Information Theory and Probabilistic Programming

Samuel Cheng

School of ECE
University of Oklahoma

November 5, 2023

Exponential family distributions

Motivation

- Consider random variable X with some known expectation $E[T_i(X)] = \mu_i$, what is the most probable distribution?

Motivation

- Consider random variable X with some known expectation $E[T_i(X)] = \mu_i$, what is the most probable distribution?
- Let's maximize the differential entropy $h(x)$ given the constraint. Using Lagrange multiplier,

Motivation

- Consider random variable X with some known expectation $E[T_i(X)] = \mu_i$, what is the most probable distribution?
- Let's maximize the differential entropy $h(x)$ given the constraint. Using Lagrange multiplier,

$$\underbrace{\arg\max_{p(x)} \left[h(p) + \sum_{i=1}^{m} \eta_i (E[T_i(X)] - \mu_i) + \underbrace{\lambda \left(\int_x p(x) dx - 1 \right)}_{p(x) \text{ normalized to } 1} + \underbrace{\int_x \tilde{g}(x) p(x) dx}_{p(x) \geq 0} \right]}_{L(p)}$$

Motivation

- Consider random variable X with some known expectation $E[T_i(X)] = \mu_i$, what is the most probable distribution?
- Let's maximize the differential entropy $h(x)$ given the constraint. Using Lagrange multiplier,

$$\underbrace{\arg\max_{p(x)} \left[h(p) + \sum_{i=1}^{m} \eta_i (E[T_i(X)] - \mu_i) + \underbrace{\lambda \left(\int_x p(x)dx - 1 \right)}_{p(x) \text{ normalized to } 1} + \underbrace{\int_x \tilde{g}(x)p(x)dx}_{p(x)\geq 0} \right]}_{L(p)}$$

$$\frac{\partial L(p)}{\partial p(x')} = -\log p(x') - 1 + \sum_{i=1}^{m} \eta_i T_i(x') + \lambda + \tilde{g}(x')$$

Motivation

- Consider random variable X with some known expectation $E[T_i(X)] = \mu_i$, what is the most probable distribution?
- Let's maximize the differential entropy $h(x)$ given the constraint. Using Lagrange multiplier,

$$\underbrace{\arg\max_{p(x)} \left[h(p) + \sum_{i=1}^{m} \eta_i (E[T_i(X)] - \mu_i) + \underbrace{\lambda \left(\int_x p(x)dx - 1 \right)}_{p(x) \text{ normalized to } 1} + \underbrace{\int_x \tilde{g}(x)p(x)dx}_{p(x)\geq 0} \right]}_{L(p)}$$

$$\frac{\partial L(p)}{\partial p(x')} = -\log p(x') - 1 + \sum_{i=1}^{m} \eta_i T_i(x') + \lambda + \tilde{g}(x') = 0$$

Motivation

- Consider random variable X with some known expectation $E[T_i(X)] = \mu_i$, what is the most probable distribution?
- Let's maximize the differential entropy $h(x)$ given the constraint. Using Lagrange multiplier,

$$\arg\max_{p(x)} \underbrace{\left[h(p) + \sum_{i=1}^{m} \eta_i (E[T_i(X)] - \mu_i) + \underbrace{\lambda \left(\int_x p(x)dx - 1 \right)}_{p(x) \text{ normalized to } 1} + \underbrace{\int_x \tilde{g}(x) p(x) dx}_{p(x) \geq 0} \right]}_{L(p)}$$

$$\frac{\partial L(p)}{\partial p(x')} = -\log p(x') - 1 + \sum_{i=1}^{m} \eta_i T_i(x') + \lambda + \tilde{g}(x') = 0$$

$$\Rightarrow \quad p(x') = \underbrace{\exp(\tilde{g}(x'))}_{g(x')} [\exp(\underbrace{1 - \lambda}_{A(\eta)}) \exp(\eta^\top T(x'))]$$

Motivation

- Consider random variable X with some known expectation $E[T_i(X)] = \mu_i$, what is the most probable distribution?
- Let's maximize the differential entropy $h(x)$ given the constraint. Using Lagrange multiplier,

$$\arg\max_{p(x)} \underbrace{\left[h(p) + \sum_{i=1}^{m} \eta_i (E[T_i(X)] - \mu_i) + \underbrace{\lambda \left(\int_x p(x)dx - 1 \right)}_{p(x) \text{ normalized to } 1} + \underbrace{\int_x \tilde{g}(x) p(x) dx}_{p(x) \geq 0} \right]}_{L(p)}$$

$$\begin{aligned} \frac{\partial L(p)}{\partial p(x')} &= -\log p(x') - 1 + \sum_{i=1}^{m} \eta_i T_i(x') + \lambda + \tilde{g}(x') = 0 \\ \Rightarrow \quad p(x') &= \underbrace{\exp(\tilde{g}(x'))}_{g(x')} [\exp(\underbrace{1 - \lambda}_{A(\eta)}) \exp(\eta^\top T(x'))] \\ &= g(x') \exp(\eta^\top T(x') - A(\eta)) \end{aligned}$$

Anatomy of exponential family probability function

$$p(x) = g(x)[\exp(\eta^\top T(x) - A(\eta))]$$

- $T(x)$: sufficient statistics of the distribution. As the name suggests, knowing the expectation of $T(X)$ is sufficient to derive the distribution. (That was how we derived in the first place)

Anatomy of exponential family probability function

$$p(x) = g(x)[\exp(\eta^\top T(x) - A(\eta))]$$

- $T(x)$: sufficient statistics of the distribution. As the name suggests, knowing the expectation of $T(X)$ is sufficient to derive the distribution. (That was how we derived in the first place)
- $A(\eta)$: log-partition function

Anatomy of exponential family probability function

$$p(x) = g(x)[\exp(\eta^\top T(x) - A(\eta))]$$

- $T(x)$: sufficient statistics of the distribution. As the name suggests, knowing the expectation of $T(X)$ is sufficient to derive the distribution. (That was how we derived in the first place)
- $A(\eta)$: log-partition function
- η: the natural parameter. Above is known as the natural form, an "unnatural" one can be

$$p(x) = g(x)[\exp(\eta(\theta)^\top T(x) - A(\theta))]$$

Log-partition function

Since $p(x)$ should be normalized to 1, so we have

$$A(\eta) = \ln \int_x g(x) \exp(\eta^\top T(x)) dx$$

Log-partition function

Since $p(x)$ should be normalized to 1, so we have

$$A(\eta) = \ln \int_x g(x) \exp(\eta^\top T(x)) dx$$

Note that

$$\frac{\partial A(\eta)}{\partial \eta_i} = \frac{\int_x g(x) \exp(\eta^\top T(x)) T_i(x) dx}{\int_x g(x) \exp(\eta^\top T(x)) dx}$$

Log-partition function

Since $p(x)$ should be normalized to 1, so we have

$$A(\eta) = \ln \int_x g(x) \exp(\eta^\top T(x)) dx$$

Note that

$$\frac{\partial A(\eta)}{\partial \eta_i} = \frac{\int_x g(x) \exp(\eta^\top T(x)) T_i(x) dx}{\int_x g(x) \exp(\eta^\top T(x)) dx} = \int_x p(x) T_i(x) dx = E[T_i(X)]$$

Log-partition function

Since $p(x)$ should be normalized to 1, so we have

$$A(\eta) = \ln \int_x g(x) \exp(\eta^\top T(x)) dx$$

Note that

$$\frac{\partial A(\eta)}{\partial \eta_i} = \frac{\int_x g(x) \exp(\eta^\top T(x)) T_i(x) dx}{\int_x g(x) \exp(\eta^\top T(x)) dx} = \int_x p(x) T_i(x) dx = E[T_i(X)]$$

and

$$\begin{aligned}\frac{\partial^2 A(\eta)}{\partial \eta_j \partial \eta_i} = \frac{\partial}{\partial \eta_j}\left(\frac{\int_x g(x) \exp(\eta^\top T(x)) T_i(x) dx}{\int_x g(x) \exp(\eta^\top T(x)) dx}\right) = \frac{\int_x g(x) \exp(\eta^\top T(x)) T_j(x) T_i(x) dx}{\int_x g(x) \exp(\eta^\top T(x)) dx} \\ - \frac{\int_x g(x) \exp(\eta^\top T(x)) T_i(x) dx}{\int_x g(x) \exp(\eta^\top T(x)) dx} \frac{\int_x g(x) \exp(\eta^\top T(x)) T_j(x) dx}{\int_x g(x) \exp(\langle \eta, T(x) \rangle) dx} \\ = E[T_i(X) T_j(X)] - E[T_i(X)] E[T_j(X)]\end{aligned}$$

Gaussian distribution

- Gaussian distribution belongs to the exponential family as

$$p(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{(x-\mu)^2}{2\sigma^2}\right) = \frac{1}{\sqrt{2\pi}} \exp\left(\frac{\mu}{\sigma^2}x - \frac{1}{2\sigma^2}x^2 - \frac{\mu^2}{2\sigma^2} - \log\sigma\right)$$

Gaussian distribution

- Gaussian distribution belongs to the exponential family as

$$p(x|\mu,\sigma^2)=\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(\frac{(x-\mu)^2}{2\sigma^2}\right)=\frac{1}{\sqrt{2\pi}}\exp\left(\frac{\mu}{\sigma^2}x-\frac{1}{2\sigma^2}x^2-\frac{\mu^2}{2\sigma^2}-\log\sigma\right)$$

- Thus,

$$\begin{aligned}\eta&=\left[\frac{\mu}{\sigma^2},-\frac{1}{2\sigma^2}\right]\\ T(x)&=\left[x,x^2\right]\\ A(\eta)&=\frac{\mu^2}{2\sigma^2}+\log\sigma=-\frac{\eta_1^2}{4\eta_2}-\frac{1}{2}\log(-2\eta_2)\\ g(x)&=\frac{1}{\sqrt{2\pi}}\end{aligned}$$

Conjugate prior

- Consider observations $x_i \sim G(\cdot|\eta)$ for $i = 1, \cdots, n$ and the prior $\eta \sim F(\cdot|\lambda)$

Conjugate prior

- Consider observations $x_i \sim G(\cdot|\eta)$ for $i = 1, \cdots, n$ and the prior $\eta \sim F(\cdot|\lambda)$
- The posterior distribution of η is then given by

$$p(\eta|x^n, \lambda) \propto F(\eta|\lambda) \prod_{i=1}^{n} G(x_i|\eta)$$

Conjugate prior

- Consider observations $x_i \sim G(\cdot|\eta)$ for $i = 1, \cdots, n$ and the prior $\eta \sim F(\cdot|\lambda)$
- The posterior distribution of η is then given by

$$p(\eta|x^n, \lambda) \propto F(\eta|\lambda) \prod_{i=1}^{n} G(x_i|\eta)$$

- If the prior $F(\cdot|\lambda)$ has the same form as the above posterior, we call $F(\cdot|\lambda)$ a conjugate prior of $G(\cdot|\eta)$

Conjugate prior of exponential family

- Any exponential family distribution has conjugate prior which also belongs to the exponential family. More precisely, consider

$$p(x|\eta) = g(x)\exp(\eta^\top T(x) - A(\eta))$$

Conjugate prior of exponential family

- Any exponential family distribution has conjugate prior which also belongs to the exponential family. More precisely, consider

  $$p(x|\eta) = g(x)\exp(\eta^\top T(x) - A(\eta))$$

  It is easy to verify that following is its conjugate prior

  $$p(\eta|\lambda) = \tilde{g}(\eta)\exp(\lambda_1^\top \eta - \lambda_2 A(\eta) - \tilde{A}(\lambda))$$

Conjugate prior of exponential family

- Any exponential family distribution has conjugate prior which also belongs to the exponential family. More precisely, consider
$$p(x|\eta) = g(x)\exp(\eta^\top T(x) - A(\eta))$$
It is easy to verify that following is its conjugate prior
$$p(\eta|\lambda) = \tilde{g}(\eta)\exp(\lambda_1^\top \eta - \lambda_2 A(\eta) - \tilde{A}(\lambda))$$
- Then, we have
$$p(\eta|x^n, \lambda) \propto p(\eta|\lambda)\prod_{i=1}^{n} p(x_i|\eta)$$

Conjugate prior of exponential family

- Any exponential family distribution has conjugate prior which also belongs to the exponential family. More precisely, consider

$$p(x|\eta) = g(x)\exp(\eta^\top T(x) - A(\eta))$$

It is easy to verify that following is its conjugate prior

$$p(\eta|\lambda) = \tilde{g}(\eta)\exp(\lambda_1^\top \eta - \lambda_2 A(\eta) - \tilde{A}(\lambda))$$

- Then, we have

$$p(\eta|x^n,\lambda) \propto p(\eta|\lambda)\prod_{i=1}^n p(x_i|\eta)$$

$$= \left(\tilde{g}(\eta)\prod_{i=1}^n g(x_i)\right)\exp\left(\underbrace{\left(\lambda_1 + \sum_{i=1}^n T(x_i)\right)^\top}_{\lambda_1 \leftarrow \lambda_1 + \sum_{i=1}^n T(x_i)} \eta - \underbrace{(\lambda_2 + n)}_{\lambda_2 \leftarrow \lambda_2 + n} A(\eta) - \tilde{A}(\lambda)\right)$$

Binomial distribution belongs to the exponential family

For a fixed n,

$$f(x|\eta) = f(x|p) = \binom{n}{x} p^x (1-p)^{n-x} = \binom{n}{x} \exp\left(x \log p + (n-x)\log(1-p)\right)$$

Binomial distribution belongs to the exponential family

For a fixed n,

$$f(x|\eta) = f(x|p) = \binom{n}{x} p^x (1-p)^{n-x} = \underbrace{\binom{n}{x}}_{g(x)} \exp\left(\underbrace{x}_{T(x)} \underbrace{\log \frac{p}{1-p}}_{\eta} - \underbrace{n \log \frac{1}{(1-p)}}_{A(\eta)}\right)$$

Binomial distribution belongs to the exponential family

For a fixed n,

$$f(x|\eta) = f(x|p) = \binom{n}{x} p^x (1-p)^{n-x} = \underbrace{\binom{n}{x}}_{g(x)} \exp\left(\underbrace{x}_{T(x)} \underbrace{\log \frac{p}{1-p}}_{\eta} - \underbrace{n \log \frac{1}{(1-p)}}_{A(\eta)} \right)$$

Beta prior:

$$f(p|\alpha,\beta) = \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)}$$

Binomial distribution belongs to the exponential family

For a fixed n,

$$f(x|\eta) = f(x|p) = \binom{n}{x} p^x (1-p)^{n-x} = \underbrace{\binom{n}{x}}_{g(x)} \exp\left(\underbrace{x}_{T(x)} \underbrace{\log \frac{p}{1-p}}_{\eta} - \underbrace{n \log \frac{1}{(1-p)}}_{A(\eta)} \right)$$

Beta prior:

$$f(p|\alpha,\beta) = \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)} = \tfrac{1}{B(\alpha,\beta)} \exp((\alpha-1)\log(p) + (\beta-1)\log(1-p))$$

Binomial distribution belongs to the exponential family

For a fixed n,

$$f(x|\eta) = f(x|p) = \binom{n}{x} p^x (1-p)^{n-x} = \underbrace{\binom{n}{x}}_{g(x)} \exp\left(\underbrace{x}_{T(x)} \underbrace{\log \frac{p}{1-p}}_{\eta} - \underbrace{n \log \frac{1}{(1-p)}}_{A(\eta)} \right)$$

Beta prior:

$$f(p|\alpha,\beta) = \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)} = \tfrac{1}{B(\alpha,\beta)} \exp((\alpha-1)\log(p) + (\beta-1)\log(1-p))$$

$$= \tfrac{1}{B(\alpha,\beta)} \exp(\underbrace{(\alpha-1)}_{\lambda_1} \log \tfrac{p}{1-p} - \underbrace{\frac{\alpha+\beta-2}{n}}_{\lambda_2} n \log \tfrac{1}{1-p})$$

Binomial distribution belongs to the exponential family

For a fixed n,

$$f(x|\eta) = f(x|p) = \binom{n}{x} p^x (1-p)^{n-x} = \underbrace{\binom{n}{x}}_{g(x)} \exp\left(\underbrace{x}_{T(x)} \underbrace{\log \frac{p}{1-p}}_{\eta} - \underbrace{n \log \frac{1}{(1-p)}}_{A(\eta)} \right)$$

Beta prior:

$$f(p|\alpha,\beta) = \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)} = \tfrac{1}{B(\alpha,\beta)} \exp((\alpha-1)\log(p) + (\beta-1)\log(1-p))$$

$$= \tfrac{1}{B(\alpha,\beta)} \exp(\underbrace{(\alpha-1)}_{\lambda_1} \log \tfrac{p}{1-p} - \underbrace{\frac{\alpha+\beta-2}{n}}_{\lambda_2} n \log \tfrac{1}{1-p})$$

Recall: $\lambda_1 \leftarrow \lambda_1 + T(x) \Rightarrow \alpha \leftarrow \alpha + x$

Binomial distribution belongs to the exponential family

For a fixed n,

$$f(x|\eta) = f(x|p) = \binom{n}{x} p^x (1-p)^{n-x} = \underbrace{\binom{n}{x}}_{g(x)} \exp\left(\underbrace{x}_{T(x)} \underbrace{\log \frac{p}{1-p}}_{\eta} - \underbrace{n \log \frac{1}{(1-p)}}_{A(\eta)} \right)$$

Beta prior:

$$f(p|\alpha,\beta) = \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)} = \tfrac{1}{B(\alpha,\beta)} \exp((\alpha-1)\log(p) + (\beta-1)\log(1-p))$$

$$= \tfrac{1}{B(\alpha,\beta)} \exp(\underbrace{(\alpha-1)}_{\lambda_1} \log \tfrac{p}{1-p} - \underbrace{\frac{\alpha+\beta-2}{n}}_{\lambda_2} n \log \tfrac{1}{1-p})$$

Recall: $\lambda_1 \leftarrow \lambda_1 + T(x) \Rightarrow \alpha \leftarrow \alpha + x$ and $\lambda_2 \leftarrow \lambda_2 + 1 \Rightarrow \alpha + \beta \leftarrow \alpha + \beta + n$

Binomial distribution belongs to the exponential family

For a fixed n,

$$f(x|\eta) = f(x|p) = \binom{n}{x} p^x (1-p)^{n-x} = \underbrace{\binom{n}{x}}_{g(x)} \exp\left(\underbrace{x}_{T(x)} \underbrace{\log \frac{p}{1-p}}_{\eta} - \underbrace{n \log \frac{1}{(1-p)}}_{A(\eta)} \right)$$

Beta prior:

$$f(p|\alpha,\beta) = \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)} = \tfrac{1}{B(\alpha,\beta)} \exp((\alpha-1)\log(p) + (\beta-1)\log(1-p))$$

$$= \tfrac{1}{B(\alpha,\beta)} \exp(\underbrace{(\alpha-1)}_{\lambda_1} \log \tfrac{p}{1-p} - \underbrace{\frac{\alpha+\beta-2}{n}}_{\lambda_2} n \log \tfrac{1}{1-p})$$

Recall: $\lambda_1 \leftarrow \lambda_1 + T(x) \Rightarrow \alpha \leftarrow \alpha + x$ and $\lambda_2 \leftarrow \lambda_2 + 1, \lambda_1 \leftarrow \lambda_1 + T(x) \Rightarrow \beta \leftarrow \beta + n - k$

Unit variance Gaussian

- Consider unit variance Gaussian $p(x|\mu) = \frac{1}{\sqrt{2\pi}} \exp(-(x-\mu)^2/2)$

Unit variance Gaussian

- Consider unit variance Gaussian $p(x|\mu) = \frac{1}{\sqrt{2\pi}} \exp(-(x-\mu)^2/2) = \frac{\exp(-x^2/2)}{\sqrt{2\pi}} \exp(\mu x - \mu^2/2)$.

Unit variance Gaussian

- Consider unit variance Gaussian $p(x|\mu) = \frac{1}{\sqrt{2\pi}} \exp(-(x-\mu)^2/2) = \frac{\exp(-x^2/2)}{\sqrt{2\pi}} \exp(\mu x - \mu^2/2)$. Thus, $\eta = \mu$, $T(x) = x$, $g(x) = \frac{\exp(-x^2/2)}{\sqrt{2\pi}}$, and $A(\eta) = \eta^2/2$

Unit variance Gaussian

- Consider unit variance Gaussian $p(x|\mu) = \frac{1}{\sqrt{2\pi}} \exp(-(x-\mu)^2/2) = \frac{\exp(-x^2/2)}{\sqrt{2\pi}} \exp(\mu x - \mu^2/2)$. Thus, $\eta = \mu$, $T(x) = x$, $g(x) = \frac{\exp(-x^2/2)}{\sqrt{2\pi}}$, and $A(\eta) = \eta^2/2$

$$p(x|\eta) = g(x) \exp(\eta x - \eta^2/2)$$

Unit variance Gaussian

- Consider unit variance Gaussian $p(x|\mu) = \frac{1}{\sqrt{2\pi}} \exp(-(x-\mu)^2/2) = \frac{\exp(-x^2/2)}{\sqrt{2\pi}} \exp(\mu x - \mu^2/2)$. Thus, $\eta = \mu$, $T(x) = x$, $g(x) = \frac{\exp(-x^2/2)}{\sqrt{2\pi}}$, and $A(\eta) = \eta^2/2$

$$p(x|\eta) = g(x) \exp(\eta x - \eta^2/2)$$

- So the conjugate prior is $p(\eta|\lambda) \propto \tilde{g}(\eta) \exp(\lambda_1 \eta - \frac{\lambda_2 \eta^2}{2} - \tilde{A}(\lambda))$.

Unit variance Gaussian

- Consider unit variance Gaussian $p(x|\mu) = \frac{1}{\sqrt{2\pi}} \exp(-(x-\mu)^2/2) = \frac{\exp(-x^2/2)}{\sqrt{2\pi}} \exp(\mu x - \mu^2/2)$. Thus, $\eta = \mu$, $T(x) = x$, $g(x) = \frac{\exp(-x^2/2)}{\sqrt{2\pi}}$, and $A(\eta) = \eta^2/2$

$$p(x|\eta) = g(x)\exp(\eta x - \eta^2/2)$$

- So the conjugate prior is $p(\eta|\lambda) \propto \cancel{\tilde{g}(\eta)} \exp(\lambda_1 \eta - \frac{\lambda_2 \eta^2}{2} - \tilde{A}(\lambda))$. It is a Gaussian distribution.

Unit variance Gaussian

- Consider unit variance Gaussian $p(x|\mu)=\frac{1}{\sqrt{2\pi}}\exp(-(x-\mu)^2/2)=\frac{\exp(-x^2/2)}{\sqrt{2\pi}}\exp(\mu x-\mu^2/2)$. Thus, $\eta=\mu$, $T(x)=x$, $g(x)=\frac{\exp(-x^2/2)}{\sqrt{2\pi}}$, and $A(\eta)=\eta^2/2$
$$p(x|\eta)=g(x)\exp(\eta x-\eta^2/2)$$

- So the conjugate prior is $p(\eta|\lambda)\propto\cancel{\tilde{g}(\eta)}\exp(\lambda_1\eta-\frac{\lambda_2\eta^2}{2}-\tilde{A}(\lambda))$. It is a Gaussian distribution. By inspection,
$$p(\eta|\lambda)=\sqrt{\frac{\lambda_2}{2\pi}}\exp\left(-\frac{\lambda_2}{2}\left(\eta-\frac{\lambda_1}{\lambda_2}\right)^2\right)$$

Unit variance Gaussian

- Consider unit variance Gaussian $p(x|\mu) = \frac{1}{\sqrt{2\pi}}\exp(-(x-\mu)^2/2) = \frac{\exp(-x^2/2)}{\sqrt{2\pi}}\exp(\mu x - \mu^2/2)$. Thus, $\eta = \mu$, $T(x) = x$, $g(x) = \frac{\exp(-x^2/2)}{\sqrt{2\pi}}$, and $A(\eta) = \eta^2/2$

$$p(x|\eta) = g(x)\exp(\eta x - \eta^2/2)$$

- So the conjugate prior is $p(\eta|\lambda) \propto \cancel{\tilde{g}(\eta)}\exp(\lambda_1\eta - \frac{\lambda_2\eta^2}{2} - \tilde{A}(\lambda))$. It is a Gaussian distribution. By inspection,

$$p(\eta|\lambda) = \sqrt{\frac{\lambda_2}{2\pi}}\exp\left(-\frac{\lambda_2}{2}\left(\eta - \frac{\lambda_1}{\lambda_2}\right)^2\right)$$

Thus, $\mu_\eta = \frac{\lambda_1}{\lambda_2}$ and $\sigma_\eta^2 = \frac{1}{\lambda_2} \Rightarrow \lambda_1 = \frac{\mu_\eta}{\sigma_\eta^2}$ and $\lambda_2 = \frac{1}{\sigma_\eta^2}$

Unit variance Gaussian

- Consider unit variance Gaussian $p(x|\mu) = \frac{1}{\sqrt{2\pi}}\exp(-(x-\mu)^2/2) = \frac{\exp(-x^2/2)}{\sqrt{2\pi}}\exp(\mu x - \mu^2/2)$. Thus, $\eta = \mu$, $T(x) = x$, $g(x) = \frac{\exp(-x^2/2)}{\sqrt{2\pi}}$, and $A(\eta) = \eta^2/2$

$$p(x|\eta) = g(x)\exp(\eta x - \eta^2/2)$$

- So the conjugate prior is $p(\eta|\lambda) \propto \cancel{\tilde{g}(\eta)}\exp(\lambda_1\eta - \frac{\lambda_2\eta^2}{2} - \tilde{A}(\lambda))$. It is a Gaussian distribution. By inspection,

$$p(\eta|\lambda) = \sqrt{\frac{\lambda_2}{2\pi}}\exp\left(-\frac{\lambda_2}{2}\left(\eta - \frac{\lambda_1}{\lambda_2}\right)^2\right)$$

Thus, $\mu_\eta = \frac{\lambda_1}{\lambda_2}$ and $\sigma_\eta^2 = \frac{1}{\lambda_2} \Rightarrow \lambda_1 = \frac{\mu_\eta}{\sigma_\eta^2}$ and $\lambda_2 = \frac{1}{\sigma_\eta^2}$

- For posterior update given observations $T(x_i) = x_i$,

$$\begin{cases}\lambda_1 \leftarrow \lambda_1 + \sum_{i=1}^n x_i \\ \lambda_2 \leftarrow \lambda_2 + n\end{cases} \Rightarrow \begin{cases}\mu_\eta \leftarrow \frac{\lambda_1 + \sum_{i=1}^n x_i}{\lambda_2 + n} = \frac{\mu_\eta/\sigma_\eta^2 + \sum_{i=1}^n x_i}{1/\sigma_\eta^2 + n} \\ \sigma_\eta^2 \leftarrow \frac{1}{\lambda_2 + n} = \frac{1}{1/\sigma_\eta^2 + n}\end{cases}$$

Remark

Try not to confuse σ_η^2 and the variance of observation

- Variance of observation is 1
- σ_η^2 is the variance of μ since the mean of of the observation is a random variable also (with mean $\mu_\eta = \lambda_1/\lambda_2$). But σ_η^2 decreases as more observations are made as expected

Reference

- The exponential family: Basics
- Exponential families

Fisher information and Cramer-Rao bound

Score and Fisher information

- For a family of density $f(x;\theta)$ parametrized by θ, we define the **score** V as a random variable of fraction of change of $f(X;\theta)$ w.r.t. θ. That is, $V \triangleq \frac{\frac{\partial f(X;\theta)}{\partial\theta}}{f(X;\theta)} = \frac{\partial}{\partial\theta} \ln f(X;\theta)$
- Note that $E[V] = \int \frac{\partial f(x;\theta)}{\partial\theta} \frac{1}{f(x;\theta)} f(x;\theta)dx = \frac{\partial}{\partial\theta} \int f(x;\theta)dx = \frac{\partial}{\partial\theta} 1 = 0$
- We define the Fisher information $J(\theta)$ for X w.r.t. θ as $Var(V) = E[V^2]$

Score and Fisher information for n i.i.d. X

- $V(X_1, \cdots, X_n) = \frac{\partial}{\partial\theta} \ln f(X_1, \cdots, X_n) = \sum_{i=1}^n \frac{\partial}{\partial\theta} \ln f(X_i) = \sum_{i=1}^n V(X_i)$
- $E[V(X_1, \cdots, X_n)] = \sum_{i=1}^n E[V(X_i)] = 0$
- $J(\theta; X_1, \cdots, X_n) = E[V(X_1, \cdots, X_n)^2] = E[(\sum_{i=1}^n V(X_i))^2] = E[\sum_{i=1}^n V(X_i)^2] = \sum_{i=1}^n J(\theta; X_i) = nJ(\theta)$

Cramer-Rao lower bound

- For any unbiased estimator T of θ out of X, i.e., $E[T(X)]=\theta$. The variance of the estimator is lower bounded by the inverse of Fisher information $J(\theta;X)$. That is, $Var(T)=E[T^2(X)]\geq\frac{1}{J(\theta;X)}$
- Proof: consider the Cauchy-Schwarz inequality $E^2[(T-E[T])(V-E[V])]\leq E[(T-E[T])^2]E[(V-E[V])^2]=Var(T)Var(V)=Var(T)J(\theta)$ and $E[(T-E[T])(V-E[V])]=E[TV]-E[T]E[V]=E[TV]=\int T(x)\frac{\partial f(x;\theta)/\partial\theta}{f(x;\theta)}f(x;\theta)dx=\frac{\partial}{\partial\theta}\int T(x)f(x;\theta)d\theta=\frac{\partial}{\partial\theta}E[T]=\frac{\partial}{\partial\theta}\theta=1$
  $\square$

Proof of Cauchy-Schwarz Inequality (real inner product space)

The Cauchy-Schwarz Inequality

In a real inner product space, for any vectors **u** and **v**,

$$|\langle \mathbf{u}, \mathbf{v} \rangle| \leq ||\mathbf{u}|| \cdot ||\mathbf{v}||$$

Case 1: $\mathbf{v} = \mathbf{0}$

If $\mathbf{v} = \mathbf{0}$, then $\langle \mathbf{u}, \mathbf{v} \rangle = 0$, and the inequality holds trivially.

Case 2: $\mathbf{v} \neq \mathbf{0}$

For $\mathbf{v} \neq \mathbf{0}$, we have

$$0 \leq \langle \mathbf{u} - \lambda \mathbf{v}, \mathbf{u} - \lambda \mathbf{v} \rangle \leq \langle \mathbf{u}, \mathbf{u} \rangle - 2\lambda \langle \mathbf{u}, \mathbf{v} \rangle + \lambda^2 \langle \mathbf{v}, \mathbf{v} \rangle$$

Substitute $\lambda = \frac{\langle \mathbf{u},\mathbf{v} \rangle}{\langle \mathbf{v},\mathbf{v} \rangle}$ (that minimizes the right-hand side),

$$0 \leq \langle \mathbf{u}, \mathbf{u} \rangle - \frac{\langle \mathbf{u}, \mathbf{v} \rangle^2}{\langle \mathbf{v}, \mathbf{v} \rangle}$$

Introduction

Square-Integrable Random Variables

A random variable X is square-integrable if $E[X^2] < \infty$, where $E[\cdot]$ denotes expectation.

Expectation as Inner Product

For random variables X and Y, the inner product is defined as:

$$\langle X, Y \rangle = E[XY]$$

- The set of all square-integrable random variables forms an inner product space with expectation as inner product
- This concept is fundamental in probability theory and functional analysis.

Sanity check of the inner product properties

1. **Conjugate Symmetry** (for real variables, symmetry):

$$\langle X, Y\rangle = E[XY] = E[YX] = \langle Y, X\rangle$$

2. **Linearity in the First Argument**:

$$\langle aX + Z, Y\rangle = E[(aX + Z)Y] = aE[XY] + E[ZY] = a\langle X, Y\rangle + \langle Z, Y\rangle$$

3. **Positive-Definiteness**:

$$\langle X, X\rangle = E[X^2] \geq 0$$

$$\langle X, X\rangle = E[X^2] = 0 \Leftrightarrow X = 0\,\text{(almost surely)}$$

Example of Cramer-Rao lower bound

- Consider a normally distributed source $\sim \mathcal{N}(\mu, \sigma^2)$ with known variance σ^2 and we try to estimate the mean μ. Giving n samples $X_1, X_2, \cdots, X_n$
    - A reasonable estimate of μ is simply the average of the samples $\hat{\mu} = \frac{1}{n}\sum_{i=1}^n X_i$
    - The estimate is unbiased as $E[\hat{\mu}] = \frac{1}{n}\sum_{i=1}^n E[X_i] = \mu$
    - And the variance is $Var(\hat{\mu}) = E[(\hat{\mu} - \mu)^2]$

$$= \frac{1}{n^2}\left(\sum_{i=1}^n E[(X_i - \mu)^2] + 2\sum_{i \neq j}^n E[(X_i - \mu)(X_j - \mu)]\right)$$

$$= \frac{1}{n^2}\left(\sum_{i=1}^n E[(X_i - \mu)^2] + 2\sum_{i \neq j}^n E[(X_i - \mu)]E[(X_j - \mu)]\right)$$

$$= \frac{1}{n^2}\left(\sum_{i=1}^n E[(X_i - \mu)^2]\right) = \frac{n\sigma^2}{n^2} = \frac{\sigma^2}{n}$$

- We will use the Cramer-Rao lower bound to show that such estimate is optimal

Example of Cramer-Rao lower bound

- Let's compute $J(\mu; X_1, \cdots, X_n)$, which is equal to $nJ(\mu; X)$. And

$$J(\mu; X) = E\left[\left(\frac{\partial}{\partial\mu}\ln\left(\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(X-\mu)^2}{2\sigma^2}}\right)\right)^2\right]$$

$$= E\left[\left(\frac{X-\mu}{\sigma^2}\right)^2\right] = \frac{1}{\sigma^4}E[(X-\mu)^2] = \frac{1}{\sigma^2}$$

- So $J(\mu; X_1, \cdots, X_n) = \frac{n}{\sigma^2}$ and by Cramer-Rao lower bound, any unbiased estimator cannot has variance less than $\frac{\sigma^2}{n}$. And thus the mean estimate using average samples described in the last slide is optimal

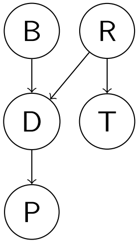
Graphical model and BP

This time...

- Bayesian Net
- Belief Propagation Algorithm
- LDPC/IRA Codes

Bayesian Net

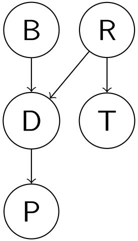
- Relationship of variables depicted by a directed graph with no loop
- Given a variable's parents, the variable is conditionally independent of any non-descendants
- Reduce model complexity
- Facilitate easier inference



Burlgar and racoon

Burlgar: B; Racoon: R; Dog barked: D; Police called: P; Trash can fell: T

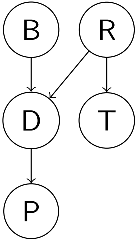
$$p(p, d, b, t, r) = p(p|d, b, t, r)p(d|b, t, r)p(b|t, r)p(t|r)p(r)$$



Burlgar and racoon

Burlgar: B; Racoon: R; Dog barked: D; Police called: P; Trash can fell: T

$$\begin{aligned} p(p,d,b,t,r) &= p(p|d,b,t,r)p(d|b,t,r)p(b|t,r)p(t|r)p(r) \\ &= \underbrace{p(p|d,\cancel{b},\cancel{t},\cancel{r})}_{2\ parameters}p(d|b,\cancel{t},r)p(b|\cancel{t},\cancel{r})p(t|r)p(r) \end{aligned}$$



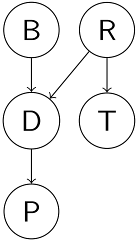
Burlgar and racoon

Burlgar: B; Racoon: R; Dog barked: D; Police called: P; Trash can fell: T

$$\begin{aligned} p(p,d,b,t,r) &= p(p|d,b,t,r)p(d|b,t,r)p(b|t,r)p(t|r)p(r) \\ &= \underbrace{p(p|d,\cancel{b},\cancel{t},\cancel{r})}_{2\ parameters} p(d|b,\cancel{t},r)p(b|\cancel{t},\cancel{r})p(t|r)p(r) \end{aligned}$$

P	D	$p(p\|d)$
p	$\neg d$	0.01
p	d	0.4
$\neg p$	$\neg d$	0.99
$\neg p$	d	0.6

T	R	$p(t\|r)$
t	$\neg r$	0.05
t	r	0.7
$\neg t$	$\neg r$	0.95
$\neg t$	r	0.3



Burlgar and racoon

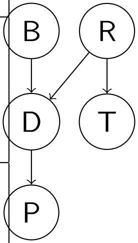
Burlgar: B; Racoon: R; Dog barked: D; Police called: P; Trash can fell: T

$$
\begin{aligned}
p(p,d,b,t,r) =& p(p|d,b,t,r)p(d|b,t,r)p(b|t,r)p(t|r)p(r) \\
=& \underbrace{p(p|d,\cancel{b},\cancel{t},\cancel{r})}_{2\ parameters}p(d|b,\cancel{t},r)p(b|\cancel{t},\cancel{r})p(t|r)p(r)
\end{aligned}
$$

P	D	$p(p\|d)$
p	$\neg d$	0.01
p	d	0.4
$\neg p$	$\neg d$	0.99
$\neg p$	d	0.6

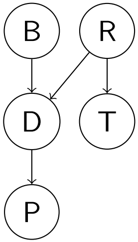
T	R	$p(t\|r)$
t	$\neg r$	0.05
t	r	0.7
$\neg t$	$\neg r$	0.95
$\neg t$	r	0.3

D	B	R	$p(d\|b,r)$
d	$\neg b$	$\neg r$	0.1
d	$\neg b$	r	0.5
d	b	$\neg r$	1
d	b	r	1
$\neg d$	$\neg b$	$\neg r$	0.9
$\neg d$	$\neg b$	r	0.5
$\neg d$	b	$\neg r$	0
$\neg d$	b	r	0



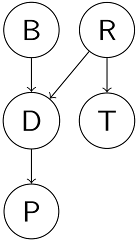
Comparison of # parameters

- # parameters of complete model: $2^5 - 1 = 31$



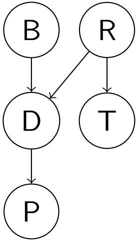
Comparison of # parameters

- # parameters of complete model: $2^5 - 1 = 31$
- # parameters of Bayesian net:



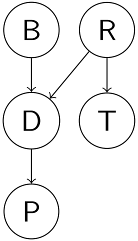
Comparison of # parameters

- # parameters of complete model: $2^5 - 1 = 31$
- # parameters of Bayesian net:
  - $p(p|d)$: 2



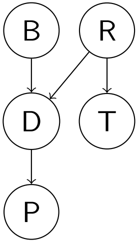
Comparison of # parameters

- # parameters of complete model: $2^5 - 1 = 31$
- # parameters of Bayesian net:
    - $p(p|d)$: 2
    - $p(d|b, r)$: 4



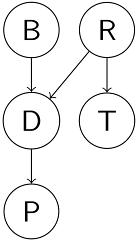
Comparison of # parameters

- # parameters of complete model: $2^5 - 1 = 31$
- # parameters of Bayesian net:
  - $p(p|d)$: 2
  - $p(d|b,r)$: 4
  - $p(b)$: 1



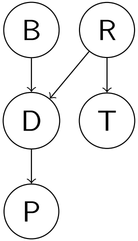
Comparison of # parameters

- # parameters of complete model: $2^5 - 1 = 31$
- # parameters of Bayesian net:
  - $p(p|d)$: 2
  - $p(d|b,r)$: 4
  - $p(b)$: 1
  - $p(t|r)$: 2



Comparison of # parameters

- # parameters of complete model: $2^5 - 1 = 31$
- # parameters of Bayesian net:
  - $p(p|d)$: 2
  - $p(d|b,r)$: 4
  - $p(b)$: 1
  - $p(t|r)$: 2
  - $p(r)$: 1
  - Total: $2 + 4 + 1 + 2 + 1 = 10$
- The model size reduces to less than $\frac{1}{3}$!



Burglar and racoon

Question: What is the probability of a burglar visit if police was called but trash can stayed untouched?

Let $p(r) = 0.2$ and $p(b) = 0.01$

D	B	R	$p(d\|b, r)$
d	$\neg b$	$\neg r$	0.1
d	$\neg b$	r	0.5
d	b	$\neg r$	1
d	b	r	1
$\neg d$	$\neg b$	$\neg r$	0.9
$\neg d$	$\neg b$	r	0.5
$\neg d$	b	$\neg r$	0
$\neg d$	b	r	0

Burglar and racoon

Question: What is the probability of a burglar visit if police was called but trash can stayed untouched?

Let $p(r) = 0.2$ and $p(b) = 0.01$

D	B	R	$p(d\|b,r)$
d	$\neg b$	$\neg r$	0.1
d	$\neg b$	r	0.5
d	b	$\neg r$	1
d	b	r	1
$\neg d$	$\neg b$	$\neg r$	0.9
$\neg d$	$\neg b$	r	0.5
$\neg d$	b	$\neg r$	0
$\neg d$	b	r	0

$\Rightarrow$

D	B	R	$p(d,b,r)$
d	$\neg b$	$\neg r$	0.0792
d	$\neg b$	r	0.099
d	b	$\neg r$	0.008
d	b	r	0.002
$\neg d$	$\neg b$	$\neg r$	0.7128
$\neg d$	$\neg b$	r	0.099
$\neg d$	b	$\neg r$	0
$\neg d$	b	r	0

Burglar and racoon

Question: What is the probability of a burglar visit if police was called but trash can stayed untouched?

P	D	$p(p\|d)$
p	$\neg d$	0.01
p	d	0.4
$\neg p$	$\neg d$	0.99
$\neg p$	d	0.6

P	D	B	R	$p(d,b,r,p)$
p	d	$\neg b$	$\neg r$	0.0792
p	d	$\neg b$	r	0.099
p	d	b	$\neg r$	0.008
p	d	b	r	0.002
p	$\neg d$	$\neg b$	$\neg r$	0.7128
p	$\neg d$	$\neg b$	r	0.099
p	$\neg d$	b	$\neg r$	0
p	$\neg d$	b	r	0
$\cdots$				

Burglar and racoon

Question: What is the probability of a burglar visit if police was called but trash can stayed untouched?

