과 비선형 계획법

비선형 계획법

▶ 볼록 계획 문제

[볼록의 예; 도형 평면 안에 있는 점을 연결할 때
같은 도형 평면 상에 존재]



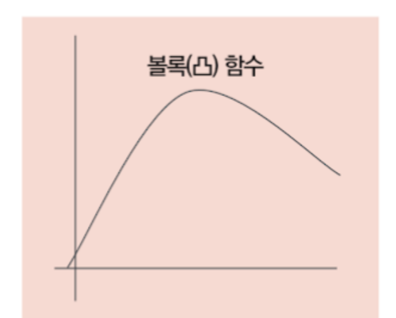
※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

2 선형 계획법과 비선형 계획법

비선형 계획법

▶ 볼록 계획 문제

[볼록함수; 돌출부가 하나인 함수]



※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

2 선형 계획법과 비선형 계획법

비선형 계획법

- 볼록 함수
  - 볼록 집합: 집합에 존재하는 그 어떤 점을 잡아도 그 점들 사이에 있는 모든 점들 역시 그 집합에 포함되는 집합
  - 최적 값이 한 개 밖에 존재하지 않음
  - 따라서 위로 볼록한 경우 최대값이 최적값이 되며, 아래로 볼록한 경우 최소값이 최적값이 됨

$$f=(\alpha x+(1-\alpha)y)\leq \alpha f(x)+(1-\alpha)f(y)$$

※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

2 선형 계획법과 비선형 계획법

비선형 계획법

- 볼록 최적화
  - 목적 함수가 볼록 함수인 경우 사용 가능한 방법

2 선형 계획법과 비선형 계획법

비선형 계획법

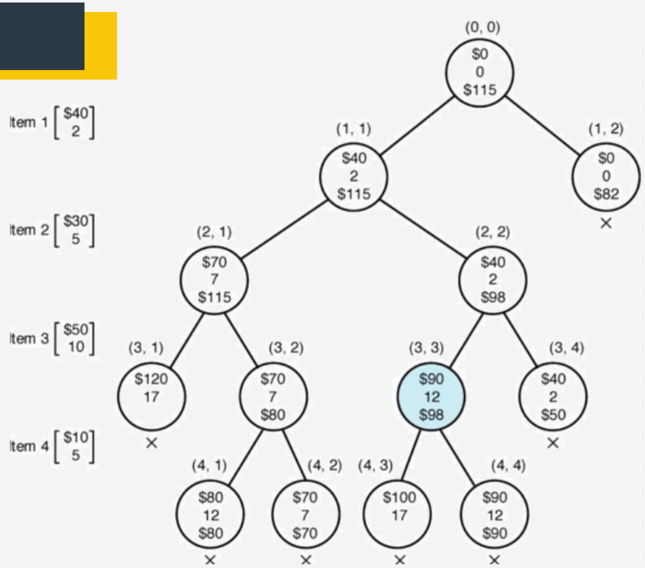
- 분기 한정법
  - 선형 계획 문제와 볼록 계획 문제를 조합
  - 모든 후보해를 체계적으로 늘어 놓아 최적화할 수치의 상한과 하한을 추정하고 불필요한 값들을 제거하는 과정 수행
  - 이산 최적화나 조합 최적화를 풀 때 사용

2 선형 계획법과 비선형 계획법

비선형 계획법

▶ 분기 한정법

[배낭 문제 예]



※ 출처 : http://timewizhan.tistory.com/entry/분기한정법-branchandbound [Small talk with computer]

2 회귀 분석

1 회귀 분석

- 회귀 : 함수로 변화를 나타내고 피팅하는 것
- 회귀 분석의 종류
  : 단순 회귀, 다항식 회귀, 로지스틱 회귀, 다중 회귀 등
- 피팅에 사용하는 방법 : 최소 제곱법

1 회귀 분석

- 주어진 데이터로 어떤 함수를 만들어 낸 후 함수를 피팅하는 작업
- 피팅
  : 함수에서 발생하는 차이(잔차의 크기)가 최소화되도록 함수를 조정하는 것
- 잔차
  : 회귀 직선을 통해 얻은 값(시뮬레이션값)과 실제값과의 차이

