Lecture 6: Model Selection, Multivariate Classification and Dimensionality Reduction

STAT261: Introduction to Machine Learning

Lecture 6, April 13

Outline: Multivariate Classification and Dimensionality Reduction

- Multivariate Data and Multiple Measurements
  - Multivariate Parameters
  - Multivariate Gaussian
- Parametric Classification
  - Different Covariances
  - Quadratic Discriminant
- Dimensionality Reduction

Multivariate Data

- Multiple measurements (sensors)
- *d* inputs/features/attributes: *d*-variate
- *N* instances/observations/examples

$$\mathbf{X} = \begin{bmatrix} X_1^1 & X_2^1 & \cdots & X_d^1 \\ X_1^2 & X_2^2 & \cdots & X_d^2 \\ \vdots & & & \\ X_1^N & X_2^N & \cdots & X_d^N \end{bmatrix}$$

Multivariate Parameters

$$\text{Mean}: E[\mathbf{x}] = \boldsymbol{\mu} = [\mu_1, \ldots, \mu_d]^T$$

$$\text{Covariance}: \sigma_{ij} \equiv \text{Cov}(X_i, X_j)$$

$$\text{Correlation}: \text{Corr}(X_i, X_j) \equiv \rho_{ij} = \frac{\sigma_{ij}}{\sigma_i \sigma_j}$$

$$\Sigma \equiv \text{Cov}(\mathbf{X}) = E\left[(\mathbf{X} - \mu)(\mathbf{X} - \mu)^T\right] = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1d} \\ \sigma_{21} & \sigma_2^2 & \cdots & \sigma_{2d} \\ \vdots & & & \\ \sigma_{d1} & \sigma_{d2} & \cdots & \sigma_d^2 \end{bmatrix}$$

Parameter Estimation

$$\text{Sample mean } \mathbf{m} : m_i = \frac{\sum_{t=1}^{N} x_i^t}{N}, i = 1,\ldots,d$$

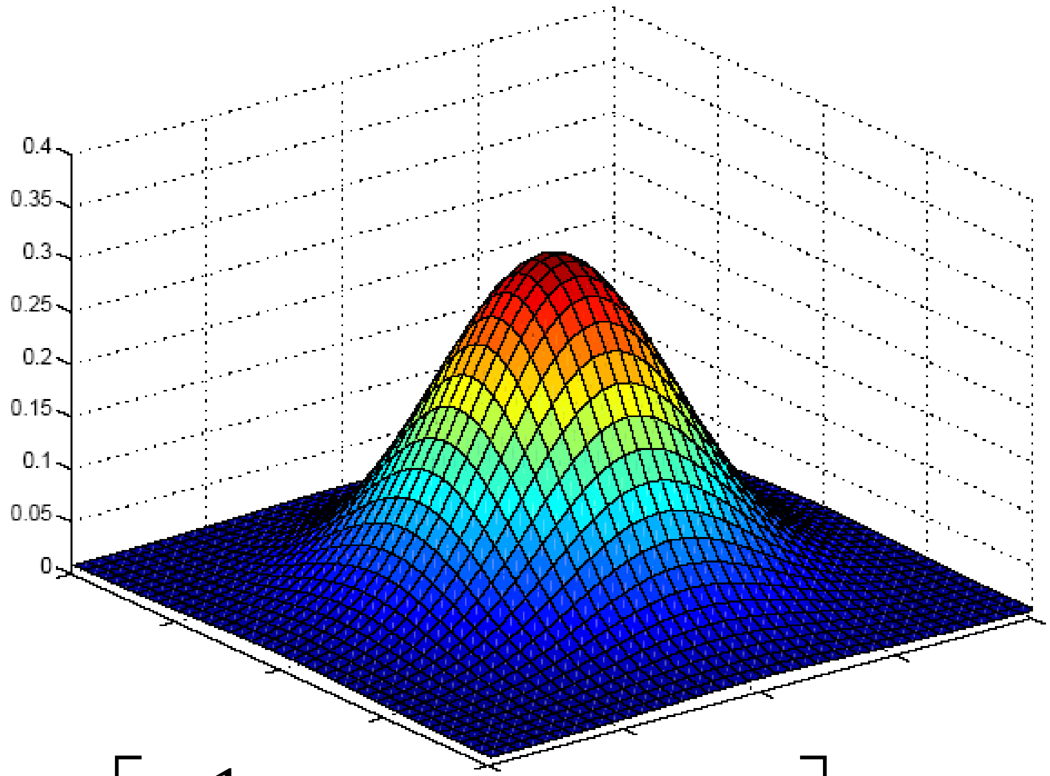
$$\text{Covariance matrix } \mathbf{S} : s_{ij} = \frac{\sum_{t=1}^{N} \left(x_i^t - m_i\right)\left(x_j^t - m_j\right)}{N}$$

$$\text{Correlation matrix } \mathbf{R} : r_{ij} = \frac{s_{ij}}{s_i s_j}$$

Estimation of Missing Values

- What to do if certain instances have missing attributes?
- Ignore those instances: not a good idea if the sample is small
- Use ‘missing’ as an attribute: may give information
- Imputation: Fill in the missing value
  - Mean imputation: Use the most likely value (e.g., mean)
  - Imputation by regression: Predict based on other attributes

