Machine Learning 1

- Lecture 5 -

Bayesian Linear Regression

Linear Classification

- Patrick Forré -

Slides created by:
Rianne van den Berg

Image credit: Kirillm | Getty Images

Overview

1. Bayesian linear regression
2. Generalization error decomposition
3. Classification and decision theory
4. Linear Discriminant Analysis (LDA)

Overview

1. **Bayesian linear regression**
2. Generalization error decomposition
3. Classification and decision theory
4. Linear Discriminant Analysis (LDA)

Linear Basis Models

- Data set $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ with targets. $\mathbf{t} = (t_1, \ldots, t_N)^\top$
- Linear basis model: $t = y(\mathbf{x}, \mathbf{w}) + \varepsilon$
- $\mathbf{x}_n \in \mathbb{R}^D$ $t_n \in \mathbb{R}$
- $y(\mathbf{x}, \mathbf{w}) = \mathbf{w}^\top \boldsymbol{\phi}(\mathbf{x})$
- $\mathbf{w} = \begin{pmatrix} w_0 \\ \vdots \\ w_{M-1} \end{pmatrix}$ $\phi_0 = 1$ $w_0 = \text{bias / offset}$
- $\boldsymbol{\phi}(\mathbf{x}) = \begin{pmatrix} \phi_0(\mathbf{x}) \\ \vdots \\ \phi_{M-1}(\mathbf{x}) \end{pmatrix}$
- $\varepsilon \sim \mathcal{N}(0, \beta^{-1})$

$$\boldsymbol{\Phi} = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \cdots & \phi_{M-1}(\mathbf{x}_1) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \cdots & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix}$$

Recap: Conditioning Gaussians

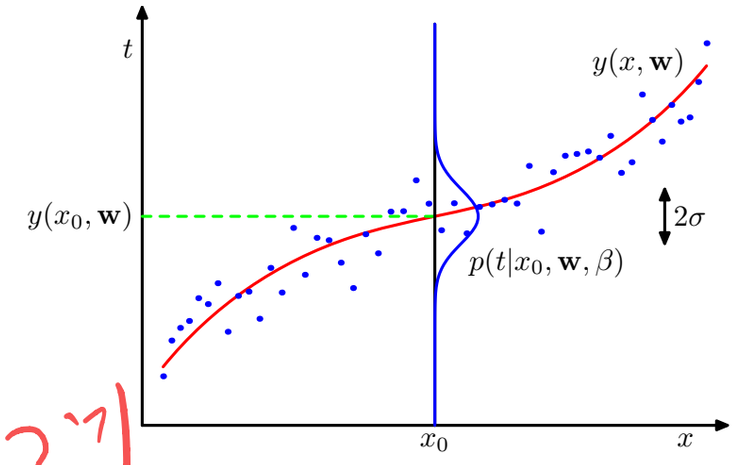
If $\begin{pmatrix} X_1 \\ X_2 \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \begin{pmatrix} \Sigma_{11}, \Sigma_{12} \\ \Sigma_{21}, \Sigma_{22} \end{pmatrix} \right)$ then:

1. $X_1 \sim \mathcal{N}\left(\mu_1, \Sigma_{11}\right)$

2. $X_1 \,|\, X_2 \sim \mathcal{N}\left(\mu_{1|2}(X_2), \Sigma_{1|2}\right)$ with:

   - $\mu_{1|2}(x_2) := \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(x_2 - \mu_2)$
   - $\Sigma_{1|2} := \Sigma_{11} + \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}$

Bayesian Linear Regression



- Data: $\mathbf{t} = (t_1, ..., t_N)^T \quad \mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N)^T$
- Likelihood: $p(t \,|\, \mathbf{x}, \mathbf{w}, \beta) =$ $\mathcal{N}(t|y(x,w), \beta^{-1})$

$$p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \beta) \overset{iid}{=} \prod_{n=1}^{N} \mathcal{N}(t_n|y(\mathbf{x}_n, \mathbf{w}), \beta^{-1}) = \mathcal{N}(\mathbf{t}|\Phi w, \beta^{-1} I_N)$$

- Conjugate prior: $p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m}_0, \mathbf{S}_0)$

$$\mathbf{\Phi} = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \cdots & \phi_{M-1}(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \cdots & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix}$$

- Posterior distribution:

$$p(\mathbf{w}|\mathbf{t}, \mathbf{X}) = \frac{p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \beta)p(\mathbf{w})}{p(\mathbf{t}|\mathbf{X}, \beta)} = \mathcal{N}(w|m_N, S_N)$$ Bishop Eq. 2.116

- Maximum A Posteriori estimate:

$$\mathbf{w}_{\mathrm{MAP}} = m_N$$

$$\mathbf{S}_N^{-1} = \mathbf{S}_0^{-1} + \beta \mathbf{\Phi}^T \mathbf{\Phi}$$

$$\mathbf{m}_N = \mathbf{S}_N(\mathbf{S}_0^{-1}\mathbf{m}_0 + \beta \mathbf{\Phi}^T \mathbf{t})$$

Bayesian Linear Regression

- Special simple prior: $p(\mathbf{w}|\alpha) = \mathcal{N}(w|0, \alpha^{-1} I_M)$

$\mathbf{m}_0 = \mathbf{0}$

$\mathbf{S}_0 = \alpha^{-1}\mathbf{I}_M$

$\alpha \in \mathbb{R}$

- Posterior $p(\mathbf{w}|\mathbf{X}, \mathbf{t}, \alpha, \beta) = \mathcal{N}(\mathbf{w}|\mathbf{m}_N, \mathbf{S}_N)$

$$\mathbf{m}_N = \mathbf{S}_N(\mathbf{S}_0^{-1}\mathbf{m}_0 + \beta\mathbf{\Phi}^T\mathbf{t}) = \beta \cdot S_N \cdot \Phi^T t$$

$$\mathbf{S}_N^{-1} = \mathbf{S}_0^{-1} + \beta\mathbf{\Phi}^T\mathbf{\Phi} = \alpha \cdot I_M + \beta \Phi^T \Phi$$

$$p(\mathbf{w}|\mathbf{X}, \mathbf{t}, \alpha, \beta) = \mathcal{N}(w|m_N, S_N, \alpha, \beta)$$

$$= \mathcal{N}(w|\beta S_N \cdot \Phi^T \cdot t, (\alpha I_M + \beta \Phi^T \Phi)$$

$$\mathbf{\Phi} = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \cdots & \phi_{M-1}(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \cdots & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix}$$

Bayesian Linear Regression

Limiting cases: $p(\mathbf{w}|\mathbf{X}, \mathbf{t}, \alpha, \beta) = \mathcal{N}(\mathbf{w}|\mathbf{m}_N, \mathbf{S}_N)$

$$\mathbf{m}_N = \beta \mathbf{S}_N \mathbf{\Phi}^T \mathbf{t}$$
$$\mathbf{S}_N^{-1} = \alpha \mathbb{1} + \beta \mathbf{\Phi}^T \mathbf{\Phi}$$

- Infinitely broad prior: no restriction on **w**!

$$p(\mathbf{w}|\alpha) = \mathcal{N}(\mathbf{w}|0, \alpha^{-1}\mathbb{1}) \quad \alpha \to 0$$

$$\lim_{\alpha \to 0} \mathbf{m}_N = \lim_{\alpha \to 0} \beta \left(\alpha \mathbb{1} + \beta \mathbf{\Phi}^T \mathbf{\Phi}\right)^{-1} \mathbf{\Phi}^T \mathbf{t} = \left(\mathbf{\Phi}^T \mathbf{\Phi}\right)^{-1} \mathbf{\Phi}^T \mathbf{t}$$

$\alpha \to 0$

$= \mathbf{w}_{ML}$ MLE

- Infinitely narrow prior:

$$p(\mathbf{w}|\alpha) = \mathcal{N}(\mathbf{w}|0, \alpha^{-1}\mathbb{1}) \quad \alpha \to \infty$$

$$\lim_{\alpha \to \infty} \mathbf{m}_N = \lim_{\alpha \to \infty} \beta \left(\alpha \mathbb{1} + \beta \mathbf{\Phi}^T \mathbf{\Phi}\right)^{-1} \mathbf{\Phi}^T \mathbf{t} = 0$$

$\alpha \to \infty$

$= m_0$ prior

$$\lim_{\alpha \to \infty} \mathbf{S}_N = \lim_{\alpha \to \infty} \left(\alpha \mathbb{1} + \beta \mathbf{\Phi}^T \mathbf{\Phi}\right)^{-1} = 0$$

Example: Sequential Bayesian Learning

Data: sequences of input *x*, target *t*

Synthetic data generated by $x \sim \mathcal{U}(x|-1,1)$ $t = f(x,\mathbf{a}) + \varepsilon$

$f(x,\mathbf{a}) = a_0 + a_1 \cdot x$ $\varepsilon \sim \mathcal{N}(0, 0.2^2)$

$a_0 = -0.3 \quad a_1 = 0.5$

Target modeling: $p(t'|x', \mathbf{w}, \beta) = \mathcal{N}(t'|y(x', \mathbf{w}), \beta^{-1})$ $\beta^{-1} = 0.2^{-2}$

Linear model: $y(x, \mathbf{w}) = w_0 + w_1 x$

Prior: $p(\mathbf{w}|\alpha) = \mathcal{N}(\mathbf{w}|\mathbf{0}, \alpha^{-1}\mathbb{1})$ $\alpha = 2$

When data arrives sequentially: posterior after N-1 datapoints is prior for arrival of N-th datapoint!

