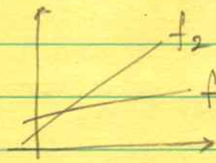
#기계학습

3/10.

기계 학습 개념.

· 가로축 → 특징.

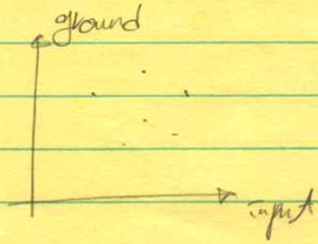
· 세로축 → 목표치.

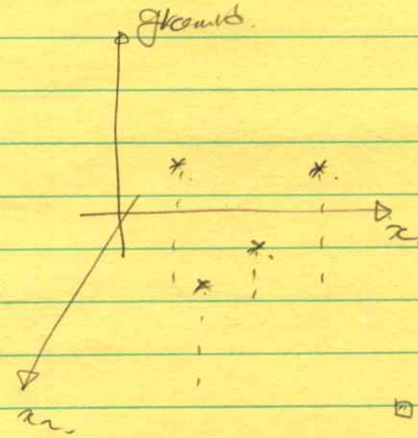


$\sim\rightarrow \underset{*}{w}x + \underset{*}{b}$

$*$: 학습 대상.

- 특징 공간.





⇒ d-차원...

· 데이터베이스의 중요성...

⇒ DB↑ ⇒ 정확도↑ ... 가격↑.

간단한 예시.

$f_\theta(x_i)$: 예측함수의 출력.

y_i : 목표값

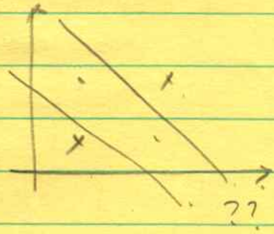
$$Loss^{MSE} = J(\theta) = 1/n \cdot \sum_{i=1}^{n} (f_\theta(x_i) - y_i)^2$$

$$\overset{\text{변수}}{\theta_1} = (w_1, b_1)^T \rightarrow \theta_2 \rightarrow \theta_3 \dots \rightarrow \hat{\theta}$$

$$\underset{\theta}{\operatorname{argmin}}\ \underbrace{J(\theta)}_{\text{최소화시키는 } \theta?} = \hat{\theta}$$

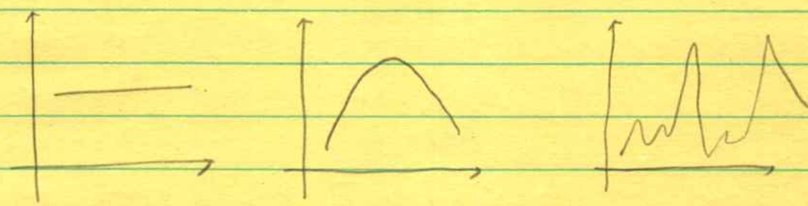
#기계학습.
3/10.

비선형 모델. 자주 나타나는 문제들.



→ 비선형 모델 필요.

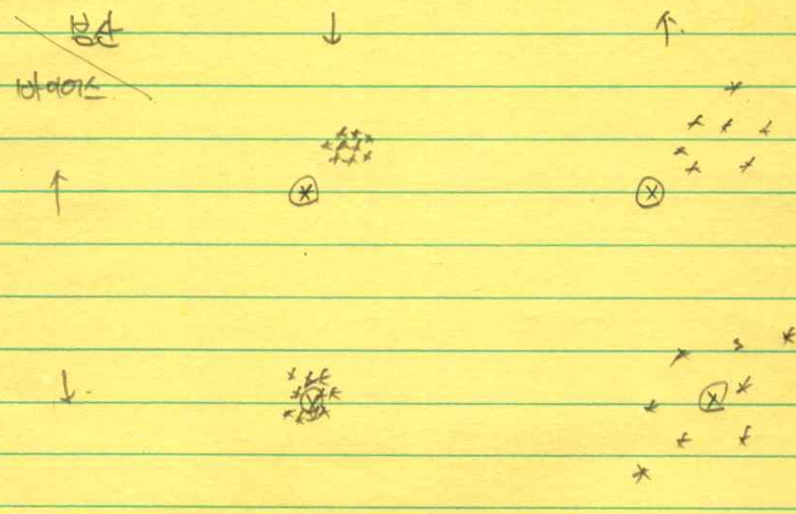
. 1차 . 2차 ... N차



과소적합 ←→ 과대적합.

