

# CS 2750 Machine Learning

## Lecture 5

### Density estimation

Milos Hauskrecht  
[milos@cs.pitt.edu](mailto:milos@cs.pitt.edu)  
5329 Sennott Square

---

CS 2750 Machine Learning

### Outline

**Outline:**

- **Density estimation:** ✓
  - Maximum likelihood (ML)
  - Bayesian parameter estimates
  - MAP
- **Bernoulli distribution.** ✓
- **Binomial distribution** ✓
- **Multinomial distribution**
- **Normal distribution**
- **Exponential family**

---

CS 2750 Machine Learning

## Parametric density estimation

### Parametric density estimation:

- A set of random variables  $\mathbf{X} = \{X_1, X_2, \dots, X_d\}$
- **A model of the distribution** over variables in  $\mathbf{X}$   
with **parameters**  $\Theta : \hat{p}(\mathbf{X} | \Theta)$
- **Data**  $D = \{D_1, D_2, \dots, D_n\}$

**Objective:** find parameters  $\Theta$  such that  $p(\mathbf{X} | \Theta)$  describes data  $D$  the best

---

CS 2750 Machine Learning

## Parameter estimation (learning)

- **Maximum likelihood (ML)**  
 $\Theta_{ML} = \arg \max_{\Theta} p(D | \Theta, \xi)$
- **Bayesian parameter estimation**  
keep the **posterior density**  $p(\Theta | D, \xi)$
- **Maximum a posteriori probability (MAP)**  
 $\Theta_{MAP} = \arg \max_{\Theta} p(\Theta | D, \xi)$
- **Expected value**

$$\Theta_{EXP} = \int_{\Theta} \Theta p(\Theta | D, \xi) d\Theta$$

---

CS 2750 Machine Learning

## Multinomial distribution

**Example: Multi-way coin toss, roll of dice**

• **Data:** a set of  $N$  outcomes (multi-set)

$N_i$  - a number of times an outcome  $i$  has been seen

**Model parameters:**  $\boldsymbol{\theta} = (\theta_1, \theta_2, \dots, \theta_k)$  s.t.  $\sum_{i=1}^k \theta_i = 1$   
 $\theta_i$  - probability of an outcome  $i$

**Probability of data** (likelihood)

$$P(N_1, N_2, \dots, N_k | \boldsymbol{\theta}, \xi) = \frac{N!}{N_1! N_2! \dots N_k!} \theta_1^{N_1} \theta_2^{N_2} \dots \theta_k^{N_k} \quad \text{Multinomial distribution}$$

**ML estimate:**

$$\theta_{i,ML} = \frac{N_i}{N}$$

CS 2750 Machine Learning

## Posterior density and MAP estimate

**Choice of the prior: Dirichlet distribution**

$$Dir(\boldsymbol{\theta} | \alpha_1, \dots, \alpha_k) = \frac{\Gamma(\sum_{i=1}^k \alpha_i)}{\prod_{i=1}^k \Gamma(\alpha_i)} \theta_1^{\alpha_1-1} \theta_2^{\alpha_2-1} \dots \theta_k^{\alpha_k-1}$$

**Dirichlet is the conjugate choice for multinomial**

$$P(D | \boldsymbol{\theta}, \xi) = P(N_1, N_2, \dots, N_k | \boldsymbol{\theta}, \xi) = \frac{N!}{N_1! N_2! \dots N_k!} \theta_1^{N_1} \theta_2^{N_2} \dots \theta_k^{N_k}$$

**Posterior density**

$$p(\boldsymbol{\theta} | D, \xi) = \frac{P(D | \boldsymbol{\theta}, \xi) Dir(\boldsymbol{\theta} | \alpha_1, \alpha_2, \dots, \alpha_k)}{P(D | \xi)} = Dir(\boldsymbol{\theta} | \alpha_1 + N_1, \dots, \alpha_k + N_k)$$

**MAP estimate:**

$$\theta_{i,MAP} = \frac{\alpha_i + N_i - 1}{\sum_{i=1 \dots k} (\alpha_i + N_i) - k}$$