Example: Sequential Bayesian Learning

- Data generated by $t = a_0 + a_1 x + \varepsilon$

  $a_0 = -0.3 \quad a_1 = 0.5$

- Prior

  $p(\mathbf{w}|\alpha) = \mathcal{N}(\mathbf{w}|\mathbf{0}, \alpha^{-1}\mathbb{1})$

- Sample 1 datapoint

- Likelihood

  $p(t_1|x_1, \mathbf{w}, \beta) =$

- Posterior

  $p(\mathbf{w}|x_1, t_1, \alpha, \beta) \propto$

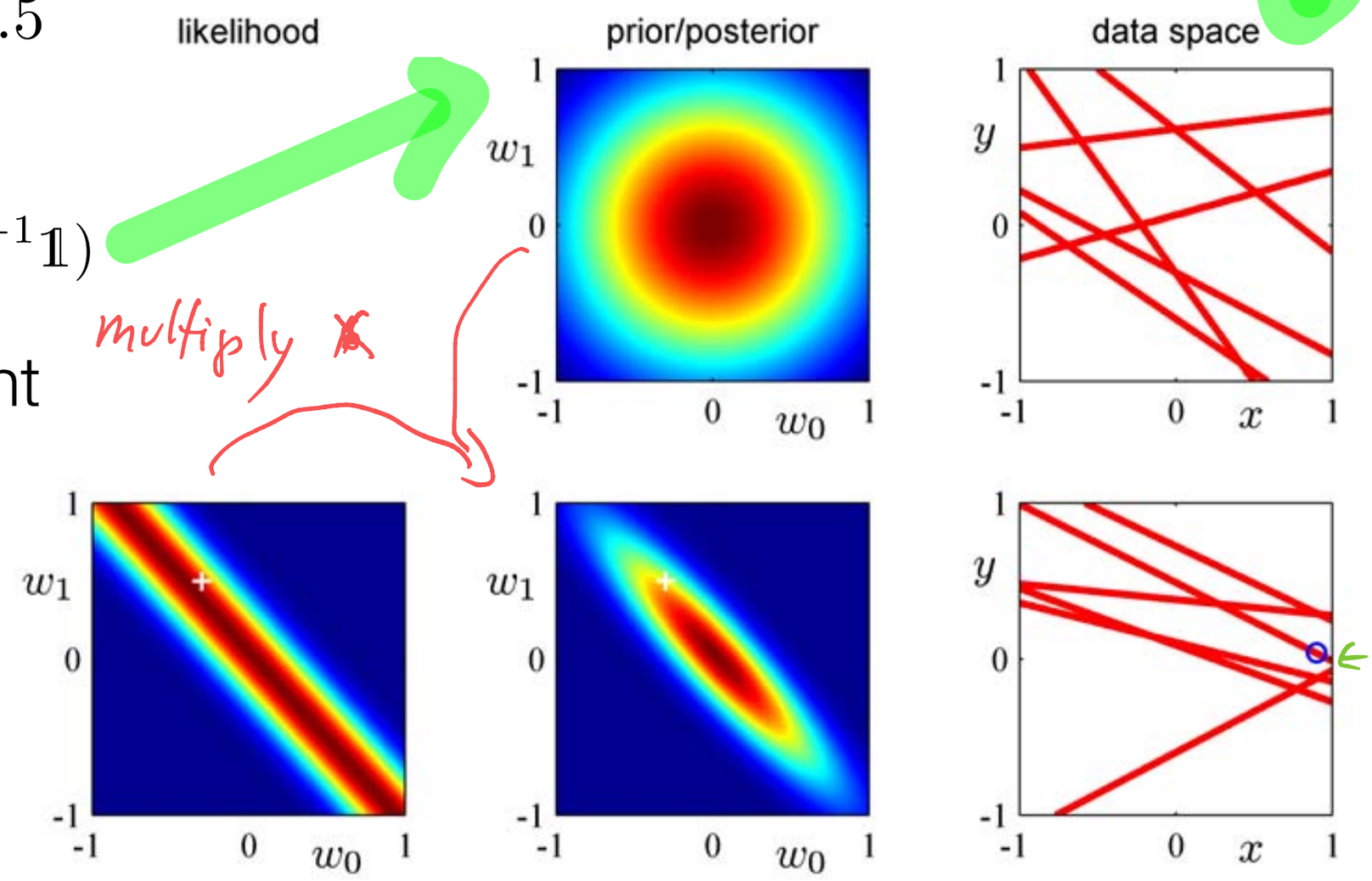


Figure: Sequential Bayesian learning (Bishop 3.7)

Example: Sequential Bayesian Learning

- Sample second datapoint:

- Posterior → prior :

- Likelihood?

$$p(t_2|x_2, \mathbf{w}, \beta)$$

OR

$$p(t_1, t_2|x_1, x_2, \mathbf{w}, \beta)$$

- Posterior

$$p(\mathbf{w}|(x_1, t_1), (x_2, t_2), \alpha, \beta) \propto$$

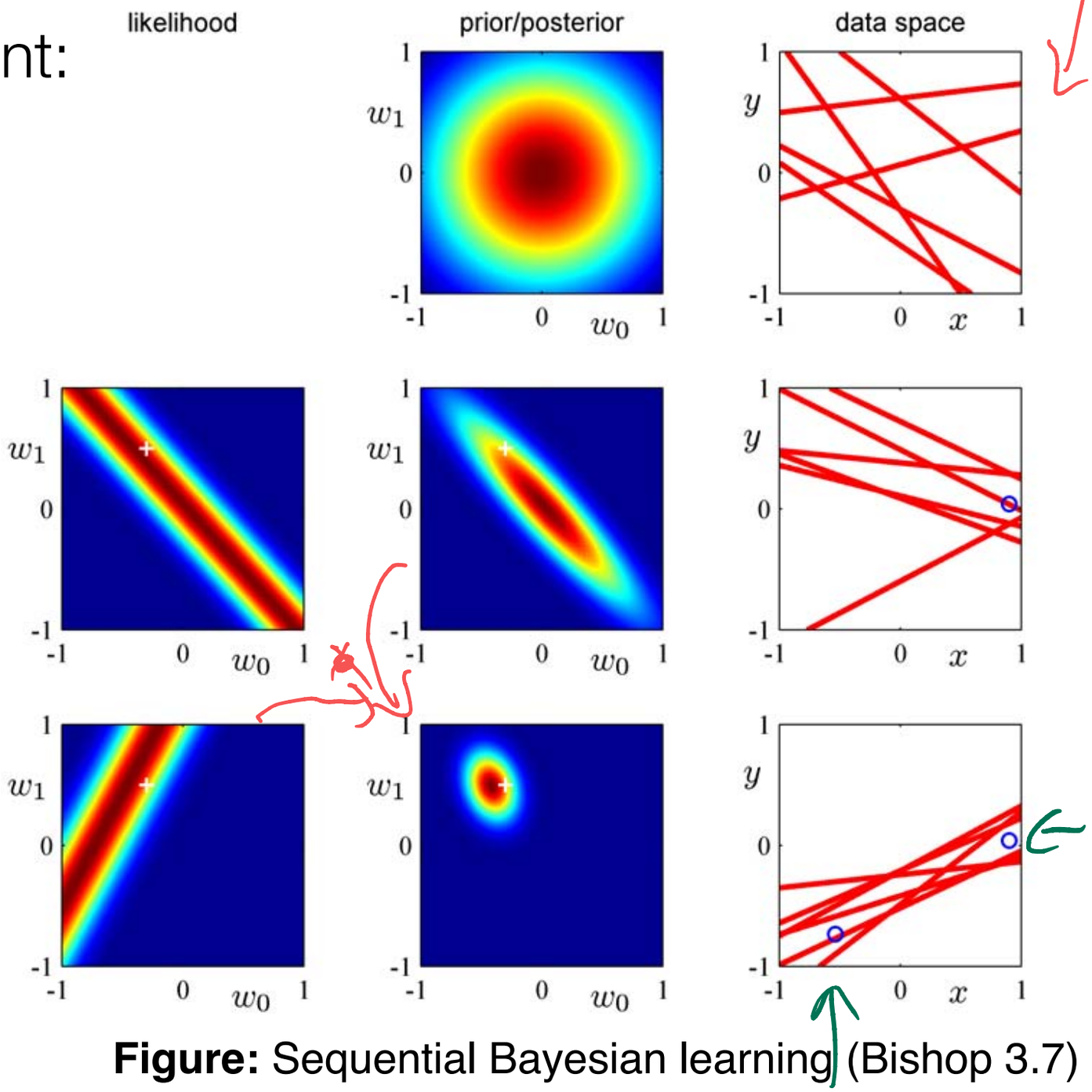


Figure: Sequential Bayesian learning (Bishop 3.7)

Example: Sequential Bayesian Learning

- After 19 datapoints

- Prior

$$p(\mathbf{w}|\{(x_n, t_n)\}_{n=1}^{19}, \alpha, \beta)$$

- Likelihood

$$p(t_{20}|x_{20}, \mathbf{w}, \beta)$$

- Posterior

$$p(\mathbf{w}|\{(x_n, t_n)\}_{n=1}^{20}, \alpha, \beta) \propto$$

$p(t_{20}|x_{20}, w) \cdot p(w|(x_n, t_n)_{n=1}^{19})$

- Much sharper posterior!