적절한 모델과 수준.

. 바이어스.



* 기계학습.
3/10.

검증

in: 모델 집합 ⇒ 학습집합, [검증집합], 테스트 집합.

out: 최적 모델, 성능.

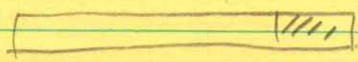
훈련 → 학습 → 검증 → 테스트 저장...

반복

가장 높은 성능의 모델

부트스트랩.

· 난수를 이용해 샘플링.



임의 비율로 나눠서 각 집합에 사용.

in: 모델 집합, 샘플링 비율 p, 시행 횟수 T.

→ pn개 샘플 훈련집합에서 추출
X. X'

X'에서 모델 학습 → X - X'로 loss 평가..

데이터 확대

· 데이터 ↑ ⇒ 일반화 ↑.

· 인위적으로 데이터를 변형시킴 ... 회전, 와핑, 바이어스 스케일링, 노이즈, 색감 등 ...

가중치 규제.

$y = 1005.n1\, x^{12} - 2nn44.c\, x^{11} + \cdots$ → 가중치 ⇑.

$J(\theta) = 1/n \cdot \sum_{i=1}^{n} (f_\theta(x_i) - y_i)^2 + \lambda\|\theta\|_2^2$ ~ 가중치 규제항.

#기계학습

#기계학습의 유형.

· 지도학습.

X와 Y 모두 주어짐 ⇒ 회귀 · 분류 문제.

· 비지도학습.

X 주어지나 Y 주어지지 않음. → 군집화 · 밀도추정 등.

· 강화학습.

목표가 주어짐 · 보상 제공.

· 준지도학습.

X 주어짐 · 일부만 Y 주어짐.

· 결정론적 학습 · 스토캐스틱 학습.

결정론적 → 항상 정확히 재현.
스토캐스틱 → X.

∴ 스토캐스틱에 시드 설정하여 결정론적으로

기계학습

기계학습과 수학.

· 수학 → 목적함수 정의. 최적화 이론 제공.

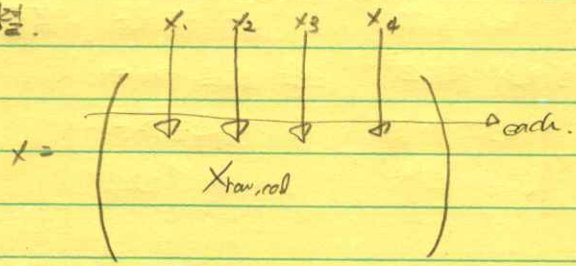
벡터

· 특징을 벡터로 표현 — 특징 벡터

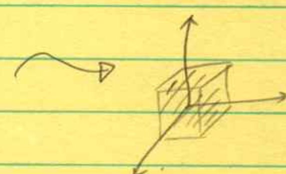
· 여러 특징 벡터: 첨자로 구분.

$$x_1 = \begin{pmatrix} 1.00, \\ 2.11, \\ 3.23 \end{pmatrix}, \quad x_2 = \begin{pmatrix} 2.71, \\ 3.91, \\ 4.20 \end{pmatrix}, \quad x_3 = \begin{pmatrix} 4.12, \\ 3.38, \\ 0.19 \end{pmatrix}.$$

행렬.



* 행렬로 간결하게 수식 같은 표현 가능.



⇒ 사진의 경우, $w \times h \times 3$ (channel)

나중에 채널은 앞에 옴.