Multivariate Normal Distribution



$$\mathbf{x} \sim \mathcal{N}_d(\boldsymbol{\mu}, \Sigma)$$

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

Multivariate Normal Distribution

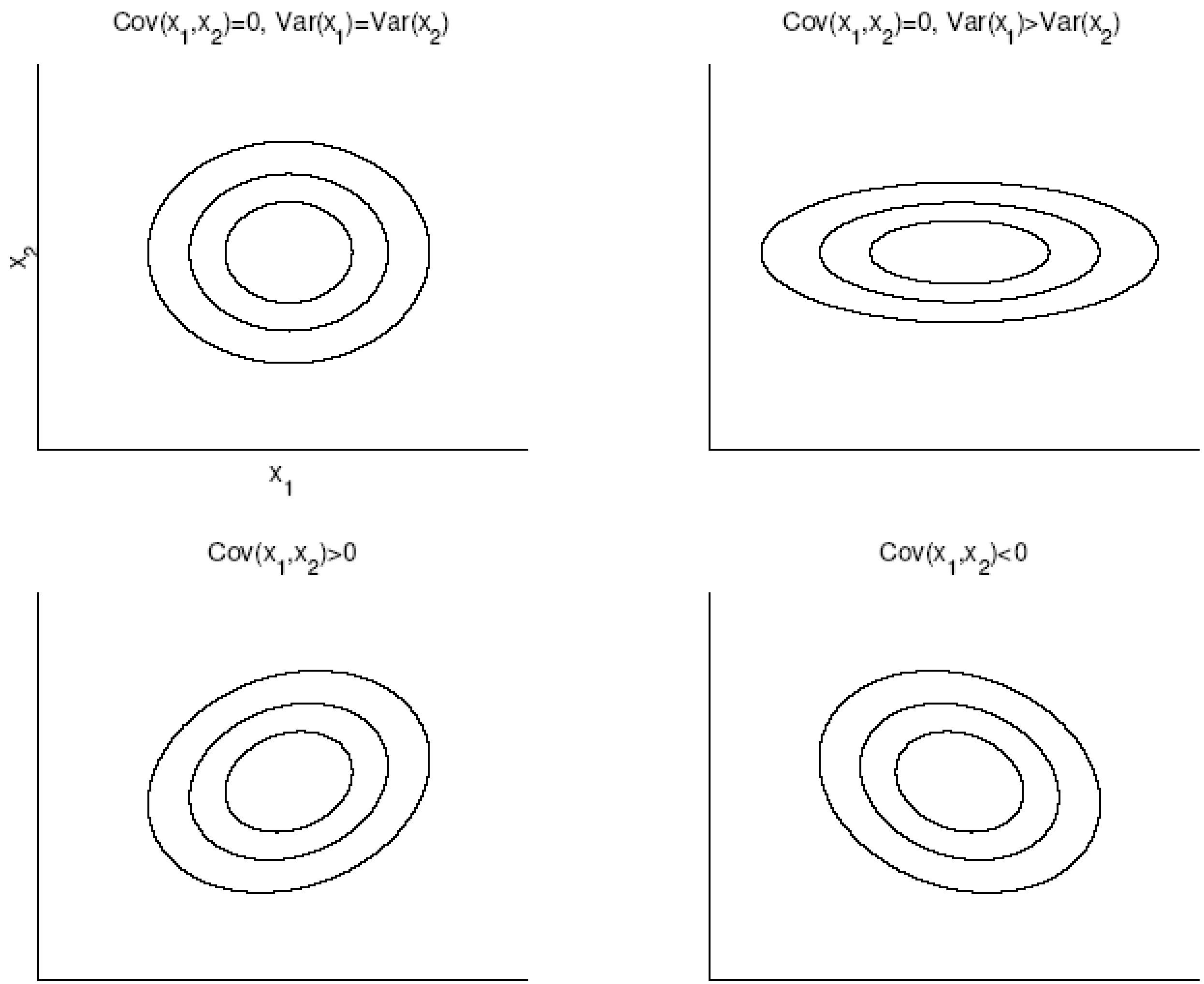
- Mahalanobis distance: $(\boldsymbol{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{x} - \boldsymbol{\mu})$
  measures the distance from $\boldsymbol{x}$ to $\boldsymbol{\mu}$ in terms of Σ
  (normalizes for difference in variances and correlations)
- Bivariate: $d = 2$

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix}$$

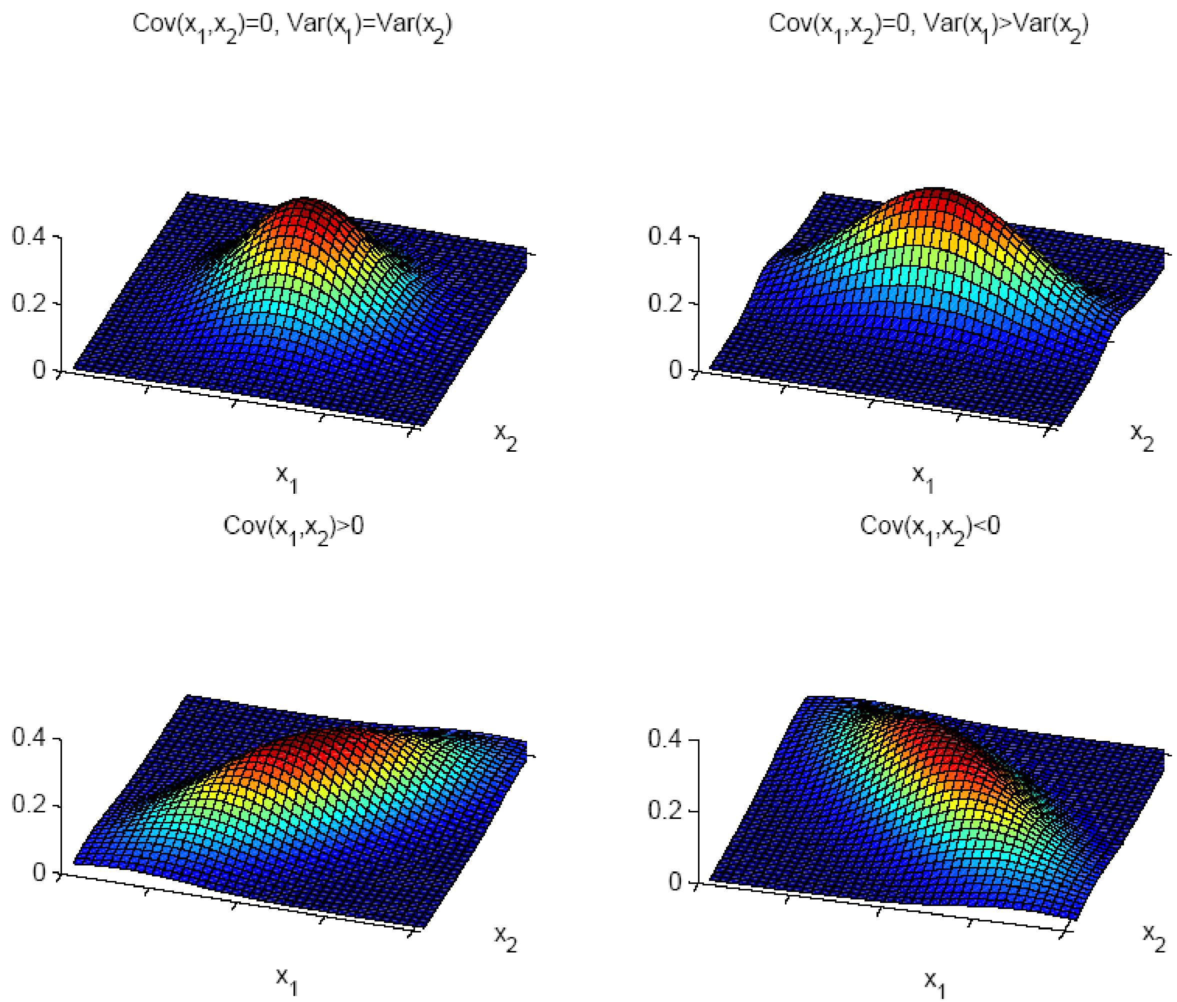
$$p(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \exp\left[-\frac{1}{2(1-\rho^2)} \left(z_1^2 - 2\rho z_1 z_2 + z_2^2 \right) \right]$$

$$z_i = (x_i - \mu_i) / \sigma_i$$

Bivariate Normal



Bivariate Normal



Independent Inputs: Naive Bayes

- If x_i are independent, off diagonals of Σ are 0, Mahalanobis distance reduces to weighted (by $1/\sigma_i$) Euclidean distance:

$$p(\mathbf{x}) = \prod_{i=1}^{d} p_i(x_i) = \frac{1}{(2\pi)^{d/2} \prod_{i=1}^{d} \sigma_i} \exp\left[-\frac{1}{2} \sum_{i=1}^{d} \left(\frac{x_i - \mu_i}{\sigma_i} \right)^2 \right]$$