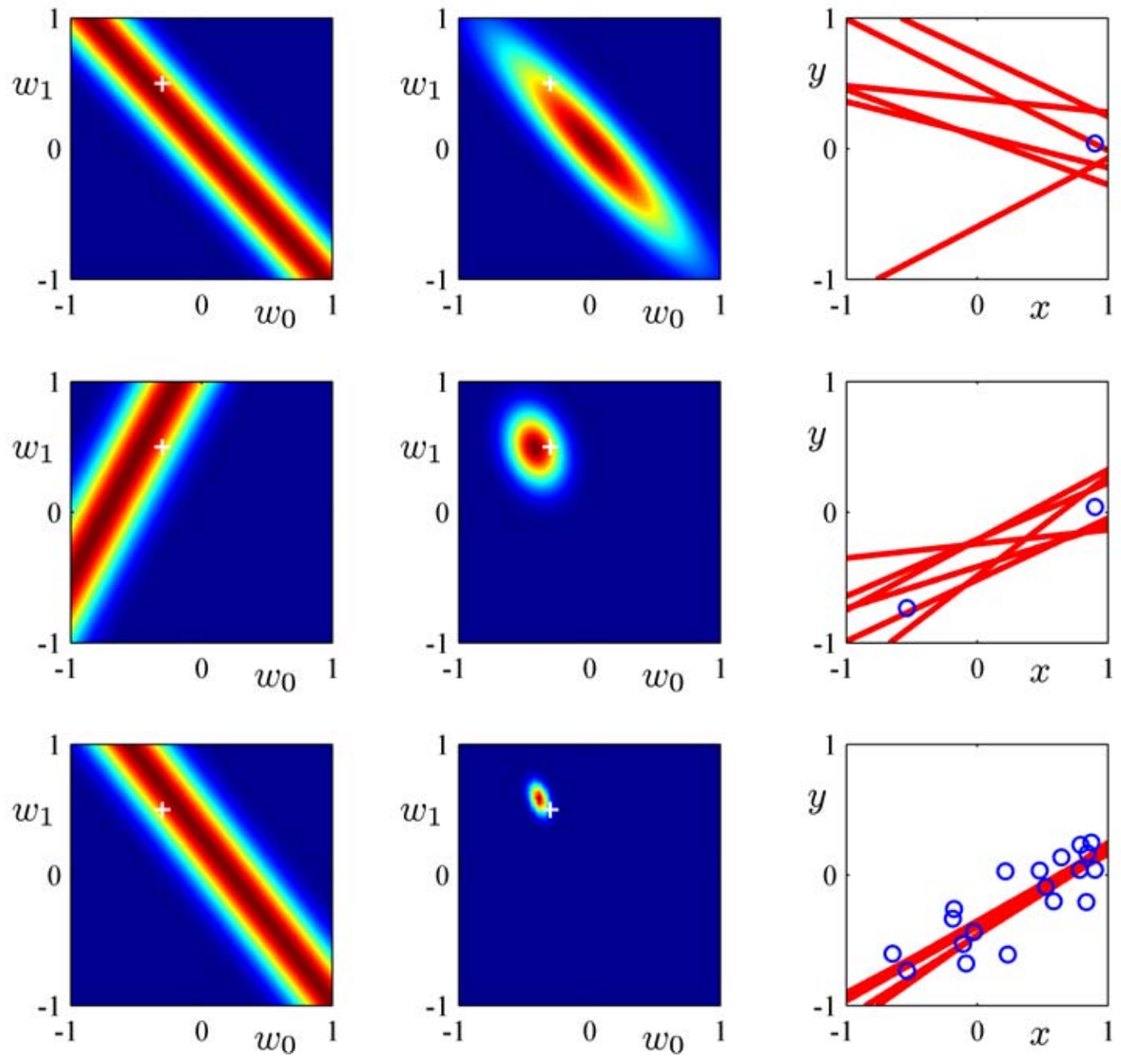


Figure: Sequential Bayesian learning (Bishop 3.7)

Infinite Data in Bayesian Linear Regression

- Poster distribution after observing *N* data points:

$$p(\mathbf{w}|\mathbf{X}, \mathbf{t}, \alpha, \beta) = \mathcal{N}(\mathbf{w}|\mathbf{m}_N, \mathbf{S}_N)$$

$$\mathbf{m}_N = \beta \mathbf{S}_N \mathbf{\Phi}^T \mathbf{t}$$

$$\mathbf{S}_N^{-1} = \alpha \mathbb{1} + \beta \mathbf{\Phi}^T \mathbf{\Phi}$$

$$\mathbf{\Phi} = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \cdots & \phi_{M-1}(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \cdots & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix}$$

- After an infinite amount of data :

$$\lim_{N \to \infty} S_N \stackrel{?}{=}$$

$$\lim_{N \to \infty} \left[\mathbf{\Phi}^T \mathbf{\Phi} \right]_{ij} = \infty \qquad \left(\Phi^T \Phi\right)_{ij} = \sum_{n=1}^{N} \left(\Phi^T\right)_{in} \Phi_{nj} = \sum_{n=1}^{N} \phi_i(x_n)\, \phi_j(x_n)$$

$$\lim_{N \to \infty} \mathbf{m}_N = \lim_{N \to \infty} \beta \mathbf{S}_N \mathbf{\Phi}^T \mathbf{t} \approx w_{ML}$$

Predictive Distribution

- Observed dataset with inputs $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, .., \mathbf{x}_N)^T$ and targets $\mathbf{t} = (t_1, t_2, ..., t_N)^T$
- Likelihood $p(t'|\mathbf{x}', \mathbf{w}, \beta) = \mathcal{N}(t|\boldsymbol{\phi}(\mathbf{x}')^T\mathbf{w}, \beta^{-1})$
- Prior $p(\mathbf{w}\,|\,\alpha) = \mathcal{N}(\mathbf{w}\,|\,\mathbf{0}, \alpha^{-1}\mathbf{I})$
- Posterior distribution

$$p(\mathbf{w}|\mathbf{X}, \mathbf{t}, \alpha, \beta) = \mathcal{N}(\mathbf{w}|\mathbf{m}_N, \mathbf{S}_N)$$

$$\mathbf{m}_N = \beta\mathbf{S}_N\boldsymbol{\Phi}^T\mathbf{t}$$

$$\mathbf{S}_N^{-1} = \alpha\mathbf{I} + \beta\boldsymbol{\Phi}^T\boldsymbol{\Phi}$$

- Predictive distribution for new input $\mathbf{x}'$

$$p(t'|\mathbf{x}', \mathbf{X}, \mathbf{t}, \alpha, \beta) = \mathcal{N}(t'\,|\,\mathbf{m}_N^T\boldsymbol{\phi}(\mathbf{x}),\ \sigma_N^2(\mathbf{x}))$$

Bishop Eq. 2.115

$$\sigma_N^2(\mathbf{x}') = \frac{1}{\beta} + \boldsymbol{\phi}(\mathbf{x}')^T\mathbf{S}_N\boldsymbol{\phi}(\mathbf{x}')$$

irreducible noise

$$\lim_{N\to\infty} \sigma_N^2(\mathbf{x}') = \frac{1}{\beta}$$

← Variance of true distribution

Predictive Distribution

- Datasets generated with

$$t = \sin(2\pi x) + \varepsilon$$
$$\varepsilon \sim \mathcal{N}(0, \beta^{-1})$$

- Dataset sizes N = 1, 2, 4, 25

- Model:

$$y(x, \mathbf{w}) = \boldsymbol{\phi}(x)^T \mathbf{w}$$

$$\phi_j(x) :=$$

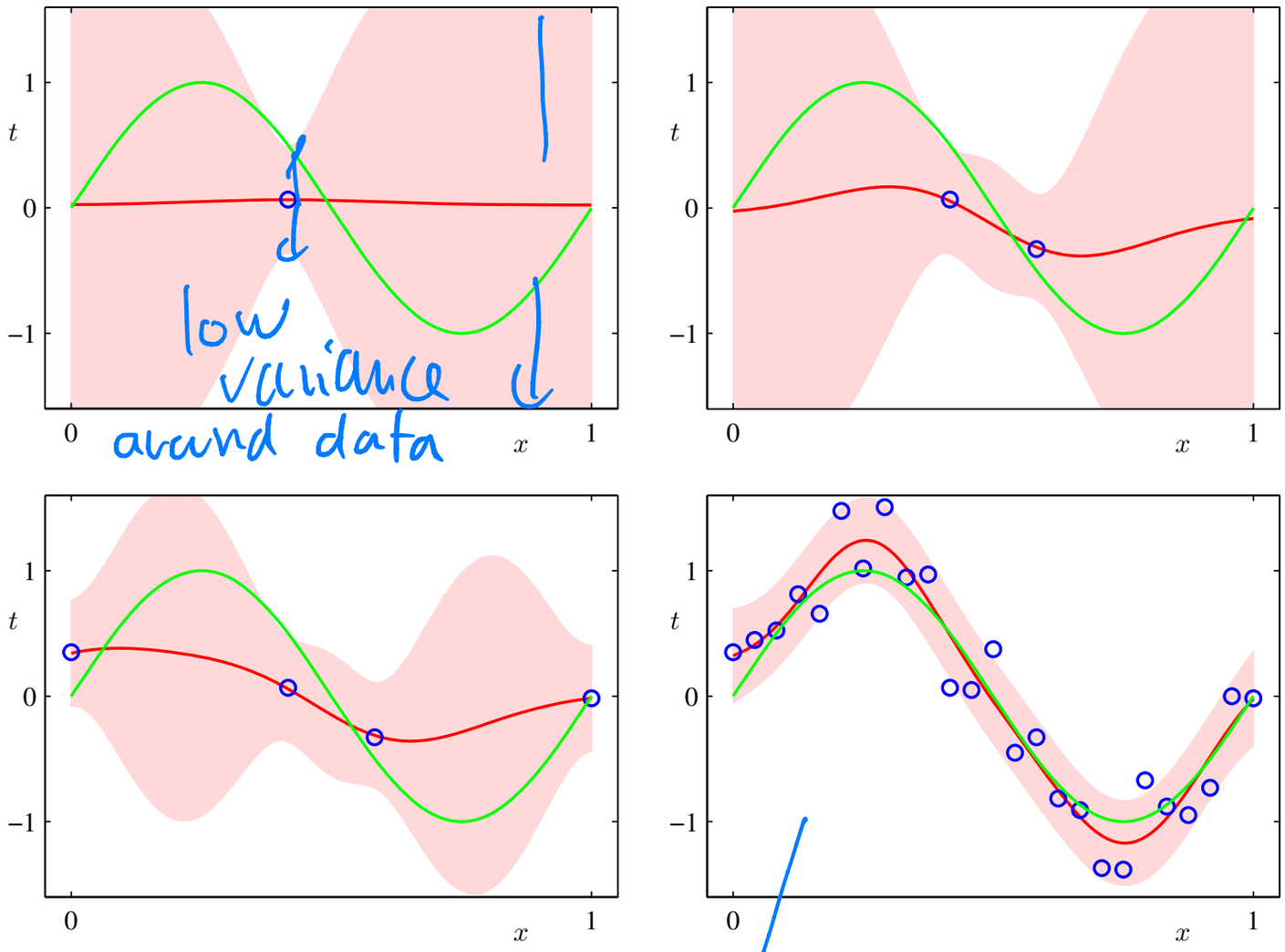


Figure: Predictive distribution (Bishop 3.8)

$$p(t'|x', \mathbf{X}, \mathbf{t}, \alpha, \beta) = \mathcal{N}(t|\boldsymbol{\phi}(x')^T \mathbf{m}_N, \sigma_N^2(x'))$$

$$\sigma_N^2(x') = \frac{1}{\beta} + \boldsymbol{\phi}(x')^T \mathbf{S}_N \boldsymbol{\phi}(x')$$

$$\mathbf{m}_N = \beta \mathbf{S}_N \boldsymbol{\Phi}^T \mathbf{t} \qquad \mathbf{S}_N^{-1} = \alpha \mathbb{1} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}$$

