<데이터분석과기계학습 3주차>

로지스틱 회귀 / SVM

인공지능융합공학부 데이터사이언스전공

곽찬희

지난 시간에 배운 내용

- Perceptron 과 Neuron
  - ✓ Weighted sum, step function, threshold
- Adaline
  - ✓ Objective function, cost function
  - ✓ Activation function
  - ✓ GD, SGD, MBGD
- 그 외에
  - ✓ Standardization
  - ✓ Epoch, learning rate (eta)

3.1. 분류 알고리즘의 선택

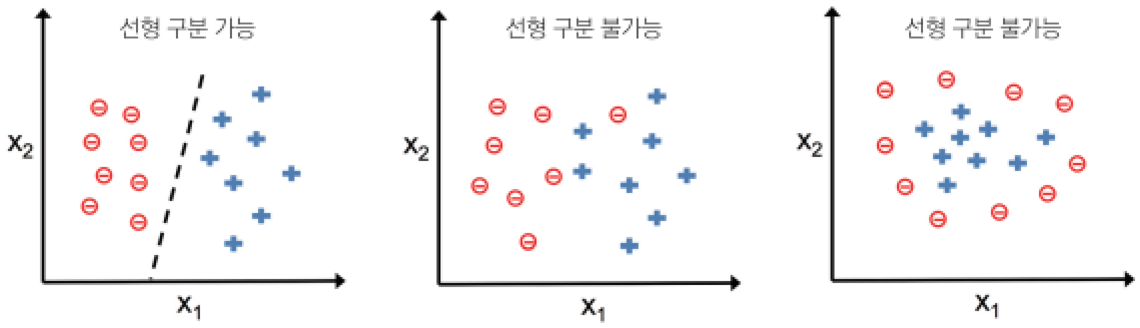
- 모든 경우에 뛰어난 성능을 낼 수 있는 분류 모델은 존재하지 않음
- 몇 개의 알고리즘을 비교해가며 최선의 모델을 선택하는 것이 권장됨
  - ✓ 샘플의 개수, 잡음 데이터의 양, 선형 구분 가능 여부 등등등

- 분류 알고리즘 선택의 5단계
  1. 특성 선택 및 훈련 샘플 준비
  2. 성능 지표 선택
  3. 분류 모델과 최적화 알고리즘을 선택
  4. 모델 성능 평가
  5. 알고리즘 튜닝

3.3. 로지스틱 회귀를 사용한 클래스 확률 모델링

- 퍼셉트론의 가장 큰 단점은 클래스가 선형으로 구분되지 않을 때 사용할 수 없다는 점
  - ✓ 이 경우 학습이 무한으로 반복됨 (수렴하지 않음)

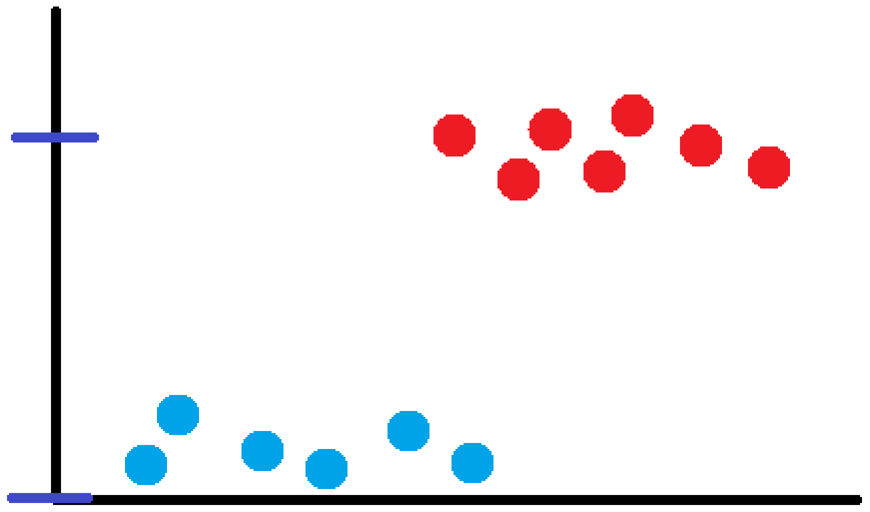
▼ 그림 2-3 선형적으로 구분되는 데이터셋과 그렇지 못한 데이터셋



- 선형 이진 분류에 더 강력한 알고리즘은 로지스틱 회귀 (logistic regression)

3.3. 로지스틱 회귀를 사용한 클래스 확률 모델링

- 직관적인 이해를 먼저 시도해봅시다.
  - ✓ 이진 분류를 한다 -> 결과값(y) 가 0 or 1 이 나와야 한다.



