Linear Models for Regression

TTT4185 Machine Learning for Signal Processing

Giampiero Salvi

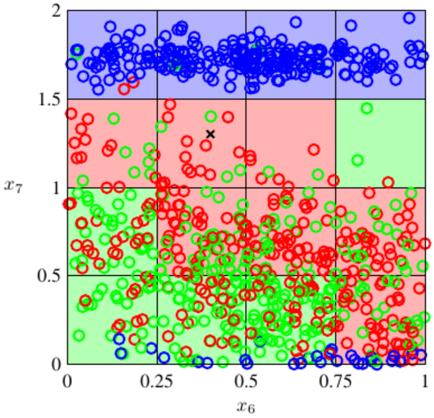
Department of Electronic Systems
NTNU

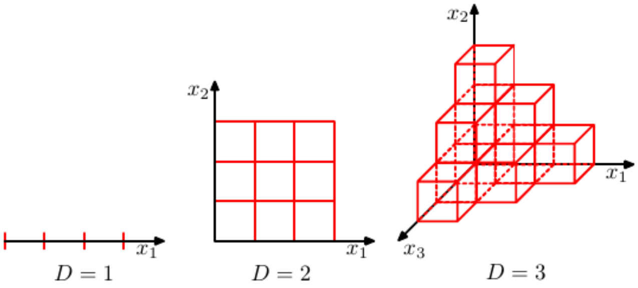
HT2021

Outline

Outline

Curse of dimensionality (non-parametric case)





Figures from Bishop

Curse of dimensionality (parametric case)

1-dimension $x \in \mathbb{R}$, third order polynomial

$$y(x, w) = w_0 + w_1 x + w_2 x^2 + w_3 x^3$$
$$(4 \text{ parameters})$$

D-dimension $\mathbf{x} = \{x_1, \ldots, x_D\} \in \mathbb{R}^D$, third order polynomial

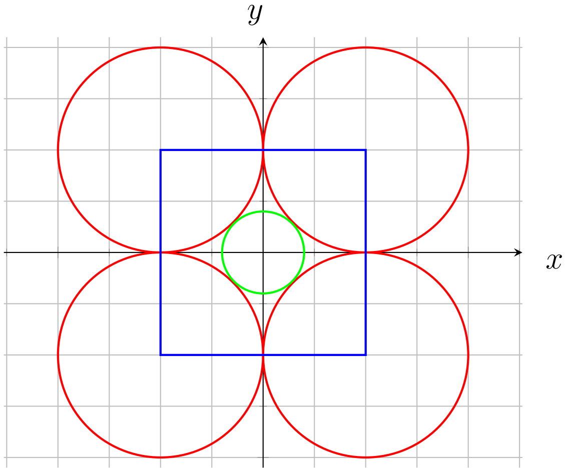
$$y(x, w) = w_0 + \sum_{i=1}^{D} w_i x_i + \sum_{i=1}^{D} \sum_{j=1}^{D} w_{ij} x_i x_j \sum_{i=1}^{D} \sum_{j=1}^{D} \sum_{k=1}^{D} w_{ijk} x_i x_j x_k$$
$$(1 + D + D^2 + D^3 \text{ parameters})$$

Example 28×28 images (MNIST): $D = 784$, $\#$ parameters = 482.505.745

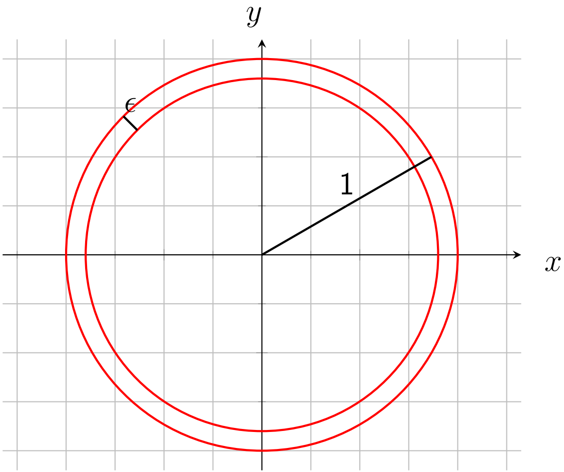
High dimensions and intuition

- radius of red circles $= 1$
- side of blue square $= 2$
- what is the radius of the green circle?
- what is the radius of the sphere in 3D?
- how about higher dimensions?