P	D	$p(p\|d)$
p	$\neg d$	0.01
p	d	0.4
$\neg p$	$\neg d$	0.99
$\neg p$	d	0.6

P	D	B	R	$p(d,b,r,p)$
p	d	$\neg b$	$\neg r$	0.0792
p	d	$\neg b$	r	0.099
p	d	b	$\neg r$	0.008
p	d	b	r	0.002
p	$\neg d$	$\neg b$	$\neg r$	0.007128
p	$\neg d$	$\neg b$	r	0.00099
p	$\neg d$	b	$\neg r$	0
p	$\neg d$	b	r	0
$\cdots$				

Burglar and racoon

Question: What is the probability of a burglar visit if police was called but trash can stayed untouched?

P	D	$p(p\|d)$
p	$\neg d$	0.01
p	d	0.4
$\neg p$	$\neg d$	0.99
$\neg p$	d	0.6

P	D	B	R	$p(d,b,r,p)$
p	d	$\neg b$	$\neg r$	0.03168
p	d	$\neg b$	r	0.0396
p	d	b	$\neg r$	0.0032
p	d	b	r	0.0008
p	$\neg d$	$\neg b$	$\neg r$	0.007128
p	$\neg d$	$\neg b$	r	0.00099
p	$\neg d$	b	$\neg r$	0
p	$\neg d$	b	r	0
$\cdots$				

Burglar and racoon

Question: What is the probability of a burglar visit if police was called but trash can stayed untouched?

T	R	$p(t\|r)$
t	$\neg r$	0.05
t	r	0.7
$\neg t$	$\neg r$	0.95
$\neg t$	r	0.3

T	P	D	B	R	$p(d,b,r,p,t)$
$\neg t$	p	d	$\neg b$	$\neg r$	0.03168
$\neg t$	p	d	$\neg b$	r	0.0396
$\neg t$	p	d	b	$\neg r$	0.0032
$\neg t$	p	d	b	r	0.0008
$\neg t$	p	$\neg d$	$\neg b$	$\neg r$	0.007128
$\neg t$	p	$\neg d$	$\neg b$	r	0.00099
$\neg t$	p	$\neg d$	b	$\neg r$	0
$\neg t$	p	$\neg d$	b	r	0
$\cdots$					

Burglar and racoon

Question: What is the probability of a burglar visit if police was called but trash can stayed untouched?

T	R	$p(t\|r)$
t	$\neg r$	0.05
t	r	0.7
$\neg t$	$\neg r$	0.95
$\neg t$	r	0.3

T	P	D	B	R	$p(d,b,r,p,t)$
$\neg t$	p	d	$\neg b$	$\neg r$	0.030096
$\neg t$	p	d	$\neg b$	r	0.0396
$\neg t$	p	d	b	$\neg r$	0.00304
$\neg t$	p	d	b	r	0.0008
$\neg t$	p	$\neg d$	$\neg b$	$\neg r$	0.0067716
$\neg t$	p	$\neg d$	$\neg b$	r	0.00099
$\neg t$	p	$\neg d$	b	$\neg r$	0
$\neg t$	p	$\neg d$	b	r	0
$\cdots$					

Burglar and racoon

Question: What is the probability of a burglar visit if police was called but trash can stayed untouched?

T	R	$p(t\|r)$
t	$\neg r$	0.05
t	r	0.7
$\neg t$	$\neg r$	0.95
$\neg t$	r	0.3

T	P	D	B	R	$p(d,b,r,p,t)$
$\neg t$	p	d	$\neg b$	$\neg r$	0.030096
$\neg t$	p	d	$\neg b$	r	0.01188
$\neg t$	p	d	b	$\neg r$	0.00304
$\neg t$	p	d	b	r	0.00024
$\neg t$	p	$\neg d$	$\neg b$	$\neg r$	0.0067716
$\neg t$	p	$\neg d$	$\neg b$	r	0.000297
$\neg t$	p	$\neg d$	b	$\neg r$	0
$\neg t$	p	$\neg d$	b	r	0
$\cdots$					

Burglar and racoon

Question: What is the probability of a burglar visit if police was called but trash can stayed untouched?

Normalize...

T	P	D	B	R	$p(d,b,r,p)$
$\neg t$	p	d	$\neg b$	$\neg r$	0.030096
$\neg t$	p	d	$\neg b$	r	0.01188
$\neg t$	p	d	b	$\neg r$	0.00304
$\neg t$	p	d	b	r	0.00024
$\neg t$	p	$\neg d$	$\neg b$	$\neg r$	0.0067716
$\neg t$	p	$\neg d$	$\neg b$	r	0.000297
$\neg t$	p	$\neg d$	b	$\neg r$	0
$\neg t$	p	$\neg d$	b	r	0
...					

Burglar and racoon

Question: What is the probability of a burglar visit if police was called but trash can stayed untouched?

Normalize...

T	P	D	B	R	$p(d,b,r,p)$
$\neg t$	p	d	$\neg b$	$\neg r$	0.57518
$\neg t$	p	d	$\neg b$	r	0.22704
$\neg t$	p	d	b	$\neg r$	0.058099
$\neg t$	p	d	b	r	0.0045868
$\neg t$	p	$\neg d$	$\neg b$	$\neg r$	0.12942
$\neg t$	p	$\neg d$	$\neg b$	r	0.0056761
$\neg t$	p	$\neg d$	b	$\neg r$	0
$\neg t$	p	$\neg d$	b	r	0
$\cdots$					

Burglar and racoon

Question: What is the probability of a burglar visit if police was called but trash can stayed untouched?

$$
\begin{aligned}
&p(b|\neg t, p) \\
=&0.058099 + 0.0045868 \\
\approx&0.0626
\end{aligned}
$$

T	P	D	B	R	$p(d,b,r,p)$
$\neg t$	p	d	$\neg b$	$\neg r$	0.57518
$\neg t$	p	d	$\neg b$	r	0.22704
$\neg t$	p	d	b	$\neg r$	0.058099
$\neg t$	p	d	b	r	0.0045868
$\neg t$	p	$\neg d$	$\neg b$	$\neg r$	0.12942
$\neg t$	p	$\neg d$	$\neg b$	r	0.0056761
$\neg t$	p	$\neg d$	b	$\neg r$	0
$\neg t$	p	$\neg d$	b	r	0
$\cdots$					

Belief Propagation Algorithm

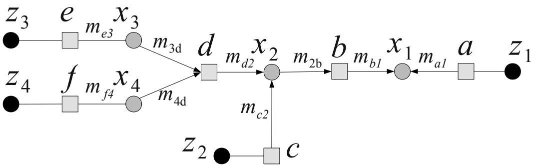
- It is also known to be the sum-product algorithm
- The goal of belief propagation is to efficiently compute the marginal distribution out of the joint distribution of multiple variables. This is essential for inferring the outcome of a particular variable with insufficient information
- The belief propagation algorithm is usually applied to problems modeled by a undirected graph (Markov random field) or a factor graph
- Rather than giving a rigorous proof of the algorithm, we will provide a simple example to illustrate the basic idea

Factor Graph

- A factor graph is a bipartite graph describing the correlation among several random variables. It generally contains two different types of nodes in the graph: variable nodes and factor nodes
- A variable node that is usually shown as circles corresponds to a random variable
- A factor node that is usually shown as a square connects variable nodes whose corresponding variables are immediately related

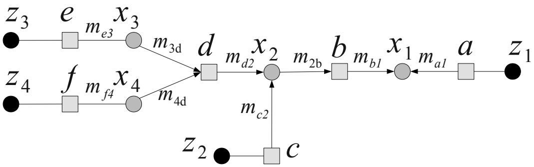
An Example

- A factor graph example is shown below. We have 8 *discrete* random variables, x_1^4 and z_1^4, depicted by 8 variable nodes



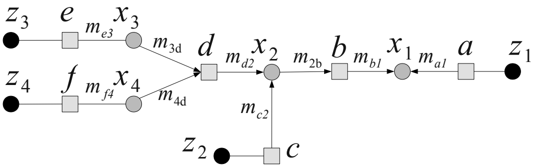
An Example

- A factor graph example is shown below. We have 8 *discrete* random variables, x_1^4 and z_1^4, depicted by 8 variable nodes
- Among the variable nodes, random variables x_1^4 (indicated by light circles) are unknown and variables z_1^4 (indicated by dark circles) are observed with known outcomes $\tilde{z}_1^4$



An Example

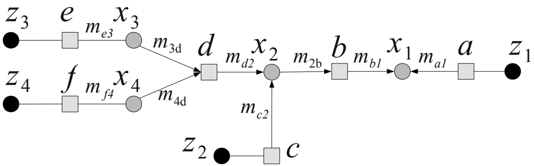
- A factor graph example is shown below. We have 8 *discrete* random variables, x_1^4 and z_1^4, depicted by 8 variable nodes
- Among the variable nodes, random variables x_1^4 (indicated by light circles) are unknown and variables z_1^4 (indicated by dark circles) are observed with known outcomes $\tilde{z}_1^4$
- The relationships among variables are captured entirely by the figure. For example, given x_1^4, z_1, z_2, z_3, and z_4 are conditional independent of each other. Moreover, (x_3, x_4) are conditional independent of x_1 given x_2



- The joint probability $p(x^4, z^4)$ of all variables can be decomposed into factor functions with subsets of all variables as arguments in the following

$$p(x^4, z^4) = p(x^4)p(z_1|x_1)p(z_2|x_2)p(z_3|x_3)p(z_4|x_4)$$

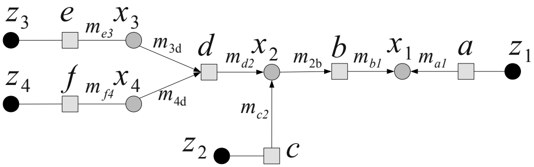
- Note that each factor function corresponds to a factor node in the factor graph.
- The arguments of the factor function correspond to the variable nodes that the factor node connects to.



- The joint probability $p(x^4, z^4)$ of all variables can be decomposed into factor functions with subsets of all variables as arguments in the following

$$\begin{aligned} p(x^4, z^4) &= p(x^4)p(z_1|x_1)p(z_2|x_2)p(z_3|x_3)p(z_4|x_4) \\ &= \underbrace{p(x_1, x_2)}_{f_b(x_1,x_2)}\underbrace{p(x_3, x_4|x_2)}_{f_d(x_2,x_3,x_4)}\underbrace{p(z_3|x_3)}_{f_e(x_3,z_3)}\underbrace{p(z_1|x_1)}_{f_a(x_1,z_1)}\underbrace{p(z_4|x_4)}_{f_f(x_4,z_4)}\underbrace{p(z_2|x_2)}_{f_c(x_2,z_2)} \end{aligned}$$

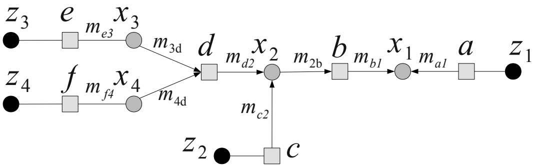
- Note that each factor function corresponds to a factor node in the factor graph.
- The arguments of the factor function correspond to the variable nodes that the factor node connects to.



- The joint probability $p(x^4, z^4)$ of all variables can be decomposed into factor functions with subsets of all variables as arguments in the following

$$\begin{aligned} p(x^4, z^4) &= p(x^4)p(z_1|x_1)p(z_2|x_2)p(z_3|x_3)p(z_4|x_4) \\ &= \underbrace{p(x_1, x_2)}_{f_b(x_1,x_2)}\underbrace{p(x_3, x_4|x_2)}_{f_d(x_2,x_3,x_4)}\underbrace{p(z_3|x_3)}_{f_e(x_3,z_3)}\underbrace{p(z_1|x_1)}_{f_a(x_1,z_1)}\underbrace{p(z_4|x_4)}_{f_f(x_4,z_4)}\underbrace{p(z_2|x_2)}_{f_c(x_2,z_2)} \\ &= f_b(x_1, x_2)f_d(x_2, x_3, x_4)f_e(x_3, z_3)f_a(x_1, z_1)f_f(x_4, z_4)f_c(x_2, z_2) \end{aligned}$$

- Note that each factor function corresponds to a factor node in the factor graph.
- The arguments of the factor function correspond to the variable nodes that the factor node connects to.



One common problem in probability inference is to estimate the value of a variable given incomplete information. For example, we may want to estimate x_1 given z^4 as $\tilde{z}^4$. The optimum estimate $\hat{x}_1$ will satisfy

$$\hat{x}_1 = \arg\max_{x_1} p(x_1|\tilde{z}^4) = \arg\max_{x_1} \frac{p(x_1, \tilde{z}^4)}{p(\tilde{z}^4)} = \arg\max_{x_1} p(x_1, \tilde{z}^4).$$

This requires us to compute the marginal distribution $p(x_1, \tilde{z}^4)$ out of the joint probability $p(x^4, \tilde{z}^4)$. Note that

$$p(x_1, \tilde{z}^4) = \sum_{x_2^4} p(x^4, \tilde{z}^4)$$

One common problem in probability inference is to estimate the value of a variable given incomplete information. For example, we may want to estimate x_1 given z^4 as $\tilde{z}^4$. The optimum estimate $\hat{x}_1$ will satisfy

$$\hat{x}_1 = \arg\max_{x_1} p(x_1|\tilde{z}^4) = \arg\max_{x_1} \frac{p(x_1, \tilde{z}^4)}{p(\tilde{z}^4)} = \arg\max_{x_1} p(x_1, \tilde{z}^4).$$

This requires us to compute the marginal distribution $p(x_1, \tilde{z}^4)$ out of the joint probability $p(x^4, \tilde{z}^4)$. Note that

$$p(x_1, \tilde{z}^4) = \sum_{x_2^4} p(x^4, \tilde{z}^4)$$

$$= \sum_{x_2^4} f_a(x_1, \tilde{z}_1) f_b(x_1, x_2) f_c(x_2, \tilde{z}_2) f_d(x_2, x_3, x_4) f_e(x_3, \tilde{z}_3) f_f(x_4, \tilde{z}_4)$$

One common problem in probability inference is to estimate the value of a variable given incomplete information. For example, we may want to estimate x_1 given z^4 as $\tilde{z}^4$. The optimum estimate $\hat{x}_1$ will satisfy

$$\hat{x}_1 = \arg\max_{x_1} p(x_1|\tilde{z}^4) = \arg\max_{x_1} \frac{p(x_1,\tilde{z}^4)}{p(\tilde{z}^4)} = \arg\max_{x_1} p(x_1,\tilde{z}^4).$$

This requires us to compute the marginal distribution $p(x_1,\tilde{z}^4)$ out of the joint probability $p(x^4,\tilde{z}^4)$. Note that

$$\begin{aligned} p(x_1,\tilde{z}^4) &= \sum_{x_2^4} p(x^4,\tilde{z}^4) \\ &= \sum_{x_2^4} f_a(x_1,\tilde{z}_1) f_b(x_1,x_2) f_c(x_2,\tilde{z}_2) f_d(x_2,x_3,x_4) f_e(x_3,\tilde{z}_3) f_f(x_4,\tilde{z}_4) \\ &= \underbrace{f_a(x_1,\tilde{z}_1)}_{m_{a1}} \underbrace{\sum_{x_2} f_b(x_1,x_2) \underbrace{\underbrace{f_c(x_2,\tilde{z}_2)}_{m_{c2}} \underbrace{\sum_{x_3,x_4} f_d(x_2,x_3,x_4) \underbrace{f_e(x_3,\tilde{z}_3)}_{m_{3d}} \underbrace{f_f(x_4,\tilde{z}_4)}_{m_{4d}}}_{m_{d2}}}_{m_{2b}}}_{m_{b1}} \end{aligned}$$

We can see from the last equation that the joint probability can be computed by combining a sequence of messages passing from a variable node i to a factor node a (m_{ia}) and vice versa (m_{ai}). More precisely, we can write

$$m_{a1}(x_1) \leftarrow f_a(x_1, \tilde{z}_1) = \sum_{z_1} f_a(x_1, z_1) \underbrace{p(z_1)}_{m_{1a}},$$

$$m_{c2}(x_2) \leftarrow f_c(x_2, \tilde{z}_2) = \sum_{z_2} f_c(x_2, z_2) \underbrace{p(z_2)}_{m_{2c}},$$

$$m_{e3}(x_3) \leftarrow f_e(x_3, \tilde{z}_3) = \sum_{z_3} f_e(x_3, z_3) \underbrace{p(z_3)}_{m_{3e}},$$

$$m_{f4}(x_4) \leftarrow f_f(x_4, \tilde{z}_4) = \sum_{z_4} f_f(x_4, z_4) \underbrace{p(z_4)}_{m_{4f}},$$

where $p(z_i) = \begin{cases} 1, & z_i = \tilde{z}_i \\ 0, & \text{otherwise} \end{cases}$

$$p(x_1, \tilde{z}^4) = \underbrace{f_a(x_1, \tilde{z}_1)}_{m_{a1}} \underbrace{\sum_{x_2} f_b(x_1, x_2) \underbrace{\underbrace{f_c(x_2, \tilde{z}_2)}_{m_{c2}} \underbrace{\sum_{x_3, x_4} f_d(x_2, x_3, x_4) \underbrace{f_e(x_3, \tilde{z}_3)}_{m_{3d}} \underbrace{f_f(x_4, \tilde{z}_4)}_{m_{4d}}}_{m_{d2}}}_{m_{2b}}}_{m_{b1}} \tag{1}$$

$$m_{3d}(x_3) \leftarrow m_{e3}(x_3) = f_e(x_3, \tilde{z}_3),$$
$$m_{4d}(x_4) \leftarrow m_{f4}(x_4) = f_f(x_4, \tilde{z}_4),$$

$$p(x_1, \tilde{z}^4) = \underbrace{f_a(x_1, \tilde{z}_1)}_{m_{a1}} \underbrace{\sum_{x_2} f_b(x_1, x_2) \underbrace{\underbrace{f_c(x_2, \tilde{z}_2)}_{m_{c2}} \underbrace{\sum_{x_3, x_4} f_d(x_2, x_3, x_4) \underbrace{f_e(x_3, \tilde{z}_3)}_{\color{red}{m_{3d}}} \underbrace{f_f(x_4, \tilde{z}_4)}_{\color{red}{m_{4d}}}}_{m_{d2}}}_{m_{2b}}}_{m_{b1}} \tag{1}$$

$$
\begin{aligned}
m_{3d}(x_3) &\leftarrow m_{e3}(x_3) = f_e(x_3, \tilde{z}_3), \\
m_{4d}(x_4) &\leftarrow m_{f4}(x_4) = f_f(x_4, \tilde{z}_4), \\
m_{d2}(x_2) &\leftarrow \sum_{x_3, x_4} f_d(x_2, x_3, x_4) m_{3d}(x_3) m_{4d}(x_4),
\end{aligned}
$$

$$
p(x_1, \tilde{z}^4) = \underbrace{f_a(x_1, \tilde{z}_1)}_{m_{a1}} \underbrace{\sum_{x_2} f_b(x_1, x_2) \underbrace{\underbrace{f_c(x_2, \tilde{z}_2)}_{m_{c2}} \underbrace{\sum_{x_3, x_4} f_d(x_2, x_3, x_4) \underbrace{f_e(x_3, \tilde{z}_3)}_{m_{3d}} \underbrace{f_f(x_4, \tilde{z}_4)}_{m_{4d}}}_{\color{red}{m_{d2}}}}_{m_{2b}}}_{m_{b1}} \qquad (1)
$$

$$
\begin{aligned}
m_{3d}(x_3) &\leftarrow m_{e3}(x_3) = f_e(x_3, \tilde{z}_3),\\
m_{4d}(x_4) &\leftarrow m_{f4}(x_4) = f_f(x_4, \tilde{z}_4),\\
m_{d2}(x_2) &\leftarrow \sum_{x_3, x_4} f_d(x_2, x_3, x_4) m_{3d}(x_3) m_{4d}(x_4),\\
m_{2b}(x_2) &\leftarrow m_{c2}(x_2) m_{d2}(x_2),
\end{aligned}
$$

$$
p(x_1, \tilde{z}^4) = \underbrace{f_a(x_1, \tilde{z}_1)}_{m_{a1}} \underbrace{\sum_{x_2} f_b(x_1, x_2) \underbrace{\underbrace{f_c(x_2, \tilde{z}_2)}_{m_{c2}} \underbrace{\sum_{x_3, x_4} f_d(x_2, x_3, x_4) \underbrace{f_e(x_3, \tilde{z}_3)}_{m_{3d}} \underbrace{f_f(x_4, \tilde{z}_4)}_{m_{4d}}}_{m_{d2}}}_{\color{red}{m_{2b}}}}_{m_{b1}} \tag{1}
$$

$$\begin{aligned}
m_{3d}(x_3) &\leftarrow m_{e3}(x_3) = f_e(x_3, \tilde{z}_3), \\
m_{4d}(x_4) &\leftarrow m_{f4}(x_4) = f_f(x_4, \tilde{z}_4), \\
m_{d2}(x_2) &\leftarrow \sum_{x_3, x_4} f_d(x_2, x_3, x_4) m_{3d}(x_3) m_{4d}(x_4), \\
m_{2b}(x_2) &\leftarrow m_{c2}(x_2) m_{d2}(x_2), \\
m_{b1}(x_1) &\leftarrow \sum_{x_2} f_b(x_1, x_2) m_{2b}(x_2),
\end{aligned}$$

$$p(x_1, \tilde{z}^4) = \underbrace{f_a(x_1, \tilde{z}_1)}_{m_{a1}} \underbrace{\sum_{x_2} f_b(x_1, x_2) \underbrace{\underbrace{f_c(x_2, \tilde{z}_2)}_{m_{c2}} \underbrace{\sum_{x_3, x_4} f_d(x_2, x_3, x_4) \underbrace{f_e(x_3, \tilde{z}_3)}_{m_{3d}} \underbrace{f_f(x_4, \tilde{z}_4)}_{m_{4d}}}_{m_{d2}}}_{m_{2b}}}_{m_{b1}} \tag{1}$$

$$
\begin{aligned}
m_{3d}(x_3) &\leftarrow m_{e3}(x_3) = f_e(x_3, \tilde{z}_3), \\
m_{4d}(x_4) &\leftarrow m_{f4}(x_4) = f_f(x_4, \tilde{z}_4), \\
m_{d2}(x_2) &\leftarrow \sum_{x_3, x_4} f_d(x_2, x_3, x_4) m_{3d}(x_3) m_{4d}(x_4), \\
m_{2b}(x_2) &\leftarrow m_{c2}(x_2) m_{d2}(x_2), \\
m_{b1}(x_1) &\leftarrow \sum_{x_2} f_b(x_1, x_2) m_{2b}(x_2), \\
p(x_1, \tilde{z}^4) &\leftarrow m_{a1}(x_1) m_{b1}(x_1),
\end{aligned}
$$

$$
p(x_1, \tilde{z}^4) = \underbrace{f_a(x_1, \tilde{z}_1)}_{m_{a1}} \underbrace{\sum_{x_2} f_b(x_1, x_2) \underbrace{\underbrace{f_c(x_2, \tilde{z}_2)}_{m_{c2}} \underbrace{\sum_{x_3, x_4} f_d(x_2, x_3, x_4) \underbrace{f_e(x_3, \tilde{z}_3)}_{m_{3d}} \underbrace{f_f(x_4, \tilde{z}_4)}_{m_{4d}}}_{m_{d2}}}_{m_{2b}}}_{m_{b1}} \tag{1}
$$

Belief propagation algorithm

- **Initialization**: For any variable node i, if the prior probability of x_i is known and equal to $p(x_i)$, for $a \in N(i)$,
- **Message passing**:
- **Belief update**:
- **Stopping criteria**: repeat message update and/or belief update until the algorithm stops when maximum number of iterations is reached or some other conditions are satisfied.

Belief propagation algorithm

- **Initialization**: For any variable node i, if the prior probability of x_i is known and equal to $p(x_i)$, for $a \in N(i)$,

$$m_{ia}(x_i) \leftarrow p(x_i)$$

- **Message passing**:

- **Belief update**:

- **Stopping criteria**: repeat message update and/or belief update until the algorithm stops when maximum number of iterations is reached or some other conditions are satisfied.

Belief propagation algorithm

- **Initialization**: For any variable node i, if the prior probability of x_i is known and equal to $p(x_i)$, for $a \in N(i)$,

$$m_{ia}(x_i) \leftarrow p(x_i)$$

- **Message passing**:

$$m_{ia}(x_i) \leftarrow \prod_{b \in N(i) \backslash a} m_{bi}(x_i),$$

$$m_{ai}(x_i) \leftarrow \sum_{\mathbf{x}_a} f_a(\mathbf{x}_a) \prod_{j \in N(a) \backslash i} m_{ja}(x_j) \qquad \text{("sum-product")}$$

- **Belief update**:

- **Stopping criteria**: repeat message update and/or belief update until the algorithm stops when maximum number of iterations is reached or some other conditions are satisfied.

Belief propagation algorithm

- **Initialization**: For any variable node i, if the prior probability of x_i is known and equal to $p(x_i)$, for $a \in N(i)$,

$$m_{ia}(x_i) \leftarrow p(x_i)$$

- **Message passing**:

$$m_{ia}(x_i) \leftarrow \prod_{b \in N(i) \backslash a} m_{bi}(x_i),$$

$$m_{ai}(x_i) \leftarrow \sum_{\mathbf{x}_a} f_a(\mathbf{x}_a) \prod_{j \in N(a) \backslash i} m_{ja}(x_j) \qquad (\text{``sum-product''})$$

- **Belief update**:

$$\beta_i(x_i) \leftarrow \prod_{a \in N(i)} m_{ai}(x_i)$$

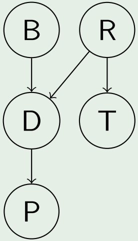
- **Stopping criteria**: repeat message update and/or belief update until the algorithm stops when maximum number of iterations is reached or some other conditions are satisfied.

Remark

- We have not assumed the precise phyical meanings of the factor functions themselves. The only assumption we made is that the joint probability can be decomposed into the factor functions and apparently this decomposition is not unique
- The belief propagation algorithm as shown above is exact only because the corresponding graph is a tree and has no loop. If loop exists, the algorithm is not exact and generally the final belief may not even converge
- While the result is no longer exact, applying BP algorithm for general graphs (sometimes refer to as loopy BP) works well in many applications such as LDPC decoding

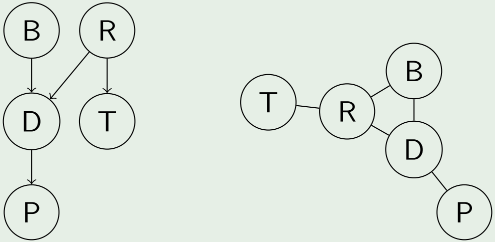
Burglar and racoon revisit

Question: What is the probability of a burglar visit if police was called but trash can stayed untouched?



Burglar and racoon revisit

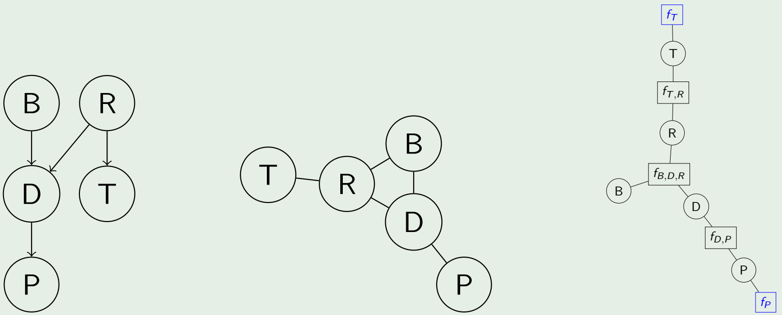
Question: What is the probability of a burglar visit if police was called but trash can stayed untouched?



Moralization...

Burglar and racoon revisit

Question: What is the probability of a burglar visit if police was called but trash can stayed untouched?



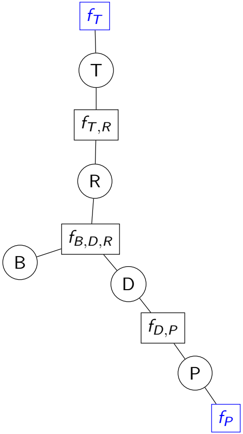
Convert to factor graph..

Using belief propagation...

$$\begin{cases} f_P(p) & = 1 \\ f_P(\neg p) & = 0 \end{cases}$$

$$\begin{cases} f_T(t) & = 0 \\ f_T(\neg t) & = 1 \end{cases}$$

$$f_{B,D,R}(b,d,r) = p(b,d,r)$$
$$f_{T,R}(t,r) = p(t|r)$$
$$f_{D,P}(d,p) = p(p|d)$$



Some History of LDPC Codes

- Before 1990's, the strategy for channel code has always been looking for codes that can be decoded optimally. This leads to a wide range of so-called algebraic codes. It turns out the "optimally-decodable" codes are usually poor codes
- Until early 1990's, researchers had basically agreed that the Shannon capacity was restricted to theoretical interest and could hardly be reached in practice
- The introduction of turbo codes gave a huge shock to the research community. The community were so dubious about the amazing performance of turbo codes that they did not accept the finding initially until independent researchers had verified the results
- The low-density parity-check (LDPC) codes were later rediscovered and both LDPC codes and turbo codes are based on the same philosophy differs from codes in the past. Instead of designing and using codes that can be decoded "optimally", let us just pick some *random* codes and perform decoding "sub-optimally"

LDPC Codes

- As its name suggests, LDPC codes refer to codes that with sparse (low-density) parity check matrices. In other words, there are only few ones in a parity check matrix and the rest are all zeros

LDPC Codes

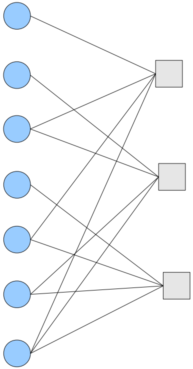
- As its name suggests, LDPC codes refer to codes that with sparse (low-density) parity check matrices. In other words, there are only few ones in a parity check matrix and the rest are all zeros
- We learn from the proof of Channel Coding Theorem that random code is asymptotically optimum. This suggests that if we just generate a code randomly with a very long code length. It is likely that we will get a very good code.

LDPC Codes

- As its name suggests, LDPC codes refer to codes that with sparse (low-density) parity check matrices. In other words, there are only few ones in a parity check matrix and the rest are all zeros
- We learn from the proof of Channel Coding Theorem that random code is asymptotically optimum. This suggests that if we just generate a code randomly with a very long code length. It is likely that we will get a very good code.
- The problem is: how do we perform decoding? Due to the lack of structure of a random code, tricks that enable fast decoding for structured algebraic codes that were widely used before 1990's are unrealizable here
- Solution: Belief propagation!

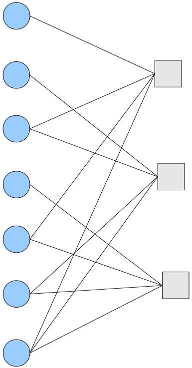
Tanner Graph

- An LDPC code can be represented using a Tanner graph as shown on the right



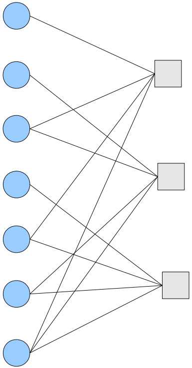
Tanner Graph

- An LDPC code can be represented using a Tanner graph as shown on the right
- Each circle x_i represents a code bit sent to the decoder



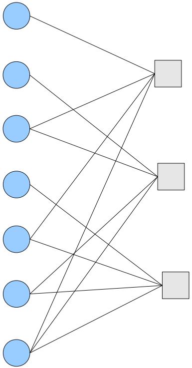
Tanner Graph

- An LDPC code can be represented using a Tanner graph as shown on the right
- Each circle x_i represents a code bit sent to the decoder
- Each square represents a check bit with value equal to the sum of code bit connecting to it



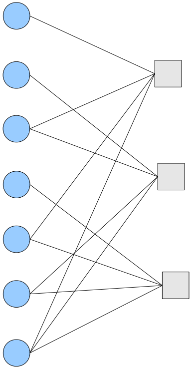
Tanner Graph

- An LDPC code can be represented using a Tanner graph as shown on the right
- Each circle x_i represents a code bit sent to the decoder
- Each square represents a check bit with value equal to the sum of code bit connecting to it
- The vector $x_1, x_2, \cdots, x_N$ is a codeword only if all checks are zero



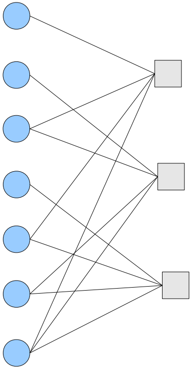
Tanner Graph

- An LDPC code can be represented using a Tanner graph as shown on the right
- Each circle x_i represents a code bit sent to the decoder
- Each square represents a check bit with value equal to the sum of code bit connecting to it
- The vector $x_1, x_2, \cdots, x_N$ is a codeword only if all checks are zero
- By default, the mapping between a codeword to the actual message is non-trivial for an LDPC code



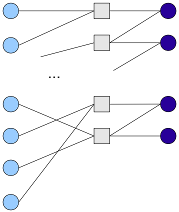
Tanner Graph

- An LDPC code can be represented using a Tanner graph as shown on the right
- Each circle x_i represents a code bit sent to the decoder
- Each square represents a check bit with value equal to the sum of code bit connecting to it
- The vector $x_1, x_2, \cdots, x_N$ is a codeword only if all checks are zero
- By default, the mapping between a codeword to the actual message is non-trivial for an LDPC code
- It would be great if the actual message is included in the codeword. That is, some of the bits in the codeword spell out the actual message $\Rightarrow$ IRA codes



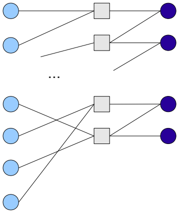
IRA Codes

- Irregular repeated accumulate (IRA) code a type of systematic LDPC code, i.e., each codeword can be partitioned into message bits and syndrome bits



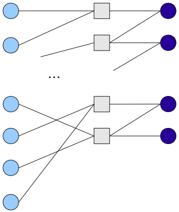
IRA Codes

- Irregular repeated accumulate (IRA) code a type of systematic LDPC code, i.e., each codeword can be partitioned into message bits and syndrome bits
- As shown on the right, light blue circles correspond to the input message bits and the dark blue circle correspond to the syndrome bits



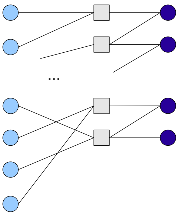
IRA Codes

- Irregular repeated accumulate (IRA) code a type of systematic LDPC code, i.e., each codeword can be partitioned into message bits and syndrome bits
- As shown on the right, light blue circles correspond to the input message bits and the dark blue circle correspond to the syndrome bits
- To ensure the top check bit is satisfied, the top syndrome bit will be set to be the sum of message bits connecting to the check



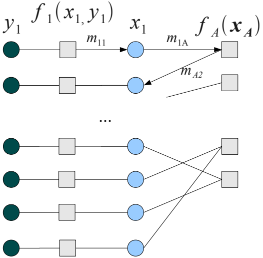
IRA Codes

- Irregular repeated accumulate (IRA) code a type of systematic LDPC code, i.e., each codeword can be partitioned into message bits and syndrome bits
- As shown on the right, light blue circles correspond to the input message bits and the dark blue circle correspond to the syndrome bits
- To ensure the top check bit is satisfied, the top syndrome bit will be set to be the sum of message bits connecting to the check
- The computed syndrome bit will then pass to the next check and again we can ensure the next check bit is satisfied by setting that second syndrome bit as the sum of message bits conecting to the check + *last syndrome bit*. All (dark blue) syndrome bits can be assigned in similar token



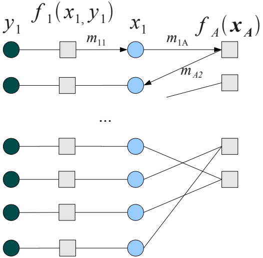
LDPC Decoding

- $x_1, \cdots, x_N$ (light blue): transmitted bits
- $y_1, \cdots, y_N$ (dark grey): received bits



LDPC Decoding

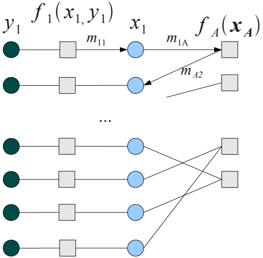
- $x_1, \cdots, x_N$ (light blue): transmitted bits
- $y_1, \cdots, y_N$ (dark grey): received bits
- $p(x^N, y^N) = \prod_i \underbrace{p(y_i|x_i)}_{f_i(x_i,y_i)} \underbrace{p(x^N)}_{\prod_A f_A(\mathbf{x}_A)}$



LDPC Decoding

- $x_1, \cdots, x_N$ (light blue): transmitted bits
- $y_1, \cdots, y_N$ (dark grey): received bits
- $p(x^N, y^N) = \prod_i \underbrace{p(y_i|x_i)}_{f_i(x_i,y_i)} \underbrace{p(x^N)}_{\prod_A f_A(\mathbf{x}_A)}$
- $f_i(x_i, y_i) = p(y_i|x_i)$ and

$$f_A(\mathbf{x}) = \begin{cases} 0, & \mathbf{x} \text{ contains even number of 1,} \\ 1, & \mathbf{x} \text{ contains odd number of 1.} \end{cases}$$



Variable Node Update

- Since the unknown variables are binary, it is more convenient to represent the messages using likelihood or log-likelihood ratios. Define

$$l_{ai} \triangleq \frac{m_{ai}(0)}{m_{ai}(1)}, \qquad L_{ai} \triangleq \log l_{ai} \tag{2}$$

  and

$$l_{ia} \triangleq \frac{m_{ia}(0)}{m_{ia}(1)}, \qquad L_{ia} \triangleq \log l_{ia} \tag{3}$$

  for any variable node i and factor node a.