3.3.1. 로지스틱 회귀의 이해와 조건부 확률

- 오즈비 (odds ratio)
  - ✓ 특정 이벤트가 발생할 확률 (발생할 확률 대비 발생하지 않을 확률)

$$\frac{p}{1-p}$$

  - ✓ 예) 동전 던지기의 확률이 ½ 일 때 앞면이 나올 확률 대비 뒷면이 나올 확률의 비율(오즈비)은 1
    만약, 동전을 조작해 앞면의 확률이 2/3 이 된다면, 오즈비는 2

- 오즈비에 자연로그를 취한 값을 로짓 함수라고 부름 (logit function, log + odds)

$$logit(P) = \log\frac{p}{(1-p)}$$

3.3.1. 로지스틱 회귀의 이해와 조건부 확률

- 여기서 수의 범위를 생각해봅시다
  - ✓ p는 확률이므로 0~1의 값을 가짐
  - ✓ 로짓 함수는 -∞~∞ 의 값을 가짐
  - ✓ 이를 반대로 생각해 로짓 함수에 $w^T \cdot x$ 를 넣을 수 있다면 확률로 변환할 수 있음

$$logit(P) = \log \frac{p}{(1-p)}$$

$$\mathrm{logit}\left(\mathrm{P}(\mathrm{y}=1|\mathrm{x})\right) = \mathrm{w}_0\mathrm{x}_0 + \mathrm{w}_1\mathrm{x}_1 + \cdots + \mathrm{w}_\mathrm{m}\mathrm{x}_\mathrm{m} = \sum_{\mathrm{i}=0}^{\mathrm{m}} \mathrm{w}_\mathrm{i}\mathrm{x}_\mathrm{i} = \mathrm{w}^\mathrm{T}\mathrm{x}$$

- 로짓 함수를 p에 대해 정리하자면,

$$e^t = \frac{p}{1-p}$$

$$\frac{1-p}{p} = \frac{1}{e^t}$$

$$\frac{1}{p} = \frac{1+e^t}{e^t}$$

$$p = \frac{1}{1+e^{-t}}$$

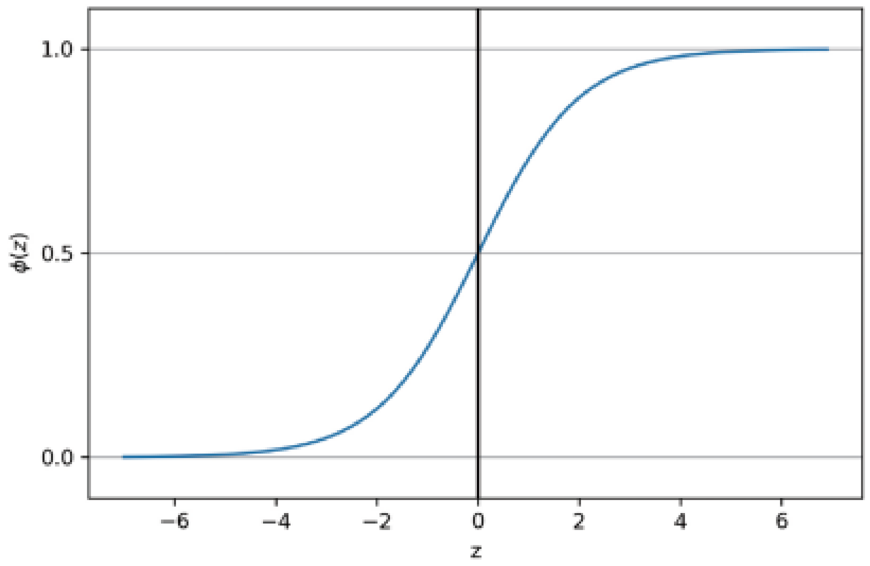
$$= \frac{1}{1+e^{-w^T \cdot x}}$$

3.3.1. 로지스틱 회귀의 이해와 조건부 확률

- (로지스틱) 시그모이드 함수 (logistic sigmoid function)
  - ✓ 로짓 함수의 역함수
  - ✓ ∞로 가면 1에 점근적으로 접근
  - ✓ -∞로 가면 0에 점근적으로 접근
  - ✓ x, w에 대한 시그모이드 함수의 출력은 특정 샘플이 클래스 1에 속할 확률을 의미
    - $\phi(z) = P(y = 1|x; w)$
  - ✓ ϕ (0) = 0.5 를 기준으로 함
  - ✓ 비선형구조!

$$logit(P) = \log\frac{p}{(1-p)} \Rightarrow \phi(z) = \frac{1}{1+e^{-z}}$$

▼ 그림 3-2 시그모이드 곡선



3.3.1. 로지스틱 회귀의 이해와 조건부 확률

- **예)** $\phi(z) = P(y = 1|x;\ w) = 0.8$ **이라면, 클래스 1에 속할 확률이 80%**
- **반대로 클래스 0에 속할 확률은** $P(y = 0|x;\ w) = 1 - P(y = 1|x;\ w) = 0.2$ **즉 20%**
- **이를 임계 함수(threshold function) 에 적용해서 최종적인 판단을 하는데, 이 때 0.5를 기준값으로 가짐**