3Blue1Brown
`https://youtu.be/zwAD6dRSVyI`



High dimensions and intuition

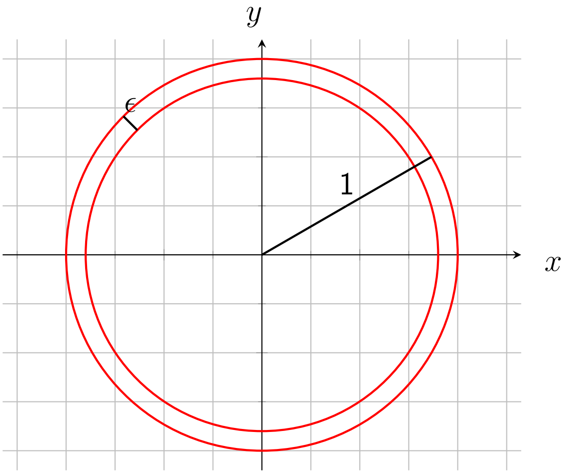


- What is ratio between the volume between the spheres and the volume of the large sphere?

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

- In D dimensions $V_D(r) = K_D r^D$
- Examples:
  - 2D: $K_2 = \pi$
  - 3D: $K_3 = \frac{4}{3}\pi$
  - ...

High dimensions and intuition



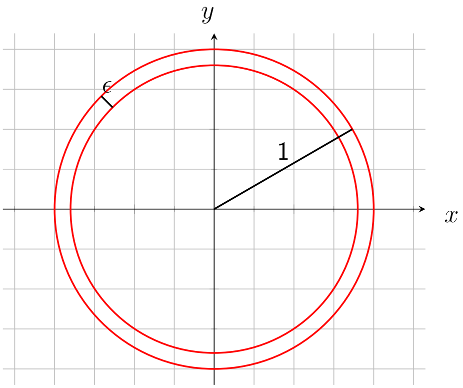
- What is ratio between the volume between the spheres and the volume of the large sphere?

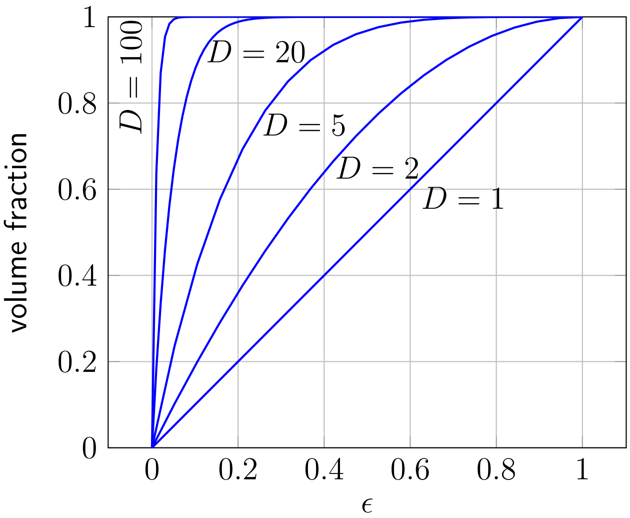
$$\frac{V_D(1) - V_D(1-\epsilon)}{V_D(1)} = \dots$$

- In D dimensions $V_D(r) = K_D r^D$

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

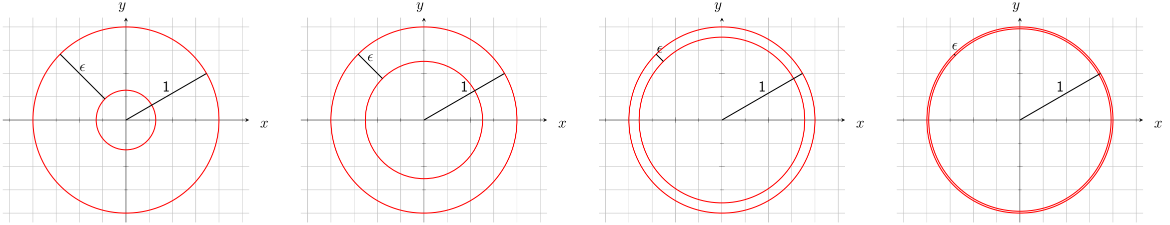
High dimensions and intuition





Where is 90% of the Volume?