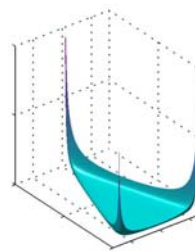
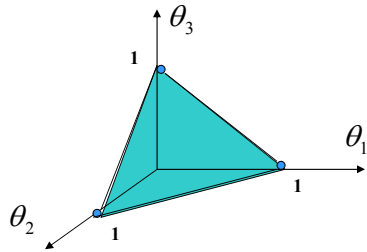
CS 2750 Machine Learning

## Dirichlet distribution

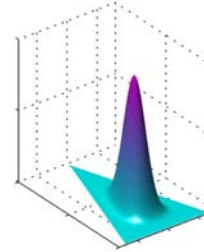
**Dirichlet distribution:**

$$Dir(\boldsymbol{\theta} | \alpha_1, \dots, \alpha_k) = \frac{\Gamma(\sum_{i=1}^k \alpha_i)}{\prod_{i=1}^k \Gamma(\alpha_i)} \theta_1^{\alpha_1-1} \theta_2^{\alpha_2-1} \dots \theta_k^{\alpha_k-1}$$

**Assume: k=3**



$\alpha_k = 10^{-1}$



$\alpha_k = 10^1$

CS 2750 Machine Learning

## Expected value

**The result is analogous to the result for binomial**

$$E(\boldsymbol{\theta}) = \int_{0 \leq \theta_i \leq 1, \sum \theta_i = 1} \boldsymbol{\theta} Dir(\boldsymbol{\theta} | \boldsymbol{\eta}) d\boldsymbol{\theta} = \left( \frac{\eta_1}{\eta_1 + \eta_2 + \eta_k}, \dots, \frac{\eta_i}{\eta_1 + \eta_2 + \eta_k}, \dots, \frac{\eta_k}{\eta_1 + \eta_2 + \eta_k} \right)$$

**Expectation based parameter estimate**

$$E(\boldsymbol{\theta}) = \left( \frac{\alpha_1 + N_1}{\alpha_1 + N_1 + \dots + \alpha_k + N_k}, \dots, \frac{\alpha_i + N_i}{\alpha_1 + N_1 + \dots + \alpha_k + N_k}, \dots, \frac{\alpha_k + N_k}{\alpha_1 + N_1 + \dots + \alpha_k + N_k} \right)$$

**Represents the predictive probability** of an event  $x=i$

$$P(x=i | \boldsymbol{\theta}, \xi) = \frac{\alpha_i + N_i}{\alpha_1 + N_1 + \dots + \alpha_k + N_k}$$

CS 2750 Machine Learning

## Other distributions

### The same ideas can be applied to other distributions

- Typically we choose distributions that behave well so that computations lead to a nice solutions

- **Exponential family of distributions**

**Conjugate choices** for some of the distributions from the exponential family:

- **Binomial – Beta**
- **Multinomial - Dirichlet**
- **Exponential – Gamma**
- **Poisson – Inverse Gamma**
- **Gaussian - Gaussian (mean) and Wishart (covariance)**

---

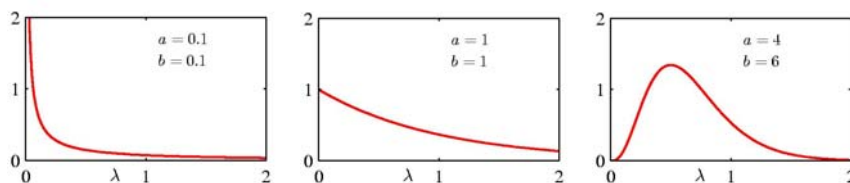
CS 2750 Machine Learning

## Gamma distribution

- **Gamma distribution**

$$\text{Gam}(\lambda|a, b) = \frac{1}{\Gamma(a)} b^a \lambda^{a-1} \exp(-b\lambda)$$

$$\mathbb{E}[\lambda] = \frac{a}{b} \quad \text{var}[\lambda] = \frac{a}{b^2}$$



---

CS 2750 Machine Learning

## Other distributions

### Exponential distribution:

- A special case of Gamma for  $a=1$

$$p(x | b) = \left(\frac{1}{b}\right) e^{-\frac{x}{b}}$$

### Poisson distribution:

$$p(x | \lambda) = \frac{e^{-\lambda} \lambda^x}{x!} \quad \text{for } x \in \{0, 1, 2, \dots\}$$