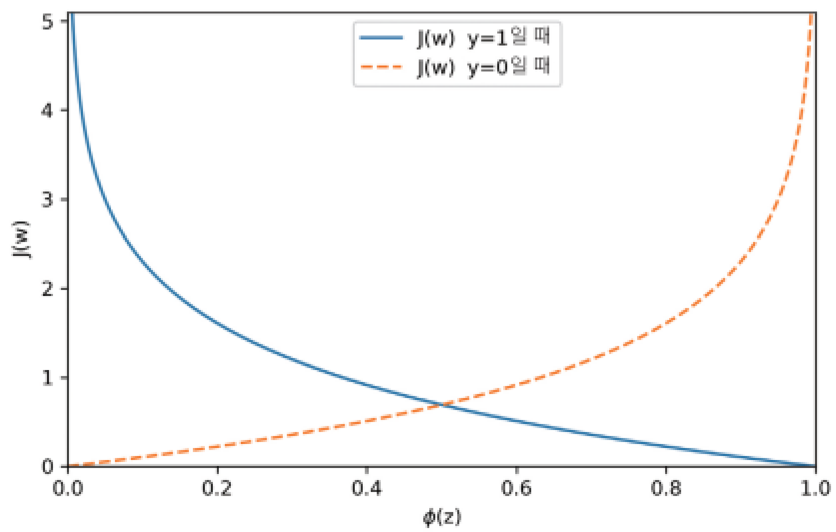
3.3.2. 로지스틱 비용 함수의 가중치 학습

- 지난 주에 배웠던 제곱 오차합 비용 함수는 다음과 같음
  - ✓ $J(w) = \sum_i \frac{1}{2}\left(\phi\left(z^i\right)^i - y^i\right)$
- 로지스틱 회귀 모델에서는 확률을 최대화하려는 가능도 (log likelihood)를 사용
  - ✓ $L(w) = P(y|x;w) = \prod_{i=1}^{n} P\left(y^i \middle| x^i;w\right) = \prod_{i=1}^{n} (\phi\left(z^i\right)^{y^i})(1-\phi\left(z^i\right)^{1-y^i})$
  - ✓ 쉽게 풀어쓰면, 1은 1로, 0은 0으로 분류하는 것을 최대화. Why?
- 위 식에 로그를 취하면 다음과 같음
  - ✓ $l(w) = logL(w) = \sum_{i=1}^{n}\left[\left(y^i \log(\phi(z^i)\right) + \left(1-y^i\right)\log\left(1-\phi(z^i)\right)\right]$
  - ✓ 로그를 취하면, 곱->합 으로 바꿀 수 있어 미분하기 쉬움
  - ✓ Underflow 예방 (수치가 너무 작아 프로그램이 계산하기 어려운 상황)

3.3.2. 로지스틱 비용 함수의 가중치 학습

- 최종적으로 log likelihood 함수를 사용하여 비용 함수를 만들자면 다음과 같음
  - ✓ $J(w) = -\sum_{i=1}^{n}\left[\left(y^i \log(\phi(z^i)) + (1-y^i)\log\left(1-\phi(z^i)\right)\right)\right]$
  - ✓ 이 식은 교차 엔트로피 (Cross entropy) 를 구하는 식과 동일 (엔트로피: 불확실성을 계산)
- 만약 샘플이 1개라면 비용 함수는 다음과 같이 계산됨
  - ✓ $J(\phi(z), y; w) = -\left[(y\log(\phi(z)) + (1-y)\log\left(1-\phi(z)\right)\right]$
  - ✓ 여기서 y값에 따라 최종 값은 다음과 같음
  - ✓ $J(\phi(z), y; w) = \begin{cases} -\log(\phi(z)) & y=1 \\ -\log(1-\phi(z)) & y=0 \end{cases}$