↳ $3 \times ? \times ?$

텐서 → 3차 이상의 숫자 배열 구조.

*

#기계학습.

놈과 유사도.

· 벡터의 p차 놈.

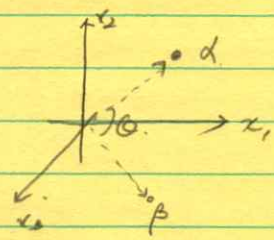
$$\|x\|_p = \left(\sum_{i=1}^{d} |x_i|^p \right)^{1/p}$$

· $x = (3\ \ -4\ \ 1)$ 의 2차 놈: $\|x\|_2 = (3^2 + (-4)^2 + 1^2)^{1/2} = 5.099$

· 유사도.

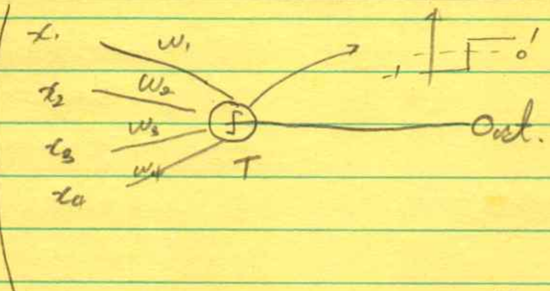
$$\left.\begin{aligned} \cos(0) &= 1 \\ \cos(\pi/2) &= 0 \end{aligned}\right\}$$ 주기함수. → 유사도에 활용.

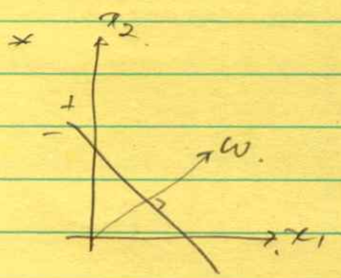
$$\text{Cosine\_Similarity}(a,b) = \frac{a}{\|a\|} \cdot \frac{b}{\|b\|} = \cos(\theta)$$



퍼셉트론.

$$f(\ ) = w_1 x_1 + w_2 x_2 + w_3 x_3 + w_4 x_4 + b$$





$$\left.\begin{aligned} w &= (w_0,\ w_1\ \ w_2\ \cdots\ w_d)^T \\ x &= (1\ \ x_1\ \ x_2\ \cdots\ x_d)^T \end{aligned}\right\} \quad f = w^T x$$

$$\rightsquigarrow \overset{out}{O} = \tau(w \cdot x) \text{ → 활성함수}$$

$$\tau(a) = \begin{cases} 1 & a \geq T \\ -1 & a < T \end{cases}$$

* 기계학습.

03/17.

\# 출력이 여러개인 퍼셉트론.

· 출력은 벡터 $o = (o_1, o_2, \cdots, o_c)^T$ 로 표기

· j번째 퍼셉트론 가중치 $w_j = (w_1, w_2, \cdots w_c)^T$ 로 표기.

$$\Rightarrow o = \tau\begin{pmatrix} w_1 \cdot x \\ w_2 \cdot x \\ \vdots \\ w_c \cdot x \end{pmatrix}.$$

* 분류: $\overset{?}{o} = \tau(\overset{\text{알}}{w} \cdot \overset{\text{알}}{x})$

학습: $\overset{\text{알}}{o} = \tau(\overset{?}{w} \cdot x)$

\# 선형결합과 벡터공간.

$c = \alpha_1 a + \alpha_2 b$. → 기저 벡터 a와 b의 선형결합.

* 두 벡터가 독립이다. → · 상수배가 아니다. · 평행하지 않다.

· 외적 결과가 0이다 · 기울기가 다르다

· sin 결과가 0이다

독립 벡터의 결합으로 벡터공간 만듦.

\# 역행렬.

