FE540 금융공학 인공지능 및 기계학습

Gaussian Models

Kee-Eung Kim
Department of Computer Science
KAIST

Korea Advanced Institute of Science and Technology
한국과학기술원

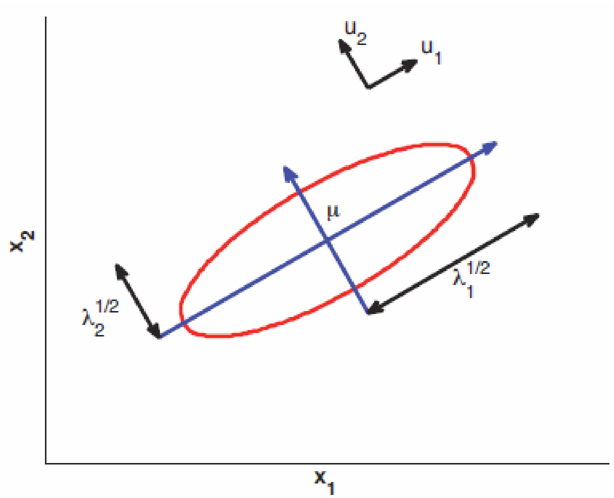
Multi-Variate Normal (Gaussian)

$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]$$

- Given eigen-decomposition $\boldsymbol{\Sigma} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top$
  - $\boldsymbol{\Sigma}^{-1} = \mathbf{U}^{-\top}\boldsymbol{\Lambda}^{-1}\mathbf{U}^{-1} = \mathbf{U}\boldsymbol{\Lambda}^{-1}\mathbf{U}^\top = \sum_{i=1}^{D} \frac{1}{\lambda_i}\mathbf{u}_i\mathbf{u}_i^\top$
  - exponent = Mahalanobis distance

$$(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}) = (\mathbf{x}-\boldsymbol{\mu})^\top \left(\sum_i \frac{1}{\lambda_i}\mathbf{u}_i\mathbf{u}_i^\top\right)(\mathbf{x}-\boldsymbol{\mu}) = \sum_i \frac{y_i^2}{\lambda_i}$$

where $y_i = \mathbf{u}_i^\top(\mathbf{x}-\boldsymbol{\mu})$



- Euclidean distance in a transformed coordinate system, shifted by $\boldsymbol{\mu}$ and rotated by $\mathbf{U}$

MLE for MVN

- Given N iid samples $\mathbf{x}_i \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$
  - $\hat{\boldsymbol{\mu}}_{\mathrm{ML}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i \equiv \bar{\mathbf{x}}$
  - $\hat{\boldsymbol{\Sigma}}_{\mathrm{ML}} = \frac{1}{N} \sum_{i=1}^{N} (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^\top = \frac{1}{N} (\sum_i \mathbf{x}_i \mathbf{x}_i^\top) - \bar{\mathbf{x}} \bar{\mathbf{x}}^\top$
- Derivation

Gaussian as Maximum Entropy

- Given the mean and covariance, Gaussian is the distribution with maximum entropy
  - Captures first two moments estimated from data, while making no further assumption
- Proof
  - for simplicity, assume $\hat{\boldsymbol{\mu}} = 0$
  - Let $q(\mathbf{x})$ be *any* density satisfying $\int x_i x_j q(\mathbf{x}) d\mathbf{x} = \Sigma_{ij}$
  - Let $p(\mathbf{x}) = \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ (which implies $\int x_i x_j p(\mathbf{x}) d\mathbf{x} = \Sigma_{ij}$)

$$
\begin{aligned}
0 \leq KL(q||p) &= \int q(\mathbf{x}) \log \frac{q(\mathbf{x})}{p(\mathbf{x})} d\mathbf{x} \\
&= -h(q) - \int q(\mathbf{x}) \log p(\mathbf{x}) d\mathbf{x} \\
&= -h(q) - \int p(\mathbf{x}) \log p(\mathbf{x}) d\mathbf{x} \\
&= -h(q) + h(p)
\end{aligned}
$$