- Then,

$$L_{ia} \leftarrow \sum_{b \in N(i) \backslash i} L_{ai}. \tag{4}$$

Check Node Update

- Assuming that we have three variable nodes 1,2, and 3 connecting to the check node *a*, then the check to variable node updates become

$$m_{a1}(1) \leftarrow m_{2a}(1)m_{3a}(0) + m_{2a}(0)m_{3a}(1) \tag{5}$$

$$m_{a1}(0) \leftarrow m_{2a}(0)m_{3a}(0) + m_{2a}(1)m_{3a}(1) \tag{6}$$

- Substitute in the likelihood ratios and log-likelihood ratios, we have

$$l_{a1} \triangleq \frac{m_{a1}(0)}{m_{a1}(1)} \leftarrow \frac{1 + l_{2a}l_{3a}}{l_{2a} + l_{3a}} \tag{7}$$

and

$$e^{L_{a1}} = l_{a1} \leftarrow \frac{1 + e^{L_{2a}}e^{L_{3a}}}{e^{L_{2a}} + e^{L_{3a}}}. \tag{8}$$

- Note that

$$\tanh\left(\frac{L_{a1}}{2}\right) = \frac{e^{\frac{L_{a1}}{2}} - e^{-\frac{L_{a1}}{2}}}{e^{\frac{L_{a1}}{2}} + e^{-\frac{L_{a1}}{2}}} = \frac{e^{L_{a1}} - 1}{e^{L_{a1}} + 1} \tag{9}$$

$$\leftarrow \frac{1 + e^{L_{2a}}e^{L_{3a}} - e^{L_{2a}} - e^{L_{3a}}}{1 + e^{L_{2a}}e^{L_{3a}} + e^{L_{2a}} + e^{L_{3a}}} \tag{10}$$

$$= \frac{(e^{L_{2a}} - 1)(e^{L_{3a}} - 1)}{(e^{L_{2a}} + 1)(e^{L_{3a}} + 1)} \tag{11}$$

$$= \tanh\left(\frac{L_{2a}}{2}\right)\tanh\left(\frac{L_{3a}}{2}\right). \tag{12}$$

- When we have more than 3 variable nodes connecting to the check node a, it is easy to show using induction that

$$\tanh\left(\frac{L_{ai}}{2}\right) \leftarrow \prod_{j\in N(a)\backslash i} \tanh\left(\frac{L_{ja}}{2}\right). \tag{13}$$

Method of Type

Motivation

- In previous lectures, we have introduced LLN and typical sequences. In a sense that every sequences drawn from a discrete memoryless source is typical

Motivation

- In previous lectures, we have introduced LLN and typical sequences. In a sense that every sequences drawn from a discrete memoryless source is typical
- Take coin tossing as example again, if $Pr(Head) = 0.6$, and we throw the coin 1000 times. We expect that almost all drawn sequences with have about 600 heads. And the rest have negligible probability

Motivation

- In previous lectures, we have introduced LLN and typical sequences. In a sense that every sequences drawn from a discrete memoryless source is typical
- Take coin tossing as example again, if $Pr(Head) = 0.6$, and we throw the coin 1000 times. We expect that almost all drawn sequences with have about 600 heads. And the rest have negligible probability
- However, sometimes we are interested in the probability of getting say 400 heads, even though we know that the probability is negligible

Motivation

- In previous lectures, we have introduced LLN and typical sequences. In a sense that every sequences drawn from a discrete memoryless source is typical
- Take coin tossing as example again, if $Pr(Head) = 0.6$, and we throw the coin 1000 times. We expect that almost all drawn sequences with have about 600 heads. And the rest have negligible probability
- However, sometimes we are interested in the probability of getting say 400 heads, even though we know that the probability is negligible → method of types

Motivation

By the end of the class, we will be able to solve the following nontrivial puzzle

- Tom throws a unbiased dice for 10,000 times and adds all values

Motivation

By the end of the class, we will be able to solve the following nontrivial puzzle

- Tom throws a unbiased dice for 10,000 times and adds all values
- For whatever reason, he is not happy until the sum is at least 40,000. If not, he will just throw the dice again for 10,000

Motivation

By the end of the class, we will be able to solve the following nontrivial puzzle

- Tom throws a unbiased dice for 10,000 times and adds all values
- For whatever reason, he is not happy until the sum is at least 40,000. If not, he will just throw the dice again for 10,000
- Now, by the time he eventually got a sequence with sum at least 40,000, *approximately how many ones in the sequence?*

Type class

Continue with the coin-tossing example

- Recall that the probability of getting a particular sequence with 600 heads is
$$0.6^{600}0.4^{400}$$

Type class

Continue with the coin-tossing example

- Recall that the probability of getting a particular sequence with 600 heads is
$$0.6^{600}0.4^{400} = 2^{-1000(-0.6\log 0.6-0.4\log 0.4)}$$

Type class

Continue with the coin-tossing example

- Recall that the probability of getting a particular sequence with 600 heads is $$0.6^{600}0.4^{400} = 2^{-1000(-0.6\log 0.6 - 0.4\log 0.4)} = 2^{-NH(X)}$$

Type class

Continue with the coin-tossing example

- Recall that the probability of getting a particular sequence with 600 heads is
$$0.6^{600}0.4^{400} = 2^{-1000(-0.6\log 0.6 - 0.4\log 0.4)} = 2^{-NH(X)}$$
- How about the probability of getting a particular sequence with 400 heads? It is
$$0.6^{400}0.4^{600} = 2^{-1000(-0.4\log 0.6 - 0.6\log 0.4)}$$

Type class

Continue with the coin-tossing example

- Recall that the probability of getting a particular sequence with 600 heads is
$$0.6^{600}0.4^{400}=2^{-1000(-0.6\log 0.6-0.4\log 0.4)}=2^{-NH(X)}$$
- How about the probability of getting a particular sequence with 400 heads? It is

$$\begin{aligned}0.6^{400}0.4^{600}&=2^{-1000(-0.4\log 0.6-0.6\log 0.4)}\\&=2^{-1000(-0.4\log 0.4-0.6\log 0.6+0.4\log\frac{0.4}{0.6}+0.6\log\frac{0.6}{0.4})}\end{aligned}$$

Type class

Continue with the coin-tossing example

- Recall that the probability of getting a particular sequence with 600 heads is
$$0.6^{600}0.4^{400}=2^{-1000(-0.6\log 0.6-0.4\log 0.4)}=2^{-NH(X)}$$
- How about the probability of getting a particular sequence with 400 heads? It is

$$\begin{aligned}0.6^{400}0.4^{600}&=2^{-1000(-0.4\log 0.6-0.6\log 0.4)}\\&=2^{-1000(-0.4\log 0.4-0.6\log 0.6+0.4\log\frac{0.4}{0.6}+0.6\log\frac{0.6}{0.4})}\\&=2^{-N(H(X)+KL((0.4,0.6)||(0.6,0.4))}\end{aligned}$$

Type class

Continue with the coin-tossing example

- Recall that the probability of getting a particular sequence with 600 heads is
$$0.6^{600}0.4^{400}=2^{-1000(-0.6\log 0.6-0.4\log 0.4)}=2^{-NH(X)}$$
- How about the probability of getting a particular sequence with 400 heads? It is
$$\begin{aligned}0.6^{400}0.4^{600}&=2^{-1000(-0.4\log 0.6-0.6\log 0.4)}\\&=2^{-1000(-0.4\log 0.4-0.6\log 0.6+0.4\log\frac{0.4}{0.6}+0.6\log\frac{0.6}{0.4})}\\&=2^{-N(H(X)+KL((0.4,0.6)||(0.6,0.4))}\end{aligned}$$
- Every sequence with 400 heads has the same probability. And in general, sequences with the same fraction of outcomes have same probability and we can put them into the same (type) class

Type class

- For convenience, let us denote the number of a in the sequence x^N as $\mathscr{N}(a|x^N)$

Type class

- For convenience, let us denote the number of a in the sequence x^N as $\mathscr{N}(a|x^N)$
- Then for any valid distribution of X, $p(x)$, we will define a type class $T(p_X)$ as the set containing all sequences such that $\frac{\mathscr{N}(a|x^N)}{N} \approx p(a), \forall a \in \mathcal{X}$

Type class

- For convenience, let us denote the number of a in the sequence x^N as $\mathscr{N}(a|x^N)$
- Then for any valid distribution of X, $p(x)$, we will define a type class $T(p_X)$ as the set containing all sequences such that $\frac{\mathscr{N}(a|x^N)}{N} \approx p(a), \forall a \in \mathcal{X}$
- Let us reserve $q(x)$ as the true distribution of x (i.e., $q(Head) = 0.6$ and $q(Tail) = 0.4$). And in general, we expect all sequences drawn from the source should belongs to $T(q)$ asymptotically

Type class

- For convenience, let us denote the number of a in the sequence x^N as $\mathscr{N}(a|x^N)$
- Then for any valid distribution of X, $p(x)$, we will define a type class $T(p_X)$ as the set containing all sequences such that $\frac{\mathscr{N}(a|x^N)}{N} \approx p(a)$, $\forall a \in \mathcal{X}$
- Let us reserve $q(x)$ as the true distribution of x (i.e., $q(Head) = 0.6$ and $q(Tail) = 0.4$). And in general, we expect all sequences drawn from the source should belongs to $T(q)$ asymptotically
- Let's also refer p_{x^N} as the empirical distribution of x^N. That is $p_{x^N}(a) = \frac{\mathscr{N}(a|x^N)}{N}$. So $T(p_{x^N})$ is the type class containing x^N

Example

Let $\mathcal{X} \in \{1,2,3\}$ and $x^N = 11321$

- $p_{x^N}(1) = \frac{3}{5}$,

Example

Let $\mathcal{X} \in \{1, 2, 3\}$ and $x^N = 11321$

- $p_{x^N}(1) = \frac{3}{5}$, $p_{x^N}(2) = \frac{1}{5}$, $p_{x^N}(3) = \frac{1}{5}$

Example

Let $\mathcal{X} \in \{1, 2, 3\}$ and $x^N = 11321$

- $p_{x^N}(1) = \frac{3}{5}$, $p_{x^N}(2) = \frac{1}{5}$, $p_{x^N}(3) = \frac{1}{5}$
- $T(p_{x^N}) = \{11123, 11132, 11231, 11321, \cdots\}$ containing all sequences with three 1's, one 2, and one 3

Example

Let $\mathcal{X} \in \{1,2,3\}$ and $x^N = 11321$

- $p_{x^N}(1) = \frac{3}{5}$, $p_{x^N}(2) = \frac{1}{5}$, $p_{x^N}(3) = \frac{1}{5}$
- $T(p_{x^N}) = \{11123, 11132, 11231, 11321, \cdots\}$ containing all sequences with three 1's, one 2, and one 3
- $|T(p_{x^N})| = \frac{5!}{3!1!1!} = 20$.

Example

Let $\mathcal{X} \in \{1,2,3\}$ and $x^N = 11321$

- $p_{x^N}(1) = \frac{3}{5}$, $p_{x^N}(2) = \frac{1}{5}$, $p_{x^N}(3) = \frac{1}{5}$
- $T(p_{x^N}) = \{11123, 11132, 11231, 11321, \cdots\}$ containing all sequences with three 1's, one 2, and one 3
- $|T(p_{x^N})| = \frac{5!}{3!1!1!} = 20$. In general,

$$|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$$

Example

Let $\mathcal{X} \in \{1, 2, 3\}$ and $x^N = 11321$

- $p_{x^N}(1) = \frac{3}{5}$, $p_{x^N}(2) = \frac{1}{5}$, $p_{x^N}(3) = \frac{1}{5}$
- $T(p_{x^N}) = \{11123, 11132, 11231, 11321, \cdots\}$ containing all sequences with three 1's, one 2, and one 3
- $|T(p_{x^N})| = \frac{5!}{3!1!1!} = 20$. In general,

$$|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))! \cdots}$$

Actually we don't care too much what $|T(p)|$ is exactly. We will provide bounds for $|T(p)|$ as we come back later on

Example

Let $\mathcal{X} \in \{1,2,3\}$ and $x^N = 11321$

- $p_{x^N}(1) = \frac{3}{5}$, $p_{x^N}(2) = \frac{1}{5}$, $p_{x^N}(3) = \frac{1}{5}$
- $T(p_{x^N}) = \{11123, 11132, 11231, 11321, \cdots\}$ containing all sequences with three 1's, one 2, and one 3
- $|T(p_{x^N})| = \frac{5!}{3!1!1!} = 20$. In general,

$$|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$$

  Actually we don't care too much what $|T(p)|$ is exactly. We will provide bounds for $|T(p)|$ as we come back later on

- And for any sequence $\mathbf{y}$ in $T(p_{x^N})$, $p(\mathbf{y}) = q(1)^3 q(2) q(3)$, where $q(\cdot)$ is the true distribution

Type sequence probability

Even though we have seen that in the coin toss example, let's restate it more formally.

Theorem 1

If $x^N \in T(p)$ and $q(\cdot)$ is the true distribution of X, the probability of getting x^N from sampling $q(\cdot)$ for N times, as denoted as $q^N(x^N)$, is given by

$$2^{-N(H(p)+KL(p||q))}$$

Type sequence probability

Even though we have seen that in the coin toss example, let's restate it more formally.

Theorem 1

If $x^N \in T(p)$ and $q(\cdot)$ is the true distribution of X, the probability of getting x^N from sampling $q(\cdot)$ for N times, as denoted as $q^N(x^N)$, is given by

$$2^{-N(H(p)+KL(p||q))}$$

Proof

$$q^N(x^N) = \prod_{i=1}^{N} q(x_i) = 2^{\sum_{i=1}^{N} \log q(x_i)}$$

Type sequence probability

Even though we have seen that in the coin toss example, let's restate it more formally.

Theorem 1

If $x^N \in T(p)$ and $q(\cdot)$ is the true distribution of X, the probability of getting x^N from sampling $q(\cdot)$ for N times, as denoted as $q^N(x^N)$, is given by

$$2^{-N(H(p)+KL(p||q))}$$

Proof

$$q^N(x^N) = \prod_{i=1}^{N} q(x_i) = 2^{\sum_{i=1}^{N} \log q(x_i)} = 2^{\sum_{a\in\mathcal{X}} \mathscr{N}(a|x^N) \log q(a)}$$

Type sequence probability

Even though we have seen that in the coin toss example, let's restate it more formally.

Theorem 1

If $x^N \in T(p)$ and $q(\cdot)$ is the true distribution of X, the probability of getting x^N from sampling $q(\cdot)$ for N times, as denoted as $q^N(x^N)$, is given by

$$2^{-N(H(p)+KL(p||q))}$$

Proof

$$q^N(x^N) = \prod_{i=1}^{N} q(x_i) = 2^{\sum_{i=1}^{N} \log q(x_i)} = 2^{\sum_{a\in\mathcal{X}} \mathcal{N}(a|x^N) \log q(a)}$$

$$= 2^{-N \sum_{a\in\mathcal{X}} -p_{x_N}(a) \log q(a)}$$

Type sequence probability

Even though we have seen that in the coin toss example, let's restate it more formally.

Theorem 1

If $x^N \in T(p)$ and $q(\cdot)$ is the true distribution of X, the probability of getting x^N from sampling $q(\cdot)$ for N times, as denoted as $q^N(x^N)$, is given by

$$2^{-N(H(p)+KL(p||q))}$$

Proof

$$q^N(x^N) = \prod_{i=1}^{N} q(x_i) = 2^{\sum_{i=1}^{N} \log q(x_i)} = 2^{\sum_{a\in\mathcal{X}} \mathcal{N}(a|x^N) \log q(a)}$$

$$= 2^{-N \sum_{a\in\mathcal{X}} -p_{x_N}(a) \log q(a)} = 2^{-N\left(-\sum_{a\in\mathcal{X}} p(a) \log p(a) - \sum_{a\in\mathcal{X}} p(a) \log \frac{p(a)}{q(a)}\right)}$$

Type sequence probability

Even though we have seen that in the coin toss example, let's restate it more formally.

Theorem 1

If $x^N \in T(p)$ and $q(\cdot)$ is the true distribution of X, the probability of getting x^N from sampling $q(\cdot)$ for N times, as denoted as $q^N(x^N)$, is given by

$$2^{-N(H(p)+KL(p||q))}$$

Proof

$$\begin{aligned} q^N(x^N) &= \prod_{i=1}^N q(x_i) = 2^{\sum_{i=1}^N \log q(x_i)} = 2^{\sum_{a\in\mathcal{X}} \mathcal{N}(a|x^N) \log q(a)} \\ &= 2^{-N\sum_{a\in\mathcal{X}} -p_{x_N}(a)\log q(a)} = 2^{-N\left(-\sum_{a\in\mathcal{X}} p(a)\log p(a) - \sum_{a\in\mathcal{X}} p(a)\log\frac{p(a)}{q(a)}\right)} \\ &= 2^{-N(H(p)+KL(p||q))} \end{aligned}$$

Probability of a sequence in the “typical” class

If $x^N \in T(q)$, where $q(\cdot)$ is the true distribution of X, then

$$q^N(x^N) = 2^{-NH(q)} = 2^{-NH(X)}$$

Probability of a sequence in the “typical” class

If $x^N \in T(q)$, where $q(\cdot)$ is the true distribution of X, then

$$q^N(x^N) = 2^{-NH(q)} = 2^{-NH(X)}$$

Remarks

- Note that the probability is exactly equal to $2^{-NH(X)}$

Probability of a sequence in the “typical” class

If $x^N \in T(q)$, where $q(\cdot)$ is the true distribution of X, then

$$q^N(x^N) = 2^{-NH(q)} = 2^{-NH(X)}$$

Remarks

- Note that the probability is exactly equal to $2^{-NH(X)}$
- Recall that this is the probability of a typical sequence supposed to be. Therefore, any x^N in $T(q)$ is a typical sequence ($T(q) \subset A_\epsilon^N(X)$)

Set of all empirical distribution $\mathcal{P}_N(X)$

Denote $\mathcal{P}_N(X)$ as the set of all empirical distribution of X in a length-N sequence

Set of all empirical distribution $\mathcal{P}_N(X)$

Denote $\mathcal{P}_N(X)$ as the set of all empirical distribution of X in a length-N sequence

Example

If $X \in \{0, 1\}$,

$$\mathcal{P}_N(X) = \left\{ (p_X(0), p_X(1)) : \left(\frac{0}{N}, \frac{N}{N} \right), \left(\frac{1}{N}, \frac{N-1}{N} \right), \cdots, \left(\frac{N}{N}, \frac{0}{N} \right) \right\}$$

Note that $|\mathcal{P}_N(X)| = N + 1$

Set of all empirical distribution $\mathcal{P}_N(X)$

Denote $\mathcal{P}_N(X)$ as the set of all empirical distribution of X in a length-N sequence

Example

If $X \in \{0, 1\}$,

$$\mathcal{P}_N(X) = \left\{(p_X(0), p_X(1)) : \left(\frac{0}{N}, \frac{N}{N}\right), \left(\frac{1}{N}, \frac{N-1}{N}\right), \cdots, \left(\frac{N}{N}, \frac{0}{N}\right)\right\}$$

Note that $|\mathcal{P}_N(X)| = N + 1$

- Since a type is uniquely characterized by a distribution of X in a length-N sequence

Set of all empirical distribution $\mathcal{P}_N(X)$

Denote $\mathcal{P}_N(X)$ as the set of all empirical distribution of X in a length-N sequence

Example

If $X \in \{0, 1\}$,

$$\mathcal{P}_N(X) = \left\{ (p_X(0), p_X(1)) : \left(\frac{0}{N}, \frac{N}{N} \right), \left(\frac{1}{N}, \frac{N-1}{N} \right), \cdots, \left(\frac{N}{N}, \frac{0}{N} \right) \right\}$$

Note that $|\mathcal{P}_N(X)| = N + 1$

- Since a type is uniquely characterized by a distribution of X in a length-N sequence
- Each element p of $\mathcal{P}_N(X)$ corresponds a type $T(p)$

Set of all empirical distribution $\mathcal{P}_N(X)$

Denote $\mathcal{P}_N(X)$ as the set of all empirical distribution of X in a length-N sequence

Example

If $X \in \{0, 1\}$,

$$\mathcal{P}_N(X) = \left\{ (p_X(0), p_X(1)) : \left(\frac{0}{N}, \frac{N}{N}\right), \left(\frac{1}{N}, \frac{N-1}{N}\right), \cdots, \left(\frac{N}{N}, \frac{0}{N}\right) \right\}$$

Note that $|\mathcal{P}_N(X)| = N + 1$

- Since a type is uniquely characterized by a distribution of X in a length-N sequence
- Each element p of $\mathcal{P}_N(X)$ corresponds a type $T(p)$
- Number of types is $|\mathcal{P}_N(X)|$

Number of types

It is not too difficult to count the exact number of types. But in practice, we don't quite bother with it as long as we know that the number is relatively "small"

Theorem 2

$$|\mathcal{P}_N(X)| \leq (N+1)^{|\mathcal{X}|}$$

Number of types

It is not too difficult to count the exact number of types. But in practice, we don't quite bother with it as long as we know that the number is relatively “small”

Theorem 2

$$|\mathcal{P}_N(X)| \leq (N+1)^{|\mathcal{X}|}$$

Proof

Note that each type is specified by the empirical probability of each outcome of X. And the possible values of the empirical probabilities are $\frac{0}{N}$, $\frac{1}{N}$, $\cdots$, $\frac{N}{N}$ ($N+1$ of them).

Number of types

It is not too difficult to count the exact number of types. But in practice, we don't quite bother with it as long as we know that the number is relatively "small"

Theorem 2

$$|\mathcal{P}_N(X)| \leq (N+1)^{|\mathcal{X}|}$$

Proof

Note that each type is specified by the empirical probability of each outcome of X. And the possible values of the empirical probabilities are $\frac{0}{N}, \frac{1}{N}, \cdots, \frac{N}{N}$ ($N+1$ of them). Since there are $|\mathcal{X}|$ elements, the number of types is bounded by $(N+1)^{|\mathcal{X}|}$

Size of a type class

Recall that $|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$ but the following bounds are much more useful in practice

Theorem 3

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

Size of a type class

Recall that $|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$ but the following bounds are much more useful in practice

Theorem 3

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

Proof

Let's assume $p(\cdot)$ is the actual distribution of X here

$$1 \geq \sum_{x^N \in T(p)} p^N(x^N)$$

Size of a type class

Recall that $|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$ but the following bounds are much more useful in practice

Theorem 3

$$\frac{1}{(N+1)^{|\mathcal{X}|}}2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

Proof

Let's assume $p(\cdot)$ is the actual distribution of X here

$$1 \geq \sum_{x^N \in T(p)} p^N(x^N) = \sum_{x^N \in T(p)} 2^{-NH(p)} = |T(p)|2^{-NH(p)}$$

Size of a type class

Recall that $|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$ but the following bounds are much more useful in practice

Theorem 3

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

Proof

Let's assume $p(\cdot)$ is the actual distribution of X here

$$1 \geq \sum_{x^N \in T(p)} p^N(x^N) = \sum_{x^N \in T(p)} 2^{-NH(p)} = |T(p)| 2^{-NH(p)}$$

$$1 = \sum_{\hat{p} \in \mathcal{P}_N} Pr(T(\hat{p}))$$

Size of a type class

Recall that $|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$ but the following bounds are much more useful in practice

Theorem 3

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

Proof

Let's assume $p(\cdot)$ is the actual distribution of X here

$$1 \geq \sum_{x^N \in T(p)} p^N(x^N) = \sum_{x^N \in T(p)} 2^{-NH(p)} = |T(p)| 2^{-NH(p)}$$

$$1 = \sum_{\hat{p} \in \mathcal{P}_N} Pr(T(\hat{p})) \leq \sum_{\hat{p} \in \mathcal{P}_N} \max_{\tilde{p}} Pr(T(\tilde{p}))$$

Size of a type class

Recall that $|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$ but the following bounds are much more useful in practice

Theorem 3

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

Proof

Let's assume $p(\cdot)$ is the actual distribution of X here

$$1 \geq \sum_{x^N \in T(p)} p^N(x^N) = \sum_{x^N \in T(p)} 2^{-NH(p)} = |T(p)| 2^{-NH(p)}$$

$$1 = \sum_{\hat{p} \in \mathcal{P}_N} Pr(T(\hat{p})) \leq \sum_{\hat{p} \in \mathcal{P}_N} \max_{\tilde{p}} Pr(T(\tilde{p})) = \sum_{\hat{p} \in \mathcal{P}_N} Pr(T(p))$$

Size of a type class

Recall that $|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$ but the following bounds are much more useful in practice

Theorem 3

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

Proof

Let's assume $p(\cdot)$ is the actual distribution of X here

$$1 \geq \sum_{x^N \in T(p)} p^N(x^N) = \sum_{x^N \in T(p)} 2^{-NH(p)} = |T(p)| 2^{-NH(p)}$$

$$1 = \sum_{\hat{p} \in \mathcal{P}_N} Pr(T(\hat{p})) \leq \sum_{\hat{p} \in \mathcal{P}_N} \max_{\tilde{p}} Pr(T(\tilde{p})) = \sum_{\hat{p} \in \mathcal{P}_N} Pr(T(p)) \leq (N+1)^{|\mathcal{X}|} Pr(T(p))$$

Size of a type class

Recall that $|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$ but the following bounds are much more useful in practice

Theorem 3

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

Proof

Let's assume $p(\cdot)$ is the actual distribution of X here

$$1 \geq \sum_{x^N \in T(p)} p^N(x^N) = \sum_{x^N \in T(p)} 2^{-NH(p)} = |T(p)| 2^{-NH(p)}$$

$$1 = \sum_{\hat{p} \in \mathcal{P}_N} Pr(T(\hat{p})) \leq \sum_{\hat{p} \in \mathcal{P}_N} \max_{\tilde{p}} Pr(T(\tilde{p})) = \sum_{\hat{p} \in \mathcal{P}_N} Pr(T(p)) \leq (N+1)^{|\mathcal{X}|} Pr(T(p))$$

$$= (N+1)^{|\mathcal{X}|} |T(p)| 2^{-NH(p)}$$

Probability of a type class

Theorem 4

Let the true distribution of X is $q(\cdot)$, then

$$\frac{2^{-N(KL(p||q))}}{(N+1)^{|\mathcal{X}|}} \leq Pr(T(p)) \leq 2^{-N(KL(p||q))}$$

Probability of a type class

Theorem 4

Let the true distribution of X is $q(\cdot)$, then

$$\frac{2^{-N(KL(p||q))}}{(N+1)^{|\mathcal{X}|}} \leq Pr(T(p)) \leq 2^{-N(KL(p||q))}$$

Proof

From Theorem 1, each sequence in $T(p)$ has probability $2^{-N(H(p)+KL(p||q))}$ and since $\frac{1}{(N+1)^{|\mathcal{X}|}}2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$ from Theorem 3,

$$\frac{1}{(N+1)^{|\mathcal{X}|}}2^{NH(p)}2^{-N(H(p)+KL(p||q))} \leq Pr(T(p)) \leq 2^{NH(p)}2^{-N(H(p)+KL(p||q))}$$

Summary of type

- Type class $T(p)$ contains all sequences with empirical distribution of p. That is,

$$T(p) = \left\{ x^N : \frac{\mathscr{N}(a|x^N)}{N} = p(a) \right\}$$

Summary of type

- Type class $T(p)$ contains all sequences with empirical distribution of p. That is,

$$T(p) = \left\{ x^N : \frac{\mathcal{N}(a|x^N)}{N} = p(a) \right\}$$

- All sequences in the type class $T(p)$ has the same probability ($q(\cdot)$ is the true distribution)

$$q^N(x^N) = 2^{-N(H(p)+KL(p||q)}$$

Summary of type

- Type class $T(p)$ contains all sequences with empirical distribution of p. That is,

$$T(p)=\left\{x^N:\frac{\mathcal{N}(a|x^N)}{N}=p(a)\right\}$$

- All sequences in the type class $T(p)$ has the same probability ($q(\cdot)$ is the true distribution)

$$q^N(x^N)=2^{-N(H(p)+KL(p||q)}$$

- There are about $2^{NH(p)}$ sequences in $T(p)$

$$\frac{1}{(N+1)^{|\mathcal{X}|}}2^{NH(p)}\leq|T(p)|\leq 2^{NH(p)}$$

Summary of type

- Type class $T(p)$ contains all sequences with empirical distribution of p. That is,

$$T(p) = \left\{ x^N : \frac{\mathcal{N}(a|x^N)}{N} = p(a) \right\}$$

- All sequences in the type class $T(p)$ has the same probability ($q(\cdot)$ is the true distribution)

$$q^N(x^N) = 2^{-N(H(p)+KL(p||q)}$$

- There are about $2^{NH(p)}$ sequences in $T(p)$

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

- Probability of getting a sequence in $T(p)$ is about $2^{-N(KL(p||q))}$. More precisely,

$$\frac{2^{-N(KL(p||q))}}{(N+1)^{|\mathcal{X}|}} \leq Pr(T(p)) \leq 2^{-N(KL(p||q))}$$

Summary of type

- Type class $T(p)$ contains all sequences with empirical distribution of p. That is,

$$T(p) = \left\{ x^N : \frac{\mathcal{N}(a|x^N)}{N} = p(a) \right\}$$

- All sequences in the type class $T(p)$ has the same probability ($q(\cdot)$ is the true distribution)

$$q^N(x^N) = 2^{-N(H(p)+KL(p||q)}$$

- There are about $2^{NH(p)}$ sequences in $T(p)$

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \le |T(p)| \le 2^{NH(p)}$$

- Probability of getting a sequence in $T(p)$ is about $2^{-N(KL(p||q))}$. More precisely,

$$\frac{2^{-N(KL(p||q))}}{(N+1)^{|\mathcal{X}|}} \le Pr(T(p)) \le 2^{-N(KL(p||q))}$$

- There are $(N+1)^{|\mathcal{X}|}$ types

Rationale

- For the compression scheme (such as Huffmann coding) that we discussed earlier in this class, one needs to know the source distribution ahead to design the encoder and decoder

Rationale

- For the compression scheme (such as Huffmann coding) that we discussed earlier in this class, one needs to know the source distribution ahead to design the encoder and decoder
- Question: Is it possible to construct compression scheme without knowing the source distribution and still performs as good?

Rationale

- For the compression scheme (such as Huffmann coding) that we discussed earlier in this class, one needs to know the source distribution ahead to design the encoder and decoder
- Question: Is it possible to construct compression scheme without knowing the source distribution and still performs as good?
- Answer: Yes. At least theoretically → universal source coding

Theory of universal source coding

Given any source Q with $H(Q) < R$, there exists a length-N universal code of rate R such that the source can be decoded losslessly as $N \to \infty$

Theory of universal source coding

Given any source Q with $H(Q) < R$, there exists a length-N universal code of rate R such that the source can be decoded losslessly as $N \to \infty$

Proof

Let $R_N = R - |\mathcal{X}|\frac{\log(N+1)}{N}$, and consider the set of sequences $A = \{x^N : H(p_{x^N}) < R_N\}$ as the code book.

Theory of universal source coding

Given any source Q with $H(Q) < R$, there exists a length-N universal code of rate R such that the source can be decoded losslessly as $N \to \infty$

Proof

Let $R_N = R - |\mathcal{X}|\frac{\log(N+1)}{N}$, and consider the set of sequences $A = \{x^N : H(p_{x^N}) < R_N\}$ as the code book. Note that the rate is $< R$ as

$$|A| = \sum_{p:H(p)<R_N} |T(p)|$$

Theory of universal source coding

Given any source Q with $H(Q) < R$, there exists a length-N universal code of rate R such that the source can be decoded losslessly as $N \to \infty$

Proof

Let $R_N = R - |\mathcal{X}|\frac{\log(N+1)}{N}$, and consider the set of sequences $A = \{x^N : H(p_{x^N}) < R_N\}$ as the code book. Note that the rate is $< R$ as

$$|A| = \sum_{p:H(p)<R_N} |T(p)| \leq \sum_{p:H(p)<R_N} 2^{NH(p)}$$

Theory of universal source coding

Given any source Q with $H(Q) < R$, there exists a length-N universal code of rate R such that the source can be decoded losslessly as $N \to \infty$

Proof

Let $R_N = R - |\mathcal{X}|\frac{\log(N+1)}{N}$, and consider the set of sequences $A = \{x^N : H(p_{x^N}) < R_N\}$ as the code book. Note that the rate is $< R$ as

$$|A| = \sum_{p:H(p)<R_N} |T(p)| \leq \sum_{p:H(p)<R_N} 2^{NH(p)} < \sum_{p:H(p)<R_N} 2^{NR_N}$$

Theory of universal source coding

Given any source Q with $H(Q) < R$, there exists a length-N universal code of rate R such that the source can be decoded losslessly as $N \to \infty$

Proof

Let $R_N = R - |\mathcal{X}|\frac{\log(N+1)}{N}$, and consider the set of sequences $A = \{x^N : H(p_{x^N}) < R_N\}$ as the code book. Note that the rate is $< R$ as

$$|A| = \sum_{p:H(p)<R_N} |T(p)| \leq \sum_{p:H(p)<R_N} 2^{NH(p)} < \sum_{p:H(p)<R_N} 2^{NR_N}$$

$$\leq (N+1)^{|\mathcal{X}|} 2^{NR_N}$$

Theory of universal source coding

Given any source Q with $H(Q) < R$, there exists a length-N universal code of rate R such that the source can be decoded losslessly as $N \to \infty$

Proof

Let $R_N = R - |\mathcal{X}|\frac{\log(N+1)}{N}$, and consider the set of sequences $A = \{x^N : H(p_{x^N}) < R_N\}$ as the code book. Note that the rate is $< R$ as

$$
\begin{aligned}
|A| = \sum_{p:H(p)<R_N} |T(p)| &\leq \sum_{p:H(p)<R_N} 2^{NH(p)} < \sum_{p:H(p)<R_N} 2^{NR_N} \\
&\leq (N+1)^{|\mathcal{X}|} 2^{NR_N} = 2^{N\left(R_N + |\mathcal{X}|\frac{\log(N+1)}{N}\right)} = 2^{NR}
\end{aligned}
$$

Theory of universal source coding

Given any source Q with $H(Q) < R$, there exists a length-N universal code of rate R such that the source can be decoded losslessly as $N \to \infty$

Proof

Let $R_N = R - |\mathcal{X}|\frac{\log(N+1)}{N}$, and consider the set of sequences $A = \{x^N : H(p_{x^N}) < R_N\}$ as the code book. Note that the rate is $< R$ as

$$|A| = \sum_{p:H(p)<R_N} |T(p)| \leq \sum_{p:H(p)<R_N} 2^{NH(p)} < \sum_{p:H(p)<R_N} 2^{NR_N}$$

$$\leq (N+1)^{|\mathcal{X}|} 2^{NR_N} = 2^{N\left(R_N + |\mathcal{X}|\frac{\log(N+1)}{N}\right)} = 2^{NR}$$

- Encoder: given input, check if input is in A, output index if so. Otherwise, declare failure
- Decoder: simply map index back to the sequence

Theory of universal source coding

Proof (con't)

Note that the probability of error P_e is given by

$$P_e = \sum_{p:H(p)>R_N} Pr(T(p))$$

Theory of universal source coding

Proof (con't)

Note that the probability of error P_e is given by

$$P_e = \sum_{p:H(p)>R_N} Pr(T(p)) \leq \sum_{p:H(p)>R_N} \max_{\tilde{p}:H(\tilde{p})>R_N} Pr(T(\tilde{p}))$$

Theory of universal source coding

Proof (con't)

Note that the probability of error P_e is given by

$$P_e = \sum_{p:H(p)>R_N} Pr(T(p)) \leq \sum_{p:H(p)>R_N} \max_{\tilde{p}:H(\tilde{p})>R_N} Pr(T(\tilde{p}))$$

$$\leq (1+N)^{|\mathcal{X}|} 2^{-N\left(\min_{\tilde{p}:H(\tilde{p})>R_N} KL(\tilde{p}||q)\right)}$$

Theory of universal source coding

Proof (con't)

Note that the probability of error P_e is given by

$$P_e = \sum_{p:H(p)>R_N} Pr(T(p)) \leq \sum_{p:H(p)>R_N} \max_{\tilde{p}:H(\tilde{p})>R_N} Pr(T(\tilde{p}))$$

$$\leq (1+N)^{|\mathcal{X}|} 2^{-N\left(\min_{\tilde{p}:H(\tilde{p})>R_N} KL(\tilde{p}||q)\right)}$$

- If $H(q) < R$, as $R_N \to R$ as N increases, we can find some N_0 such that $H(q) < R_N$ for all $N \geq N_0$

Theory of universal source coding

Proof (con't)

Note that the probability of error P_e is given by

$$P_e = \sum_{p:H(p)>R_N} Pr(T(p)) \leq \sum_{p:H(p)>R_N} \max_{\tilde{p}:H(\tilde{p})>R_N} Pr(T(\tilde{p}))$$

$$\leq (1+N)^{|\mathcal{X}|} 2^{-N\left(\min_{\tilde{p}:H(\tilde{p})>R_N} KL(\tilde{p}||q)\right)}$$

- If $H(q) < R$, as $R_N \to R$ as N increases, we can find some N_0 such that $H(q) < R_N$ for all $N \geq N_0$
- Therefore, any p in $\{p : H(p) > R_N\}$ cannot be the same as q

Theory of universal source coding

Proof (con't)

Note that the probability of error P_e is given by

$$P_e = \sum_{p:H(p)>R_N} Pr(T(p)) \leq \sum_{p:H(p)>R_N} \max_{\tilde{p}:H(\tilde{p})>R_N} Pr(T(\tilde{p}))$$

$$\leq (1+N)^{|\mathcal{X}|} 2^{-N\left(\min_{\tilde{p}:H(\tilde{p})>R_N} KL(\tilde{p}||q)\right)}$$

- If $H(q) < R$, as $R_N \to R$ as N increases, we can find some N_0 such that $H(q) < R_N$ for all $N \geq N_0$
- Therefore, any p in $\{p : H(p) > R_N\}$ cannot be the same as q
- $\Rightarrow \min_{\tilde{p}:H(\tilde{p})>R_N} KL(\tilde{p}||q) > 0$ for $N \geq N_0$

Theory of universal source coding

Proof (con't)

Note that the probability of error P_e is given by

$$P_e = \sum_{p:H(p)>R_N} Pr(T(p)) \leq \sum_{p:H(p)>R_N} \max_{\tilde{p}:H(\tilde{p})>R_N} Pr(T(\tilde{p}))$$
$$\leq (1+N)^{|\mathcal{X}|} 2^{-N\left(\min_{\tilde{p}:H(\tilde{p})>R_N} KL(\tilde{p}||q)\right)}$$

- If $H(q) < R$, as $R_N \to R$ as N increases, we can find some N_0 such that $H(q) < R_N$ for all $N \geq N_0$
- Therefore, any p in $\{p : H(p) > R_N\}$ cannot be the same as q
- $\Rightarrow \min_{\tilde{p}:H(\tilde{p})>R_N} KL(\tilde{p}||q) > 0$ for $N \geq N_0$
- Hence, $P_e \to 0$ as $N \to \infty$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
    - Construct a dictionary including all previously seen segments

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
    - Construct a dictionary including all previously seen segments
    - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 10110111011110111
  - First parse segment into segments that haven't seen before ⇒

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
    - Construct a dictionary including all previously seen segments
    - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101111 0111
    - First parse segment into segments that haven't seen before $\Rightarrow \overset{1}{1}$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101111 0111
  - First parse segment into segments that haven't seen before $\Rightarrow \overset{1}{1}, \overset{2}{0}$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101111 0111
  - First parse segment into segments that haven't seen before $\Rightarrow \overset{1}{1}, \overset{2}{0}, \overset{3}{11}$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
    - Construct a dictionary including all previously seen segments
    - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101111 0111
    - First parse segment into segments that haven't seen before $\Rightarrow \overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
    - Construct a dictionary including all previously seen segments
    - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101111110111
    - First parse segment into segments that haven't seen before $\Rightarrow \overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}, \overset{5}{110}$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
    - Construct a dictionary including all previously seen segments
    - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101111 0111
    - First parse segment into segments that haven't seen before $\Rightarrow \overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}, \overset{5}{110}, \overset{6}{111}$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101111 0111
  - First parse segment into segments that haven't seen before $\Rightarrow \overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}, \overset{5}{110}, \overset{6}{111}, \overset{7}{10}$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101111110111
  - First parse segment into segments that haven't seen before $\Rightarrow \overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}, \overset{5}{110}, \overset{6}{111}, \overset{7}{10}, \overset{8}{111}$
  - Encode each segment into representation containing a pair of numbers:

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
    - Construct a dictionary including all previously seen segments
    - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101111110111
    - First parse segment into segments that haven't seen before $\Rightarrow \overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}, \overset{5}{110}, \overset{6}{111}, \overset{7}{10}, \overset{8}{111}$
    - Encode each segment into representation containing a pair of numbers: 1) index of segment (excluding the last bit) in the dictionary;

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
    - Construct a dictionary including all previously seen segments
    - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101111 0111
    - First parse segment into segments that haven't seen before $\Rightarrow \overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}, \overset{5}{110}, \overset{6}{111}, \overset{7}{10}, \overset{8}{111}$
    - Encode each segment into representation containing a pair of numbers: 1) index of segment (excluding the last bit) in the dictionary; 2) the last bit

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 10110111011110111
  - First parse segment into segments that haven't seen before $\Rightarrow \overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}, \overset{5}{110}, \overset{6}{111}, \overset{7}{10}, \overset{8}{111}$
  - Encode each segment into representation containing a pair of numbers: 1) index of segment (excluding the last bit) in the dictionary; 2) the last bit $\Rightarrow$ $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
    - Construct a dictionary including all previously seen segments
    - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101110111
    - First parse segment into segments that haven't seen before $\Rightarrow \overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}, \overset{5}{110}, \overset{6}{111}, \overset{7}{10}, \overset{8}{111}$
    - Encode each segment into representation containing a pair of numbers: 1) index of segment (excluding the last bit) in the dictionary; 2) the last bit $\Rightarrow$ $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
    - Encode representation to bit stream. Note that as the dictionary grows, number of bits needed to store the index increases $\Rightarrow$ 0100011101011001110010110

Lempel-Ziv decoding

- Decode bitstream back to representation 0100011101011001110010110 $\Rightarrow$ $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

Lempel-Ziv decoding

- Decode bitstream back to representation 0100011101011001110010110 $\Rightarrow$ $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

1

1

$\Rightarrow 1$

Lempel-Ziv decoding

- Decode bitstream back to representation 0100011101011001110010110 ⇒ $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

1	2
1	0

⇒ 10

Lempel-Ziv decoding

- Decode bitstream back to representation 0100011101011001110010110 $\Rightarrow$ $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

1	2	3
1	0	11

$\Rightarrow 1011$

Lempel-Ziv decoding

- Decode bitstream back to representation 0100011101011001110010110 $\Rightarrow$ $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

1	2	3	4
1	0	11	01

$\Rightarrow 101101$

Lempel-Ziv decoding

- Decode bitstream back to representation 0100011101011001110010110 $\Rightarrow$ $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

1	2	3	4	5
1	0	11	01	110

$\Rightarrow$ 101101110

Lempel-Ziv decoding

- Decode bitstream back to representation $0100011101011001110010110 \Rightarrow$ $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

1	2	3	4	5	6
1	0	11	01	110	111

$\Rightarrow 101101110111$

Lempel-Ziv decoding

- Decode bitstream back to representation 0100011101011001110010110 ⇒ $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

1	2	3	4	5	6	7
1	0	11	01	110	111	10

⇒ 10110111011110

Lempel-Ziv decoding

- Decode bitstream back to representation 0100011101011001110010110 $\Rightarrow$ $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

1	2	3	4	5	6	7	8
1	0	11	01	110	111	10	111

$\Rightarrow$ 10110111011110111

Motivation

- Let's revisit some coin tossing example. Say if a coin is fair, and we toss if for 1000 times, we know that we will almost always get 500 heads. So getting, say, 400 heads has neglible probability

Motivation

- Let's revisit some coin tossing example. Say if a coin is fair, and we toss if for 1000 times, we know that we will almost always get 500 heads. So getting, say, 400 heads has neglible probability
- However, if we insist finding the probability of getting 400 heads, from discussion up to now, we know that it is just

$$Pr(T((0.4, 0.6))) \approx 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))}$$

Motivation

- Let's revisit some coin tossing example. Say if a coin is fair, and we toss if for 1000 times, we know that we will almost always get 500 heads. So getting, say, 400 heads has neglible probability
- However, if we insist finding the probability of getting 400 heads, from discussion up to now, we know that it is just

$$Pr(T((0.4,0.6))) \approx 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))}$$

- Now, what if we are interested in the probability of a more general case? Say what is the probability of getting > 300 and < 400 heads?

Sanov's Theorem

Let $\mathcal{E} = \{p : 0.3 \le p(Head) \le 0.4\}$ and $q(\cdot) = (0.5, 0.5)$ is the true distribution, then

$$Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_{1000})$$

Sanov's Theorem

Let $\mathcal{E} = \{p : 0.3 \leq p(Head) \leq 0.4\}$ and $q(\cdot) = (0.5, 0.5)$ is the true distribution, then

$$Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_{1000}) = \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} Pr(T(p))$$

Sanov's Theorem

Let $\mathcal{E} = \{p : 0.3 \leq p(Head) \leq 0.4\}$ and $q(\cdot) = (0.5, 0.5)$ is the true distribution, then

$$Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_{1000}) = \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} Pr(T(p)) \approx \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} 2^{-1000(KL(p||q))}$$

Sanov's Theorem

Let $\mathcal{E} = \{p : 0.3 \leq p(Head) \leq 0.4\}$ and $q(\cdot) = (0.5, 0.5)$ is the true distribution, then

$$Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_{1000}) = \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} Pr(T(p)) \approx \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} 2^{-1000(KL(p||q))}$$

$$= 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))} + 2^{-1000(KL((0.399,0.601)||(0.5,0.5)))} +$$

$$2^{-1000(KL((0.398,0.602)||(0.5,0.5)))} + \cdots + 2^{-1000(KL((0.3,0.7)||(0.5,0.5)))}$$

Sanov's Theorem

Let $\mathcal{E} = \{p : 0.3 \leq p(Head) \leq 0.4\}$ and $q(\cdot) = (0.5, 0.5)$ is the true distribution, then

$$Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_{1000}) = \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} Pr(T(p)) \approx \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} 2^{-1000(KL(p||q))}$$

$$= 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))} + 2^{-1000(KL((0.399,0.601)||(0.5,0.5)))} +$$

$$2^{-1000(KL((0.398,0.602)||(0.5,0.5)))} + \cdots + 2^{-1000(KL((0.3,0.7)||(0.5,0.5)))}$$

$$\leq |\mathcal{P}_{1000}| 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))}$$

Sanov's Theorem

Let $\mathcal{E} = \{p : 0.3 \leq p(Head) \leq 0.4\}$ and $q(\cdot) = (0.5, 0.5)$ is the true distribution, then

$$\begin{aligned} Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_{1000}) = \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} Pr(T(p)) \approx \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} 2^{-1000(KL(p||q))} \\ = 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))} + 2^{-1000(KL((0.399,0.601)||(0.5,0.5)))} + \\ 2^{-1000(KL((0.398,0.602)||(0.5,0.5)))} + \cdots + 2^{-1000(KL((0.3,0.7)||(0.5,0.5)))} \\ \leq |\mathcal{P}_{1000}| 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))} \end{aligned}$$

Sanov's Theorem

Let $X_1, X_2, \cdots, X_N$ be i.i.d. $\sim q(\cdot)$ and $\mathcal{E}$ be a set of distribution. Then

$$Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_N) \leq (N+1)^{|\mathcal{X}|} 2^{-N(KL(p^*||q))},$$

where $p^* = \arg\min_{p \in \mathcal{E}} KL(p||q)$.

