Information Theory

Jin Young Choi
Seoul National University

Outline

- Information
- Entropy
- Cross Entropy
- Error Backpropagation Learning
- Mutual Information
- Kullback Leibler Divergence
- Independent Component Analysis (ICA)
- Learning for ICA
- Blind Source Separation

Information

- Discrete random variable X is defined in the sample set Ψ

$$\Psi = \{x_k | k = 0, \ \pm 1, \ \ldots, \ \pm K\}$$

- Event $X = x_k$ occurs with probability $p_k = P(X = x_k)$

- **Information** ≡ surprise ≡ uncertainty
  The amount of information of the event is related to the *inverse* of the probability of occurrence. That is, the lower the probability p_k is, the more "surprise" there is, and the more "information".

$$I(x_k) = \log(\frac{1}{p_k}) = -\log p_k$$

내일도 지구가 회전한다	$p_k = 1$: 정보(×), surprise(×)
내일 미국이 북한을 공격한다	$p_k \ll 1$: 정보(O), surprise(O)

Information

- base=2 ⇒ 정보단위 bits
- base=e ⇒ 정보단위 nats
- 32 bit : 한 code의 정보는 $I(x_k) = -\log(\frac{1}{2^{32}}) = 32$

① $I(x_k) = 0$ for $p_k = 1$
② $I(x_k) \geq 0$ for $0 \leq p_k \leq 1$
③ $I(x_k) \geq I(x_i)$ for $p_k \leq p_i$

- Entropy : a measure of the *average amount of information conveyed per message*, i.e., expectation of Information

$$H(X) = E[I(X)] = \sum_{k=-K}^{K} p_k I(x_k) = -\sum_{k=-K}^{K} p_k \log p_k$$

Information

- Maximum entropy : when p_k is equiprobable.

$$0 \leq H(X) \leq -\sum_{k=-K}^{K} \frac{1}{2K+1} log \frac{1}{2K+1} = \log(2K+1)$$

- $H(X) = 0$ for an event that $p_k = 1$ o/w $p_k = 0$

- Theorem (Gray 1990)

$$\sum_{k} p_k \log(\frac{p_k}{q_k}) \geq 0$$

- Relative entropy (or Kullback – Leibler divergence)

$$D_{p\|q} = \sum_{x \in \mathrm{X}} p_X(x) \log\left(\frac{p_X(x)}{q_X(x)}\right)$$

where $p_X(x)$ is probability mass ftn.(pmf), $q_X(x)$ is reference pmf

Information

- Relative entropy (or Kullback – Leibler divergence) for neural network

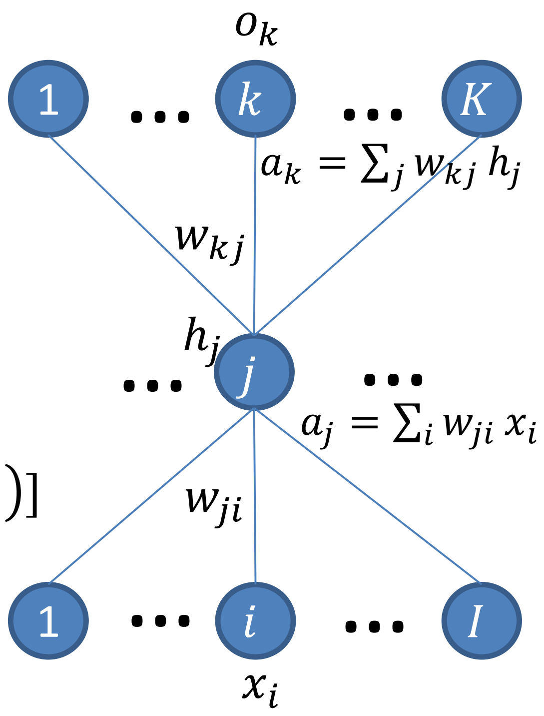
$$D_{p\|q} = \sum_{x\in\mathrm{X}} p(x)\log\left(\frac{p(x)}{q(x;W)}\right) = \sum_{x\in\mathrm{X}} p(x)\log p(x) - \sum_{x\in\mathrm{X}} p(x)\log q(x;W)$$

- Cross entropy for one-hot classification by deep learning

$$C_{p\|q}(x;W) = -\sum_x\sum_k p_k(x)\ \log q_k(x;W)$$

- Cross entropy for multi-label classification by deep learning

$$C_{p\|q}(X;\ W) = -\sum_x\sum_k[p_k(x)\ \log p_k(x;W) + (1-p_k(x))\log\big(1-p_k(x;W)\big)]$$



Backpropagation Learning Rule

- Empirical Risk Function:

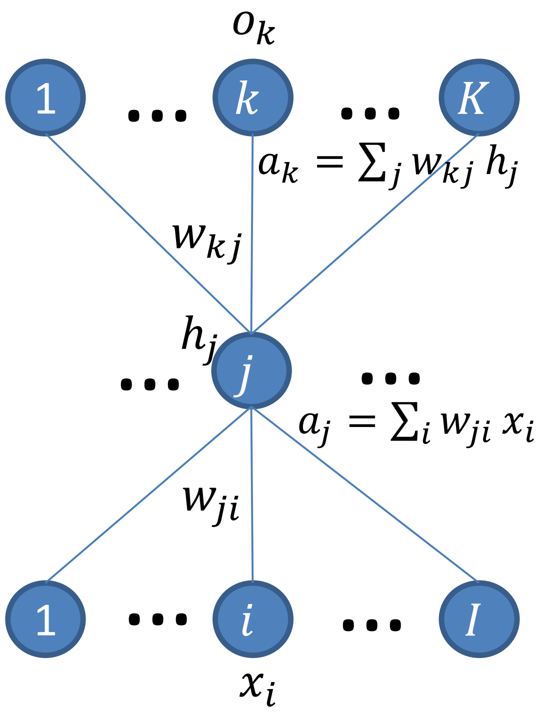
$E_d(w)$ Regression: L_2, linear
01001101: cross-entropy, sigmoid
00001000: cross-entropy, soft-max

- Gradient descent for output layer:

$$\Delta w_{kj} = -\eta \frac{\partial E_d}{\partial w_{kj}}$$

- Chain rule:

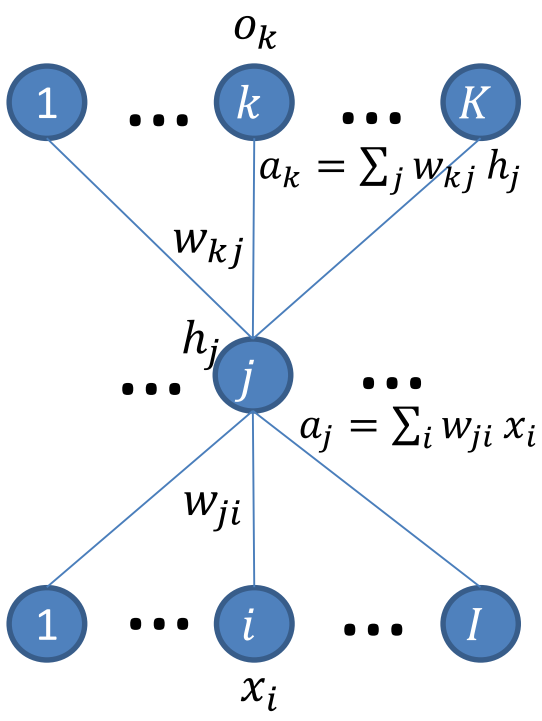
$$\frac{\partial E_d}{\partial w_{kj}} = \frac{\partial E_d}{\partial a_k}\frac{\partial a_k}{\partial w_{kj}} = \frac{\partial E_d}{\partial a_k} h_j$$



Backpropagation Learning Rule

- For multi-label classification (ex, output: 0110100), sigmoid activation function is used and the loss is defined by the cross entropy loss function:
  $E(w) = -\sum_k^K [t_k \log o_k(x,w) + (1-t_k)\log\big(1-o_k(x,w)\big)]$, where
  $o_k = \sigma(a_k) = \frac{1}{1+e^{-a_k}}$. Then find $\frac{\partial \mathrm{E}}{\partial a_k}$.

  Sol.)



$$\frac{\partial E_d}{\partial w_{kj}} = \frac{\partial E_d}{\partial a_k}\frac{\partial a_k}{\partial w_{kj}} = \frac{\partial E_d}{\partial a_k} h_j$$

$$\Delta w_{kj} = -\eta \frac{\partial E_d}{\partial w_{kj}} = \eta \delta_k h_j$$

Backpropagation Learning Rule

- For multi-label classification (ex, output: 0110100), sigmoid activation function is used and the loss is defined by the cross entropy loss function:
  $E(w) = -\sum_k^K [t_k \log o_k(x,w) + (1-t_k)\log(1 - o_k(x,w))]$, where

  $o_k = \sigma(a_k) = \frac{1}{1+e^{-a_k}}$. Then find $\frac{\partial \mathrm{E}}{\partial a_k}$.

  Sol.)

  $\frac{\partial \mathrm{E}}{\partial a_k} = \frac{\partial \mathrm{E}}{\partial o_k} \frac{\partial o_k}{\partial a_k}$.

  $\frac{\partial o_k}{\partial a_k} = \sigma(a_k)(1 - \sigma(a_k)) = o_k(1 - o_k)$,

Backpropagation Learning Rule

- For multi-label classification (ex, output: 0110100), sigmoid activation function is used and the loss is defined by the cross entropy loss function: $E(w) = -\sum_k^K [t_k \log o_k(x,w) + (1-t_k)\log\big(1-o_k(x,w)\big)]$, where $o_k = \sigma(a_k) = \frac{1}{1+e^{-a_k}}$. Then find $\frac{\partial \mathrm{E}}{\partial a_k}$.

Sol.)

$$\frac{\partial \mathrm{E}}{\partial a_k} = \frac{\partial \mathrm{E}}{\partial o_k}\frac{\partial o_k}{\partial a_k}.$$

$$\frac{\partial o_k}{\partial a_k} = \sigma(a_k)\big(1-\sigma(a_k)\big) = o_k(1-o_k).$$

$$\frac{\partial \mathrm{E}}{\partial a_k} = -t_k \frac{1}{o_k}\frac{\partial o_k}{\partial a_k} - (1-t_k)\frac{-1}{1-o_k}\frac{\partial o_k}{\partial a_k}$$

Backpropagation Learning Rule

- For multi-label classification (ex, output: 0110100), sigmoid activation function is used and the loss is defined by the cross entropy loss function: $E(w) = -\sum_k^K [t_k \log o_k(x,w) + (1-t_k)\log\big(1-o_k(x,w)\big)]$, where $o_k = \sigma(a_k) = \frac{1}{1+e^{-a_k}}$. Then find $\frac{\partial \mathrm{E}}{\partial a_k}$.

Sol.)

$$\frac{\partial \mathrm{E}}{\partial a_k} = \frac{\partial \mathrm{E}}{\partial o_k}\frac{\partial o_k}{\partial a_k}.$$

$$\frac{\partial o_k}{\partial a_k} = \sigma(a_k)\big(1-\sigma(a_k)\big) = o_k(1-o_k).$$

$$\frac{\partial \mathrm{E}}{\partial a_k} = -t_k\frac{1}{o_k}\frac{\partial o_k}{\partial a_k} - (1-t_k)\frac{-1}{1-o_k}\frac{\partial o_k}{\partial a_k}$$

$$= -t_k\frac{1}{o_k}o_k(1-o_k) - (1-t_k)\frac{-1}{1-o_k}o_k(1-o_k)$$

Backpropagation Learning Rule

- For multi-label classification (ex, output: 0110100), sigmoid activation function is used and the loss is defined by the cross entropy loss function: $E(w) = -\sum_k^K [t_k \log o_k(x,w) + (1-t_k)\log\big(1-o_k(x,w)\big)]$, where $o_k = \sigma(a_k) = \frac{1}{1+e^{-a_k}}$. Then find $\frac{\partial \mathrm{E}}{\partial a_k}$.

Sol.)

$$\frac{\partial \mathrm{E}}{\partial a_k} = \frac{\partial \mathrm{E}}{\partial o_k}\frac{\partial o_k}{\partial a_k}.$$

$$\frac{\partial o_k}{\partial a_k} = \sigma(a_k)\big(1-\sigma(a_k)\big) = o_k(1-o_k).$$

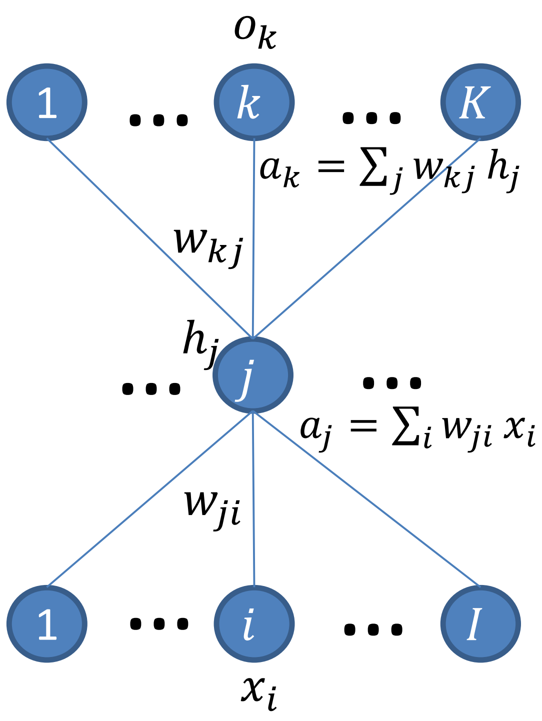
$$\frac{\partial \mathrm{E}}{\partial a_k} = -t_k\frac{1}{o_k}\frac{\partial o_k}{\partial a_k} - (1-t_k)\frac{-1}{1-o_k}\frac{\partial o_k}{\partial a_k}$$

$$= -t_k\frac{1}{o_k}o_k(1-o_k) - (1-t_k)\frac{-1}{1-o_k}o_k(1-o_k)$$

$$= -t_k(1-o_k) + (1-t_k)o_k = o_k - t_k = -(t_k - o_k) = -\delta_k$$

Backpropagation Learning Rule

- For multi-label classification (ex, output: 0110100), sigmoid activation function is used and the loss is defined by the cross entropy loss function:
  $E(w) = -\sum_k^K [t_k \log o_k(x,w) + (1-t_k)\log\big(1-o_k(x,w)\big)]$, where
  $o_k = \sigma(a_k) = \frac{1}{1+e^{-a_k}}$. Then find $\frac{\partial \mathrm{E}}{\partial a_k}$.