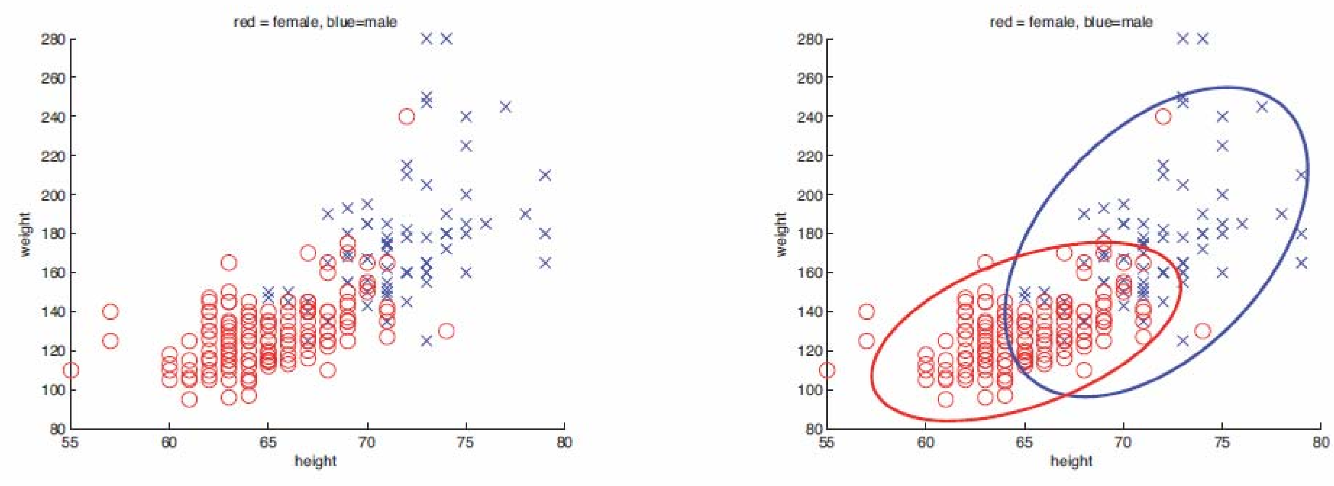
KAIST
Korea Advanced Institute of Science and Technology
한국과학기술원

Gaussian Discriminant Analysis (GDA)

□ Use MVN for class conditional densities in generative classifier $\boldsymbol{\theta} = \{\pi_c, \boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c | c = 1, \ldots, C\}$

- $p(\mathbf{x}|y = c, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$
- $p(y = c|\mathbf{x}, \boldsymbol{\theta}) = \frac{p(y=c|\boldsymbol{\theta})p(\mathbf{x}|y=c,\boldsymbol{\theta})}{\sum_{c'} p(y=c'|\boldsymbol{\theta})p(\mathbf{x}|y=c',\boldsymbol{\theta})} = \frac{\pi_c \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_c,\boldsymbol{\Sigma}_c)}{\sum_{c'} \pi_{c'} \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_{c'},\boldsymbol{\Sigma}_{c'})}$
- $\hat{y}(\mathbf{x}) = \operatorname{argmax}_c [\log \pi_c + \log \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)]$

- Nearest centroids classifier: choose the class with minimum Mahalanobis distance to $\boldsymbol{\mu}_c$ minus log-prior



(a) (b)

KAIST
Advanced Institute of Science and Technology
한국과학기술원

Quadratic Discriminant Analysis (QDA)

☐ Plug-in the definition of Gaussian into posterior:

$$p(y=c|\mathbf{x},\boldsymbol{\theta}) = \frac{\pi_c (2\pi)^{-\frac{D}{2}} |\boldsymbol{\Sigma}_c|^{-\frac{1}{2}} \exp[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_c)^\top \boldsymbol{\Sigma}_c^{-1}(\mathbf{x}-\boldsymbol{\mu}_c)]}{\sum_{c'} \pi_{c'} (2\pi)^{-\frac{D}{2}} |\boldsymbol{\Sigma}_{c'}|^{-\frac{1}{2}} \exp[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_{c'})^\top \boldsymbol{\Sigma}_{c'}^{-1}(\mathbf{x}-\boldsymbol{\mu}_{c'})]}$$

$$p(y=c|\mathbf{x},\boldsymbol{\theta}) = \log \pi_c - \frac{1}{2}\log|\boldsymbol{\Sigma}_c| - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_c)^\top \boldsymbol{\Sigma}_c^{-1}(\mathbf{x}-\boldsymbol{\mu}_c) + C$$

☐ Decision boundary: $p(y=c|\mathbf{x},\boldsymbol{\theta}) \geq p(y=c'|\mathbf{x},\boldsymbol{\theta})$