Samples drawn from Bayesian Predictive Distribution

$$p(\mathbf{w}|\mathbf{X},\mathbf{t},\alpha,\beta) = \mathcal{N}(\mathbf{w}|\mathbf{m}_N,\mathbf{S}_N) \qquad \mathbf{m}_N = \beta\mathbf{S}_N\mathbf{\Phi}^T\mathbf{t} \qquad \mathbf{S}_N^{-1} = \alpha\mathbb{1} + \beta\mathbf{\Phi}^T\mathbf{\Phi}$$

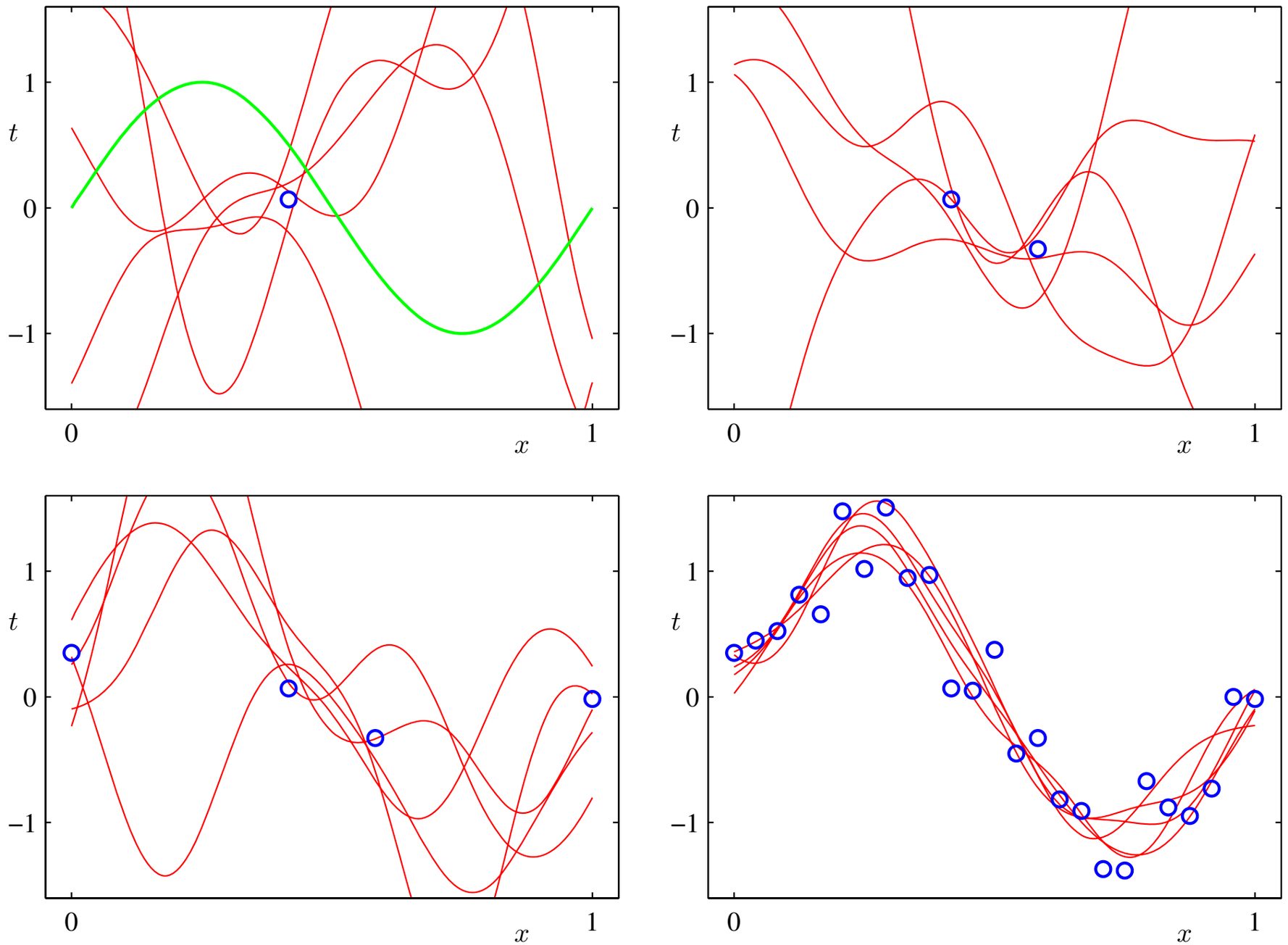
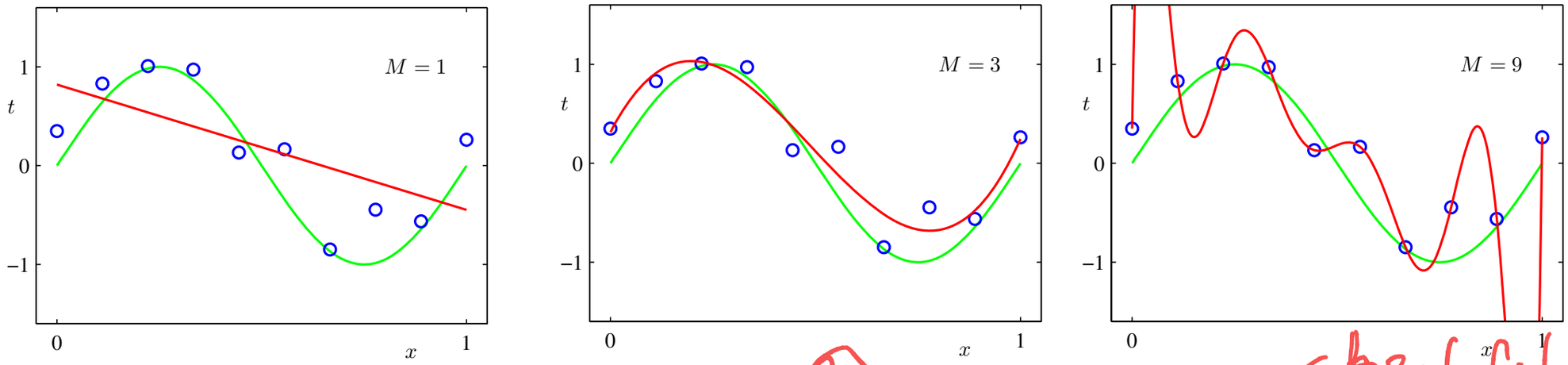


Figure: Sample functions y(x,w) with w sampled from posterior distribution (Bishop 3.9)

Bayesian Model Comparison
Polynomial Regression: How to choose M?

Model evidence: trade-off between model fit and model complexity

$$p(\mathbf{t}\,|\,\mathbf{X}, \mathscr{M}_M) = \int\int\int p(\mathbf{t}\,|\,\mathbf{X}, \mathbf{w}, \beta, \mathscr{M}_M)\, p(\mathbf{w}\,|\,\alpha)\, p(\alpha, \beta\,|\,\mathscr{M}_M)\, \mathbf{dw}\; d\alpha\, d\beta$$



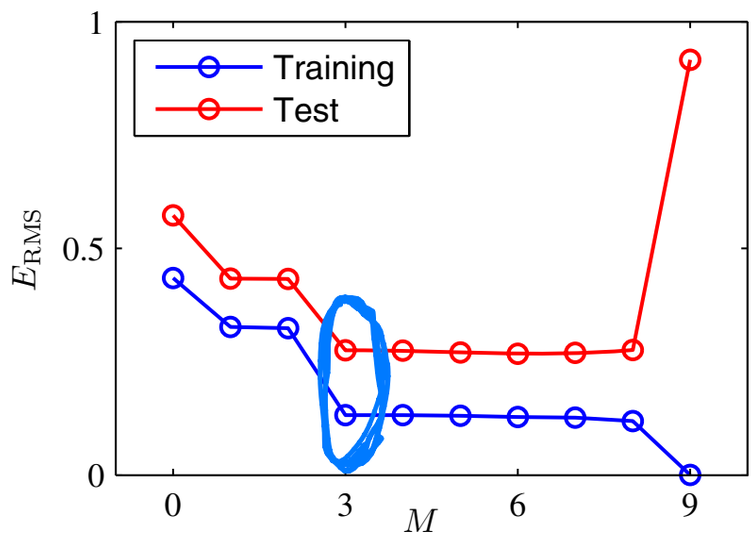


Figure: E_{rmse} (Bishop 1.5)

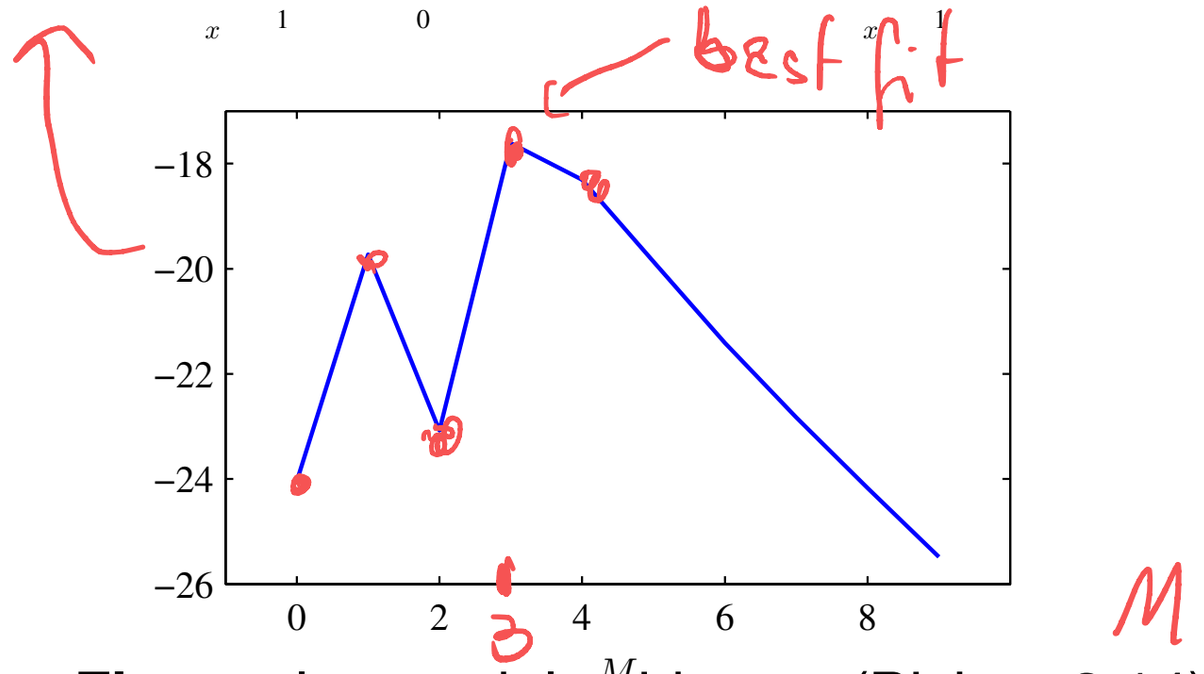


Figure: log model evidence (Bishop 3.14)