- If variances are also equal, reduces to Euclidean distance

Parametric Classification

Classification: Maximum likelihood

Pick class that would make observation most likely

- If $p(x \mid C_i) \sim N(\mu_i, \Sigma_i)$

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

- Discriminant functions

$$g_i(\mathbf{x}) = \log p(\mathbf{x} \mid C_i) + \log P(C_i)$$

$$= -\frac{d}{2}\log 2\pi - \frac{1}{2}\log|\Sigma_i| - \frac{1}{2}(\mathbf{x}-\mu_i)^T \Sigma_i^{-1} (\mathbf{x}-\mu_i) + \log P(C_i)$$

Estimation of Parameters

$$\hat{P}(C_i) = \frac{\sum_t r_i^t}{N}$$

$$\mathbf{m}_i = \frac{\sum_t r_i^t \mathbf{x}^t}{\sum_t r_i^t}$$

$$\mathbf{S}_i = \frac{\sum_t r_i^t (\mathbf{x}^t - \mathbf{m}_i)(\mathbf{x}^t - \mathbf{m}_i)^T}{\sum_t r_i^t}$$

$$g_i(\mathbf{x}) = -\frac{1}{2}\log|\mathbf{S}_i| - \frac{1}{2}(\mathbf{x} - \mathbf{m}_i)^T \mathbf{S}_i^{-1}(\mathbf{x} - \mathbf{m}_i) + \log \hat{P}(C_i)$$

Parametric Classification: Different Variances

- Different $\mathbf{S}_i$
- Quadratic discriminant

$$g_i(\mathbf{x}) = -\frac{1}{2}\log|\mathbf{S}_i| - \frac{1}{2}\left(\mathbf{x}^T\mathbf{S}_i^{-1}\mathbf{x} - 2\mathbf{x}^T\mathbf{S}_i^{-1}\mathbf{m}_i + \mathbf{m}_i^T\mathbf{S}_i^{-1}\mathbf{m}_i\right) + \log\hat{P}(C_i)$$

$$= \mathbf{x}^T\mathbf{W}_i\mathbf{x} + \mathbf{w}_i^T\mathbf{x} + w_{i0}$$

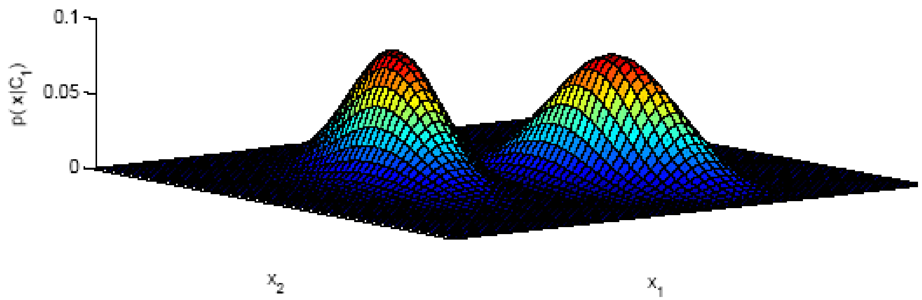
where

$$\mathbf{W}_i = -\frac{1}{2}\mathbf{S}_i^{-1}$$

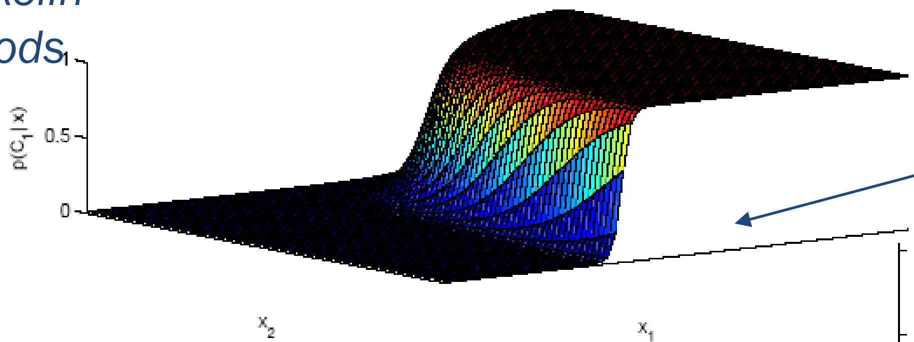
$$\mathbf{w}_i = \mathbf{S}_i^{-1}\mathbf{m}_i$$

$$w_{i0} = -\frac{1}{2}\mathbf{m}_i^T\mathbf{S}_i^{-1}\mathbf{m}_i - \frac{1}{2}\log|\mathbf{S}_i| + \log\hat{P}(C_i)$$

Parametric Classification: Different Variances

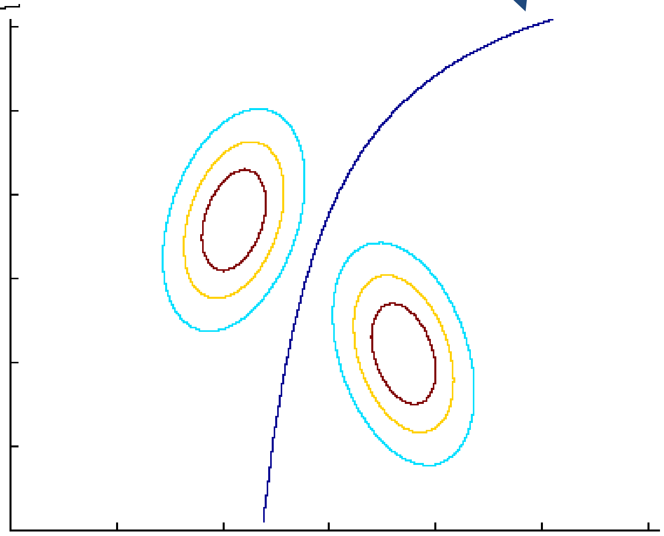


likelih
oods



posterior
for C_1

discriminant:
$P\,(C_1\,|\,\boldsymbol{x}\,) = 0.5$



Common Covariance Matrix $\mathbf{S}$

- Shared common sample covariance **S**

$$\mathbf{S} = \sum_i \hat{P}(C_i)\mathbf{S}_i$$

- Discriminant reduces to

$$g_i(\boldsymbol{x}) = -\frac{1}{2}(\boldsymbol{x} - \boldsymbol{m}_i)^T \mathbf{S}^{-1}(\boldsymbol{x} - \boldsymbol{m}_i) + \log \hat{P}(C_i)$$