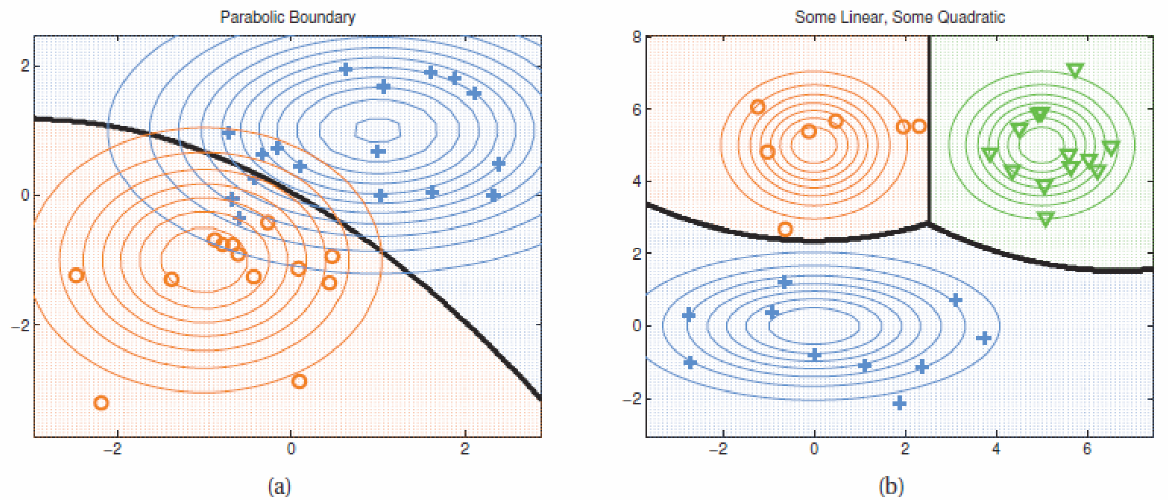


Figure 4.3 Quadratic decision boundaries in 2D for the 2 and 3 class case. Figure generated by discrimAnalysisDboundariesDemo.

Linear Discriminant Analysis (LDA)

□ Covariance matrices are tied/shared: $\boldsymbol{\Sigma}_c = \boldsymbol{\Sigma}$

$$p(y=c|\mathbf{x},\boldsymbol{\theta}) \propto \pi_c \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_c)^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_c)\right]$$

$$= \exp\left[\boldsymbol{\mu}_c^\top \boldsymbol{\Sigma}^{-1}\mathbf{x} - \frac{1}{2}\boldsymbol{\mu}_c^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_c + \log \pi_c\right] \exp\left[-\frac{1}{2}\mathbf{x}^\top \boldsymbol{\Sigma}^{-1}\mathbf{x}\right]$$

- The last term cancels out, thus

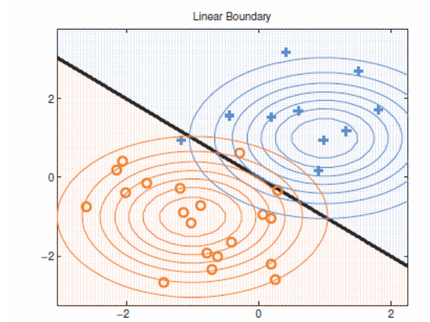
$$p(y=c|\mathbf{x},\boldsymbol{\theta}) = \frac{e^{\boldsymbol{\beta}_c^\top \mathbf{x}+\gamma_c}}{\sum_{c'} e^{\boldsymbol{\beta}_{c'}^\top \mathbf{x}+\gamma_{c'}}} = \mathcal{S}(\boldsymbol{\eta})_c$$

$$\boldsymbol{\beta}_c = \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_c$$

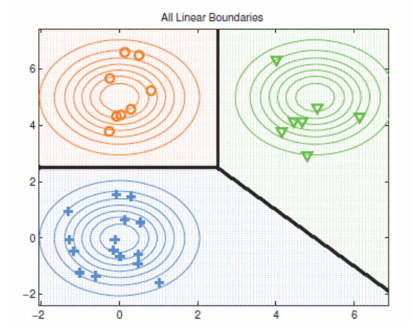
$$\gamma_c = -\frac{1}{2}\boldsymbol{\mu}_c^\top \Sigma^{-1}\boldsymbol{\mu}_c + \log \pi_c$$

$$\boldsymbol{\eta} = [\boldsymbol{\beta}_c^\top \mathbf{x} + \gamma_1, \ldots, \boldsymbol{\beta}_C^\top \mathbf{x} + \gamma_C]$$

$$\mathcal{S}(\boldsymbol{\eta})_c = \frac{e^{\eta_c}}{\sum_{c'} e^{\eta_{c'}}}$$



(a)