Sol.)

$$\frac{\partial \mathrm{E}}{\partial a_k} = \frac{\partial \mathrm{E}}{\partial o_k}\frac{\partial o_k}{\partial a_k}.$$

$$\frac{\partial o_k}{\partial a_k} = \sigma(a_k)\big(1-\sigma(a_k)\big) = o_k(1-o_k).$$

$$\frac{\partial \mathrm{E}}{\partial a_k} = -t_k\frac{1}{o_k}\frac{\partial o_k}{\partial a_k} - (1-t_k)\frac{-1}{1-o_k}\frac{\partial o_k}{\partial a_k}$$

$$= -t_k\frac{1}{o_k}o_k(1-o_k) - (1-t_k)\frac{-1}{1-o_k}o_k(1-o_k)$$

$$= -t_k(1-o_k) + (1-t_k)o_k = o_k - t_k = -(t_k - o_k) = -\delta_k$$

$$\frac{\partial E_d}{\partial w_{kj}} = \frac{\partial E_d}{\partial a_k}\frac{\partial a_k}{\partial w_{kj}} = \frac{\partial E_d}{\partial a_k}h_j$$

$$\Delta w_{kj} = -\eta\frac{\partial E_d}{\partial w_{kj}} = \eta\delta_k h_j$$

Backpropagation Learning Rule

- For multi-class classification (ex, [0 0 0 1 0 0]), the softmax activation function is used and the loss is defined by the cross entropy loss function: $E(w) = -\sum_i^K t_i \log(o_i(x,w))$, where $o_k(x,w) = \frac{e^{a_k}}{\sum_j e^{a_j}}$. The target value $t_k \in \{0,1\}$ is labelled by 1 hot vector. Then find $\frac{\partial \mathrm{E}}{\partial a_k}$.

  Sol.)

$$\frac{\partial E_n}{\partial a_k} = \frac{\partial}{\partial a_k}\left(-\sum_i^K t_i \log\left(\frac{e^{a_i}}{\sum_j e^{a_j}}\right)\right)$$

Backpropagation Learning Rule

- For multi-class classification (ex, [0 0 0 1 0 0]), the softmax activation function is used and the loss is defined by the cross entropy loss function: $E(w) = -\sum_i^K t_i \log(o_i(x,w))$, where $o_k(x,w) = \frac{e^{a_k}}{\sum_j e^{a_j}}$. The target value $t_k \in \{0,1\}$ is labelled by 1 hot vector. Then find $\frac{\partial \mathrm{E}}{\partial a_k}$.

Sol.)

$$\frac{\partial E_n}{\partial a_k} = \frac{\partial}{\partial a_k}\left(-\sum_i^K t_i \log\left(\frac{e^{a_i}}{\sum_j e^{a_j}}\right)\right)$$

$$= \frac{\partial}{\partial a_k}\left(-\sum_i^K [t_i \log(e^{a_i}) - t_i \log(\sum_j e^{a_j})]\right)$$

Backpropagation Learning Rule

- For multi-class classification (ex, [0 0 0 1 0 0]), the softmax activation function is used and the loss is defined by the cross entropy loss function: $E(w) = -\sum_i^K t_i \log(o_i(x,w))$, where $o_k(x,w) = \frac{e^{a_k}}{\sum_j e^{a_j}}$. The target value $t_k \in \{0,1\}$ is labelled by 1 hot vector. Then find $\frac{\partial \mathrm{E}}{\partial a_k}$.

Sol.)

$$\frac{\partial E_n}{\partial a_k} = \frac{\partial}{\partial a_k}\left(-\sum_i^K t_i \log\left(\frac{e^{a_i}}{\sum_j e^{a_j}}\right)\right)$$

$$= \frac{\partial}{\partial a_k}\left(-\sum_i^K [t_i \log(e^{a_i}) - t_i \log(\textstyle\sum_j e^{a_j})]\right)$$

$$= \frac{\partial}{\partial a_k}\left(-\sum_i^K [t_i a_i - t_i \log(\textstyle\sum_j e^{a_j})]\right) = -t_k + \sum_i t_i \frac{e^{a_k}}{\sum_j e^{a_j}}$$

Backpropagation Learning Rule

- For multi-class classification (ex, [0 0 0 1 0 0]), the softmax activation function is used and the loss is defined by the cross entropy loss function: $E(w) = -\sum_i^K t_i \log(o_i(x,w))$, where $o_k(x,w) = \frac{e^{a_k}}{\sum_j e^{a_j}}$. The target value $t_k \in \{0,1\}$ is labelled by 1 hot vector. Then find $\frac{\partial \mathrm{E}}{\partial a_k}$.

Sol.)

$$\frac{\partial E_n}{\partial a_k} = \frac{\partial}{\partial a_k}\left(-\sum_i^K t_i \log\left(\frac{e^{a_i}}{\sum_j e^{a_j}}\right)\right)$$

$$= \frac{\partial}{\partial a_k}\left(-\sum_i^K [t_i \log(e^{a_i}) - t_i \log(\sum_j e^{a_j})]\right)$$

$$= \frac{\partial}{\partial a_k}\left(-\sum_i^K [t_i a_i - t_i \log(\sum_j e^{a_j})]\right) = -t_k + \sum_i t_i \frac{e^{a_k}}{\sum_j e^{a_j}}$$

$$= -t_k + \frac{e^{a_k}}{\sum_j e^{a_j}} \sum_i t_i = o_k - t_k = -(t_k - o_k) = -\delta_k$$

Backpropagation Learning Rule

- For multi-class classification (ex, [0 0 0 1 0 0]), the softmax activation function is used and the loss is defined by the cross entropy loss function: $E(w) = -\sum_i^K t_i \log(o_i(x,w))$, where $o_k(x,w) = \frac{e^{a_k}}{\sum_j e^{a_j}}$. The target value $t_k \in \{0,1\}$ is labelled by 1 hot vector. Then find $\frac{\partial \mathrm{E}}{\partial a_k}$.

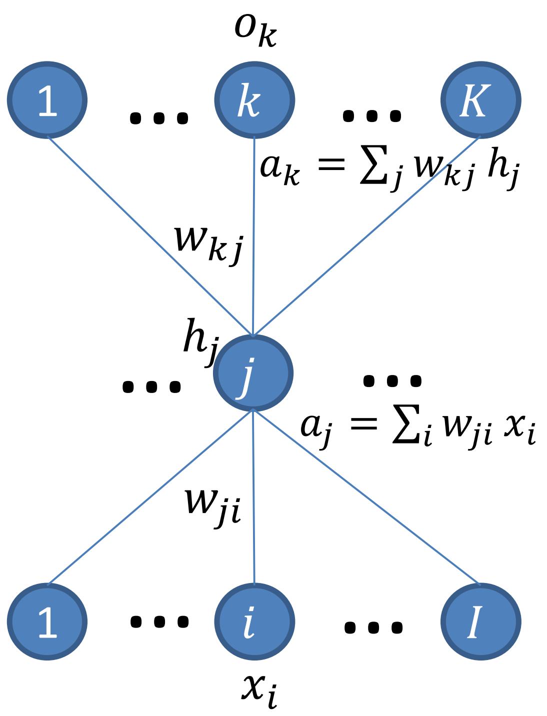
Sol.)

$$\frac{\partial E_n}{\partial a_k} = \frac{\partial}{\partial a_k}\left(-\sum_i^K t_i \log\left(\frac{e^{a_i}}{\sum_j e^{a_j}}\right)\right)$$

$$= \frac{\partial}{\partial a_k}\left(-\sum_i^K [t_i \log(e^{a_i}) - t_i \log(\textstyle\sum_j e^{a_j})]\right)$$

$$= \frac{\partial}{\partial a_k}\left(-\sum_i^K [t_i a_i - t_i \log(\textstyle\sum_j e^{a_j})]\right) = -t_k + \sum_i t_i \frac{e^{a_k}}{\sum_j e^{a_j}}$$

$$= -t_k + \frac{e^{a_k}}{\sum_j e^{a_j}} \sum_i t_i = o_k - t_k = -(t_k - o_k) = -\delta_k$$



$$\boxed{\frac{\partial E_d}{\partial w_{kj}} = \frac{\partial E_d}{\partial a_k}\frac{\partial a_k}{\partial w_{kj}} = \frac{\partial E_d}{\partial a_k} h_j}$$

$$\boxed{\Delta w_{kj} = -\eta \frac{\partial E_d}{\partial w_{kj}} = \eta \delta_k h_j}$$

Backpropagation Learning Rule

- Empirical Risk Function:

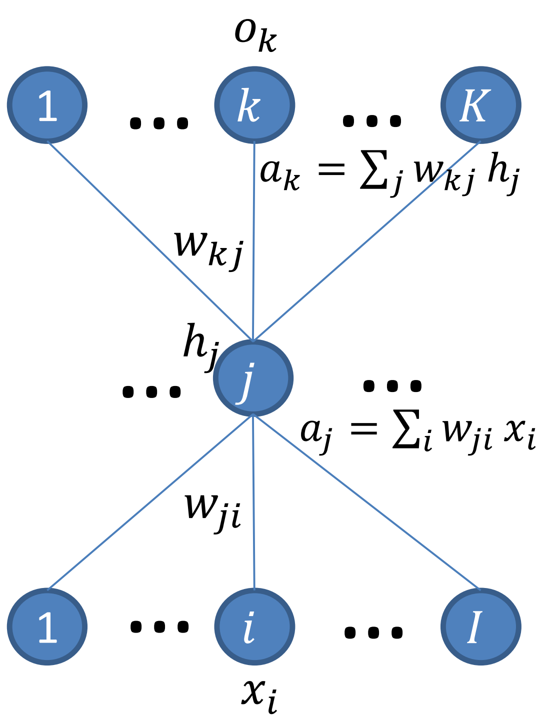
$E_d(w)$ — Regression: L_2, linear
01001101: cross-entropy, sigmoid
00001000: cross-entropy, soft-max

- Gradient descent for hidden layer:

$$\Delta w_{ji} = -\eta \frac{\partial E_d}{\partial w_{ji}}$$

- Chain rule:

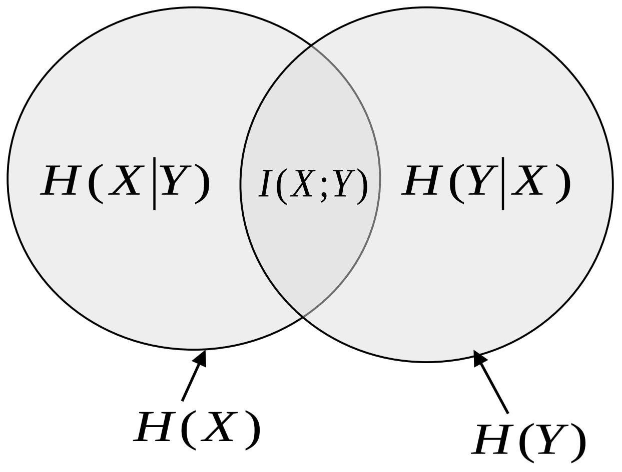
$$\frac{\partial E_d}{\partial w_{ji}} = \frac{\partial E_d}{\partial a_j}\frac{\partial a_j}{\partial w_{ji}} = \frac{\partial E_d}{\partial a_j} x_i$$



Mutual Information

- Conditional Entropy (조건부 불확실성의 량)
  Y 가 관측되고 난 후의 X 의 정보기대치 (Entropy)
  Y 와 연관이 있는 X 의 정보는 제외

- Theorem (Gray 1990)
  $H(X|Y) = H(X,Y) - H(Y)$
  $0 \leq H(X|Y) \leq H(X)$



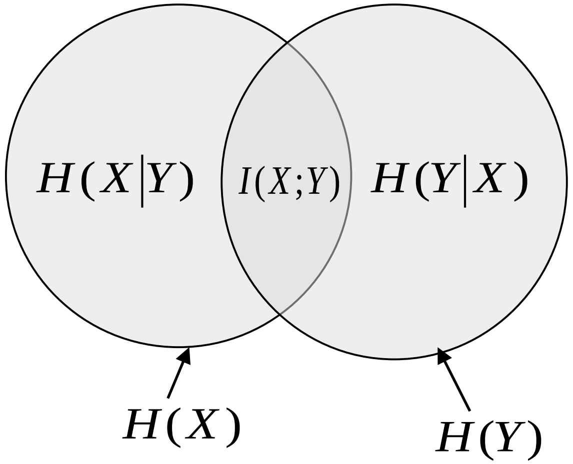
- Joint Entropy
  $H(X,Y) = -\sum_{x \in X} \sum_{y \in Y} p(x,y) \log p(x,y)$

Joint probability mass function

Mutual Information

- Mutual Information: Output Y 의 관측에 의해 알 수 있는 X 의 uncertainty (정보)

$$
\begin{aligned}
I(X;Y) &= H(X) - H(X|Y) \\
&= H(X) + H(Y) - H(X,Y) \\
&= -\sum_{x\in X} p(x)\log(p(x)) - \sum_{y\in Y} p(y)\log(p(y)) \\
&\quad + \sum_{x\in X}\sum_{y\in Y} p(x,y)\log(p(x,y)) \\
&= \sum_{x\in X}\sum_{y\in Y} p(x,y)\log\left(\frac{p(x,y)}{p(x)p(y)}\right)
\end{aligned}
$$

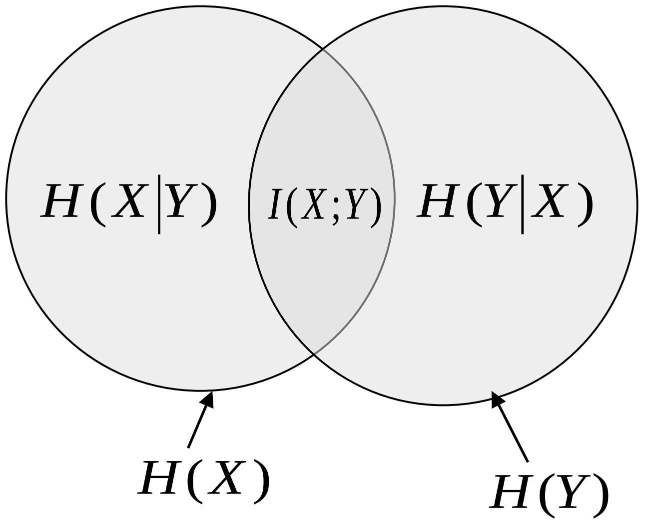


- KL-divergence & Independence ?