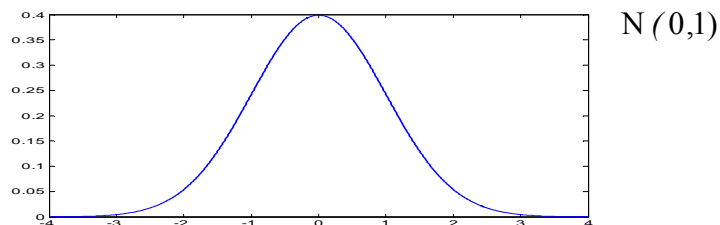
CS 2750 Machine Learning

## Gaussian (normal) distribution

- **Gaussian:**  $x \sim N(\mu, \sigma)$
- **Parameters:**  $\mu$  - mean  
 $\sigma$  - standard deviation
- **Density function:**

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

- **Example:**



CS 2750 Machine Learning

## Parameter estimates

- **Loglikelihood**  $l(D, \mu, \sigma) = \log \prod_{i=1}^n p(x_i | \mu, \sigma)$

- **ML estimates of the mean and variance:**

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

- ML variance estimate is biased

$$E_n(\hat{\sigma}^2) = E_n\left(\frac{1}{n} \sum_{i=1}^n (x_i - \hat{\mu})^2\right) = \frac{n-1}{n} \sigma^2 \neq \sigma^2$$

- **Unbiased estimate:**

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

CS 2750 Machine Learning

## Multivariate normal distribution

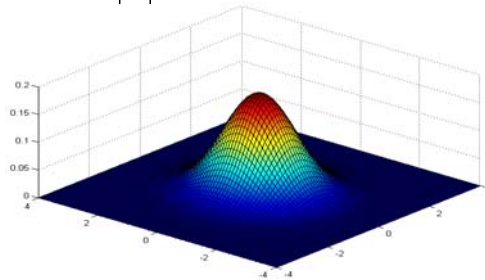
- **Multivariate normal:**  $\mathbf{x} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

- **Parameters:**  $\boldsymbol{\mu}$  - mean  
 $\boldsymbol{\Sigma}$  - covariance matrix

- **Density function:**

$$p(\mathbf{x} | \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right]$$

- **Example:**



CS 2750 Machine Learning

## Partitioned Gaussian Distributions

- **Multivariate Gaussian:**

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

– what are marginals and conditionals?

- **Example:**

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_a \\ \mathbf{x}_b \end{pmatrix} \quad \boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_a \\ \boldsymbol{\mu}_b \end{pmatrix} \quad \boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{aa} & \boldsymbol{\Sigma}_{ab} \\ \boldsymbol{\Sigma}_{ba} & \boldsymbol{\Sigma}_{bb} \end{pmatrix}$$

$$\boldsymbol{\Lambda} \equiv \boldsymbol{\Sigma}^{-1} \quad \boldsymbol{\Lambda} = \begin{pmatrix} \boldsymbol{\Lambda}_{aa} & \boldsymbol{\Lambda}_{ab} \\ \boldsymbol{\Lambda}_{ba} & \boldsymbol{\Lambda}_{bb} \end{pmatrix}$$

**Precision matrix**

CS 2750 Machine Learning

## Partitioned Conditionals and Marginals

- **Conditional density:**

$$p(\mathbf{x}_a|\mathbf{x}_b) = \mathcal{N}(\mathbf{x}_a|\boldsymbol{\mu}_{a|b}, \boldsymbol{\Sigma}_{a|b})$$