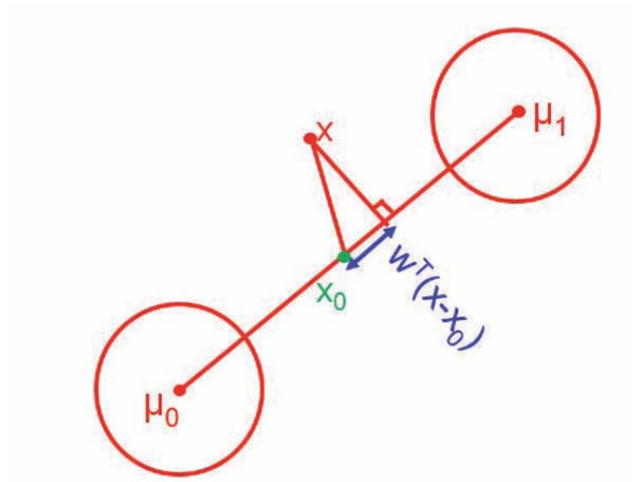


(b)

Two-Class LDA

$$p(y=1|\mathbf{x},\boldsymbol{\theta}) = \frac{e^{\boldsymbol{\beta}_1^\top \mathbf{x}+\gamma_1}}{e^{\boldsymbol{\beta}_1^\top \mathbf{x}+\gamma_1} + e^{\boldsymbol{\beta}_0^\top \mathbf{x}+\gamma_0}} = \frac{1}{1+e^{-[(\boldsymbol{\beta}_1-\boldsymbol{\beta}_0)^\top \mathbf{x}+(\gamma_1-\gamma_0)]}}$$

$$= \mathrm{sigmoid}((\boldsymbol{\beta}_1-\boldsymbol{\beta}_0)^\top \mathbf{x} + (\gamma_1-\gamma_0))$$



- Further simplifies to

$$p(y=1|\mathbf{x},\boldsymbol{\theta}) = \mathrm{sigmoid}(\mathbf{w}^\top(\mathbf{x}-\mathbf{x}_0))$$

where

$$\mathbf{w} = \boldsymbol{\beta}_1-\boldsymbol{\beta}_0 = \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)$$

$$\mathbf{x}_0 = \frac{1}{2}(\boldsymbol{\mu}_1+\boldsymbol{\mu}_0) - (\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)\frac{\log(\pi_1/\pi_0)}{(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)^\top\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)}$$

MLE for Discriminant Analysis

- Reminder: $\hat{\boldsymbol{\theta}}_{\mathrm{ML}} = \mathrm{argmax}_{\theta}\, p(\mathcal{D}|\boldsymbol{\theta})$

- log-likelihood:

$$\log p(\mathcal{D}|\boldsymbol{\theta}) = \left[\sum_{i=1}^{N}\sum_{c=1}^{C}\mathbb{I}(y_i = c)\log \pi_c\right] + \left[\sum_{c=1}^{C}\sum_{i:y_i=c}\log \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)\right]$$

- Derive ML estimation for the parameters!

KAIST
Korea Advanced Institute of Science and Technology
한국과학기술원

Four Choices for Covariance Matrix

Assumption	Covariance Matrix	# of Parameters
Shared, Hyperspheric	$\mathbf{\Sigma}_c = \mathbf{\Sigma} = \sigma^2\mathbf{I}$	1
Shared, Axis-aligned	$\mathbf{\Sigma}_c = \mathbf{\Sigma}$ with $\Sigma_{ij} = 0, i \neq j$	D
Shared, Hyperellipsoidal	$\mathbf{\Sigma}_c = \mathbf{\Sigma}$	D(D+1)/2
Different, Hyperellipsoidal	$\mathbf{\Sigma}_c$	CD(D+1)/2

- ☐ Is the most complex model always desirable?

- ☐ Regularized discriminant analysis (RDA) [Friedman 1989]
  - $\hat{\mathbf{\Sigma}}_c = \alpha\sigma^2\mathbf{I} + \beta\mathbf{\Sigma} + (1-\alpha-\beta)\mathbf{\Sigma}_c$
  - $\alpha = \beta = 0 \Rightarrow$ Quadratic classifier
  - $\alpha = 0$ and $\beta = 1 \Rightarrow$ Linear classifier
  - $\alpha = 1$ and $\beta = 0 \Rightarrow$ Nearest mean classifier

Inference with MVN

Theorem 4.3.1 (Marginals and conditionals of an MVN). *Suppose* $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2)$ *is jointly Gaussian with parameters*

$$\boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{pmatrix}, \quad \boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1} = \begin{pmatrix} \boldsymbol{\Lambda}_{11} & \boldsymbol{\Lambda}_{12} \\ \boldsymbol{\Lambda}_{21} & \boldsymbol{\Lambda}_{22} \end{pmatrix} \tag{4.67}$$