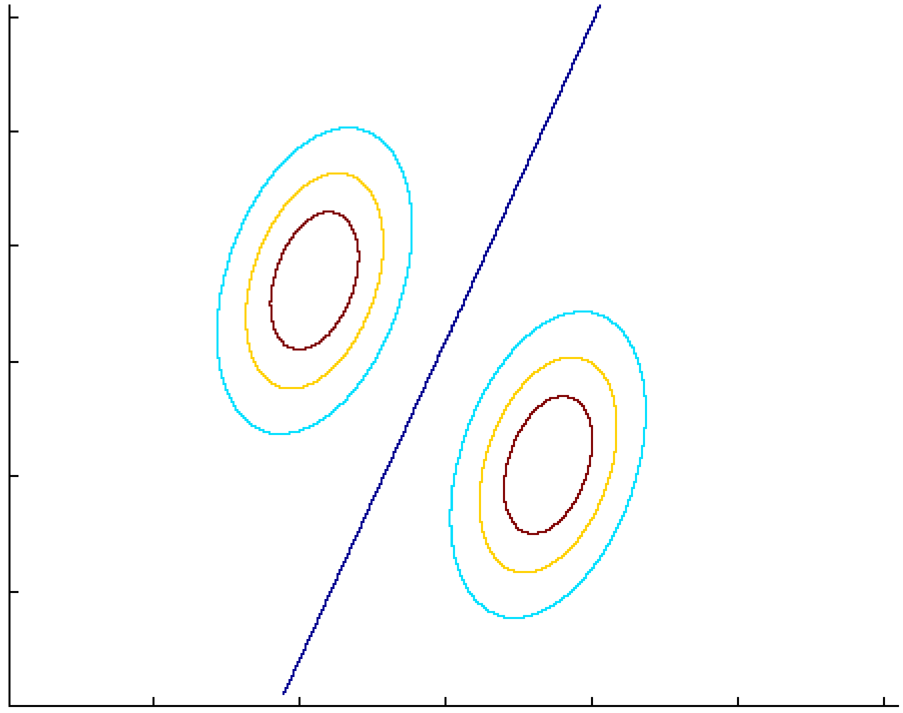
which is a linear discriminant

$$g_i(\mathbf{x}) = \mathbf{w}_i^T \mathbf{x} + w_{i0}$$

where

$$\mathbf{w}_i = \mathbf{S}^{-1}\mathbf{m}_i \quad w_{i0} = -\frac{1}{2}\mathbf{m}_i^T \mathbf{S}^{-1}\mathbf{m}_i + \log \hat{P}(C_i)$$

Common Covariance Matrix **S**



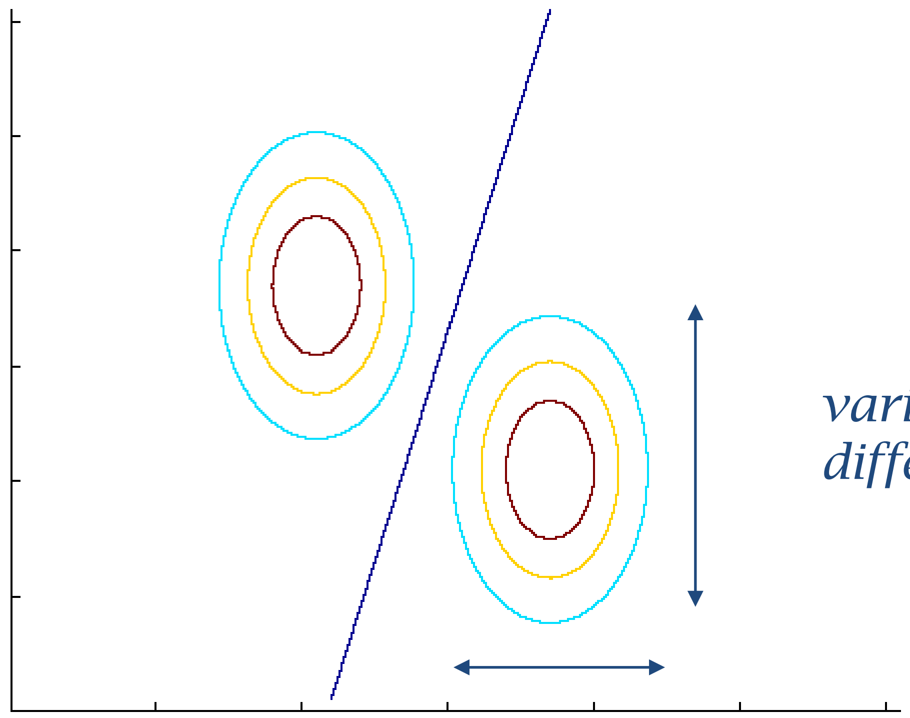
Classification: Diagonal $\mathbf{S}$'s

- When x_j $j = 1,..d$, are independent,
- Σ is diagonal
- $p(\mathbf{x}|C_i) = \prod_j p(x_j|C_i)$ (Naive Bayes' assumption)

$$g_i(\mathbf{x}) = -\frac{1}{2}\sum_{j=1}^{d}\left(\frac{x_j^t - m_{ij}}{s_j}\right)^2 + \log \hat{P}(C_i)$$

- Classify based on weighted Euclidean distance (in s_j units) to the nearest mean

Classification: Diagonal S matrics



variances may be different

Classification: Diagonal S, equal variances

- Nearest mean classifier:

Classify based on Euclidean distance to the nearest mean

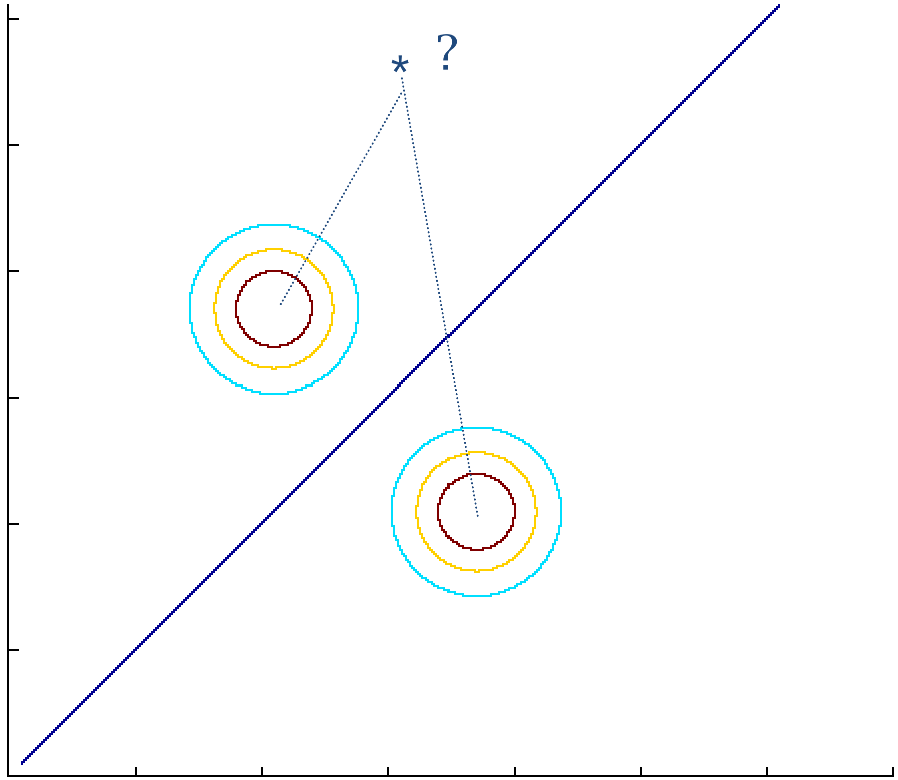
$$g_i(\mathbf{x}) = -\frac{\|\mathbf{x}-\mathbf{m}_i\|^2}{2s^2} + \log \hat{P}(C_i)$$

$$= -\frac{1}{2s^2}\sum_{j=1}^{d}\left(x_j^t - m_{ij}\right)^2 + \log \hat{P}(C_i)$$