1 회귀 분석

- 일반선형모델
  : “잔차의 분포가 정규 분포를 따르는 것”을 전제로 만들어진 함수

- 일반화 선형 모형
  : 잔차의 분포가 임의의 분포로 만들어진 경우

1 회귀 분석

- 사용 예 : 선형화 문제 해결 시 이용
- 어느 정도 신뢰할 수 있는가를 검정해서 통계 예측에 활용
- 데이터의 잔차 정도를 통해 신뢰구간 표현하여 예측 결과의 정확성 표현

1 회귀 분석

단순 회귀

- 요소들 사이의 비례 관계를 활용
- 독립 변수가 한 개인 형태
- 예 : 신장과 체중 사이의 관계, 임대주택 방 크기와 임대료 사이의 관계 등
- 직선 y = ax + b에서 기울기 a와 절편 b를 알면 임의의 x에 대해서 y를 산출
  - y : 종속 변수 , 목적 변수
  - x : 독립 변수, 설명 변수

1 회귀 분석

회귀 직선의 기울기와 절편 구하기

- $y=ax+b+e$ 에서 a, b를 구하기
- 단순 회귀 모델 식을 기반으로 한 잔차 제곱 E (목적 함수)를 구함 (식 1)

$$E=\sum_{i=1}^{n}(y_i-ax_i-b)^2$$

※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

1 회귀 분석

회귀 직선의 기울기와 절편 구하기

- 기본 식에서 a와 b를 편미분으로 연립 방정식 설립 후 잔차 제곱이 최소화 되는 값 산출

* 편미분은 a와 b가 독립 변수이므로 미분의 번거로움을 피하기 위해 사용

[기본식을 a에 대한 편미분 방정식]

$$\frac{\partial E}{\partial a} = \sum_{i=1}^{n}(2ax_i^2 + 2x_i(b - y_i)) = 0$$

[기본식을 b에 대한 편미분 방정식]

$$\frac{\partial E}{\partial b} = \sum_{i=1}^{n}(2b + 2(ax_i - y_i)) = 0$$

※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

1 회귀 분석

회귀 직선의 기울기와 절편 구하기

- 연립 편미분 방정식을 풀면 a와 b 각각을 구하는 식으로 변경 가능

$$a = \frac{n\sum_{k=1}^{n} x_k y_k - \sum_{k=1}^{n} x_k \sum_{k=1}^{n} y_k}{n\sum_{k=1}^{n} x_k^2 - \left(\sum_{k=1}^{n} x_k\right)^2}$$

$$b = \frac{\sum_{k=1}^{n} x_k^2 \sum_{k=1}^{n} y_k - \sum_{k=1}^{n} x_k y_k \sum_{k=1}^{n} x_k}{n\sum_{k=1}^{n} x_k^2 - \left(\sum_{k=1}^{n} x_k\right)^2}$$

[회귀 직선의 기울기와 절편 구하기]

※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

2 다중 회귀

- 독립 변수가 증가할 경우 독립 변수 각각을 단순 회귀와 같은 방식으로 분석

단일회귀 분석	다중회귀 분석
$y = \alpha + \beta x + \varepsilon$	$y = \alpha + \beta x_1 + \gamma x_2 + \varepsilon$

독립 변수가 x_1, x_2로 증가

※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

2 다중 회귀

- 다중 회귀 분석
  : 독립 변수가 여러 개인 경우의 회귀 분석
- 독립 변수가 여러 개가 되면 2차원 평면 상에
  그래프로 표현할 수 없음
- 2차원 평면 상으로 매핑을 위해 주성분 분석
  (PCA : Principal Component Analysis)을 이용
- 독립 변수가 데이터 개수 보다 많은 경우에는
  주성분 분석의 차원 감소를 위한 주성분 회귀와 부분
  최소 제곱 회귀를 활용

2 다중 회귀

- 독립 변수가 늘 때의 단점
  - 회귀 분석의 결과를 신뢰할 수 없는 경우 발생
  - 답을 구할 수 없는 경우 발생

2 다중 회귀

- 다중 공선성(Multicollinearity)
  - 독립 변수란 각각이 선형 독립이어야 한다는 전제
  - 독립 변수가 증가하면 독립 변수들 사이에 상관 관계가 존재할 가능성이 높아짐