Then the marginals are given by

$$\begin{aligned} p(\mathbf{x}_1) &= \mathcal{N}(\mathbf{x}_1 | \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_{11}) \\ p(\mathbf{x}_2) &= \mathcal{N}(\mathbf{x}_2 | \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_{22}) \end{aligned} \tag{4.68}$$

and the posterior conditional is given by

$$\boxed{\begin{aligned} p(\mathbf{x}_1|\mathbf{x}_2) &= \mathcal{N}(\mathbf{x}_1|\boldsymbol{\mu}_{1|2}, \boldsymbol{\Sigma}_{1|2}) \\ \boldsymbol{\mu}_{1|2} &= \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\mathbf{x}_2 - \boldsymbol{\mu}_2) \\ &= \boldsymbol{\mu}_1 - \boldsymbol{\Lambda}_{11}^{-1}\boldsymbol{\Lambda}_{12}(\mathbf{x}_2 - \boldsymbol{\mu}_2) \\ &= \boldsymbol{\Sigma}_{1|2}\left(\boldsymbol{\Lambda}_{11}\boldsymbol{\mu}_1 - \boldsymbol{\Lambda}_{12}(\mathbf{x}_2 - \boldsymbol{\mu}_2)\right) \\ \boldsymbol{\Sigma}_{1|2} &= \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21} = \boldsymbol{\Lambda}_{11}^{-1} \end{aligned}} \tag{4.69}$$

□ Interpolation, Data Imputation, ...

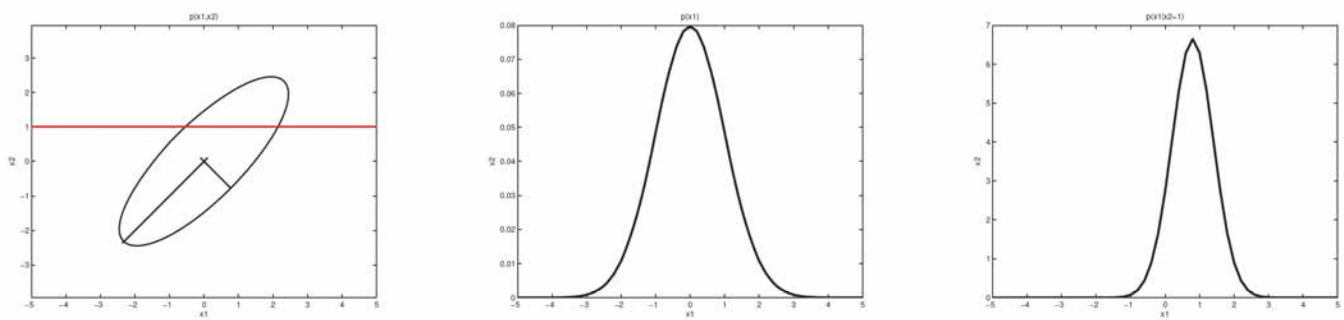
Derivation

Example: 2D Gaussian

- Given mean vector and covariance matrix

$$\boldsymbol{\mu} = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \qquad \boldsymbol{\Sigma} = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$$

  - marginal: $p(x_1) = \mathcal{N}(x_1|\mu_1, \sigma_1^2)$
  - conditional:
    $p(x_1|x_2) = \mathcal{N}(x_1|\mu_1 + \frac{\rho\sigma_1\sigma 2}{\sigma_2^2}(x_2 - \mu_2), \sigma_1^2 - \frac{(\rho\sigma_1\sigma_2)^2}{\sigma_2^2})$



(a) (b) (c)

Figure 4.9 (a) A joint Gaussian distribution $p(x_1, x_2)$ with a correlation coefficient of 0.8. We plot the 95% contour and the principal axes. (b) The unconditional marginal $p(x_1)$. (c) The conditional $p(x_1|x_2) = \mathcal{N}(x_1|0.8, 0.36)$, obtained by slicing (a) at height $x_2 = 1$. Figure generated by `gaussCondition2Ddemo2`.