$D = 2, \epsilon = 0.68$ $\quad$ $D = 5, \epsilon = 0.37$ $\quad$ $D = 20, \epsilon = 0.11$ $\quad$ $D = 100, \epsilon = 0.02$



Example: Euclidean Distance

Two points in D dimensions:

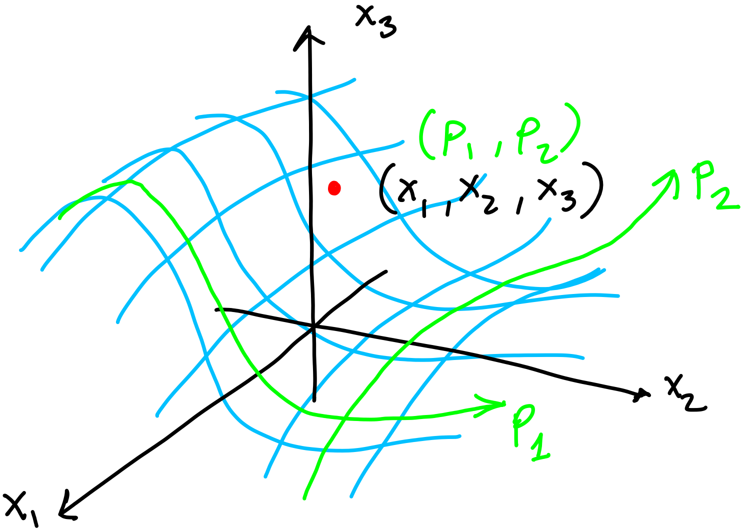
$$\mathbf{a} = (a_1, a_2, \dots, a_D)$$
$$\mathbf{b} = (b_1, b_2, \dots, b_D)$$

Euclidean square distance

$$d^2(\mathbf{a}, \mathbf{b}) = (a_1 - b_1)^2 + (a_2 - b_2)^2 + \cdots + (a_D - b_D)^2$$

If $D = 1000$ it is enough that just a few coordinates differ.

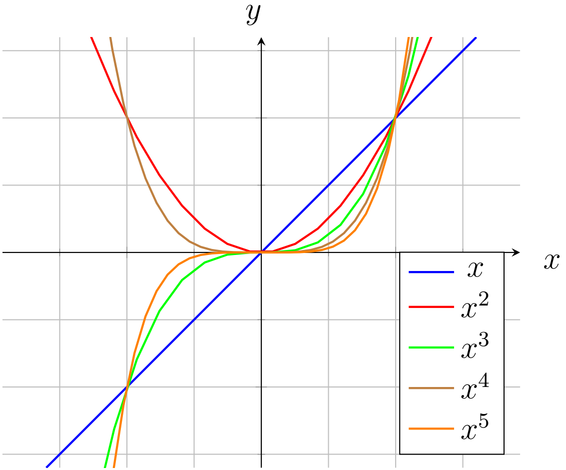
Manifolds and Dimensionality Reduction



Outline

Linear Regression with Polynomials

$$y(x, \mathbf{w}) \quad = \quad w_0 + w_1 x + \cdots + w_{M-1} x^{M-1}$$



Linear Regression with Basis Functions

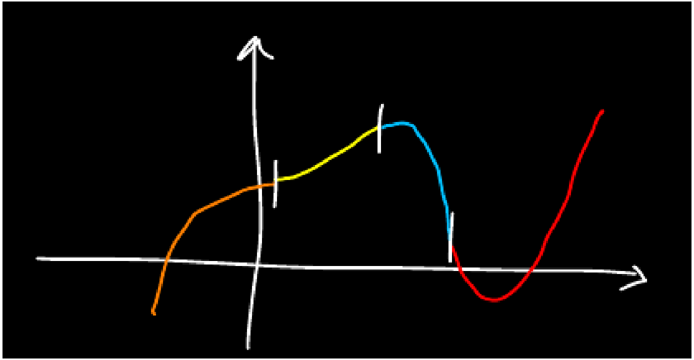
$$
\begin{aligned}
y(\mathbf{x}, \mathbf{w}) &= w_0 + w_1 \phi(\mathbf{x}) + \cdots + w_{M-1} \phi_{M-1}(\mathbf{x}) \\
&= \sum_{j=0}^{M-1} w_j \phi_j(\mathbf{x}) = \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})
\end{aligned}
$$

with:

$$
\begin{aligned}
\phi_j(\mathbf{x}) &: \mathbb{R}^D \rightarrow \mathbb{R} \\
\phi_0(\mathbf{x}) &= 1, \forall \mathbf{x} \\
\boldsymbol{\phi}(\mathbf{x}) &= [\phi_0(\mathbf{x}) \ldots \phi_{M-1}(\mathbf{x})]^T
\end{aligned}
$$