$$H(X) = I(X,X)$$

Mutual Information

- Properties of $I(X,Y)$
  ① $I(Y\,;X) = I(X\,;Y)$
  ② $I(X\,;Y) \geq 0$
  ③ $I(X\,;Y) = H(Y) - H(Y|X)$



Mutual Information

- Mutual Information for Continuous Random Variables

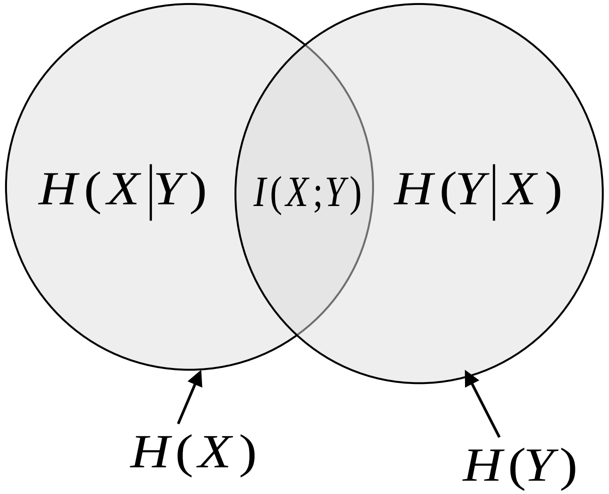
$$I(X;Y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{X,Y}(x,y)\log\left(\frac{f_{X,Y}(x,y)}{f_X(x)f_Y(y)}\right)dxdy$$

$$I(X;Y) = h(X) - h(X|Y) = h(Y) - h(Y|X)$$

$$= h(X) + h(Y) - h(X,Y)$$

$$I(X;Y) = I(Y;X)$$

$$I(X;Y) \geq 0$$



Exercise

- 프리미어리그에서 아스널과 토트넘이 경기를 하고 있다. TV를 보며 마음 껏 떠들 수 있도록 자리가 마련된 치킨집의 식객 30명과 바로 옆 삼겹살 집 식객 60명이 응원전을 펼치고 있다. 치킨집 사람들에게 어느 팀을 응원하는지 물었을 때 토트넘 10명, 아스널을 20명이 응원한다고 답했다.삼겹살 집에서는 각 팀을 몇 명이 응원하고 있는지 확인하지 못했다.
- 치킨 집에서 '토트넘'을 응원한다는 답변에 담긴 정보량(Information Gain)은?

Exercise

- 프리미어리그에서 아스널과 토트넘이 경기를 하고 있다. TV를 보며 마음 껏 떠들 수 있도록 자리가 마련된 치킨집의 식객 30명과 바로 옆 삼겹살 집 식객 60명이 응원전을 펼치고 있다. 치킨집 사람들에게 어느 팀을 응원하는지 물었을 때 토트넘 10명, 아스널을 20명이 응원한다고 답했다.삼겹살 집에서는 각 팀을 몇 명이 응원하고 있는지 확인하지 못했다.
- 치킨 집에서 '토트넘'을 응원한다는 답변에 담긴 정보량(Information Gain)은?

- 일목요연하게 내용 정리.

Exercise

- 프리미어리그에서 아스널과 토트넘이 경기를 하고 있다. TV를 보며 마음 껏 떠들 수 있도록 자리가 마련된 치킨집의 식객 30명과 바로 옆 삼겹살 집 식객 60명이 응원전을 펼치고 있다. 치킨집 사람들에게 어느 팀을 응원하는지 물었을 때 토트넘 10명, 아스널을 20명이 응원한다고 답했다.삼겹살 집에서는 각 팀을 몇 명이 응원하고 있는지 확인하지 못했다.
- 치킨 집에서 '토트넘'을 응원한다는 답변에 담긴 정보량(Information Gain)은?

- 일목요연하게 내용 정리.
- $Information = -\log p(X = x)$

	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60 - n)$명

Exercise

- 프리미어리그에서 아스널과 토트넘이 경기를 하고 있다. TV를 보며 마음 껏 떠들 수 있도록 자리가 마련된 치킨집의 식객 30명과 바로 옆 삼겹살 집 식객 60명이 응원전을 펼치고 있다. 치킨집 사람들에게 어느 팀을 응원하는지 물었을 때 토트넘 10명, 아스널을 20명이 응원한다고 답했다.삼겹살 집에서는 각 팀을 몇 명이 응원하고 있는지 확인하지 못했다.
- 치킨 집에서 '토트넘'을 응원한다는 답변에 담긴 정보량(Information Gain)은?

- 일목요연하게 내용 정리.
- $Information = -\log P(x)$

$P(X = \text{토트넘} | Y = \text{치킨집}) = 1/3$

	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60-n)$명

Exercise

- 프리미어리그에서 아스널과 토트넘이 경기를 하고 있다. TV를 보며 마음 껏 떠들 수 있도록 자리가 마련된 치킨집의 식객 30명과 바로 옆 삼겹살 집 식객 60명이 응원전을 펼치고 있다. 치킨집 사람들에게 어느 팀을 응원하는지 물었을 때 토트넘 10명, 아스널을 20명이 응원한다고 답했다.삼겹살 집에서는 각 팀을 몇 명이 응원하고 있는지 확인하지 못했다.
- 치킨 집에서 '토트넘'을 응원한다는 답변에 담긴 정보량(Information Gain)은?

- 일목요연하게 내용 정리.

- $Information\colon I(x) = -\log P(x)$

$$P(X = \text{토트넘} | Y = \text{치킨집}) = 1/3$$

$$I(X = \text{토트넘} | Y = \text{치킨집}) = \log 3$$

	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60 - n)$명

Exercise

- 치킨집에서 토트넘 응원하는 경우를 $X = 0$, 아스널 응원하는 경우를 $X = 1$이라 할 때 우측 표가 지닌 엔트로피는?

	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60 - n)$명

Exercise

- 치킨집에서 토트넘 응원하는 경우를 $X = 0$, 아스널 응원하는 경우를 $X = 1$이라 할 때 우측 표가 지닌 X 의 엔트로피는?

- $Entropy$: $H(X) = -\sum_{x \in X} p(x) \log p(x)$

	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60 - n)$명

Exercise

- 치킨집$(Y = 0)$에서 토트넘 응원하는 경우를 $X = 0$, 아스널 응원하는 경우를 $X = 1$이라 할 때 우측 표가 지닌 X 의 엔트로피는?

	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60 - n)$명

- $Entropy$: $H(X) = -\sum_x p(x) \log p(x)$
- $H(x|Y = 0) = -\sum_x p(x|Y = 0) \log p(x|Y = 0)$
- $H(x|Y = 0) = -\frac{1}{3}\log\frac{1}{3} - \frac{2}{3}\log\frac{2}{3}$

Exercise

- KL-Divergence의 의미를 생각할 때 각 음식점 에서 두팀을 응원할 확률분포간의 KL-divergence, 즉 $D_{P(X|Y=0)||P(X|Y=1)}$을 최소로 하는 n 값을 구하시오.

$D_{P(Y=0)||P(Y=1)}$을 최소로한다는 것은 각 음식점에서 두팀을 응원할 확률 분포가 같게 된다는 의미이다.

즉, $P(X|Y = 0) = P(X|Y = 1)$

	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60 - n)$명

Exercise

- KL-Divergence의 의미를 생각할 때 각 음식점 에서 두 팀을 응원할 확률분포 간의 KL-divergence, 즉 $D_{P(X|Y=0)||P(X|Y=1)}$을 최소로 하는 n 값을 구하시오.

$D_{P(Y=0)||P(Y=1)}$을 최소로한다는 것은 각 음식점에서 두팀을 응원할 확률 분포가 같게 된다는 의미이다.

즉, $P(X|Y=0) = P(X|Y=1)$

$$\frac{1}{3} = \frac{n}{60}, \qquad \frac{2}{3} = \frac{60-n}{60} \rightarrow n = 20$$

	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60-n)$명

Exercise

- $D_{P(X|Y=0)||P(X|Y=1)}$을 최소로 하는 n 값을 최적화 방법으로 구하시오.

$$n^* = \underset{n}{\operatorname{argmin}} D_{P(X|Y=0)||P(X|Y=1)}$$

	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60-n)$명

Exercise

- $D_{P(X|Y=0)||P(X|Y=1)}$을 최소로 하는 n 값을 최적화 방법으로 구하시오.

	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60-n)$명

$$n^* = \underset{n}{\operatorname{argmin}}\, D_{P(X|Y=0)||P(X|Y=1)}$$

$$D_{P(X|Y=0)||P(X|Y=1)} = \sum_x P(X=x|Y=0) \log \frac{P(X=x|Y=0)}{P(X=x|Y=1)}$$

Exercise

- $D_{P(X|Y=0)||P(X|Y=1)}$을 최소로 하는 n 값을 최적화 방법으로 구하시오.

	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60-n)$명

$$n^* = \underset{n}{\operatorname{argmin}}\, D_{P(X|Y=0)||P(X|Y=1)}$$

$$D_{P(X|Y=0)||P(X|Y=1)} = \sum_x P(X=x|Y=0)\log\frac{P(X=x|Y=0)}{P(X=x|Y=1)}$$

$$= 1/3\log\frac{\frac{1}{3}}{\frac{n}{60}} + 2/3\log\frac{\frac{2}{3}}{\frac{60-n}{60}}$$

Exercise

- $D_{P(X|Y=0)||P(X|Y=1)}$을 최소로 하는 n 값을 최적화 방법으로 구하시오.

	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60-n)$명

$$n^* = \underset{n}{\operatorname{argmin}}\, D_{P(X|Y=0)||P(X|Y=1)}$$

$$D_{P(X|Y=0)||P(X|Y=1)} = \sum_x P(X=x|Y=0)\log\frac{P(X=x|Y=0)}{P(X=x|Y=1)}$$

$$= 1/3\log\frac{\frac{1}{3}}{\frac{n}{60}} + 2/3\log\frac{\frac{2}{3}}{\frac{60-n}{60}}$$

$$\frac{d}{dn}D_{P(X|Y=0)||P(X|Y=1)} = \frac{n}{60}\left(-\frac{20}{n^2}\right) + \frac{60-n}{60}\left(\frac{40}{(60-n)^2}\right) = -\frac{1}{3n} + \frac{2}{3(60-n)} = \frac{-60+3n}{3n(60-n)} = 0 \rightarrow n = 20$$

Exercise

- $D_{P(X|Y=0)||P(X|Y=1)}$을 이용하여 구한 n이 참값이라고 할 때, 위 표가 지닌 응원팀(X)과 음식점(Y)에 관한 Mutual Information I(X, Y)을 수식을 사용하지 않고 개념적으로 구하시오. 그리고 수식을 사용하여 구하여 개념적으로 구한 경우와 비교하시오.

	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60-n)$명

Exercise

- $D_{P(X|Y=0)||P(X|Y=1)}$을 이용하여 구한 n이 참값이라고 할 때, 위 표가 지닌 응원팀(X)과 음식점(Y)에 관한 Mutual Information I(X, Y)을 수식을 사용하지 않고 개념적으로 구하시오. 그리고 수식을 사용하여 구해보고 개념적으로 구한 경우와 비교하시오.

응원팀과 음식점은 서로 독립이다. 그 이유는 음식점에 따라 두 팀을 응원하는 확률 분포가 달라지지 않기 때문이다. 따라서 Mutual Information은 0 이다.

	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60-n)$명

Exercise

- $D_{P(X|Y=0)||P(X|Y=1)}$을 이용하여 구한 n이 참값이라고 할 때, 위 표가 지닌 응원팀(X)과 음식점(Y)에 관한 Mutual Information I(X, Y)을 수식을 사용하지 않고 개념적으로 구하시오. 그리고 수식을 사용하여 구해보고 개념적으로 구한 경우와 비교하시오.

응원팀과 음식점은 서로 독립이다. 그 이유는 음식점에 따라 두 팀을 응원하는 확률 분포가 달라지지 않기 때문이다. 따라서 Mutual Information은 0 이다.

	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60-n)$명

$$\mathrm{I}(X, Y) = \sum_{x,y} p(x, y) \log \frac{p(x,y)}{p(x)p(y)} = \sum_{x,y} p(x|y)p(y) \log \frac{p(x|y)p(y)}{p(x)p(y)}$$

$$= \frac{1}{3}\frac{1}{3}\log\frac{\frac{11}{33}}{\frac{11}{33}} + \frac{2}{3}\frac{1}{3}\log\frac{\frac{21}{33}}{\frac{21}{33}} + \frac{1}{3}\frac{2}{3}\log\frac{\frac{12}{33}}{\frac{12}{33}} + \frac{2}{3}\frac{2}{3}\log\frac{\frac{22}{33}}{\frac{22}{33}} = 0 .$$

Exercise

- Mutual Information과 Conditional Entropy의 관계에 의하여 $H(X|Y)$을 구하시오.

	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60-n)$명

Exercise

- Mutual Information과 Conditional Entropy의 관계에 의하여 $H(X|Y)$을 구하시오.