Linear Gaussian Systems

- Suppose hidden $\mathbf{x} \in \mathbb{R}^{D_x}$ and observation $\mathbf{y} \in \mathbb{R}^{D_y}$
  - MVN prior and likelihood

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

$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}|\mathbf{A}\mathbf{x} + \mathbf{b}, \boldsymbol{\Sigma}_y)$$

- Bayes rule for linear Gaussian systems

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

$$\boldsymbol{\Sigma}_{x|y}^{-1} = \boldsymbol{\Sigma}_x^{-1} + \mathbf{A}^\top \boldsymbol{\Sigma}_y^{-1} \mathbf{A}$$

$$\boldsymbol{\mu}_{x|y} = \boldsymbol{\Sigma}_{x|y}[\mathbf{A}^\top \boldsymbol{\Sigma}_y^{-1}(\mathbf{y} - \mathbf{b}) + \boldsymbol{\Sigma}_x^{-1}\boldsymbol{\mu}_x]$$

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{y}|\mathbf{A}\boldsymbol{\mu}_x + \mathbf{b}, \boldsymbol{\Sigma}_y + \mathbf{A}\boldsymbol{\Sigma}_x\mathbf{A}^\top)$$

- Kalman filter, Probabilistic PCA, Gaussian processes, …

Derivation

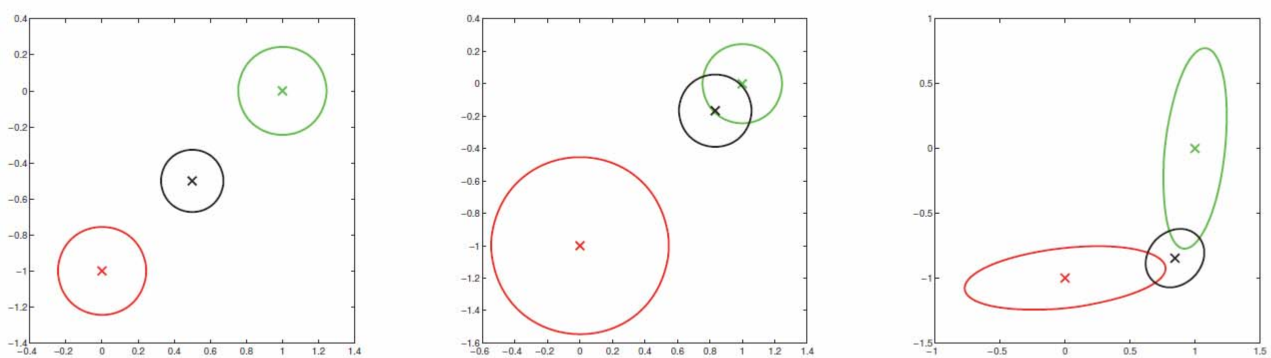
Linear Gaussian Systems

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x)$$
$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}|\mathbf{A}\mathbf{x} + \mathbf{b}, \boldsymbol{\Sigma}_y)$$

$$p(\mathbf{x}|\mathbf{y}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_{x|y}, \boldsymbol{\Sigma}_{x|y})$$
$$\boldsymbol{\Sigma}_{x|y}^{-1} = \boldsymbol{\Sigma}_x^{-1} + \mathbf{A}^\top \boldsymbol{\Sigma}_y^{-1} \mathbf{A}$$
$$\boldsymbol{\mu}_{x|y} = \boldsymbol{\Sigma}_{x|y}[\mathbf{A}^\top \boldsymbol{\Sigma}_y^{-1}(\mathbf{y} - \mathbf{b}) + \boldsymbol{\Sigma}_x^{-1}\boldsymbol{\mu}_x]$$
$$p(\mathbf{y}) = \mathcal{N}(\mathbf{y}|\mathbf{A}\boldsymbol{\mu}_x + \mathbf{b}, \boldsymbol{\Sigma}_y + \mathbf{A}\boldsymbol{\Sigma}_x\mathbf{A}^\top)$$

☐ Sensor fusion

- prior $p(\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$
- 2 noisy obs: $p(\mathbf{y}_1|\mathbf{x}) = \mathcal{N}(\mathbf{x}, \boldsymbol{\Sigma}_{y,1}), p(\mathbf{y}_2|\mathbf{x}) = \mathcal{N}(\mathbf{x}, \boldsymbol{\Sigma}_{y,2})$
- posterior $p(\mathbf{x}|\mathbf{y}_1, \mathbf{y}_2) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$