$$\begin{aligned} \boldsymbol{\Sigma}_{a|b} &= \boldsymbol{\Lambda}_{aa}^{-1} = \boldsymbol{\Sigma}_{aa} - \boldsymbol{\Sigma}_{ab}\boldsymbol{\Sigma}_{bb}^{-1}\boldsymbol{\Sigma}_{ba} \\ \boldsymbol{\mu}_{a|b} &= \boldsymbol{\Sigma}_{a|b} \{ \boldsymbol{\Lambda}_{aa}\boldsymbol{\mu}_a - \boldsymbol{\Lambda}_{ab}(\mathbf{x}_b - \boldsymbol{\mu}_b) \} \\ &= \boldsymbol{\mu}_a - \boldsymbol{\Lambda}_{aa}^{-1}\boldsymbol{\Lambda}_{ab}(\mathbf{x}_b - \boldsymbol{\mu}_b) \\ &= \boldsymbol{\mu}_a + \boldsymbol{\Sigma}_{ab}\boldsymbol{\Sigma}_{bb}^{-1}(\mathbf{x}_b - \boldsymbol{\mu}_b) \end{aligned}$$

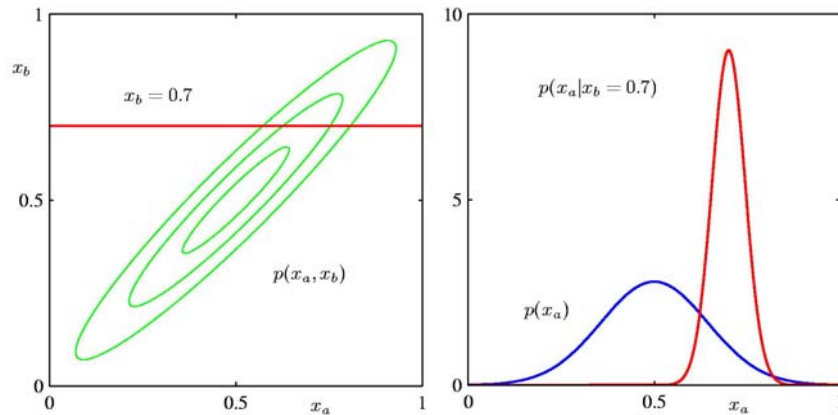
- **Marginal Density:**

$$\begin{aligned} p(\mathbf{x}_a) &= \int p(\mathbf{x}_a, \mathbf{x}_b) d\mathbf{x}_b \\ &= \mathcal{N}(\mathbf{x}_a|\boldsymbol{\mu}_a, \boldsymbol{\Sigma}_{aa}) \end{aligned}$$

CS 2750 Machine Learning



## Partitioned Conditionals and Marginals



CS 2750 Machine Learning

## Parameter estimates

- **Loglikelihood**  $l(D, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \log \prod_{i=1}^n p(\mathbf{x}_i | \boldsymbol{\mu}, \boldsymbol{\Sigma})$

- **ML estimates of the mean and covariances:**

$$\hat{\boldsymbol{\mu}} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i \quad \hat{\boldsymbol{\Sigma}} = \frac{1}{n} \sum_{i=1}^n (\mathbf{x}_i - \hat{\boldsymbol{\mu}})(\mathbf{x}_i - \hat{\boldsymbol{\mu}})^T$$

– Covariance estimate is biased

$$E_n(\hat{\boldsymbol{\Sigma}}) = E_n \left( \frac{1}{n} \sum_{i=1}^n (\mathbf{x}_i - \hat{\boldsymbol{\mu}})(\mathbf{x}_i - \hat{\boldsymbol{\mu}})^T \right) = \frac{n-1}{n} \boldsymbol{\Sigma} \neq \boldsymbol{\Sigma}$$

- **Unbiased estimate:**

$$\hat{\boldsymbol{\Sigma}} = \frac{1}{n-1} \sum_{i=1}^n (\mathbf{x}_i - \hat{\boldsymbol{\mu}})(\mathbf{x}_i - \hat{\boldsymbol{\mu}})^T$$