Limitations of Fixed Basis Functions

Advantages:

- Closed form solution for least-squares problem
- Tractable Bayesian treatment
- Nonlinear models mapping input variables to target variables through basis functions

Limitations:

- Assumption: Basis functions $\phi_j(\mathbf{x})$ are fixed, not learned.
- Curse of dimensionality: to cover growing dimensions D of input vectors, the number of basis functions needs to grow rapidly / exponentially

Overview

1. Bayesian linear regression
2. **Generalization error decomposition**
3. Classification and decision theory
4. Linear Discriminant Analysis (LDA)

Generalization Error

- Loss function: $L(t, p)$ between label t and distribution p(t'), e.g.:
  - log-loss $L(t, p) = -\log p(t)$ $\in \mathbb{R}$
  - square-loss $L(t, p) = (t - \mathbb{E}_{T' \sim p}[T'])^2$ $\in \mathbb{R}$
- Training data: $D = ((X_1, T_1), \ldots, (X_N, T_N)) \sim p_0$, (true, unknown)

  test point: $(X, T) \sim p_0$ (not seen before)
- Predictive distribution (after training with any method): $p_D := p(t \mid x, D)$ (target, input)
- Generalization error = expected test error (for input X):

  $\varepsilon_L(p_D \mid X) := \mathbb{E}_0[\, L(T, p(T' \mid X, D)) \mid X]$

Generalization Error Decomposition

- Generalization error for the square loss L:

$$\varepsilon_L(p_D) := \mathbb{E}_0[(T - \mathbb{E}[T'|X, D])^2]$$

- $\mathrm{Var}_L(p_D) := \mathbb{E}_0\left[\left(\mathbb{E}[T'|X,D] - \overline{\mathbb{E}}[T'|X]\right)^2\right]$ with

$$\overline{\mathbb{E}}[T'|X] := \mathbb{E}_0\left[\mathbb{E}[T'|X, D]\,|\,X\right]$$

← average over "several" data sets $D_1, \ldots, D_L$

- $\mathrm{Bias}_L^2(p_D) := \mathbb{E}_0\left[\left(\overline{\mathbb{E}}[T'|X] - \mathbb{E}_0[T'|X]\right)^2\right]$
- $\mathrm{Noise}_L(p_D) := \mathbb{E}_0\left[\left(T - \mathbb{E}_0[T'|X]\right)^2\right]$

best possible prediction

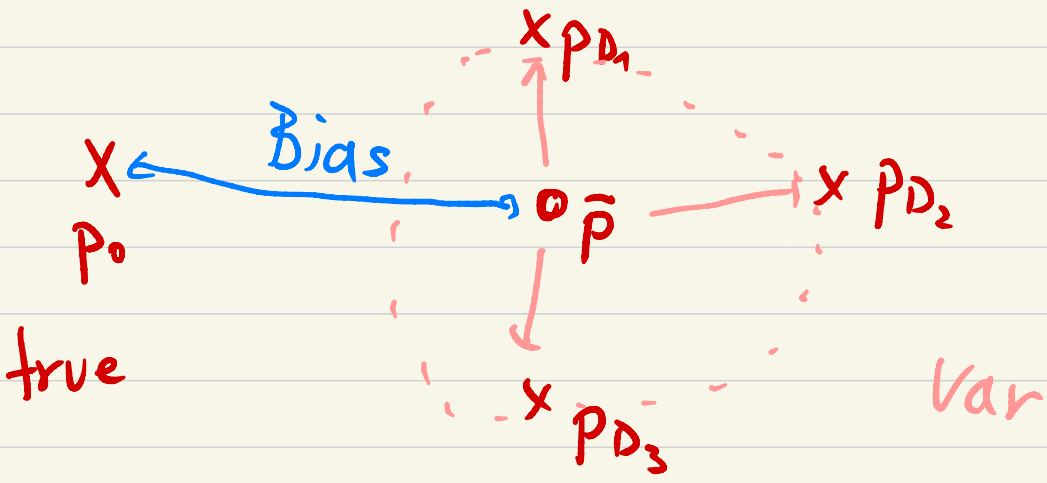
- Then:

$$\varepsilon_L(p_D) = \mathrm{Var}_L(p_D) + \mathrm{Bias}_L^2(p_D) + \mathrm{Noise}_L$$

high for flexible model

high for simple models

data distr.


p_{D_1}
Bias
p_0
$\tilde{p}$
p_{D_2}
p_{D_3}
true
Var

Bias-Variance Decomposition: Example

- Generate L datasets of N points:

$$x \sim U(0,1)$$

$$t = \sin(2\pi x) + \varepsilon \qquad \varepsilon \sim \mathcal{N}(0, \alpha^{-1})$$

$$\mathbb{E}[t|x] = \sin(2\pi x)$$

- L predictions with 24 Gaussian basis functions

$$y^{(l)}(x) = (\mathbf{w}^{(l)})^T \boldsymbol{\phi}(x)$$

$$E_D = \frac{1}{2}\sum_{i=1}^{N}\{t_n - \mathbf{w}^T\boldsymbol{\phi}(x)\}^2 + \frac{\lambda}{2}\mathbf{w}^T\mathbf{w}$$

$$\mathbb{E}_D[y_D(x)] = \bar{y}(x)$$

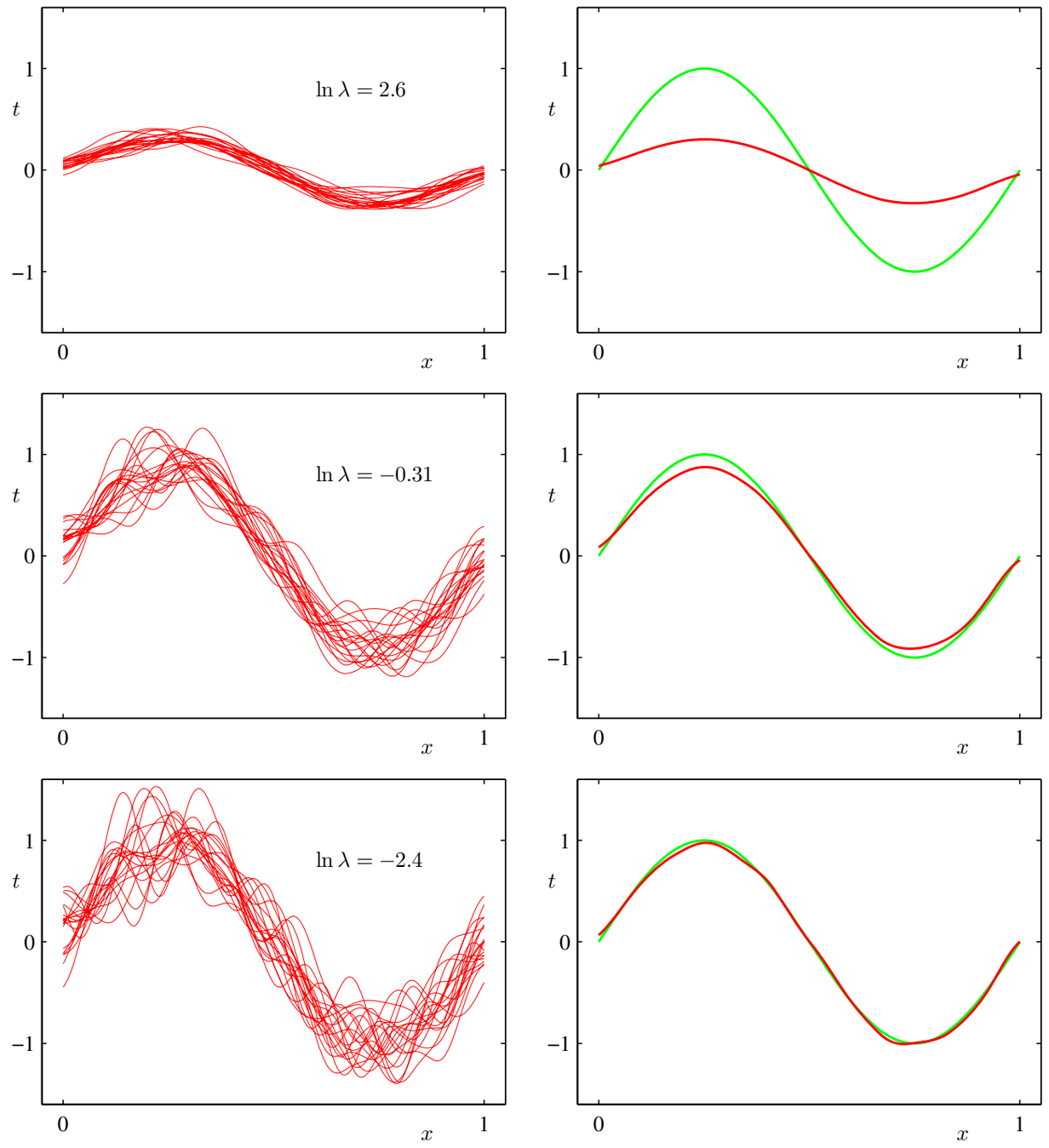


Figure: bias-variance decomposition (Bishop 3.5)