Sanov's Theorem

Let $\mathcal{E} = \{p : 0.3 \leq p(Head) \leq 0.4\}$ and $q(\cdot) = (0.5, 0.5)$ is the true distribution, then

$$\begin{aligned} Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_{1000}) &= \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} Pr(T(p)) \approx \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} 2^{-1000(KL(p||q))} \\ &= 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))} + 2^{-1000(KL((0.399,0.601)||(0.5,0.5)))} + \\ &\quad 2^{-1000(KL((0.398,0.602)||(0.5,0.5)))} + \cdots + 2^{-1000(KL((0.3,0.7)||(0.5,0.5)))} \\ &\leq |\mathcal{P}_{1000}| 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))} \end{aligned}$$

Sanov's Theorem

Let $X_1, X_2, \cdots, X_N$ be i.i.d. $\sim q(\cdot)$ and $\mathcal{E}$ be a set of distribution. Then

$$Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_N) \leq (N+1)^{|\mathcal{X}|} 2^{-N(KL(p^*||q))},$$

where $p^* = \arg\min_{p \in \mathcal{E}} KL(p||q)$. Moreover, given a rather weak condition (closure of interior of $\mathcal{E}$ is $\mathcal{E}$ itself), we have

$$\frac{1}{N} \log Pr(\mathcal{E}) \to -KL(p^*||q)$$

Conditional limit theorem

- The first part of Sanov's Theorm is easy to show as similar to the example. However, the second half will need some more math background (mostly mathematical analysis) to understand the proof and so we will skip it here

Conditional limit theorem

- The first part of Sanov's Theorm is easy to show as similar to the example. However, the second half will need some more math background (mostly mathematical analysis) to understand the proof and so we will skip it here
- The latter part of Sanov's Theorem suggests that the probability of getting $\mathcal{E}$ is the same as the probability of getting $T(p^*)$

Conditional limit theorem

- The first part of Sanov's Theorm is easy to show as similar to the example. However, the second half will need some more math background (mostly mathematical analysis) to understand the proof and so we will skip it here
- The latter part of Sanov's Theorem suggests that the probability of getting $\mathcal{E}$ is the same as the probability of getting $T(p^*)$
- It turns out that we can claim something stronger. We will state the theorem below without proof

Conditional limit theorem

- The first part of Sanov's Theorm is easy to show as similar to the example. However, the second half will need some more math background (mostly mathematical analysis) to understand the proof and so we will skip it here
- The latter part of Sanov's Theorem suggests that the probability of getting $\mathcal{E}$ is the same as the probability of getting $T(p^*)$
- It turns out that we can claim something stronger. We will state the theorem below without proof

Conditional limit theorem

Let $\mathcal{E}$ be a closed convex subset of $\mathcal{P}$ (the set of all distributions) and $q(\cdot)$ be the true distribution which is $\notin \mathcal{E}$.

Conditional limit theorem

- The first part of Sanov's Theorm is easy to show as similar to the example. However, the second half will need some more math background (mostly mathematical analysis) to understand the proof and so we will skip it here
- The latter part of Sanov's Theorem suggests that the probability of getting $\mathcal{E}$ is the same as the probability of getting $T(p^*)$
- It turns out that we can claim something stronger. We will state the theorem below without proof

Conditional limit theorem

Let $\mathcal{E}$ be a closed convex subset of $\mathcal{P}$ (the set of all distributions) and $q(\cdot)$ be the true distribution which is $\notin \mathcal{E}$. If $x_1, x_2, \cdots, x_N$ are drawn from $q(\cdot)$ and we know that $p_{x_N} \in \mathcal{E}$, then

$$\frac{\mathcal{N}(a|x_N)}{N} \to p^*(a)$$

in probability as $N \to \infty$

Examples

Coin toss

- Let's go back to our previous example. If we throw a fair coin 1000 times and some one tells you that there are 300 to 400 heads, recall

$$\mathcal{E} = \{0.3 \leq p(Head) \leq 0.4\}$$

Examples

Coin toss

- Let's go back to our previous example. If we throw a fair coin 1000 times and some one tells you that there are 300 to 400 heads, recall
$$\mathcal{E} = \{0.3 \leq p(Head) \leq 0.4\}$$
- Since apparently,
$$p^* = \arg\min_{p \in \mathcal{E}} KL(p||(0.5, 0.5)) = (0.4, 0.6)$$

Examples

Coin toss

- Let's go back to our previous example. If we throw a fair coin 1000 times and some one tells you that there are 300 to 400 heads, recall
$$\mathcal{E} = \{0.3 \leq p(Head) \leq 0.4\}$$
- Since apparently,
$$p^* = \arg\min_{p \in \mathcal{E}} KL(p||(0.5, 0.5)) = (0.4, 0.6)$$
- By conditional limit theorem, knowing the the number of head is within the range, the coin behaves like a biased coin with $p(Head) = 0.4$

Examples

Coin toss

- Let's go back to our previous example. If we throw a fair coin 1000 times and some one tells you that there are 300 to 400 heads, recall
$$\mathcal{E} = \{0.3 \leq p(Head) \leq 0.4\}$$
- Since apparently,
$$p^* = \arg\min_{p \in \mathcal{E}} KL(p||(0.5, 0.5)) = (0.4, 0.6)$$
- By conditional limit theorem, knowing the the number of head is within the range, the coin behaves like a biased coin with $p(Head) = 0.4$
- A best bet would be there are 400 heads

Examples

Lower bounds

- Let say $x_1, x_2, \cdots, x_N$ are drawn from $q(\cdot)$. And we have K functions $g_1(\cdot), g_2(\cdot), \cdots, g_K(\cdot)$ such that for $k = 1, \cdots, K$,

$$\sum_{i=1}^{N} g_k(x_i) p(x_i) \geq \alpha_k$$

Examples

Lower bounds

- Let say $x_1, x_2, \cdots, x_N$ are drawn from $q(\cdot)$. And we have K functions $g_1(\cdot), g_2(\cdot), \cdots, g_K(\cdot)$ such that for $k = 1, \cdots, K$,
$$\sum_{i=1}^{N} g_k(x_i) p(x_i) \geq \alpha_k$$
- Let $\mathcal{E} = \{p : \sum_a p(a) g_k(a) \geq \alpha_k, k = 1, \cdots, K\}$

Examples

Lower bounds

- Let say $x_1, x_2, \cdots, x_N$ are drawn from $q(\cdot)$. And we have K functions $g_1(\cdot), g_2(\cdot), \cdots, g_K(\cdot)$ such that for $k = 1, \cdots, K$,
$$\sum_{i=1}^{N} g_k(x_i)p(x_i) \geq \alpha_k$$
- Let $\mathcal{E} = \{p : \sum_a p(a)g_k(a) \geq \alpha_k, k = 1, \cdots, K\}$
- From conditional limit theorem, $\frac{\mathscr{N}(a|x^N)}{N} \to p^*(a)$, where
$$p^* = \arg\min_{p \in \mathcal{E}} KL(p||q)$$

Examples

Lower bounds

- Let say $x_1, x_2, \cdots, x_N$ are drawn from $q(\cdot)$. And we have K functions $g_1(\cdot), g_2(\cdot), \cdots, g_K(\cdot)$ such that for $k = 1, \cdots, K$,
$$\sum_{i=1}^{N} g_k(x_i)p(x_i) \geq \alpha_k$$
- Let $\mathcal{E} = \{p : \sum_a p(a)g_k(a) \geq \alpha_k, k = 1, \cdots, K\}$
- From conditional limit theorem, $\frac{\mathcal{N}(a|x^N)}{N} \to p^*(a)$, where
$$p^* = \arg\min_{p \in \mathcal{E}} KL(p||q)$$
- This is a simple constrained optimization problem and can be solved with KKT conditions. If you go through the conditions, you will find that
$$p^*(x) \propto q(x)2^{\sum_{k=1}^{K} \lambda_k g_k(x)},$$
with $\lambda_k(\sum_a p(a)g_k(a) - \alpha_k) = 0$, $\lambda_k \geq 0$, and $\sum_a p(a)g_k(a) \geq \alpha_k$

Examples

I think this example below gives a nice demonstration that the technique we have learned today can solve some amazing puzzle!

Examples

I think this example below gives a nice demonstration that the technique we have learned today can solve some amazing puzzle!

Fair dice

A fair dice is thrown 10,000 times and the sum of all outcomes is larger than 40,000, out of the 10,000 throw, how many ones do you think there are?

Fair dice

- From the result of previous example, let $g_1(x) = x$ and $\alpha_1 = 4$, we expect

$$p^*(i) = \frac{2^{\lambda i}}{\sum_{j=1}^6 2^{\lambda j}}$$

for some λ

Fair dice

- From the result of previous example, let $g_1(x) = x$ and $\alpha_1 = 4$, we expect

$$p^*(i) = \frac{2^{\lambda i}}{\sum_{j=1}^{6} 2^{\lambda j}}$$

  for some λ
- $\lambda \neq 0$ since $\sum_a p(a) g_1(a) = 3.5 < 4 = \alpha_1$ if so

Fair dice

- From the result of previous example, let $g_1(x) = x$ and $\alpha_1 = 4$, we expect

$$p^*(i) = \frac{2^{\lambda i}}{\sum_{j=1}^{6} 2^{\lambda j}}$$

  for some λ

- $\lambda \neq 0$ since $\sum_a p(a) g_1(a) = 3.5 < 4 = \alpha_1$ if so
- Since $\lambda \neq 0$, by the complementary slackness constraint $\lambda_k(\sum_a p(a) g_k(a) - \alpha_k) = 0$,

$$\sum_a p(a) g_1(a) = \alpha_1 = 4$$

Fair dice

- From the result of previous example, let $g_1(x) = x$ and $\alpha_1 = 4$, we expect

$$p^*(i) = \frac{2^{\lambda i}}{\sum_{j=1}^6 2^{\lambda j}}$$

for some λ

- $\lambda \neq 0$ since $\sum_a p(a) g_1(a) = 3.5 < 4 = \alpha_1$ if so
- Since $\lambda \neq 0$, by the complementary slackness constraint $\lambda_k(\sum_a p(a) g_k(a) - \alpha_k) = 0$,

$$\sum_a p(a) g_1(a) = \alpha_1 = 4$$

- This gives us $\lambda = 0.2519$, and thus $p^* = (0.103, 0.123, 0.146, 0.174, 0.207, 0.247)$

Fair dice

- From the result of previous example, let $g_1(x) = x$ and $\alpha_1 = 4$, we expect

$$p^*(i) = \frac{2^{\lambda i}}{\sum_{j=1}^{6} 2^{\lambda j}}$$

  for some λ
- $\lambda \neq 0$ since $\sum_a p(a) g_1(a) = 3.5 < 4 = \alpha_1$ if so
- Since $\lambda \neq 0$, by the complementary slackness constraint $\lambda_k(\sum_a p(a) g_k(a) - \alpha_k) = 0$,

$$\sum_a p(a) g_1(a) = \alpha_1 = 4$$

- This gives us $\lambda = 0.2519$, and thus $p^* = (0.103, 0.123, 0.146, 0.174, 0.207, 0.247)$
- $\#$ ones $\approx 0.103 \times 10000 = 1030$

Multivariate Gaussian

Normal distribution

- Univariate Normal: $\mathcal{N}(x;\mu,\sigma^2)=\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$
- Multivariate Normal: $\mathcal{N}(\mathbf{x};\boldsymbol{\mu},\Sigma)=\frac{1}{det(2\pi\Sigma)}e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})}$

Remark

Note that $\mathcal{N}(\mathbf{x};\boldsymbol{\mu},\Sigma)=\mathcal{N}(\boldsymbol{\mu};\mathbf{x},\Sigma)$. *It is trivial but quite useful*

Remark

Σ *is known to be the covariance matrices and it has to be (symmetric) positive definite*

Remark

Consequently, symmetric matrices are carefully studied and understood by statisticians and information theorists (more discussion couple slides later)

Covariance matrices

Definition (Covariance matrices)

Recall that for a vector random variable $\boldsymbol{X} = [X_1, X_2, \cdots, X_n]^T$, the covariance matrix $\Sigma \triangleq E[(\boldsymbol{X} - \boldsymbol{\mu})(\boldsymbol{X} - \boldsymbol{\mu})^T]$

Remark

Covariance matrices are always positive semi-definite since $\forall u$, $u^T \Sigma u = E[u^T(\boldsymbol{X} - \boldsymbol{\mu})(\boldsymbol{X} - \boldsymbol{\mu})^T u] = E[\|(\boldsymbol{X} - \boldsymbol{\mu})^T u\|^2] \geq 0$

Remark

In general, we usually would like to assume Σ *to be strictly positive definite. Because otherwise it means that some of its eigenvalues are zero and so in some dimension, there is actually no variation and is just constant along that dimension. Representing those dimension as random variable is troublesome since "*$1/\sigma^2$*" which occurs often will become infinite. Instead we can always simply strip away those dimensions to avoid complications*

Symmetric matrices

Lemma

$$(M^T)^{-1} = (M^{-1})^T$$

Symmetric matrices

Lemma

$(M^T)^{-1} = (M^{-1})^T$

Proof.

$(M^{-1})^T M^T = (MM^{-1})^T = I \Rightarrow (M^{-1})^T$ is inverse of M^T ☐

Lemma

If M is symmetric, so is M^{-1}

Symmetric matrices

Lemma

$(M^T)^{-1} = (M^{-1})^T$

Proof.

$(M^{-1})^T M^T = (MM^{-1})^T = I \Rightarrow (M^{-1})^T$ is inverse of M^T ☐

Lemma

If M is symmetric, so is M^{-1}

Proof.

$(M^{-1})^T = (M^T)^{-1} = M^{-1}$ ☐

Hermitian matrices

- An extension of transpose operation to complex matrices is the hermitian transpose operation, which is simply the transpose and conjugate of a matrix (vector)
- We denote the hermitian transpose of M as $M^\dagger \triangleq \overline{M}^T$, when $\overline{M}$ is the complex conjugate of M
- A matrix is Hermitian if $M^\dagger = M$. Note that a real symmetric matrix is Hermitian

Eigenvalues of Hermitian matrices

Lemma

If M is Hermitian ($M^{\dagger} = M$), all eigenvalues are real

Eigenvalues of Hermitian matrices

Lemma

If M is Hermitian ($M^\dagger = M$), all eigenvalues are real

Proof.

$\overline{\lambda}(x^\dagger x) = (\lambda x)^\dagger x = (Mx)^\dagger x = x^\dagger M^\dagger x = x^\dagger M x = x^\dagger(\lambda x) = \lambda(x^\dagger x)$ □

Lemma

If M is Hermitian, eigenvectors of different eigenvalues are orthogonal

Eigenvalues of Hermitian matrices

Lemma

If M is Hermitian ($M^\dagger = M$), all eigenvalues are real

Proof.

$\overline{\lambda}(x^\dagger x) = (\lambda x)^\dagger x = (Mx)^\dagger x = x^\dagger M^\dagger x = x^\dagger M x = x^\dagger(\lambda x) = \lambda(x^\dagger x)$ □

Lemma

If M is Hermitian, eigenvectors of different eigenvalues are orthogonal

Proof.

$$\begin{aligned} &\lambda_1 x_1^\dagger x_2 = (Mx_1)^\dagger x_2 = x_1^\dagger M x_2 = \lambda_2 x_1^\dagger x_2 \\ \Rightarrow &\lambda_1 \neq \lambda_2 \Rightarrow x_1^\dagger x_2 = 0 \end{aligned}$$

□

Hermitian matrices are diagonizable

Lemma

Hermitian matrices are diagonizable

Proof (*).

We will sketch the proof by construction. For any n-d Hermitian matrix M, consider an eigenvalue λ and corresponding eigenvector u, without loss of generality, let's also normalize u such that $\|u\| = 1$. Consider the subspace orthogonal to u, $U^\perp$, and let $v_1, \cdots, v_{n-1}$ be arbitrary orthonormal basis of $U^\perp$. Note that for any k, Mv_k will be orthogonal to u since

$$u^\dagger M v_k = u^\dagger M^\dagger v_k = (Mu)^\dagger v_k = \lambda u^\dagger v_k = 0.$$

Thus, $(u, v_1, \cdots, v_{n-1})^\dagger M (u, v_1, \cdots, v_{n-1}) = \left(\begin{smallmatrix} \lambda & 0 \\ 0 & M' \end{smallmatrix}\right)$. Moreover, M' is also a Hermitian matrix with one less dimension. We can apply the same process on M' and "diagonalize" one more row/column. That is, $\left(\begin{smallmatrix} 1 & 0 \\ 0 & P' \end{smallmatrix}\right)^\dagger P^\dagger M P \left(\begin{smallmatrix} 1 & 0 \\ 0 & P' \end{smallmatrix}\right) = \begin{pmatrix} \lambda & 0 & \cdots \\ 0 & \lambda' & \\ & & M'' \end{pmatrix}$. We can repeat this until the entire M is diagonalized ☐

Hermitian matrices are diagonalizable

Remark

We can find a orthogonal set of eigenvectors that diagonalize a Hermitian matrix. That is

$$(v_1, \cdots, v_n)^\dagger M \underbrace{(v_1, \cdots, v_n)}_{V} = \begin{pmatrix} \lambda_1 & 0 & \cdots \\ 0 & \lambda_2 & \\ \vdots & & \ddots \end{pmatrix},$$

and V is unitary (orthogonal), i.e., $V^\dagger V = I$ and thus $V^{-1} = V^\dagger$. Note that $v_i \perp v_j$ if $\lambda_i \neq \lambda_j$. Otherwise, we may use Gram-Schmidt

Remark

The reverse is obviously true. If a matrix can be diagonalized by a unitary matrix into a real diagonal matrix, the matrix is Hermitian

Remark

Recall that real-symmetric matrices are Hermitian, thus can be diagonalized by its eigenvectors also

Positive definite matrices

Definition (Positive definite)

For a Hermitian matrix M, it is positive definite iff $\forall x$, $x^\dagger Mx > 0$

Definition (Positive semi-definite)

For a Hermitian matrix M, it is positive semi-definite iff $\forall x$, $x^\dagger Mx \geq 0$

Remark

M is positive definite (semi-definite) iff all its eigenvalue is larger (larger or equal to) 0

Positive definite matrices

Definition (Positive definite)

For a Hermitian matrix M, it is positive definite iff $\forall x$, $x^\dagger M x > 0$

Definition (Positive semi-definite)

For a Hermitian matrix M, it is positive semi-definite iff $\forall x$, $x^\dagger M x \geq 0$

Remark

M is positive definite (semi-definite) iff all its eigenvalue is larger (larger or equal to) 0

Proof.

$\Rightarrow$: assume positive definite but some eigenvalue < 0, WLOG, let $\lambda_1 < 0$, then $v_1^\dagger M v_1 = \lambda_1 < 0$ contradicts that M is positive definite

$\Leftarrow$: If $\forall k, \lambda_k > 0$, for any x, $x^\dagger M x = (V^\dagger x)^\dagger \begin{pmatrix} \lambda_1 & 0 \\ 0 & \ddots \end{pmatrix} V^\dagger x = \sum_i \lambda_i (V^\dagger x)_i^2 > 0$ □

Eigenvectors and eigenvalues of covariance matrices

- WLOG, let's assume $\mathbf{X} = [X_1, X_2, \cdots, X_n]^T$ is zero mean. So the covariance matrix $\Sigma_X = E[\mathbf{X}\mathbf{X}^T]$

Eigenvectors and eigenvalues of covariance matrices

- WLOG, let's assume $\mathbf{X} = [X_1, X_2, \cdots, X_n]^T$ is zero mean. So the covariance matrix $\Sigma_X = E[\mathbf{X}\mathbf{X}^T]$
- Covariance matrices are real symmetric (hence Hermitian) and so can be diagonalized by its eigenvectors. That is,
  - $P^T \Sigma_X P = D$, where $P = [u_1, u_2, \cdots, u_n]$ with u_k being eigenvectors of Σ and D is a diagonal matrix with eigenvalues $\lambda_1, \lambda_2, \cdots, \lambda_n$ as the diagonal elements

Eigenvectors and eigenvalues of covariance matrices

- WLOG, let's assume $\mathbf{X} = [X_1, X_2, \cdots, X_n]^T$ is zero mean. So the covariance matrix $\Sigma_X = E[\mathbf{X}\mathbf{X}^T]$
- Covariance matrices are real symmetric (hence Hermitian) and so can be diagonalized by its eigenvectors. That is,
    - $P^T \Sigma_X P = D$, where $P = [u_1, u_2, \cdots, u_n]$ with u_k being eigenvectors of Σ and D is a diagonal matrix with eigenvalues $\lambda_1, \lambda_2, \cdots, \lambda_n$ as the diagonal elements
- Let $\mathbf{Y} = P^T \mathbf{X}$, note that the covariance matrix of $\mathbf{Y}$

$$\Sigma_Y = E[\mathbf{Y}\mathbf{Y}^T] = E[P^T \mathbf{X}\mathbf{X}^T P] = P^T E[\mathbf{X}\mathbf{X}^T] P = P^T \Sigma_X P = D$$

is diagonalized

Eigenvectors and eigenvalues of covariance matrices

- WLOG, let's assume $\mathbf{X} = [X_1, X_2, \cdots, X_n]^T$ is zero mean. So the covariance matrix $\Sigma_X = E[\mathbf{X}\mathbf{X}^T]$
- Covariance matrices are real symmetric (hence Hermitian) and so can be diagonalized by its eigenvectors. That is,
  - $P^T \Sigma_X P = D$, where $P = [u_1, u_2, \cdots, u_n]$ with u_k being eigenvectors of Σ and D is a diagonal matrix with eigenvalues $\lambda_1, \lambda_2, \cdots, \lambda_n$ as the diagonal elements
- Let $\mathbf{Y} = P^T\mathbf{X}$, note that the covariance matrix of $\mathbf{Y}$

$$\Sigma_Y = E[\mathbf{Y}\mathbf{Y}^T] = E[P^T\mathbf{X}\mathbf{X}^T P] = P^T E[\mathbf{X}\mathbf{X}^T]P = P^T \Sigma_X P = D$$

  is diagonalized
  - So the variance of Y_k is simply λ_k
  - $E[Y_i Y_j] = 0$ for $i \neq j$. That is, $Y_i \perp\!\!\!\perp Y_j$ for $i \neq j$
  - Note that the projection $\mathbf{X}$ to the eigenvectors resulting in $\mathbf{Y} = P^T\mathbf{X}$ being independent, showing that eigenvectors are the principal compoents

Principal component analysis (PCA)

- Recall that $\Sigma = E[\mathbf{X}\mathbf{X}^T]$ (assume $\mathbf{X}$ is zero-mean) and $\mathbf{Y} = P^T\mathbf{X}$ with $E[\mathbf{Y}\mathbf{Y}^T] = P^T\Sigma P = D$
- Assume that the diagonal of D (note that those are eigenvalues) are arranged in descending order that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$

---

[1] $tr(AB) = \sum_i \sum_j a_{i,j} b_{j,i} = \sum_j \sum_i b_{j,i} a_{i,j} = tr(BA)$

Principal component analysis (PCA)

- Recall that $\Sigma = E[\mathbf{X}\mathbf{X}^T]$ (assume $\mathbf{X}$ is zero-mean) and $\mathbf{Y} = P^T\mathbf{X}$ with $E[\mathbf{Y}\mathbf{Y}^T] = P^T\Sigma P = D$
- Assume that the diagonal of D (note that those are eigenvalues) are arranged in descending order that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$
  - Generate an approximate $\hat{\mathbf{Y}}$ of $\mathbf{Y}$ by setting all components except first k as 0

---

[1] $tr(AB) = \sum_i \sum_j a_{i,j} b_{j,i} = \sum_j \sum_i b_{j,i} a_{i,j} = tr(BA)$

Principal component analysis (PCA)

- Recall that $\Sigma = E[\mathbf{X}\mathbf{X}^T]$ (assume $\mathbf{X}$ is zero-mean) and $\mathbf{Y} = P^T\mathbf{X}$ with $E[\mathbf{Y}\mathbf{Y}^T] = P^T \Sigma P = D$
- Assume that the diagonal of D (note that those are eigenvalues) are arranged in descending order that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$
    - Generate an approximate $\hat{\mathbf{Y}}$ of $\mathbf{Y}$ by setting all components except first k as 0
    - The mean square error (mse) of[1] $\hat{\mathbf{Y}} = E[(\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}})]$

---

[1] $tr(AB) = \sum_i \sum_j a_{i,j} b_{j,i} = \sum_j \sum_i b_{j,i} a_{i,j} = tr(BA)$

Principal component analysis (PCA)

- Recall that $\Sigma = E[\mathbf{X}\mathbf{X}^T]$ (assume $\mathbf{X}$ is zero-mean) and $\mathbf{Y} = P^T\mathbf{X}$ with $E[\mathbf{Y}\mathbf{Y}^T] = P^T\Sigma P = D$
- Assume that the diagonal of D (note that those are eigenvalues) are arranged in descending order that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$
    - Generate an approximate $\hat{\mathbf{Y}}$ of $\mathbf{Y}$ by setting all components except first k as 0
    - The mean square error (mse) of[1] $\hat{\mathbf{Y}} = E[(\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}})] = tr(E[(\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}})])$

---

[1] $tr(AB) = \sum_i \sum_j a_{i,j} b_{j,i} = \sum_j \sum_i b_{j,i} a_{i,j} = tr(BA)$

Principal component analysis (PCA)

- Recall that $\Sigma = E[\mathbf{X}\mathbf{X}^T]$ (assume $\mathbf{X}$ is zero-mean) and $\mathbf{Y} = P^T\mathbf{X}$ with $E[\mathbf{Y}\mathbf{Y}^T] = P^T \Sigma P = D$
- Assume that the diagonal of D (note that those are eigenvalues) are arranged in descending order that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$
    - Generate an approximate $\hat{\mathbf{Y}}$ of $\mathbf{Y}$ by setting all components except first k as 0
    - The mean square error (mse) of[1] $\hat{\mathbf{Y}} = E[(\mathbf{Y} - \hat{\mathbf{Y}})^T(\mathbf{Y} - \hat{\mathbf{Y}})] = tr(E[(\mathbf{Y} - \hat{\mathbf{Y}})^T(\mathbf{Y} - \hat{\mathbf{Y}})])$ $= E[tr((\mathbf{Y} - \hat{\mathbf{Y}})^T(\mathbf{Y} - \hat{\mathbf{Y}}))]$

---

[1] $tr(AB) = \sum_i \sum_j a_{i,j} b_{j,i} = \sum_j \sum_i b_{j,i} a_{i,j} = tr(BA)$

Principal component analysis (PCA)

- Recall that $\Sigma = E[\mathbf{X}\mathbf{X}^T]$ (assume $\mathbf{X}$ is zero-mean) and $\mathbf{Y} = P^T\mathbf{X}$ with $E[\mathbf{Y}\mathbf{Y}^T] = P^T\Sigma P = D$
- Assume that the diagonal of D (note that those are eigenvalues) are arranged in descending order that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$
  - Generate an approximate $\hat{\mathbf{Y}}$ of $\mathbf{Y}$ by setting all components except first k as 0
  - The mean square error (mse) of[1] $\hat{\mathbf{Y}} = E[(\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}})] = tr(E[(\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}})])$ $= E[tr((\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}}))] = E[tr((\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})^T)] = tr(E[(\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})^T])$

---

[1] $tr(AB) = \sum_i \sum_j a_{i,j}b_{j,i} = \sum_j \sum_i b_{j,i}a_{i,j} = tr(BA)$

Principal component analysis (PCA)

- Recall that $\Sigma = E[\mathbf{X}\mathbf{X}^T]$ (assume $\mathbf{X}$ is zero-mean) and $\mathbf{Y} = P^T\mathbf{X}$ with $E[\mathbf{Y}\mathbf{Y}^T] = P^T\Sigma P = D$
- Assume that the diagonal of D (note that those are eigenvalues) are arranged in descending order that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$
  - Generate an approximate $\hat{\mathbf{Y}}$ of $\mathbf{Y}$ by setting all components except first k as 0
  - The mean square error (mse) of[1] $\hat{\mathbf{Y}} = E[(\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}})] = tr(E[(\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}})])$ $= E[tr((\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}}))] = E[tr((\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})^T)] = tr(E[(\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})^T])$ $= \sum_{i=k+1}^n \lambda_i$
  - Similarly, if we "reconstruct" $\mathbf{X}$ as $\hat{\mathbf{X}} = P\hat{\mathbf{Y}}$. The mse of $\hat{\mathbf{X}} = E[(\mathbf{X}-\hat{\mathbf{X}})^T(\mathbf{X}-\hat{\mathbf{X}})] = tr(E[(\mathbf{X}-\hat{\mathbf{X}})(\mathbf{X}-\hat{\mathbf{X}})^T])$

---

[1] $tr(AB) = \sum_i \sum_j a_{i,j}b_{j,i} = \sum_j \sum_i b_{j,i}a_{i,j} = tr(BA)$

Principal component analysis (PCA)

- Recall that $\Sigma = E[\mathbf{X}\mathbf{X}^T]$ (assume $\mathbf{X}$ is zero-mean) and $\mathbf{Y} = P^T\mathbf{X}$ with $E[\mathbf{Y}\mathbf{Y}^T] = P^T\Sigma P = D$
- Assume that the diagonal of D (note that those are eigenvalues) are arranged in descending order that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$
    - Generate an approximate $\hat{\mathbf{Y}}$ of $\mathbf{Y}$ by setting all components except first k as 0
    - The mean square error (mse) of[1] $\hat{\mathbf{Y}} = E[(\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}})] = tr(E[(\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}})])$ $= E[tr((\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}}))] = E[tr((\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})^T)] = tr(E[(\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})^T])$ $= \sum_{i=k+1}^n \lambda_i$
    - Similarly, if we "reconstruct" $\mathbf{X}$ as $\hat{\mathbf{X}} = P\hat{\mathbf{Y}}$. The mse of $\hat{\mathbf{X}} = E[(\mathbf{X}-\hat{\mathbf{X}})^T(\mathbf{X}-\hat{\mathbf{X}})] = tr(E[(\mathbf{X}-\hat{\mathbf{X}})(\mathbf{X}-\hat{\mathbf{X}})^T])= tr(PE[(\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})]P^T)$

---

[1] $tr(AB) = \sum_i \sum_j a_{i,j}b_{j,i} = \sum_j \sum_i b_{j,i}a_{i,j} = tr(BA)$

Principal component analysis (PCA)

- Recall that $\Sigma = E[\mathbf{X}\mathbf{X}^T]$ (assume $\mathbf{X}$ is zero-mean) and $\mathbf{Y} = P^T\mathbf{X}$ with $E[\mathbf{Y}\mathbf{Y}^T] = P^T\Sigma P = D$
- Assume that the diagonal of D (note that those are eigenvalues) are arranged in descending order that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$
    - Generate an approximate $\hat{\mathbf{Y}}$ of $\mathbf{Y}$ by setting all components except first k as 0
    - The mean square error (mse) of[1] $\hat{\mathbf{Y}} = E[(\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}})] = tr(E[(\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}})])$ $= E[tr((\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}}))] = E[tr((\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})^T)] = tr(E[(\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})^T])$ $= \sum_{i=k+1}^n \lambda_i$
    - Similarly, if we "reconstruct" $\mathbf{X}$ as $\hat{\mathbf{X}} = P\hat{\mathbf{Y}}$. The mse of $\hat{\mathbf{X}} = E[(\mathbf{X}-\hat{\mathbf{X}})^T(\mathbf{X}-\hat{\mathbf{X}})] =$ $tr(E[(\mathbf{X}-\hat{\mathbf{X}})(\mathbf{X}-\hat{\mathbf{X}})^T])= tr(PE[(\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})]P^T)= tr(P^TPE[(\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})])$

---

[1] $tr(AB) = \sum_i \sum_j a_{i,j} b_{j,i} = \sum_j \sum_i b_{j,i} a_{i,j} = tr(BA)$

Principal component analysis (PCA)

- Recall that $\Sigma = E[\mathbf{X}\mathbf{X}^T]$ (assume $\mathbf{X}$ is zero-mean) and $\mathbf{Y} = P^T\mathbf{X}$ with $E[\mathbf{Y}\mathbf{Y}^T] = P^T\Sigma P = D$
- Assume that the diagonal of D (note that those are eigenvalues) are arranged in descending order that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$
  - Generate an approximate $\hat{\mathbf{Y}}$ of $\mathbf{Y}$ by setting all components except first k as 0
  - The mean square error (mse) of[1] $\hat{\mathbf{Y}} = E[(\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}})] = tr(E[(\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}})])$ $= E[tr((\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}}))] = E[tr((\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})^T)] = tr(E[(\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})^T])$ $= \sum_{i=k+1}^n \lambda_i$
  - Similarly, if we "reconstruct" $\mathbf{X}$ as $\hat{\mathbf{X}} = P\hat{\mathbf{Y}}$. The mse of $\hat{\mathbf{X}} = E[(\mathbf{X}-\hat{\mathbf{X}})^T(\mathbf{X}-\hat{\mathbf{X}})] =$ $tr(E[(\mathbf{X}-\hat{\mathbf{X}})(\mathbf{X}-\hat{\mathbf{X}})^T])= tr(PE[(\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})]P^T)= tr(P^TPE[(\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})])$ $= \sum_{i=k+1}^n \lambda_i$

---

[1] $tr(AB) = \sum_i \sum_j a_{i,j} b_{j,i} = \sum_j \sum_i b_{j,i} a_{i,j} = tr(BA)$

Principal component analysis (PCA)

- Recall that $\Sigma = E[\mathbf{X}\mathbf{X}^T]$ (assume $\mathbf{X}$ is zero-mean) and $\mathbf{Y} = P^T\mathbf{X}$ with $E[\mathbf{Y}\mathbf{Y}^T] = P^T\Sigma P = D$
- Assume that the diagonal of D (note that those are eigenvalues) are arranged in descending order that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$
    - Generate an approximate $\hat{\mathbf{Y}}$ of $\mathbf{Y}$ by setting all components except first k as 0
    - The mean square error (mse) of[1] $\hat{\mathbf{Y}} = E[(\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}})] = tr(E[(\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}})])$ $= E[tr((\mathbf{Y}-\hat{\mathbf{Y}})^T(\mathbf{Y}-\hat{\mathbf{Y}}))] = E[tr((\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})^T)] = tr(E[(\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})^T])$ $= \sum_{i=k+1}^n \lambda_i$
    - Similarly, if we "reconstruct" $\mathbf{X}$ as $\hat{\mathbf{X}} = P\hat{\mathbf{Y}}$. The mse of $\hat{\mathbf{X}} = E[(\mathbf{X}-\hat{\mathbf{X}})^T(\mathbf{X}-\hat{\mathbf{X}})] =$ $tr(E[(\mathbf{X}-\hat{\mathbf{X}})(\mathbf{X}-\hat{\mathbf{X}})^T])= tr(PE[(\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})]P^T)= tr(P^TPE[(\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})])$ $= \sum_{i=k+1}^n \lambda_i$
    - Note that the eigenvectors of Σ (columns of P) are known as the principal components

---

[1] $tr(AB) = \sum_i \sum_j a_{i,j} b_{j,i} = \sum_j \sum_i b_{j,i} a_{i,j} = tr(BA)$

Practical PCA

In practice, we typically are given a dataset with samples of $\mathbf{X}$ instead of the distribution or covariance matrix of $\mathbf{X}$. Denote the data as $\mathcal{X}$ with each row is a data point and a total of m data points. Thus $\mathcal{X}$ is an m by n matrix

---

[2] I used the matlab notations for $ones(\cdot)$ and $mean(\cdot)$ here

[3] Note that $\hat{\Sigma}$ won't be full rank and positive definite as one would hope

Practical PCA

In practice, we typically are given a dataset with samples of $\mathbf{X}$ instead of the distribution or covariance matrix of $\mathbf{X}$. Denote the data as $\mathcal{X}$ with each row is a data point and a total of m data points. Thus $\mathcal{X}$ is an m by n matrix

- Data are rarely zero-mean to begin with, but we can easily preprocess it by subtracting the mean. That is[2] $\mathcal{X} \leftarrow \mathcal{X} - ones(m,1)mean(\mathcal{X})$

---

[2] I used the matlab notations for $ones(\cdot)$ and $mean(\cdot)$ here

[3] Note that $\hat{\Sigma}$ won't be full rank and positive definite as one would hope

Practical PCA

In practice, we typically are given a dataset with samples of **X** instead of the distribution or covariance matrix of **X**. Denote the data as $\mathcal{X}$ with each row is a data point and a total of m data points. Thus $\mathcal{X}$ is an m by n matrix

- Data are rarely zero-mean to begin with, but we can easily preprocess it by subtracting the mean. That is[2] $\mathcal{X} \leftarrow \mathcal{X} - ones(m,1)mean(\mathcal{X})$
- Note that $\hat{\Sigma} \approx \frac{1}{m}\mathcal{X}^T\mathcal{X}$. We could directly compute the eigenvectors and eigenvalues of $\hat{\Sigma}$ as discussed previously. But in many cases, $m < n$ making $\hat{\Sigma}$ a bad approximate[3]

---

[2] I used the matlab notations for *ones*$(\cdot)$ and *mean*$(\cdot)$ here

[3] Note that $\hat{\Sigma}$ won't be full rank and positive definite as one would hope

Practical PCA

In practice, we typically are given a dataset with samples of $\mathbf{X}$ instead of the distribution or covariance matrix of $\mathbf{X}$. Denote the data as $\mathcal{X}$ with each row is a data point and a total of m data points. Thus $\mathcal{X}$ is an m by n matrix

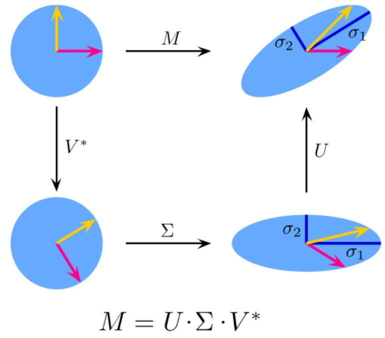
- Data are rarely zero-mean to begin with, but we can easily preprocess it by subtracting the mean. That is[2] $\mathcal{X} \leftarrow \mathcal{X} - ones(m,1)mean(\mathcal{X})$
- Note that $\hat{\Sigma} \approx \frac{1}{m}\mathcal{X}^T\mathcal{X}$. We could directly compute the eigenvectors and eigenvalues of $\hat{\Sigma}$ as discussed previously. But in many cases, $m < n$ making $\hat{\Sigma}$ a bad approximate[3]
  - A more common approach is to decompose $\mathcal{X}$ with singular value decomposition (SVD) instead

---

[2] I used the matlab notations for $ones(\cdot)$ and $mean(\cdot)$ here

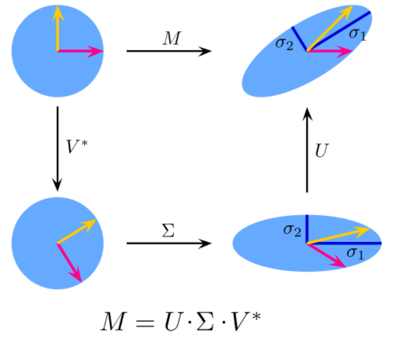
[3] Note that $\hat{\Sigma}$ won't be full rank and positive definite as one would hope

Singular value decomposition (SVD)



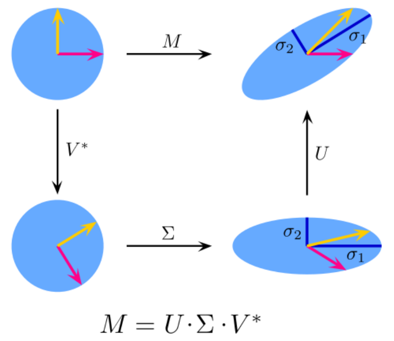
- Every matrix M can be decomposed as $M = UDV^\dagger$, where D is diagonal and U, V are unitary. The diagonal terms in Σ are known to be the singular values

Singular value decomposition (SVD)



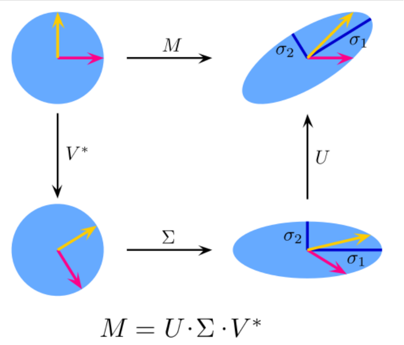
- Every matrix M can be decomposed as $M = UDV^\dagger$, where D is diagonal and U, V are unitary. The diagonal terms in Σ are known to be the singular values
- For real matrix M, we can write $M = UDV^T$ instead. U, V are now "real unitary" or orthogonal

Singular value decomposition (SVD)



- Every matrix M can be decomposed as $M = UDV^\dagger$, where D is diagonal and U, V are unitary. The diagonal terms in Σ are known to be the singular values
- For real matrix M, we can write $M = UDV^T$ instead. U, V are now "real unitary" or orthogonal
  - Note that $M^T M = VD^T U^T UDV^T = VD^2 V^T$. Therefore, V are really eigenvectors of $M^T M$ with eigenvalues equal to the square of the singular values

Singular value decomposition (SVD)



- Every matrix M can be decomposed as $M = UDV^\dagger$, where D is diagonal and U, V are unitary. The diagonal terms in Σ are known to be the singular values
- For real matrix M, we can write $M = UDV^T$ instead. U, V are now "real unitary" or orthogonal
  - Note that $M^T M = VD^T U^T UDV^T = VD^2 V^T$. Therefore, V are really eigenvectors of $M^T M$ with eigenvalues equal to the square of the singular values
  - Similar, we have $MM^T = UD^2 U^T$

PCA with SVD

So from previous slides, instead of first estimating the covariance matrix and then diagonalize it. We should directly decompose the data $\mathcal{X}$ with SVD instead. The process is summarized below

PCA with SVD

So from previous slides, instead of first estimating the covariance matrix and then diagonalize it. We should directly decompose the data $\mathcal{X}$ with SVD instead. The process is summarized below

- Estimate mean from data and subtract mean from that

PCA with SVD

So from previous slides, instead of first estimating the covariance matrix and then diagonalize it. We should directly decompose the data $\mathcal{X}$ with SVD instead. The process is summarized below

- Estimate mean from data and subtract mean from that
- Decomposed the mean subtracted data with SVD. We get $\mathcal{X} = UDV^T$

PCA with SVD

So from previous slides, instead of first estimating the covariance matrix and then diagonalize it. We should directly decompose the data $\mathcal{X}$ with SVD instead. The process is summarized below

- Estimate mean from data and subtract mean from that
- Decomposed the mean subtracted data with SVD. We get $\mathcal{X} = UDV^T$
- Note that column of V are now the principal components, and we can transform a data column as $V^T x$. The entire data set can be transformed as $\mathcal{Y} = \mathcal{X} V$

PCA with SVD

So from previous slides, instead of first estimating the covariance matrix and then diagonalize it. We should directly decompose the data $\mathcal{X}$ with SVD instead. The process is summarized below

- Estimate mean from data and subtract mean from that
- Decomposed the mean subtracted data with SVD. We get $\mathcal{X} = UDV^T$
- Note that column of V are now the principal components, and we can transform a data column as $V^T x$. The entire data set can be transformed as $\mathcal{Y} = \mathcal{X}V$
    - The first few columns of $\mathcal{Y}$ will contain most "information" regarding the original $\mathcal{X}$

PCA with SVD

So from previous slides, instead of first estimating the covariance matrix and then diagonalize it. We should directly decompose the data $\mathcal{X}$ with SVD instead. The process is summarized below

- Estimate mean from data and subtract mean from that
- Decomposed the mean subtracted data with SVD. We get $\mathcal{X} = UDV^T$
- Note that column of V are now the principal components, and we can transform a data column as $V^T x$. The entire data set can be transformed as $\mathcal{Y} = \mathcal{X}V$
  - The first few columns of $\mathcal{Y}$ will contain most “information” regarding the original $\mathcal{X}$
  - For example, they can be taken as features for recognition or one can omit other columns besides the first few for “compression” as discussed earlier

Marginalization of normal distribution

- Consider $\mathbf{Z} \sim \mathcal{N}(\mu_{\mathbf{Z}}, \Sigma_{\mathbf{Z}})$ and let say $\mathbf{X}$ is a segment of $\mathbf{Z}$. That is, $\mathbf{Z} = \begin{pmatrix} \mathbf{X} \\ \mathbf{Y} \end{pmatrix}$ for some $\mathbf{Y}$. Then how should $\mathbf{X}$ behave?