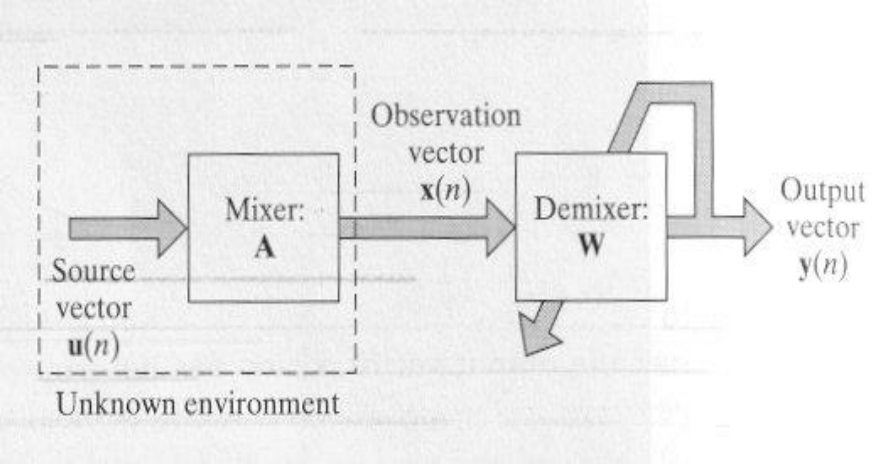
$X \quad \backslash \quad Y$	치킨집	삼겹살집
토트넘 응원자	10	n명
아스널 응원자	20	$(60-n)$명

$$\mathrm{I}(X,Y) = \mathrm{H}(X) - \mathrm{H}(X|Y) = 0$$

$$\mathrm{H}(\mathrm{X}|Y) = \mathrm{H}(X) = -\sum_x p(x) \log\, p(x) = -\frac{1}{3}\log\frac{1}{3} - \frac{2}{3}\log\frac{2}{3}$$

ICA(Independent Component Analysis)

- Blind source separation problem:
  Given N independent realizations of the observation vector X , find an estimate of the inverse of the mixing matrix A



- Algorithm of ICA:

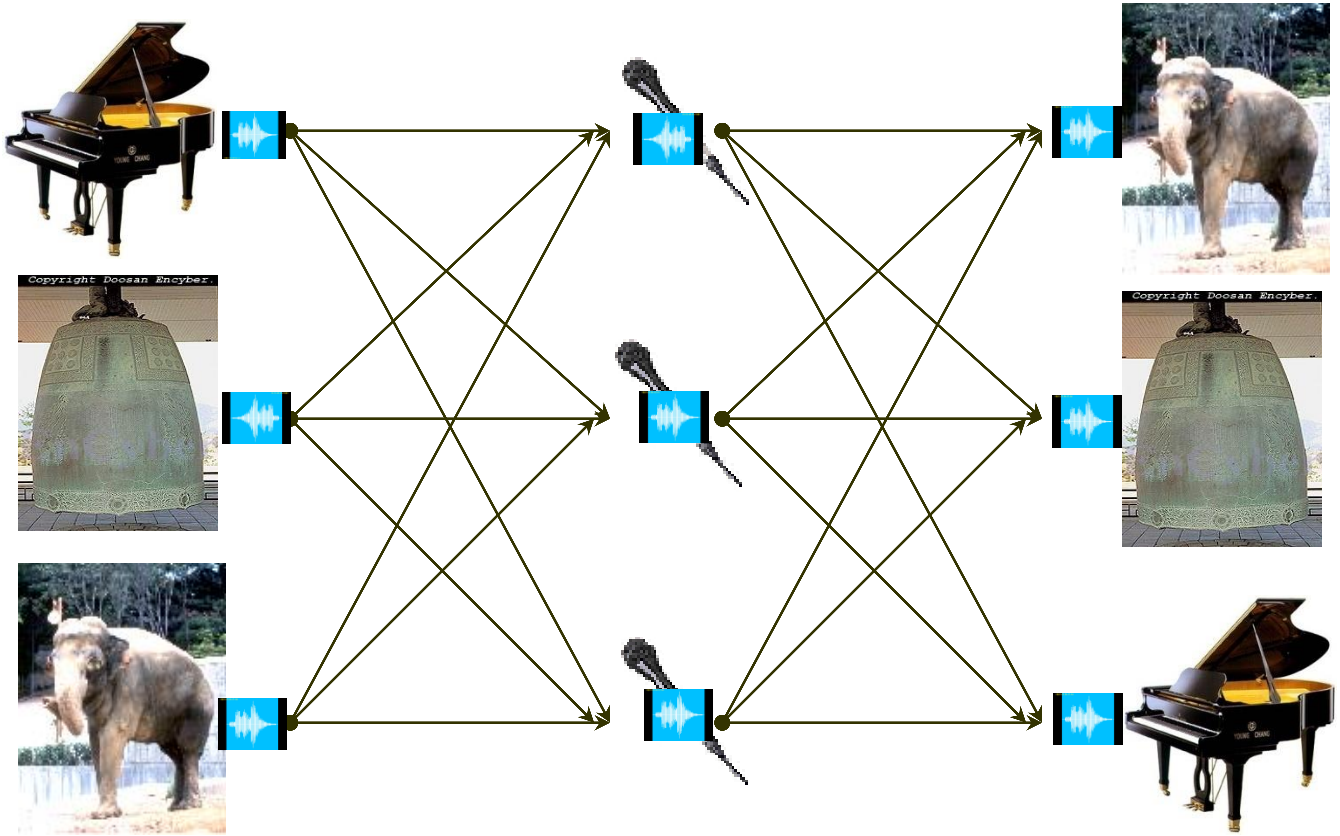
  → "as statistically independent as possible"

  → minimizing the mutual information between the each components

  of the output vector .

ICA(Independent Component Analysis)

- **ICA** Example



ICA(Independent Component Analysis)

- blind source separation problem

$U = [u_1, u_2, \ldots, u_m]^T$: Independent Sources

$X = AU, \quad A$: Mixing Matrix

$X = [x_1, x_2, \ldots, x_m]^T$: Observations

$Y = WX, : \quad W$: Demixing Matrix

U, X, Y : Zero mean Signals

$\rightarrow \quad Y = WX = WAU = DPU,$

where D: Diagonal matrix, P: Permutation matrix

$\rightarrow$ How to find W?

ICA(Independent Component Analysis)

- ICA : statistical independence
- Applications
  - Speech separation : teleconference
  - Array antenna processing
  - Multisensor biomedical records

    (태아의 심장박동을 어머니 심장박동과 분리)
  - Financial market data (Dominant data 추출)
  - Feature Extraction

ICA(Independent Component Analysis)

- Criterion for Statistical Independence

  Goal : Y_i, Y_j 간 mutual information을 최소화

  $$\min I\,(Y_i\,;Y_j) \qquad i,j = 1,\cdots,m$$

  $$I(Y_1, Y_2, \cdots, Y_m) = D_{f_Y \| \tilde{f}_Y} = \int_{-\infty}^{\infty} f_Y(y) \log\left(\frac{f_Y(y)}{\prod_{i=1}^{m} \tilde{f}_{Y_i}(y_i)}\right) dY$$

  $\tilde{f}_Y(y) = \prod_{i=1}^{m} \tilde{f}_{Y_i}(y_i),\ \tilde{f}_{Y_i}(y_i\,)$: Marginal p.d.f

- Learning Rule for ICA

  $$\Delta w_{ik} = -\eta \frac{\partial}{\partial w_{ik}} D_{f \| \tilde{f}}$$

ICA(Independent Component Analysis)

- Kullback-Leibler Divergence

$$D_{f_{\mathrm{Y}}\|\tilde{f}_{\mathrm{Y}}} = \int_{-\infty}^{\infty} f_{\mathrm{Y}}(\mathrm{y}) \log\left(\frac{f_{\mathrm{Y}}(\mathrm{y})}{\prod_{i=1}^{m} \tilde{f}_{\mathrm{Y}_i}(\mathrm{y}_i)}\right) dy$$

$$D_{f_{\mathrm{Y}}\|\tilde{f}_{\mathrm{Y}}} = \int_{-\infty}^{\infty} f_{\mathrm{Y}}(\mathrm{y}) \log f_{\mathrm{Y}}(\mathrm{y})\, dy - \sum_{i=1}^{m} \int_{-\infty}^{\infty} f_{Y}(\mathrm{y}) \log \tilde{f}_{\mathrm{Y_i}}(\mathrm{y_i})\, dy$$

- The second term is

$$\int_{-\infty}^{\infty} \log \tilde{f}_{\mathrm{Y}_i}(\mathrm{y}_i)\ \left[\int_{-\infty}^{\infty} f_{\mathrm{Y}}(\mathrm{y})\, dy^{(\mathrm{i})}\right] dy_i = \int_{-\infty}^{\infty} \tilde{f}_{\mathrm{Y}_i}(\mathrm{y}_i) \log \tilde{f}_{\mathrm{Y}_i}(\mathrm{y}_i)\, dy_i$$

$$= -\tilde{h}(\mathrm{Y}_i) \quad \text{:marginal entropy}$$

- Kullback-Leibler Divergence

$$D_{f_{\mathrm{Y}}\|\tilde{f}_{\mathrm{Y}}} = -h(\mathrm{Y}) + \sum_{i=1}^{m} \tilde{h}(\mathrm{Y}_i)$$

ICA(Independent Component Analysis)

- Entropy $h(Y)$

$$h(Y) = h(\mathrm{W}X) = h(X) + \log\left|\det(\mathrm{W})\right|,$$

$$(f_Y(y) = |\det(\mathrm{W})|^{-1} f_X(x),\ \ dy = |\det(\mathrm{W})|\, dx)$$

- Marginal entropy $h(Yi)$

Pdf of Y_i is obtained using truncate of Gram-Charlier series

$$\tilde{f}_{Y_i}(y_i(\mathrm{W})) = \alpha(y_i)[1 + \textstyle\sum_{k=3}^{\infty} c_{ik}\, H_k(y_i)]$$

where

$\alpha(y_i) = 1/\sqrt{2\pi}\,\exp(-yi^2)$

$H_k(y_i)$: Hermite polynomials

Cumulants $\{c_{ik} : k = 3, 4, \ldots,\}$ is obtained from k-th order moment of Y_i

Hermite polynomials: $H_3(y) = y^3 - 3x, H_4(y) = y^4 - 6y^2 + 3, \ldots$

ICA(Independent Component Analysis)

- $\tilde{f}_{Y_i}(y_i(\mathrm{W})) = \alpha(y_i)[1 + \sum_{k=3}^{\infty} c_{ik}\, H_k(y_i)]$

- The index grouping is done as $k = (0), (3), (4,6), (5,7,9), \ldots$
- By choosing by $k = (4,6)$

$$\tilde{f}_{Y_i}(y_i) = \alpha(y_i)\left(1 + \frac{k_{i,3}}{3!}H_3(y_i) + \frac{k_{i,4}^2}{4!}H_4(y_i) + \frac{(k_{i,6} + 10k_{i,3}^2)}{6!}H_6(y_i)\right)$$

- c_{ik} and k-th order moment of Y_i

$$k_{i,3} = m_{i,3},\ k_{i,4} = m_{i,4} - 3m_{i,2}^2$$

$$k_{i,6} = m_{i,6} - 10m_{i,3}^2 - 15m_{i,2}m_{i,4} + 30m_{i,2}^3$$

$$m_{i,k} = E\left[Y_i^k\right] = E\left[\left(\sum_{j=1}^{m} \boxed{w_{ij}} X_j\right)^k\right]$$

ICA(Independent Component Analysis)

- The cumulants are functions of W.
- Gradient of K-L divergence

$$1)\ \frac{\partial}{\partial w_{ij}}\log(\det(\mathrm{W})) = \frac{1}{\det(\mathrm{W})}\frac{\partial}{\partial w_{ij}}\det(\mathrm{W})$$

$$= \frac{A_{ij}}{\det(\mathrm{W})} \quad = (\mathrm{W}^{-T})_{ij}$$

$$2)\ \frac{\partial \kappa_{i,3}}{\partial w_{ij}} \approx 3{y_i}^2 x_j, \qquad \frac{\partial \kappa_{i,4}}{\partial w_{ij}} \approx -8{y_i}^3 x_j \ \ \ldots\ldots$$

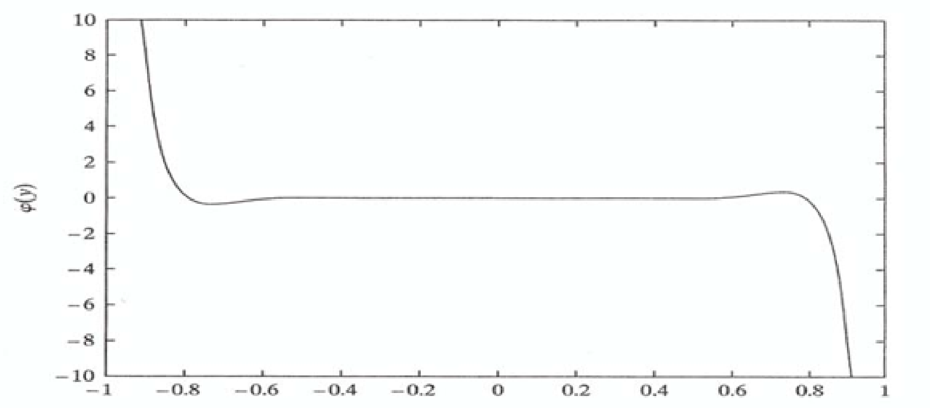
ICA(Independent Component Analysis)

- Minimization of Kullback-Leibler Divergence

$$D_{f_Y \| \tilde{f}_Y} = -h(Y) + \sum_{i=1}^{m} \tilde{h}(Y_i)$$

$$\frac{\partial}{\partial w_{ij}} D_{f \| \tilde{f}}(W) \approx -(W^{-T})_{ij} + \varphi(y_i)x_j$$

$$\varphi(y_i) = \frac{1}{2}y_i^5 + \frac{2}{3}y_i^7 + \frac{15}{2}y_i^9 + \frac{2}{15}y_i^{11} - \frac{112}{3}y_i^{13} + 128y_i^{15} - \frac{512}{3}y_i^{17}$$



ICA(Independent Component Analysis)

- Learning algorithm for ICA

$$\Delta w_{ij} = -\eta \frac{\partial}{\partial w_{ij}} D_{f\|\tilde{f}}$$

$$= \eta\left((\mathrm{W}^{-T})_{ij} - \phi(y_i)x_j\right)$$

$$\Delta \mathrm{W} = \eta(\mathrm{W}^{-T} - \phi(y)\mathrm{x}^T)$$

$$\Delta \mathrm{W} = \eta[\mathrm{I} - \phi(y)\mathrm{x}^T\mathrm{W}^T]\mathrm{W}^{-T}$$

$$= \eta[\mathrm{I} - \phi(y)\mathrm{y}^T]\mathrm{W}^{-T}$$

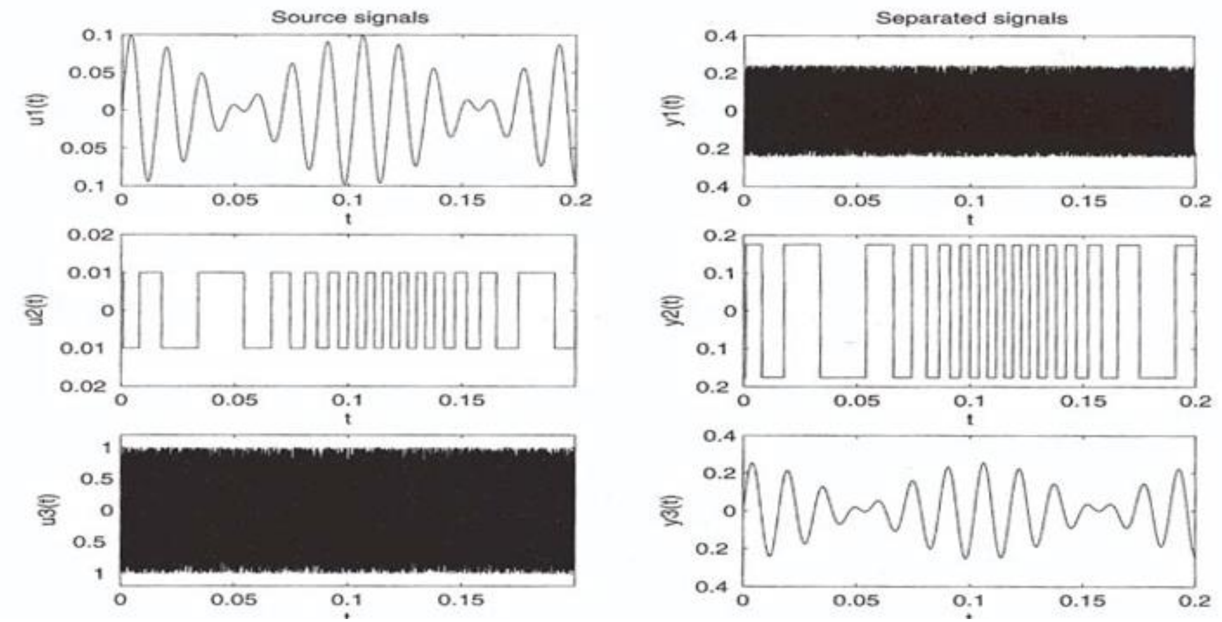
$$\mathrm{W}(n+1) = \mathrm{W}(n) + \eta(n)\big[\mathrm{I} - \phi\big(y(n)\big)y^T(n)\big]\mathrm{W}^{-T}(n)$$