- Each mean can be considered a prototype or template

This is sometimes called template matching

Diagonal **S**, equal variances

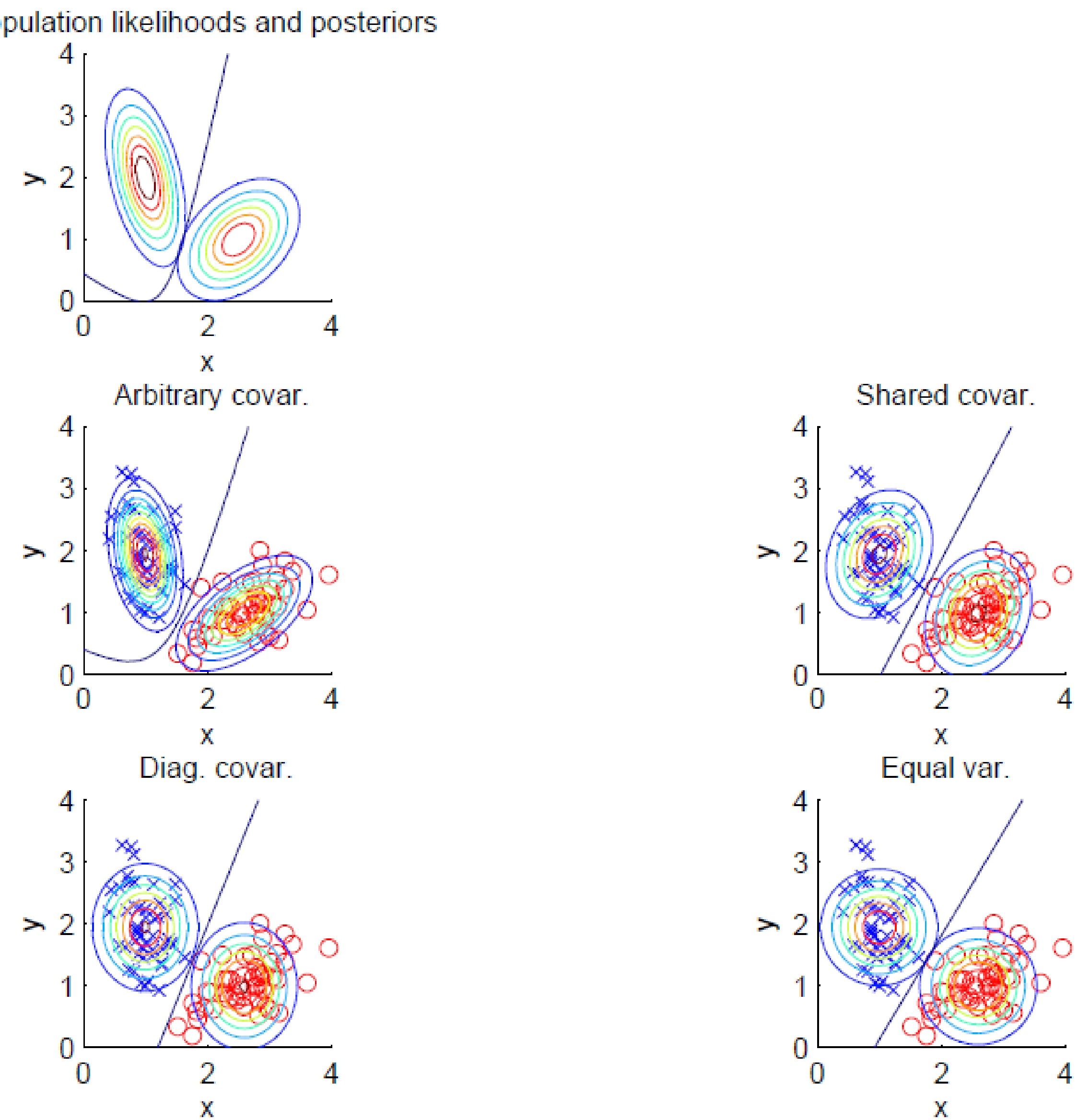


Model Selection

Assumption	*Covariance matrix*	*No of parameters*
Shared, Hyperspheric	$\mathbf{S}_i=\mathbf{S}=s^2\mathbf{I}$	1
Shared, Axis-aligned	$\mathbf{S}_i=\mathbf{S}$, with $s_{ij}=0$	d
Shared, Hyperellipsoidal	$\mathbf{S}_i=\mathbf{S}$	$d(d+1)/2$
Different, Hyperellipsoidal	$\mathbf{S}_i$	$K\, d(d+1)/2$

- As we increase complexity (less restricted **S**), bias decreases and variance increases
- Assume simple models (allow some bias) to control variance (regularization)

Population Likelihoods and Posteriors



Classification: Discrete Features

- Binary features: $p_{ij} \equiv p(x_j = 1 | C_i)$

  if x_j are independent (Naive Bayes')

$$p(x | C_i) = \prod_{j=1}^{d} p_{ij}^{x_j} (1 - p_{ij})^{(1 - x_j)}$$

  the discriminant is linear

$$g_i(\mathbf{x}) = \log p(\mathbf{x} | C_i) + \log P(C_i)$$
$$= \sum_j \left[x_j \log p_{ij} + (1 - x_j) \log (1 - p_{ij}) \right] + \log P(C_i)$$

  Estimated parameters $\hat{p}_{ij} = \dfrac{\sum_t x_j^t r_i^t}{\sum_t r_i^t}$

Discrete Features

- Multinomial (1-of-n_j) features: $x_j \in \{v_1, v_2, \ldots, v_{n_j}\}$

$$p_{ijk} \equiv p(z_{jk}=1 \mid C_i) = p(x_j = v_k \mid C_i)$$

if x_j are independent

$$p(\mathbf{x} \mid C_i) = \prod_{j=1}^{d} \prod_{k=1}^{n_j} p_{ijk}^{z_{jk}}$$

$$g_i(\mathbf{x}) = \sum_j \sum_k z_{jk} \log p_{ijk} + \log P(C_i)$$

$$\hat{p}_{ijk} = \frac{\sum_t z_{jk}^t r_i^t}{\sum_t r_i^t}$$

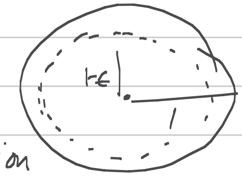
The Curse of Dimensionality

The examples of Bayes Decision theory are misleading because they are given in low-dimensional spaces (1-dim, or 2-dim)

Many pattern classification tasks occur in high dimensional spaces. In these spaces our geometric intuitions are often wrong.