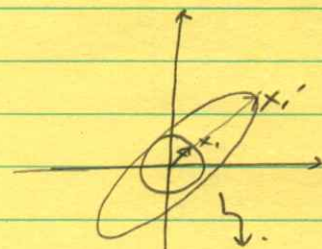
· 행렬식 det → $\det\begin{pmatrix} A & B & C \\ D & E & F \\ G & H & I \end{pmatrix} = AEI + BFG + CDH - (CEG + BDI + AFH)$

· 기하적 의미 2차원 상 넓이, 3차원 상 부피

기계학습.

고유값과 고유 벡터의 기하학적 해석.

$$x_1' = \underset{A}{\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}} \underset{x_1}{\begin{pmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix}} = \begin{pmatrix} 3/\sqrt{2} \\ 3/\sqrt{2} \end{pmatrix}$$



$\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$ 변환 연상.

* 회전 행렬

$$R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}.$$

* 스케일 행렬.

$$S = \begin{bmatrix} C & 0 \\ 0 & C \end{bmatrix} \rightarrow \text{대각선에 상수값 ... 크기 변환.}$$

$\Rightarrow A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \rightarrow$ 스케일을 키움.

$$\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

x축 y축.

A의 고유값과 같은 방향의 $\begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix}$

$x_1 = \begin{pmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix}$, $x_2 = \begin{pmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{pmatrix}$는

스케일이 변해도 방향은 유지.

기계학습

#확률과 통계. → 기계학습이 처리할 데이터는 불확실한 세상에서 발생.

⇒ 불확실성이 다루는 통계·확률 처리

·확률 변수 random variable.

윷놀이 — 모: 1/16, 도: 4/16.

→ 실제로는 윷보다 덜 나옴.

Iris — 확률 변수: 4개 (꽃받침 길이, 꽃받침 너비, 꽃잎 길이, 꽃잎 너비)^T

⇒ 4차원.

#곱 규칙과 합 규칙.

·곱 규칙: $P(y,x) = P(x|y)\,P(y)$.

·합 규칙: $P(x) = \sum_y P(y,x) = \sum_y P(x|y)\,P(y)$.

⇒ 하얀 공이 뽑힐 확률.

$P(\text{하양}) = P(\text{하양}|x)\,P(x) + P(\text{하양}|y)\,P(y) + P(\text{하양}|z)\,P(z)$

"하얀 공이 나왔다는 사실만 알고, 어느 병에서 나왔는지 잘 모를 때, 어느 병에서 나왔는지 추정하라"

$$\hat{y} = \underset{y}{\operatorname{argmax}}\ P(\underbrace{y}_{\text{병}}\,|\,\underbrace{x}_{\text{하얀공}})$$

#베이즈 정리.

$$\hat{y} = \underset{y}{\operatorname{argmax}}\ P(y|x=\text{하양})$$

$$= \underbrace{\underset{y}{\operatorname{argmax}}}_{\text{사후확률}}\ \frac{\overbrace{P(x=\text{하양}|y)}^{\text{우도}}\ \overbrace{P(y)}^{\text{사전확률}}}{P(x=\text{하양})}$$

$P(①|\text{하양}) = \alpha$ < $P(②|\text{하양}) = \beta$ < $P(③|\text{하양}) = \gamma$ 일 때

→ $P(③|\text{하양})$인 확률 최대.

#기계학습.

평균과 분산.

* 데이터의 요약 정보로서 평균과 분산.

평균 $\mu = \frac{1}{n} \sum_{i=1}^{n} x_i$　　　분산 $\sigma^2 = \frac{1}{n} \sum_{i=1}^{n} (x_i - \mu)^2$

* 평균 벡터와 공분산 행렬.