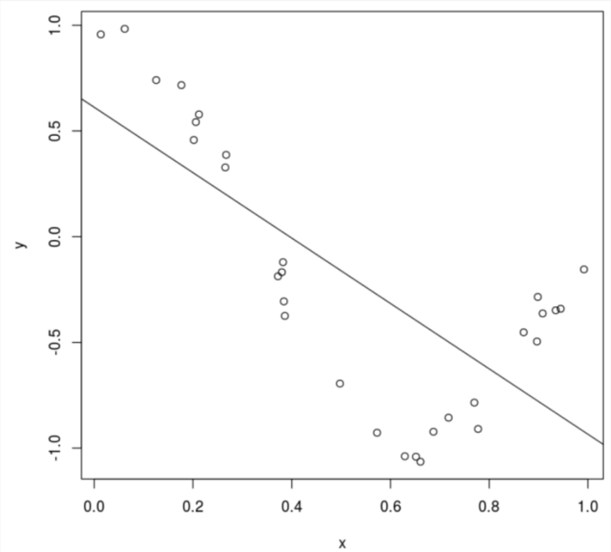
- 다중 공선성 해결법
  - 부분최소제곱회귀와 L1정규화(LASSO)
  - 독립 변수들 중 상관 관계가 높은 변수를 제거
  - 변수를 변형하거나 새로운 관측치 사용
  - 상관 관계의 이유를 파악하여 해결

2 다중 회귀

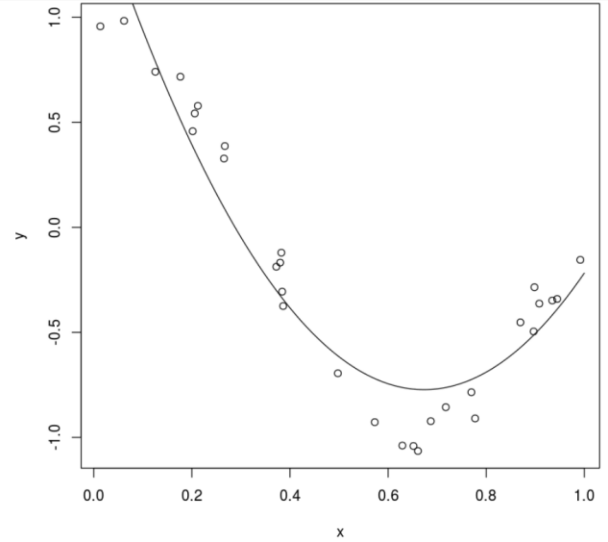
- 다항식 회귀
  - 분석하고자 하는 설명 변수와 종속 변수가 선형적인 관계가 아닐 경우 선형 회귀 모델을 사용하면 큰 오차가 발생

2 다중 회귀

[선형 회귀 모델 시 오차 발생]



[다항식 회귀 모델 로 오차 감소]

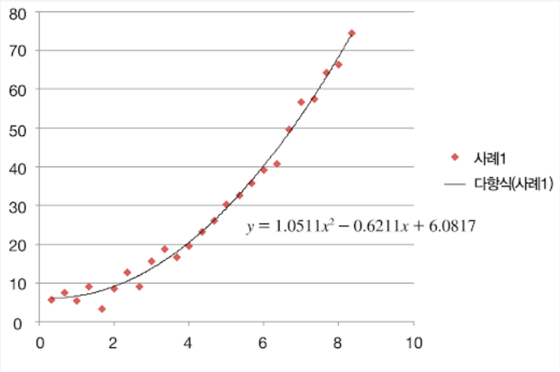


※ 출처 : https://datascienceschool.net/

2 다중 회귀

- 다항식 회귀
  - 산포도의 점 분포가 곡선 상에 위치할 경우 차수를 올려 대응하는 회귀 분석

[다항식 회귀의 사용 예]



[다항식 회귀 식 예]

$$y = \alpha + \beta x + \gamma x^2 + \varepsilon$$

※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

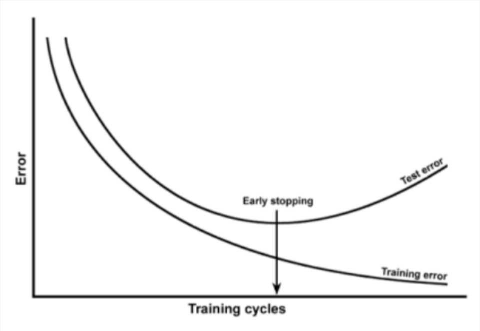
2 다중 회귀

- 과적합(Overfitting)의 문제점
  - 차수가 계속 증가 시 모든 경우의 분포를 적합한 곡선으로 표현 가능한가?
  - 훈련 중인 데이터를 사용할 경우 잔차가 0에 근접
  - 예측 데이터를 사용할 경우 오차가 크게 발생할 확률이 높음