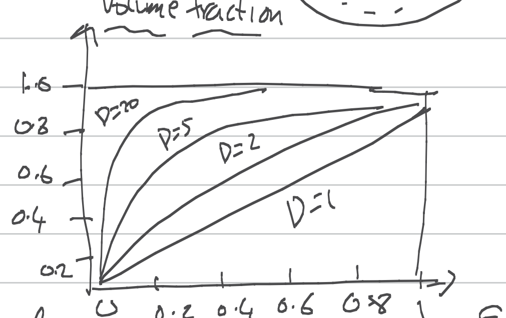
EG. Consider the volume of a sphere of radius $r=1$ in D dimension.

What fraction of its volume lies in the region between $1-\epsilon < r < 1$?



$V_D(r) = K_D r^D$

$$\frac{V_D(1) - V_D(1-\epsilon)}{V_D(1)} = 1 - (1-\epsilon)^D$$

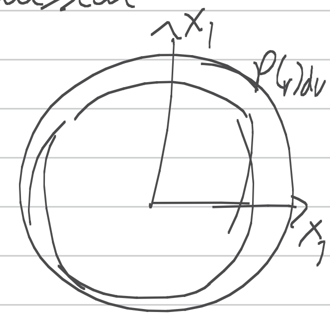


For large D, the volume fraction tends to 1 even for small ϵ.

Most of the volume is at the boundary!

(7)

E.6. Behaviour of a Gaussian distribution.



In 1-D, $p(x) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-x^2/2\sigma^2}$

In 2-D

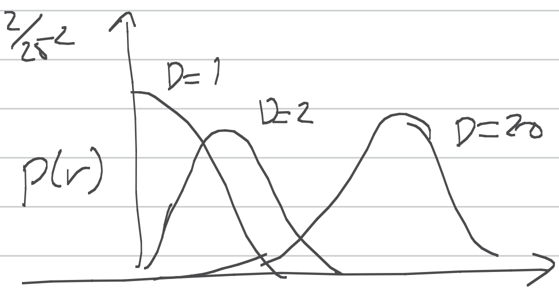
$$p(x_1, x_2) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x_1^2+x_2^2)}{2\sigma^2}}.$$

Let $r = \sqrt{x_1^2+x_2^2}$.

Then $$p(r) = \frac{r}{2\pi\sigma^2} e^{-r^2/2\sigma^2}$$

In higher dimensions

$$p(r) = \frac{r^{D-1}}{K} e^{-r^2/2\sigma^2}$$



So in high dimensions most of the probability mass of the Gaussian is concentrated on a thin shell away from the center of the Gaussian.

(8)

Learning probabilities distributions in high dimensions can require a lot of data

E.G. Gaussian Distribution in D dimension.

mean $-\underline{\mu}$ D dimension.

covariance $-\underline{\underline{\Sigma}}$ $\frac{D(D+1)}{2}$ Dimension.

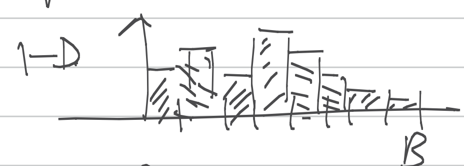
This is $O(D^2)$, not too bad.

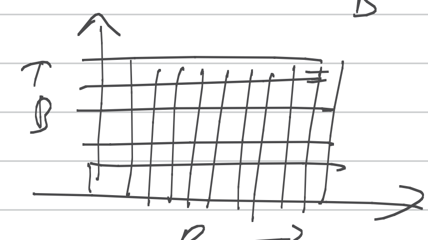
But suppose we represent the data by a histogram with B bins per dimension.

B bins in $D=1$

B^2 bins in $D=2$

B^D bins in D dimension.





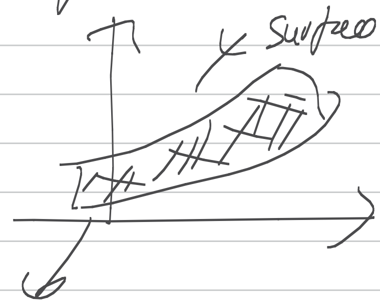
Exponential growth!

Requires exponential amount of data to learn the distribution.

(9)

How to deal with the curse of dimensionality?

In practice, data typically lies on some low-dimensional surface in the high dimensional space.



So the effective dimension of the data may be a lot smaller than the dimension of the space.

(*) Dimension Reduction Methods attempt to reduce the dimension by seeking this low dimensional surface. (Not always easy).

(*) Modeling, if we can guess distribution for the data (e.g. Gaussian) then the dependence on the dimension is not too bad.

(*) Concentrate on the Decision Boundary ~ there may be enough data to learn the decision boundary even if we cannot learn the distributions.

(10) <u>Bias and Variance</u>

This is a classical statistics perspective on generalization. First we need to introduce some statistics terminology.

Suppose we want to estimate a continuous quantity θ — e.g. the mean/variance of a Gaussian distribution (more about this in the next lecture), or the parameter of a regression line (see below) — then statisticians use an <u>estimator</u>.

The estimator is based on a set $\mathcal{X}_N = \langle \underline{x}_i : i = 1 \text{ to } N \rangle$ of examples — drawn from an unknown distribution $P(\underline{x})$.

<u>i.i.d.</u> $P(\mathcal{X}_N) = \prod_{i=1}^{N} P(\underline{x}_i)$

The task is to estimate a property θ by an estimator $\hat{\theta} = g(\mathcal{X}_N)$. Eg. like a classification rule — but based on the set $\mathcal{X}_N$ and θ is continuous.

<u>For example</u>: let $\theta = (\mu, \sigma)$ be the mean and variance of the data (data is one-dimensional in this example)

then $\mathcal{X}_N = \langle x_i : i = 1 \text{ to } N \rangle$

$$\hat{\mu}(\mathcal{X}_N) = \frac{1}{N}\sum_{i=1}^{N} x_i, \quad \hat{\sigma}^2(\mathcal{X}_N) = \frac{1}{N}\sum_{i=1}^{N}\{x_i - \hat{\mu}(\mathcal{X}_N)\}^2$$

$$g(\mathcal{X}_N) = (\hat{\mu}(\mathcal{X}_N), \hat{\sigma}^2(\mathcal{X}_N))$$

<u>Note</u>: that the estimator is a function of the set $\mathcal{X}_N$, this will be important later.