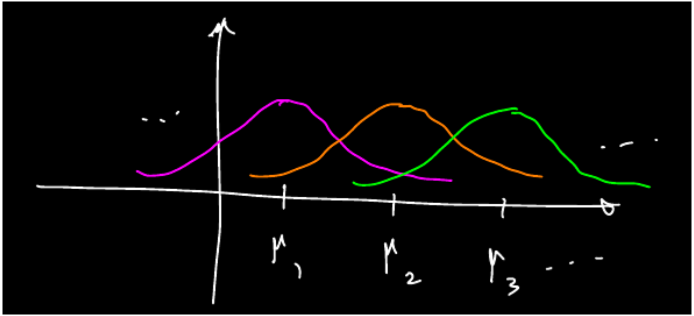
Example: Spline

- Piece-wise polynomial
- continuous up to first derivative



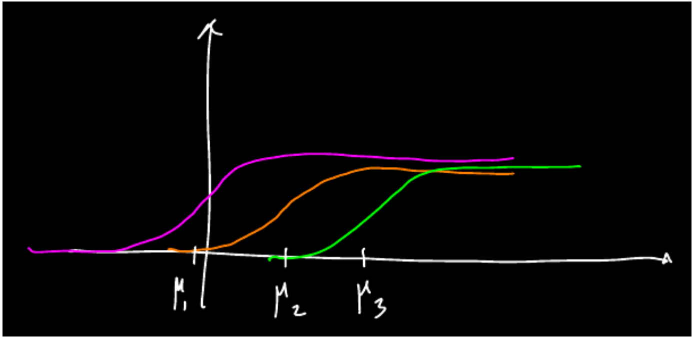
Example: Gaussian

$$\phi_j(x) \quad = \quad \exp\left\{-\frac{(x-\mu_j)^2}{2\sigma^2}\right\}$$

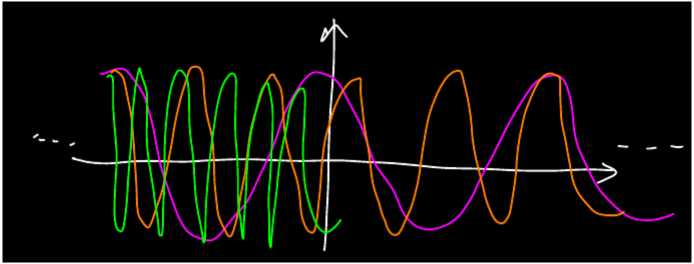


Example: Sigmoid

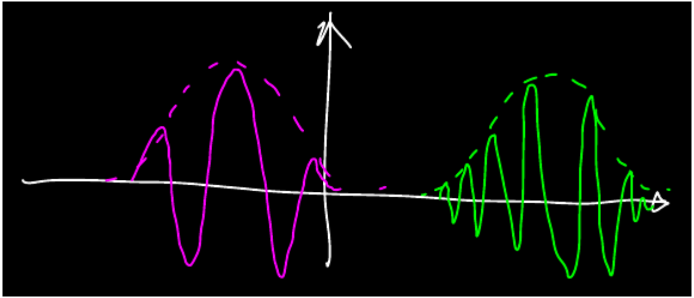
$$\phi_j(x) \quad = \quad \sigma\left(\frac{x-\mu_j}{s}\right), \text{ where } \sigma(a) = \frac{1}{1+\exp(-a)}$$



Example: Fourier



Example: Wavelets



Basis Functions: Likelihood

Model:

$$
\begin{aligned}
t &= y(\mathbf{x}, \mathbf{w}) + \epsilon \\
y(\mathbf{x}, \mathbf{w}) &= \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}) \\
p(t|\mathbf{x}, \mathbf{w}, \beta) &= \mathcal{N}(t|y(\mathbf{x}, \mathbf{w}), \beta^{-1})
\end{aligned}
$$

Data:

$$
\begin{aligned}
\mathbf{X} &= \{\mathbf{x}_1, \dots, \mathbf{x}_N\} \\
\mathbf{t} &= \{t_1, \dots, t_N\}
\end{aligned}
$$

Likelihood:

$$
p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n), \beta^{-1})
$$

Basis Functions: Maximum Likelihood Solution

$$\mathbf{w}_{\text{ML}} = \left(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\mathbf{t},$$

by defining the design matrix

$$\boldsymbol{\Phi} = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \dots & \phi_{M-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \dots & \phi_{M-1}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \dots & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix}$$