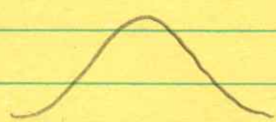
$$\mu = \frac{1}{n} \sum_{1}^{n} x_i . \qquad \Sigma = \frac{1}{n} \sum_{1}^{n} (x_i - \mu)(x_i - \mu)^T .$$

$$= \begin{pmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1d} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2d} \\ \vdots & & \ddots & \vdots \\ \sigma_{d1} & \sigma_{d2} & \cdots & \sigma_{dd} \end{pmatrix} \begin{pmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1d} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2d} \\ \vdots & & \ddots & \vdots \\ \sigma_{d1} & \sigma_{d2} & \cdots & \sigma_{dd} \end{pmatrix}$$

* 가우시안 분포.

$$N(x; \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \cdot \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right)$$

i.e. 0 1 → 표준 정규분포.



#기계학습.

메시지가 가진 정보 ... 수량화하기.

"고비 사막에 눈이 왔다." / "대관령에 눈이 왔다." } → 확률이 낮을 수록 많은 정보.

자기 정보 self information.

사건 e_i의 정보량.

*엔트로피가 높다 → 정보로서 많은 정보.

*엔트로피.

$$H(x) = -\sum_{i,k}^{(\int_x)} P(e_i)\log_2 P(e_i)$$

or $-\sum_{i,k}^{(\int_x)} P(e_i)\log_e P(e_i)$.

#교차 엔트로피. - 두 확률분포 P, Q 사이의 엔트로피.

$$
\begin{aligned}
H(P,Q) &= -\sum P(x)\log_2 Q(x) \\
&= -\sum_{i,k} P(e_i)\log_2 Q(e_i) \\
&= -\sum_x P(x)\log_2 P(x) + \sum_x P(x)\log_2 P(x) \\
&\quad - \sum_x P(x)\log_2 Q(x) \\
&= H(P) + \underbrace{\sum_x P(x)\log_2 \frac{P(x)}{Q(x)}}_{\text{KL 다이버전스}}
\end{aligned}
$$

#기계학습

#최적화와 미분.

#매개변수 공간의 탐색.

$J(\theta) \longrightarrow \arg\min J(\theta) \longrightarrow \hat{\theta}$

목적함수 최적화

* 처리 구현.

난수 생성 ⇒ 초기해 θ.

$J(\theta)$가 작아지는 방향 $d\theta$.

$\theta = \theta + d\theta$.

반복.

$\hat{\theta} = \theta$.

$\theta = \{w, b\}$.

$J(\theta) = 1/n \cdot \sum_{i=1}^{n} (f_\theta(x_i) - y_i)^2$ → 도함수 $J'(\theta)=0$을 풀어 구함.

선형회귀.

→ 최적 직선 방정식.

$\theta = \{w\}$.

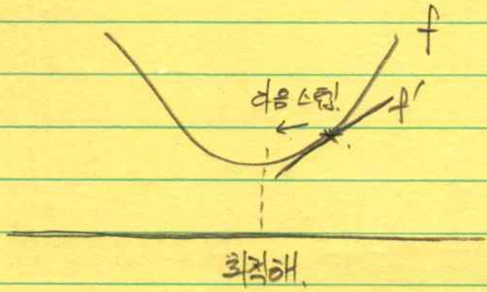
$J(w) = \sum_{x_k \in Y} -y_k (w^T \cdot x_k)$. → 스토캐스틱 경사 하강법. , 퍼셉트론

Loss → 최적 추정치.

$\theta = \{w_1, w_2, \ldots w_k\}$. → 스토캐스틱 경사 하강법. , 딥러닝.

$e = 1/2 \|y - o\|_2^2$.

→ 최적 추정치.



$f_{x+1} = f_x - f'_x \cdot \rho$ 학습률.

*보통은 전역 최적해와 유사한 지역 최적해 찾고 완료.

기계학습.

편미분. → 미분값이 이루는 벡터 「그래디언트.」

∇f, $\partial f / \partial x$, $(\partial f / \partial x_1, \partial f / \partial x_2)$ ...

⇒ 편미분으로 찾은 그래디언트를 이용해 최적해에 접근.