Marginalization of normal distribution

- Consider $\mathbf{Z} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{Z}}, \Sigma_{\mathbf{Z}})$ and let say $\mathbf{X}$ is a segment of $\mathbf{Z}$. That is, $\mathbf{Z} = \begin{pmatrix} \mathbf{X} \\ \mathbf{Y} \end{pmatrix}$ for some $\mathbf{Y}$. Then how should $\mathbf{X}$ behave?
- We can find the pdf of $\mathbf{X}$ by just marginalizing that of $\mathbf{Z}$. That is

$$
\begin{aligned}
p(\mathbf{x}) &= \int p(\mathbf{x}, \mathbf{y}) d\mathbf{y} \\
&= \frac{1}{\sqrt{\det(2\pi\Sigma)}} \int \exp\left(-\frac{1}{2} \begin{pmatrix} \mathbf{x} - \boldsymbol{\mu}_{\mathbf{X}} \\ \mathbf{y} - \boldsymbol{\mu}_{\mathbf{Y}} \end{pmatrix}^T \Sigma^{-1} \begin{pmatrix} \mathbf{x} - \boldsymbol{\mu}_{\mathbf{X}} \\ \mathbf{y} - \boldsymbol{\mu}_{\mathbf{Y}} \end{pmatrix} \right) d\mathbf{y}
\end{aligned}
$$

Marginalization of normal distribution

- Denote Σ^{-1} as Λ (also known as the precision matrix). And partition both Σ and Λ into $\Sigma = \begin{pmatrix} \Sigma_{\mathbf{XX}} & \Sigma_{\mathbf{XY}} \\ \Sigma_{\mathbf{YX}} & \Sigma_{\mathbf{YY}} \end{pmatrix}$ and $\Lambda = \begin{pmatrix} \Lambda_{\mathbf{XX}} & \Lambda_{\mathbf{XY}} \\ \Lambda_{\mathbf{YX}} & \Lambda_{\mathbf{YY}} \end{pmatrix}$

Marginalization of normal distribution

- Denote Σ^{-1} as Λ (also known as the precision matrix). And partition both Σ and Λ into $\Sigma = \begin{pmatrix} \Sigma_{\mathbf{XX}} & \Sigma_{\mathbf{XY}} \\ \Sigma_{\mathbf{YX}} & \Sigma_{\mathbf{YY}} \end{pmatrix}$ and $\Lambda = \begin{pmatrix} \Lambda_{\mathbf{XX}} & \Lambda_{\mathbf{XY}} \\ \Lambda_{\mathbf{YX}} & \Lambda_{\mathbf{YY}} \end{pmatrix}$
- Then we have

$$\begin{aligned} p(\mathbf{x}) = & \frac{1}{\sqrt{\det(2\pi\Sigma)}} \int \exp\left(-\frac{1}{2}\left[(\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}})^T \Lambda_{\mathbf{XX}} (\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}})\right.\right. \\ & + (\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}})^T \Lambda_{\mathbf{YX}} (\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}}) + (\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}})^T \Lambda_{\mathbf{XY}} (\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}}) \\ & \left.\left. +(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}})^T \Lambda_{\mathbf{YY}} (\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}})\right] \right) d\mathbf{y} \\ = & \frac{e^{-\frac{(\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}})^T \Lambda_{\mathbf{XX}} (\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}})}{2}}}{\sqrt{\det(2\pi\Sigma)}} \int \exp\left(-\frac{1}{2}\left[(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}})^T \Lambda_{\mathbf{YX}} (\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}})\right.\right. \\ & \left.\left. +(\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}})^T \Lambda_{\mathbf{XY}} (\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}}) + (\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}})^T \Lambda_{\mathbf{YY}} (\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}})\right]\right) d\mathbf{y} \end{aligned}$$

Marginalization of normal distribution

To proceed, let's apply the completing square trick on $(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}})^T\Lambda_{\mathbf{YX}}(\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}})+(\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}})^T\Lambda_{\mathbf{XY}}(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}})+(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}})^T\Lambda_{\mathbf{YY}}(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}})$. For the ease of exposition, let us denote $\tilde{\mathbf{x}}$ as $\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}}$ and $\tilde{\mathbf{y}}$ as $\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}}$. We have

Marginalization of normal distribution

To proceed, let's apply the completing square trick on $(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}})^T\Lambda_{\mathbf{YX}}(\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}})+(\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}})^T\Lambda_{\mathbf{XY}}(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}})+(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}})^T\Lambda_{\mathbf{YY}}(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}})$. For the ease of exposition, let us denote $\tilde{\mathbf{x}}$ as $\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}}$ and $\tilde{\mathbf{y}}$ as $\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}}$. We have

$$\begin{aligned}&\tilde{\mathbf{y}}^T\Lambda_{\mathbf{YX}}\tilde{\mathbf{x}}+\tilde{\mathbf{x}}^T\Lambda_{\mathbf{XY}}\tilde{\mathbf{y}}+\tilde{\mathbf{y}}^T\Lambda_{\mathbf{YY}}\tilde{\mathbf{y}}\\=&(\tilde{\mathbf{y}}+\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}}\tilde{\mathbf{x}})^T\Lambda_{\mathbf{YY}}(\tilde{\mathbf{y}}+\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}}\tilde{\mathbf{x}})-\tilde{\mathbf{x}}^T\Lambda_{\mathbf{XY}}\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}}\tilde{\mathbf{x}},\end{aligned}$$

where we use the fact that $\Lambda=\Sigma^{-1}$ is symmetric and so $\Lambda_{\mathbf{XY}}=\Lambda_{\mathbf{YX}}$

Marginalization of normal distribution

$$p(\mathbf{x}) = \frac{e^{-\frac{\tilde{\mathbf{x}}^T(\Lambda_{\mathbf{XX}}-\Lambda_{\mathbf{XY}}\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}})\tilde{\mathbf{x}}}{2}}}{\sqrt{\det(2\pi\Sigma)}} \int e^{-\frac{(\tilde{\mathbf{y}}+\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}}\tilde{\mathbf{x}})^T\Lambda_{\mathbf{YY}}(\tilde{\mathbf{y}}+\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}}\tilde{\mathbf{x}})}{2}} d\mathbf{y}$$

Marginalization of normal distribution

$$p(\mathbf{x}) = \frac{e^{-\frac{\tilde{\mathbf{x}}^T(\Lambda_{\mathbf{XX}}-\Lambda_{\mathbf{XY}}\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}})\tilde{\mathbf{x}}}{2}}}{\sqrt{\det(2\pi\Sigma)}}\int e^{-\frac{(\tilde{\mathbf{y}}+\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}}\tilde{\mathbf{x}})^T\Lambda_{\mathbf{YY}}(\tilde{\mathbf{y}}+\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}}\tilde{\mathbf{x}})}{2}}d\mathbf{y}$$

$$= \frac{\sqrt{\det(2\pi\Lambda_{\mathbf{YY}}^{-1})}}{\sqrt{\det(2\pi\Sigma)}}\exp\left(-\frac{\tilde{\mathbf{x}}^T(\Lambda_{\mathbf{XX}}-\Lambda_{\mathbf{XY}}\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}})\tilde{\mathbf{x}}}{2}\right)$$

Marginalization of normal distribution

$$
\begin{aligned}
p(\mathbf{x}) &= \frac{e^{-\frac{\tilde{\mathbf{x}}^T(\Lambda_{\mathbf{XX}}-\Lambda_{\mathbf{XY}}\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}})\tilde{\mathbf{x}}}{2}}}{\sqrt{\det(2\pi\Sigma)}} \int e^{-\frac{(\tilde{\mathbf{y}}+\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}}\tilde{\mathbf{x}})^T\Lambda_{\mathbf{YY}}(\tilde{\mathbf{y}}+\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}}\tilde{\mathbf{x}})}{2}} d\mathbf{y} \\
&= \frac{\sqrt{\det(2\pi\Lambda_{\mathbf{YY}}^{-1})}}{\sqrt{\det(2\pi\Sigma)}} \exp\left(-\frac{\tilde{\mathbf{x}}^T(\Lambda_{\mathbf{XX}}-\Lambda_{\mathbf{XY}}\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}})\tilde{\mathbf{x}}}{2}\right) \\
&\overset{(a)}{=} \frac{\sqrt{\det(2\pi\Lambda_{\mathbf{YY}}^{-1})}}{\sqrt{\det(2\pi\Sigma)}} \exp\left(-\frac{\tilde{\mathbf{x}}^T\Sigma_{\mathbf{XX}}^{-1}\tilde{\mathbf{x}}}{2}\right)
\end{aligned}
$$

Marginalization of normal distribution

$$p(\mathbf{x}) = \frac{e^{-\frac{\tilde{\mathbf{x}}^T(\Lambda_{\mathbf{XX}}-\Lambda_{\mathbf{XY}}\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}})\tilde{\mathbf{x}}}{2}}}{\sqrt{\det(2\pi\Sigma)}} \int e^{-\frac{(\tilde{\mathbf{y}}+\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}}\tilde{\mathbf{x}})^T\Lambda_{\mathbf{YY}}(\tilde{\mathbf{y}}+\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}}\tilde{\mathbf{x}})}{2}} d\mathbf{y}$$

$$= \frac{\sqrt{\det(2\pi\Lambda_{\mathbf{YY}}^{-1})}}{\sqrt{\det(2\pi\Sigma)}} \exp\left(-\frac{\tilde{\mathbf{x}}^T(\Lambda_{\mathbf{XX}}-\Lambda_{\mathbf{XY}}\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}})\tilde{\mathbf{x}}}{2}\right)$$

$$\overset{(a)}{=} \frac{\sqrt{\det(2\pi\Lambda_{\mathbf{YY}}^{-1})}}{\sqrt{\det(2\pi\Sigma)}} \exp\left(-\frac{\tilde{\mathbf{x}}^T\Sigma_{\mathbf{XX}}^{-1}\tilde{\mathbf{x}}}{2}\right)$$

$$\overset{(b)}{=} \frac{1}{\sqrt{\det(2\pi\Sigma_{\mathbf{XX}})}} \exp\left(-\frac{\tilde{\mathbf{x}}^T\Sigma_{\mathbf{XX}}^{-1}\tilde{\mathbf{x}}}{2}\right)$$

Marginalization of normal distribution

$$\begin{aligned}
p(\mathbf{x}) &= \frac{e^{-\frac{\tilde{\mathbf{x}}^T(\Lambda_{\mathbf{XX}}-\Lambda_{\mathbf{XY}}\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}})\tilde{\mathbf{x}}}{2}}}{\sqrt{\det(2\pi\Sigma)}}\int e^{-\frac{(\tilde{\mathbf{y}}+\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}}\tilde{\mathbf{x}})^T\Lambda_{\mathbf{YY}}(\tilde{\mathbf{y}}+\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}}\tilde{\mathbf{x}})}{2}}d\mathbf{y} \\
&= \frac{\sqrt{\det(2\pi\Lambda_{\mathbf{YY}}^{-1})}}{\sqrt{\det(2\pi\Sigma)}}\exp\left(-\frac{\tilde{\mathbf{x}}^T(\Lambda_{\mathbf{XX}}-\Lambda_{\mathbf{XY}}\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}})\tilde{\mathbf{x}}}{2}\right) \\
&\overset{(a)}{=} \frac{\sqrt{\det(2\pi\Lambda_{\mathbf{YY}}^{-1})}}{\sqrt{\det(2\pi\Sigma)}}\exp\left(-\frac{\tilde{\mathbf{x}}^T\Sigma_{\mathbf{XX}}^{-1}\tilde{\mathbf{x}}}{2}\right) \\
&\overset{(b)}{=} \frac{1}{\sqrt{\det(2\pi\Sigma_{\mathbf{XX}})}}\exp\left(-\frac{\tilde{\mathbf{x}}^T\Sigma_{\mathbf{XX}}^{-1}\tilde{\mathbf{x}}}{2}\right) \\
&= \frac{1}{\sqrt{\det(2\pi\Sigma_{\mathbf{XX}})}}\exp\left(-\frac{(\mathbf{x}-\mu_{\mathbf{X}})^T\Sigma_{\mathbf{XX}}^{-1}(\mathbf{x}-\mu_{\mathbf{X}})}{2}\right),
\end{aligned}$$

where (a) and (b) will be shown next

(a) $\Sigma_{\mathbf{XX}}^{-1} = \Lambda_{\mathbf{XX}} - \Lambda_{\mathbf{XY}}\Lambda_{\mathbf{YY}}^{-1}\Lambda_{\mathbf{YX}}$

Lemma

Assume $\begin{pmatrix} A & B \\ C & D \end{pmatrix}^{-1} = \begin{pmatrix} \tilde{A} & \tilde{B} \\ \tilde{C} & \tilde{D} \end{pmatrix}$*, then* $A^{-1} = \tilde{A} - \tilde{B}\tilde{D}^{-1}\tilde{C}$

Proof.

Note that $\begin{pmatrix} A & B \\ C & D \end{pmatrix}\begin{pmatrix} \tilde{A} & \tilde{B} \\ \tilde{C} & \tilde{D} \end{pmatrix} = \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix}$. Thus $A\tilde{A} + B\tilde{C} = I$ and $A\tilde{B} + B\tilde{D} = 0$. So $A(\tilde{A} - \tilde{B}\tilde{D}^{-1}\tilde{C}) = A\tilde{A} - (A\tilde{B})\tilde{D}^{-1}\tilde{C} = A\tilde{A} + B\tilde{D}\tilde{D}^{-1}\tilde{C} = A\tilde{A} + B\tilde{C} = I$ □

(b) $\det(a\Sigma) = \det(a\Sigma_{\mathbf{YY}})\det(a\Lambda_{\mathbf{XX}}^{-1})$

Lemma

Assume $\begin{pmatrix} A & B \\ C & D \end{pmatrix}^{-1} = \begin{pmatrix} \tilde{A} & \tilde{B} \\ \tilde{C} & \tilde{D} \end{pmatrix}$, *then det* $\begin{pmatrix} A & B \\ C & D \end{pmatrix} = det(D)det(\tilde{A}^{-1})$

Proof.

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix} = \begin{pmatrix} I & 0 \\ 0 & D \end{pmatrix}\begin{pmatrix} A & B \\ D^{-1}C & I \end{pmatrix} = \begin{pmatrix} I & 0 \\ 0 & D \end{pmatrix}\begin{pmatrix} I & B \\ 0 & I \end{pmatrix}\begin{pmatrix} A - BD^{-1}C & 0 \\ D^{-1}C & I \end{pmatrix}$$

$$\Rightarrow det \begin{pmatrix} A & B \\ C & D \end{pmatrix} = det(D)det(A - BD^{-1}C) = det(D)det(\tilde{A}^{-1})$$ ☐

Remark

N.B. $A - BD^{-1}C$ *is known as Schur complement*

Conditioning of normal distribution

- Consider the same $\mathbf{Z} \sim \mathcal{N}(\mu_{\mathbf{Z}}, \Sigma_{\mathbf{Z}})$ and $\mathbf{Z} = \begin{pmatrix} \mathbf{X} \\ \mathbf{Y} \end{pmatrix}$. What will $\mathbf{X}$ be like if $\mathbf{Y}$ is observed to be $\mathbf{y}$?

Conditioning of normal distribution

- Consider the same $\mathbf{Z} \sim \mathcal{N}(\mu_\mathbf{Z}, \Sigma_\mathbf{Z})$ and $\mathbf{Z} = \begin{pmatrix} \mathbf{X} \\ \mathbf{Y} \end{pmatrix}$. What will $\mathbf{X}$ be like if $\mathbf{Y}$ is observed to be $\mathbf{y}$?
- Basically, we want to find $p(\mathbf{x}|\mathbf{y}) = p(\mathbf{x}, \mathbf{y})/p(\mathbf{y})$

Conditioning of normal distribution

- Consider the same $\mathbf{Z} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{Z}}, \Sigma_{\mathbf{Z}})$ and $\mathbf{Z} = \begin{pmatrix} \mathbf{X} \\ \mathbf{Y} \end{pmatrix}$. What will $\mathbf{X}$ be like if $\mathbf{Y}$ is observed to be $\mathbf{y}$?
- Basically, we want to find $p(\mathbf{x}|\mathbf{y}) = p(\mathbf{x}, \mathbf{y})/p(\mathbf{y})$
- From previous result, we have $p(\mathbf{y}) = \mathcal{N}(\mathbf{y}; \boldsymbol{\mu}_{\mathbf{Y}}, \Sigma_{\mathbf{YY}})$. Therefore,

$$\begin{aligned} p(\mathbf{x}|\mathbf{y}) &\propto \exp\left(-\frac{1}{2}\left[\begin{pmatrix} \tilde{\mathbf{x}} \\ \tilde{\mathbf{y}} \end{pmatrix}^T \Sigma^{-1} \begin{pmatrix} \tilde{\mathbf{x}} \\ \tilde{\mathbf{y}} \end{pmatrix} - \tilde{\mathbf{y}}^T \Sigma_{\mathbf{YY}}^{-1} \tilde{\mathbf{y}}\right]\right) \\ &\propto \exp\left(-\frac{1}{2}[\tilde{\mathbf{x}}^T \Lambda_{\mathbf{XX}} \tilde{\mathbf{x}} + \tilde{\mathbf{x}}^T \Lambda_{\mathbf{XY}} \tilde{\mathbf{y}} + \tilde{\mathbf{y}}^T \Lambda_{\mathbf{YX}} \tilde{\mathbf{x}}]\right), \end{aligned}$$

where we use $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ as shorthands of $\mathbf{x} - \boldsymbol{\mu}_{\mathbf{X}}$ and $\mathbf{y} - \boldsymbol{\mu}_{\mathbf{Y}}$ as before

Conditioning of normal distribution

- Completing the square for $\tilde{\mathbf{x}}$, we have

$$
\begin{aligned}
p(\mathbf{x}|\mathbf{y}) &\propto \exp\left(-\frac{1}{2}(\tilde{\mathbf{x}} + \Lambda_{\mathbf{XX}}^{-1}\Lambda_{\mathbf{XY}}\tilde{\mathbf{y}})^T\Lambda_{\mathbf{XX}}(\tilde{\mathbf{x}} + \Lambda_{\mathbf{XX}}^{-1}\Lambda_{\mathbf{XY}}\tilde{\mathbf{y}})\right) \\
&= \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_{\mathbf{X}} + \Lambda_{\mathbf{XX}}^{-1}\Lambda_{\mathbf{XY}}(\mathbf{y} - \boldsymbol{\mu}_{\mathbf{Y}}))^T\Lambda_{\mathbf{XX}}\right. \\
&\qquad\qquad \left.(\mathbf{x} - \boldsymbol{\mu}_{\mathbf{X}} + \Lambda_{\mathbf{XX}}^{-1}\Lambda_{\mathbf{XY}}(\mathbf{y} - \boldsymbol{\mu}_{\mathbf{Y}}))\right)
\end{aligned}
$$

Conditioning of normal distribution

- Completing the square for $\tilde{\mathbf{x}}$, we have

$$\begin{aligned} p(\mathbf{x}|\mathbf{y}) &\propto \exp\left(-\frac{1}{2}(\tilde{\mathbf{x}}+\Lambda_{\mathbf{XX}}^{-1}\Lambda_{\mathbf{XY}}\tilde{\mathbf{y}})^T\Lambda_{\mathbf{XX}}(\tilde{\mathbf{x}}+\Lambda_{\mathbf{XX}}^{-1}\Lambda_{\mathbf{XY}}\tilde{\mathbf{y}})\right) \\ &= \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}}+\Lambda_{\mathbf{XX}}^{-1}\Lambda_{\mathbf{XY}}(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}}))^T\Lambda_{\mathbf{XX}}\right. \\ &\qquad \left.(\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}}+\Lambda_{\mathbf{XX}}^{-1}\Lambda_{\mathbf{XY}}(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}}))\right) \end{aligned}$$

- Therefore $\mathbf{X}|\mathbf{y}$ is Gaussian distributed with mean $\boldsymbol{\mu}_{\mathbf{X}}-\Lambda_{\mathbf{XX}}^{-1}\Lambda_{\mathbf{XY}}(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}})$ and covariance $\Lambda_{\mathbf{XX}}^{-1}$

Conditioning of normal distribution

- Completing the square for $\tilde{\mathbf{x}}$, we have

$$p(\mathbf{x}|\mathbf{y}) \propto \exp\left(-\frac{1}{2}(\tilde{\mathbf{x}}+\Lambda_{\mathbf{XX}}^{-1}\Lambda_{\mathbf{XY}}\tilde{\mathbf{y}})^T\Lambda_{\mathbf{XX}}(\tilde{\mathbf{x}}+\Lambda_{\mathbf{XX}}^{-1}\Lambda_{\mathbf{XY}}\tilde{\mathbf{y}})\right)$$
$$= \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}}+\Lambda_{\mathbf{XX}}^{-1}\Lambda_{\mathbf{XY}}(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}}))^T\Lambda_{\mathbf{XX}}\right.$$
$$\left.(\mathbf{x}-\boldsymbol{\mu}_{\mathbf{X}}+\Lambda_{\mathbf{XX}}^{-1}\Lambda_{\mathbf{XY}}(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}}))\right)$$

- Therefore $\mathbf{X}|\mathbf{y}$ is Gaussian distributed with mean $\boldsymbol{\mu}_{\mathbf{X}} - \Lambda_{\mathbf{XX}}^{-1}\Lambda_{\mathbf{XY}}(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}})$ and covariance $\Lambda_{\mathbf{XX}}^{-1}$
- Note that since $\Lambda_{\mathbf{XX}}\Sigma_{\mathbf{XY}}+\Lambda_{\mathbf{XY}}\Sigma_{\mathbf{YY}}=0 \Rightarrow \Lambda_{\mathbf{XX}}^{-1}\Lambda_{\mathbf{XY}} = -\Sigma_{\mathbf{XY}}\Sigma_{\mathbf{YY}}^{-1}$ and from (a), we have

$$\mathbf{X}|\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{X}}+\Sigma_{\mathbf{XY}}\Sigma_{\mathbf{YY}}^{-1}(\mathbf{y}-\boldsymbol{\mu}_{\mathbf{Y}}), \Sigma_{\mathbf{XX}}-\Sigma_{\mathbf{XY}}\Sigma_{\mathbf{YY}}^{-1}\Sigma_{\mathbf{YX}}),$$

where $\Sigma_{\mathbf{XX}}-\Sigma_{\mathbf{XY}}\Sigma_{\mathbf{YY}}^{-1}\Sigma_{\mathbf{YX}} \triangleq \Sigma|\Sigma_{\mathbf{YY}}$ is a Schur complement

Interpretation of conditioning

$$\mathbf{X}|\mathbf{y} \sim \mathcal{N}(\mu_{\mathbf{X}} + \Sigma_{\mathbf{XY}}\Sigma_{\mathbf{YY}}^{-1}(\mathbf{y} - \mu_{\mathbf{Y}}), \Sigma_{\mathbf{XX}} - \Sigma_{\mathbf{XY}}\Sigma_{\mathbf{YY}}^{-1}\Sigma_{\mathbf{YX}})$$

- When the observation of **Y** is exactly the mean, the conditioned mean does not change

Interpretation of conditioning

$$\mathbf{X}|\mathbf{y} \sim \mathcal{N}(\mu_{\mathbf{X}} + \Sigma_{\mathbf{XY}}\Sigma_{\mathbf{YY}}^{-1}(\mathbf{y} - \mu_{\mathbf{Y}}), \Sigma_{\mathbf{XX}} - \Sigma_{\mathbf{XY}}\Sigma_{\mathbf{YY}}^{-1}\Sigma_{\mathbf{YX}})$$

- When the observation of **Y** is exactly the mean, the conditioned mean does not change
- Otherwise, it needs to be modified and the size of the adjustment decreases with $\Sigma_{\mathbf{YY}}$, the variance of **Y** for the 1-D case.
  - The observation is less reliable with the increase of $\Sigma_{\mathbf{YY}}$. The adjustment is finally scaled by $\Sigma_{\mathbf{XY}}$, which translates the variation of **Y** to the variation of **X**

Interpretation of conditioning

$$\mathbf{X}|\mathbf{y} \sim \mathcal{N}(\mu_{\mathbf{X}} + \Sigma_{\mathbf{XY}}\Sigma_{\mathbf{YY}}^{-1}(\mathbf{y} - \mu_{\mathbf{Y}}), \Sigma_{\mathbf{XX}} - \Sigma_{\mathbf{XY}}\Sigma_{\mathbf{YY}}^{-1}\Sigma_{\mathbf{YX}})$$

- When the observation of **Y** is exactly the mean, the conditioned mean does not change
- Otherwise, it needs to be modified and the size of the adjustment decreases with $\Sigma_{\mathbf{YY}}$, the variance of **Y** for the 1-D case.
  - The observation is less reliable with the increase of $\Sigma_{\mathbf{YY}}$. The adjustment is finally scaled by $\Sigma_{\mathbf{XY}}$, which translates the variation of **Y** to the variation of **X**
  - In particular, if **X** and **Y** are negatively correlated, the sign of the adjustment will be reversed

Interpretation of conditioning

$$\mathbf{X}|\mathbf{y} \sim \mathcal{N}(\mu_{\mathbf{X}} + \Sigma_{\mathbf{XY}}\Sigma_{\mathbf{YY}}^{-1}(\mathbf{y} - \mu_{\mathbf{Y}}), \Sigma_{\mathbf{XX}} - \Sigma_{\mathbf{XY}}\Sigma_{\mathbf{YY}}^{-1}\Sigma_{\mathbf{YX}})$$

- When the observation of **Y** is exactly the mean, the conditioned mean does not change
- Otherwise, it needs to be modified and the size of the adjustment decreases with $\Sigma_{\mathbf{YY}}$, the variance of **Y** for the 1-D case.
  - The observation is less reliable with the increase of $\Sigma_{\mathbf{YY}}$. The adjustment is finally scaled by $\Sigma_{\mathbf{XY}}$, which translates the variation of **Y** to the variation of **X**
  - In particular, if **X** and **Y** are negatively correlated, the sign of the adjustment will be reversed
- As for the variance of the conditioned variable, it always decreases and the decrease is larger if $\Sigma_{\mathbf{YY}}$ is smaller and $\Sigma_{\mathbf{XY}}$ is larger (**X** and **Y** are more correlated)

Uncorrelated implies independence

$$\mathbf{X}|\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{X}} + \Sigma_{\mathbf{XY}}\Sigma_{\mathbf{YY}}^{-1}(\mathbf{y} - \boldsymbol{\mu}_{\mathbf{Y}}), \Sigma_{\mathbf{XX}} - \Sigma_{\mathbf{XY}}\Sigma_{\mathbf{YY}}^{-1}\Sigma_{\mathbf{YX}})$$

If $\mathbf{X}$ and $\mathbf{Y}$ are uncorrelated, $\Sigma_{\mathbf{XY}} = 0$. Then

$$\mathbf{X}|\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{X}}, \Sigma_{\mathbf{XX}})$$

Note that the statistics of $\mathbf{X}$ does not change with respect to $\mathbf{y}$ and so $\mathbf{X}$ is independent of $\mathbf{Y}$

$X \perp\!\!\!\perp Y|Z$ if $\rho_{XZ}\rho_{YZ} = \rho_{XY}$

Corollary

Given multivariate Gaussian variables X, Y and Z, we have X and Y are conditionally independent given Z if $\rho_{XZ}\rho_{YZ} = \rho_{XY}$, where $\rho_{XZ} = \frac{E[(X-E(X))(Z-E(Z))]}{\sqrt{E[(X-E(X))^2]E[(Z-E(Z))^2]}}$ is the correlation coefficent between X and Z. Similarly, ρ_{YZ} and ρ_{XY} are the correlation coefficients between Y and Z, and X and Y, respectively.

$X \perp\!\!\!\perp Y | Z$ if $\rho_{XZ}\rho_{YZ} = \rho_{XY}$

Proof.

- From the definition of correlation coefficient, $\Sigma = \begin{pmatrix} \sigma_{XX} & \sqrt{\sigma_{XX}\sigma_{YY}}\rho_{XY} & \sqrt{\sigma_{XX}\sigma_{ZZ}}\rho_{XZ} \\ \sqrt{\sigma_{XX}\sigma_{YY}}\rho_{XY} & \sigma_{YY} & \sqrt{\sigma_{YY}\sigma_{ZZ}}\rho_{YZ} \\ \sqrt{\sigma_{XX}\sigma_{ZZ}}\rho_{XZ} & \sqrt{\sigma_{YY}\sigma_{ZZ}}\rho_{YZ} & \sigma_{ZZ} \end{pmatrix}$

$X \perp\!\!\!\perp Y|Z$ if $\rho_{XZ}\rho_{YZ} = \rho_{XY}$

Proof.

- From the definition of correlation coefficient, $\Sigma = \begin{pmatrix} \sigma_{XX} & \sqrt{\sigma_{XX}\sigma_{YY}}\rho_{XY} & \sqrt{\sigma_{XX}\sigma_{ZZ}}\rho_{XZ} \\ \sqrt{\sigma_{XX}\sigma_{YY}}\rho_{XY} & \sigma_{YY} & \sqrt{\sigma_{YY}\sigma_{ZZ}}\rho_{YZ} \\ \sqrt{\sigma_{XX}\sigma_{ZZ}}\rho_{XZ} & \sqrt{\sigma_{YY}\sigma_{ZZ}}\rho_{YZ} & \sigma_{ZZ} \end{pmatrix}$
- Then from the conditioning result, we have

$$
\begin{aligned}
\Sigma_{\left(\begin{smallmatrix} X \\ Y \end{smallmatrix}\right)|Z} &= \begin{pmatrix} \sigma_{XX} & \sqrt{\sigma_{XX}\sigma_{YY}}\rho_{XY} \\ \sqrt{\sigma_{XX}\sigma_{YY}}\rho_{XY} & \sigma_{YY} \end{pmatrix} \\
&\qquad - \begin{pmatrix} \sqrt{\sigma_{XX}\sigma_{ZZ}}\rho_{XZ} & \sqrt{\sigma_{YY}\sigma_{ZZ}}\rho_{YZ} \end{pmatrix} \sigma_{ZZ}^{-1} \begin{pmatrix} \sqrt{\sigma_{XX}\sigma_{ZZ}}\rho_{XZ} \\ \sqrt{\sigma_{YY}\sigma_{ZZ}}\rho_{YZ} \end{pmatrix} \\
&= \begin{pmatrix} \sigma_{XX}(1-\rho_{XZ}^2) & \sqrt{\sigma_{XX}\sigma_{YY}}(\rho_{XY} - \rho_{XZ}\rho_{YZ}) \\ \sqrt{\sigma_{XX}\sigma_{YY}}(\rho_{XY} - \rho_{XZ}\rho_{YZ}) & \sigma_{YY}(1-\rho_{YZ}^2) \end{pmatrix}
\end{aligned}
$$

$X \perp Y | Z$ if $\rho_{XZ}\rho_{YZ} = \rho_{XY}$

Proof.

- From the definition of correlation coefficient, $\Sigma = \begin{pmatrix} \sigma_{XX} & \sqrt{\sigma_{XX}\sigma_{YY}}\rho_{XY} & \sqrt{\sigma_{XX}\sigma_{ZZ}}\rho_{XZ} \\ \sqrt{\sigma_{XX}\sigma_{YY}}\rho_{XY} & \sigma_{YY} & \sqrt{\sigma_{YY}\sigma_{ZZ}}\rho_{YZ} \\ \sqrt{\sigma_{XX}\sigma_{ZZ}}\rho_{XZ} & \sqrt{\sigma_{YY}\sigma_{ZZ}}\rho_{YZ} & \sigma_{ZZ} \end{pmatrix}$
- Then from the conditioning result, we have

$$
\begin{aligned}
\Sigma_{\left(\begin{smallmatrix} X \\ Y \end{smallmatrix}\right)|Z} &= \begin{pmatrix} \sigma_{XX} & \sqrt{\sigma_{XX}\sigma_{YY}}\rho_{XY} \\ \sqrt{\sigma_{XX}\sigma_{YY}}\rho_{XY} & \sigma_{YY} \end{pmatrix} \\
&\quad - \begin{pmatrix} \sqrt{\sigma_{XX}\sigma_{ZZ}}\rho_{XZ} & \sqrt{\sigma_{YY}\sigma_{ZZ}}\rho_{YZ} \end{pmatrix} \sigma_{ZZ}^{-1} \begin{pmatrix} \sqrt{\sigma_{XX}\sigma_{ZZ}}\rho_{XZ} \\ \sqrt{\sigma_{YY}\sigma_{ZZ}}\rho_{YZ} \end{pmatrix} \\
&= \begin{pmatrix} \sigma_{XX}(1-\rho_{XZ}^2) & \sqrt{\sigma_{XX}\sigma_{YY}}(\rho_{XY}-\rho_{XZ}\rho_{YZ}) \\ \sqrt{\sigma_{XX}\sigma_{YY}}(\rho_{XY}-\rho_{XZ}\rho_{YZ}) & \sigma_{YY}(1-\rho_{YZ}^2) \end{pmatrix}
\end{aligned}
$$

- Therefore, X and Y are uncorrelated given Z when the off-diagonal is zero and this gives us $\rho_{XY} = \rho_{XZ}\rho_{YZ}$. Since for Gaussian variables, uncorrelatedness implies independence. This concludes the proof.