ICA(Independent Component Analysis)

- Experiments

$$u_1(n) = 0.1\sin(400n)\cos(30n)$$
$$u_2(n) = 0.01\, sgn(\sin(500n + 9\cos(40n))$$
$$u_3(n) = noise\ uniformly\ distributed\ in\ [-1, 1]$$

$$A = \begin{bmatrix} 0.56 & 0.79 & -0.37 \\ -0.75 & 0.65 & 0.86 \\ 0.17 & 0.32 & -0.48 \end{bmatrix}$$



Exercise

- In computer science(CS) department, the probability of dropping the machine learning(ML) course in March is 1/6, that in April is 1/3, and the probability of taking ML course to the end without dropping is 1/2, whereas those in Electrical engineering(EE) department are 1/8, 1/8, and 3/4, respectively. Meanwhile, the portions of CS & EE students in ML course are 1/5 & 4/5, respectively. Letting X be the random variable on dropping or not of a student, and Y be the random variable on the department of a student, find the followings.

  1. Conditional entropy $H(X|Y)$.
  2. Mutual information $I(X;Y)$.

Exercise

- In computer science(CS) department, the probability of dropping the machine learning(ML) course in March is 1/6, that in April is 1/3, and the probability of taking ML course to the end without dropping is 1/2, whereas those in Electrical engineering(EE) department are 1/8, 1/8, and 3/4, respectively. Meanwhile, the portions of CS & EE students in ML course are 1/5 & 4/5, respectively. Letting X be the random variable on dropping or not of a student, and Y be the random variable on the department of a student, find $H(X|Y)$, $I(X;Y)$.
- 서술식을 수식으로 변경:

Exercise

- In computer science(CS) department, the probability of dropping the machine learning(ML) course in March is 1/6, that in April is 1/3, and the probability of taking ML course to the end without dropping is 1/2, whereas those in Electrical engineering(EE) department are 1/8, 1/8, and 3/4, respectively. Meanwhile, the portions of CS & EE students in ML course are 1/5 & 4/5, respectively. Letting X be the random variable on dropping or not of a student, and Y be the random variable on the department of a student, find $H(X|Y)$, $I(X;Y)$.
- 서술식을 수식으로 변경:

- X: random variable on dropping or not of a student
- Y: random variable on the department of a student
- X=0: Mar. drop, X=1: Apr. drop, X=2: No drop
- Y=0: CS, Y=1: EE
- $P(X=0|Y=0)=1/6, P(X=1|Y=0)=1/3, P(X=2|Y=0)=1/2$
- $P(X=0|Y=1)=1/8, P(X=1|Y=1)=1/8, P(X=2|Y=1)=3/4$
- $P(Y=0)=1/5,\ P(Y=1)=4/5$
- $H(X|Y)=?,\ I(X;Y)=?.$

Exercise

- X: random variable on dropping or not of a student
- Y: random variable on the department of a student
- X=0: Mar. drop, X=1: Apr. drop, X=2: No drop
- Y=0: CS, Y=1: EE
- $P(X = 0|Y = 0) = 1/6, P(X = 1|Y = 0) = 1/3, P(X = 2|Y = 0) = 1/2$
- $P(X = 0|Y = 1) = 1/8, P(X = 1|Y = 1) = 1/8, P(X = 2|Y = 1) = 3/4$
- $P(Y = 0) = 1/5,\ P(Y = 1) = 4/5$
- $H(X|Y) =?,\ I(X;Y) =?.$
- Sol. $H(X|Y) =?,\ I(X;Y) =?.$

Exercise

- X: random variable on dropping or not of a student
- Y: random variable on the department of a student
- X=0: Mar. drop, X=1: Apr. drop, X=2: No drop
- Y=0: CS, Y=1: EE
- $P(X=0|Y=0)=1/6, P(X=1|Y=0)=1/3, P(X=2|Y=0)=1/2$
- $P(X=0|Y=1)=1/8, P(X=1|Y=1)=1/8, P(X=2|Y=1)=3/4$
- $P(Y=0)=1/5,\ P(Y=1)=4/5$
- $H(X|Y)=?,\ I(X;Y)=?.$

- Sol. $H(X|Y)=?,\ I(X;Y)=?.$

$$H(X|Y)=H(X,Y)-H(Y).$$

Exercise

- X: random variable on dropping or not of a student
- Y: random variable on the department of a student
- X=0: Mar. drop, X=1: Apr. drop, X=2: No drop
- Y=0: CS, Y=1: EE
- $P(X=0|Y=0)=1/6, P(X=1|Y=0)=1/3, P(X=2|Y=0)=1/2$
- $P(X=0|Y=1)=1/8, P(X=1|Y=1)=1/8, P(X=2|Y=1)=3/4$
- $P(Y=0)=1/5,\ P(Y=1)=4/5$
- $H(X|Y)=?,\ I(X;Y)=?.$

- Sol. $H(X|Y)=?,\ I(X;Y)=?.$

$$H(X|Y)=H(X,Y)-H(Y).$$

$$H(Y)=-\Sigma_{y\in Y}p(y)logp(y)=-\frac{1}{5}\log\frac{1}{5}-\frac{4}{5}\log\frac{4}{5}=0.7219$$

Exercise

- X: random variable on dropping or not of a student
- Y: random variable on the department of a student
- X=0: Mar. drop, X=1: Apr. drop, X=2: No drop
- Y=0: CS, Y=1: EE
- $P(X = 0|Y = 0) = 1/6, P(X = 1|Y = 0) = 1/3, P(X = 2|Y = 0) = 1/2$
- $P(X = 0|Y = 1) = 1/8, P(X = 1|Y = 1) = 1/8, P(X = 2|Y = 1) = 3/4$
- $P(Y = 0) = 1/5,\ P(Y = 1) = 4/5$
- $H(X|Y) =?,\ I(X;Y) =?.$

- Sol. $H(X|Y) =?,\ I(X;Y) =?.$

$$H(X|Y) = H(X,Y) - H(Y).$$

$$H(Y) = -\Sigma_{y\in Y} p(y) log p(y) = -\frac{1}{5}\log\frac{1}{5} - \frac{4}{5}\log\frac{4}{5} = 0.7219$$

$$H(X,Y) = \ -\Sigma_{x\in X}\Sigma_{y\in Y} p(x,y) log p(x,y)$$

Exercise

- X: random variable on dropping or not of a student
- Y: random variable on the department of a student
- X=0: Mar. drop, X=1: Apr. drop, X=2: No drop
- Y=0: CS, Y=1: EE
- $P(X = 0|Y = 0) = 1/6, P(X = 1|Y = 0) = 1/3, P(X = 2|Y = 0) = 1/2$
- $P(X = 0|Y = 1) = 1/8, P(X = 1|Y = 1) = 1/8, P(X = 2|Y = 1) = 3/4$
- $P(Y = 0) = 1/5,\ P(Y = 1) = 4/5$
- $H(X|Y)$ =?, $I(X;Y)$ =?.

- Sol. $H(X|Y)$ =?, $I(X;Y)$ =?.

$$H(X|Y) = H(X,Y) - H(Y).$$

$$H(Y) = -\Sigma_{y\in Y} p(y) logp(y) = -\frac{1}{5}\log\frac{1}{5} - \frac{4}{5}\log\frac{4}{5} = 0.7219$$

$$H(X,Y) = \ -\Sigma_{x\in X}\Sigma_{y\in Y} p(x,y) logp(x,y)$$

$$H(X,Y) = \ -\Sigma_{x\in X}\Sigma_{y\in Y} p(x|y) \times p(y) logp(x|y) \times p(y)$$

Exercise

- X: random variable on dropping or not of a student
- Y: random variable on the department of a student
- X=0: Mar. drop, X=1: Apr. drop, X=2: No drop
- Y=0: CS, Y=1: EE
- $P(X=0|Y=0)=1/6, P(X=1|Y=0)=1/3, P(X=2|Y=0)=1/2$
- $P(X=0|Y=1)=1/8, P(X=1|Y=1)=1/8, P(X=2|Y=1)=3/4$
- $P(Y=0)=1/5,\ P(Y=1)=4/5$
- $H(X|Y)=?,\ I(X;Y)=?.$
- Sol. $H(X|Y)=?,\ I(X;Y)=?.$

$H(X|Y)=H(X,Y)-H(Y).$

$$H(Y)=-\Sigma_{y\in Y}p(y)logp(y)=-\frac{1}{5}\log\frac{1}{5}-\frac{4}{5}\log\frac{4}{5}=0.7219$$

$$H(X,Y)=\ -\Sigma_{x\in X}\Sigma_{y\in Y}p(x,y)logp(x,y)$$

$$H(X,Y)=\ -\Sigma_{x\in X}\Sigma_{y\in Y}p(x|y)\times p(y)logp(x|y)\times p(y)$$

$$=-\frac{1}{6}*\frac{1}{5}\log\left(\frac{1}{6}*\frac{1}{5}\right)-\frac{1}{3}*\frac{1}{5}\log\left(\frac{1}{3}*\frac{1}{5}\right)-\frac{1}{2}*\frac{1}{5}\log\left(\frac{1}{2}*\frac{1}{5}\right)$$

$$-\frac{1}{8}*\frac{4}{5}\log\left(\frac{1}{8}*\frac{4}{5}\right)-\frac{1}{8}*\frac{4}{5}\log\left(\frac{1}{8}*\frac{4}{5}\right)-\frac{3}{4}*\frac{4}{5}\log\left(\frac{3}{4}*\frac{4}{5}\right)\ =\ 1.8628$$

Exercise

- X: random variable on dropping or not of a student
- Y: random variable on the department of a student
- X=0: Mar. drop, X=1: Apr. drop, X=2: No drop
- Y=0: CS, Y=1: EE
- $P(X=0|Y=0)=1/6, P(X=1|Y=0)=1/3, P(X=2|Y=0)=1/2$
- $P(X=0|Y=1)=1/8, P(X=1|Y=1)=1/8, P(X=2|Y=1)=3/4$
- $P(Y=0)=1/5,\ P(Y=1)=4/5$
- $H(X|Y)=?,\ I(X;Y)=?.$
- Sol. $H(X|Y)=?,\ I(X;Y)=?.$

$$H(X|Y)=H(X,Y)-H(Y).$$

$$H(Y)=-\Sigma_{y\in Y}p(y)logp(y)=-\frac{1}{5}\log\frac{1}{5}-\frac{4}{5}\log\frac{4}{5}=0.7219$$

$$H(X,Y)=\ -\Sigma_{x\in X}\Sigma_{y\in Y}p(x,y)logp(x,y)$$

$$H(X,Y)=\ -\Sigma_{x\in X}\Sigma_{y\in Y}p(x|y)\times p(y)logp(x|y)\times p(y)$$

$$=-\frac{1}{6}*\frac{1}{5}\log\left(\frac{1}{6}*\frac{1}{5}\right)-\frac{1}{3}*\frac{1}{5}\log\left(\frac{1}{3}*\frac{1}{5}\right)-\frac{1}{2}*\frac{1}{5}\log\left(\frac{1}{2}*\frac{1}{5}\right)$$

$$-\frac{1}{8}*\frac{4}{5}\log\left(\frac{1}{8}*\frac{4}{5}\right)-\frac{1}{8}*\frac{4}{5}\log\left(\frac{1}{8}*\frac{4}{5}\right)-\frac{3}{4}*\frac{4}{5}\log\left(\frac{3}{4}*\frac{4}{5}\right)\ =\ 1.8628$$

$$\boxed{\begin{aligned}H(X|Y)&=1.8628-0.7219\\&=\ 1.1409\end{aligned}}$$

Exercise

- X: random variable on dropping or not of a student
- Y: random variable on the department of a student
- X=0: Mar. drop, X=1: Apr. drop, X=2: No drop
- Y=0: CS, Y=1: EE
- $P(X = 0|Y = 0) = 1/6, P(X = 1|Y = 0) = 1/3, P(X = 2|Y = 0) = 1/2$
- $P(X = 0|Y = 1) = 1/8, P(X = 1|Y = 1) = 1/8, P(X = 2|Y = 1) = 3/4$
- $P(Y = 0) = 1/5,\ P(Y = 1) = 4/5$
- Sol. $H(X|Y) =?,\ I(X;Y) =?.$

  $I(X;Y) = H(X) + H(Y) - H(X,Y) =?$

Exercise

- X: random variable on dropping or not of a student
- Y: random variable on the department of a student
- X=0: Mar. drop, X=1: Apr. drop, X=2: No drop
- Y=0: CS, Y=1: EE
- $P(X=0|Y=0)=1/6, P(X=1|Y=0)=1/3, P(X=2|Y=0)=1/2$
- $P(X=0|Y=1)=1/8, P(X=1|Y=1)=1/8, P(X=2|Y=1)=3/4$
- $P(Y=0)=1/5,\ P(Y=1)=4/5$
- Sol. $H(X|Y)$ =?, $I(X;Y)$ =?.

  $I(X;Y)=H(X)+H(Y)-\ H(X,Y)$ =?