오류 역전파

(# 연쇄법칙.)

$$f(x) = g(h(x))$$
$$f'(x) = g'(h(x))\, h'(x).$$

∝ Batch Gradient Descent.

스토캐스틱 경사 하강 Stochastic Gradient Descent.

한 샘플의 그래디언트 계산 후 즉시 갱신

배치 경사 하강 Batch Gradient Descent.

샘플의 그래디언트를 평균한 후 한 번에 갱신.

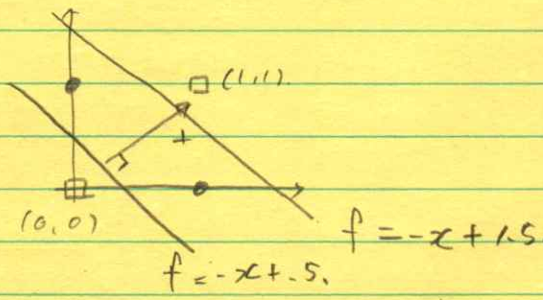
…기계학습과 수학

기계 있음.

신경망.

· 깊이 깊어짐 ⇒ 비선형 ⇑.
그레디언트 ⇑. → 제어 어려움.
피드백 ↓

* 비선형 XOR 문제



목적함수 설계.

$J(\theta)$ 혹은 $J(w)$ → 조건.
· $J \geq 0$
· w^* 최적이면 $J(w) = 0$.
· 틀리는 샘플 ↑ 많을수록 $J(w)$ … ↑.

$= \sum_{x \in Y} -y_k (w^T x_k)$ 일때

가중 정답. 측정.

1	-1	→ ?
1	1	→
-1	1	→

Y는 w^* 틀리는 샘플의 집합.

2

기계학습.

그레이디언트.

$$\frac{\partial J(w)}{\partial w_i} = \sum \frac{\partial(-y_k(w_0 x_{k0} + w_1 x_{k1} + \cdots + w_i x_{ki} + \cdots + w_d x_{kd}))}{\partial w_i}$$

$$= \sum_{x_k \in Y} -y_k x_{ki} \quad (i=0,1,\cdots,d).$$

$$w_i = w_i + \rho \sum_{x_k \in Y} y_k x_{ki} \quad (i=0,1,\cdots,d).$$

학습.

훈련집합 X와 Y에 대해, 샘플을 모두 맞추 ($Y=\emptyset$)일 경우 학습 종료.

민스키「Perceptrons」

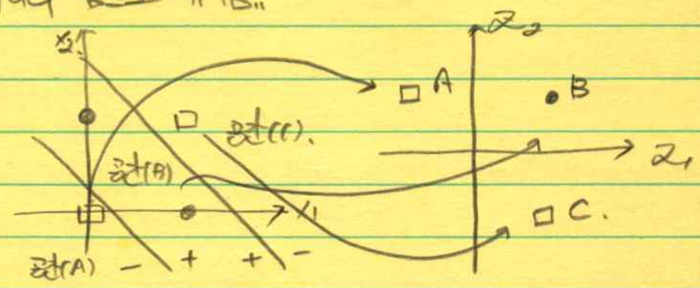
* 퍼셉트론의 한계 → 선형 분류기.

⇒ 다층 퍼셉트론。

루멜하트「Parallel Distributed Processing」

- 은닉층 사용 ··· 특징공간을 분류하기 쉬운 공간으로 변환.
- 시그모이드 활성함수.
- 오류 역전파 알고리즘.

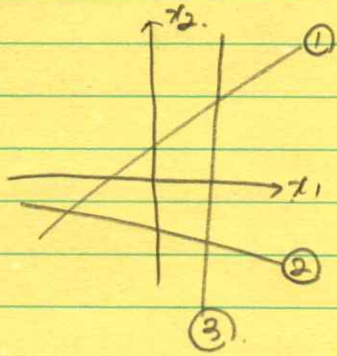
"공간이 하나의 점으로 매핑"



특징공간 X를 특징공간 Z로 변환.