Bias-Variance Decomposition: Example

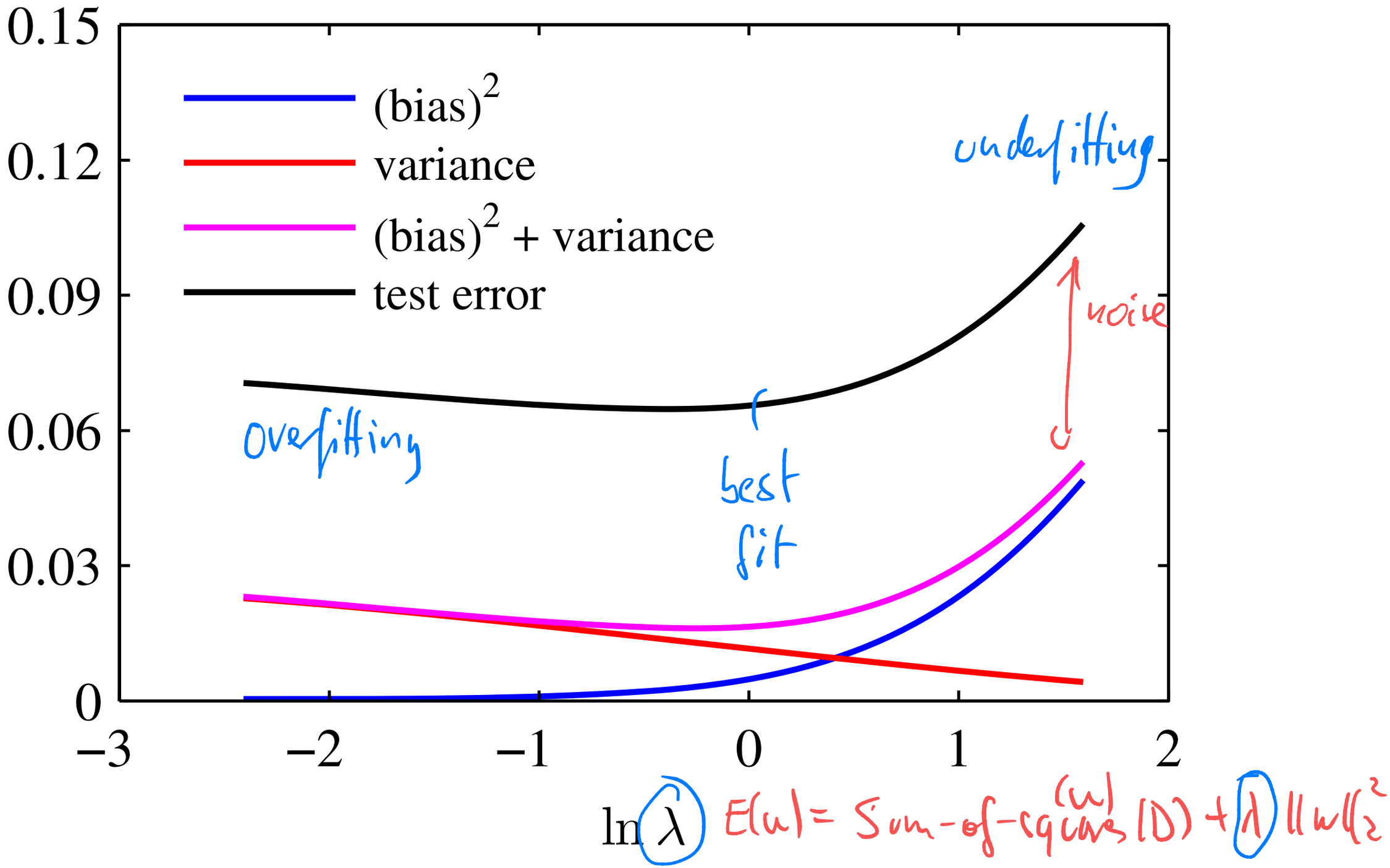


Figure: bias-variance decomposition (Bishop 3.6)

Overview

1. Bayesian linear regression
2. Generalization error decomposition
3. **Classification and decision theory**
4. Linear Discriminant Analysis (LDA)

supervised
↳ targets discrete

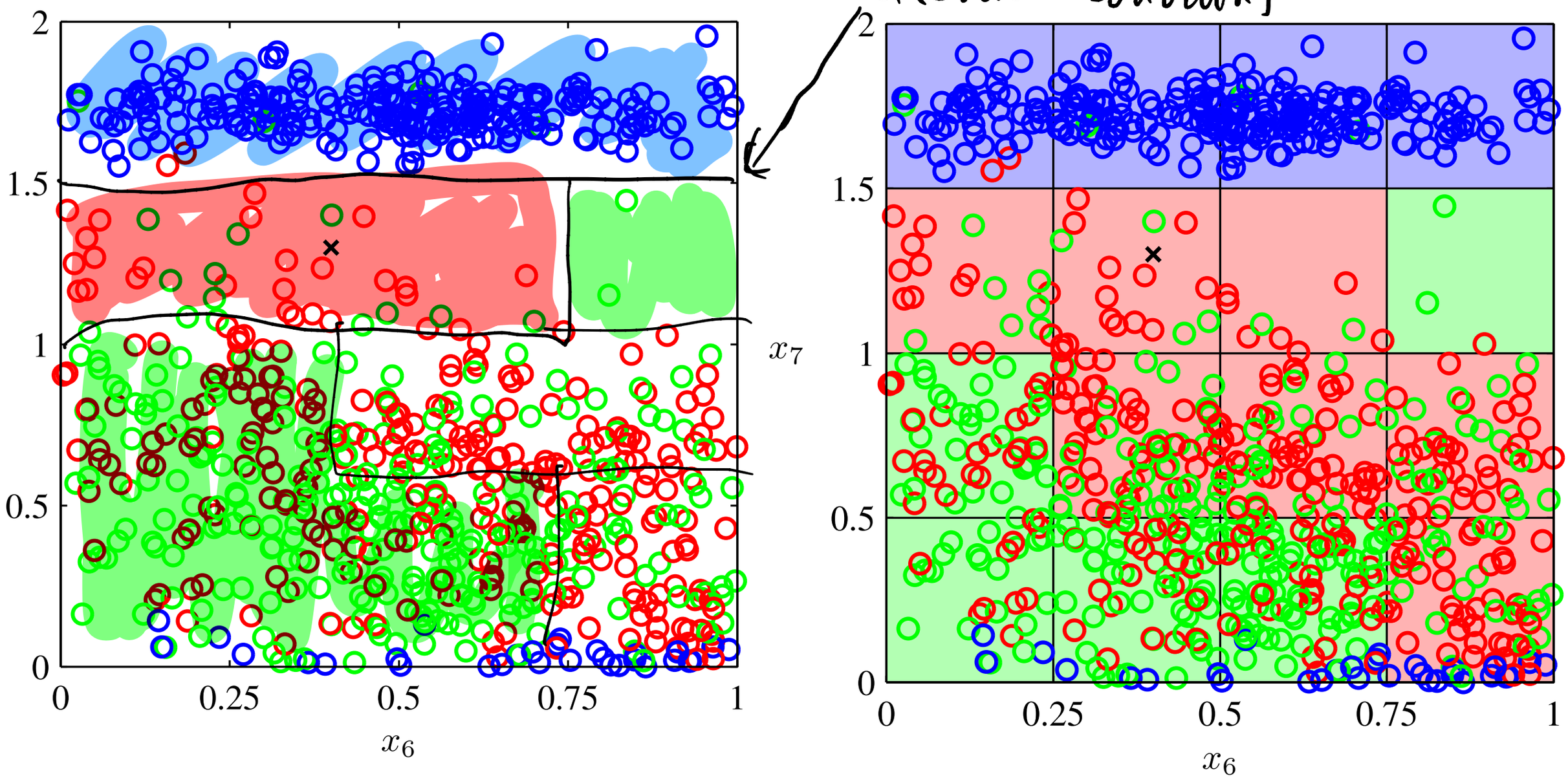
Classification through decision regions

- Input: $\mathbf{x} = (x_1, ..., x_D)^T$ $x \in \mathbb{R}^D$
- Target: $t \in \{C_1, .., C_K\}$ K classes
  - 2-class targets: $\{0, 1\}$
  - Multi-class targets $\{C_1, .., C_K\}$

Strategy:

- Divide input space $\mathbb{R}^D$ into K *decision regions*.
- Assign each decision region to a class
- Boundaries of decision regions are called *decision boundaries/surfaces.*

Classification through Decision Regions



Figures: 3 class problem with decision boundaries. (Bishop 1.19 & 1.20)

Linear Classification

- Linear Classification: consider only *linear* decision boundaries
- For D - dimensional input space:
  decision surface is a $(D-1)$-dimensional hyperplane
- Datasets whose classes can be separated *exactly* by linear decision surfaces are called separable

D-dim

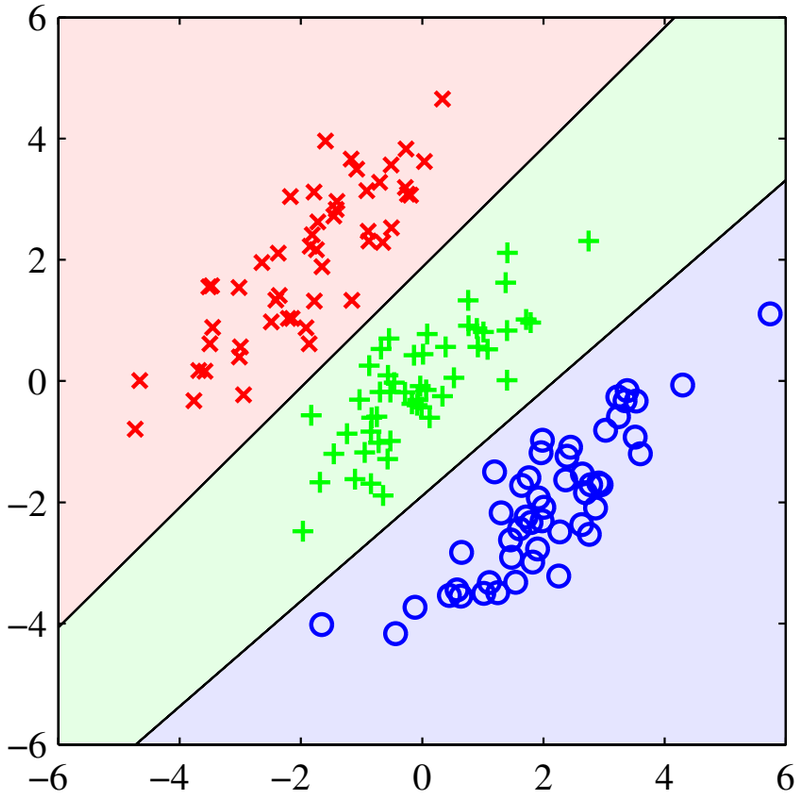


Figure: Linearly separable dataset (Bishop 4.5)