  $H(X,Y)=1.8628, H(Y)=0.7219$

Exercise

- X: random variable on dropping or not of a student
- Y: random variable on the department of a student
- X=0: Mar. drop, X=1: Apr. drop, X=2: No drop
- Y=0: CS, Y=1: EE
- $P(X = 0|Y = 0) = 1/6, P(X = 1|Y = 0) = 1/3, P(X = 2|Y = 0) = 1/2$
- $P(X = 0|Y = 1) = 1/8, P(X = 1|Y = 1) = 1/8, P(X = 2|Y = 1) = 3/4$
- $P(Y = 0) = 1/5,\ P(Y = 1) = 4/5$
- Sol. $H(X|Y) =?,\ I(X;Y) =?.$

  $I(X;Y) = H(X) + H(Y) - H(X,Y) =?$

  $H(X,Y) = 1.8628, H(Y) = 0.7219$

  $H(X) = -\Sigma_{x \in X} p(x) log p(x)$

Exercise

- X: random variable on dropping or not of a student
- Y: random variable on the department of a student
- X=0: Mar. drop, X=1: Apr. drop, X=2: No drop
- Y=0: CS, Y=1: EE
- $P(X = 0|Y = 0) = 1/6, P(X = 1|Y = 0) = 1/3, P(X = 2|Y = 0) = 1/2$
- $P(X = 0|Y = 1) = 1/8, P(X = 1|Y = 1) = 1/8, P(X = 2|Y = 1) = 3/4$
- $P(Y = 0) = 1/5,\ P(Y = 1) = 4/5$
- Sol. $H(X|Y) =?,\ I(X;Y) =?.$

  $I(X;Y) = H(X) + H(Y) - H(X,Y) =?$

  $H(X,Y) = 1.8628, H(Y) = 0.7219$

  $H(X) = -\Sigma_{x \in X} p(x) log p(x)$

  By total probability,

  $P(X = x) = \Sigma_{y \in Y} P(X = x|Y = y) P(Y = y)$

Exercise

- X: random variable on dropping or not of a student
- Y: random variable on the department of a student
- X=0: Mar. drop, X=1: Apr. drop, X=2: No drop
- Y=0: CS, Y=1: EE
- $P(X = 0|Y = 0) = 1/6, P(X = 1|Y = 0) = 1/3, P(X = 2|Y = 0) = 1/2$
- $P(X = 0|Y = 1) = 1/8, P(X = 1|Y = 1) = 1/8, P(X = 2|Y = 1) = 3/4$
- $P(Y = 0) = 1/5,\ P(Y = 1) = 4/5$
- Sol. $H(X|Y) =?,\ I(X;Y) =?.$

$I(X;Y) = H(X) + H(Y) - H(X,Y) =?$

$H(X,Y) = 1.8628, H(Y) = 0.7219$

$H(X) = -\Sigma_{x\in X} p(x) log p(x)$

By total probability,

$P(X = x) = \Sigma_{y\in Y} P(X = x|Y = y)P(Y = y)$

$P(X = 0) = \frac{1}{6} * \frac{1}{5} + \frac{1}{8} * \frac{4}{5} = \frac{2}{15}, P(X = 1) = \frac{1}{3} * \frac{1}{5} + \frac{1}{8} * \frac{4}{5} = \frac{1}{6}, P(X = 3) = \frac{1}{2} * \frac{1}{5} + \frac{3}{4} * \frac{4}{5} = \frac{7}{10}$

Exercise

- X: random variable on dropping or not of a student
- Y: random variable on the department of a student
- X=0: Mar. drop, X=1: Apr. drop, X=2: No drop
- Y=0: CS, Y=1: EE
- $P(X=0|Y=0)=1/6, P(X=1|Y=0)=1/3, P(X=2|Y=0)=1/2$
- $P(X=0|Y=1)=1/8, P(X=1|Y=1)=1/8, P(X=2|Y=1)=3/4$
- $P(Y=0)=1/5,\ P(Y=1)=4/5$
- Sol. $H(X|Y)=?,\ I(X;Y)=?.$

$I(X;Y)=H(X)+H(Y)-\ H(X,Y)=?$

$H(X,Y)=1.8628, H(Y)=0.7219$

$H(X)=-\Sigma_{x\in X}p(x)logp(x)$

By total probability,

$P(X=x)=\Sigma_{y\in Y}P(X=x|Y=y)P(Y=y)$

$P(X=0)=\frac{1}{6}*\frac{1}{5}+\frac{1}{8}*\frac{4}{5}=\frac{2}{15}, P(X=1)=\frac{1}{3}*\frac{1}{5}+\frac{1}{8}*\frac{4}{5}=\frac{1}{6}, P(X=3)=\frac{1}{2}*\frac{1}{5}+\frac{3}{4}*\frac{4}{5}=\frac{7}{10}$

$H(X)=\ -\left[\frac{2}{15}\log\left(\frac{2}{15}\right)+\frac{1}{6}\log\left(\frac{1}{6}\right)+\frac{7}{10}\log\left(\frac{7}{10}\right)+\right]=1.1786$

Exercise

- X: random variable on dropping or not of a student
- Y: random variable on the department of a student
- X=0: Mar. drop, X=1: Apr. drop, X=2: No drop
- Y=0: CS, Y=1: EE
- $P(X=0|Y=0)=1/6, P(X=1|Y=0)=1/3, P(X=2|Y=0)=1/2$
- $P(X=0|Y=1)=1/8, P(X=1|Y=1)=1/8, P(X=2|Y=1)=3/4$
- $P(Y=0)=1/5,\ P(Y=1)=4/5$
- Sol. $H(X|Y)=?,\ I(X;Y)=?.$

$I(X;Y)=H(X)+H(Y)-\ H(X,Y)=?$

$H(X,Y)=1.8628, H(Y)=0.7219$

$H(X)=-\Sigma_{x\in X}p(x)logp(x)$

$$\boxed{\begin{aligned} I(X,Y) &= 1.1786+0.7219-1.8628 \\ &= 0.038 \end{aligned}}$$

By total probability,

$P(X=x)=\Sigma_{y\in Y}P(X=x|Y=y)P(Y=y)$

$P(X=0)=\frac{1}{6}*\frac{1}{5}+\frac{1}{8}*\frac{4}{5}=\frac{2}{15}, P(X=1)=\frac{1}{3}*\frac{1}{5}+\frac{1}{8}*\frac{4}{5}=\frac{1}{6}, P(X=3)=\frac{1}{2}*\frac{1}{5}+\frac{3}{4}*\frac{4}{5}=\frac{7}{10}$

$H(X)=\ -\left[\frac{2}{15}\log\left(\frac{2}{15}\right)+\frac{1}{6}\log\left(\frac{1}{6}\right)+\frac{7}{10}\log\left(\frac{7}{10}\right)+\right]=1.1786$

Summary

- Information
- Entropy
- Cross Entropy
- Error Backpropagation Learning
- Mutual Information
- Kullback Leibler Divergence
- Independent Component Analysis (ICA)
- Learning for ICA
- Blind Source Separation

Reference: Simon Haykin, *Neural Networks: A Comprehensive Foundation,* Prentice Hall

Optimization (I)

Jin Young Choi
Seoul National University

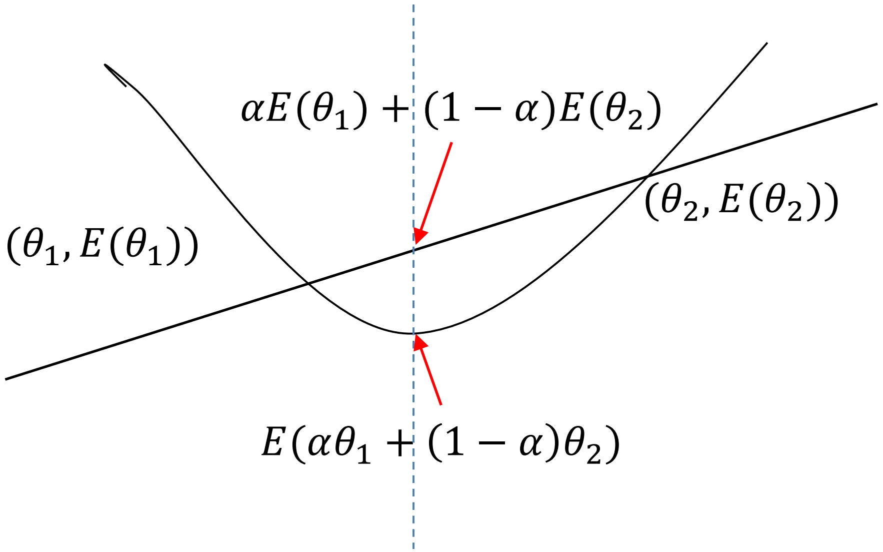
Outline

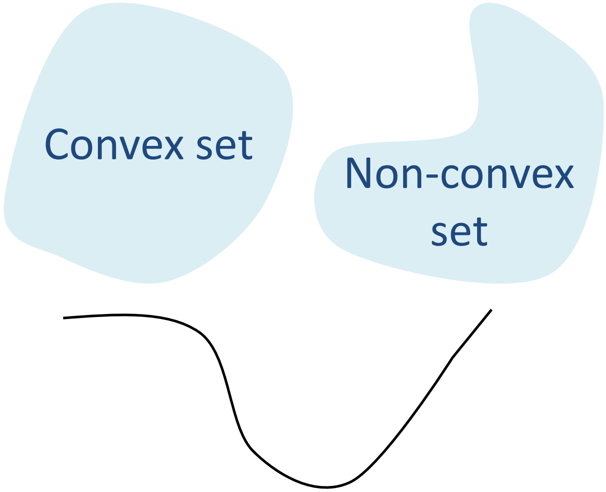
- Constraint Convex Optimization
- linear/quadratic programming
- dual problem, KKT conditions
- Minimization techniques
  - Gradient Descent Minimization
  - Newton Minimization
  - Gauss Newton Minimization
  - (In)Equality Constraint Minimization

Convex Optimization

- Definition: $E : R^n \rightarrow R$ is convex function if $\mathbf{dom}\ \ E$ is a convex set and $\alpha\theta_1 + (1-\alpha)\theta_2 \in dom\ E$
- $E(\alpha\theta_1 + (1-\alpha)\theta_2) \leq \alpha E(\theta_1) + (1-\alpha)E(\theta_2)$,

  where $\theta_1, \theta_2 \in \mathbf{dom}\ \ E, \quad 0 \leq \alpha \leq 1$





Non-convex function

Convex Function Conditions

2nd-order conditions: for twice differentiable f with convex domain

f is convex if and only if

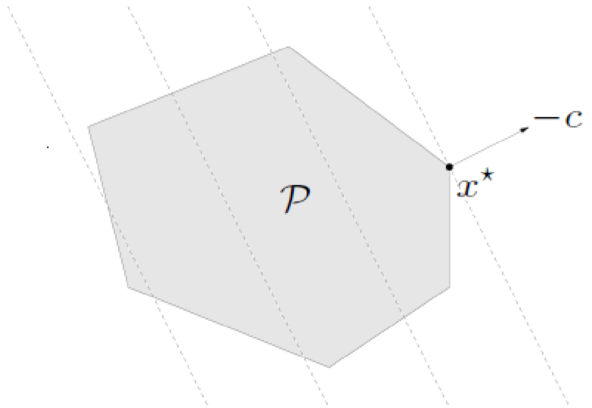
$$\nabla^2 f(x) \succeq 0 \quad \text{for all } x \in \textbf{dom}\, f$$

If $\nabla^2 f(x) \succ 0$ for all $x \in \textbf{dom}\, f$, then f is strictly convex

Linear program (LP)

- Formulation

$$\begin{aligned} &\text{minimize} && c^T x + d \\ &\text{subject to} && Gx \leq h \\ & && Ax = b \end{aligned}$$



- convex problem with affine objective and constraint functions
- feasible set is a polyhedron

Linear program (LP)

- Example: norm minimization problem

$$\text{minimize } \|x\|_1$$

- equivalent to an LP

Linear program (LP)

- Example: norm minimization problem

$$\text{minimize } \|x\|_1$$

- equivalent to an LP

$$\text{minimize} \sum |x_i|$$

$$\begin{aligned} &\text{minimize} && \sum s_i = 1^T s \\ &\text{subject to} && |x|_i \le s_i,\ \ i = 1, \ldots n \end{aligned}$$

$$\begin{aligned} &\text{minimize} && \sum s_i = 1^T s \\ &\text{subject to} && -s_i \le x_i \le s_i,\ \ i = 1, \ldots n \end{aligned}$$

Quadratic program (QP)

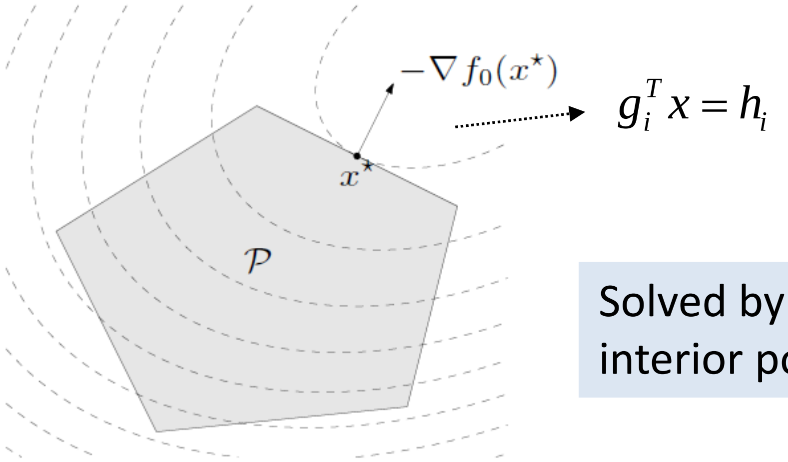
- Formulation

  minimize $\quad (1/2)x^T P x + q^T x + r$

  subject to $\quad Gx \leq h$

  $\qquad\qquad\quad Ax = b$

- $P \in \mathbf{S}_+^n$, so objective is convex quadratic minimize a convex quadratic function over a polyhedron



Solved by KKT condition or interior point method

Constraint Convex Optimization

- Standard Convex Problem

  minimize $E_0(\theta)$

  subject to $g_i(\theta) \leq 0, \quad i = 1, \cdots, m$

  $h_i(\theta) \leq 0, \quad i = 1, \cdots, p$

- where $E_0(\theta), g_i(\theta), h_i(\theta)$ are convex.
- Lagrangian

$$L(\theta, \lambda, \nu) = E_0(\theta) + \sum_{i=1}^{m} \lambda_i g_i(\theta) + \sum_{i=1}^{p} \nu_i h_i(\theta) \qquad \lambda_i \geq 0, \lambda_i g_i(\theta) \leq 0$$

$$= E_0(\theta) + \lambda^T g(\theta) + \nu^T h(\theta)$$

$$[\lambda_1 \quad \dots \quad \lambda_m] \begin{bmatrix} g_1(\theta) \\ \dots \\ g_n(\theta) \end{bmatrix} + [\nu_1 \quad \dots \quad \nu_m] \begin{bmatrix} h_1(\theta) \\ \dots \\ h_n(\theta) \end{bmatrix}$$