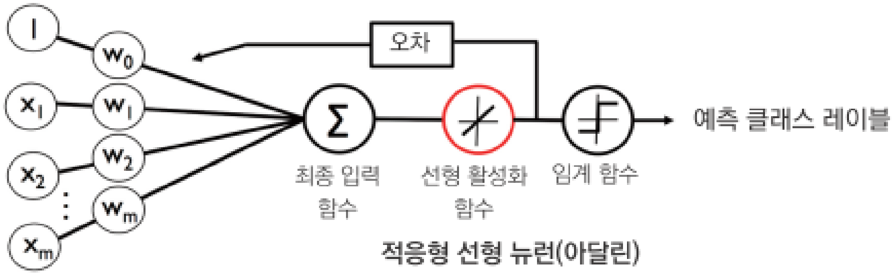
그림 3-4 시그모이드 활성화 대비 로지스틱 비용 그래프

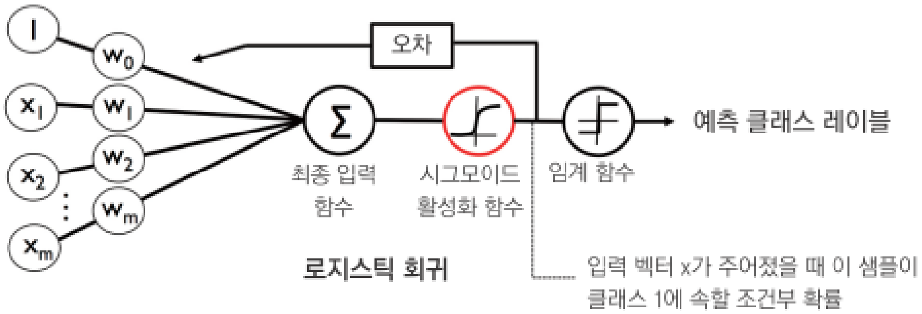


정리하자면

- Input * weights -> weighted sum
- Sigmoid 는 입력을 확률값으로 변환
- Step function 은 확률값을 0/1 로 변환
- 퍼셉트론/아달린과 차이점?
  - ✓ 아달린은 y=x의 선형 활성화 함수
  - ✓ 로지스틱은 sigmoid (비선형 함수)
- 비선형이 왜 중요할까요?

▼ 그림 3-3 아달린과 로지스틱 회귀의 차이점





3.3.5. 규제를 사용하여 과대적합 피하기

- 과대적합(overfitting)
  - ✓ 훈련 데이터에 과하게 학습되어 다른 데이터에서의 성능이 떨어지는 현상
- 과소적합(underfitting)
  - ✓ 모델이 훈련 데이터에서 패턴을 감지하지 못함
- 편향(bias)와 분산(variance, 변동성)의 적당한 절충점을 찾는 것이 필요.

Bias 와 Variance - 쉬운 버전

- LB - LV: 과녁의 중앙에 지속적으로 맞춤
- LB - HV: 평균적으로 중앙을 노리고 있으나 넓게 퍼짐
- HB - LV: 엉뚱한 곳을 지속적으로 맞춤
- HB- HV: 평균적으로 엉뚱한 곳에 넓게 퍼짐

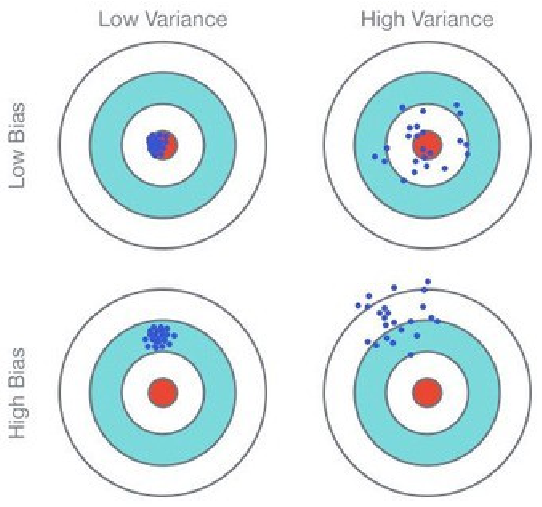


Fig. 1: Graphical Illustration of bias-variance trade-off , Source: Scott Fortmann-Roe., Understanding Bias-Variance Trade-off

Error 분해하기

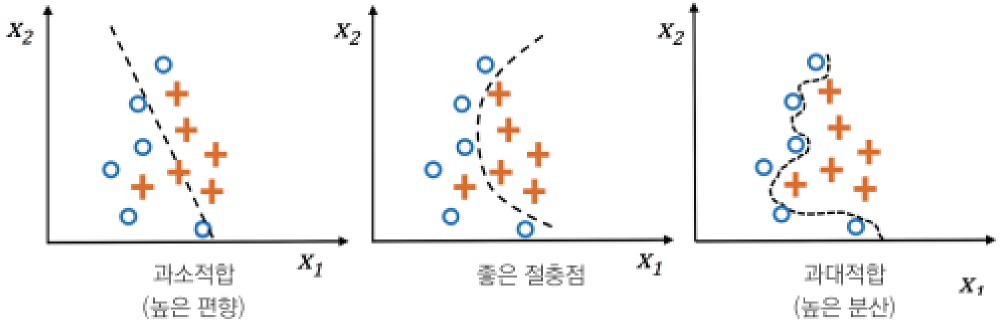
- **MSE (Mean Squared Error) 기준으로 Error를 분해하면 다음과 같다**

✓ $$\begin{aligned} Expected\ MSE &= E\left[\left(Y-\hat{Y}\right)^2\middle|X\right] \\ &= \sigma^2 + \left(E[\hat{Y}]-\hat{Y}\right)^2 + E\left[\hat{Y}-E[\hat{Y}]\right]^2 \\ &= Irreducible\ error + Bias^2\left(\hat{Y}\right) + Variance\left(\hat{Y}\right) \end{aligned}$$