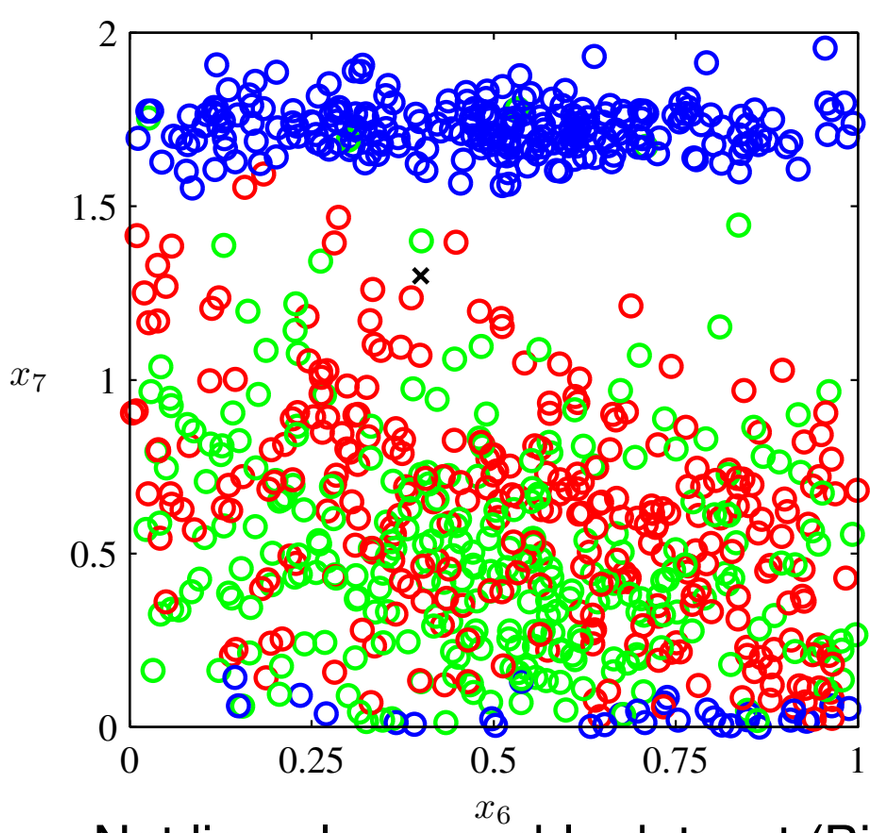


Figure: Not linearly separable dataset (Bishop 1.19)

Multiple Classes ($K > 2$)

- K=2 classes:
  - 1 classifier determines
- Multiple classes: $K > 2$
  - K-1 classifiers:
- One-versus-the-rest

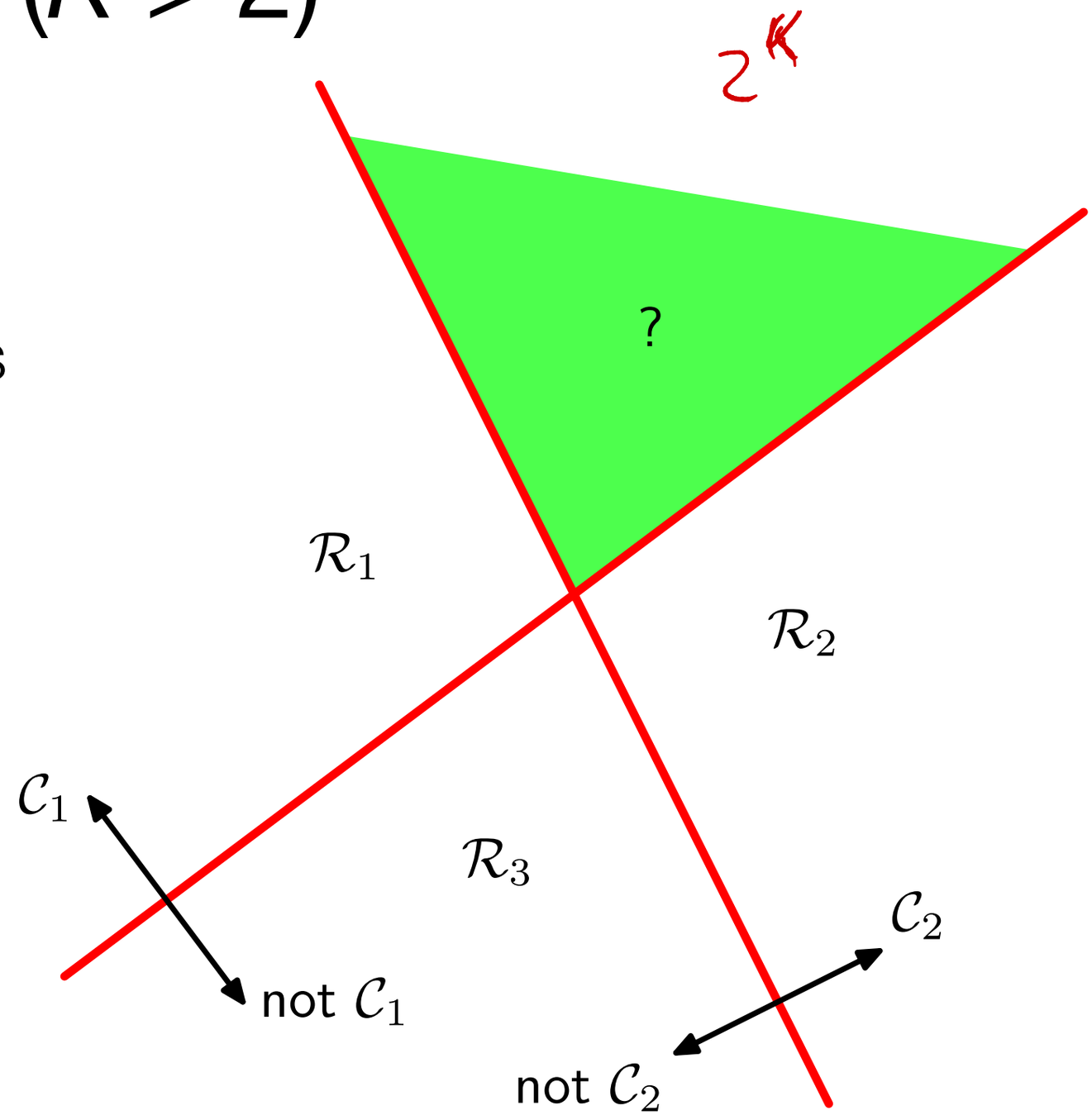
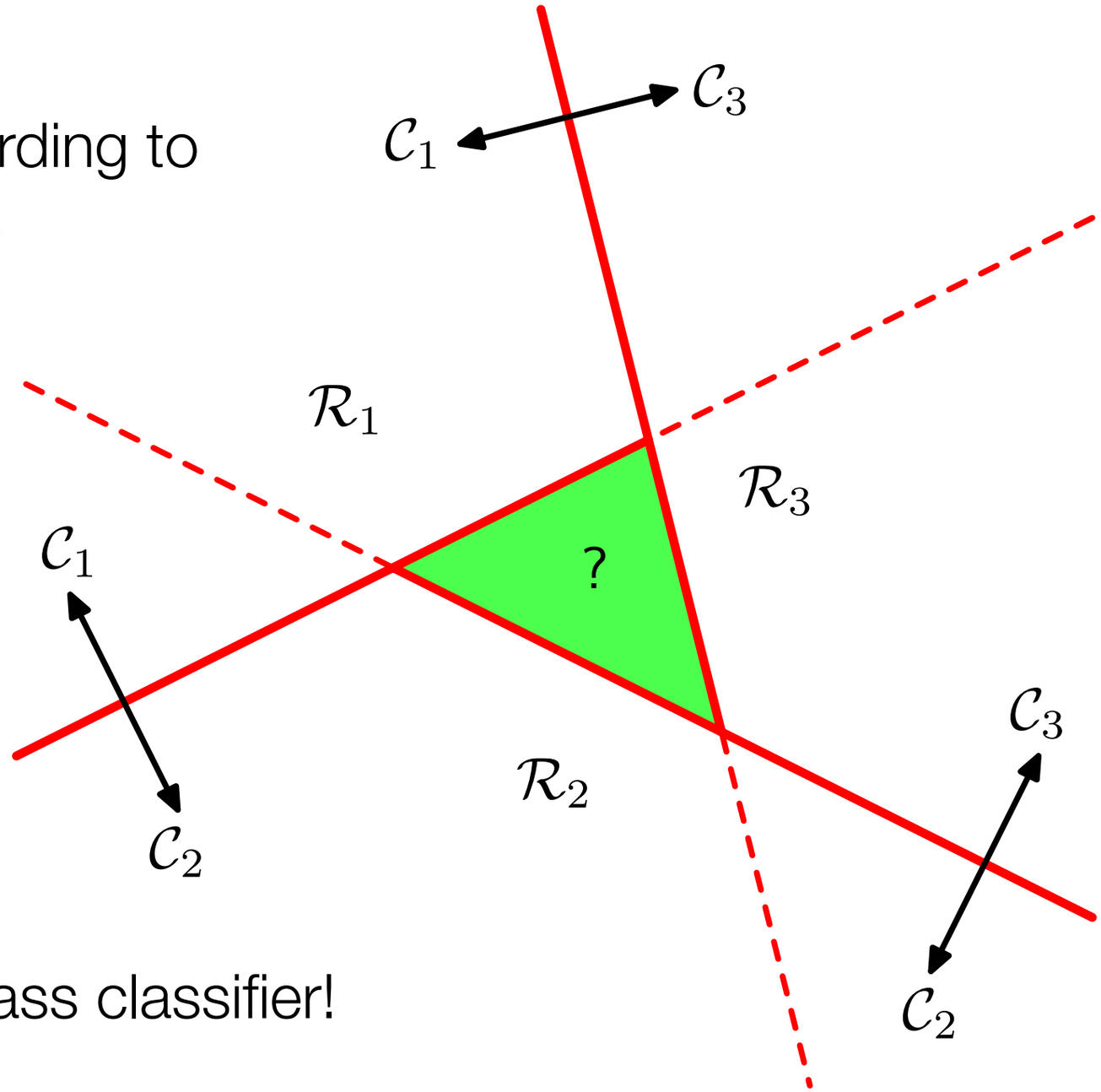