Gaussian Process

- Consider a 1-D discrete-time signal, and say the signal is joint Gaussian and two points are conditional independent given points in the middle
- If the variance is stationary and say the correlation coefficent between two adjacent points is ρ, further assume that the variance is normalized to 1. WLOG, then

$$\Sigma = \begin{pmatrix} 1 & \rho & \rho^2 & \cdots & \\ \rho & 1 & \rho & \rho^2 & \cdots \\ \rho^2 & \rho & 1 & \rho & \cdots \\ & & \cdots & & \end{pmatrix}$$

Product of normal distributions

- Assume that we tries to recover some vector parameter $\mathbf{x}$, which is subject to multivariate Gaussian noise

Product of normal distributions

- Assume that we tries to recover some vector parameter $\mathbf{x}$, which is subject to multivariate Gaussian noise
- Say we made two measurements $\mathbf{y}_1$ and $\mathbf{y}_2$, where $\mathbf{Y}_1 \sim \mathcal{N}(\mathbf{x}, \Sigma_{\mathbf{Y}_1})$ and $\mathbf{Y}_2 \sim \mathcal{N}(\mathbf{x}, \Sigma_{\mathbf{Y}_2})$. Note that even though both measurements have mean $\mathbf{x}$, they have different covariance
    - This variation, for instance, can be due to environment change between the two measurements

Product of normal distributions

- Assume that we tries to recover some vector parameter $\mathbf{x}$, which is subject to multivariate Gaussian noise
- Say we made two measurements $\mathbf{y}_1$ and $\mathbf{y}_2$, where $\mathbf{Y}_1 \sim \mathcal{N}(\mathbf{x}, \Sigma_{\mathbf{Y}_1})$ and $\mathbf{Y}_2 \sim \mathcal{N}(\mathbf{x}, \Sigma_{\mathbf{Y}_2})$. Note that even though both measurements have mean $\mathbf{x}$, they have different covariance
    - This variation, for instance, can be due to environment change between the two measurements
- Now, if we want to compute the overall likelihood, $p(\mathbf{y}_1, \mathbf{y}_2|\mathbf{x})$. Assuming that $\mathbf{Y}_1$ and $\mathbf{Y}_2$ are conditionally independent given $\mathbf{X}$, we have

$$\begin{aligned} p(\mathbf{y}_1, \mathbf{y}_2|\mathbf{x}) &= p(\mathbf{y}_1|\mathbf{x})p(\mathbf{y}_2|\mathbf{x}) \\ &= \mathcal{N}(\mathbf{y}_1; \mathbf{x}, \Sigma_{\mathbf{Y}_1})\mathcal{N}(\mathbf{y}_2; \mathbf{x}, \Sigma_{\mathbf{Y}_2}). \end{aligned}$$

Product of normal distributions

- Assume that we tries to recover some vector parameter $\mathbf{x}$, which is subject to multivariate Gaussian noise
- Say we made two measurements $\mathbf{y}_1$ and $\mathbf{y}_2$, where $\mathbf{Y}_1 \sim \mathcal{N}(\mathbf{x}, \Sigma_{\mathbf{Y}_1})$ and $\mathbf{Y}_2 \sim \mathcal{N}(\mathbf{x}, \Sigma_{\mathbf{Y}_2})$. Note that even though both measurements have mean $\mathbf{x}$, they have different covariance
  - This variation, for instance, can be due to environment change between the two measurements
- Now, if we want to compute the overall likelihood, $p(\mathbf{y}_1, \mathbf{y}_2|\mathbf{x})$. Assuming that $\mathbf{Y}_1$ and $\mathbf{Y}_2$ are conditionally independent given $\mathbf{X}$, we have

$$\begin{aligned} p(\mathbf{y}_1, \mathbf{y}_2|\mathbf{x}) &= p(\mathbf{y}_1|\mathbf{x})p(\mathbf{y}_2|\mathbf{x}) \\ &= \mathcal{N}(\mathbf{y}_1; \mathbf{x}, \Sigma_{\mathbf{Y}_1})\mathcal{N}(\mathbf{y}_2; \mathbf{x}, \Sigma_{\mathbf{Y}_2}). \end{aligned}$$

- Essentially, we just need to compute the product of two Gaussian pdfs. Such computation is very useful and it occurs often when one needs to perform inference

Product of normal distributions

As in previous cases, the product turns out to be normal also. However, unlike them, the product is not a pdf and so it does not normalize to 1. So we have to compute both the scaling factor and the exponent explicitly. Let us start with the exponent.

$$\mathcal{N}(\mathbf{y}_1; \mathbf{x}, \Sigma_{\mathbf{Y}_1})\mathcal{N}(\mathbf{y}_2; \mathbf{x}, \Sigma_{\mathbf{Y}_2})$$

Product of normal distributions

As in previous cases, the product turns out to be normal also. However, unlike them, the product is not a pdf and so it does not normalize to 1. So we have to compute both the scaling factor and the exponent explicitly. Let us start with the exponent.

$$\begin{aligned}&\mathcal{N}(\mathbf{y}_1;\mathbf{x},\Sigma_{\mathbf{Y}_1})\mathcal{N}(\mathbf{y}_2;\mathbf{x},\Sigma_{\mathbf{Y}_2})\\ \propto &\exp\left(-\frac{1}{2}[(\mathbf{x}-\mathbf{y}_1)^T\Lambda_{\mathbf{Y}_1}(\mathbf{x}-\mathbf{y}_1)+(\mathbf{x}-\mathbf{y}_2)^T\Lambda_{\mathbf{Y}_2}(\mathbf{x}-\mathbf{y}_2)]\right)\end{aligned}$$

Product of normal distributions

As in previous cases, the product turns out to be normal also. However, unlike them, the product is not a pdf and so it does not normalize to 1. So we have to compute both the scaling factor and the exponent explicitly. Let us start with the exponent.

$$\begin{aligned}
&\mathcal{N}(\mathbf{y}_1; \mathbf{x}, \Sigma_{\mathbf{Y}_1})\mathcal{N}(\mathbf{y}_2; \mathbf{x}, \Sigma_{\mathbf{Y}_2}) \\
\propto &\exp\left(-\frac{1}{2}[(\mathbf{x}-\mathbf{y}_1)^T\Lambda_{\mathbf{Y}_1}(\mathbf{x}-\mathbf{y}_1)+(\mathbf{x}-\mathbf{y}_2)^T\Lambda_{\mathbf{Y}_2}(\mathbf{x}-\mathbf{y}_2)]\right) \\
\propto &\exp\left(-\frac{1}{2}[\mathbf{x}^T(\Lambda_{\mathbf{Y}_1}+\Lambda_{\mathbf{Y}_2})\mathbf{x}-(\mathbf{y}_2^T\Lambda_{\mathbf{Y}_2}+\mathbf{y}_1^T\Lambda_{\mathbf{Y}_1})\mathbf{x}-\mathbf{x}^T(\Lambda_{\mathbf{Y}_2}\mathbf{y}_2+\Lambda_{\mathbf{Y}_1}\mathbf{y}_1)]\right)
\end{aligned}$$

Product of normal distributions

As in previous cases, the product turns out to be normal also. However, unlike them, the product is not a pdf and so it does not normalize to 1. So we have to compute both the scaling factor and the exponent explicitly. Let us start with the exponent.

$$\begin{aligned}
&\mathcal{N}(\mathbf{y}_1;\mathbf{x},\Sigma_{\mathbf{Y}_1})\mathcal{N}(\mathbf{y}_2;\mathbf{x},\Sigma_{\mathbf{Y}_2})\\
\propto&\exp\left(-\frac{1}{2}[(\mathbf{x}-\mathbf{y}_1)^T\Lambda_{\mathbf{Y}_1}(\mathbf{x}-\mathbf{y}_1)+(\mathbf{x}-\mathbf{y}_2)^T\Lambda_{\mathbf{Y}_2}(\mathbf{x}-\mathbf{y}_2)]\right)\\
\propto&\exp\left(-\frac{1}{2}[\mathbf{x}^T(\Lambda_{\mathbf{Y}_1}+\Lambda_{\mathbf{Y}_2})\mathbf{x}-(\mathbf{y}_2^T\Lambda_{\mathbf{Y}_2}+\mathbf{y}_1^T\Lambda_{\mathbf{Y}_1})\mathbf{x}-\mathbf{x}^T(\Lambda_{\mathbf{Y}_2}\mathbf{y}_2+\Lambda_{\mathbf{Y}_1}\mathbf{y}_1)]\right)\\
\propto&e^{-\frac{1}{2}[(\mathbf{x}-(\Lambda_{\mathbf{Y}_1}+\Lambda_{\mathbf{Y}_2})^{-1}(\Lambda_{\mathbf{Y}_2}\mathbf{y}_2+\Lambda_{\mathbf{Y}_1}\mathbf{y}_1))^T(\Lambda_{\mathbf{Y}_1}+\Lambda_{\mathbf{Y}_2})(\mathbf{x}-(\Lambda_{\mathbf{Y}_1}+\Lambda_{\mathbf{Y}_2})^{-1}(\Lambda_{\mathbf{Y}_2}\mathbf{y}_2+\Lambda_{\mathbf{Y}_1}\mathbf{y}_1))]}
\end{aligned}$$

Product of normal distributions

As in previous cases, the product turns out to be normal also. However, unlike them, the product is not a pdf and so it does not normalize to 1. So we have to compute both the scaling factor and the exponent explicitly. Let us start with the exponent.

$$
\begin{aligned}
&\mathcal{N}(\mathbf{y}_1;\mathbf{x},\Sigma_{\mathbf{Y}_1})\mathcal{N}(\mathbf{y}_2;\mathbf{x},\Sigma_{\mathbf{Y}_2})\\
\propto&\exp\left(-\frac{1}{2}[(\mathbf{x}-\mathbf{y}_1)^T\Lambda_{\mathbf{Y}_1}(\mathbf{x}-\mathbf{y}_1)+(\mathbf{x}-\mathbf{y}_2)^T\Lambda_{\mathbf{Y}_2}(\mathbf{x}-\mathbf{y}_2)]\right)\\
\propto&\exp\left(-\frac{1}{2}[\mathbf{x}^T(\Lambda_{\mathbf{Y}_1}+\Lambda_{\mathbf{Y}_2})\mathbf{x}-(\mathbf{y}_2^T\Lambda_{\mathbf{Y}_2}+\mathbf{y}_1^T\Lambda_{\mathbf{Y}_1})\mathbf{x}-\mathbf{x}^T(\Lambda_{\mathbf{Y}_2}\mathbf{y}_2+\Lambda_{\mathbf{Y}_1}\mathbf{y}_1)]\right)\\
\propto& e^{-\frac{1}{2}[(\mathbf{x}-(\Lambda_{\mathbf{Y}_1}+\Lambda_{\mathbf{Y}_2})^{-1}(\Lambda_{\mathbf{Y}_2}\mathbf{y}_2+\Lambda_{\mathbf{Y}_1}\mathbf{y}_1))^T(\Lambda_{\mathbf{Y}_1}+\Lambda_{\mathbf{Y}_2})(\mathbf{x}-(\Lambda_{\mathbf{Y}_1}+\Lambda_{\mathbf{Y}_2})^{-1}(\Lambda_{\mathbf{Y}_2}\mathbf{y}_2+\Lambda_{\mathbf{Y}_1}\mathbf{y}_1))]}\\
\propto&\mathcal{N}(\mathbf{x};(\Lambda_{\mathbf{Y}_1}+\Lambda_{\mathbf{Y}_2})^{-1}(\Lambda_{\mathbf{Y}_2}\mathbf{y}_2+\Lambda_{\mathbf{Y}_1}\mathbf{y}_1),(\Lambda_{\mathbf{Y}_2}+\Lambda_{\mathbf{Y}_1})^{-1})
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
&\mathcal{N}(\mathbf{y}_1;\mathbf{x},\Sigma_{\mathbf{Y}_1})\mathcal{N}(\mathbf{y}_2;\mathbf{x},\Sigma_{\mathbf{Y}_2})\\
=&K(\mathbf{y}_1,\mathbf{y}_2,\Sigma_{\mathbf{Y}_1},\Sigma_{\mathbf{Y}_2})\mathcal{N}(\mathbf{x};(\Lambda_{\mathbf{Y}_1}+\Lambda_{\mathbf{Y}_2})^{-1}(\Lambda_{\mathbf{Y}_2}\mathbf{y}_2+\Lambda_{\mathbf{Y}_1}\mathbf{y}_1),(\Lambda_{\mathbf{Y}_2}+\Lambda_{\mathbf{Y}_1})^{-1})
\end{aligned}
$$

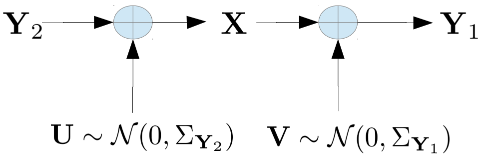
for some scaling factor $K(\mathbf{y}_1,\mathbf{y}_2,\Sigma_{\mathbf{Y}_1},\Sigma_{\mathbf{Y}_2})$ independent of $\mathbf{x}$

Product of normal distributions

- One can compute the scaling factor $K(\mathbf{y}_1, \mathbf{y}_2, \Sigma_{\mathbf{Y}_1}, \Sigma_{\mathbf{Y}_2})$ directly

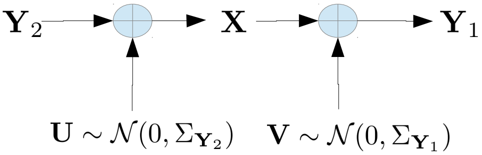
Product of normal distributions

- One can compute the scaling factor $K(\mathbf{y}_1, \mathbf{y}_2, \Sigma_{\mathbf{Y}_1}, \Sigma_{\mathbf{Y}_2})$ directly
- However, it is much easier to take advantage for the following setup when $\mathbf{Y}_1 \perp\!\!\!\perp \mathbf{Y}_2 | \mathbf{X}$ as shown below



Product of normal distributions

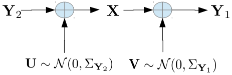
- One can compute the scaling factor $K(\mathbf{y}_1, \mathbf{y}_2, \Sigma_{\mathbf{Y}_1}, \Sigma_{\mathbf{Y}_2})$ directly
- However, it is much easier to take advantage for the following setup when $\mathbf{Y}_1 \perp\!\!\!\perp \mathbf{Y}_2 | \mathbf{X}$ as shown below



- Since $\mathcal{N}(\mathbf{y}_2; \mathbf{x}, \Sigma_{\mathbf{Y}_2}) = \mathcal{N}(\mathbf{x}; \mathbf{y}_2, \Sigma_{\mathbf{Y}_2})$ and $\mathbf{Y}_1 \perp\!\!\!\perp \mathbf{Y}_2 | \mathbf{X}$, we have

$$\mathcal{N}(\mathbf{y}_1; \mathbf{x}, \Sigma_{\mathbf{Y}_1}) \mathcal{N}(\mathbf{y}_2; \mathbf{x}, \Sigma_{\mathbf{Y}_2}) = \underbrace{\mathcal{N}(\mathbf{y}_1; \mathbf{x}, \Sigma_{\mathbf{Y}_1})}_{p(y_1|x)=p(y_1|x,y_2)} \underbrace{\mathcal{N}(\mathbf{x}; \mathbf{y}_2, \Sigma_{\mathbf{Y}_2})}_{p(x|y_2)} = p(\mathbf{y}_1, \mathbf{x} | \mathbf{y}_2)$$

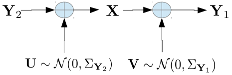
Product of normal distributions

- Then, marginalizing **x** out from $p(\mathbf{y}_1, \mathbf{x}|\mathbf{y}_2)$, we have $p(\mathbf{y}_1|\mathbf{y}_2) = \int p(\mathbf{y}_1, \mathbf{x}|\mathbf{y}_2) d\mathbf{x}$. However, from the figure, 

$$\int p(\mathbf{y}_1, \mathbf{x}|\mathbf{y}_2) d\mathbf{x} = p(\mathbf{y}_1|\mathbf{y}_2) = \mathcal{N}(\mathbf{y}_1; \mathbf{y}_2, \Sigma_{\mathbf{Y}_2} + \Sigma_{\mathbf{Y}_1})$$

Product of normal distributions

- Then, marginalizing $\mathbf{x}$ out from $p(\mathbf{y}_1, \mathbf{x}|\mathbf{y}_2)$, we have $p(\mathbf{y}_1|\mathbf{y}_2) = \int p(\mathbf{y}_1, \mathbf{x}|\mathbf{y}_2) d\mathbf{x}$. However, from the figure,



$$\int p(\mathbf{y}_1, \mathbf{x}|\mathbf{y}_2) d\mathbf{x} = p(\mathbf{y}_1|\mathbf{y}_2) = \mathcal{N}(\mathbf{y}_1; \mathbf{y}_2, \Sigma_{\mathbf{Y}_2} + \Sigma_{\mathbf{Y}_1})$$

- On the other hand,

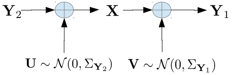
$$\int p(\mathbf{y}_1, \mathbf{x}|\mathbf{y}_2) d\mathbf{x} = \int \mathcal{N}(\mathbf{y}_1; \mathbf{x}, \Sigma_{\mathbf{Y}_1}) \mathcal{N}(\mathbf{y}_2; \mathbf{x}, \Sigma_{\mathbf{Y}_2}) d\mathbf{x}$$

$$= \int K(\mathbf{y}_1, \mathbf{y}_2, \Sigma_{\mathbf{Y}_1}, \Sigma_{\mathbf{Y}_2}) \mathcal{N}(\mathbf{x}; (\Lambda_{\mathbf{Y}_1} + \Lambda_{\mathbf{Y}_2})^{-1}(\Lambda_{\mathbf{Y}_2}\mathbf{y}_2 + \Lambda_{\mathbf{Y}_1} y), (\Lambda_{\mathbf{Y}_2} + \Lambda_{\mathbf{Y}_1})^{-1}) d\mathbf{x}$$

$$= K(\mathbf{y}_1, \mathbf{y}_2, \Sigma_{\mathbf{Y}_1}, \Sigma_{\mathbf{Y}_2}).$$

Product of normal distributions

- Then, marginalizing $\mathbf{x}$ out from $p(\mathbf{y}_1, \mathbf{x}|\mathbf{y}_2)$, we have $p(\mathbf{y}_1|\mathbf{y}_2) = \int p(\mathbf{y}_1, \mathbf{x}|\mathbf{y}_2) d\mathbf{x}$. However, from the figure,



$$\int p(\mathbf{y}_1, \mathbf{x}|\mathbf{y}_2) d\mathbf{x} = p(\mathbf{y}_1|\mathbf{y}_2) = \mathcal{N}(\mathbf{y}_1; \mathbf{y}_2, \Sigma_{\mathbf{Y}_2} + \Sigma_{\mathbf{Y}_1})$$

- On the other hand,

$$\int p(\mathbf{y}_1, \mathbf{x}|\mathbf{y}_2) d\mathbf{x} = \int \mathcal{N}(\mathbf{y}_1; \mathbf{x}, \Sigma_{\mathbf{Y}_1}) \mathcal{N}(\mathbf{y}_2; \mathbf{x}, \Sigma_{\mathbf{Y}_2}) d\mathbf{x}$$

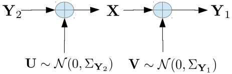
$$= \int K(\mathbf{y}_1, \mathbf{y}_2, \Sigma_{\mathbf{Y}_1}, \Sigma_{\mathbf{Y}_2}) \mathcal{N}(\mathbf{x}; (\Lambda_{\mathbf{Y}_1} + \Lambda_{\mathbf{Y}_2})^{-1}(\Lambda_{\mathbf{Y}_2}\mathbf{y}_2 + \Lambda_{\mathbf{Y}_1} y), (\Lambda_{\mathbf{Y}_2} + \Lambda_{\mathbf{Y}_1})^{-1}) d\mathbf{x}$$

$$= K(\mathbf{y}_1, \mathbf{y}_2, \Sigma_{\mathbf{Y}_1}, \Sigma_{\mathbf{Y}_2}).$$

- Thus we have $K(\mathbf{y}_1, \mathbf{y}_2, \Sigma_{\mathbf{Y}_1}, \Sigma_{\mathbf{Y}_2}) = \mathcal{N}(\mathbf{y}_1; \mathbf{y}_2, \Sigma_{\mathbf{Y}_2} + \Sigma_{\mathbf{Y}_1})$

Product of normal distributions

- Then, marginalizing $\mathbf{x}$ out from $p(\mathbf{y}_1, \mathbf{x}|\mathbf{y}_2)$, we have $p(\mathbf{y}_1|\mathbf{y}_2) = \int p(\mathbf{y}_1, \mathbf{x}|\mathbf{y}_2) d\mathbf{x}$. However, from the figure,



$$\int p(\mathbf{y}_1, \mathbf{x}|\mathbf{y}_2) d\mathbf{x} = p(\mathbf{y}_1|\mathbf{y}_2) = \mathcal{N}(\mathbf{y}_1; \mathbf{y}_2, \Sigma_{\mathbf{Y}_2} + \Sigma_{\mathbf{Y}_1})$$

- On the other hand,

$$\begin{aligned}
&\int p(\mathbf{y}_1, \mathbf{x}|\mathbf{y}_2) d\mathbf{x} = \int \mathcal{N}(\mathbf{y}_1; \mathbf{x}, \Sigma_{\mathbf{Y}_1}) \mathcal{N}(\mathbf{y}_2; \mathbf{x}, \Sigma_{\mathbf{Y}_2}) d\mathbf{x} \\
&= \int K(\mathbf{y}_1, \mathbf{y}_2, \Sigma_{\mathbf{Y}_1}, \Sigma_{\mathbf{Y}_2}) \mathcal{N}(\mathbf{x}; (\Lambda_{\mathbf{Y}_1} + \Lambda_{\mathbf{Y}_2})^{-1} (\Lambda_{\mathbf{Y}_2} \mathbf{y}_2 + \Lambda_{\mathbf{Y}_1} y), (\Lambda_{\mathbf{Y}_2} + \Lambda_{\mathbf{Y}_1})^{-1}) d\mathbf{x} \\
&= K(\mathbf{y}_1, \mathbf{y}_2, \Sigma_{\mathbf{Y}_1}, \Sigma_{\mathbf{Y}_2}).
\end{aligned}$$

- Thus we have $K(\mathbf{y}_1, \mathbf{y}_2, \Sigma_{\mathbf{Y}_1}, \Sigma_{\mathbf{Y}_2}) = \mathcal{N}(\mathbf{y}_1; \mathbf{y}_2, \Sigma_{\mathbf{Y}_2} + \Sigma_{\mathbf{Y}_1})$ and so

$$\begin{aligned}
&\mathcal{N}(\mathbf{y}_1; \mathbf{x}, \Sigma_{\mathbf{Y}_1}) \mathcal{N}(\mathbf{y}_2; \mathbf{x}, \Sigma_{\mathbf{Y}_2}) \\
=& \mathcal{N}(\mathbf{y}_1; \mathbf{y}_2, \Sigma_{\mathbf{Y}_2} + \Sigma_{\mathbf{Y}_1}) \mathcal{N}(\mathbf{x}; (\Lambda_{\mathbf{Y}_1} + \Lambda_{\mathbf{Y}_2})^{-1} (\Lambda_{\mathbf{Y}_2} \mathbf{y}_2 + \Lambda_{\mathbf{Y}_1} y), (\Lambda_{\mathbf{Y}_2} + \Lambda_{\mathbf{Y}_1})^{-1})
\end{aligned}$$

Division of normal distributions

- To compute $\frac{\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_1,\Sigma_1)}{\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_2,\Sigma_2)}$, note that from the product formula earlier

$$
\begin{aligned}
&\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_2,\Sigma_2)\mathcal{N}(\mathbf{x};(\Lambda_1-\Lambda_2)^{-1}(\Lambda_1\boldsymbol{\mu}_1-\Lambda_2\boldsymbol{\mu}_2),(\Lambda_1-\Lambda_2)^{-1})\\
=&\mathcal{N}(\boldsymbol{\mu}_2;(\Lambda_1-\Lambda_2)^{-1}(\Lambda_1\boldsymbol{\mu}_1-\Lambda_2\boldsymbol{\mu}_2),\Lambda_2^{-1}+(\Lambda_1-\Lambda_2)^{-1})\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_1,\Sigma_1)
\end{aligned}
$$

Division of normal distributions

- To compute $\frac{\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_1,\Sigma_1)}{\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_2,\Sigma_2)}$, note that from the product formula earlier

$$\begin{aligned}&\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_2,\Sigma_2)\mathcal{N}(\mathbf{x};(\Lambda_1-\Lambda_2)^{-1}(\Lambda_1\boldsymbol{\mu}_1-\Lambda_2\boldsymbol{\mu}_2),(\Lambda_1-\Lambda_2)^{-1})\\ =&\mathcal{N}(\boldsymbol{\mu}_2;(\Lambda_1-\Lambda_2)^{-1}(\Lambda_1\boldsymbol{\mu}_1-\Lambda_2\boldsymbol{\mu}_2),\Lambda_2^{-1}+(\Lambda_1-\Lambda_2)^{-1})\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_1,\Sigma_1)\end{aligned}$$

- Therefore,

$$\begin{aligned}\frac{\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_1,\Sigma_1)}{\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_2,\Sigma_2)}&=\frac{\mathcal{N}(\mathbf{x};(\Lambda_1-\Lambda_2)^{-1}(\Lambda_1\boldsymbol{\mu}_1-\Lambda_2\boldsymbol{\mu}_2),(\Lambda_1-\Lambda_2)^{-1})}{\mathcal{N}(\boldsymbol{\mu}_2;(\Lambda_1-\Lambda_2)^{-1}(\Lambda_1\boldsymbol{\mu}_1-\Lambda_2\boldsymbol{\mu}_2),\Lambda_2^{-1}+(\Lambda_1-\Lambda_2)^{-1})}\\ &=\frac{\mathcal{N}(\mathbf{x};\boldsymbol{\mu},(\Lambda_1-\Lambda_2)^{-1})}{\mathcal{N}(\boldsymbol{\mu}_2;\boldsymbol{\mu},\Lambda_2^{-1}+(\Lambda_1-\Lambda_2)^{-1})},\end{aligned}$$

where $\boldsymbol{\mu}=(\Lambda_1-\Lambda_2)^{-1}(\Lambda_1\boldsymbol{\mu}_1-\Lambda_2\boldsymbol{\mu}_2)$

Division of normal distributions

- To compute $\frac{\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_1,\Sigma_1)}{\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_2,\Sigma_2)}$, note that from the product formula earlier

$$\begin{aligned}&\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_2,\Sigma_2)\mathcal{N}(\mathbf{x};(\Lambda_1-\Lambda_2)^{-1}(\Lambda_1\boldsymbol{\mu}_1-\Lambda_2\boldsymbol{\mu}_2),(\Lambda_1-\Lambda_2)^{-1})\\=&\mathcal{N}(\boldsymbol{\mu}_2;(\Lambda_1-\Lambda_2)^{-1}(\Lambda_1\boldsymbol{\mu}_1-\Lambda_2\boldsymbol{\mu}_2),\Lambda_2^{-1}+(\Lambda_1-\Lambda_2)^{-1})\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_1,\Sigma_1)\end{aligned}$$

- Therefore,

$$\begin{aligned}\frac{\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_1,\Sigma_1)}{\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_2,\Sigma_2)}&=\frac{\mathcal{N}(\mathbf{x};(\Lambda_1-\Lambda_2)^{-1}(\Lambda_1\boldsymbol{\mu}_1-\Lambda_2\boldsymbol{\mu}_2),(\Lambda_1-\Lambda_2)^{-1})}{\mathcal{N}(\boldsymbol{\mu}_2;(\Lambda_1-\Lambda_2)^{-1}(\Lambda_1\boldsymbol{\mu}_1-\Lambda_2\boldsymbol{\mu}_2),\Lambda_2^{-1}+(\Lambda_1-\Lambda_2)^{-1})}\\&=\frac{\mathcal{N}(\mathbf{x};\boldsymbol{\mu},(\Lambda_1-\Lambda_2)^{-1})}{\mathcal{N}(\boldsymbol{\mu}_2;\boldsymbol{\mu},\Lambda_2^{-1}+(\Lambda_1-\Lambda_2)^{-1})},\end{aligned}$$

  where $\boldsymbol{\mu}=(\Lambda_1-\Lambda_2)^{-1}(\Lambda_1\boldsymbol{\mu}_1-\Lambda_2\boldsymbol{\mu}_2)$

- Note that the final pdf will be Gaussian-like if $\Lambda_1\succeq\Lambda_2$. Otherwise, one can still write out the pdf using the precision matrix. But the covariance matrix will not be defined (Try plot some pdfs out yourselves)

Product of normal distributions

Let us try to interpret the product as the overall likelihood after making two observations. Consider the simpler case when $\mathbf{X}$, $\mathbf{Y}_1$ and $\mathbf{Y}_2$ are all scaler

Product of normal distributions

Let us try to interpret the product as the overall likelihood after making two observations. Consider the simpler case when $\mathbf{X}$, $\mathbf{Y}_1$ and $\mathbf{Y}_2$ are all scaler

- The mean considering both observations, $(\Lambda_{\mathbf{Y}_1} + \Lambda_{\mathbf{Y}_2})^{-1}(\Lambda_{\mathbf{Y}_2}\mathbf{y}_2 + \Lambda_{\mathbf{Y}_1}y)$, is essential a weighted average of observations $\mathbf{y}_2$ and $\mathbf{y}_1$
    - The weight is higher when the precision $\Lambda_{\mathbf{Y}_2}$ or $\Lambda_{\mathbf{Y}_1}$ is larger

Product of normal distributions

Let us try to interpret the product as the overall likelihood after making two observations. Consider the simpler case when $\mathbf{X}$, $\mathbf{Y}_1$ and $\mathbf{Y}_2$ are all scaler

- The mean considering both observations, $(\Lambda_{\mathbf{Y}_1} + \Lambda_{\mathbf{Y}_2})^{-1}(\Lambda_{\mathbf{Y}_2}\mathbf{y}_2 + \Lambda_{\mathbf{Y}_1}y)$, is essential a weighted average of observations $\mathbf{y}_2$ and $\mathbf{y}_1$
  - The weight is higher when the precision $\Lambda_{\mathbf{Y}_2}$ or $\Lambda_{\mathbf{Y}_1}$ is larger
- The overall variance $(\Lambda_{\mathbf{Y}_2} + \Lambda_{\mathbf{Y}_1})^{-1}$ is always smaller than the individual variance $\Sigma_{\mathbf{Y}_2}$ and $\Sigma_{\mathbf{Y}_1}$
  - We are more certain with $\mathbf{x}$ after considering both $\mathbf{y}_1$ and $\mathbf{y}_2$

Product of normal distributions

Let us try to interpret the product as the overall likelihood after making two observations. Consider the simpler case when $\mathbf{X}$, $\mathbf{Y}_1$ and $\mathbf{Y}_2$ are all scaler

- The mean considering both observations, $(\Lambda_{\mathbf{Y}_1} + \Lambda_{\mathbf{Y}_2})^{-1}(\Lambda_{\mathbf{Y}_2}\mathbf{y}_2 + \Lambda_{\mathbf{Y}_1}y)$, is essential a weighted average of observations $\mathbf{y}_2$ and $\mathbf{y}_1$
    - The weight is higher when the precision $\Lambda_{\mathbf{Y}_2}$ or $\Lambda_{\mathbf{Y}_1}$ is larger
- The overall variance $(\Lambda_{\mathbf{Y}_2} + \Lambda_{\mathbf{Y}_1})^{-1}$ is always smaller than the individual variance $\Sigma_{\mathbf{Y}_2}$ and $\Sigma_{\mathbf{Y}_1}$
    - We are more certain with $\mathbf{x}$ after considering both $\mathbf{y}_1$ and $\mathbf{y}_2$
- The scaling factor, $\mathcal{N}(\mathbf{y}_1; \mathbf{y}_2, \Sigma_{\mathbf{Y}_2} + \Sigma_{\mathbf{Y}_1})$, can be interpreted as how much one can believe on the overall likelihood.
    - The value is reasonable since when the two observations are far away with respect to the overall variance $\Sigma_{\mathbf{Y}_2} + \Sigma_{\mathbf{Y}_1}$, the likelihood will become less reliable
    - The scaling factor is especially useful when we deal with mixture of Gaussian to be discussed next

Mixture of Gaussians

Mixture of Gaussians

Consider an electrical system that outputs signal of different statistics when it is on and off

- When the system is on, the output signal S behaves like $\mathcal{N}(5, 1)$. When the system is off is off, S behaves like $\mathcal{N}(0, 1)$

Mixture of Gaussians

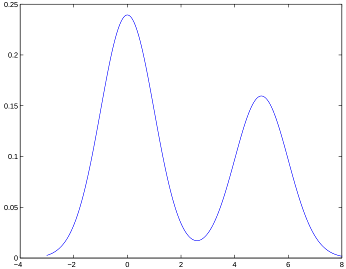
Consider an electrical system that outputs signal of different statistics when it is on and off

- When the system is on, the output signal S behaves like $\mathcal{N}(5,1)$. When the system is off is off, S behaves like $\mathcal{N}(0,1)$
- If someone measuring the signal does not know the status of the system but only knows that the system is on 40% of the time, then to the observer, the signal S behaves like a mixture of Gaussians

Mixture of Gaussians

Consider an electrical system that outputs signal of different statistics when it is on and off

- When the system is on, the output signal S behaves like $\mathcal{N}(5,1)$. When the system is off is off, S behaves like $\mathcal{N}(0,1)$
- If someone measuring the signal does not know the status of the system but only knows that the system is on 40% of the time, then to the observer, the signal S behaves like a mixture of Gaussians
- The pdf of S will be $0.4\mathcal{N}(s;5,1)+0.6\mathcal{N}(s;0,1)$ as shown below



Mixture of Gaussians

- A main limitation of normal distribution is that it is unimodal

Mixture of Gaussians

- A main limitation of normal distribution is that it is unimodal
- Mixture of Gaussian distribution allows multimodal and can virtually model any pdfs. But there is a computational cost for this gain

Mixture of Gaussians

- A main limitation of normal distribution is that it is unimodal
- Mixture of Gaussian distribution allows multimodal and can virtually model any pdfs. But there is a computational cost for this gain
- Let us illustrate this with the following example:
  - Consider two mixtures of Gaussian likelihood of x given two observations y_1 and y_2 as follows:

$$p(y_1|x) = 0.6\mathcal{N}(x; 0, 1) + 0.4\mathcal{N}(x; 5, 1);$$
$$p(y_2|x) = 0.5\mathcal{N}(x; -2, 1) + 0.5\mathcal{N}(x; 4, 1).$$

What is the overall likelihood, $p(y_1, y_2|x)$?

Mixture of Gaussians

- A main limitation of normal distribution is that it is unimodal
- Mixture of Gaussian distribution allows multimodal and can virtually model any pdfs. But there is a computational cost for this gain
- Let us illustrate this with the following example:
  - Consider two mixtures of Gaussian likelihood of x given two observations y_1 and y_2 as follows:

$$p(y_1|x) = 0.6\mathcal{N}(x;0,1) + 0.4\mathcal{N}(x;5,1);$$
$$p(y_2|x) = 0.5\mathcal{N}(x;-2,1) + 0.5\mathcal{N}(x;4,1).$$

  What is the overall likelihood, $p(y_1, y_2|x)$?

- As usual, it is reasonable to assume the observations to be conditionally independent given x. Then,

$$\begin{aligned} p(y_1,y_2|x) &= p(y_1|x)p(y_2|x) \\ &= (0.6\mathcal{N}(x;0,1) + 0.4\mathcal{N}(x;5,1))(0.5\mathcal{N}(x;-2,1) + 0.5\mathcal{N}(x;4,1)) \\ &= 0.3\mathcal{N}(x;0,1)\mathcal{N}(x;-2,1) + 0.2\mathcal{N}(x;5,1)\mathcal{N}(x;-2,1) \\ &\quad + 0.3\mathcal{N}(x;0,1)\mathcal{N}(x;4,1) + 0.2\mathcal{N}(x;5,1)\mathcal{N}(x;4,1) \end{aligned}$$

Explosion of Gaussians

- The last step involves computing products of Gaussians but we have learned it in previous sections. Using the previous result,

$$\begin{aligned} p(y_1, y_2|x) = &\, 0.3\mathcal{N}(-2; 0, 2)\mathcal{N}(x; -1, 0.5) + 0.2\mathcal{N}(-2; 5, 2)\mathcal{N}(x; 1.5, 0.5) \\ &+ 0.3\mathcal{N}(4; 0, 2)\mathcal{N}(x; 2, 0.5) + 0.2\mathcal{N}(4; 5, 2)\mathcal{N}(x; 4.5, 0.5). \end{aligned}$$

Explosion of Gaussians

- The last step involves computing products of Gaussians but we have learned it in previous sections. Using the previous result,

$$\begin{aligned} p(y_1, y_2|x) = &\, 0.3\mathcal{N}(-2; 0, 2)\mathcal{N}(x; -1, 0.5) + 0.2\mathcal{N}(-2; 5, 2)\mathcal{N}(x; 1.5, 0.5) \\ &+ 0.3\mathcal{N}(4; 0, 2)\mathcal{N}(x; 2, 0.5) + 0.2\mathcal{N}(4; 5, 2)\mathcal{N}(x; 4.5, 0.5). \end{aligned}$$

  So we have the overall likelihood is a mixture of four Gaussians

Explosion of Gaussians

- The last step involves computing products of Gaussians but we have learned it in previous sections. Using the previous result,

$$
\begin{aligned}
p(y_1, y_2|x) = &\ 0.3\mathcal{N}(-2; 0, 2)\mathcal{N}(x; -1, 0.5) + 0.2\mathcal{N}(-2; 5, 2)\mathcal{N}(x; 1.5, 0.5) \\
&+ 0.3\mathcal{N}(4; 0, 2)\mathcal{N}(x; 2, 0.5) + 0.2\mathcal{N}(4; 5, 2)\mathcal{N}(x; 4.5, 0.5).
\end{aligned}
$$

  So we have the overall likelihood is a mixture of four Gaussians
- Let’s repeat our discussion but with n observations instead. The overall likelihood will be a mixture of 2^n Gaussians!
  - Therefore, the computation will quickly become intractable as the number of observations increases

Explosion of Gaussians

- The last step involves computing products of Gaussians but we have learned it in previous sections. Using the previous result,

$$\begin{aligned} p(y_1, y_2|x) = & \; 0.3\mathcal{N}(-2; 0, 2)\mathcal{N}(x; -1, 0.5) + 0.2\mathcal{N}(-2; 5, 2)\mathcal{N}(x; 1.5, 0.5) \\ & + 0.3\mathcal{N}(4; 0, 2)\mathcal{N}(x; 2, 0.5) + 0.2\mathcal{N}(4; 5, 2)\mathcal{N}(x; 4.5, 0.5). \end{aligned}$$

  So we have the overall likelihood is a mixture of four Gaussians
- Let's repeat our discussion but with n observations instead. The overall likelihood will be a mixture of 2^n Gaussians!
  - Therefore, the computation will quickly become intractable as the number of observations increases
  - Fortunately, in reality, some of the Gaussians in the mixture tend to have a very small weight

Reduce number of components in Gaussian mixtures

- For instance, in our previous numerical example, if we continue our numerical computation for the two observation example, we have

$$\begin{aligned} p(y_1, y_2|x) = {} & 0.4163\mathcal{N}(x; -1, 0.5) + 3.5234 \times 10^{-6}\mathcal{N}(x; 1.5, 0.5) \\ & + 0.0202\mathcal{N}(x; 2, 0.5) + 0.5734\mathcal{N}(x; 4.5, 0.5). \end{aligned}$$

Reduce number of components in Gaussian mixtures

- For instance, in our previous numerical example, if we continue our numerical computation for the two observation example, we have

$$p(y_1,y_2|x)=0.4163\mathcal{N}(x;-1,0.5)+3.5234\times 10^{-6}\mathcal{N}(x;1.5,0.5) \\ +0.0202\mathcal{N}(x;2,0.5)+0.5734\mathcal{N}(x;4.5,0.5).$$

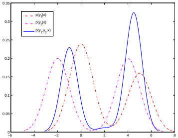
- We can see that the weight for the component at mean 1.5 is very small. And the component at mean 2 has a rather small weight also.

Reduce number of components in Gaussian mixtures

- For instance, in our previous numerical example, if we continue our numerical computation for the two observation example, we have

$$
\begin{aligned}
p(y_1, y_2|x) = {} & 0.4163\mathcal{N}(x; -1, 0.5) + 3.5234 \times 10^{-6}\mathcal{N}(x; 1.5, 0.5) \\
& + 0.0202\mathcal{N}(x; 2, 0.5) + 0.5734\mathcal{N}(x; 4.5, 0.5).
\end{aligned}
$$

- We can see that the weight for the component at mean 1.5 is very small. And the component at mean 2 has a rather small weight also.
- Even with the four Gaussian components, the overall likelihood is essentially just a bimodal distribution as shown in the figure below



Reduce number of components in Gaussian mixtures

- Therefore, we may approximate $p(y_1, y_2|x)$ with only two of its original component as $0.4163/(0.4163+0.5734)\mathcal{N}(x;-1,0.5)+0.5734/(0.4163+0.5734)\mathcal{N}(x;4.5,0.5)=0.4206\mathcal{N}(x;-1,0.5)+0.5794\mathcal{N}(x;4.5,0.5)$

Reduce number of components in Gaussian mixtures

- Therefore, we may approximate $p(y_1, y_2|x)$ with only two of its original component as $0.4163/(0.4163 + 0.5734)\mathcal{N}(x; -1, 0.5) + 0.5734/(0.4163 + 0.5734)\mathcal{N}(x; 4.5, 0.5) = 0.4206\mathcal{N}(x; -1, 0.5) + 0.5794\mathcal{N}(x; 4.5, 0.5)$
- However, it is not always a good approximation strategy just to dump away the small components in a Gaussian mixture

Another example

Consider

$$p(x) = 0.1\mathcal{N}(x;-0.2,1) + 0.1\mathcal{N}(x;-0.1,1) + 0.1\mathcal{N}(x;0,1) + 0.1\mathcal{N}(x;0.1,1) \\ + 0.1\mathcal{N}(x;0.2,1) + 0.5\mathcal{N}(x;5,1).$$

Another example

Consider

$$p(x) = 0.1\mathcal{N}(x; -0.2, 1) + 0.1\mathcal{N}(x; -0.1, 1) + 0.1\mathcal{N}(x; 0, 1) + 0.1\mathcal{N}(x; 0.1, 1) \\ + 0.1\mathcal{N}(x; 0.2, 1) + 0.5\mathcal{N}(x; 5, 1).$$

- Let say we want to reduce $p(x)$ to only a mixture of two Gaussians. It is tempting to just dumping four smallest one and renormalized the weight. For example, if we choose to remove the first four components, we have

$$\hat{p}(x) = 1/6\mathcal{N}(x; 0.2, 1) + 5/6\mathcal{N}(x; 5, 1)$$

Another example

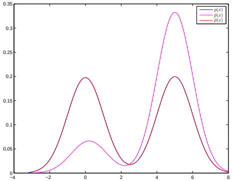
Consider

$$p(x) = 0.1\mathcal{N}(x; -0.2, 1) + 0.1\mathcal{N}(x; -0.1, 1) + 0.1\mathcal{N}(x; 0, 1) + 0.1\mathcal{N}(x; 0.1, 1) \\ + 0.1\mathcal{N}(x; 0.2, 1) + 0.5\mathcal{N}(x; 5, 1).$$

- Let say we want to reduce $p(x)$ to only a mixture of two Gaussians. It is tempting to just dumping four smallest one and renormalized the weight. For example, if we choose to remove the first four components, we have

$$\hat{p}(x) = 1/6\mathcal{N}(x; 0.2, 1) + 5/6\mathcal{N}(x; 5, 1)$$

- The approximation $\hat{p}(x)$ is significantly different from $p(x)$ as shown below



Merging components

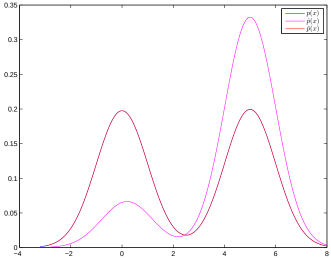
- The problem is that while the first five components are all relatively small compared to the last one, they are all quite similar and their combined contribution is comparable to the latter

Merging components

- The problem is that while the first five components are all relatively small compared to the last one, they are all quite similar and their combined contribution is comparable to the latter
- Actually the first five components are so similar that their combined contribution can be accurately modeled as one Gaussian

Merging components

- The problem is that while the first five components are all relatively small compared to the last one, they are all quite similar and their combined contribution is comparable to the latter
- Actually the first five components are so similar that their combined contribution can be accurately modeled as one Gaussian
- So rather than discarding the components, one can get a much more accurate approximation by merging them. The approximation is illustrated as $\tilde{p}(x)$ in the figure below



Merging components

To successfully obtain such approximation $\tilde{p}(x)$, we have to answer two questions:

- which components to merge?
- how to merge them?

Which Components to Merge?

It is reasonable to pick similar components to merge. The question is how do will gauge the similarity between two components.

Which Components to Merge?

It is reasonable to pick similar components to merge. The question is how do will gauge the similarity between two components.

- Consider two pdfs $p(\mathbf{x})$ and $q(\mathbf{x})$, note that we can define an inner product of $p(\mathbf{x})$ and $q(\mathbf{x})$ by

$$\langle p(\mathbf{x}), q(\mathbf{x})\rangle = \int p(\mathbf{x})q(\mathbf{x})d\mathbf{x}$$

Which Components to Merge?

It is reasonable to pick similar components to merge. The question is how do will gauge the similarity between two components.

- Consider two pdfs $p(\mathbf{x})$ and $q(\mathbf{x})$, note that we can define an inner product of $p(\mathbf{x})$ and $q(\mathbf{x})$ by

$$\langle p(\mathbf{x}), q(\mathbf{x})\rangle = \int p(\mathbf{x})q(\mathbf{x})d\mathbf{x}$$

- Note that the inner product is well defined and $\langle p(\mathbf{x}), p(\mathbf{x})\rangle \geq 0$

Which Components to Merge?

It is reasonable to pick similar components to merge. The question is how do will gauge the similarity between two components.

- Consider two pdfs $p(\mathbf{x})$ and $q(\mathbf{x})$, note that we can define an inner product of $p(\mathbf{x})$ and $q(\mathbf{x})$ by

$$\langle p(\mathbf{x}), q(\mathbf{x})\rangle = \int p(\mathbf{x})q(\mathbf{x})d\mathbf{x}$$

- Note that the inner product is well defined and $\langle p(\mathbf{x}), p(\mathbf{x})\rangle \geq 0$
- By Cauchy-Schwartz inequality,

$$\frac{\langle p(\mathbf{x}), q(\mathbf{x})\rangle}{\sqrt{\langle p(\mathbf{x}), p(\mathbf{x})\rangle\langle q(\mathbf{x}), q(\mathbf{x})\rangle}} = \frac{\int p(\mathbf{x})q(\mathbf{x})d\mathbf{x}}{\sqrt{\int p(\mathbf{x})^2 d\mathbf{x} \int q(\mathbf{x})^2 d\mathbf{x}}} \leq 1$$

Which Components to Merge?