기계학습

특징공간 변환.



	①<0	①>0
②<0	a	b
②>0	c	d

(③>0)

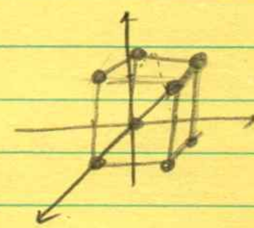
	①<0	①>0
②<0	e	f
②>0	g	h

(③<0).

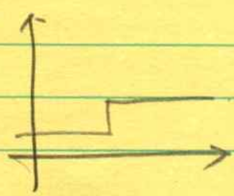
, 퍼셉트론 3개 사용시

, 8개의 공간으로 분할.

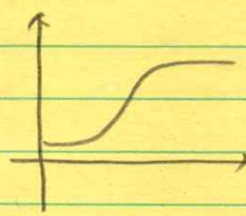
각 공간 "유형"을 점으로 생각하여 새 공간에 매핑.



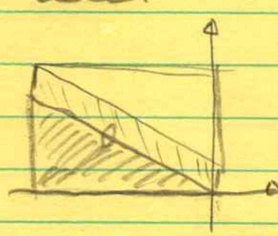
공간분할.



시그모이드의 딱딱한 공간분할.



3개의 sig.의 부드러운 공간분할



기계학습

3/19.

활성함수.

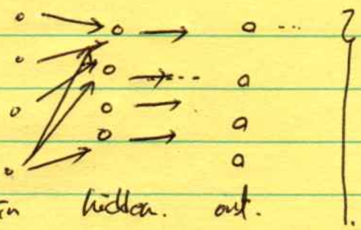
$0 \le f \le 1 \rightarrow (b-a)f \rightarrow (b-a)f+a$

$\Rightarrow 0 \le (b-a)f \le b-a \Rightarrow a \le (b-a)f+a \le b$.

3/24.

구조

· 입력층 - 은닉층 - 출력층.



let.

U^1 입력층 - 은닉층 연결.

U^2 은닉층 - 출력층 연결.

$\cdots U^1 = \begin{pmatrix} u^1_{10} & u^1_{11} & \cdots & u^1_{1d} \\ u^1_{20} & & & \\ \vdots & \vdots & \ddots & \vdots \\ u^1_{p0} & & \cdots & u^1_{pd} \end{pmatrix}$

* j번째 은닉 노드: $z_j = \tau(zsum_j)$, $j = 1, 2, \cdots, p$.

$zsum_j = \underline{u}_j^1 \underline{x}$

$\underline{u}_j^1 = (u^1_{j0}, u^1_{j1}, \cdots u^1_{jd})$,

$\underline{x} = (1, x_1, x_2, \cdots, x_d)^T$. □

오류 역전파 알고리즘의 설계.

미니배치 스토캐스틱 경사하강법. → 한 번에 t개의 샘플처리.

⇒GPU 이용 병렬처리

3/24.
기계학습.

4이킷런.

Model Selection → 경사하강법. / 과적합. → 발생가능 ⇒ 방지책. { regulation. / regularization

* 교차 검증 : 과적합 문제 해결 { K폴드 검증. / stratified K폴드.

* Grid Search CV

데이터 전처리 : 머신러닝 알고리즘 → ** NaN/Null 등 처리 X. / 문자열 값 X. → 인코딩. → label encoding. / one-hot encoding

피처 스케일링 - 정규화.

- Standard Scaler. : 가우시안을 따르도록.
- MinMax Scaler. : 0~1 범위

기계학습.
3/21.

Ensemble 앙상블.

- 어느 수준 이상 분류기 개선 어려움.
- 복수의 약한 분류기 사용.
- 비정형(이미지·영상·음성 등) → 딥러닝 ↑
  정형 → 앙상블 ↑

Voting.

각 결과 → 투표.
여러 알고리즘.

Bagging.

같은 알고리즘.
다른 데이터셋.
(중첩 허용).
→ Voting.