Figure: one-versus-the-rest classifiers (Bishop 4.2)

Multiple Classes ($K > 2$)

- K(K-1)/2 classifiers:
- Points are classified according to majority vote of classifiers
- one-versus-one

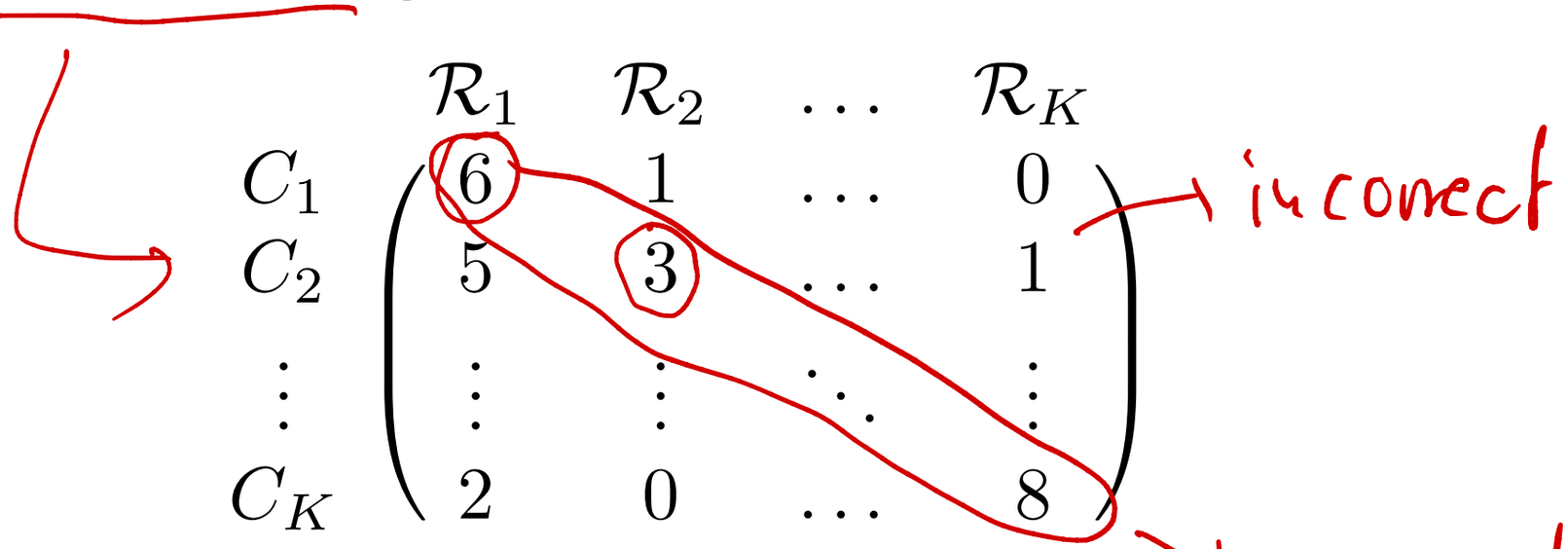


- **Solution**: Make one K-class classifier! (See later)

Figure: one-versus-one classifiers (Bishop 4.2)

Decision theory

- Dataset: Input vectors $\mathbf{x}$, ground truth targets $t \in \{C_1, \ldots, C_K\}$
- Divide input space into K decision regions $\mathcal{R}_k$
- Every observed datapoint
- Confusion matrix: ground truth classes vs. predicted classes



- Diagonal elements: correctly classified
- Off-diagonal elements: misclassified

Decision theory: Misclassification Rate

- Classification goal: Minimize the misclassification rate
- Assume observations are drawn from joint distribution
- Probability of a misclassification:

$$p(\text{mistake}) = \sum_{i=1}^{K} \sum_{k \neq i} p(\mathbf{x} \in R_i, C_k)$$

$$= 1 - \sum_{k=1}^{K} p(\mathbf{x} \in R_k, C_k)$$

Minimizing misclassification rate

- Assign x to class C_k if $p(x, C_k) \geq p(x, C_j) \quad \forall j \neq k$
- Note: $p(x, C_k) = p(C_k|x)p(x)$

$p(C_k|x) \geq p(C_j|x)$

Decision theory: Misclassification Rate

$\hat{x}$: Decision boundary

x_0 : Optimal Decision boundary

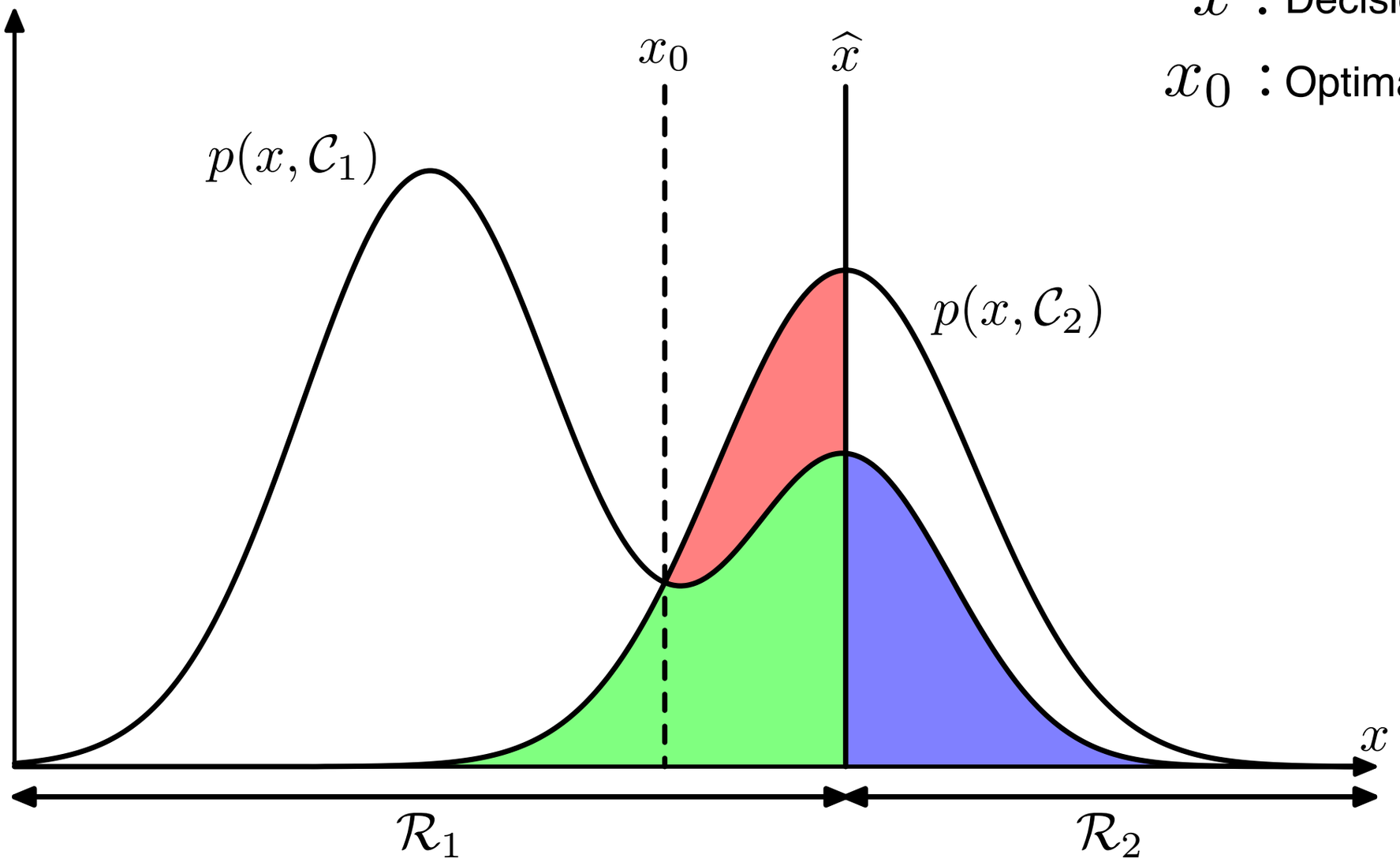


Figure: joint probability distributions and decision boundary (Bishop 1.24)

Minimizing the Misclassification Rate: Problems

- Not all errors have the same impact!

Example: Medical diagnosis of cancer

- Error 1: Label a healthy person as having cancer.

- Error 2: Label a sick person as healthy. Lack of treatment!

- If cancer only occurs in 1% of all patients, a classifier which labels everyone as healthy has a misclassification rate of 1%!

Expected Loss

- Possible solution: use different weights for different error types

$$L = \begin{array}{c} \begin{array}{cc} \text{label cancer} & \text{label healthy} \end{array} \\ \begin{pmatrix} 0 & 1000 \\ 1 & 0 \end{pmatrix} \end{array} \begin{array}{c} \text{true cancer} \\ \text{true healthy} \end{array}$$

- Expected loss: $\mathbb{E}[L] = \sum_{k,j} L_{kj} \int_{\mathcal{R}_j} p(x, C_k) \mathrm{d}x$

Minimize expected loss:

- Assign *x* to C_k if $\sum_{j=1}^{K} L_{jk} p(x, C_j)$ is minimal