CS 2750 Machine Learning

## Posterior of a multivariate normal

- Assume a prior on the mean  $\boldsymbol{\mu}$  that is normally distributed:

$$p(\boldsymbol{\mu}) \approx N(\boldsymbol{\mu}_p, \boldsymbol{\Sigma}_p)$$

- Then the posterior of  $\boldsymbol{\mu}$  is normally distributed

$$\begin{aligned} p(\boldsymbol{\mu} | D) &\approx \left( \prod_{i=1}^n \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}|^{1/2}} \exp \left[ -\frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}) \right] \right) \\ &\quad * \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}_p|^{1/2}} \exp \left[ -\frac{1}{2} (\boldsymbol{\mu} - \boldsymbol{\mu}_p)^T \boldsymbol{\Sigma}_p^{-1} (\boldsymbol{\mu} - \boldsymbol{\mu}_p) \right] \\ &= \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}_n|^{1/2}} \exp \left[ -\frac{1}{2} (\boldsymbol{\mu} - \boldsymbol{\mu}_n)^T \boldsymbol{\Sigma}_n^{-1} (\boldsymbol{\mu} - \boldsymbol{\mu}_n) \right] \end{aligned}$$

CS 2750 Machine Learning

## Posterior of a multivariate normal

- Then the posterior of  $\boldsymbol{\mu}$  is normally distributed

$$p(\boldsymbol{\mu} | D) = \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}_n|^{1/2}} \exp \left[ -\frac{1}{2} (\boldsymbol{\mu} - \boldsymbol{\mu}_n)^T \boldsymbol{\Sigma}_n^{-1} (\boldsymbol{\mu} - \boldsymbol{\mu}_n) \right]$$

$$\boldsymbol{\Sigma}_n^{-1} = n\boldsymbol{\Sigma}^{-1} + \boldsymbol{\Sigma}_p^{-1}$$

$$\boldsymbol{\mu}_n = \boldsymbol{\Sigma}_p \left( \boldsymbol{\Sigma}_p + \frac{1}{n} \boldsymbol{\Sigma} \right)^{-1} \left( \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i \right) + \frac{1}{n} \boldsymbol{\Sigma} \left( \boldsymbol{\Sigma}_p + \frac{1}{n} \boldsymbol{\Sigma} \right)^{-1} \boldsymbol{\mu}_p$$

$$\boldsymbol{\Sigma}_n = \boldsymbol{\Sigma}_p \left( \boldsymbol{\Sigma}_p + \frac{1}{n} \boldsymbol{\Sigma} \right)^{-1} \frac{1}{n} \boldsymbol{\Sigma}$$

CS 2750 Machine Learning

## Sequential Bayesian parameter estimation

- **Sequential Bayesian approach**

- Under the iid the estimates of the posterior can be computed incrementally for a sequence of data points

$$p(\Theta | D, \xi) = \frac{p(D | \Theta, \xi) p(\Theta | \xi)}{\int_{\Theta} p(D | \Theta, \xi) p(\Theta | \xi) d\Theta}$$

- If we use a conjugate prior we get back the same posterior
- Assume we split the data D in the last element  $\mathbf{x}$  and the rest  $p(D | \Theta) = P(x | \Theta) P(D_{n-1} | \Theta)$

- **Then:** A “new” prior

$$p(\Theta | D, \xi) = \frac{P(x | \Theta) \overbrace{P(D_{n-1} | \Theta) p(\Theta | \xi)}^{\text{A “new” prior}}}{\int_{\Theta} P(x | \Theta) P(D_{n-1} | \Theta) p(\Theta | \xi) d\Theta}$$

---

CS 2750 Machine Learning

## Exponential family

### Exponential family:

- all probability mass / density functions that can be written in the exponential normal form

$$f(\mathbf{x} | \boldsymbol{\eta}) = \frac{1}{Z(\boldsymbol{\eta})} h(\mathbf{x}) \exp[\boldsymbol{\eta}^T t(\mathbf{x})]$$

- $\boldsymbol{\eta}$  a vector of **natural (or canonical) parameters**
- $t(\mathbf{x})$  a function referred to as a **sufficient statistic**
- $h(\mathbf{x})$  a function of  $\mathbf{x}$  (it is less important)
- $Z(\boldsymbol{\eta})$  a normalization constant (a **partition function**)

$$Z(\boldsymbol{\eta}) = \int h(\mathbf{x}) \exp\{\boldsymbol{\eta}^T t(\mathbf{x})\} d\mathbf{x}$$