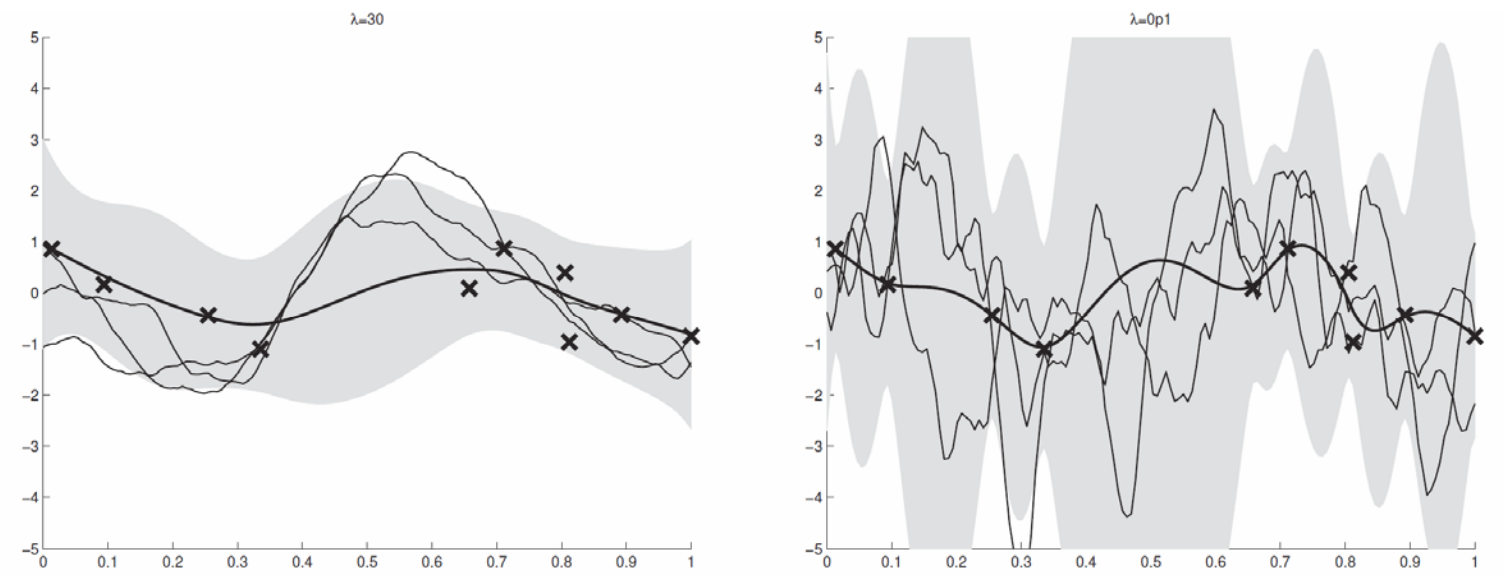
(a) (b) (c)

KAIST
orea Advanced Institute of Science and Technology
한국과학기술원

Linear Gaussian Systems

☐ Interpolation of noisy data

- N observations and $N-D$ unknowns
- assume $\mathbf{y} = \mathbf{A}\mathbf{x} + \boldsymbol{\epsilon}$, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$
- e.g. $\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$
- use the prior in the noise-free case: $\mathbf{\Sigma}_x = (\lambda^2 \mathbf{L}^\top \mathbf{L})^{-1}$
- compute posterior mean and variance



(a) (b)

Parameter Inference in MVN

- Infer $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\Sigma})$ from data $\mathbf{x}_i \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$
  - We cover mean only; covariance matrix in the textbook
  - likelihood: $p(\mathcal{D}|\boldsymbol{\mu}) = \mathcal{N}(\bar{\mathbf{x}}|\boldsymbol{\mu}, \frac{1}{N}\boldsymbol{\Sigma})$
  - prior: $p(\boldsymbol{\mu}) = \mathcal{N}(\boldsymbol{\mu}|\mathbf{m}_0, \mathbf{V}_0)$
  - compute posterior:

$$p(\boldsymbol{\mu}|\mathcal{D}, \boldsymbol{\Sigma}) = \mathcal{N}(\boldsymbol{\mu}|\mathbf{m}_N, \mathbf{V}_N)$$

$$\mathbf{V}_N^{-1} = \mathbf{V}_0^{-1} + N\boldsymbol{\Sigma}^{-1}$$

$$\mathbf{m}_N = \mathbf{V}_N(\boldsymbol{\Sigma}^{-1}(N\bar{\mathbf{x}} + \mathbf{V}_0^{-1}\mathbf{m}_0)$$

[Bonus Material]
Applications of MVN Inference

Korea Advanced Institute of Science and Technology
한국과학기술원

Interpolation with MVN

- Simplified version of Gaussian Process Regression
- Goal: estimate function f from data $y_i = f(t_i)$
  - Discretize: $x_j = f(s_j), \quad s_j = jh, \quad h = T/D, 1 \leq j \leq D$
  - Assume smooth function values:

$$x_j = \frac{1}{2}(x_{j-1} + x_{j+1}) + \epsilon_j, \quad 2 \leq j \leq D-2, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, (1/\lambda)\mathbf{I})$$