(11)

To evaluate the estimator, we want to measure how much it differs from the correct θ. It is attractive to use a quadratic errors (this helps the analysis)

$$(g(X_N) - \theta)^2$$ — but this depends on the data set X_N

So we need to get the expected error with respect to the set X_N,

$$r(g, \theta) = E_X[g(X) - \theta]^2 = \int (g(X) - \theta)^2 P(X) dX.$$

mean square error.

distribution over set $X = \{x_i\}$.

$$b_\theta(g) = E_X[g(X)] - \theta,$$ bias of estimator

If $b_\theta(g) = 0$ for all θ, then $g(.)$ is an unbiased estimator of θ

E.g. consider $\hat{\mu}(X) = \frac{1}{N} \sum_{i=1}^{N} x_i$, estimate of the mean

$$E_X[\hat{\mu}(X)] = E_X\left[\frac{1}{N} \sum_{i=1}^{N} x_i\right] = \frac{1}{N} \sum_{i=1}^{N} E_{x_i}(x_i)$$

$$= \frac{1}{N} N\mu = \mu$$

$\sum_{x_i} x_i P(x_i) = \mu$

mean of the distribution that generated the data.

Hence $\hat{\mu}(.)$ is an unbiased estimator of μ.

($\hat{\mu}(X)$ will depend on X, but on average it gives you the right answer).

We can compute the variance of the estimator — i.e. how much it varies depending on X.

(12) Generating Process $P(\mathcal{X})$

$P(\mathcal{X})$ can generate many possible datasets $\mathcal{X}_1, \mathcal{X}_2, \mathcal{X}_3 \ldots$

$\mathcal{X}_1 = \{x_1 \ldots x_N\}$ $\quad \mathcal{X}_2 = \{x_{N+1}, \ldots, x_{2N}\}$ $\quad \mathcal{X}_3 = \{x_{2N+1}, \ldots, x_{3N}\}$

$g(\mathcal{X}_1)$, $g(\mathcal{X}_2)$, $g(\mathcal{X}_3)$, these estimators will vary

Calculate their mean $\Rightarrow$ e.g. $E_{\mathcal{X}}(g(\mathcal{X}))$

Calculate the variance $\Rightarrow Var(g(\mathcal{X})) = E_{\mathcal{X}}\{(g(\mathcal{X}) - E_{\mathcal{X}}\{g(\mathcal{X})\})^2\}$.

For the estimator $\hat{\mu}(\mathcal{X})$ of the mean, we know that $E_{\mathcal{X}}(\hat{\mu}(\mathcal{X})) = \mu$ unbiased (previous page)

$$Var_{\mathcal{X}}(\hat{\mu}) = Var_{\mathcal{X}}\left(\frac{\sum_{i=1}^{N} x_i}{N}\right) = \frac{1}{N^2}\sum_{i=1}^{N} Var_{x_i}(x_i)$$

$$Var_{x_i}(x_i) = \sum_{x_i} p(x_i)(x_i-\mu)^2$$

$$= \frac{N\sigma^2}{N} = \frac{\sigma^2}{N}$$

Hence, the variance of the estimator tends to zero as $N \to \infty$ with fall off rate $1/N$

Note: this $1/N$ fall-off rate is true for any linear estimator – i.e. $g(\mathcal{X})$ is a linear function of the elements $x_1 \ldots x_N$ in set $\mathcal{X}$.

Next, consider the bias of $\hat{\sigma}^2(\mathcal{X}) = \frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{\mu}_i)^2$

By similar analysis (to analysis of $\hat{\mu}(\mathcal{X})$) we find that

$$E_{\mathcal{X}}[\hat{\sigma}^2(\mathcal{X})] = \frac{1}{N}\left\{\sum_{i=1}^{N} E_{x_i}(x_i^2) - N\hat{\mu}^2\right\} = \frac{N-1}{N} E_x((x-\mu)^2) = \frac{N-1}{N}\sigma^2$$

(uses $\sum_{i=1}^{N}(x_i - \hat{\mu})^2 = \sum_{i=1}^{N} x_i^2 - \hat{\mu}^2$). biased, but asymptotically unbiased. as $N \to \infty$

(13) Bias Variance Dilemma:

Note Title 3/29/2008

Dataset $\mathcal{X} = \langle (x_i, y_i) : i = 1 \text{ to } N \rangle$,
sampled from $P(x,y) = p(y|x)p(x)$
y is continuous valued.

Let $g(x)$ be an estimator of y

Claim I: $\underbrace{\langle (y-g(x))^2 \rangle_{P(y|x)}}_{\text{Expected error w.r.t. } P(y|x)} = \underbrace{(y - \langle y \rangle_{P(y|x)})^2}_{\text{Variance of the process}} + \underbrace{(\langle y \rangle_{P(y|x)} - g(x))^2}_{\text{squared error}}$

→ Nothing we can do about this - indep of our estimator $g(\,)$

$$\boxed{\text{Here } \langle f(y,x) \rangle_{P(y|x)} = \sum_y f(y,x) P(y|x)}$$

Proof: write $(y-g(x))^2 = (y - \langle y \rangle_{P(y|x)} + \langle y \rangle_{P(y|x)} - g(x))^2$

$= (y - \langle y \rangle_{P(y|x)})^2 + (\langle y \rangle_{P(y|x)} - g(x))^2 + 2(y - \langle y \rangle_{P(y|x)})(\langle y \rangle_{P(y|x)} - g(x))$

Take expectation with respect to $P(y|x)$

Then $\langle (y-g(x))^2 \rangle_{P(y|x)} = \langle (y - \langle y \rangle_{P(y|x)})^2 \rangle + (\langle y \rangle_{P(y|x)} - g(x))^2$.

because $(\langle y \rangle_{P(y|x)} - g(x))^2$ is indep of y and $\langle (y - \langle y \rangle_{P(y|x)}) \rangle_{P(y|x)} = 0$

Claim I says we can decompose the expected error (w.r.t. $P(y|x)$) into part we have no control over $\langle (y - \langle y \rangle_{P(y|x)})^2 \rangle$ and part which depends on $g(\cdot)$ and the data $\mathcal{X}$.

$(\langle y \rangle_{P(y|x)} - g(x))^2$

(14) <u>Next</u> we study the expectation of $(\langle y\rangle_{P(y|x)} - g(\mathcal{X}))^2$ with respect to $P(\mathcal{X})$ i.e. how it depends on the particular sample $\mathcal{X} = (x_1, \dots x_N)$ from $P(\mathcal{X})$.