It is reasonable to pick similar components to merge. The question is how do will gauge the similarity between two components.

- Consider two pdfs $p(\mathbf{x})$ and $q(\mathbf{x})$, note that we can define an inner product of $p(\mathbf{x})$ and $q(\mathbf{x})$ by

$$\langle p(\mathbf{x}), q(\mathbf{x})\rangle = \int p(\mathbf{x})q(\mathbf{x})d\mathbf{x}$$

- Note that the inner product is well defined and $\langle p(\mathbf{x}), p(\mathbf{x})\rangle \geq 0$
- By Cauchy-Schwartz inequality,

$$\frac{\langle p(\mathbf{x}), q(\mathbf{x})\rangle}{\sqrt{\langle p(\mathbf{x}), p(\mathbf{x})\rangle\langle q(\mathbf{x}), q(\mathbf{x})\rangle}} = \frac{\int p(\mathbf{x})q(\mathbf{x})d\mathbf{x}}{\sqrt{\int p(\mathbf{x})^2 d\mathbf{x} \int q(\mathbf{x})^2 d\mathbf{x}}} \leq 1$$

- The inner product maximizes $(=1)$ when $p(\mathbf{x}) = q(\mathbf{x})$. This suggests a very reasonable similarity measure between two pdfs

Similarity measure

- Let’s define

$$Sim(p(\mathbf{x}), q(\mathbf{x})) \triangleq \frac{\int p(\mathbf{x})q(\mathbf{x})d\mathbf{x}}{\sqrt{\int p(\mathbf{x})^2 d\mathbf{x} \int q(\mathbf{x})^2 d\mathbf{x}}}$$

Similarity measure

- Let's define

$$Sim(p(\mathbf{x}), q(\mathbf{x})) \triangleq \frac{\int p(\mathbf{x})q(\mathbf{x})d\mathbf{x}}{\sqrt{\int p(\mathbf{x})^2 d\mathbf{x} \int q(\mathbf{x})^2 d\mathbf{x}}}$$

- In particular, if $p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_p, \Sigma_p)$ and $q(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_q, \Sigma_q)$, we have (please verify)

$$Sim(\mathcal{N}(\boldsymbol{\mu}_p, \Sigma_p), \mathcal{N}(\boldsymbol{\mu}_q, \Sigma_q)) = \frac{\mathcal{N}(\boldsymbol{\mu}_p; \boldsymbol{\mu}_q, \Sigma_p + \Sigma_q)}{\sqrt{\mathcal{N}(0; 0, 2\Sigma_p)\mathcal{N}(0; 0, 2\Sigma_q)}},$$

  which can be computed very easily and is equal to one only when means and covariances are the same

How to Merge Components?

Say we have n components $\mathcal{N}(\boldsymbol{\mu}_1, \Sigma_1)$, $\mathcal{N}(\boldsymbol{\mu}_2, \Sigma_2)$, $\cdots$, $\mathcal{N}(\boldsymbol{\mu}_n, \Sigma_n)$ with weights $w_1, w_2, \cdots, w_n$. What should the combined component be like?

How to Merge Components?

Say we have n components $\mathcal{N}(\boldsymbol{\mu}_1, \Sigma_1)$, $\mathcal{N}(\boldsymbol{\mu}_2, \Sigma_2)$, $\cdots$, $\mathcal{N}(\boldsymbol{\mu}_n, \Sigma_n)$ with weights $w_1, w_2, \cdots, w_n$. What should the combined component be like?

- Combined component weight should equal to total weight $\sum_{i=1}^n w_i$

How to Merge Components?

Say we have n components $\mathcal{N}(\boldsymbol{\mu}_1, \Sigma_1)$, $\mathcal{N}(\boldsymbol{\mu}_2, \Sigma_2)$, $\cdots$, $\mathcal{N}(\boldsymbol{\mu}_n, \Sigma_n)$ with weights $w_1, w_2, \cdots, w_n$. What should the combined component be like?

- Combined component weight should equal to total weight $\sum_{i=1}^n w_i$
- Combined mean will simply be $\sum_{i=1}^n \hat{w}_i \boldsymbol{\mu}_i$, where $\hat{w}_i = \frac{w_i}{\sum_{i=1}^n w_i}$

How to Merge Components?

Say we have n components $\mathcal{N}(\boldsymbol{\mu}_1, \Sigma_1)$, $\mathcal{N}(\boldsymbol{\mu}_2, \Sigma_2)$, $\cdots$, $\mathcal{N}(\boldsymbol{\mu}_n, \Sigma_n)$ with weights $w_1, w_2, \cdots, w_n$. What should the combined component be like?

- Combined component weight should equal to total weight $\sum_{i=1}^n w_i$
- Combined mean will simply be $\sum_{i=1}^n \hat{w}_i \boldsymbol{\mu}_i$, where $\hat{w}_i = \frac{w_i}{\sum_{i=1}^n w_i}$
- For combined covariance, it may be tempting to approximate it as $\sum_{i=1}^n \hat{w}_i \Sigma_i$.

How to Merge Components?

Say we have n components $\mathcal{N}(\boldsymbol{\mu}_1, \Sigma_1)$, $\mathcal{N}(\boldsymbol{\mu}_2, \Sigma_2)$, $\cdots$, $\mathcal{N}(\boldsymbol{\mu}_n, \Sigma_n)$ with weights $w_1, w_2, \cdots, w_n$. What should the combined component be like?

- Combined component weight should equal to total weight $\sum_{i=1}^n w_i$
- Combined mean will simply be $\sum_{i=1}^n \hat{w}_i \boldsymbol{\mu}_i$, where $\hat{w}_i = \frac{w_i}{\sum_{i=1}^n w_i}$
- For combined covariance, it may be tempting to approximate it as $\sum_{i=1}^n \hat{w}_i \Sigma_i$.
    - However, it is an underestimate

How to Merge Components?

Say we have n components $\mathcal{N}(\boldsymbol{\mu}_1, \Sigma_1)$, $\mathcal{N}(\boldsymbol{\mu}_2, \Sigma_2)$, $\cdots$, $\mathcal{N}(\boldsymbol{\mu}_n, \Sigma_n)$ with weights $w_1, w_2, \cdots, w_n$. What should the combined component be like?

- Combined component weight should equal to total weight $\sum_{i=1}^n w_i$
- Combined mean will simply be $\sum_{i=1}^n \hat{w}_i \boldsymbol{\mu}_i$, where $\hat{w}_i = \frac{w_i}{\sum_{i=1}^n w_i}$
- For combined covariance, it may be tempting to approximate it as $\sum_{i=1}^n \hat{w}_i \Sigma_i$.
  - However, it is an underestimate
  - Because the weighted sum only counted the contribution of variation among each component, it did not take into account the variation due to different means across components.

How to Merge Components?

Say we have n components $\mathcal{N}(\boldsymbol{\mu}_1,\Sigma_1)$, $\mathcal{N}(\boldsymbol{\mu}_2,\Sigma_2)$, $\cdots$, $\mathcal{N}(\boldsymbol{\mu}_n,\Sigma_n)$ with weights $w_1, w_2, \cdots, w_n$. What should the combined component be like?

- Combined component weight should equal to total weight $\sum_{i=1}^n w_i$
- Combined mean will simply be $\sum_{i=1}^n \hat{w}_i \boldsymbol{\mu}_i$, where $\hat{w}_i = \frac{w_i}{\sum_{i=1}^n w_i}$
- For combined covariance, it may be tempting to approximate it as $\sum_{i=1}^n \hat{w}_i \Sigma_i$.
    - However, it is an underestimate
    - Because the weighted sum only counted the contribution of variation among each component, it did not take into account the variation due to different means across components.
    - Instead, let's denote $\mathbf{X}$ as the variable sampled from the mixture. That is, $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}_i,\Sigma_i)$ with probability $\hat{w}_i$. Then, we have (please verify)

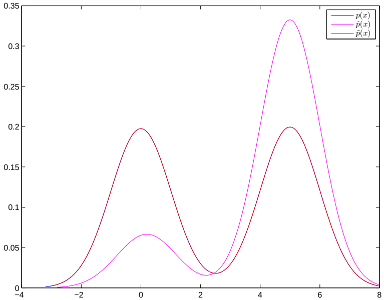
$$\begin{aligned}\Sigma &= E[\mathbf{X}\mathbf{X}^T] - E[\mathbf{X}]E[\mathbf{X}]^T \\ &= \sum_{i=1}^n \hat{w}_i(\Sigma_i + \boldsymbol{\mu}_i\boldsymbol{\mu}_i^T) - \sum_{i=1}^n\sum_{j=1}^n \hat{w}_i\hat{w}_j\boldsymbol{\mu}_i\boldsymbol{\mu}_j^T.\end{aligned}$$

Now, go back to our previous numerical example

- Recall that $p(x) = 0.1\mathcal{N}(x; -0.2, 1) + 0.1\mathcal{N}(x; -0.1, 1) + 0.1\mathcal{N}(x; 0, 1) + 0.1\mathcal{N}(x; 0.1, 1) + 0.1\mathcal{N}(x; 0.2, 1) + 0.5\mathcal{N}(x; 5, 1)$

Now, go back to our previous numerical example

- Recall that $p(x) = 0.1\mathcal{N}(x; -0.2, 1) + 0.1\mathcal{N}(x; -0.1, 1) + 0.1\mathcal{N}(x; 0, 1) + 0.1\mathcal{N}(x; 0.1, 1) + 0.1\mathcal{N}(x; 0.2, 1) + 0.5\mathcal{N}(x; 5, 1)$
- If we merge the five smallest components (one can easily check that they are also more similar to each other than to the last component), we have $\tilde{p}(x) = 0.5\mathcal{N}(x; 0, 1.02) + 0.5\mathcal{N}(x; 5, 1)$ as shown again below. The approximate pdf is virtually indistinguishable from the original



Review multivariate normal

- Marginalization of a normal distribution is still a normal distribution
- Conditioning of normal distribution:
  $\mathbf{X}|\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{X}} + \Sigma_{\mathbf{XY}}\Sigma_{\mathbf{YY}}^{-1}(\mathbf{y} - \boldsymbol{\mu}_{\mathbf{Y}}), \Sigma_{\mathbf{XX}} - \Sigma_{\mathbf{XY}}\Sigma_{\mathbf{YY}}^{-1}\Sigma_{\mathbf{YX}})$
- Product of normal distribution:
  $\mathcal{N}(\mathbf{y}_1; \mathbf{x}, \Sigma_{\mathbf{Y}_1})\mathcal{N}(\mathbf{y}_2; \mathbf{x}, \Sigma_{\mathbf{Y}_2}) =$
  $\mathcal{N}(\mathbf{y}_1; \mathbf{y}_2, \Sigma_{\mathbf{Y}_2} + \Sigma_{\mathbf{Y}_1})\mathcal{N}(\mathbf{x}; (\Lambda_{\mathbf{Y}_1} + \Lambda_{\mathbf{Y}_2})^{-1}(\Lambda_{\mathbf{Y}_2}\mathbf{y}_2 + \Lambda_{\mathbf{Y}_1}y), (\Lambda_{\mathbf{Y}_2} + \Lambda_{\mathbf{Y}_1})^{-1})$
- Division of normal distribution:

$$\frac{\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_1, \Sigma_1)}{\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_2, \Sigma_2)} = \frac{\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, (\Lambda_1 - \Lambda_2)^{-1})}{\mathcal{N}(\boldsymbol{\mu}_2; \boldsymbol{\mu}, \Lambda_2^{-1} + (\Lambda_1 - \Lambda_2)^{-1})},$$

  where $\boldsymbol{\mu} = (\Lambda_1 - \Lambda_2)^{-1}(\Lambda_1\boldsymbol{\mu}_1 - \Lambda_2\boldsymbol{\mu}_2)$
- Similarity measure

$$Sim(\mathcal{N}(\boldsymbol{\mu}_p, \Sigma_p), \mathcal{N}(\boldsymbol{\mu}_q, \Sigma_q)) = \frac{\mathcal{N}(\boldsymbol{\mu}_p; \boldsymbol{\mu}_q, \Sigma_p + \Sigma_q)}{\sqrt{\mathcal{N}(0; 0, 2\Sigma_p)\mathcal{N}(0; 0, 2\Sigma_q)}},$$

Normal distribution revisit

For a univariate normal random variable, the pdf is given by

$$\begin{aligned} Norm(x|\mu,\sigma^2) =& \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \\ =& \sqrt{\frac{\lambda}{2\pi}} exp\left(-\frac{\lambda(x-\mu)^2}{2}\right) \end{aligned}$$

with

$$E[X|\mu,\sigma^2] = \mu,$$

$$E[(X-\mu)^2|\mu,\sigma^2] = \sigma^2,$$

Recall that $\lambda = \frac{1}{\sigma^2}$ is the precision parameter that simplifies computations in many cases

Conjugate prior of normal distribution for fixed σ_2

Consider σ^2 fixed and μ as the model parameter, then the posterior probability is given by

$$p(\mu|x;\sigma^2) \propto p(\mu,x;\sigma^2)$$

Conjugate prior of normal distribution for fixed σ_2

Consider σ^2 fixed and μ as the model parameter, then the posterior probability is given by

$$\begin{aligned} p(\mu|x;\sigma^2) &\propto p(\mu,x;\sigma^2) \\ =&p(\mu)Norm(x|\mu;\sigma^2) \end{aligned}$$

Conjugate prior of normal distribution for fixed σ_2

Consider σ^2 fixed and μ as the model parameter, then the posterior probability is given by

$$\begin{aligned} p(\mu|x;\sigma^2) &\propto p(\mu,x;\sigma^2) \\ =&p(\mu)Norm(x|\mu;\sigma^2) \\ \propto&p(\mu)exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \end{aligned}$$

Conjugate prior of normal distribution for fixed σ_2

Consider σ^2 fixed and μ as the model parameter, then the posterior probability is given by

$$\begin{aligned} p(\mu|x;\sigma^2) &\propto p(\mu,x;\sigma^2) \\ =&p(\mu)Norm(x|\mu;\sigma^2) \\ \propto&p(\mu)exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \end{aligned}$$

It is apparent that the posterior will keep the same form if $p(\mu)$ is also normal. Therefore, normal distribution is the conjugate prior of itself for fixed variance

Posterior distribution of normal variable for fixed σ^2

Given prior $p(\mu) = Norm(\mu|\mu_0, \sigma_0^2)$ and likelihood $Norm(x|\mu; \sigma^2)$. Let's find the posterior probability,

$$
\begin{aligned}
&p(\mu|x; \sigma^2, \mu_0, \sigma_0^2) \\
=&Const \cdot Norm(\mu|\mu_0, \sigma_0^2) Norm(x|\mu; \sigma^2)
\end{aligned}
$$

Posterior distribution of normal variable for fixed σ^2

Given prior $p(\mu) = Norm(\mu|\mu_0, \sigma_0^2)$ and likelihood $Norm(x|\mu; \sigma^2)$. Let's find the posterior probability,

$$
\begin{aligned}
& p(\mu|x; \sigma^2, \mu_0, \sigma_0^2) \\
=& Const \cdot Norm(\mu|\mu_0, \sigma_0^2) Norm(x|\mu; \sigma^2) \\
=& Const2 \cdot exp\left(-\frac{(x-\mu)^2}{2\sigma^2} - \frac{(\mu-\mu_0)^2}{2\sigma_0^2}\right) \\
=& Norm\left(\mu; \tilde{\mu}, \tilde{\sigma}^2\right),
\end{aligned}
$$

Posterior distribution of normal variable for fixed σ^2

Given prior $p(\mu) = Norm(\mu|\mu_0, \sigma_0^2)$ and likelihood $Norm(x|\mu; \sigma^2)$. Let's find the posterior probability,

$$\begin{aligned} & p(\mu|x; \sigma^2, \mu_0, \sigma_0^2) \\ =& Const \cdot Norm(\mu|\mu_0, \sigma_0^2) Norm(x|\mu; \sigma^2) \\ =& Const2 \cdot exp \left(-\frac{(x-\mu)^2}{2\sigma^2} - \frac{(\mu-\mu_0)^2}{2\sigma_0^2} \right) \\ =& Norm\left(\mu; \tilde{\mu}, \tilde{\sigma}^2\right), \end{aligned}$$

where $\tilde{\mu} = \frac{\sigma_0^2 x + \mu_0 \sigma^2}{\sigma_0^2 + \sigma^2}$ and $\tilde{\sigma}^2 = \frac{\sigma_0^2 \sigma^2}{\sigma_0^2 + \sigma^2}$. Alternatively, $\tilde{\lambda} = \lambda_0 + \lambda$ and $\tilde{\mu} = \frac{\lambda}{\tilde{\lambda}} x + \frac{\lambda_0}{\tilde{\lambda}} \mu_0$. Note that we have already came across the more general expression when we studied product of multivariate normal distribution

Conjugate prior of normal distribution for fixed μ

Consider μ fixed and λ as the model parameter

$$p(x|\lambda;\mu) \propto p(x,\lambda;\mu) = p(\lambda)Norm(x|\lambda;\mu)$$

Conjugate prior of normal distribution for fixed μ

Consider μ fixed and λ as the model parameter

$$\begin{aligned} p(x|\lambda;\mu) &\propto p(x,\lambda;\mu) = p(\lambda) Norm(x|\lambda;\mu) \\ &\propto p(\lambda)\sqrt{\lambda}\exp\left(-\frac{\lambda(x-\mu)^2}{2}\right) \end{aligned}$$

Conjugate prior of normal distribution for fixed μ

Consider μ fixed and λ as the model parameter

$$
\begin{aligned}
p(x|\lambda;\mu) \propto & p(x,\lambda;\mu) = p(\lambda) Norm(x|\lambda;\mu) \\
\propto & p(\lambda)\sqrt{\lambda}\exp\left(-\frac{\lambda(x-\mu)^2}{2}\right)
\end{aligned}
$$

More generally, when we have N observations from the same source,

$$
\begin{aligned}
p(x_1,\cdots,x_N,\lambda;\mu) = & p(\lambda)\prod_{i=1}^{N} Norm(x_i|\lambda;\mu) \\
\propto & p(\lambda)\lambda^{\frac{N}{2}}\exp\left(-\lambda\sum_{i=1}^{N}\frac{(x_i-\mu)^2}{2}\right)
\end{aligned}
$$

Conjugate prior of normal distribution for fixed μ

Consider μ fixed and λ as the model parameter

$$\begin{aligned} p(x|\lambda;\mu) \propto & p(x,\lambda;\mu) = p(\lambda) Norm(x|\lambda;\mu) \\ \propto & p(\lambda)\sqrt{\lambda}\exp\left(-\frac{\lambda(x-\mu)^2}{2}\right) \end{aligned}$$

More generally, when we have N observations from the same source,

$$\begin{aligned} p(x_1,\cdots,x_N,\lambda;\mu) = & p(\lambda)\prod_{i=1}^{N} Norm(x_i|\lambda;\mu) \\ \propto & p(\lambda)\lambda^{\frac{N}{2}}\exp\left(-\lambda\sum_{i=1}^{N}\frac{(x_i-\mu)^2}{2}\right) \end{aligned}$$

From inspection, the conjugate prior should have a form $\lambda^a\exp(-b\lambda)$

Gamma distribution

The distribution with the desired form described in previous slide turns out to be the Gamma distribution. Its pdf, mean, and variance (please verify the mean and variance) are given by

$$Gamma(\lambda|a,b) = \frac{1}{\Gamma(a)} b^a \lambda^{a-1} exp(-b\lambda)$$

$$E[\lambda] = \frac{a}{b}$$

$$Var[\lambda] = \frac{a}{b^2},$$

where $a, b > 0$ and $\lambda \geq 0$

Gamma distribution

The distribution with the desired form described in previous slide turns out to be the Gamma distribution. Its pdf, mean, and variance (please verify the mean and variance) are given by

$$Gamma(\lambda|a,b) = \frac{1}{\Gamma(a)} b^a \lambda^{a-1} exp(-b\lambda)$$

$$E[\lambda] = \frac{a}{b}$$

$$Var[\lambda] = \frac{a}{b^2},$$

where $a, b > 0$ and $\lambda \geq 0$

N.B. when $a = 1$, Gamma reduces to the exponential distribution. When a is integer, it reduces to Erlang distribution

Posterior distribution of normal variable for fixed μ

Posterior probability given Normal likelihood (fixed mean) and Gamma prior

$$p(\lambda|x, a, b; \mu) = Const1 \cdot Gamma(\lambda|a, b) Norm(x|\lambda; \mu)$$

Posterior distribution of normal variable for fixed μ

Posterior probability given Normal likelihood (fixed mean) and Gamma prior

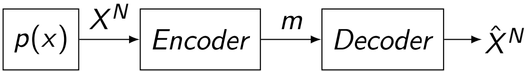
$$\begin{aligned} p(\lambda|x,a,b;\mu) =& Const1 \cdot Gamma(\lambda|a,b) Norm(x|\lambda;\mu) \\ =& Const2 \cdot \lambda^{a-1} \exp(-b\lambda)\sqrt{\lambda}\exp\left(-\lambda\frac{(x-\mu)^2}{2}\right) \\ =& Gamma\left(\lambda;\tilde{a},\tilde{b}\right), \end{aligned}$$

where $\tilde{a} \leftarrow a + \frac{1}{2}$ and $\tilde{b} \leftarrow b + \frac{(x-\mu)^2}{2}$

Conjugate prior summary

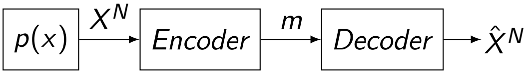
Distribution	Likelihood $p(\mathbf{x}\|\theta)$	Prior $p(\theta)$	Distribution
Bernoulli	$(1-\theta)^{(1-x)}\theta^x$	$\propto (1-\theta)^{(a-1)}\theta^{(b-1)}$	Beta
Binomial	$\propto (1-\theta)^{(N-x)}\theta^x$	$\propto (1-\theta)^{(a-1)}\theta^{(b-1)}$	Beta
Multinomial	$\propto \theta_1^{x_1}\theta_2^{x_2}\theta_3^{x_3}$	$\propto \theta_1^{\alpha_1-1}\theta_2^{\alpha_2-1}\theta_3^{\alpha_3-1}$	Dirichlet
Normal (fixed σ^2)	$\propto \exp\left(-\frac{(x-\theta)^2}{2\sigma^2}\right)$	$\propto \exp\left(-\frac{(\theta-\mu_0)^2}{2\sigma_0^2}\right)$	Normal
Normal (fixed μ)	$\propto \sqrt{\theta}\exp\left(-\frac{\theta(x-\mu)^2}{2}\right)$	$\propto \theta^{a-1} exp(-b\theta)$	Gamma
Poisson	$\propto \theta^x \exp(-\theta)$	$\propto \theta^{a-1} exp(-b\theta)$	Gamma

Rate-distortion problem



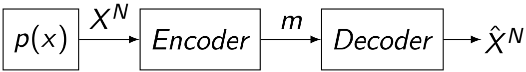
- We know that $H(X)$ bits are needed on average to represent each sample of a source X

Rate-distortion problem



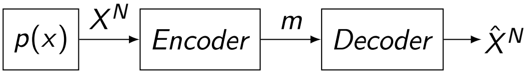
- We know that $H(X)$ bits are needed on average to represent each sample of a source X
- If X is continuous, there is no way to recover X precisely

Rate-distortion problem



- We know that $H(X)$ bits are needed on average to represent each sample of a source X
- If X is continuous, there is no way to recover X precisely
- Let say we are satisfied as long as we can recover X up to certain fidelity, how many bits are needed per sample?

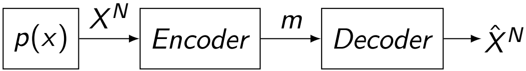
Rate-distortion problem



- We know that $H(X)$ bits are needed on average to represent each sample of a source X
- If X is continuous, there is no way to recover X precisely
- Let say we are satisfied as long as we can recover X up to certain fidelity, how many bits are needed per sample?
- There is an apparent rate (bits per sample) and distortion (fidelity) trade-off. We expect that needed rate is smaller if we allow a lower fidelity (higher distortion). What we are really interested in is a rate-distortion function

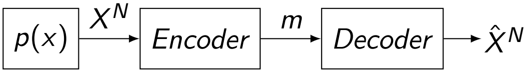
Rate-distortion function

$$m \in \{1, 2, \cdots, M\}$$



Rate-distortion function

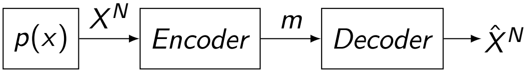
$$m \in \{1, 2, \cdots, M\}$$



$$R = \frac{\log M}{N}, \qquad D = E[d(\hat{X}^N, X^N)] = \frac{1}{N}\sum_{i=1}^{N} d(\hat{X}_i, X_i)$$

Rate-distortion function

$$m \in \{1, 2, \cdots, M\}$$

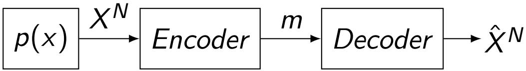


$$R = \frac{\log M}{N}, \qquad D = E[d(\hat{X}^N, X^N)] = \frac{1}{N}\sum_{i=1}^{N} d(\hat{X}_i, X_i)$$

- Maybe you can guess at this point. For given X and $\hat{X}$, the required rate is simply $I(X;\hat{X})$

Rate-distortion function

$$m \in \{1, 2, \cdots, M\}$$

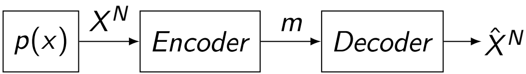


$$R = \frac{\log M}{N}, \qquad D = E[d(\hat{X}^N, X^N)] = \frac{1}{N}\sum_{i=1}^{N} d(\hat{X}_i, X_i)$$

- Maybe you can guess at this point. For given X and $\hat{X}$, the required rate is simply $I(X; \hat{X})$
- How is it related to the distortion though?

Rate-distortion function

$$m \in \{1, 2, \cdots, M\}$$

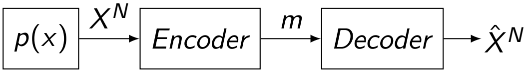


$$R = \frac{\log M}{N}, \qquad D = E[d(\hat{X}^N, X^N)] = \frac{1}{N}\sum_{i=1}^{N} d(\hat{X}_i, X_i)$$

- Maybe you can guess at this point. For given X and $\hat{X}$, the required rate is simply $I(X;\hat{X})$
- How is it related to the distortion though?
- Note that we have a freedom to pick $p(\hat{x}|x)$ such that $E[d(\hat{X}^N, X^N)]$ (less than or) equal to the desired $\mathcal{D}$

Rate-distortion function

$$m \in \{1, 2, \cdots, M\}$$



$$\boxed{p(x)} \xrightarrow{X^N} \boxed{Encoder} \xrightarrow{m} \boxed{Decoder} \rightarrow \hat{X}^N$$

$$R = \frac{\log M}{N}, \qquad D = E[d(\hat{X}^N, X^N)] = \frac{1}{N}\sum_{i=1}^{N} d(\hat{X}_i, X_i)$$

- Maybe you can guess at this point. For given X and $\hat{X}$, the required rate is simply $I(X;\hat{X})$
- How is it related to the distortion though?
- Note that we have a freedom to pick $p(\hat{x}|x)$ such that $E[d(\hat{X}^N, X^N)]$ (less than or) equal to the desired $\mathcal{D}$
- Therefore given $\mathcal{D}$, the rate-distortion function is simply

$$R(\mathcal{D}) = min_{p(\hat{x}|x)} I(\hat{X}; X)$$

such that $E[d(\hat{X}^N, X^N)] \leq \mathcal{D}$

Binary symmetric source

- Let’s try to compress outcome from a fair coin toss

Binary symmetric source

- Let’s try to compress outcome from a fair coin toss
- We know that we need 1 bit to compress the outcome losslessly, what if we have only 0.5 bit per sample?

Binary symmetric source

- Let's try to compress outcome from a fair coin toss
- We know that we need 1 bit to compress the outcome losslessly, what if we have only 0.5 bit per sample?
- In this case, we can't losslessly recover the outcome. But how good will we do?

Binary symmetric source

- Let's try to compress outcome from a fair coin toss
- We know that we need 1 bit to compress the outcome losslessly, what if we have only 0.5 bit per sample?
- In this case, we can't losslessly recover the outcome. But how good will we do?
- We need to introduce a distortion measure first. Note that we have two types of errors: taking head as tail and taking tail as head. A natural measure will just weights both error equally

$$d(X=H,\hat{X}=T)=d(X=T,\hat{X}=H)=1$$
$$d(X=H,\hat{X}=H)=d(X=T,\hat{X}=T)=0$$

Binary symmetric source

- Let's try to compress outcome from a fair coin toss
- We know that we need 1 bit to compress the outcome losslessly, what if we have only 0.5 bit per sample?
- In this case, we can't losslessly recover the outcome. But how good will we do?
- We need to introduce a distortion measure first. Note that we have two types of errors: taking head as tail and taking tail as head. A natural measure will just weights both error equally

$$d(X=H,\hat{X}=T)=d(X=T,\hat{X}=H)=1$$
$$d(X=H,\hat{X}=H)=d(X=T,\hat{X}=T)=0$$

- If rate is >1 bit, we know that distortion is 0. How about rate is 0, what distortion suppose to be?

Binary symmetric source

- Let's try to compress outcome from a fair coin toss
- We know that we need 1 bit to compress the outcome losslessly, what if we have only 0.5 bit per sample?
- In this case, we can't losslessly recover the outcome. But how good will we do?
- We need to introduce a distortion measure first. Note that we have two types of errors: taking head as tail and taking tail as head. A natural measure will just weights both error equally

$$d(X=H,\hat{X}=T)=d(X=T,\hat{X}=H)=1$$
$$d(X=H,\hat{X}=H)=d(X=T,\hat{X}=T)=0$$

- If rate is > 1 bit, we know that distortion is 0. How about rate is 0, what distortion suppose to be?
- If decoders know nothing, the best bet will be just always decode head (or tail). Then $D = E[d(X,H)] = 0.5$

Binary symmetric source

For $0 < D < 0.5$, denote Z as the prediction error such that $X = \hat{X} + Z$.

Binary symmetric source

For $0 < D < 0.5$, denote Z as the prediction error such that $X = \hat{X} + Z$. Note that

$$Pr(Z = 1) = D$$

Binary symmetric source

For $0 < D < 0.5$, denote Z as the prediction error such that $X = \hat{X} + Z$. Note that

$$Pr(Z = 1) = D$$

$$R = min_{p(\hat{x}|x)} I(\hat{X}; X) = min_{p(\hat{x}|x)} H(X) - H(X|\hat{X})$$

Binary symmetric source

For $0 < D < 0.5$, denote Z as the prediction error such that $X = \hat{X} + Z$. Note that

$$Pr(Z = 1) = D$$

$$\begin{aligned} R &= min_{p(\hat{x}|x)} I(\hat{X}; X) = min_{p(\hat{x}|x)} H(X) - H(X|\hat{X}) \\ &= min_{p(\hat{x}|x)} H(X) - H(\hat{X} + Z|\hat{X}) \end{aligned}$$

Binary symmetric source

For $0 < D < 0.5$, denote Z as the prediction error such that $X = \hat{X} + Z$. Note that

$$Pr(Z = 1) = D$$

$$\begin{aligned} R &= min_{p(\hat{x}|x)} I(\hat{X}; X) = min_{p(\hat{x}|x)} H(X) - H(X|\hat{X}) \\ &= min_{p(\hat{x}|x)} H(X) - H(\hat{X} + Z|\hat{X}) \\ &= min_{p(\hat{x}|x)} H(X) - H(Z|\hat{X}) \end{aligned}$$

Binary symmetric source

For $0 < D < 0.5$, denote Z as the prediction error such that $X = \hat{X} + Z$. Note that

$$Pr(Z = 1) = D$$

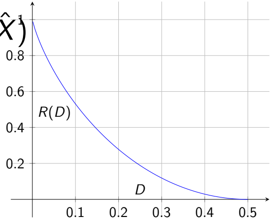
$$\begin{aligned} R &= min_{p(\hat{x}|x)} I(\hat{X}; X) = min_{p(\hat{x}|x)} H(X) - H(X|\hat{X}) \\ &= min_{p(\hat{x}|x)} H(X) - H(\hat{X} + Z|\hat{X}) \\ &= min_{p(\hat{x}|x)} H(X) - H(Z|\hat{X}) \\ &= min_{p(\hat{x}|x)} H(X) - H(Z) \end{aligned}$$

Binary symmetric source

For $0 < D < 0.5$, denote Z as the prediction error such that $X = \hat{X} + Z$. Note that

$$Pr(Z = 1) = D$$

$$\begin{aligned} R &= min_{p(\hat{x}|x)} I(\hat{X}; X) = min_{p(\hat{x}|x)} H(X) - H(X|\hat{X}) \\ &= min_{p(\hat{x}|x)} H(X) - H(\hat{X} + Z|\hat{X}) \\ &= min_{p(\hat{x}|x)} H(X) - H(Z|\hat{X}) \\ &= min_{p(\hat{x}|x)} H(X) - H(Z) \\ &= 1 - H(D) \end{aligned}$$



N.B. The above can be modelled by $\hat{X}$ going through a BSC with cross-over probability D with the output X. Such BSC is often called a test channel. Note that the channel has to be symmetric. Otherwise, Z will not be independent of $\hat{X}$

Lecture 13

Previously...

- Converse Proof of Channel Coding Theorem
- Non-white Gaussian Channel
- Rate-distortion problems

This time

- Proof of the Rate-distortion Theorem

Gaussian source

- Consider $X \sim \mathcal{N}(0, \sigma_X^2)$. To determine the rate-distortion function, we need first to decide the distortion measure. An intuitive will be just the square error. That is,

$$d(\hat{X}, X) = (\hat{X} - X)^2$$

Gaussian source

- Consider $X \sim \mathcal{N}(0, \sigma_X^2)$. To determine the rate-distortion function, we need first to decide the distortion measure. An intuitive will be just the square error. That is,

$$d(\hat{X}, X) = (\hat{X} - X)^2$$

- Given $E[d(\hat{X}, X)] = D$, what is the minimum rate required?

Gaussian source

- Consider $X \sim \mathcal{N}(0, \sigma_X^2)$. To determine the rate-distortion function, we need first to decide the distortion measure. An intuitive will be just the square error. That is,

$$d(\hat{X}, X) = (\hat{X} - X)^2$$

- Given $E[d(\hat{X}, X)] = D$, what is the minimum rate required?
- Like before, let us denote $Z = X - \hat{X}$ as the prediction error. Note that $Var(Z) = D$

Gaussian source

- Consider $X \sim \mathcal{N}(0, \sigma_X^2)$. To determine the rate-distortion function, we need first to decide the distortion measure. An intuitive will be just the square error. That is,

$$d(\hat{X}, X) = (\hat{X} - X)^2$$

- Given $E[d(\hat{X}, X)] = D$, what is the minimum rate required?
- Like before, let us denote $Z = X - \hat{X}$ as the prediction error. Note that $Var(Z) = D$

$$R(D) = min_{p(\hat{x}|x)} I(\hat{X}; X) = min_{p(\hat{x}|x)} h(X) - h(X|\hat{X})$$

Gaussian source

- Consider $X \sim \mathcal{N}(0, \sigma_X^2)$. To determine the rate-distortion function, we need first to decide the distortion measure. An intuitive will be just the square error. That is,

$$d(\hat{X}, X) = (\hat{X} - X)^2$$

- Given $E[d(\hat{X}, X)] = D$, what is the minimum rate required?
- Like before, let us denote $Z = X - \hat{X}$ as the prediction error. Note that $Var(Z) = D$

$$\begin{aligned} R(D) &= min_{p(\hat{x}|x)} I(\hat{X}; X) = min_{p(\hat{x}|x)} h(X) - h(X|\hat{X}) \\ &= min_{p(\hat{x}|x)} h(X) - h(Z + \hat{X}|\hat{X}) \end{aligned}$$

Gaussian source

- Consider $X \sim \mathcal{N}(0, \sigma_X^2)$. To determine the rate-distortion function, we need first to decide the distortion measure. An intuitive will be just the square error. That is,

$$d(\hat{X}, X) = (\hat{X} - X)^2$$

- Given $E[d(\hat{X}, X)] = D$, what is the minimum rate required?
- Like before, let us denote $Z = X - \hat{X}$ as the prediction error. Note that $Var(Z) = D$

$$\begin{aligned} R(D) &= min_{p(\hat{x}|x)} I(\hat{X}; X) = min_{p(\hat{x}|x)} h(X) - h(X|\hat{X}) \\ &= min_{p(\hat{x}|x)} h(X) - h(Z + \hat{X}|\hat{X}) \\ &= min_{p(\hat{x}|x)} h(X) - h(Z|\hat{X}) \end{aligned}$$

Gaussian source

- Consider $X \sim \mathcal{N}(0, \sigma_X^2)$. To determine the rate-distortion function, we need first to decide the distortion measure. An intuitive will be just the square error. That is,

$$d(\hat{X}, X) = (\hat{X} - X)^2$$

- Given $E[d(\hat{X}, X)] = D$, what is the minimum rate required?
- Like before, let us denote $Z = X - \hat{X}$ as the prediction error. Note that $Var(Z) = D$

$$\begin{aligned} R(D) &= min_{p(\hat{x}|x)} I(\hat{X}; X) = min_{p(\hat{x}|x)} h(X) - h(X|\hat{X}) \\ &= min_{p(\hat{x}|x)} h(X) - h(Z + \hat{X}|\hat{X}) \\ &= min_{p(\hat{x}|x)} h(X) - h(Z|\hat{X}) \\ &= min_{p(\hat{x}|x)} h(X) - h(Z) \end{aligned}$$

Gaussian source

- Consider $X \sim \mathcal{N}(0, \sigma_X^2)$. To determine the rate-distortion function, we need first to decide the distortion measure. An intuitive will be just the square error. That is,

$$d(\hat{X}, X) = (\hat{X} - X)^2$$

- Given $E[d(\hat{X}, X)] = D$, what is the minimum rate required?
- Like before, let us denote $Z = X - \hat{X}$ as the prediction error. Note that $Var(Z) = D$

$$\begin{aligned} R(D) &= min_{p(\hat{x}|x)} I(\hat{X}; X) = min_{p(\hat{x}|x)} h(X) - h(X|\hat{X}) \\ &= min_{p(\hat{x}|x)} h(X) - h(Z + \hat{X}|\hat{X}) \\ &= min_{p(\hat{x}|x)} h(X) - h(Z|\hat{X}) \\ &= min_{p(\hat{x}|x)} h(X) - h(Z) \\ &= \frac{1}{2} \log \frac{\sigma_X^2}{D} \end{aligned}$$

Forward proof

Forward statement

Given distortion constraint $\mathcal{D}$, we can find scheme such that the require rate is no bigger than

$$R(\mathcal{D}) = \min_{p(\hat{x}|x)} I(X;\hat{X}),$$

where the $\hat{X}$ introduced by $p(\hat{x}|x)$ should satisfy $E[d(X,\hat{X})] \leq \mathcal{D}$

Forward proof

Forward statement

Given distortion constraint $\mathcal{D}$, we can find scheme such that the require rate is no bigger than

$$R(\mathcal{D}) = \min_{p(\hat{x}|x)} I(X;\hat{X}),$$

where the $\hat{X}$ introduced by $p(\hat{x}|x)$ should satisfy $E[d(X,\hat{X})] \leq \mathcal{D}$

Code book construction

Let say $p^*(\hat{x}|x)$ is the distribution that achieve the rate-distortion optimiation problem. Randomly construct 2^{NR} codewords as follows

Forward proof

Forward statement

Given distortion constraint $\mathcal{D}$, we can find scheme such that the require rate is no bigger than

$$R(\mathcal{D}) = \min_{p(\hat{x}|x)} I(X;\hat{X}),$$

where the $\hat{X}$ introduced by $p(\hat{x}|x)$ should satisfy $E[d(X,\hat{X})] \leq \mathcal{D}$

Code book construction

Let say $p^*(\hat{x}|x)$ is the distribution that achieve the rate-distortion optimiation problem. Randomly construct 2^{NR} codewords as follows

- Sample X from the source and pass X into $p^*(\hat{x}|x)$ to obtain $\hat{X}$

Forward proof

Forward statement

Given distortion constraint $\mathcal{D}$, we can find scheme such that the require rate is no bigger than

$$R(\mathcal{D}) = \min_{p(\hat{x}|x)} I(X;\hat{X}),$$

where the $\hat{X}$ introduced by $p(\hat{x}|x)$ should satisfy $E[d(X,\hat{X})] \leq \mathcal{D}$

Code book construction

Let say $p^*(\hat{x}|x)$ is the distribution that achieve the rate-distortion optimiation problem. Randomly construct 2^{NR} codewords as follows

- Sample X from the source and pass X into $p^*(\hat{x}|x)$ to obtain $\hat{X}$
- Repeat this N time to get a length-N codeword
- Store the i-th codeword as $\mathbf{C}(i)$

Forward proof

Forward statement

Given distortion constraint $\mathcal{D}$, we can find scheme such that the require rate is no bigger than

$$R(\mathcal{D}) = \min_{p(\hat{x}|x)} I(X;\hat{X}),$$

where the $\hat{X}$ introduced by $p(\hat{x}|x)$ should satisfy $E[d(X,\hat{X})] \leq \mathcal{D}$

Code book construction

Let say $p^*(\hat{x}|x)$ is the distribution that achieve the rate-distortion optimiation problem. Randomly construct 2^{NR} codewords as follows

- Sample X from the source and pass X into $p^*(\hat{x}|x)$ to obtain $\hat{X}$
- Repeat this N time to get a length-N codeword
- Store the i-th codeword as $\mathbf{C}(i)$

Note that the code rate is $\frac{\log 2^{NR}}{N} = R$ as desired

Covering lemma and distortion typical sequences

We say joint typical sequences x^N and $\hat{x}^N$ are distortion typical ($(x^N, \hat{x}^N) \in \mathcal{A}_{d,\epsilon}^N$) if $|d(x^N, \hat{x}^N) - E[d(X, \hat{X})]| \leq \epsilon$

Covering lemma and distortion typical sequences

We say joint typical sequences x^N and $\hat{x}^N$ are distortion typical ($(x^N, \hat{x}^N) \in \mathcal{A}^N_{d,\epsilon}$) if $|d(x^N, \hat{x}^N) - E[d(X, \hat{X})]| \le \epsilon$