  - small λ = wiggly, large λ = very smooth
  - matrix-vector notation: $\mathbf{Lx} = \boldsymbol{\epsilon}$
    using $(D-2) \times D$ second-order finite difference matrix

$$\mathbf{L} = \frac{1}{2}\begin{pmatrix} -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & & \ddots & & \\ & & & -1 & 2 & -1 \end{pmatrix}$$

  - hence, prior: $p(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\mathbf{0}, (\lambda^2\mathbf{L}^\top\mathbf{L})^{-1}) \propto \exp\left(-\frac{\lambda^2}{2}\|\mathbf{Lx}\|_2^2\right)$

Interpolation with MVN

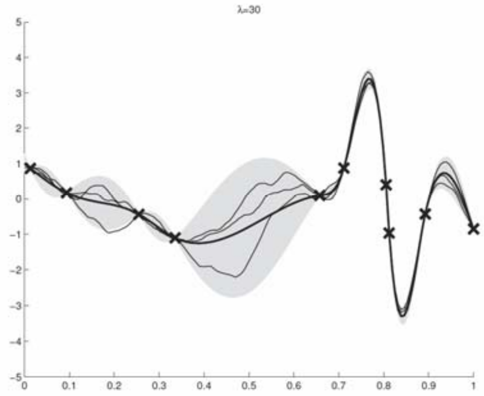
- Let $\mathbf{x}_2$ be N noise-free data and $\mathbf{x}_1$ be $D-N$ unknowns
- Interpolation = conditional on $\mathbf{x}_1$
  - $\mathbf{L} = [\mathbf{L}_1 \ \mathbf{L}_2], \quad \mathbf{L}_1 \in \mathbb{R}^{(D-2)\times(D-N)}, \mathbf{L}_2 \in \mathbb{R}^{(D-2)\times N}$
  - precision matrix (assume $\lambda = 1$)

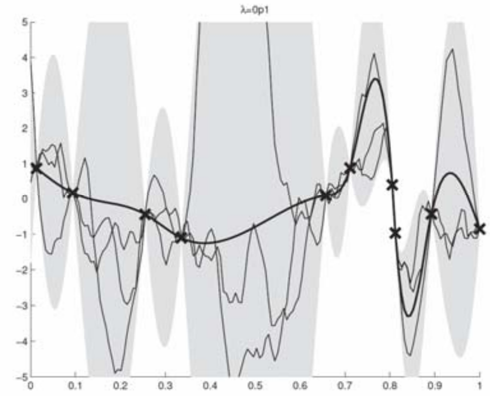
$$\mathbf{\Lambda} = \mathbf{L}^\top \mathbf{L} = \begin{pmatrix} \mathbf{L}_1^\top \mathbf{L}_1 & \mathbf{L}_1^\top \mathbf{L}_2 \\ \mathbf{L}_2^\top \mathbf{L}_1 & \mathbf{L}_2^\top \mathbf{L}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{\Lambda}_{11} & \mathbf{\Lambda}_{12} \\ \mathbf{\Lambda}_{21} & \mathbf{\Lambda}_{22} \end{pmatrix}$$

  - conditionals: $p(\mathbf{x}_1|\mathbf{x}_2) = \mathcal{N}(\boldsymbol{\mu}_{1|2}, \mathbf{\Sigma}_{1|2})$

$$\boldsymbol{\mu}_{1|2} = -\mathbf{\Lambda}_{11}^{-1}\mathbf{\Lambda}_{12}\mathbf{x}_2 = -(\mathbf{L}_1^\top \mathbf{L}_1)^{-1}\mathbf{L}_1^\top \mathbf{L}_2 \mathbf{x}_2$$

$$\mathbf{\Sigma}_{1|2} = \mathbf{\Lambda}_{11}^{-1}$$





KAIST
Korea Advanced Institute of Science and Technology
한국과학기술원

Data Imputation

- Some entries are missing – guess the values
  - $\mathbf{h}_i$: indices of missing or hidden entries in the i-th instance
  - $\mathbf{v}_i$: indices of visible entries in the i-th instance
  - imputation = compute $\hat{x}_{h_{ij}} = E[x_{h_{ij}} | \mathbf{x}_{\mathbf{v}_i}, \boldsymbol{\theta}]$
    using marginal distribution $p(x_{h_{ij}} | \mathbf{x}_{\mathbf{v}_i}, \boldsymbol{\theta})$
  - bonus: $\mathrm{Var}[x_{h_{ij}} | \mathbf{v}_i, \boldsymbol{\theta}]$ as a measure of confidence
  - multiple imputation also possible