2 다중 회귀

과적합(Overfitting)의 문제점

- 그림은 과적합의 예로 훈련 데이터로는 오차가 적을 수 있으나 어느 순간 실제 데이터를 사용하는 순간 오차가 다시 증가
- 실제 회귀 분석 시 독립 변수의 차수를 낮게 설정하여 과적합 문제를 피하는 것이 중요
- 신경망과 서포트 벡터 머신 등을 활용하여 과적합을 해결



[과적합 예]

※ 출처 : http://untitledtblog.tistory.com/68

2 다중 회귀

- 최소 제곱
  - 회귀분석에서 함수에 피팅할 때에는 잔차가 최소가 되도록 함수를 조정
  - 최소제곱법은 잔차를 최소화 하는 일반적 사용법
  - 단순회귀에서 언급한 연립 편미분 방정식을 만들어 답을 구할 수 있음
  - 독립 변수가 늘고 비선형 함수 모델 사용 시 대응관계가 복잡함

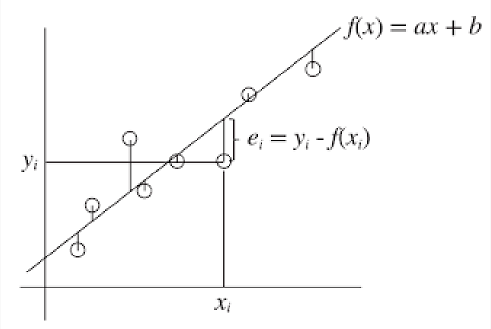
2 다중 회귀

▶ 최소 제곱

[잔차 제곱의 합을 구하는 식]

$$e = \sum_{i=1}^{n}(y_i - f(x_i))^2$$

[최소제곱법 그래프]



※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

2 다중 회귀

- 최소 제곱
  - 행렬을 이용하여 최소제곱법 계산

[행렬을 사용한 최소제곱법 식]

$$f(x) = \alpha + \beta x_1 + \gamma x_2 = \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \end{bmatrix} [\ x_0 \quad x_1 \quad x_2 \quad \dots\ ] = \omega^T X \qquad (x_0 = 1)$$

x : 독립 변수
w : 계수
f(x) : 선형 함수
w^T : w의 전치 행렬

※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

2 다중 회귀

최소 제곱

- 잔차 제곱의 합도 행렬로 표시하여 구할 수 있음

[행렬로 표현한 잔차 제곱 합]