✓ **즉, 애러 = 피할 수 없는 애러 + 편차 + 분산 으로 구성됨.**
- Bias: 평균 추정치와 추정치 간의 차, 추정이 잘못되었을 때 발생, underfitting 과 관련.
- Variance: 추정이 일정하지 않을 때 증가, overfitting 과 관련

✓ **따라서 애러를 줄이기 위해서는 편차를 줄이거나, 분산을 줄이거나, 둘 다 줄여야 함**
- 하지만 하나를 줄이면 하나가 올라감 (bias-variance tradeoff)

그림 3-7 과대적합과 과소적합이 결정 경계에 미치는 영향



3.3.5. 규제를 사용하여 과대적합 피하기

- 규제(regularization)
  - ✓ 공선성(collinearity, 특성 간 높은 상관관계)을 다루거나 데이터에서 잡음을 제거하여 과대 적합을 방지할 수 있는 유용한 방법
  - ✓ L2 규제 (혹은 L2 축소, 가중치 감쇠, Ridge regression) 는 다음과 같음 (λ 는 규제 파라미터)

$$\frac{\lambda}{2}\|w\|^2 = \frac{\lambda}{2}\sum_{j=1}^{m} w_j^2$$

- L2 규제를 적용한 로지스틱 회귀 비용함수는 다음과 같음

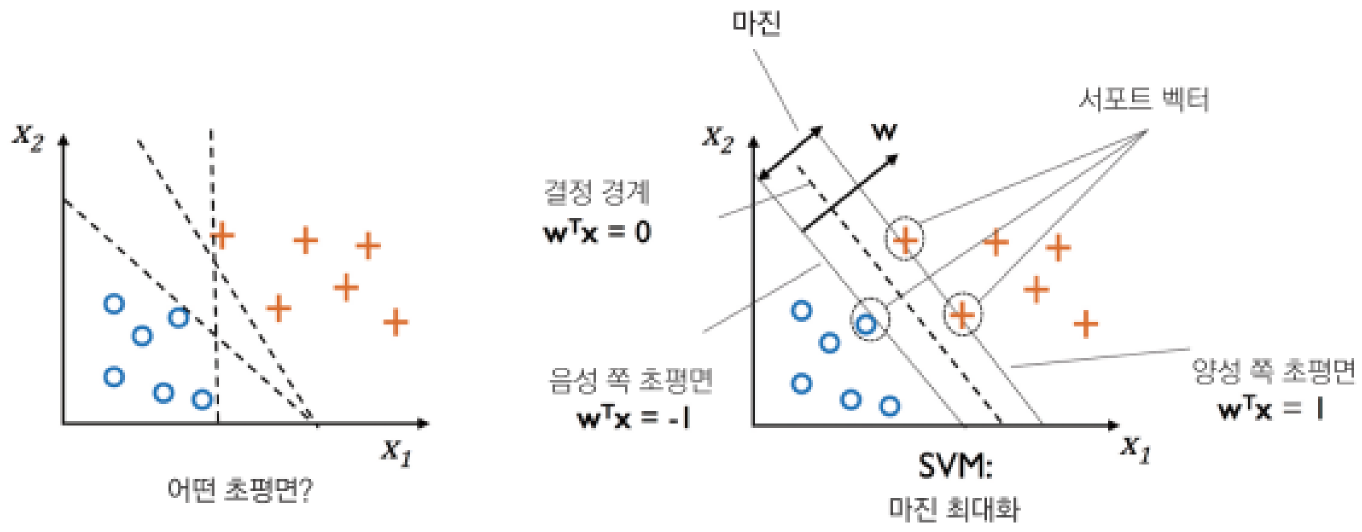
$$J(w) = -\sum_{i=1}^{n}\left[\left(y^i \log(\phi(z^i)) + \left(1 - y^i\right)\log\left(1 - \phi\left(z^i\right)\right)\right] + \frac{\lambda}{2}\|w\|^2$$

- 쉽게 설명하면, λ가 커질 수록 weight 가 감소, λ가 작아질수록 weight 증가 가능
  - ✓ 참고, L1 규제는 $\frac{\lambda}{2}|w|$, Lasso regression

3.4. 서포트 벡터 머신을 사용한 최대 마진 분류

- 서포트 벡터 머신(Support Vector Machine, SVM)
  - ✓ Support Vector 와 결정 경계가 되는 초평면(Hyperplane) 사이의 마진을 최대화하는 것이 목표
  - ✓ (쉽게) 양 집단을 가장 멀리 떨어뜨리기 위해서 어떻게 평면을 만들어야 할까?