- Other common form:

$$f(\mathbf{x} | \boldsymbol{\eta}) = h(\mathbf{x}) \exp[\boldsymbol{\eta}^T t(\mathbf{x}) - A(\boldsymbol{\eta})] \quad \log Z(\boldsymbol{\eta}) = A(\boldsymbol{\eta})$$

---

CS 2750 Machine Learning

## Exponential family: examples

- **Bernoulli distribution**

$$\begin{aligned} p(x | \pi) &= \pi^x (1 - \pi)^{1-x} \\ &= \exp \left\{ \log \left( \frac{\pi}{1 - \pi} \right) x + \log(1 - \pi) \right\} \\ &= \exp \{ \log(1 - \pi) \} \exp \left\{ \log \left( \frac{\pi}{1 - \pi} \right) x \right\} \end{aligned}$$

- **Exponential family**

$$f(\mathbf{x} | \boldsymbol{\eta}) = \frac{1}{Z(\boldsymbol{\eta})} h(\mathbf{x}) \exp [\boldsymbol{\eta}^T t(\mathbf{x})]$$

- **Parameters**

$$\begin{aligned} \boldsymbol{\eta} &= ? & t(\mathbf{x}) &= ? \\ Z(\boldsymbol{\eta}) &= ? & h(\mathbf{x}) &= ? \end{aligned}$$

---

CS 2750 Machine Learning

## Exponential family: examples

- **Bernoulli distribution**

$$\begin{aligned} p(x | \pi) &= \pi^x (1 - \pi)^{1-x} \\ &= \exp \left\{ \log \left( \frac{\pi}{1 - \pi} \right) x + \log(1 - \pi) \right\} \\ &= \exp \{ \log(1 - \pi) \} \exp \left\{ \log \left( \frac{\pi}{1 - \pi} \right) x \right\} \end{aligned}$$

- **Exponential family**

$$f(\mathbf{x} | \boldsymbol{\eta}) = \frac{1}{Z(\boldsymbol{\eta})} h(\mathbf{x}) \exp [\boldsymbol{\eta}^T t(\mathbf{x})]$$

- **Parameters**

$$\begin{aligned} \boldsymbol{\eta} &= \log \frac{\pi}{1 - \pi} \quad (\text{note } \pi = \frac{1}{1 + e^{-\eta}}) & t(\mathbf{x}) &= x \\ Z(\boldsymbol{\eta}) &= \frac{1}{1 - \pi} = 1 + e^\eta & h(\mathbf{x}) &= 1 \end{aligned}$$

---

CS 2750 Machine Learning

## Exponential family: examples

- **Univariate Gaussian distribution**

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

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

- **Exponential family**

$$f(\mathbf{x} | \boldsymbol{\eta}) = \frac{1}{Z(\boldsymbol{\eta})} h(x) \exp[\boldsymbol{\eta}^T t(x)]$$

- **Parameters**

$$\boldsymbol{\eta} = ? \qquad t(\mathbf{x}) = ?$$

$$Z(\boldsymbol{\eta}) = ? \qquad h(\mathbf{x}) = ?$$

CS 2750 Machine Learning

## Exponential family: examples

- **Univariate Gaussian distribution**

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

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

- **Exponential family**

$$f(\mathbf{x} | \boldsymbol{\eta}) = \frac{1}{Z(\boldsymbol{\eta})} h(x) \exp[\boldsymbol{\eta}^T t(x)]$$

- **Parameters**

$$\boldsymbol{\eta} = \begin{bmatrix} \mu / 2\sigma^2 \\ -1 / 2\sigma^2 \end{bmatrix} \qquad t(\mathbf{x}) = \begin{bmatrix} x \\ x^2 \end{bmatrix}$$

$$Z(\boldsymbol{\eta}) = \exp\left\{\frac{\mu}{2\sigma^2} + \log \sigma\right\} = \exp\left\{-\frac{\eta_1^2}{4\eta_2} - \frac{1}{2} \log(-2\eta_2)\right\}$$