$$E = (Y - \omega^T X)^T (Y - \omega^T X)$$

- 단순 회귀와 같이 E에 관해 w 각각의 성분을 편미분하여 이 값이 0이 되도록 하는 방정식을 행렬로 표현

[정규 방정식] $$X^T X \omega^T = X^T Y$$

※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

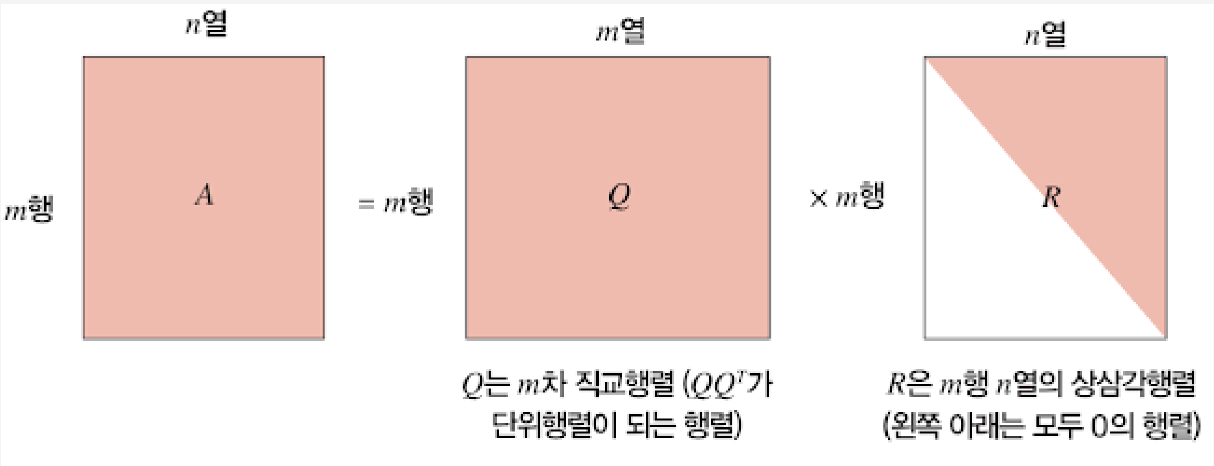
2 다중 회귀

- 정규방정식
  - 행렬 기반의 방정식으로 선형회귀상에서 알지 못하는 값을 예측하는 방법
  - w 를 구할 때는 $w^T=(X^TX)^{-1}X^TY$와 같은 식으로 변형해 X^TX의 역행렬을 구하는 방법 사용
  - 역행렬을 구할 수 없을 때에는
    - QR 분해 알고리즘
    - 특이값 분해

2 다중 회귀

QR 분해

[QR 분해 알고리즘]



※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

2 다중 회귀

▶ R언어에서의 QR 분해 함수

[R 언어의 QR 분해 알고리즘]

```
x <- matrix(1:36, 9)     # 9행 4열의 행렬을 만듭니다
qrval <- qr(x)   # QR 분해를 실행하는 qrval 함수
qr.Q(qrval)      # x를 QR 분해한 Q의 행렬을 구합니다
qr.R(qrval)      # x를 QR 분해한 R의 행렬을 구합니다
```

※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

2 다중 회귀

▶ 파이썬에서의 QR 분해 함수

[파이썬 언어의 QR 분해 알고리즘]

```
import numpy as np
import scipy.linalg  # SciPy 선형대수 라이브러리

x = np.ones((9, 4))
Q, R = scipy.linalg.qr(x)
print(x)
print(Q)
print(R)
```

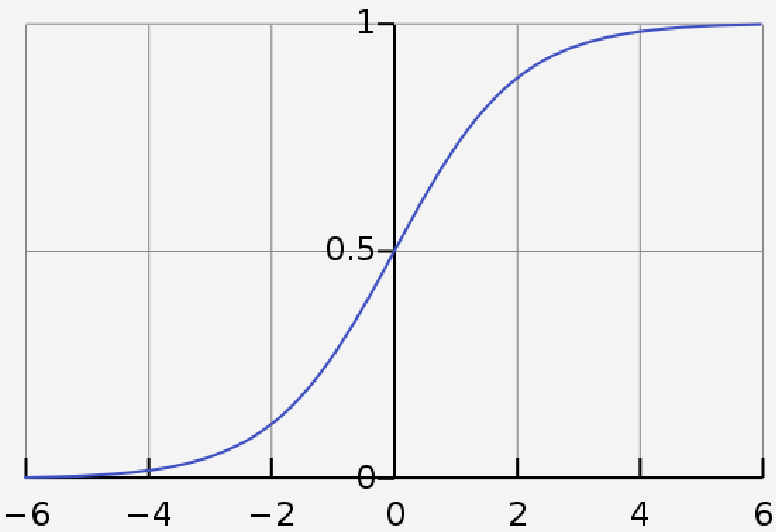
※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

3 로지스틱 회귀

- 종속 변수에 약간의 수정을 가한 선형 회귀
- 다항식 회귀처럼 일반화 선형 모델의 하나로 분류
- 종속 변수가 범주형 데이터를 대상으로 하여
  입력 데이터가 주어졌을 때 해당 데이터의 결과가
  특정 분류로 나뉨 → 분류 기법으로 사용

3 로지스틱 회귀

[로지스틱 함수]



※ 출처 : https://ko.wikipedia.org/wiki/로지스틱_회귀

3 로지스틱 회귀

- 로짓(Logit) 변환
  : 일반식의 종속 변수 y에 로그를 적용하여 y’로 변환하는 것

[로지스틱 모델의 일반식]

$$y = \frac{e^x}{1 + e^x}$$

※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

3 로지스틱 회귀

▶ 로짓 함수를 이용한 함수 피팅

[로지스틱 회귀 모델 식]

$$y' = \ln\left(\frac{y}{1-y}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + \varepsilon$$

$$y' = \beta x + \varepsilon$$

※ 출처 : 처음 배우는 인공지능, 다다 사토시, 한빛미디어, 2017

3 로지스틱 회귀

- 응용
  - 의학 또는 소셜 분석을 포함한 다양한 분야에 사용
  - 예) 부상을 입은 환자들의 사망 예측
    (Trauma and Injury Severity Score (TRISS))
  - 환자의 특정 병 감염 여부를 예측