▼ 그림 3-9 서포트 벡터 머신



3.4.1. 최대 마진

- 최대 마진을 만드는 이유?
  - ✓ 일반화의 오차를 낮추기 위해 (작은 마진의 모델은 과적합되기 쉬움)

$$w_0 + w^T x_{pos} = 1$$
$$w_0 + w^T x_{neg} = -1$$

  - ✓ 두 식을 정리하면

$$w^T\left(x_{pos} - x_{neg}\right) = 2$$

이 때, 벡터 w의 길이를 $\|w\| = \sqrt{\sum_{j=1}^{m} w_j^2}$ 로 정리하면,

$$\frac{w^T\left(x_{pos} - x_{neg}\right)}{\|w\|} = \frac{2}{\|w\|}$$

로, 좌변이 최대화하고자 하는 마진이 되고, 이는 두 support vector 인 x_{pos} 와 x_{neg} 사이의 거리와 같음. 즉 마진의 최대화 = 거리의 최대화

3.4.1. 최대 마진

- 앞의 제약을 다음과 같이 다시 쓸 수 있음

  $w_0 + w^T x^i \geq 1$ ($y^i = 1$ 일 때)

  $w_0 + w^T x^i \leq -1$ ($y^i = -1$ 일 때)

  i = 1…N 까지 (N은 데이터셋의 샘플 개수)

- 이를 간단히 나타내면 다음과 같음

$$y^i\left(w_0 + w^T x^i\right) \geq 1 \ \forall_i$$

- 실제로는 $\frac{1}{2}\|w\|^2$을 최소화하는 것이 더 쉬움
  - ✓ Quadratic Programming 이용 (이차계획법)

3.4.2. 슬랙 변수를 사용하여 비선형 분류 문제 다루기

- 소프트 마진 분류 (soft margin classification)
  - ✓ 선형적으로 구분되지 않는 데이터에서 선형 제약 조건을 완화하기 위해 슬랙 변수 ξ(Xi, 크사이) 도입

$$w_0 + w^T x^i \geq 1 - \xi^i \ (y^i = 1\text{일 때})$$
$$w_0 + w^T x^i \leq -1 + \xi^i \ (y^i = -1\text{일 때})$$

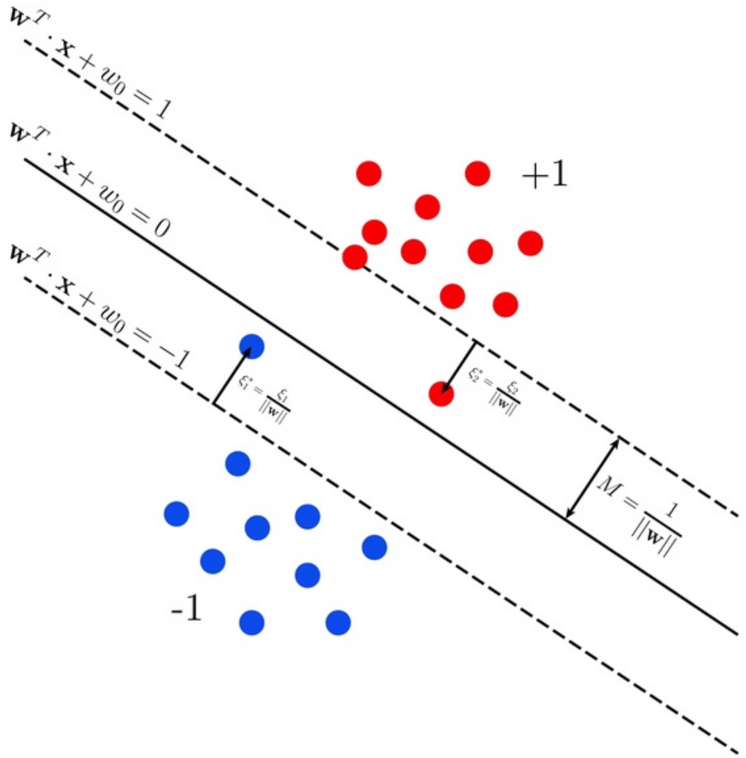
  - ✓ 이 제약 조건에서 최소화할 새 목적 함수는 다음과 같음

$$\frac{1}{2}\|w\|^2 + C\left(\sum_i \xi^i\right)$$

  - ✓ C를 통해 분류 오차 비용 조정 가능
    - C가 크다 -> 오차에 대한 비용이 크다 -> error 를 허용하는 폭이 작다 -> overfitting (복잡한 모델)
    - C가 작다 -> 오차에 대한 비용이 작다 -> error 를 허용하는 폭이 크다 -> underfitting (단순한 모델)