Dual Problem

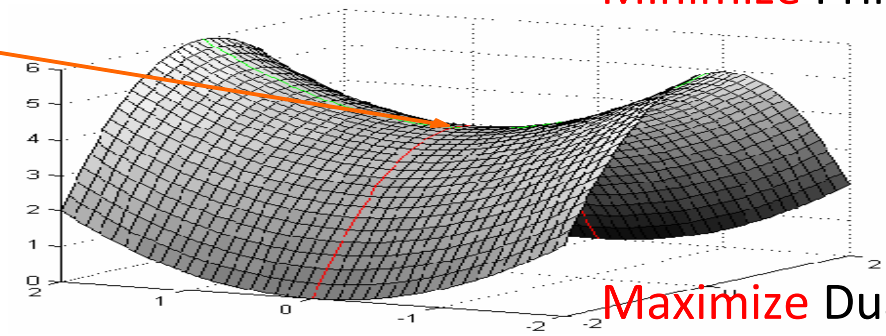
- Lagrange dual function:

$$l(\lambda,\nu) = \inf_{\theta\in D} L(\theta,\lambda,\nu)$$
$$= \inf_{\theta\in D}\left(E_0(\theta) + \sum_{i=1}^{m} \lambda_i g_i(\theta) + \sum_{i=1}^{p} \nu_i h_i(\theta) \right)$$

- $l(\lambda,\nu)$ is concave

Minimize Primal problem

Saddle point



Maximize Dual problem

Dual Problem

- **Lower bound property:**

  if $\lambda \geq 0$, then $\boxed{l(\lambda, \nu) \leq p^*}$

  where p^* is the optimal solution of the primal problem

- **Strong Duality**

  For the standard convex problem,

$$\boxed{\max_{\lambda,\nu} l(\lambda, \nu) = p^*}$$

$$\boxed{\begin{aligned} l(\lambda,\nu) &= \inf_{\theta \in D} L(\theta, \lambda, \nu) \\ &= \inf_{\theta \in D} \left(E_0(\theta) + \sum_{i=1}^{m} \lambda_i g_i(\theta) + \sum_{i=1}^{p} \nu_i h_i(\theta) \right) \end{aligned}}$$

Dual Problem

$$\begin{array}{ll} \text{minimize} & \theta^T\theta \\ \text{subject to} & A\theta = b \end{array}$$

dual function : $L(\theta, \nu) = \theta^T\theta + \nu^T(A\theta - b)$

Since quadratic, $\nabla_\theta L(\theta, \nu) = 0 = 2\theta + A^T\nu$

$$\theta = -\tfrac{1}{2}A^T\nu$$

$$g(\nu) = \inf_\theta L(\theta, \nu) = \frac{1}{4}\nu^T AA^T\nu - \frac{1}{2}\nu^T AA^T\nu - \nu^T b$$
$$= -\frac{1}{4}\nu^T AA^T\nu - \nu^T b$$

lower bound property: $p^* \geq -(1/4)\nu^T AA^T\nu - b^T\nu$ for all ν

$$\rightarrow \theta^* = -\tfrac{1}{2}A^T\nu^*$$

KKT Condition

- **Karush-Kuhn-Tucker(KKT) Conditions**

1. Primal constraints:

$$g_i(\theta) \le 0,\ \ i = 1,\ \ldots,\ m,\ \ h_i(\theta) = 0,\quad i = 1,\ \ldots,\ p$$

2. Dual constraints:

$$\lambda_i \ge 0,\ \ i = 1,\ \ldots,\ m$$

3. Complementary slackness:

$$\lambda_i g_i(\theta) = 0,\ \ i = 1,\ \ldots,\ m$$

4. Gradient of Lagrangian w.r.t. θ vanishes:

$$\nabla E_0(\theta) + \sum_{i=1}^{m} \lambda_i \nabla g_i(\theta) + \sum_{i=1}^{p} \nu_i \nabla h_i(\theta) = 0$$

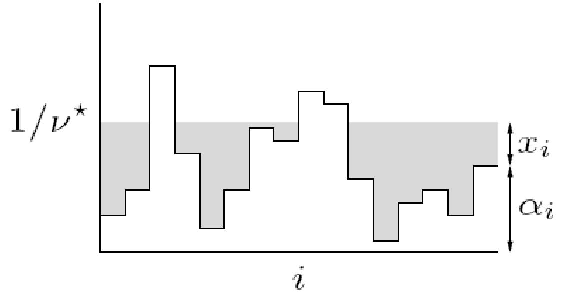
- If θ, λ, ν satisfy KKT for a convex problem, they are optimal.

$$L(\theta, \lambda, \nu) = E_0(\theta) + \sum_{i=1}^{m} \lambda_i g_i(\theta) + \sum_{i=1}^{p} \nu_i h_i(\theta)$$

KKT Condition

- **example: water-filling** (assume $\alpha_i > 0$)

$$\begin{array}{ll} \text{minimize} & -\sum_{i=1}^{n} \log(x_i + \alpha_i) \\ \text{subject to} & x \succeq 0, \quad \mathbf{1}^T x = 1 \end{array}$$



convex constraint & feasible

→ Strong duality holds

→ Solution of KKT condition is optimal

KKT Condition

- **example: water-filling** (assume $\alpha_i > 0$)

$$\begin{array}{ll} \text{minimize} & -\sum_{i=1}^{n} \log(x_i + \alpha_i) \\ \text{subject to} & x \succeq 0, \quad \mathbf{1}^T x = 1 \end{array}$$

$$L(x,\lambda,\nu) = -\sum_{i=1}^{n} \log(x_i + \alpha_i) - \sum \lambda_i x_i + \nu(1^T x - 1)$$

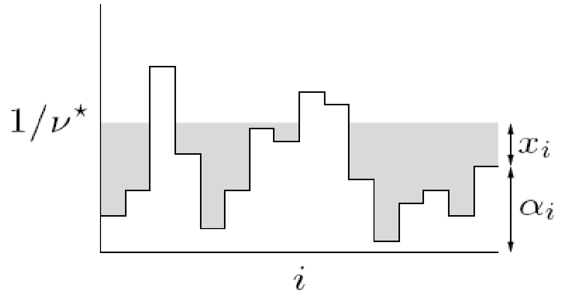
x is optimal iff there exist $x, \lambda \in \mathbf{R}^n, \nu \in \mathbf{R}$ satisfying KKT condition:

$$1.\ x \succeq 0,\quad \mathbf{1}^T x = 1 \quad 2.\ \lambda \succeq 0, \quad 3.\ \lambda_i x_i = 0, \quad 4.\ \frac{1}{x_i + \alpha_i} + \lambda_i = \nu$$

- if $\lambda_i = 0,\ x_i = 1/\nu - \alpha_i \geq 0 (\Rightarrow \nu \leq 1/\alpha_i)$
- if $x_i = 0,\ \lambda_i = \nu - 1/\alpha_i \geq 0 (\Rightarrow \nu \geq 1/\alpha_i)$
- determine ν from $\mathbf{1}^T x = \sum_{i=1}^{n} \max\{0, 1/\nu - \alpha_i\} = 1$

interpretation

- n patches; level of patch i is at height α_i
- flood area with unit amount of water
- resulting level is $1/\nu^*$

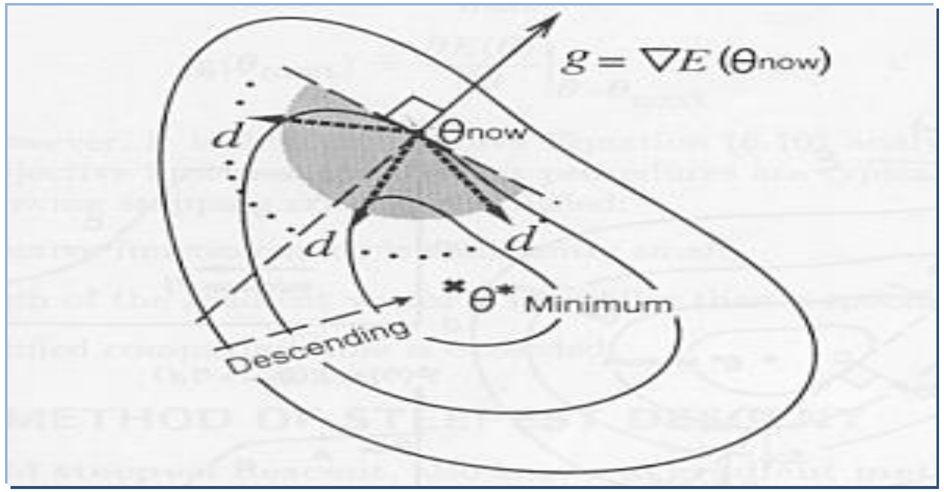


Gradient Descent Minimization

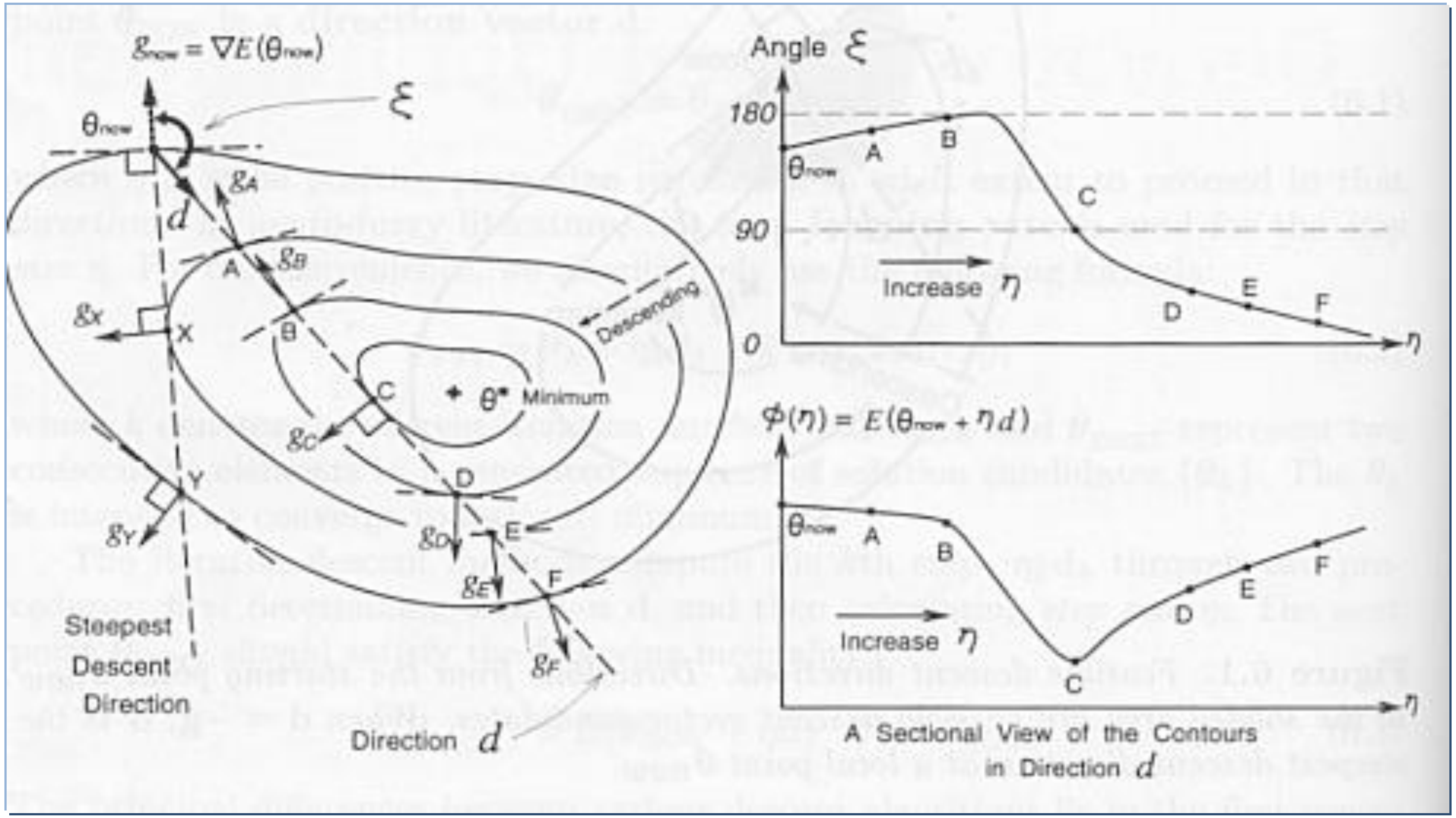
- Gradient Descent Update Rule (Steepest Descent for $G = I$)

$$\theta_{next} = \theta_{now} - \eta G \nabla E(\theta_{now}), \qquad G > 0$$

$$\nabla E(\theta) \overset{def}{=} \left[\frac{\partial E(\theta)}{\partial \theta_1}, \frac{\partial E(\theta)}{\partial \theta_2}, \ldots, \frac{\partial E(\theta)}{\partial \theta_n}\right]^t$$



Gradient Descent Minimization



Gradient Descent Minimization

- Optimal Learning Rate
  - Necessary Condition

$$\nabla^T E\left(\theta_{next}\right)\nabla E\left(\theta_{now}\right)=0,$$
$$\nabla^T E\left(\theta_{now}-\eta\nabla E\left(\theta_{now}\right)\right)\nabla E\left(\theta_{now}\right)=0$$

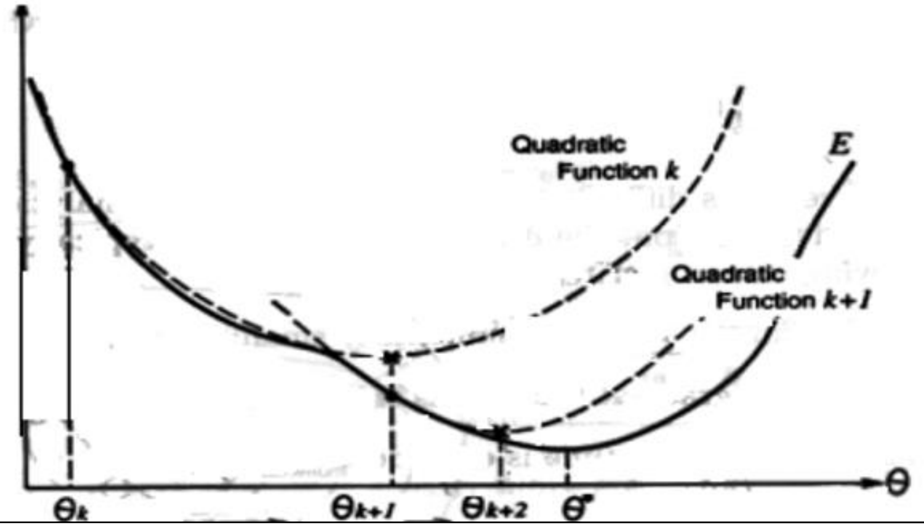
- Learning Rate Search Methods
  - Initial Bracketing
  - Line Searching
  - Secant Method (Approximate Newton Method)
  - Bisection Method
  - Golden section search method

Newton Minimization

- The objective function $f(\theta)$ can be approximated by a quadratic form:

$$E(\theta) = E(\theta_{now}) + \Lambda^T(\theta - \theta_{now}) + \frac{1}{2}(\theta - \theta_{now})^T \mathbf{H}(\theta - \theta_{now})$$

$$\text{where} \quad \mathbf{H} = \nabla^2 E(\theta_{now}), \quad \Lambda = \nabla E(\theta_{now}).$$



Newton Minimization

- Since the equation defines a quadratic function
  - its minimum can be determined by differentiating & setting to 0.

$$E(\theta) \cong E(\theta_{now}) + \Lambda^T(\theta - \theta_{now}) + \frac{1}{2}(\theta - \theta_{now})^T \mathbf{H}(\theta - \theta_{now})$$

$$\Lambda + \mathbf{H}(\theta_{next} - \theta_{now}) = 0$$

$$\theta_{next} = \theta_{now} - \mathbf{H}^{-1}\Lambda$$

$$\theta_{next} = \theta_{now} - \eta \boxed{G} \nabla E(\theta_{now}), \qquad G > 0$$

Gauss Newton Minimization

- Key idea: Not to use Hassian matrix, we use linearized approximation of learning model.