<u>Claim II</u> The Bias-Variance Result.

$$\langle (\langle y\rangle_{P(y|x)} - g(\mathcal{X}))^2\rangle_{P(\mathcal{X})} = \underbrace{(\langle y\rangle_{P(y|x)} - \langle g(\mathcal{X})\rangle_{P(\mathcal{X})})^2}_{\text{Bias}} + \underbrace{\langle (g(\mathcal{X}) - \langle g(\mathcal{X})\rangle_{P(\mathcal{X})})^2\rangle_{P(\mathcal{X})}}_{\text{Variance}}$$

The first term depends on the <u>bias</u> – the difference between the best estimate $\langle y\rangle_{P(y|x)}$ (if we knew the distribution) and the expectation of our estimator $\langle g(\mathcal{X})\rangle_{P(\mathcal{X})}$

The second term is the variance of the estimator $g(\mathcal{X})$ – i.e. how much it depends on the sample set $\mathcal{X}$ (more later)

<u>Proof</u>: Write $(\langle y\rangle_{P(y|x)} - g(\mathcal{X}))^2$

$$= (\langle y\rangle_{P(y|x)} - \langle g(\mathcal{X})\rangle_{P(\mathcal{X})} + \langle g(\mathcal{X})\rangle_{P(\mathcal{X})} - g(\mathcal{X}))^2$$

$$= (\langle y\rangle_{P(y|x)} - \langle g(\mathcal{X})\rangle_{P(\mathcal{X})})^2 + (\langle g(\mathcal{X})\rangle_{P(\mathcal{X})} - g(\mathcal{X}))^2$$
$$+ 2(\langle y\rangle_{P(y|x)} - \langle g(\mathcal{X})\rangle_{P(\mathcal{X})})(\langle g(\mathcal{X})\rangle_{P(\mathcal{X})} - g(\mathcal{X}))$$

Take expectation w.r.t. $P(\mathcal{X})$

$$\langle (\langle y\rangle_{P(y|x)} - g(\mathcal{X}))^2\rangle_{P(\mathcal{X})} = (\langle y\rangle_{P(y|x)} - \langle g(\mathcal{X})\rangle_{P(\mathcal{X})})^2 + \langle (g(\mathcal{X}) - \langle g(\mathcal{X})\rangle_{P(\mathcal{X})})^2\rangle_{P(\mathcal{X})}$$

Because $\langle y\rangle_{P(y|x)} - \langle g(\mathcal{X})\rangle_{P(\mathcal{X})}$ is independent of $\mathcal{X}$ and $\langle g(\mathcal{X}) - \langle g(\mathcal{X})\rangle_{P(\mathcal{X})}\rangle_{P(\mathcal{X})} = 0$

(15)

What does this mean?

Distribution $P(x)$

Dataset $\mathcal{X}_1 = \{x_1, \dots x_N\}$, $\mathcal{X}_2 = \{x_{N+1}, \dots x_{2N}\}$

$\dots$ $\mathcal{X}_m = \{x_{(m-1)N+1}, \dots x_{mN}\}$

For each data we get an estimate of y

$g(\mathcal{X}_1)$, $g(\mathcal{X}_2)$ .. $g(\mathcal{X}_m)$

The mean estimate is $\bar{g} = 1/m \sum_{i=1}^{m} g(\mathcal{X}_i)$

the variance is $Var(\bar{g}) = \frac{1}{2} \sum_{i=1}^{m} (g(\mathcal{X}_i) - \bar{g})^2$

To get good generalization we want the variance to be small, so that it isn't sensitive to the data we have trained the classifier on.

Ideally we want to have a classifier $g(.)$ which has small bias and variance.

In practice, there is often a trade-off between bias and variance.

A complex classifier can give a good fit to the data (compared to a simple classifier) but can have high variance because it over-fits the data. So it gives different results on different datasets.

(16)

Bias/Variance Dilemma

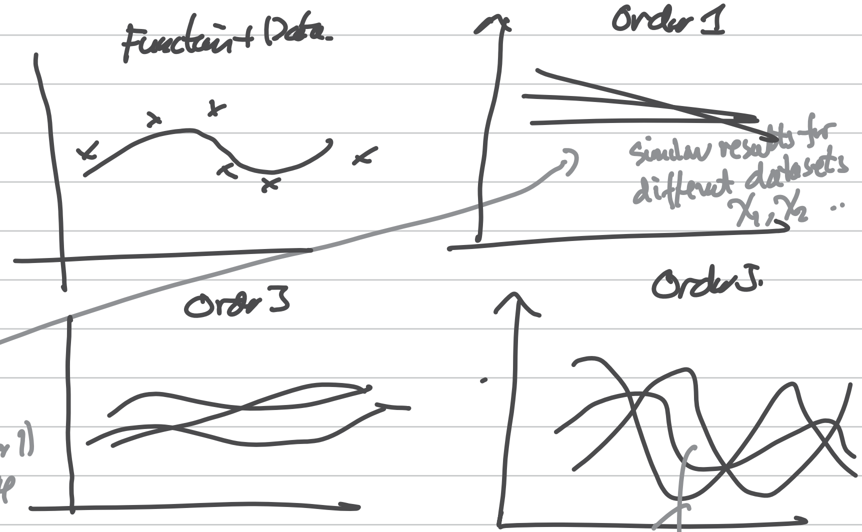
The data is generated by

$f(x) = 2\sin(1.5x)$

$\epsilon \sim N(0,1)$

Fit an order 1 polynomial (straight line) to the data.

Fit order 3 poly. Better fit (than order 1) but more variance



Best fit, High Variance

A more complex models gives better fit to the data (i.e. to underlying model)

→ reduces bias.

But small changes in dataset lead to bigger variation in fitted model

→ increases variance.

Low orders — risk of <u>underfitting</u>

High orders — risk of <u>overfitting</u>. (fit the noise, not the function)

To get a small error, we should have the proper inductive bias and have large enough dataset so that variability is constrained by data.

Note: take many high-variance models, use average (later)

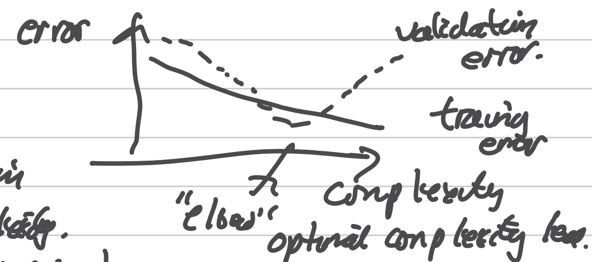
(17)

What to do? How to test generalization?

<u>Cross-validation</u> → divide dataset into two parts as training & validation set.

Train models of different complexity and test their error on the validation set.

As model complexity increases, error on training set decreases. But error on validation set decreases then increase.



<u>regularization</u>

augmented error function

$E' =$ error on data $+ \lambda \cdot$ model complexity.

(λ optimized using cross-validation)

<u>structural risk minimization</u> (Vapnik) (vc dim)

— also penalizes model complexity.

<u>Minimum Description Length</u> (Rissanen)

penalize complexity by cost of encoding model.

<u>Bayesian Model Selection</u>. if some prior knowledge

$$p(\text{model} \mid \text{data}) = \frac{p(\text{data} \mid \text{model})\, p(\text{model})}{p(\text{data})}$$

(give higher prob to simpler models)