$$h(\mathbf{x}) = 1/\sqrt{2\pi}$$

CS 2750 Machine Learning

## Exponential family

- For iid samples, the likelihood of data is

$$\begin{aligned} P(D | \boldsymbol{\eta}) &= \prod_{i=1}^n p(\mathbf{x}_i | \boldsymbol{\eta}) = \prod_{i=1}^n h(\mathbf{x}_i) \exp[\boldsymbol{\eta}^T t(\mathbf{x}_i) - A(\boldsymbol{\eta})] \\ &= \left[ \prod_{i=1}^n h(\mathbf{x}_i) \right] \exp \left[ \sum_{i=1}^n \boldsymbol{\eta}^T t(\mathbf{x}_i) - nA(\boldsymbol{\eta}) \right] \\ &= \left[ \prod_{i=1}^n h(\mathbf{x}_i) \right] \exp \left[ \boldsymbol{\eta}^T \left( \sum_{i=1}^n t(\mathbf{x}_i) \right) - nA(\boldsymbol{\eta}) \right] \end{aligned}$$

- **Important:**

- the dimensionality of the sufficient statistic remains the same for different sample sizes (that is, different number of examples in D)

CS 2750 Machine Learning

## Exponential family

- The log likelihood of data is

$$\begin{aligned} l(D, \boldsymbol{\eta}) &= \log \left[ \prod_{i=1}^n h(\mathbf{x}_i) \right] \exp \left[ \boldsymbol{\eta}^T \left( \sum_{i=1}^n t(\mathbf{x}_i) \right) - nA(\boldsymbol{\eta}) \right] \\ &= \log \left[ \prod_{i=1}^n h(\mathbf{x}_i) \right] + \left[ \boldsymbol{\eta}^T \left( \sum_{i=1}^n t(\mathbf{x}_i) \right) - nA(\boldsymbol{\eta}) \right] \end{aligned}$$

- Optimizing the loglikelihood

$$\nabla_{\boldsymbol{\eta}} l(D, \boldsymbol{\eta}) = \left( \sum_{i=1}^n t(\mathbf{x}_i) \right) - n \nabla_{\boldsymbol{\eta}} A(\boldsymbol{\eta}) = \mathbf{0}$$

- For the ML estimate it must hold

$$\nabla_{\boldsymbol{\eta}} A(\boldsymbol{\eta}) = \frac{1}{n} \left( \sum_{i=1}^n t(\mathbf{x}_i) \right)$$

CS 2750 Machine Learning

## Exponential family

- **Rewriting the gradient:**

$$\nabla_{\boldsymbol{\eta}} A(\boldsymbol{\eta}) = \nabla_{\boldsymbol{\eta}} \log Z(\boldsymbol{\eta}) = \nabla_{\boldsymbol{\eta}} \log \int h(\mathbf{x}) \exp \{ \boldsymbol{\eta}^T t(\mathbf{x}) \} d\mathbf{x}$$

$$\nabla_{\boldsymbol{\eta}} A(\boldsymbol{\eta}) = \frac{\int t(\mathbf{x}) h(\mathbf{x}) \exp \{ \boldsymbol{\eta}^T t(\mathbf{x}) \} d\mathbf{x}}{\int h(\mathbf{x}) \exp \{ \boldsymbol{\eta}^T t(\mathbf{x}) \} d\mathbf{x}}$$

$$\nabla_{\boldsymbol{\eta}} A(\boldsymbol{\eta}) = \int t(\mathbf{x}) h(\mathbf{x}) \exp \{ \boldsymbol{\eta}^T t(\mathbf{x}) - A(\boldsymbol{\eta}) \} d\mathbf{x}$$

$$\nabla_{\boldsymbol{\eta}} A(\boldsymbol{\eta}) = E(t(\mathbf{x}))$$

- **Result:** 
$$E(t(\mathbf{x})) = \frac{1}{n} \left( \sum_{i=1}^n t(\mathbf{x}_i) \right)$$

- **For the ML estimate, the parameters  $\boldsymbol{\eta}$  should be adjusted such that the expectation of the statistic  $t(\mathbf{x})$  is equal to the observed sample statistics**