3.4.2. 슬랙 변수를 사용하여 비선형 분류 문제 다루기

- 제약조건을 완화한다 의 의미
  - ✓ SV 안쪽으로 들어가도 어느 정도 허용하겠다



https://qph.fs.quoracdn.net/main-qimg-28d2b87b43129f6ee789bb3f646870d0

3.5. 커널 SVM을 사용한 비선형 문제 풀기

- 선형(직선)으로 구분되지 않는 분류 문제를 풀기 위해 커널 SVM을 도입
- 고차원으로 문제를 변형

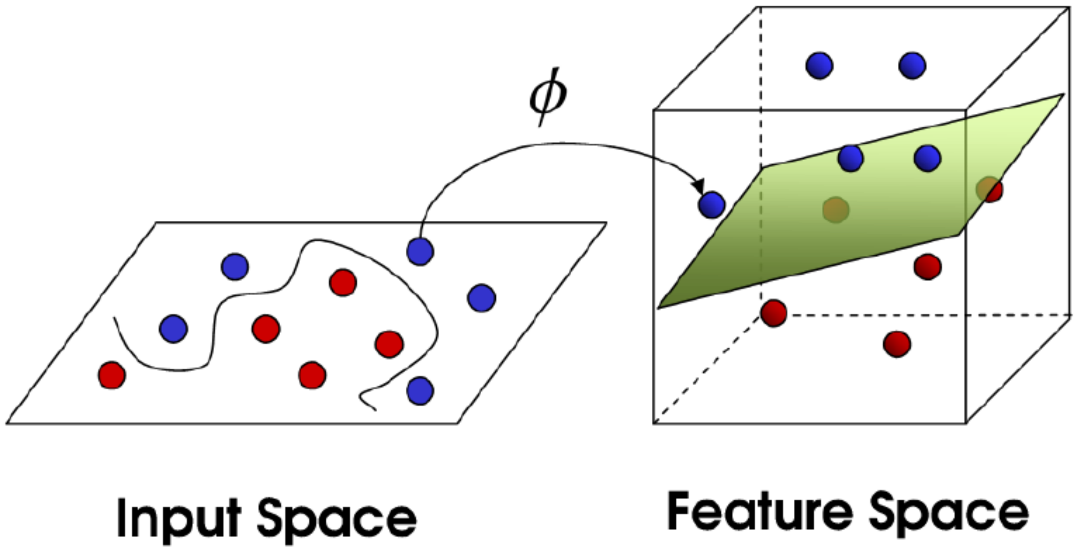
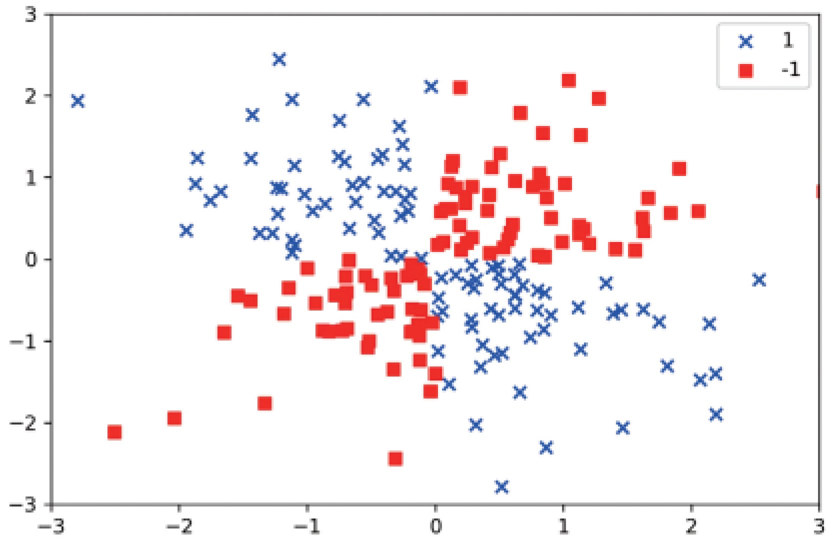


그림 3-12 간단한 XOR 데이터셋



https://ratsgo.github.io/machine%20learning/2017/05/30/SVM3/

강남대학교 KANGNAM UNIVERSITY

3.5.1. 비선형 데이터를 위한 커널 방법

- **Kernel method**
  - ✓ 매핑함수를 사용하여 원본 특성의 비선형 조합을 선형적으로 구분되는 고차원 공간에 투영
  - ✓ 예를 들어