Basis Functions: Maximum Likelihood Solution

Equivalent to the linear regression solution in $\mathbf{x} \in \mathbb{R}^D$:

$$\mathbf{w}_{\text{ML}} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{t},$$

with

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1D} \\ x_{21} & x_{22} & \dots & x_{2D} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N1} & x_{N2} & \dots & x_{ND} \end{pmatrix}.$$

Basis Functions: Maximum Likelihood Solution

Equivalent to the linear regression solution in $\mathbf{x} \in \mathbb{R}^D$:

$$\mathbf{w}_{\text{ML}} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{t},$$

with

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1D} \\ x_{21} & x_{22} & \dots & x_{2D} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N1} & x_{N2} & \dots & x_{ND} \end{pmatrix}.$$

The basis functions $\phi_j(\mathbf{x}_N)$ act as feature extraction!

Basis Functions

- equivalent to linear models using $\mathbf{\Phi}$ instead of $\mathbf{X}$
- all other results hold:
  - overfitting of ML
  - regularization (MAP)
  - Bayesian models

Outline

Bias/Variance Decomposition

- Maximum Likelihood (least squares) leads to overfitting
- limiting the complexity of the model risks to miss trends in data
- regularization helps, but we need to find value for λ

Decision theory

Under L^2 loss, best decision is conditional expectation

$$h(\mathbf{x}) \quad = \quad \mathbb{E}[t|\mathbf{x}] = \int t \; p(t|\mathbf{x})dt,$$

where $p(t|\mathbf{x})$ is the true (unknown) distribution

Expected Loss (theoretical distribution)

If we predict the answer with $y(\mathbf{x})$, the expected (square) loss is:

$$\begin{aligned}\mathbb{E}[L] &= \iint L(t, y(\mathbf{x}))p(\mathbf{x}, t)d\mathbf{x}dt = \\ &= \iint \{y(\mathbf{x}) - t\}^2 p(\mathbf{x}, t)d\mathbf{x}dt \quad \text{(square loss)}\end{aligned}$$

We can compare this to the theoretically optimal estimation $h(\mathbf{x})$

Expected Loss (theoretical distribution)

$$\begin{aligned}\mathbb{E}[L] &= \ldots \\ &= \int \{y(\mathbf{x}) - h(\mathbf{x})\}^2 p(\mathbf{x}) d\mathbf{x} + \quad \leftarrow \text{ sub-optimal inference} \\ &+ \iint \{h(x) - t\}^2 p(\mathbf{x}, t) d\mathbf{x} dt \quad \leftarrow \text{ intrinsic noise}\end{aligned}$$

Expected Loss from Data

- we do not know $p(\mathbf{x}, t)$
- we imagine we have many data sets drawn from $p(\mathbf{x}, t)$
- for every data set $\mathcal{D}$ we obtain:
  - a model $y(\mathbf{x}, \mathcal{D})$
  - an expected loss $\mathbb{E}_{\mathcal{D}}[L]$
- then we can average over data sets.

Bias and Variance (single input value)

For a single value of $\mathbf{x}$

$$
\begin{aligned}
\mathbb{E}_{\mathcal{D}}\left[\{y(\mathbf{x}) - h(\mathbf{x})\}^2\right] &= \dots \\
&= \{\mathbb{E}_{\mathcal{D}}[y(\mathbf{x}, \mathcal{D})] - h(\mathbf{x})\}^2 + && \text{(bias)}^2 \\
&\quad + \mathbb{E}_{\mathcal{D}}\left[\{y(\mathbf{x}, \mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x}, \mathcal{D})]\}^2\right] && \text{variance}
\end{aligned}
$$

Bias and Variance (general case)