CS 2750 Machine Learning

## Moments of the distribution

- **For the exponential family**

- The k-th moment of the statistic corresponds to the k-th derivative of  $A(\boldsymbol{\eta})$
- If  $x$  is a component of  $t(\mathbf{x})$  then we get the moments of the distribution by differentiating its corresponding natural parameter

- **Example: Bernoulli**  $p(x | \pi) = \exp \left\{ \log \left( \frac{\pi}{1 - \pi} \right) x + \log(1 - \pi) \right\}$

$$A(\boldsymbol{\eta}) = \log \frac{1}{1 - \pi} = \log(1 + e^{\boldsymbol{\eta}})$$

- **Derivatives:**

$$\frac{\partial A(\boldsymbol{\eta})}{\partial \boldsymbol{\eta}} = \frac{\partial}{\partial \boldsymbol{\eta}} \log(1 + e^{\boldsymbol{\eta}}) = \frac{e^{\boldsymbol{\eta}}}{(1 + e^{\boldsymbol{\eta}})} = \frac{1}{(1 + e^{-\boldsymbol{\eta}})} = \pi$$

$$\frac{\partial A(\boldsymbol{\eta})}{\partial \boldsymbol{\eta}^2} = \frac{\partial}{\partial \boldsymbol{\eta}} \frac{1}{(1 + e^{-\boldsymbol{\eta}})} = \pi(1 - \pi)$$

CS 2750 Machine Learning

## Conjugate priors

For any member of the exponential family

$$f(\mathbf{x} | \boldsymbol{\eta}) = \frac{1}{Z(\boldsymbol{\eta})} h(\mathbf{x}) \exp[\boldsymbol{\eta}^T \mathbf{t}(\mathbf{x})]$$

there exists a prior:

$$p(\boldsymbol{\eta} | \boldsymbol{\chi}, \nu) = u(\boldsymbol{\chi}, \nu) g(\boldsymbol{\eta})^\nu \exp[\nu \boldsymbol{\eta}^T \boldsymbol{\chi}]$$

Such that for  $n$  examples, the posterior is

$$p(\boldsymbol{\eta} | D, \boldsymbol{\chi}, \nu) \propto g(\boldsymbol{\eta})^{\nu+n} \exp\left[\boldsymbol{\eta}^T \left(\left[\sum_{i=1}^n \mathbf{t}(x_i)\right] + \nu \boldsymbol{\chi}\right)\right]$$

Note that:

$$P(D | \boldsymbol{\eta}) = \left(\frac{1}{Z(\boldsymbol{\eta})}\right)^n \left[\prod_{i=1}^n h(\mathbf{x}_i)\right] \exp\left[\boldsymbol{\eta}^T \left(\sum_{i=1}^n \mathbf{t}(\mathbf{x}_i)\right)\right]$$

CS 2750 Machine Learning

## Conjugate priors

For any member of the exponential family

$$f(\mathbf{x} | \boldsymbol{\eta}) = \frac{1}{Z(\boldsymbol{\eta})} h(\mathbf{x}) \exp[\boldsymbol{\eta}^T \mathbf{t}(\mathbf{x})]$$

there exists a prior:

$$p(\boldsymbol{\eta} | \boldsymbol{\chi}, \nu) = u(\boldsymbol{\chi}, \nu) g(\boldsymbol{\eta})^\nu \exp[\nu \boldsymbol{\eta}^T \boldsymbol{\chi}]$$

Such that for  $n$  examples, the posterior is

$$p(\boldsymbol{\eta} | D, \boldsymbol{\chi}, \nu) \propto g(\boldsymbol{\eta})^{\nu+n} \exp\left[\boldsymbol{\eta}^T \left(\left[\sum_{i=1}^n \mathbf{t}(x_i)\right] + \nu \boldsymbol{\chi}\right)\right]$$

Pseudo-observation  
↓

Note that:

$$P(D | \boldsymbol{\eta}) = \left(\frac{1}{Z(\boldsymbol{\eta})}\right)^n \left[\prod_{i=1}^n h(\mathbf{x}_i)\right] \exp\left[\boldsymbol{\eta}^T \left(\sum_{i=1}^n \mathbf{t}(\mathbf{x}_i)\right)\right]$$

CS 2750 Machine Learning