$$\phi(x_1, x_2) = (z_1, z_2, z_3) = (x_1, x_2, x_1^2 + x_2^2)$$

그림 3-12 간단한 XOR 데이터셋

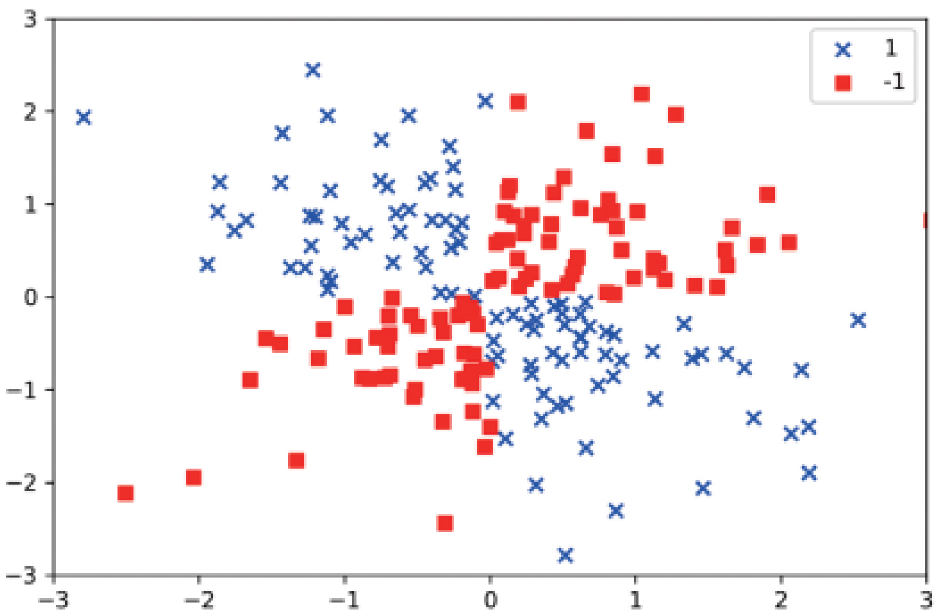
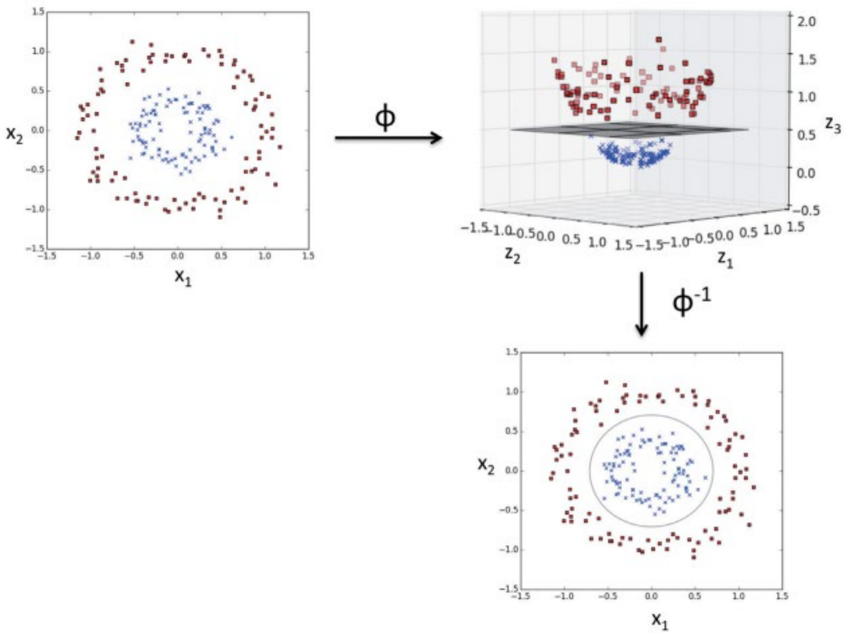


그림 3-13 고차원 공간에서 찾은 결정 경계의 예



3.5.2. 커널 기법을 활용해 고차원 공간에서 분할 초평면 찾기

- 커널을 이용한 방식의 문제점: 새로운 특성을 만드는 계산 비용이 비쌈
- 두 포인트 사이의 점곱 $\phi(x^i)^T\phi(x^j)$을 효율적으로 계산하기 위해 커널 함수 (kernel function)을 $\kappa(x^i, x^j) = \phi(x^i)^T\phi(x^j)$ 로 정의함
- 가장 널리 사용되는 커널 중 하나는 방사 기저 함수 (Radial Basis Function, RBF, 또는 가우시안 커널, Gaussian kernel)

$$\kappa(x^i, x^j) = \exp\left(-\frac{\|x^i - x^j\|^2}{2\sigma^2}\right) = \exp\left(-\gamma\|x^i - x^j\|^2\right)$$

γ 값이 작다 ->지수값이 1에 가까워짐 -> 샘플의 유사도를 높게 판단 -> 단순한 결정 경계
γ 값이 크다 ->지수값이 0에 가까워짐 -> 샘플의 유사도를 낮게 판단 -> 복잡한 결정 경계

- 커널은 유사도 함수(similarity function)으로도 해석이 가능
  - ✓ 음수 부호가 거리 측정을 유사도 점수로 바꾸는 역할
  - ✓ 1(매우 비슷한 샘플) - 0(매우 다른 샘플)의 범위를 가짐

자…

- 이해를 위해 노력을 해봅시다.
- 이해가 되지 않는 부분은 제게 알려주시면 다시 더 쉬운 설명을 생각해보겠습니다.
- 개념과 수학 간의 관계에 대해서 잘 생각해보셨으면…