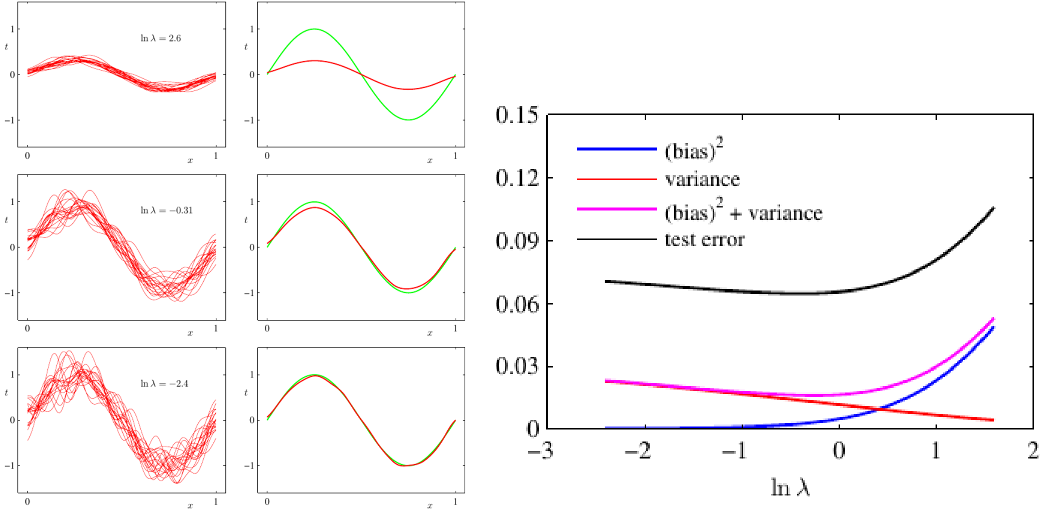
Integrating over all possible values of $\mathbf{x}$:

$$\text{expected loss} = (\text{bias})^2 + \text{variance} + \text{noise}$$

Where:

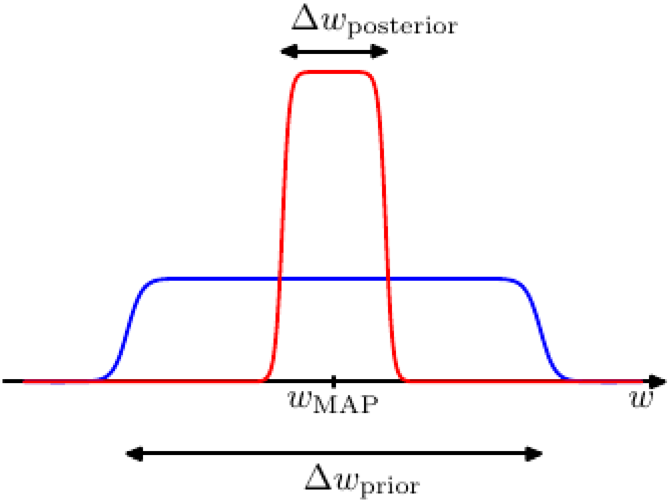
$$\begin{aligned}
(\text{bias})^2 &= \int \{\mathbb{E}_{\mathcal{D}}[y(\mathbf{x},\mathcal{D})] - h(\mathbf{x})\}^2 p(\mathbf{x}) d\mathbf{x} \\
\text{variance} &= \int \mathbb{E}_{\mathcal{D}}\left[\{y(\mathbf{x},\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x},\mathcal{D})]\}^2\right] p(\mathbf{x}) d\mathbf{x} \\
(\text{noise}) &= \int \{h(\mathbf{x}) - t\}^2 p(\mathbf{x},t) d\mathbf{x} dt
\end{aligned}$$

Bias/Variance Example

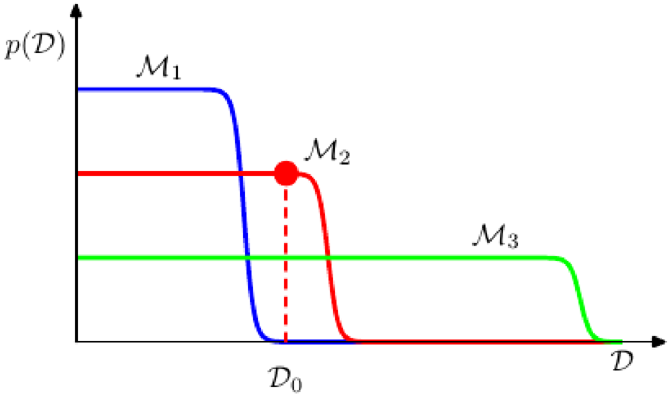


Outline

Bayesian Model Evidence



Bayesian Model Selection



Limitation of Linear Models

- basis functions $\phi_J(\mathbf{x})$ are fixed (not trained)
- the number of basis functions grow with dimensionality of input $\mathbf{x}$

Solution: exploit manifold

- dimesionality reduction methods
- support vector machines
- neural networks