Boosting

여러 알고리즘 순차 학습.
예측 틀린 데이터
→ next 알고리즘은 맞출 수 있게 가중치
i.e. GBM. XGBoost. …

Stacking

- 여러 모델 예측값 → 학습 모델화.
  ⇒ 다른 모델로 재 학습.
  (여러 모델).

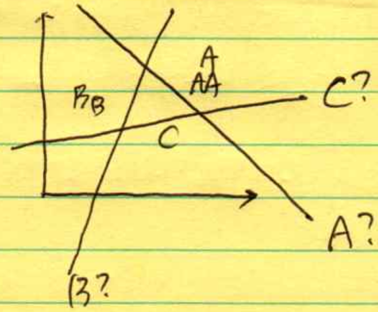
* Hard Voting vs Soft Voting
  → 다수결. → 확률 평균 ↑

Voting Classifier.

* Logistic Regression + KNN
  ⇒ Voting Classifier (soft).

기계학습.

다중 클래스 분류 문제.



* One vs All

*Multi Nomial.

$$\sigma\begin{pmatrix}\theta_0^1 x_0 + \theta_1^1 x_1 + \theta_2^1 x_2 \\ \theta_0^2 x_0 + \theta_1^2 x_1 + \theta_2^2 x_2 \\ \theta_0^3 x_0 + \theta_1^3 x_1 + \theta_2^3 x_3\end{pmatrix} = \begin{bmatrix}2. \\ 1. \\ .1\end{bmatrix} \longrightarrow F\begin{bmatrix}2 \\ 1 \\ 0.1\end{bmatrix} = \begin{bmatrix}.66 \\ .24 \\ .10\end{bmatrix}$$

* Softmax.

$$F\begin{bmatrix}2. \\ 1. \\ .1\end{bmatrix} \longrightarrow S(y^{(i)}) = \frac{e^{y^{(i)}}}{\sum_j e^{y^{(j)}}} \longrightarrow \begin{bmatrix}.66 \\ .24 \\ .10\end{bmatrix}$$

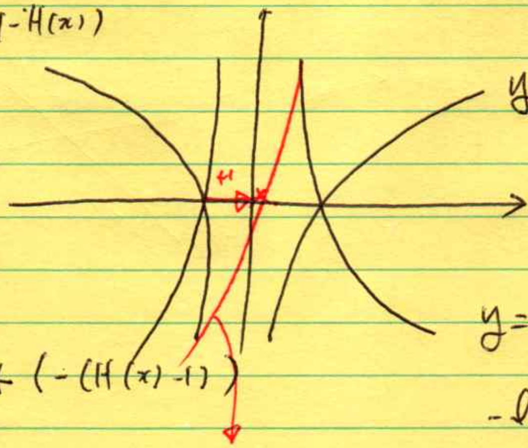
+) Confusion Mat.

* Cross Entropy.

K-최근접 이웃.

- 어떤 데이터 → K개의 가장 가까운 이웃.
- 특징 공간 내의 모든 데이터 필요.
- 컴퓨팅 비용 ↑.
- 단순, 직관 / 학습 단계 필요X.

$y = -\log(1 - H(x))$



$y = f(x).$

$y = -f(x).$

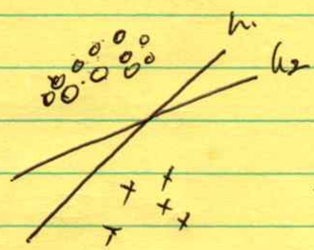
$-y = \log(-(H(x) - 1))$

$-\log(H(x)/1)$

$-y = \log(-H(x)).$

$C: \underset{\text{정답}}{-y} \log(H(x)) - (\underset{\text{오답}}{1-y}) \log(1 - H(x))$

Support Vector Machine.



h_1 : G 나쁜 분류.

마진 내에 G 적은 값이 모이게

⇒ 법선 상 거리 ↑ 하는 선 결정.