- $E(\theta) = \frac{1}{2}\|d - g(x,\theta)\|^2$

  $g(x,\theta) \approx g(x,\theta_{now}) + J^T(\theta - \theta_{now}),$

  where Jacobian $J = {}^{dg(x,\theta)}/_{d\theta}\big|_{\theta=\theta_{now}}$

- $E(\theta) = \frac{1}{2}\|d - g(x,\theta_{now}) - J^T(\theta - \theta_{now})\|^2$

Gauss Newton Minimization

- Since the function is quadratic for θ,
- Its minimum can be determined by differentiating & setting to 0.
  - $E(\theta) = \frac{1}{2}\|d - g(x, \theta_{now}) - J^T(\theta - \theta_{now})\|^2$
  - $\nabla E(\theta) = -J\left(d - g(x, \theta_{now}) - J^T(\theta - \theta_{now})\right) = 0$

- Update Rule
  - $\theta_{next} = \theta_{now} + (JJ^T)^{-1}J(d - g(x, \theta_{now}))$
  - $\theta_{next} = \theta_{now} - (JJ^T)^{-1}\nabla E(\theta_{now})$

    $[\because \nabla E(\theta_{now}) = -J\left(d - g(x, \theta_{now})\right)]$

Equality constrained minimization

$$\begin{array}{ll} \text{minimize} & f(\theta) \\ \text{subject to} & A\theta = b, \quad A \in \mathbf{R}^{p\times n}, rank(A) = p, p \leq n \end{array}$$

- f convex, twice continuously differentiable
- we assume p^* is finite and attained

KKT optimality conditions:

$L(\theta, v) = f(\theta) + v^T(A\theta - b)$

$$\begin{aligned} &(4)\ \nabla f(\theta) + A^T v^* = 0 \\ &(1)\ \ A\theta^* = b \end{aligned}$$

- If $f = \left(\frac{1}{2}\right)\theta^T P\theta + q^T\theta + r$

$\nabla f(\theta) = P\theta + q$

$$\begin{aligned} &P\theta^* + q + A^T v^* = 0 \\ &A\theta^* = b \end{aligned} \quad \Rightarrow \begin{bmatrix} P & A^T \\ A & 0 \end{bmatrix}\begin{bmatrix} \theta^* \\ v^* \end{bmatrix} = \begin{bmatrix} -q \\ b \end{bmatrix}$$

Equality constrained minimization

- **equality constrained quadratic minimization** (with $P \in \mathrm{S}_{+}^{\mathrm{n}}$)

$$\begin{array}{ll} \text{minimize} & (1/2)\theta^T P\theta + q^T\theta + r \\ \text{subject to} & A\theta = b,\ \ \rho(A) = p \end{array}$$

- optimality condition:

$$\begin{bmatrix} P & A^T \\ A & 0 \end{bmatrix}\begin{bmatrix} \theta^* \\ \nu^* \end{bmatrix} = \begin{bmatrix} -q \\ b \end{bmatrix}$$

- coefficient matrix is called KKT matrix
- KKT matrix is nonsingular if and only if

$$A\theta = 0,\ \theta \neq 0 \quad \Rightarrow \quad \theta^T P\theta > 0$$

- equivalent condition for nonsingularity:

$$P + A^T A \succ 0$$

$$N(A) \neq N(P) \\ \Rightarrow \rho([P \quad A^T]) = n$$

$$\theta^T(P + A^T A)\theta \\ = \theta^T P\theta + \theta^T A^T A\theta \succ 0$$

$$\rho\left(\begin{bmatrix} P & A^T \\ A & 0 \end{bmatrix}\right) = n + p$$

Newton step

- Original Problem

minimize $\quad f(\theta)$

subject to $\quad A\theta = b$

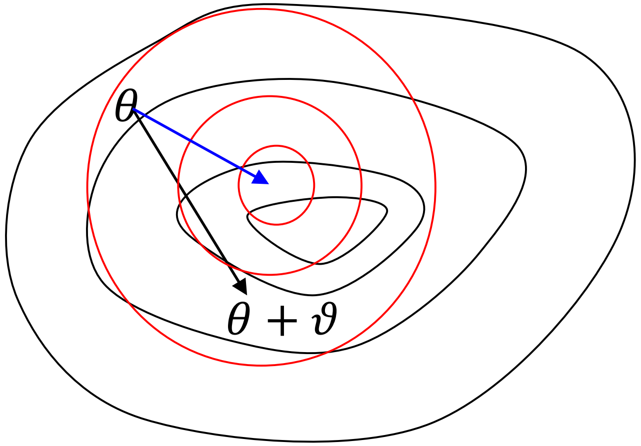
$$L(\theta, v) = f(\theta) + v^T(A\theta - b)$$

$$(4)\ \nabla f(\theta) + A^T v^* = 0$$
$$(1)\ A\theta^* = b$$

- Second order approximation

minimize $\quad \hat{f}(\theta + \vartheta) = f(\theta) + \nabla f(\theta)^T \vartheta + (1/2)\vartheta^T \nabla^2 f(\theta)\vartheta$

subject to $\quad A(\theta + \vartheta) = b$



- Optimal solution of ϑ becomes Newton step $\Delta\theta_{nt}$

Newton step

- Second order approximation

$$\begin{array}{ll} \text{minimize} & \hat{f}(\theta+\vartheta) = f(\theta) + \nabla f(\theta)^T\vartheta + (1/2)\vartheta^T\nabla^2 f(\theta)\vartheta \\ \text{subject to} & A(\theta+\vartheta) = b \end{array}$$

- Optimality condition for optimal point (Newton step) $\Delta\theta_{nt}$

$$\nabla_{\Delta x_{nt}}\hat{f}(\theta+\Delta\theta_{nt}) + A^T w = 0, \qquad A(\theta+\Delta\theta_{nt}) = b$$

$$\begin{array}{l} \nabla f(\theta) + \nabla^2 f(\theta)\Delta\theta_{nt} + A^T w = 0 \\ A\Delta\theta_{nt} = 0 \end{array} \qquad \begin{bmatrix} \nabla^2 f(\theta) & A^T \\ A & 0 \end{bmatrix}\begin{bmatrix} \Delta\theta_{nt} \\ w \end{bmatrix} = \begin{bmatrix} -\nabla f(\theta) \\ 0 \end{bmatrix}$$

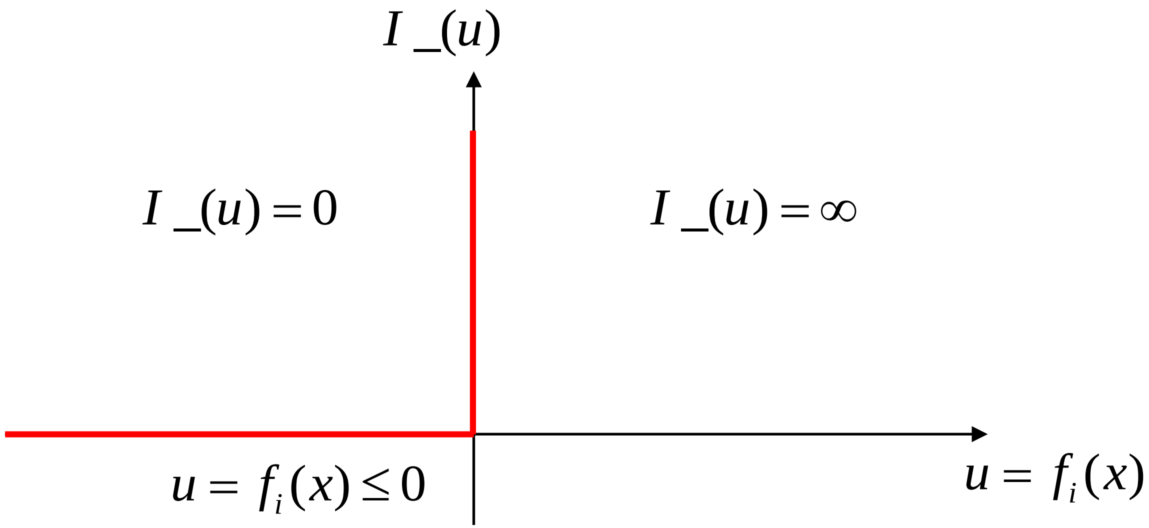
Logarithmic barrier

Original formulation

minimize $\quad f_0(\theta)$

subject to $\quad f_i(\theta) \leq 0, \qquad i = 1, \dots, m$

$\qquad\qquad\quad A\theta = b$



Reformulation via indicator function:

minimize $\quad f_0(x) + \sum_{i=1}^{m} I\_(f_i(x))$

subject to $\quad Ax = b$

Where $I\_(u) = 0$ if $u \leq 0$, $\quad I\_(u) = \infty$, otherwise (indicator function of $\mathbf{R}\_$)

Inequality constrained minimization

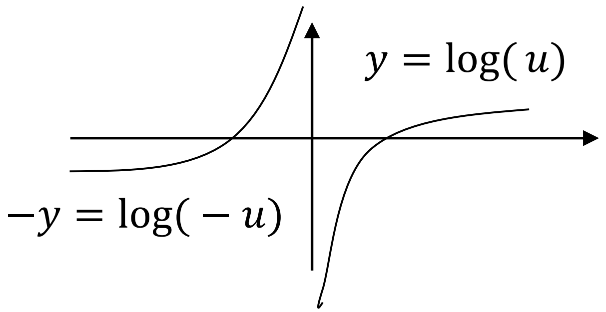
- **original problem**

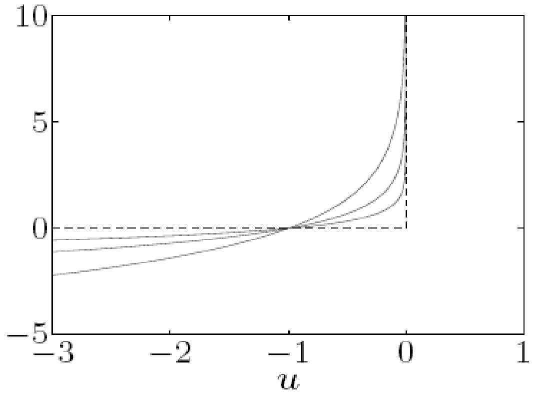
$$\begin{aligned} &\text{minimize} && f_0(\theta) \\ &\text{subject to} && f_i(\theta) \le 0, \quad i = 1, \dots, m \\ & && A\theta = b \end{aligned}$$

- **approximation via logarithmic barrier**

$$\begin{aligned} &\text{minimize} && f_0(\theta) - (1/t) \textstyle\sum_{i=1}^{m} \log(-f_i(\theta)) \\ &\text{subject to} && A\theta = b \end{aligned}$$

where $t > 0, -(1/t)\log(-u)$





Central path

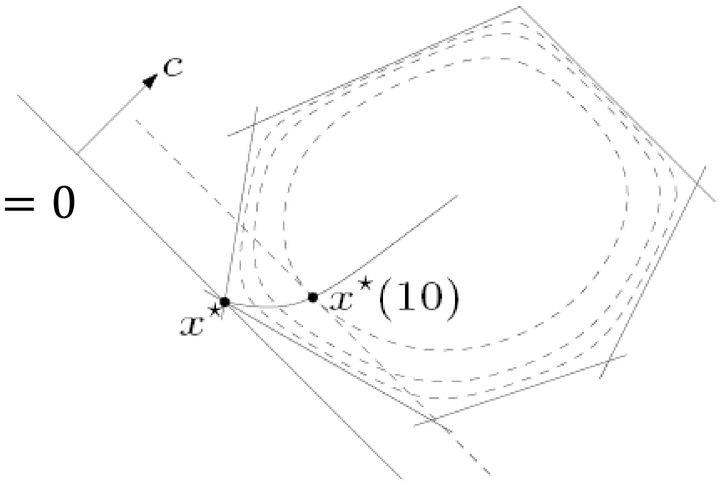
- **example**: central path for an LP

minimize $\quad c^T x$

subject to $\quad a_i^T x \leq b_i, \quad i = 1, \dots, 6$

minimize $\quad tc^T x - \sum_{i=1}^{6} \log(b_i - a_i^T x)$ → Interior Point Method

$$y = c^T x \to c^T x - y = 0$$



Summary

- Constraint Convex Optimization
- linear/quadratic programming
- dual problem, KKT conditions
- Minimization techniques
  - Gradient Descent Minimization
  - Newton Minimization
  - Gauss Newton Minimization
  - (In)Equality Constraint Minimization