- By LLN, every pair of sequences sampled from the joint source will virtually be distortion typical

Covering lemma and distortion typical sequences

We say joint typical sequences x^N and $\hat{x}^N$ are distortion typical ($(x^N, \hat{x}^N) \in \mathcal{A}^N_{d,\epsilon}$) if $|d(x^N, \hat{x}^N) - E[d(X, \hat{X})]| \leq \epsilon$

- By LLN, every pair of sequences sampled from the joint source will virtually be distortion typical
- Consequently, $(1-\delta)2^{N(H(X,\hat{X})-\epsilon)} \leq |\mathcal{A}^N_{d,\epsilon}| \leq 2^{N(H(X,\hat{X})+\epsilon)}$ as before

Covering lemma and distortion typical sequences

We say joint typical sequences x^N and $\hat{x}^N$ are distortion typical ($(x^N, \hat{x}^N) \in \mathcal{A}^N_{d,\epsilon}$) if $|d(x^N, \hat{x}^N) - E[d(X, \hat{X})]| \leq \epsilon$

- By LLN, every pair of sequences sampled from the joint source will virtually be distortion typical
- Consequently, $(1-\delta)2^{N(H(X,\hat{X})-\epsilon)} \leq |\mathcal{A}^N_{d,\epsilon}| \leq 2^{N(H(X,\hat{X})+\epsilon)}$ as before
- For two independently drawn sequences $\hat{X}^N$ and X^N, the probability for them to be distortion typical will be just the same as before. In particular, $(1-\delta)2^{-N(I(X;\hat{X})-3\epsilon)} \leq Pr((X^N, \hat{X}^N) \in \mathcal{A}^N_{d,\epsilon}(X, \hat{X}))$

Covering lemma for distortion typical sequences

Covering lemma for distortion typical sequences

$$Pr((X^N, \hat{X}^N(m)) \notin \mathcal{A}_{d,\epsilon}^{(N)}(X, \hat{X}) \text{ for all } m)$$

Covering lemma for distortion typical sequences

$$
\begin{aligned}
& Pr((X^N, \hat{X}^N(m)) \notin \mathcal{A}_{d,\epsilon}^{(N)}(X, \hat{X}) \text{ for all } m) \\
= & \prod_{m=1}^{M} Pr((X^N, \hat{X}^N(m)) \notin \mathcal{A}_{d,\epsilon}^{(N)}(\hat{X}, X))
\end{aligned}
$$

Covering lemma for distortion typical sequences

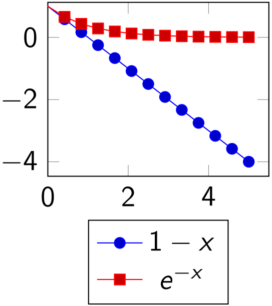
$$
\begin{aligned}
& Pr((X^N, \hat{X}^N(m)) \notin \mathcal{A}_{d,\epsilon}^{(N)}(X,\hat{X}) \text{ for all } m) \\
&= \prod_{m=1}^{M} Pr((X^N, \hat{X}^N(m)) \notin \mathcal{A}_{d,\epsilon}^{(N)}(\hat{X},X)) \\
&= \prod_{m=1}^{M} \left[1 - Pr((X^N, \hat{X}^N(m)) \in \mathcal{A}_{d,\epsilon}^{(N)}(\hat{X},X))\right]
\end{aligned}
$$

Covering lemma for distortion typical sequences

$$
\begin{aligned}
&Pr((X^N, \hat{X}^N(m)) \notin \mathcal{A}_{d,\epsilon}^{(N)}(X, \hat{X}) \text{ for all } m)\\
=&\prod_{m=1}^{M} Pr((X^N, \hat{X}^N(m)) \notin \mathcal{A}_{d,\epsilon}^{(N)}(\hat{X}, X))\\
=&\prod_{m=1}^{M} \left[1 - Pr((X^N, \hat{X}^N(m)) \in \mathcal{A}_{d,\epsilon}^{(N)}(\hat{X}, X))\right]\\
\leq&(1-(1-\delta)2^{-N(I(\hat{X};X)+3\epsilon)})^M
\end{aligned}
$$

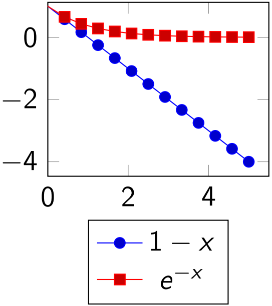
Covering lemma for distortion typical sequences

$$
\begin{aligned}
& Pr((X^N, \hat{X}^N(m)) \notin \mathcal{A}_{d,\epsilon}^{(N)}(X, \hat{X}) \text{ for all } m) \\
= & \prod_{m=1}^{M} Pr((X^N, \hat{X}^N(m)) \notin \mathcal{A}_{d,\epsilon}^{(N)}(\hat{X}, X)) \\
= & \prod_{m=1}^{M} \left[1 - Pr((X^N, \hat{X}^N(m)) \in \mathcal{A}_{d,\epsilon}^{(N)}(\hat{X}, X))\right] \\
\leq & (1 - (1-\delta)2^{-N(I(\hat{X};X)+3\epsilon)})^M \\
\leq & \exp(-M(1-\delta)2^{-N(I(\hat{X};X)+3\epsilon)})
\end{aligned}
$$



Covering lemma for distortion typical sequences

$$
\begin{aligned}
& Pr((X^N, \hat{X}^N(m)) \notin \mathcal{A}_{d,\epsilon}^{(N)}(X, \hat{X}) \text{ for all } m) \\
= & \prod_{m=1}^{M} Pr((X^N, \hat{X}^N(m)) \notin \mathcal{A}_{d,\epsilon}^{(N)}(\hat{X}, X)) \\
= & \prod_{m=1}^{M} \left[1 - Pr((X^N, \hat{X}^N(m)) \in \mathcal{A}_{d,\epsilon}^{(N)}(\hat{X}, X)) \right] \\
\leq & (1 - (1-\delta) 2^{-N(I(\hat{X};X)+3\epsilon)})^M \\
\leq & \exp(-M(1-\delta) 2^{-N(I(\hat{X};X)+3\epsilon)}) \\
\leq & \exp(-(1-\delta) 2^{-N(I(\hat{X};X)-R+3\epsilon)}) \to 0 \text{ as } N \to \infty \text{ and } R > I(X;\hat{X}) + 3\epsilon
\end{aligned}
$$



Forward proof

Encoding

Given input X^N, find out of the codewords the one that is jointly typical with X^N. And say, if the codeword is $\mathbf{C}(i)$, output index i to the decoder

Forward proof

Encoding

Given input X^N, find out of the codewords the one that is jointly typical with X^N. And say, if the codeword is $\mathbf{C}(i)$, output index i to the decoder

Decoding

Upon receiving the index i, simply output $\mathbf{C}(i)$

Forward proof

Encoding

Given input X^N, find out of the codewords the one that is jointly typical with X^N. And say, if the codeword is $\mathbf{C}(i)$, output index i to the decoder

Decoding

Upon receiving the index i, simply output $\mathbf{C}(i)$

Performance analysis

- First of all, the only point of failure lies on encoding, that is when the encoder cannot find a codeword jointly typical with X^N

Forward proof

Encoding

Given input X^N, find out of the codewords the one that is jointly typical with X^N. And say, if the codeword is $\mathbf{C}(i)$, output index i to the decoder

Decoding

Upon receiving the index i, simply output $\mathbf{C}(i)$

Performance analysis

- First of all, the only point of failure lies on encoding, that is when the encoder cannot find a codeword jointly typical with X^N
- By covering Lemma, encoding failure is negligible as long as $R > I(X;\hat{X})$

Forward proof

Encoding

Given input X^N, find out of the codewords the one that is jointly typical with X^N. And say, if the codeword is $\mathbf{C}(i)$, output index i to the decoder

Decoding

Upon receiving the index i, simply output $\mathbf{C}(i)$

Performance analysis

- First of all, the only point of failure lies on encoding, that is when the encoder cannot find a codeword jointly typical with X^N
- By covering Lemma, encoding failure is negligible as long as $R > I(X; \hat{X})$
- If encoding is successful, $\mathbf{C}(i)$ and X^N should be distortion typical. Therefore, $E[d(\mathbf{C}(i); X^N)] \sim E[d(\hat{X}, X)] \leq \mathcal{D}$ as desired

Converse proof

Converse statement

If rate is smaller than $R(\mathcal{D})$, distortion will be larger than $\mathcal{D}$

Converse proof

Converse statement

If rate is smaller than $R(\mathcal{D})$, distortion will be larger than $\mathcal{D}$

Alternative statement

If distortion is less than or equal to $\mathcal{D}$, the rate must be larger than $R(\mathcal{D})$

Converse proof

Converse statement

If rate is smaller than $R(\mathcal{D})$, distortion will be larger than $\mathcal{D}$

Alternative statement

If distortion is less than or equal to $\mathcal{D}$, the rate must be larger than $R(\mathcal{D})$

In the proof, we need to use the convex property of $R(\mathcal{D})$. That is,

$$R(a\mathcal{D}_1 + (1-a)\mathcal{D}_2) \geq aR(\mathcal{D}_1) + (1-a)R(\mathcal{D}_2)$$

So we will digress a little bit to show this convex property first

Log-sum inequality

Log-sum inequality

For any $a_1, \cdots, a_n \geq 0$ and $b_1, \cdots, b_n \geq 0$, we have

$$\sum_i a_i \log_2 \frac{a_i}{b_i} \geq \left(\sum_i a_i \right) \log_2 \frac{\sum_i a_i}{\sum_i b_i}.$$

Log-sum inequality

Log-sum inequality

For any $a_1, \cdots, a_n \geq 0$ and $b_1, \cdots, b_n \geq 0$, we have

$$\sum_i a_i \log_2 \frac{a_i}{b_i} \geq \left(\sum_i a_i \right) \log_2 \frac{\sum_i a_i}{\sum_i b_i}.$$

Proof

We can define two distributions $p(x)$ and $q(x)$ with $p(x_i) = \frac{a_i}{\sum_i a_i}$ and $q(x_i) = \frac{b_i}{\sum_i b_i}$. Since $p(x)$ and $q(x)$ are both non-negative and sum up to 1, they are indeed valid probability mass functions.

Log-sum inequality

Log-sum inequality

For any $a_1, \cdots, a_n \geq 0$ and $b_1, \cdots, b_n \geq 0$, we have

$$\sum_i a_i \log_2 \frac{a_i}{b_i} \geq \left(\sum_i a_i \right) \log_2 \frac{\sum_i a_i}{\sum_i b_i}.$$

Proof

We can define two distributions $p(x)$ and $q(x)$ with $p(x_i) = \frac{a_i}{\sum_i a_i}$ and $q(x_i) = \frac{b_i}{\sum_i b_i}$. Since $p(x)$ and $q(x)$ are both non-negative and sum up to 1, they are indeed valid probability mass functions. Then, we have

Log-sum inequality

Log-sum inequality

For any $a_1, \cdots, a_n \geq 0$ and $b_1, \cdots, b_n \geq 0$, we have

$$\sum_i a_i \log_2 \frac{a_i}{b_i} \geq \left(\sum_i a_i \right) \log_2 \frac{\sum_i a_i}{\sum_i b_i}.$$

Proof

We can define two distributions $p(x)$ and $q(x)$ with $p(x_i) = \frac{a_i}{\sum_i a_i}$ and $q(x_i) = \frac{b_i}{\sum_i b_i}$. Since $p(x)$ and $q(x)$ are both non-negative and sum up to 1, they are indeed valid probability mass functions. Then, we have

$$0 \leq KL(p(x)\|q(x)) = \sum_i p(x_i) \log_2 \frac{p(x_i)}{q(x_i)}$$

Log-sum inequality

Log-sum inequality

For any $a_1, \cdots, a_n \geq 0$ and $b_1, \cdots, b_n \geq 0$, we have

$$\sum_i a_i \log_2 \frac{a_i}{b_i} \geq \left(\sum_i a_i\right) \log_2 \frac{\sum_i a_i}{\sum_i b_i}.$$

Proof

We can define two distributions $p(x)$ and $q(x)$ with $p(x_i) = \frac{a_i}{\sum_i a_i}$ and $q(x_i) = \frac{b_i}{\sum_i b_i}$. Since $p(x)$ and $q(x)$ are both non-negative and sum up to 1, they are indeed valid probability mass functions. Then, we have

$$\begin{aligned} 0 \leq KL(p(x)\|q(x)) &= \sum_i p(x_i) \log_2 \frac{p(x_i)}{q(x_i)} \\ &= \sum_i \frac{a_i}{\sum_i a_i} \left(\log_2 \frac{a_i}{b_i} - \log_2 \frac{\sum_i a_i}{\sum_i b_i}\right) \end{aligned}$$

Convexity of KL-Divergence

For any four distributions $p_1(\cdot)$, $p_2(\cdot)$, $q_1(\cdot)$, and $q_2(\cdot)$, we have

$$\lambda_1 KL(p_1\|q_1) + \lambda_2 KL(p_2\|q_2) \geq KL(\lambda_1 p_1 + \lambda_2 p_2\|\lambda_1 q_1 + \lambda_2 q_2),$$

where $\lambda_1, \lambda_2 \geq 0$ and $\lambda_1 + \lambda_2 = 1$

Convexity of KL-Divergence

For any four distributions $p_1(\cdot)$, $p_2(\cdot)$, $q_1(\cdot)$, and $q_2(\cdot)$, we have

$$\lambda_1 KL(p_1\|q_1) + \lambda_2 KL(p_2\|q_2) \geq KL(\lambda_1 p_1 + \lambda_2 p_2\|\lambda_1 q_1 + \lambda_2 q_2),$$

where $\lambda_1, \lambda_2 \geq 0$ and $\lambda_1 + \lambda_2 = 1$

Proof

$$\begin{aligned}&\lambda_1 KL(p_1\|q_1) + \lambda_2 KL(p_2\|q_2)\\ =&\lambda_1 \sum_{x\in\mathcal{X}} p_1(x) \log \frac{p_1(x)}{q_1(x)} + \lambda_2 \sum_{x\in\mathcal{X}} p_2(x) \log \frac{p_2(x)}{q_2(x)}\end{aligned}$$

Convexity of KL-Divergence

For any four distributions $p_1(\cdot)$, $p_2(\cdot)$, $q_1(\cdot)$, and $q_2(\cdot)$, we have

$$\lambda_1 KL(p_1\|q_1) + \lambda_2 KL(p_2\|q_2) \geq KL(\lambda_1 p_1 + \lambda_2 p_2\|\lambda_1 q_1 + \lambda_2 q_2),$$

where $\lambda_1, \lambda_2 \geq 0$ and $\lambda_1 + \lambda_2 = 1$

Proof

$$\begin{aligned}
&\lambda_1 KL(p_1\|q_1) + \lambda_2 KL(p_2\|q_2)\\
=&\lambda_1 \sum_{x\in\mathcal{X}} p_1(x) \log \frac{p_1(x)}{q_1(x)} + \lambda_2 \sum_{x\in\mathcal{X}} p_2(x) \log \frac{p_2(x)}{q_2(x)}\\
=&\sum_{x\in\mathcal{X}} \lambda_1 p_1(x) \log \frac{\lambda_1 p_1(x)}{\lambda_1 q_1(x)} + \lambda_2 p_2(x) \log \frac{\lambda_2 p_2(x)}{\lambda_2 q_2(x)}
\end{aligned}$$

Convexity of KL-Divergence

For any four distributions $p_1(\cdot)$, $p_2(\cdot)$, $q_1(\cdot)$, and $q_2(\cdot)$, we have

$$\lambda_1 KL(p_1\|q_1) + \lambda_2 KL(p_2\|q_2) \geq KL(\lambda_1 p_1 + \lambda_2 p_2\|\lambda_1 q_1 + \lambda_2 q_2),$$

where $\lambda_1, \lambda_2 \geq 0$ and $\lambda_1 + \lambda_2 = 1$

Proof

$$\begin{aligned}
&\lambda_1 KL(p_1\|q_1) + \lambda_2 KL(p_2\|q_2) \\
=&\lambda_1 \sum_{x\in\mathcal{X}} p_1(x) \log \frac{p_1(x)}{q_1(x)} + \lambda_2 \sum_{x\in\mathcal{X}} p_2(x) \log \frac{p_2(x)}{q_2(x)} \\
=&\sum_{x\in\mathcal{X}} \lambda_1 p_1(x) \log \frac{\lambda_1 p_1(x)}{\lambda_1 q_1(x)} + \lambda_2 p_2(x) \log \frac{\lambda_2 p_2(x)}{\lambda_2 q_2(x)} \\
\geq&\sum_{x\in\mathcal{X}} (\lambda_1 p_1(x) + \lambda_2 p_2(x)) \log \frac{\lambda_1 p_1(x) + \lambda_2 p_2(x)}{\lambda_1 q_1(x) + \lambda_2 q_2(x)} \quad \text{(by log-sum inequality)}
\end{aligned}$$

Convexity of KL-Divergence

For any four distributions $p_1(\cdot)$, $p_2(\cdot)$, $q_1(\cdot)$, and $q_2(\cdot)$, we have

$$\lambda_1 KL(p_1\|q_1)+\lambda_2 KL(p_2\|q_2)\geq KL(\lambda_1 p_1+\lambda_2 p_2\|\lambda_1 q_1+\lambda_2 q_2),$$

where $\lambda_1,\lambda_2\geq 0$ and $\lambda_1+\lambda_2=1$

Proof

$$\begin{aligned}
&\lambda_1 KL(p_1\|q_1)+\lambda_2 KL(p_2\|q_2)\\
=&\lambda_1\sum_{x\in\mathcal{X}}p_1(x)\log\frac{p_1(x)}{q_1(x)}+\lambda_2\sum_{x\in\mathcal{X}}p_2(x)\log\frac{p_2(x)}{q_2(x)}\\
=&\sum_{x\in\mathcal{X}}\lambda_1 p_1(x)\log\frac{\lambda_1 p_1(x)}{\lambda_1 q_1(x)}+\lambda_2 p_2(x)\log\frac{\lambda_2 p_2(x)}{\lambda_2 q_2(x)}\\
\geq&\sum_{x\in\mathcal{X}}(\lambda_1 p_1(x)+\lambda_2 p_2(x))\log\frac{\lambda_1 p_1(x)+\lambda_2 p_2(x)}{\lambda_1 q_1(x)+\lambda_2 q_2(x)}\quad\text{(by log-sum inequality)}\\
=&KL(\lambda_1 p_1+\lambda_2 p_2\|\lambda_1 q_1+\lambda_2 q_2)
\end{aligned}$$

Convexity of $I(X;Y)$ with respect to $p(y|x)$

For any random variables X and Y, $I(X;Y)$ is a convex function of $p(y|x)$ for a fixed $p(x)$

Convexity of $I(X;Y)$ with respect to $p(y|x)$

For any random variables X and Y, $I(X;Y)$ is a convex function of $p(y|x)$ for a fixed $p(x)$

Remark

$I(X;Y)$ is concave with respect to $p(x)$ for fixed $p(y|x)$ though. A proof is given in Cover and Thomas and will be omitted here

Convexity of $I(X;Y)$ with respect to $p(y|x)$

For any random variables X and Y, $I(X;Y)$ is a convex function of $p(y|x)$ for a fixed $p(x)$

Remark

$I(X;Y)$ is concave with respect to $p(x)$ for fixed $p(y|x)$ though. A proof is given in Cover and Thomas and will be omitted here

Proof

Let us write

$$\begin{aligned} I(X;Y) &= KL(p(x,y)\|p(x)p(y)) \\ &= KL\left(p(x)p(y|x)\middle\|p(x)\sum_x p(x)p(y|x)\right) \triangleq f(p(y|x)) \end{aligned}$$

Convexity of $I(X;Y)$ with respect to $p(y|x)$

For any random variables X and Y, $I(X;Y)$ is a convex function of $p(y|x)$ for a fixed $p(x)$

Remark

$I(X;Y)$ is concave with respect to $p(x)$ for fixed $p(y|x)$ though. A proof is given in Cover and Thomas and will be omitted here

Proof

Let us write

$$\begin{aligned} I(X;Y) &= KL(p(x,y)\|p(x)p(y)) \\ &= KL\left(p(x)p(y|x)\middle\|p(x)\sum_x p(x)p(y|x)\right) \triangleq f(p(y|x)) \end{aligned}$$

We want to show

$$\lambda f(p_1(y|x)) + (1-\lambda) f(p_2(y|x)) \geq f(\lambda p_1(y|x) + (1-\lambda) p_2(y|x))$$

Proof

Continue from previous slide, we have

$$
\begin{aligned}
&\lambda f(p_1(y|x)) + (1-\lambda) f(p_2(y|x)) \\
=&\lambda KL\left(p(x)p_1(y|x)\middle\| p(x)\sum_x p(x)p_1(y|x)\right) \\
&\quad + (1-\lambda) KL\left(p(x)p_2(y|x)\middle\| p(x)\sum_x p(x)p_2(y|x)\right)
\end{aligned}
$$

Proof

Continue from previous slide, we have

$$
\begin{aligned}
&\lambda f(p_1(y|x)) + (1-\lambda) f(p_2(y|x)) \\
=&\lambda KL\left(p(x)p_1(y|x)\middle\| p(x)\sum_x p(x)p_1(y|x)\right) \\
&\quad + (1-\lambda)KL\left(p(x)p_2(y|x)\middle\| p(x)\sum_x p(x)p_2(y|x)\right) \\
\geq& KL\left(\lambda p(x)p_1(y|x) + (1-\lambda)p(x)p_2(y|x)\middle\| \lambda p(x)\sum_x p(x)p_1(y|x)\right. \\
&\quad \left. + (1-\lambda)p(x)\sum_x p(x)p_2(y|x)\right)
\end{aligned}
$$

Proof

Continue from previous slide, we have

$$
\begin{aligned}
&\lambda f(p_1(y|x)) + (1-\lambda) f(p_2(y|x)) \\
=&\lambda KL\left(p(x)p_1(y|x)\middle\| p(x)\sum_x p(x)p_1(y|x)\right) \\
&\quad + (1-\lambda) KL\left(p(x)p_2(y|x)\middle\| p(x)\sum_x p(x)p_2(y|x)\right) \\
\geq& KL\left(\lambda p(x)p_1(y|x) + (1-\lambda)p(x)p_2(y|x)\middle\| \lambda p(x)\sum_x p(x)p_1(y|x)\right. \\
&\quad \left. + (1-\lambda)p(x)\sum_x p(x)p_2(y|x)\right) \\
=& KL\left(p(x)[\lambda p_1(y|x) + (1-\lambda)p_2(y|x)]\middle\| p(x)\sum_x p(x)[\lambda p_1(y|x) + (1-\lambda)p_2(y|x)]\right)
\end{aligned}
$$

Proof

Continue from previous slide, we have

$$\begin{aligned}
&\lambda f(p_1(y|x)) + (1-\lambda) f(p_2(y|x)) \\
=&\lambda KL\left(p(x)p_1(y|x)\middle\| p(x)\sum_x p(x)p_1(y|x)\right) \\
&\quad + (1-\lambda) KL\left(p(x)p_2(y|x)\middle\| p(x)\sum_x p(x)p_2(y|x)\right) \\
\geq&KL\left(\lambda p(x)p_1(y|x) + (1-\lambda)p(x)p_2(y|x)\middle\| \lambda p(x)\sum_x p(x)p_1(y|x)\right. \\
&\quad \left. + (1-\lambda)p(x)\sum_x p(x)p_2(y|x)\right) \\
=&KL\left(p(x)[\lambda p_1(y|x) + (1-\lambda)p_2(y|x)]\middle\| p(x)\sum_x p(x)[\lambda p_1(y|x) + (1-\lambda)p_2(y|x)]\right) \\
=&f(\lambda p_1(y|x) + (1-\lambda)p_2(y|x))
\end{aligned}$$

Convexity of $R(\mathcal{D})$

Recall that $R(\mathcal{D}) = \min_{p(\hat{x}|x)} I(\hat{X}; X)$ with $E[d(X, \hat{X})] \leq \mathcal{D}$
We want to show that

$$R(\lambda\mathcal{D}_1 + (1-\lambda)\mathcal{D}_2) \leq \lambda R(\mathcal{D}_1) + (1-\lambda)R(\mathcal{D}_2)$$

Convexity of $R(\mathcal{D})$

Recall that $R(\mathcal{D}) = \min_{p(\hat{x}|x)} I(\hat{X}; X)$ with $E[d(X, \hat{X})] \leq \mathcal{D}$
We want to show that

$$R(\lambda\mathcal{D}_1 + (1-\lambda)\mathcal{D}_2) \leq \lambda R(\mathcal{D}_1) + (1-\lambda)R(\mathcal{D}_2)$$

Proof

Let $p_1^*(\hat{x}|x)$ and $p_2^*(\hat{x}|x)$ be the distributions that optimize $R(\mathcal{D}_1)$ and $R(\mathcal{D}_2)$. Let's try to time share between the two distributions.

Convexity of $R(\mathcal{D})$

Recall that $R(\mathcal{D}) = \min_{p(\hat{x}|x)} I(\hat{X};X)$ with $E[d(X,\hat{X})] \leq \mathcal{D}$
We want to show that

$$R(\lambda\mathcal{D}_1 + (1-\lambda)\mathcal{D}_2) \leq \lambda R(\mathcal{D}_1) + (1-\lambda)R(\mathcal{D}_2)$$

Proof

Let $p_1^*(\hat{x}|x)$ and $p_2^*(\hat{x}|x)$ be the distributions that optimize $R(\mathcal{D}_1)$ and $R(\mathcal{D}_2)$. Let's try to time share between the two distributions. That is, using $p_1^*(\hat{x}|x)$ with λ fraction of time and $p_2^*(\hat{x}|x)$ with $(1-\lambda)$ fraction of time.

Convexity of $R(\mathcal{D})$

Recall that $R(\mathcal{D}) = \min_{p(\hat{x}|x)} I(\hat{X}; X)$ with $E[d(X, \hat{X})] \leq \mathcal{D}$
We want to show that

$$R(\lambda\mathcal{D}_1 + (1-\lambda)\mathcal{D}_2) \leq \lambda R(\mathcal{D}_1) + (1-\lambda)R(\mathcal{D}_2)$$

Proof

Let $p_1^*(\hat{x}|x)$ and $p_2^*(\hat{x}|x)$ be the distributions that optimize $R(\mathcal{D}_1)$ and $R(\mathcal{D}_2)$. Let's try to time share between the two distributions. That is, using $p_1^*(\hat{x}|x)$ with λ fraction of time and $p_2^*(\hat{x}|x)$ with $(1-\lambda)$ fraction of time. The resulting distortion will be $\lambda\mathcal{D}_1 + (1-\lambda)\mathcal{D}_2$.

Convexity of $R(\mathcal{D})$

Recall that $R(\mathcal{D}) = \min_{p(\hat{x}|x)} I(\hat{X};X)$ with $E[d(X,\hat{X})] \leq \mathcal{D}$
We want to show that

$$R(\lambda\mathcal{D}_1 + (1-\lambda)\mathcal{D}_2) \leq \lambda R(\mathcal{D}_1) + (1-\lambda)R(\mathcal{D}_2)$$

Proof

Let $p_1^*(\hat{x}|x)$ and $p_2^*(\hat{x}|x)$ be the distributions that optimize $R(\mathcal{D}_1)$ and $R(\mathcal{D}_2)$. Let's try to time share between the two distributions. That is, using $p_1^*(\hat{x}|x)$ with λ fraction of time and $p_2^*(\hat{x}|x)$ with $(1-\lambda)$ fraction of time. The resulting distortion will be $\lambda\mathcal{D}_1 + (1-\lambda)\mathcal{D}_2$.
Therefore,

$$\lambda R(\mathcal{D}_1) + (1-\lambda)R(\mathcal{D}_2) = \lambda I(\hat{X}_1;X) + (1-\lambda)I(\hat{X}_2;X)$$

Convexity of $R(\mathcal{D})$

Recall that $R(\mathcal{D}) = \min_{p(\hat{x}|x)} I(\hat{X}; X)$ with $E[d(X, \hat{X})] \leq \mathcal{D}$
We want to show that

$$R(\lambda\mathcal{D}_1 + (1-\lambda)\mathcal{D}_2) \leq \lambda R(\mathcal{D}_1) + (1-\lambda)R(\mathcal{D}_2)$$

Proof

Let $p_1^*(\hat{x}|x)$ and $p_2^*(\hat{x}|x)$ be the distributions that optimize $R(\mathcal{D}_1)$ and $R(\mathcal{D}_2)$. Let's try to time share between the two distributions. That is, using $p_1^*(\hat{x}|x)$ with λ fraction of time and $p_2^*(\hat{x}|x)$ with $(1-\lambda)$ fraction of time. The resulting distortion will be $\lambda\mathcal{D}_1 + (1-\lambda)\mathcal{D}_2$. Therefore,

$$\lambda R(\mathcal{D}_1) + (1-\lambda)R(\mathcal{D}_2) = \lambda I(\hat{X}_1; X) + (1-\lambda)I(\hat{X}_2; X)$$
$$=\lambda f(p_1^*(\hat{x}|x)) + (1-\lambda)f(p_2^*(\hat{x}|x))$$

Convexity of $R(\mathcal{D})$

Recall that $R(\mathcal{D}) = \min_{p(\hat{x}|x)} I(\hat{X}; X)$ with $E[d(X, \hat{X})] \leq \mathcal{D}$
We want to show that

$$R(\lambda\mathcal{D}_1 + (1-\lambda)\mathcal{D}_2) \leq \lambda R(\mathcal{D}_1) + (1-\lambda)R(\mathcal{D}_2)$$

Proof

Let $p_1^*(\hat{x}|x)$ and $p_2^*(\hat{x}|x)$ be the distributions that optimize $R(\mathcal{D}_1)$ and $R(\mathcal{D}_2)$. Let's try to time share between the two distributions. That is, using $p_1^*(\hat{x}|x)$ with λ fraction of time and $p_2^*(\hat{x}|x)$ with $(1-\lambda)$ fraction of time. The resulting distortion will be $\lambda\mathcal{D}_1 + (1-\lambda)\mathcal{D}_2$.
Therefore,

$$\begin{aligned}\lambda R(\mathcal{D}_1) + (1-\lambda)R(\mathcal{D}_2) &= \lambda I(\hat{X}_1; X) + (1-\lambda)I(\hat{X}_2; X)\\ =&\lambda f(p_1^*(\hat{x}|x)) + (1-\lambda)f(p_2^*(\hat{x}|x)) \geq f(\lambda p_1^*(\hat{x}|x) + (1-\lambda)p_2^*(\hat{x}|x))\end{aligned}$$

Convexity of $R(\mathcal{D})$

Recall that $R(\mathcal{D}) = \min_{p(\hat{x}|x)} I(\hat{X}; X)$ with $E[d(X, \hat{X})] \leq \mathcal{D}$
We want to show that

$$R(\lambda\mathcal{D}_1 + (1-\lambda)\mathcal{D}_2) \leq \lambda R(\mathcal{D}_1) + (1-\lambda)R(\mathcal{D}_2)$$

Proof

Let $p_1^*(\hat{x}|x)$ and $p_2^*(\hat{x}|x)$ be the distributions that optimize $R(\mathcal{D}_1)$ and $R(\mathcal{D}_2)$. Let's try to time share between the two distributions. That is, using $p_1^*(\hat{x}|x)$ with λ fraction of time and $p_2^*(\hat{x}|x)$ with $(1-\lambda)$ fraction of time. The resulting distortion will be $\lambda\mathcal{D}_1 + (1-\lambda)\mathcal{D}_2$. Therefore,

$$\begin{aligned} &\lambda R(\mathcal{D}_1) + (1-\lambda)R(\mathcal{D}_2) = \lambda I(\hat{X}_1; X) + (1-\lambda)I(\hat{X}_2; X) \\ =&\lambda f(p_1^*(\hat{x}|x)) + (1-\lambda)f(p_2^*(\hat{x}|x)) \geq f(\lambda p_1^*(\hat{x}|x) + (1-\lambda)p_2^*(\hat{x}|x)) \\ =&I(\tilde{X}; X) \end{aligned}$$

Convexity of $R(\mathcal{D})$

Recall that $R(\mathcal{D}) = \min_{p(\hat{x}|x)} I(\hat{X};X)$ with $E[d(X,\hat{X})] \le \mathcal{D}$
We want to show that

$$R(\lambda\mathcal{D}_1 + (1-\lambda)\mathcal{D}_2) \le \lambda R(\mathcal{D}_1) + (1-\lambda)R(\mathcal{D}_2)$$

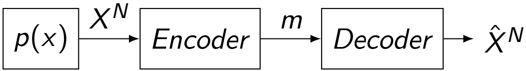
Proof

Let $p_1^*(\hat{x}|x)$ and $p_2^*(\hat{x}|x)$ be the distributions that optimize $R(\mathcal{D}_1)$ and $R(\mathcal{D}_2)$. Let's try to time share between the two distributions. That is, using $p_1^*(\hat{x}|x)$ with λ fraction of time and $p_2^*(\hat{x}|x)$ with $(1-\lambda)$ fraction of time. The resulting distortion will be $\lambda\mathcal{D}_1 + (1-\lambda)\mathcal{D}_2$. Therefore,

$$\begin{aligned}\lambda R(\mathcal{D}_1) + (1-\lambda)R(\mathcal{D}_2) &= \lambda I(\hat{X}_1;X) + (1-\lambda)I(\hat{X}_2;X)\\ =&\lambda f(p_1^*(\hat{x}|x)) + (1-\lambda)f(p_2^*(\hat{x}|x)) \ge f(\lambda p_1^*(\hat{x}|x) + (1-\lambda)p_2^*(\hat{x}|x))\\ =&I(\tilde{X};X) \ge R(\lambda\mathcal{D}_1 + (1-\lambda)\mathcal{D}_2),\end{aligned}$$

where $\tilde{X} = \begin{cases}\hat{X}_1 & \text{with } \lambda \text{ fraction of time}\\ \hat{X}_2 & \text{with } (1-\lambda) \text{ fraction of time}\end{cases}$

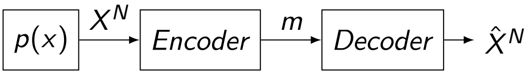
Converse proof



$$p(x) \xrightarrow{X^N} \boxed{Encoder} \xrightarrow{m} \boxed{Decoder} \rightarrow \hat{X}^N$$

$H(M)$

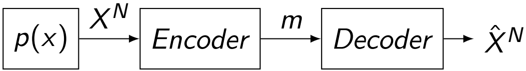
Converse proof



$p(x)$ X^N *Encoder* m *Decoder* $\hat{X}^N$

$$H(M) \geq H(M) - H(M|X^N) = I(M; X^N)$$

Converse proof



$p(x)$ X^N *Encoder* m *Decoder* $\hat{X}^N$

$$H(M) \geq H(M) - H(M|X^N) = I(M;X^N) \geq I(\hat{X}^N;X^N)$$

$$= H(X^N) - H(X^N|\hat{X}^N)$$

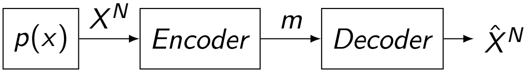
Converse proof

$$\boxed{p(x)} \xrightarrow{X^N} \boxed{Encoder} \xrightarrow{m} \boxed{Decoder} \rightarrow \hat{X}^N$$

$$H(M) \geq H(M) - H(M|X^N) = I(M;X^N) \geq I(\hat{X}^N;X^N)$$

$$= H(X^N) - H(X^N|\hat{X}^N) = \sum_{i=1}^{N} H(X_i) - \sum_{i=1}^{N} H(X_i|\hat{X}^N, X^{i-1})$$

Converse proof



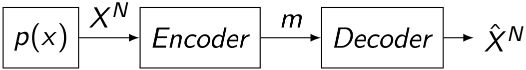
$p(x)$ $\xrightarrow{X^N}$ *Encoder* $\xrightarrow{m}$ *Decoder* $\rightarrow \hat{X}^N$

$$H(M) \geq H(M) - H(M|X^N) = I(M;X^N) \geq I(\hat{X}^N;X^N)$$

$$= H(X^N) - H(X^N|\hat{X}^N) = \sum_{i=1}^N H(X_i) - \sum_{i=1}^N H(X_i|\hat{X}^N, X^{i-1})$$

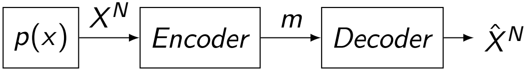
$$\geq \sum_{i=1}^N H(X_i) - \sum_{i=1}^N H(X_i|\hat{X}_i) = \sum_{i=1}^N I(X_i;\hat{X}_i)$$

Converse proof



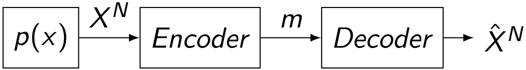
$$
\begin{aligned}
H(M) &\geq H(M) - H(M|X^N) = I(M;X^N) \geq I(\hat{X}^N;X^N) \\
&= H(X^N) - H(X^N|\hat{X}^N) = \sum_{i=1}^N H(X_i) - \sum_{i=1}^N H(X_i|\hat{X}^N, X^{i-1}) \\
&\geq \sum_{i=1}^N H(X_i) - \sum_{i=1}^N H(X_i|\hat{X}_i) = \sum_{i=1}^N I(X_i;\hat{X}_i) \\
&\geq \sum_{i=1}^N R(E[d(X_i,\hat{X}_i)]) = N\left(\frac{1}{N}\sum_{i=1}^N R(E[d(X_i;\hat{X}_i)])\right)
\end{aligned}
$$

Converse proof



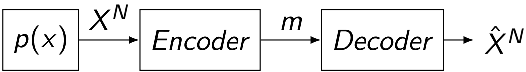
$$
\begin{aligned}
H(M) &\geq H(M) - H(M|X^N) = I(M;X^N) \geq I(\hat{X}^N;X^N) \\
&= H(X^N) - H(X^N|\hat{X}^N) = \sum_{i=1}^{N} H(X_i) - \sum_{i=1}^{N} H(X_i|\hat{X}^N, X^{i-1}) \\
&\geq \sum_{i=1}^{N} H(X_i) - \sum_{i=1}^{N} H(X_i|\hat{X}_i) = \sum_{i=1}^{N} I(X_i;\hat{X}_i) \\
&\geq \sum_{i=1}^{N} R(E[d(X_i,\hat{X}_i)]) = N\left(\frac{1}{N}\sum_{i=1}^{N} R(E[d(X_i;\hat{X}_i)])\right) \\
&\geq NR\left(\frac{1}{N}\sum_{i=1}^{N} E[d(X_i;\hat{X}_i)]\right)
\end{aligned}
$$

Converse proof



$$
\begin{aligned}
H(M) &\geq H(M) - H(M|X^N) = I(M;X^N) \geq I(\hat{X}^N;X^N) \\
&= H(X^N) - H(X^N|\hat{X}^N) = \sum_{i=1}^N H(X_i) - \sum_{i=1}^N H(X_i|\hat{X}^N, X^{i-1}) \\
&\geq \sum_{i=1}^N H(X_i) - \sum_{i=1}^N H(X_i|\hat{X}_i) = \sum_{i=1}^N I(X_i;\hat{X}_i) \\
&\geq \sum_{i=1}^N R(E[d(X_i,\hat{X}_i)]) = N\left(\frac{1}{N}\sum_{i=1}^N R(E[d(X_i;\hat{X}_i)])\right) \\
&\geq NR\left(\frac{1}{N}\sum_{i=1}^N E[d(X_i;\hat{X}_i)]\right) = NR\left(E\left[\frac{1}{N}\sum_{i=1}^N d(X_i;\hat{X}_i)\right]\right)
\end{aligned}
$$

Converse proof



$$
\begin{aligned}
H(M) &\geq H(M) - H(M|X^N) = I(M;X^N) \geq I(\hat{X}^N;X^N) \\
&= H(X^N) - H(X^N|\hat{X}^N) = \sum_{i=1}^{N} H(X_i) - \sum_{i=1}^{N} H(X_i|\hat{X}^N, X^{i-1}) \\
&\geq \sum_{i=1}^{N} H(X_i) - \sum_{i=1}^{N} H(X_i|\hat{X}_i) = \sum_{i=1}^{N} I(X_i;\hat{X}_i) \\
&\geq \sum_{i=1}^{N} R(E[d(X_i,\hat{X}_i)]) = N\left(\frac{1}{N}\sum_{i=1}^{N} R(E[d(X_i;\hat{X}_i)])\right) \\
&\geq NR\left(\frac{1}{N}\sum_{i=1}^{N} E[d(X_i;\hat{X}_i)]\right) = NR\left(E\left[\frac{1}{N}\sum_{i=1}^{N} d(X_i;\hat{X}_i)\right]\right) \\
&= NR(E[d(X^N;\hat{X}^N)]) \geq NR(D)
\end{aligned}
$$

More inequalities

Lemma (Anup Rao, CSE 533, Lecture 2, Lemma 3)

If $k \leq n/2$, *then* $\sum_{i=0}^{k} \binom{n}{i} \leq 2^{nH(k/n)}$

Proof.

Consider length-n binary sequence $X_1, X_2, \cdots, X_n$ uniformly sampled from a set of binary sequences with at most k 1's. Since there are $\sum_{i=0}^{k} \binom{n}{i}$ so many sequences, $H(X_1, X_2, \cdots, X_n) = \log \sum_{i=0}^{k} \binom{n}{i}$. On the other hand, $H(X_1, X_2, \cdots, X_n) \leq \sum_{i=1}^{n} H(X_i) = nH(k/n)$. Raise both sides with the power of two and we get the proof ☐

Example

Say we have 2^n people watching a subset of $2n$ movies. Each of them have at least watch 90% of all movies. At least two people actually watch the same set

Proof.

Let's count how many different subsets a person can watch, which is

$$\sum_{i=0.9(2n)}^{2n} \binom{2n}{i} = \sum_{i=0}^{0.1(2n)} \binom{2n}{i} \leq 2^{2nH(0.1)} < 2^n$$

since $H(0.1) = 0.469 < 0.5$.
As we have 2^n people, by pigeon hole principle, there must be at least a pair who watched the same set ☐